Politecnico di Milano
Dipartimento di Elettronica, Informazione e Bioingegneria
Doctoral Programme In Information Engineering

# Methodologies and Tools for Games with a Purpose design and Gamified Applications

Doctoral Dissertation of:

**Luca Galli**

Supervisor:

**Prof. Piero Fraternali**

Tutor:

**Prof. Pier Luca Lanzi**

The Chair of the Doctoral Program:

**Prof. Carlo Fiorini**

2012-2014 – XXVII January

# Acknowledgments

This work was carried out under the supervision of Prof. Piero Fraternali.
I sincerely thank him for his guidance, the precious support, the trust and the opportunities he has given me while working in our research group on several European Projects[1]. Many thanks also go to Prof. Pier Luca Lanzi, that suggested me to follow the PhD program at Politecnico and supported me in times of need.

I would also like to thank all the colleagues that worked with me during these years, especially Chiara, with which I shared failures, successes and joyful moments, Alessandro for his precious suggestions and insights, Andrea and Riccardo for their jokes, Giorgia and Carlo, that turned from students I followed into invaluable collaborators, Roman, Ilio, Eleonora, Luca.

A special mention goes to Lev, a dear friend that I would have never met if I would have not undertaken my PhD path. To my parents, that have always supported me with kindness and excitement; without you I would have never accomplished such results. To my grandparents, that raised me with love, teaching me how to be curious and love sciences. To my brother, that was always trying to distract me from my duties with his jokes and strange ideas.

Finally, I wish to thank Prof. Georgios Yannakakis, that reviewed this work, for the time spent in reading this dissertation and for providing useful advices on how to improve it.

[1]The research leading to these results has been partly supported by the European Communitys Seventh Framework Programme FP7//2007-2013 - The CUbRIK project. For further information see `http://www.cubrikproject.eu/`.

# Abstract

Human computation is a research area that focuses on exploiting human intelligence to solve computational problems that are beyond the capacity of existing Artificial Intelligence (AI) algorithms.

The growth of the Web and Social Networks provides a massive amount of persons that can be leveraged to perform complex tasks, but a fundamental issue in exploiting the contribution of crowds is how to engage the potential users for the specified purposes and how to ensure the quality of their contribution.

To overcome the problem, a set of approaches have been developed; Games with a Purpose (GWAPs) are digital games in which the players' actions in the game contribute to a real-world purpose outside of the game, whether it be predicting protein structures or providing labels for images.

The standard way to accomplish the same type of work is to "crowdsource" the work directly using a service like Amazon Mechanical Turk in which contributors are paid as workers. To address the lack of extrinsic motivation that plagues traditional human computation platforms, GWAP provide intrinsic motivation in the form of entertainment. Many GWAP have been developed since the release of the first instance, the ESP Game in 2003.

But not all GWAP seem to have lived up to the initial hype of transforming millions of hours typically poured into traditional games into useful and productive work. The problem that GWAP have faced since their inception is related to the fact that the very fundamental mechanisms on which they rely on, to guarantee the quality of the submitted results, have been considered as "Game Mechanics" while in reality they are simply validation mechanisms.

For this reason, even the most famous GWAP were centered on experiences that aimed at maximizing the throughput of high quality submitted content instead of focusing on the entertainment dimension typical of other digital games, producing applications that were perceived as non games by their users.

As it happened with GWAP, gamification, the process of using game design techniques and game mechanics to enhance traditional applications, has been able to accomplish significative results but also catastrophic failures. Once again, this phenomenon has to be attributed to poor design due to the lack of guidelines and best practices to support the development.

The main reason is the inherent difficulty of the design of both GWAP and gamified applications, which resides in the tradeoff between purposiveness and playfulness: in a traditional application, the improper insertion of gaming elements may result artificial and thus not produce the desired engagement effects, while on the contrary spoiling the users productivity, symmetrically in a GWAP the task to be solved may mismatch with the game mechanics, thus decreasing the playability of the game and failing to attract people and engage them in the execution of the task.

Another common challenge of human computation systems is data reliability. Humans are expected to be unreliable, especially in ludic environments where a playful interaction with the system to test its borders is expected. Therefore, players may generate false data either on purpose or for other reasons. Different strategies have evolved to deal with this issue but they are typically tailored just to the particular task they have been applied to. As human computation tasks are by definition not efficiently solvable by an algorithm, it is necessary to find new means to handle this challenge.

The lack of established GWAP design paradigms, the difficulties of player engagement and retention and the issues of choosing or defining the right validation techniques in order to obtain meaningful results are limiting the capabilities that these systems may offer.

The proposed framework investigates the design of game mechanics and motivation techniques in games in order to solve human computation tasks by providing a set of tools that will be used to ease the development of interactive media applications that have to be integrated within media refinement tasks fulfilled by players.

The work has also, dually, investigated the methodologies and approaches for gamification, that is the injection of game-like features in traditional applications (e.g. software development, customer relationship management) to improve key performance indicators.

# Table of Contents

# List of Figures

# List of Tables

# Part I

# Introduction

# Chapter 1

# Introduction

In the last years, the interest for the "*-sourcing" trend and its derivatives has been increasing at a phenomenal rate [1]. It all started at the beginning of the 1990s, with the introduction of "outsourcing", the process of of subcontracting a process, such as product design or manufacturing, to a third-party company [2]. The benefits of the approach were clear: same or better quality, less effort, less money spent. With time, the trend evolved and at the beginning of the 2000s, the new best practice was to outsource processes done at a company in one country to the same or another company in another, different country; it became "offshoring". Usually the wage and working conditions of the target countries for this kind of operations allowed, considering the IT field, same quality software at huge discounts; India, for instance, was the first country to benefit from the offshoring trend, given its large pool of English speaking people and technically proficient manpower.
The final step in the evolution of these business processes is now "crowdsourcing", the act of taking a task traditionally performed by a designated agent (such as an employee or a contractor) and outsourcing it by making an open call to an undefined but large group of people. Several studies confirm that the powers of the crowd allow organizations to accomplish tasks that were once feasible to just a specialized few [3][4].

It has to be kept in mind though that crowdsourcing is not just a natural improvement over the previous business processes, but is a complex phenomenon that has increased its importance due to several developments.

Technological advancements (from product design software to digital video cameras) are breaking down the cost barriers that once separated amateurs from professionals, and have given birth to a new class of contributors, non professional individuals who are knowledgeable, educated and committed, able to perform work in what were previously considered professional standards.

Social media have gathered in the years a solid user base of million of users that have integrated in their daily routines practices like creating or sharing content on the Web, in textual form (e.g. forum or social network posts, blog articles, comments or discussions) or as multimedia content (e.g. audio, photos, videos), as a job or in their spare time.

Recent reports show that Crowdsourcing is not just a mere trend, but a growing field with a big market. Blinder [5] argues that about 20% of current American jobs could be sent down a wire. These include tasks like programming, accounting, marketing, and even machine operators. Recent evidence for crowd work [6] in particular suggests that its volume will be roughly 454 billion dollars per year. That is 91 billion hours per year, employing about 45 million full-time workers.

Competences, Affordances and the success of the Crowdsourcing approach have ultimately influenced the raise of a new research area that studies the process of channelling the vast internet population to perform tasks or provide data towards solving difficult problems that no known efficient computer algorithms can yet solve: Human Computation [7][8].

Crowdsourcing and Human Computation may appear the same at first glance but there is a subtle difference among them: while crowdsourcing replaces traditional human workers with member of the public, human computation replaces *computers* with humans. Just as distributed computing projects like UC Berkeleys SETI at home [9] have tapped the unused processing power of millions of individual computers, so distributed labour networks are using the Internet to exploit the spare processing power of millions of human brains.

The Human Computation paradigm itself is not a novelty: the approach was taken for astronomical and other complex calculations. Perhaps the first example of organized human computing was by the Frenchman Alexis Claude Clairaut, when he divided the computation to determine timing of the return of Halley's Comet with two colleagues, Joseph Lalande and Nicole-Reine Lepaute [10]. The term "computer", in use from the early 17th century, meant "one who computes": a person performing mathematical calculations, before electronic computers became commercially available.

The interest for the approach was initially raised back again thanks to the rapid growth of the Web and initiatives like "the Open Mind Initiative"[11] a web-based collaborative framework for collecting large knowledge bases from non-expert contributors. Computer scientists that were trying to emulate human abilities were now seeking once more for human contributors to solve "AI hard"problems. [12] This idea has been then further refined by other researchers [13], with the introduction of social Web and collective intelligence [14].

However, many tasks that are trivial for humans continue to challenge even the most sophisticated computer algorithms. To solve most of these tasks, high level perceptual and abstraction capabilities along with the ability for complex reasoning when facing unexpected problems are needed. Image annotation tasks are a typical example: annotating images requires the ability of recognizing objects and scenes in photos, which clearly is typical of humans. Machines, on the other hand, do not have the abstraction capabilities to discriminate between objects and figures, and thus, if provided with a large dictionary of shapes, they can only state if two objects are similar by analyzing key points (i.e. significant points in the image), that by their very nature are susceptible to variance in lighting conditions and non-affine transformations.

Although participating in social networks and consulting blogs are the most prominent activities performed on the Internet, a Nielsen research [15] reveals that another rather surprising activity has gained a lot of momentum recently: Gaming. In the United States Online games overtook personal email to become the second most heavily used activity behind social networks - accounting for 10 percent of all U.S. Internet time, with more than 407 millions of hours spent each month. The impact of digital games in the global game market is also all but negligible, with an estimated revenue of $81.4bn in 2014, up 7.8% compared to 2013 as stated by a Newzoo research [16].

Luis Von Ahn has been the first to understand the possibility offered by conciliating the natural desire for players to be entertained with the necessity of gathering volunteers in order to solve human computation tasks, defining a brand new subgenre of "Serious Games "by creating the first Game with a Purpose (GWAP), the ESP Game [17], which represent the first seamless integration of game play and computation. A serious game is a game designed for a primary purpose other than pure entertainment. A Game with a Purpose is a Serious Game, in which players generate useful data as a by-product of play.

The ESP Game is a GWAP in which people provide meaningful, accurate labels for images on the Web as a side effect of playing the game; assigning to an image of a man and a dog textual annotations such as “dog”, “man”and “pet”is an example of the computational tasks solved by the game. These annotations have then been used to improve Web-based image search, which typically involves noisy information (such as filenames and adjacent text). Rather than using computer-vision techniques that do not work well enough, the ESP Game motivates its players to do the work of labelling images disguising the task beneath a ludic activity.

Since their creation GWAP have been able to solve the most diverse problems, ranging from combinatorial optimization tasks such as in Foldit [18] and Plummings [19], to contextual reasoning for audio or image annotation such as in Tagatune [20] and Verbosity [21], to aesthetic judgment problems as in the collaborative Tree Generator of [22], to the interaction with objects and elements in the real world, such as in City Explorer [23]. But even if a game is properly designed, sometimes intrinsic motivation for the players does not suffice and increasing the participation of the players while mantaining a solid user base becomes a not so trivial task. This problem is commonplace in several other fields, and not limited just to digital games: it has been estimated that US businesses lose between $450 and $550 billion per year due to disengagement, which manifests itself in turnover rates, poor productivity, and below average work [24]

One of the most recent manifestations of the impact of game concepts in software development to boost users’ involvement is called “gamification ”; this term refers to the process of using game design techniques and game mechanics to enhance traditional applications, to drive behaviors, develop skills, or simply engage and retain people [25]. The Deloitte consulting firm cited gamification as one of its Top 10 Technology trends for 2014 [26], stating: “Serious gaming simulations and game mechanics such as leaderboards, achievements and skill-based learning are becoming embedded in day-to-day business processes, driving adoption, performance and engagement”; a report by M2 Research confirms the economic importance of gamification and foresees that the associated market will reach $2.8 billion in 2016 [27]. The reason for such interests resides in the role of gamification techniques in exploiting basic psychological urges, such as competition, goal-setting, and status/reputation seeking that are used by websites and social media outlets to vastly increase user engagement, encourage specific behaviors, and even provide hints as to what actions are possible within an application [28].

## 1.1 Problem Statement

Games with a purpose (GWAP) are digital games in which the players' actions in the game contribute to a real-world purpose outside of the game, whether it be predicting protein structures or providing labels for images. The standard way to accomplish the same type of work is to "crowdsource"the work directly using a service like Mechanical Turk [29]. One outstanding need of crowdsourcing platforms and GWAP derive from the fact that Human Sensor data, Internet of Things and Social media are producing massive amounts of content that can be used for collective intelligence applications in many sectors, ranging from multimedia content annotation to environmental monitoring. Human computation architectures are a mean to solve the problem by offering the best of two approaches: while machines are able to manage content acquisition at a scale, handling discretization and orchestrating distribution of work, humans complement machine learning algorithms in content analysis.
To address the lack of extrinsic motivation that plagues traditional human computation platforms, GWAP provide intrinsic motivation in the form of entertainment compared to traditional incentives such as payment. The more fun and addicting the game is the more work players will be willing to contribute for free.

Many GWAP have been developed since the release of the ESP Game [17] in 2003. But not all GWAP seem to have lived up to the initial hype of transforming millions of hours typically poured into traditional games into useful and productive work [30]. Even after the initial success of the ESP Game, the whole original GWAP suite was shut down in 2011 [31].
The problem that GWAP as faced since their inception is related to the fact that the very fundamental mechanisms on which they rely on, to guarantee the quality of the submitted results, have been considered as "Game Mechanics"while, in reality, they are simply validation mechanisms. For this reason,

even the most famous GWAP were centered on experiences that aimed at maximizing the throughput of high quality submitted content instead of focusing on the entertainment dimension typical of other digital games. Indeed, some works have confirmed that users were perceiving GWAP like ESP Game as "non-games"and this is a consequence of the lack of structured development and design guidelines borrowed from traditional game design literature [30].

The main reason is the inherent difficulty of the design of both GWAPs and gamified applications, which resides in the trade-off between purposiveness and playfulness: in a traditional application, the improper insertion of gaming elements may result artificial and thus not produce the desired engagement effects, while on the contrary spoiling the user's productivity; symmetrically, in a GWAP the task to be solved may mismatch with the game mechanics, thus decreasing the "playability"of the game and failing to attract people and engage them in the execution of the task.

Another common challenge of human computation systems is data reliability. Humans are expected to be unreliable, especially in ludic environments where a playful interaction with the system to test its borders is expected. Therefore, players may generate false data either on purpose or for other reasons. Different strategies have evolved to deal with this issue, but they are typically tailored just to the particular task they have been applied to. As human computation tasks are by definition not efficiently solvable by an algorithm, it is necessary to find new means to handle this challenge.

As it has happened with GWAP, Gamification has been able to accomplish considerable results but also catastrophic failures. Once again, this phenomenon has to be attributed to poor design due to the lack of guidelines and best practices to support the development and the mantainance of a gamification platform since its inception.

**Problem Statement:**
**The problem addressed in the thesis is the lack of methodological guidelines, software tools and reusable architecture patterns and components for the effective development of Games with a Purpose and Gamified Business Applications, which could turn the ad hoc processes employed for developing these solutions into a systematic one.**

## 1.2 Research Objectives

The viewpoint of this thesis is that gamification and GWAP design are two sides of the same coin. Both aim at designing an engaging user experience for a problem solving purpose; on the one hand GWAP aims at solving a computational task for which no system was previously designed by means of an entertaining activity; on the other hand a gamified application focuses on maximizing the performance of users in an application having a well defined goal, exploiting elements of game mechanics.

The purposes of this work, in order to tackle with the Problem Statement defined beforehand, are thus: investigating the design and development process of GWAP and gamified applications, defining a conceptual model and an architecture able to cater for all the aspects relevant to human computation platforms and gaming scenarios and providing methodological tools for matching computational tasks and business objectives to be achieved with the best suited game mechanics.

To achieve the desired results, the defined objectives are:

- **Obj 1.** To Analyze and evaluate previous research work and approaches related to GWAP and gamified application design. This objective has to be accomplished by exploring three main areas of concern:
  - **Obj 1.1** Human Computation and Crowdsourcing
  - **Obj 1.2** Serious Games and GWAP
  - **Obj 1.3** Gamification
- **Obj 2.** To Define a comprehensive model and a software architecture

able to cater for all the relevant characteristics of a social user and performer in Human Computation, GWAP and Gamified applications

- **Obj 3.** To Define a development process for the design and implementation of both GWAP and gamified applications.
- **Obj 4.** To Define proper game mechanics that are necessary for the development of a GWAP or a gamified application and how this can be mapped to the problem at hand to be solved.
- **Obj 5.** To Detail techniques for Evaluating, Comparing and Aggregating the results of GWAP and Gamified applications and proposing novel approaches for improving the quality of human contributions in both contexts.
- **Obj 6.** To Validate the work qualitatively and quantitatively in two real case scenarios, where the development lifecycle and the reference architecture has been used.
  - **Obj 6.1** Enterprise Gamification Platform: Webratio Headquarters
  - **Obj 6.2** GWAP for Image Segmentation: Sketchness

## 1.3 Outline

The rest of the work is organized as follows.

**Part 2: Background** defines the context of the work in terms of Human Computation, Games with a Purpose and Gamification, to reach **Obj 1.**. For each area, the most meaningful applications are presented. In particular:

- Chapter 2, **Human Computation**, briefly presents the evolution of Human Computation, its main applications and dimensions.
- Chapter 3, **Games with a Purpose**, describes the constitutive elements of any game, reports the main differences between Serious Games and Games with a Purpose and provides an extensive survey of the existing and past GWAP and the context in which they have been applied.
- Chapter 4, **Gamification**, defines the term “Gamification”and why it derives from the term “Game”, describes how this approach motivates users to act in a system and details the mechanics inherited from tradi-

tional game design. Finally, successful stories of companies able to apply gamification in a meaningful and successful way are provided.

**Part 3: Design** discusses the development processes, the components and the mechanics of both GWAP and Gamified Applications and presents a model and the software architecture able to support the implementation of such applications. In particular:

- Chapter 5, **GWAP Design** draws from the literature a development process for traditional games that is used as a starting point for the definition of the GWAP development process. The concept of human computation task is described and possible tasks for multimedia content enrichment detailed. Traditional game mechanics that could be applied to the annotation of multimedia content are described and matched against suitable tasks. Possible validation mechanics used to improve the quality of the submitted annotations are provided, along with a general purpose aggregation algorithm that could be used for computing a unique result out of several contributions. It covers **Obj 3**,**Obj 4** and **Obj 5**.

- Chapter 6, **Gamified Applications Design** provides a distinction between game development and gamification development, describing the process for the development of gamified applications. The mechanics that are used in gamified applications are defined and evaluation metrics used to measure the effectiveness of gamification elements in a particular context provided. It covers **Obj 3**,**Obj 4** and **Obj 5**.

- Chapter 7, **Human Computation Architecture**, describes the design of a unified data model for representing the relevant aspects of users, their social ties and activities, the communities where they are active, the actions they can contribute, and the contents which are the object of such actions; such a model defines the foundations over which any human computation platform can be built upon and include also tailored structures for managing GWAP and Gamified Applications. It covers most of the aspects needed to fulfill **Obj 2.**.

**Part 4: Case Studies** presents two detailed case studies that make use of the development process, the mechanics and the data structures defined in the previous chapters, to reach **Obj 6**.In particular:

- Chapter 8, **Gamified Application: Webratio Headquarters**, presents an Enterprise gamification platform that has been used to increase the participation level for existing users in order to achieve desired business objectives that were not met in everyday usage. It fulfills **Obj 6.1**.

- Chapter 9, **Game with a Purpose: Sketchness**, presents an open source GWAP that was developed for a hybrid fashion trend analysis application to gather human contributors for image segmentation tasks. It fulfills **Obj 6.2**.

Finally in **Part 5** conclusions are drawn and future work directions are presented.

# Part II

# Background

# Chapter 2

# Human Computation

Recent years growth of the Web as a content production and social interaction platform stimulated the increase of the interest in Human Computation and Crowdsourcing areas. This has allowed to leverage on the ability of people over the Internet to perform tasks.

This chapter briefly presents Human Computation evolution, its main applications and dimensions. It's important to understand what is the role of Human Computation in the today's society and development of Web.

## 2.1 Introduction to Human Computation

The Web has evolved from a publishing platform, where the interaction of users was prevalently limited to the publication of personal content or to the access of content created by others, to a collaborative and social tool, where users spend time online for sharing information and opinions, cooperating in the execution of tasks, playing games, and participating to the collective life of communities. In year 2011, according to the *US Digital Consumer Report* by Nielsen [32], social network/blog usage and gaming are respectively the first and second busiest online activity performed in the US by fixed network users, surpassing the background for the diffusion of a new computation paradigm, called Human Computation [33], applied in business, entertainment and science, where the online time spent by users is harnessed to help in the cooperative solution of tasks. According to the definition of **Luis von Ahn**, a pioneer in the systematic use of people in online problem solving, human computation is a paradigm for utilizing human processing power to solve problems that computers cannot yet solve [33].

This definition is further refined by **Alexander J. Quinn** in "Human computation: a survey and taxonomy of a growing field"
[8], who distills several recent definitions from the literature into two distinctive features of a human computation system:

- the problems fit the general paradigm of computation, and as such might someday be solvable by computers;
- the human participation is directed by the computational system or process.

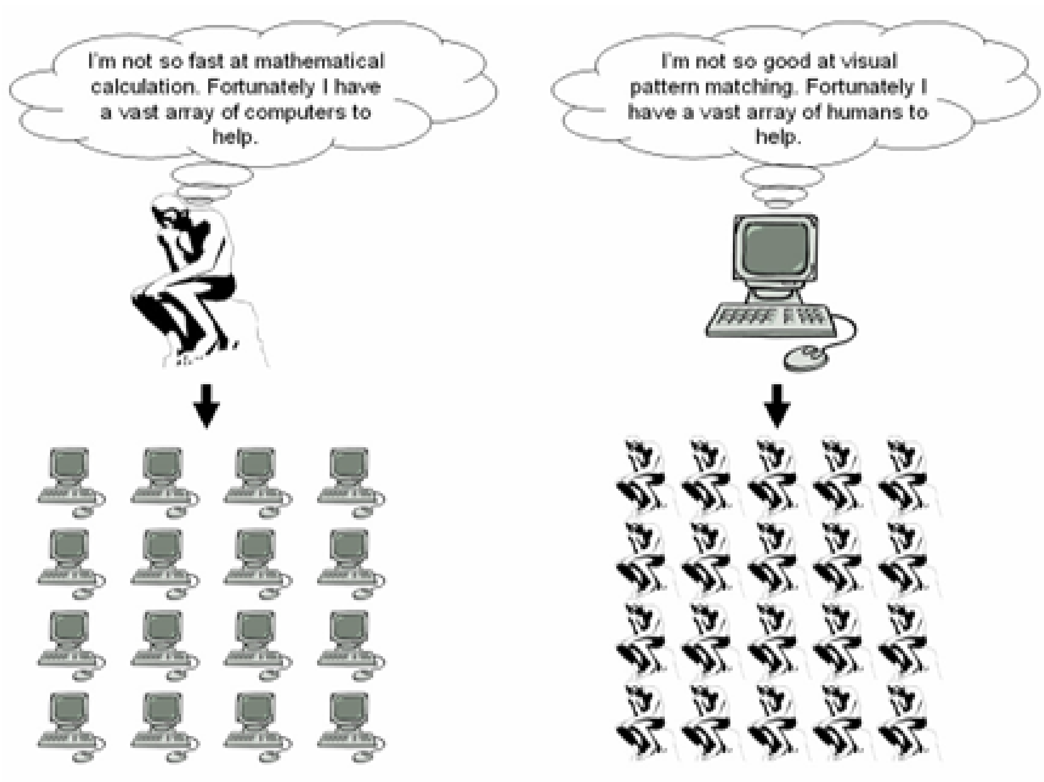



Figure 2.1: The interdependence between computers and humans.

The common baseline of the approaches that exploit humans in computing is the intelligent partition of functionality between machines and human beings: networked machines are used for task splitting, coordination, communication, and result collection; humans participate with their intuition, decision making power, and social links [34].

A classical example of a human computation scenario is content analysis for multimedia search applications. In this domain, the goal is automatically classifying non-textual assets, audio, images, video, to enable information retrieval and similarity search, for example, finding songs similar to a tune whistled by the user or images with content resembling a given picture. Recognizing the meaning of aural and visual content is one of the skills where humans outperform machines, matured by living beings in millions of years of evolution. It is now commonly recognized that multimedia content analysis can benefit from large scale classification performed by humans; applications like Google Labeler and the system proposed by **X. Hu, M. Stalnacke**, **T. B. Minde**, **R. Carlsson** and **S. Larsson** [35] submit images from a large collection to human users for receiving feedback about their content and position, which can be integrated with machine-based feature extraction algorithms.

The founding principle of Human Computation, the structured collaboration of humans and machines in problem solving, is as old as computer science. The widespread use of the term can be traced back to the seminal the work of Luis Von Ahn on online games as a general incentive mechanism for encouraging human participation to problem solving.

Human Computation, due to its goal of harmonizing the work of human and computer processors, is inherently a multi-disciplinary topic. Figure highlights the most relevant areas that contribute to shaping Human Computation as a research focus.

Computer science contributes system development techniques and architectures for designing and deploying distributed systems, possibly implemented on top of heterogeneous platforms (e.g., crowdsourcing, social networks, or gaming platforms) and accessible through application programming interfaces and with multiple access devices. Besides the architectural side, human-computer interaction issues are also relevant, with a specific focus on the modeling of the user's behavior, on the design of high quality interfaces for the execution of tasks, and on the adaptation of the user interface to different access devices.

Data and knowledge management bring to human computation the ability of extracting meaning from the trails of human activities, as necessary, e.g., when the collaboration of humans is sought on such unstructured platforms as open social networks or blogs.

The organization of collective work also draws from workflow management

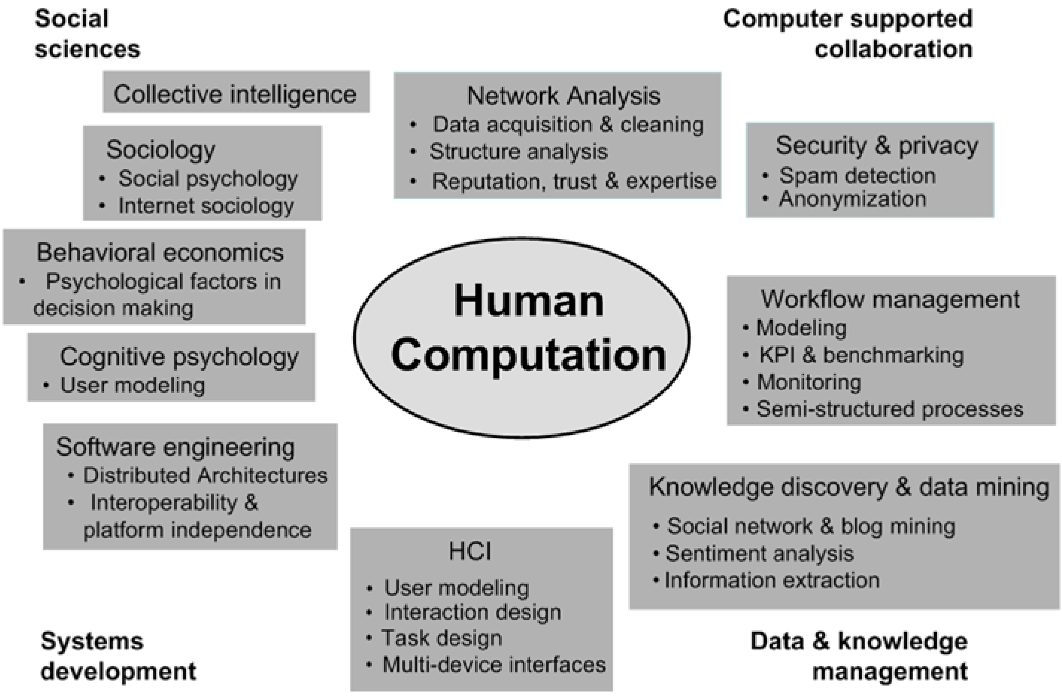



Figure 2.2: Most relevant areas that contribute to shaping Human Computation as a research focus.

techniques, which investigate abstract models for representing processes, key performance indicators and quality monitoring methods, and also deal with the case of partially specified processes, where activities are conducted in part freely and in part according to some organizational constraints.

When the contribution of people is harnessed on a massive scale and on the public internet, privacy and security research becomes relevant too. On one side, malicious behavior detection is required, to avert individual or collective attempts at cheating with human computation applications (e.g., randomly performing tasks in a crowdsourcing platform in order to gain money). On the other side, it may be necessary to gather and process public data, while preserving the anonymity of the users who contributed them, as, e.g., in online large scale market analysis.

When human computation applications build on the social connections of people, e.g., for spreading a task or a game virally within a community, network analysis methods can be employed, to understand the structure of the social ties, identify or predict the most influential members of the community, find experts on a topic, and minimize the time for completing a task. An important application of this discipline is also trust computation, which supports the

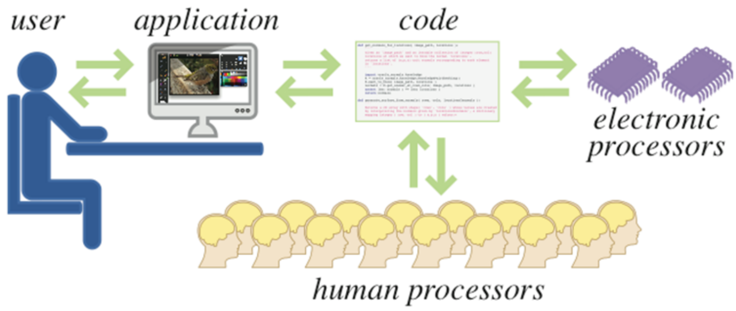



Figure 2.3: Human Computation Cycle.

selection of human performers according to their expected reliability, based on the role and activities they play in the community.

Social sciences complement the view of computer science with an insight into the individual and collective cognitive and decision processes that drive the human side of human computation applications. Cognitive psychology studies the mental processes that influence human behavior: perception, memory, thought, speech, and problem solving, and supports the investigation of the interactions between users and computer-based systems. Behavioral economics focuses on the specific aspect of human decision-making, a central problem in human computation applications; it investigates the cognitive mechanisms that may introduce bias and non-rationality systematically in the way in which individuals judge and make decisions, like, for example, the influence of prior knowledge, the change in preferences induced by irrelevant options, and the distorted perception of future probability based on past experience.

When human computation applications exploit group behavior, the study of group dynamics as conducted in psychology, sociology, political science, and anthropology can shed light on the processes that occur within a social group or between different social groups. The methods and theories of sociology have been applied to the user groups that are constituted with the mediation of computer and network architectures, the so called online or virtual communities. Studies on participation, equality, and cultural diversity can be relevant to the design or to the interpretation of the results of human computation applications.

Finally, human computation can be seen as a specific way of harnessing the power of the so-called collective intelligence, which arises when a large number of loosely connected individuals cooperate or compete and in doing so achieve some goal that may transcend the intention and the capacity of each individual. It has to be noted that the contribution of the single may be imprecise or even completely wrong, thus redundancy in the submitted contribution and aggregation techniques able to provide a unique solution out of the work of hundreds are necessary means to solve the inherent limitations of human computation.

## 2.2 Applications

Under the broad umbrella of Human Computation, several applications with different goals, architectures, and users can be recognized. In this Section, we illustrate some of the most paradigmatic examples.

In Section 2.2.1 we present the concept of Crowdsearching, a new paradigm for searching in the Web using the crowd as source of information, in Section 2.2.2 we define the notion of GWAP, providing some examples while in Section 2.2.3 we introduce the Human Sensing concept, describing also an application.

### 2.2.1 Crowdsourcing

Human Computation is related with other fields, such as crowdsourcing, Social Computing and Collective Intelligence as illustrated in Figure 2.4. The following paragraph is centered on Crowdsourcing but for better understanding, the other fields are also briefly introduced in order to outline similarities and differences.

**Crowdsourcing**: is "the outsourcing of work, traditionally performed by employees, to an undefined, generally large group of people in the form of an open call" [36]. So it does not involve computation directly like HC.

**Social computing**: "describes any type of computing application in which software serves as an intermediary or a focus for a social relation"[1]. In despite of its name, its purpose is not computing.

**Collective intelligence** is defined very broadly as "groups of individuals doing things collectively that seem intelligent".

When dealing with a human crowd the main issue is to engage users to perform tasks. A user can be motivated to perform a task due to its nature or

[1] Schuler, 1994

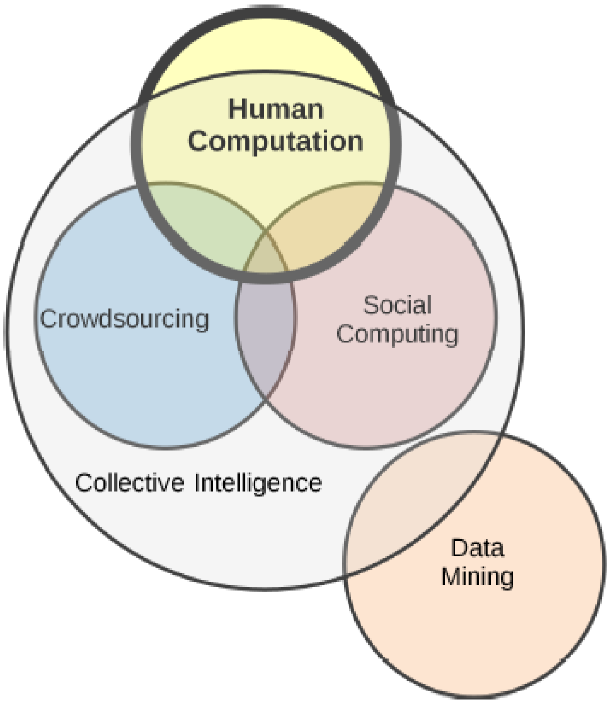



Figure 2.4: Human Computation relation with CrowdSourcing, Social Computing and Collective intelligence.

to the revenue he/she gets for performing such task. The most effective way for recruiting and motivating users is to give them money. For instance Amazon's Mechanical Turk [29] is an online tool for performing Human Intelligent Task (HIT) in exchange of money rewards.

CrowdSourcing is a relatively new term that is a part of the "sourcing" trend. CrowdSourcing came after Outsourcing and Offshoring, but among the three is possibly the one that could lead to higher savings, as shown in Figure 2.5.

With respect to Figure 2.5, the general terms that have been used are described as follows:

- **Outsourcing**: is the process of outsourcing the data center or outsourcing application development. The resulting product has generally the same or better quality while requiring less effort and less money.

- **Offshoring**: outsource processes done at a company in one country to the same or another company in another, different country; it allows the development of quality software at huge cost savings.

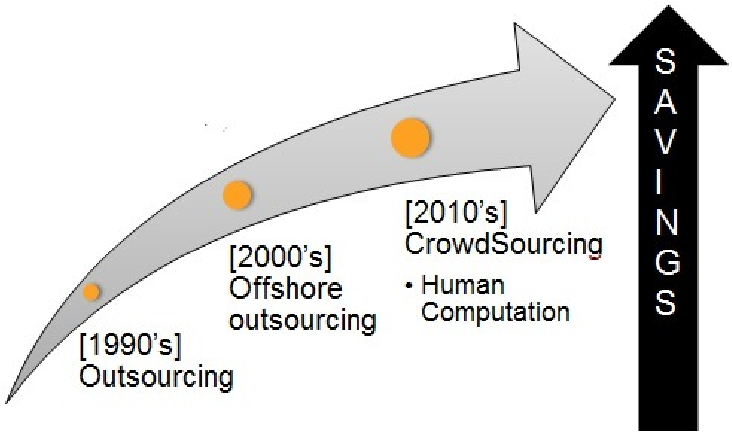



Figure 2.5: The development of Sourcing from Outsourcing to CrowdSourcing.

- **CrowdSourcing**: the act of taking a task traditionally performed by a designated agent (such as an employee or a contractor) and outsourcing it by making an open call to an undefined but large group of people. It its latest form, it employs common people's spare time to create content, solve computational problems or validating data through digital platforms able to orchestrate thousands of users at the same time.

A first form of crowdsourcing is found in vertical markets, where a community of professional or amateur contributors is called in for supplying specific products or services. An example is iStockPhoto[2], a content marketplace where photos made both by professionals and amateurs are sold at affordable prices. Other examples are found in graphic design, e.g., the 99designs[3] design contest web site where customers can post requirements for web page or logo creation and designers compete by submitting proposals; in fashion design, e.g., the *Threadless*[4] community and marketplace for T-shirt and garment design; and even in highly technical and specialized services, like collaborative innovation, e.g., *Innocentive*[5], where technical challenges for product innovation are addressed to a community of problem solvers.

A different form of crowdsourcing is provided by horizontal platforms, which broke the execution of micro-tasks in different domains, like speech and handwriting transcription, data collection and verification. A typical mi-

[2] `http://istockphoto.com`
[3] `http://99designs.com`
[4] `http://www.threadless.com`
[5] `http://www.innocentive.com`

Figure 2.6: Crowdsourcing main concept.

crotask brokerage platform has a Web interface that can be used by two kinds of people: work providers can enter in the system the specification of a piece of work they need (e.g., collecting addresses of businesses, classifying products by category, geo-referencing location names, etc); work performers can enrol, declare their skills, and take up and perform a piece of work. The application manages the work life cycle: performer assignment, time and price negotiation, result submission and verification, and payment. In some cases, the application is also able to split complex tasks into micro-tasks that can be assigned independently [37], e.g., breaking a complex form into sub-forms that can be filled by different workers. In addition to the web interface, some platforms offer Application Programming Interfaces (APIs), whereby third parties can integrate the distributed work management functionality into their custom applications. Examples of horizontal microtask crowdsourcing markets are Amazon Mechanical Turk and Microtask.com.

Figure 2.7 shows the worker and client flows of Amazon Mechanical Turk. Tasks, called Human Intelligence Tasks (HITs) are packaged in groups offered as a bundle by the same requester, and groups are displayed in order of number of HITs.

HITs have a descriptive title, an expiration date, a time slot for completing them, and the amount paid per solved HIT. A survey conducted in 2010 on the demographics of Amazon Mechanical Turk [38] by Panos Ipeirotis revealed that the population of workers is mainly located in the United States (46.80%),


Figure 2.7: CrowdSourcing with Amazon Mechanical Turk.

followed by India (34.00%), and then by the rest of the world (19.20%), with a higher percentage of young workers; most workers spend a day or less per week on Mechanical Turk, and complete 20-100 HITs per week, which generates a weekly income of less than 20 USD.

Another important project of CrowdSourcing is **CrowdSearcher**[6], a crowd-management system that implements a paradigm that embodies crowds and social network communities as first-class sources for the information management and extraction on the Web; its main aim is to provide an effective way of controlling the crowd in crowdsourcing campaigns. CrowdSearcher can be characterized as a multi-platform, reactive, expertise based and social networking based crowdsourcing approach.

Controlling means adapting the behaviour of the crowdsourcing systems in response to the quantity and timing of completed tasks, the quality of responses

[6] `http://crowdsearcher.searchcomputing.com`

and task results, and the profile, availability and reliability of performers.

### 2.2.2 Human Sensing

By the end of 2013, the total mobile cellular subscriptions will reach almost 7 billion worldwide, a penetration of 87%, with more than 1 billion mobile broadband subscriptions, which has made mobile broadband the fastest growing ICT service (+43%) in 2013. The biggest part of mobile terminals are equipped with sensors; a US survey as of February 2012 [39] shows that almost half of adult cell phone owners use their phones to get real-time location-based information. The combination of mobile terminal diffusion, broadband, sensors, including cameras and geo-positioning devices, provides a unique opportunity for developing large scale crowdsourcing applications in sectors that depend on the engagement of users distributed over a territory. These applications could be used both for collecting data, when other methods are inapplicable or too costly, and for rapidly spreading information and triggering action. Crowdsourcing can take advantage of people's mobility and of the increasing diffusion of mobile terminals equipped with sensors and broadband capacity.

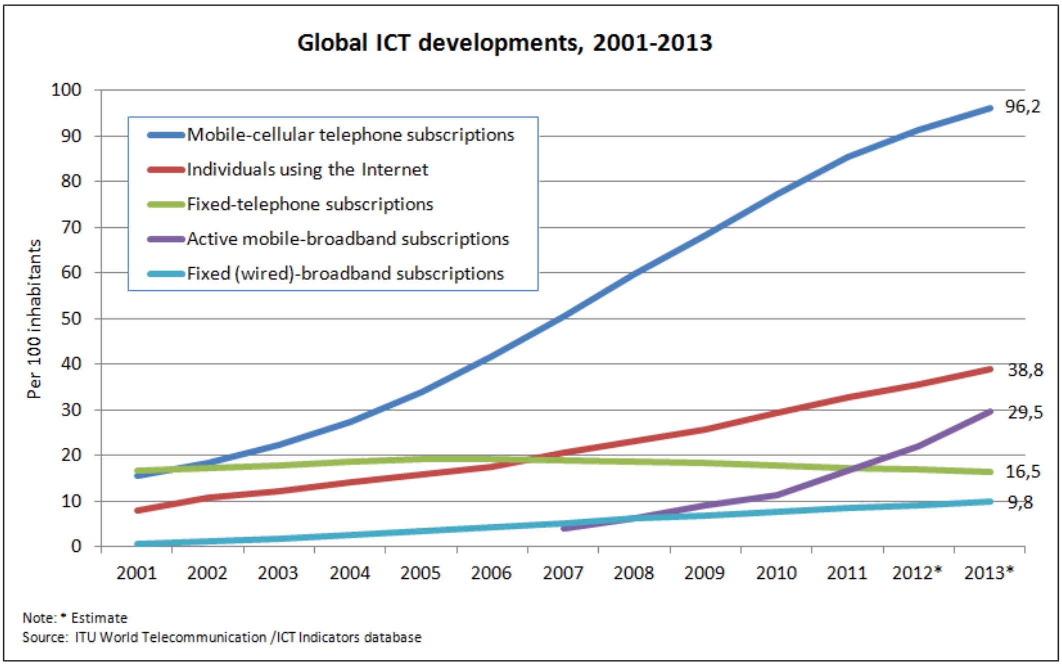



Figure 2.8: Global ICT developments 2001-2013.

As the sensor network and ubiquitous computing communities increasingly focus on creating environments that are seamlessly aware of and responsive to

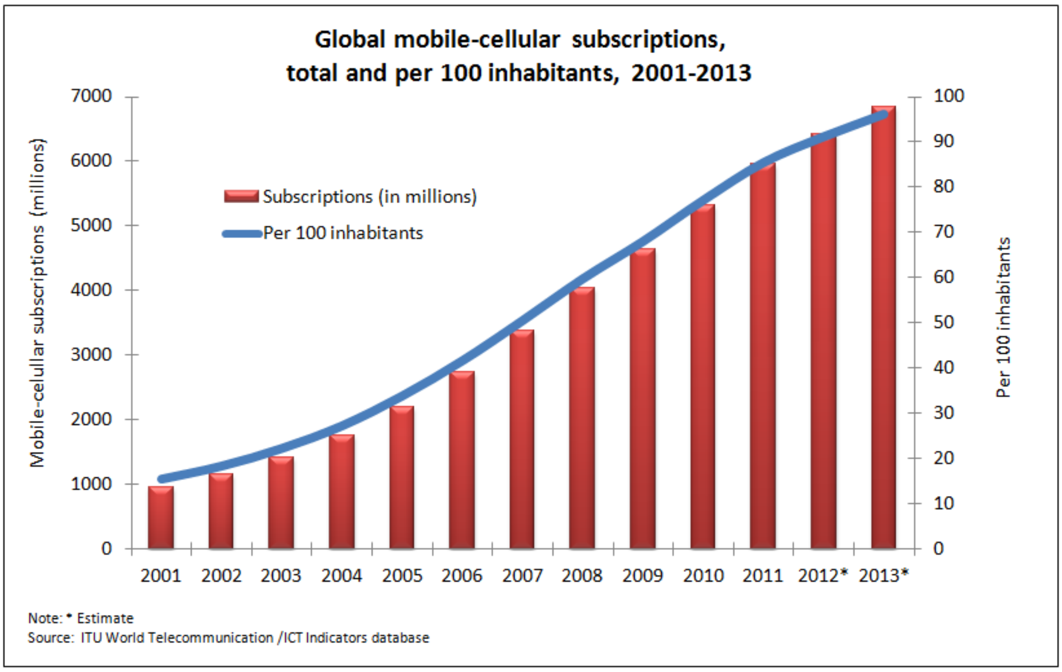



Figure 2.9: Mobile cellular world.

the humans that inhabit them, the need to sense people will become ever more pressing. Human-Sensing encompasses issues from the lowest level instantaneous sensing challenges all the way to large-scale data mining.

**Human sensing** denotes the assignment of data collection tasks to a crowd [40] [41]. The focus is on the real-time collection of data, in order to realize time-critical decision support systems and emergency management. Application areas include pollution monitoring [42], [43], traffic and road condition control [44], and earthquake monitoring [45].

**Human sensing** applications have been developed particularly in the environment protection field, where data collection and integration is critical for decision making and human sensed observations can be used to gather a broad range of physical data, e.g., air quality in urban spaces [42], surveillance of invasive species [46], noise pollution [47], and water quality. For example, Figure 2.10 [48] shows Creek Watch, a mobile and fixed Internet application whereby people can post data about watersheds rapidly and without other instrumentation than a standard mobile phone, like the amount of water, the rate of flow, the presence of trash and pictures of the waterway. The application design has focused both on the user interfaces, on the incentive mechanisms for engaging citizens, and on the utility of data for the scientific community that consumes them.

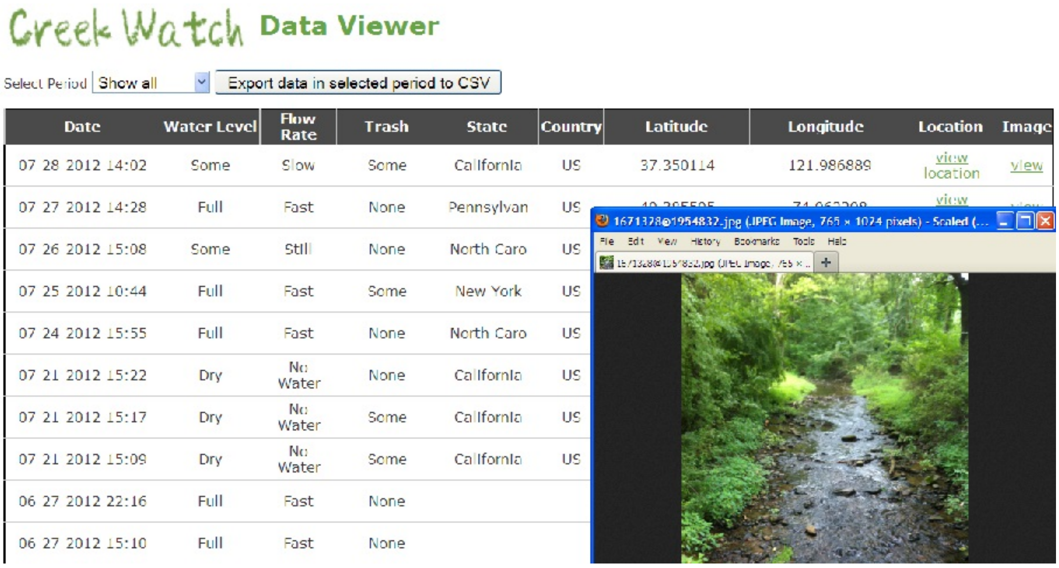



Figure 2.10: The interface of the Creek Watch data collection application.

Also social media have been experimented for harvesting heterogeneous and complex data, such as reporting on urban flooding events using geo-referenced tweet functionalities [49]. Using the Twitter.com microblogging service and a custom smart-phone application citizens can report events according to an existing professional controlled vocabulary (e.g. "basement flooding", "powerline down"), which is particularly useful in emergency conditions to deliver timely response. Similar experiments of streaming human visual experience into data have been conducted in Thailand to map flooded areas and the associated damage, and in the Netherlands, to engage citizens in the management of emergency service, like fires. *Social Mobilisation* is an approach that goes beyond human sensing as it aims at spreading information among the population and trigger action. Its specificity is that it addresses problems with time constraints, where the efficiency of task spreading and of solution finding is essential, and exploits the social network connections among people as a vehicle for information diffusion. The *DARPA Network Challenge* [50] is an example of the problem and of the techniques employed to face it.

The challenge required teams to determine the coordinates of ten red weather balloons placed at unknown locations in the United States. The winning team employed a novel recursive incentive mechanism that permitted

Figure 2.11: A $40,000 online challenge (DARPA Network Challenge) proposed by the US government has been won by a team of researchers from the Massachusetts Institute of Technology - just hours after it was launched.

them to locate all balloons in under nine hours. Applications are also found in safety critical sectors [51], like civil protection [52] and disease control [53].

## 2.3 Dimensions of Human Computation (What, Who, How)

How is the concept of human computation related to other concepts, such as crowdsourcing, collective intelligence and social computing? In the spirit of crowd wisdom, let's first examine these concepts as they are defined in Wikipedia (Table 2.1).

Based on these definitions, "crowdsourcing" can be considered a method or a tool that human computation systems can use to distribute tasks through an open call. The term "social computing" is a broad concept that covers everything to do with social behavior and computing. Finally, "collective intelligence" refers to the emergent intelligent behavior of a group of individuals, which includes non-humans and non-living things.

None of the related concepts emphasize the idea of explicit control. There

Table 2.1: Difference between Crowdsourcing, Collective Intelligence and Social Computing.

| | |
|---|---|
| **Crowdsourcing** | The act of outsourcing tasks, traditionally performed by an employee or contractor, to an undefined, large group of people or community (a crowd) through an open call. |
| **Collective Intelligence** | A shared or group intelligence that emerges from the collaboration and competition of many individuals and appears in consensus decision making in bacteria, animals, humans and computer network. |
| **Social Computing** | Technology for supporting any sort of social behavior in or through computational system, e.g., blogs, email, instant messaging, social network services, wiki and social bookmarking.<br>Technology for supporting computations that are carried out by groups of people, e.g., collaborative filtering, online auctions, tagging and verification games. |

is no explicit decomposition or assignment of task, no explicitly designed mechanisms for ensuring that the human computers tell the truth. What is different about human computation systems is the level of explicit control, instead of focusing on studying human behavior, the focus of human computation research is on algorithms, which either specify exactly what gets processed, by whom and how, or explicitly organize human efforts to solve the problem in a well-defined manner.

Conceptually, there are three aspects - “what”, “who” and “how” - of any human computation systems where explicit control can be applied.

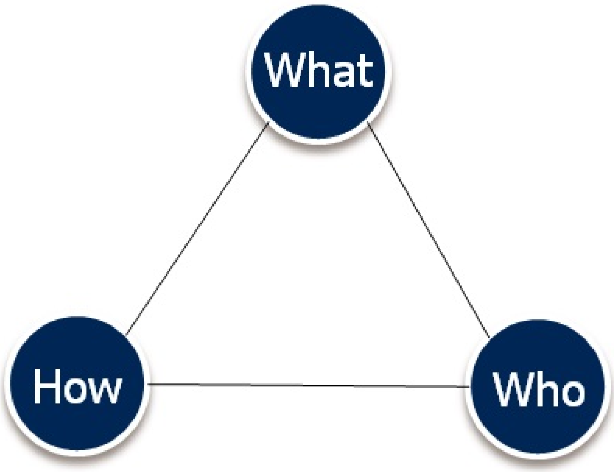



Figure 2.12: Human Computation Dimensions: What, Who and How.

### The "What" Aspect

We must have an algorithm that outlines in what manner to solve the problem. We need to organize a set of operations and a combination of control structures that specify how the operations are to be arranged and executed.

Some research questions relevant to the "what" aspect of human computation include the following:

- What tasks can be performed adequately by machines, therefore eliminating the need for human involvement? Can we leverage the complementary abilities of both humans [54] and machines to make computation more accurate and efficient?
- How do we decompose complex tasks into manageable units of computation and order them in such a way to handle the mode of behaviour of human workers?
- How do we aggregate noisy and complex outputs from multiple human computers in the absence of ground truth?

### The "Who" Aspect

Now that we know what operations we need to perform, the next question is to whom each operation should be assigned. While some tasks can be done by non-experts, other are more knowledge intensive and require special expertise. For example, a classification of a rare insect species will have a better result if it will be done by a specialist in insects than by a person with no insects knowledge. Some research questions relevant to the "who" aspect of human computation include:

- What are some effective algorithms and interfaces for routing tasks?
- How do we model the expertise of workers, which may be changing over time?
- What are the strategies for allocating tasks to workers, if their availability, expertise, interests, competence and intents are known versus unknown?

### The "How" Aspect

Now when we know operation to be done and who will do them the last question is how can the system motivate workers to participate and to carry out the computational tasks to their best abilities. Some research questions relevant to the "how" aspect of human computation include:

- How do we motivate people to have a long-term interaction with the system, by creating an environment that meets their particular needs?
- How do we design game mechanisms that incentivize workers to tell the truth, i.e., generate accurate outputs [55]?
- What are the new markets, organizational structures or interaction models for defining how workers relate to each other?

# Chapter 3

# Games with a Purpose

## 3.1 Components of a game

Not all the systems that are considered as "games" share the same structure, although there are several elements common to all of them and necessary to be able to consider a game as such. In the following a description of all these elements, necessary to create a gameful experience are provided, based on the analysis that has been done in [56].

#### Players

A game is primarily an experience designed for players; it is the only form of entertainment that is designed to require active participation from their consumer, a voluntary participant that accepts the rules and constraint of a game and strives to employ worse rather than better means to reach a particular objective. Players are thus the main elements of a game. Without at least a player we cannot say to have any game at all, since they are targets and enablers at the same time.

#### Objectives

Every gameful system lay out specific goals that its players have to reach when participating in the experience. This is very different from other experiences in which we can participate in general. When you watch a film or read a book there is no clear-cut objective presented for you to accomplish during the experience of course, there is one for the characters, but not for the consumers.

In life, we set our own objectives and work as hard as we feel necessary to achieve them. One does not need to accomplish all of her objectives to have a successful life. In games, however, the objective is a key element without which the experience loses much of its structure, and our need to work toward the objective is a measure of our involvement in the game.

### Rules

If complete freedom was given to the players to reach a particular objective, the purpose of the game, that is create an entertaining experience for the participants, would be less effective: in order to reach the goal, the players would perform the most efficient, but not necessarily enjoyable, actions to achieve it. A good designer has the role to define rules to limit player behaviour and proscribe reactive events.

Rules are fundamental pieces of any gameful experience that define allowable actions by the users and consequential reaction from the system; they may be used also to define game objects and concepts that could be used to reach a particular objective in the game.

In traditional games, there is no authority that is enforcing the players to respect the rules, but usually every participant is willing to respect them because they recognize that they are a key structural element and without them the game would not function. In a digital game, on the other hand, the rules could be enforced by the application in charge of managing the state of the game and the interaction between the players and the system.

### Conflicts

The rules and the admissible actions that can be performed in a game tend to deter players from accomplishing goals directly. This particular challenge for the players is one of the distinctive element of games: conflict, which the players work to resolve in their own favor. Designing conflicts requires deep knowledge from the game designer, which has the hard task of balancing the rules and the affordable actions to make reaching the goal as difficult as she thinks it would be beneficial for the entire experience.

### Resources

In a game, there are particular objects or elements that hold a high value because they can help the players in reaching their objectives, but they are usually scarce in order to pose an additional challenge. Finding and managing resources is a key part of many games, whether those resources are cards, weapons, time, units, turns, or terrain.

Resources are, by definition, items made valuable by their scarcity and utility. In the real world, and in game worlds, resources can be used to further our aims; they can be combined to make new products or items; and they can be bought and sold in various types of markets.

### Outcomes

The last element that most of the games have in common is that for all their rules and constraints, the outcome of both experiences is uncertain, though there is the certainty of a measurable and unequal outcome of some kind (e.g. a winner, a loser).

The outcome of a game differs from the objective in that all players can achieve the objective, but other factors within the system can determine which of them actually win the game. The aspect of uncertainty in outcome is a key motivator for the players. If players can anticipate the outcome of a game, they will stop playing.

## 3.2 Serious Games

In 2011, Nielsen reported that gaming has become the second most popular activity on the Internet in the US. The massive amount of time that people spend in online gaming is being more and more exploited by developing games that trascend pure entertainment purposes. A rediscovered trend, albeit not new, is to use such applications for Education [57], [58], [59], Military Strategy[60], [61], Weather Forecast, Government[60], Corporate Training[60], Healthcare[62], [63] among the others.

Through the media of games and simulation, the player learns in a natural and spontaneous way, absorbing messages and information not structured analytically but inserted within the gaming mechanism and the challenges that she has to face.

According to the Serious Games initiative which started in 2002 and coordinated by Ben Sawyer, Dave Rejeski, and others [64], games that have these features are considered to be part of a particular category called Serious Games. Serious Games refer to "applications of interactive technology that extend far beyond the traditional videogame market, including: training, policy exploration, analytics, visualization, simulation, education and health and therapy".

Thanks to the increasing interest in the application of immersive technologies in education, a fluorishing market is emerging for this particular subgenre of games. AmbientInsight has predicted that, during the forecast period 2013-2018, the compound annual growth rate (CAGR) for Game-based Mobile Learning products in North America is 12.5% and revenues will rise steadily from $227.97 million in 2013 to $410.27 million by 2018. [65]

## 3.3 Gwap as Serious Games

The application of gaming technology, process, and design to the solution of tasks that are relatively easy to complete by humans but computationally rather infeasible to solve [12] has given birth to a special subgenre of Serious Games, called Games with a Purpose (GWAP)[66].

The GWAP concept refers to the design of an online game for the purpose of embedding a computational task in its gameplay; a GWAP may help the resolution of complex problems, where algorithms do not suffice and human contribution can make a difference. This idea has several successful implementations, ranging from multimedia content processing [67] to protein folding [68]. Thanks to these games, people are generating useful data for scientific purposes as a by-product of their gameplay actions while enjoying the entertainment experience.

Most works on GWAPs focus on embedding a specific problem solving task into an enjoyable user experience and on evaluating the quality and quantity of output produced by players; this is the first distinction with respect to traditional serious games, that are usually covering a broader experience and aim at educating or informing their users.

As counterintuitive as it seems, since the purpose of a GWAP is to obtain meaningful contributions by the players, the entertainment aspect is usually not the main focus: trying to find mechanisms able to elicit good responses from the player has been the starting point for most of the past works that are presented in 3.4

## 3.4 Survey of Existing Games

A common way to get contributors to participate in a human computation grid is to use extrinsic motivation. Extrinsic motivation comes from an outer source, such as granting access to special web resources, or simply by paying the contributors. Systems such as Mechanical Turk, Microtask, or Crowdflower allow customers to upload small tasks such as reviewing a website or tagging images or sound files. The customers then pay other users, so-called workers, to solve these tasks. Another project that uses motivation not through the system itself is reCAPTCHA. This project serves the protection of publicly available web services from abuse by automated systems. A typical reCAPTCHA is an image containing several distorted characters. Users type these characters to prove that they are indeed human. The system generates these images from scanned documents. The solutions entered by humans improve the digitization process [12].

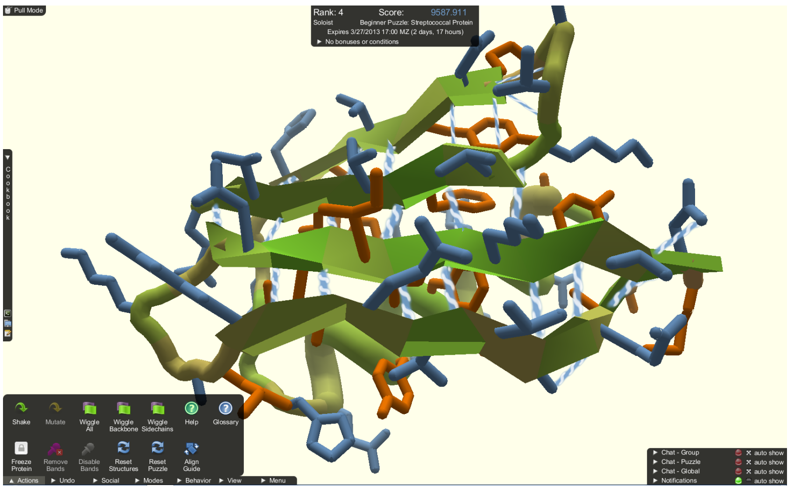
In contrast to digital human computation games, where players are motivated to spend cognitive effort wholly out of their own interest, all these systems provide motivation through secondary elements. Even though, the systems and services mentioned above are easy to use and the implementation of tasks is relatively simple, other projects demonstrate the power of digital games in the domain of human computation. Common tasks for human computation games are relation learning or resource labeling. Well-known examples in this regard come from the Games with a Purpose (GWAP) series[69]. It consisted in a website that was offering several games that were hiding computational tasks as part of their game mechanics.

Hereby a list of the most prominent games with a purpose that can be found in literature is provided, based on our research on the field. To ease the classification, the games have been divided in categories based on the problem spaces defined in [70].

### 3.4.1 Intuitive Decisions

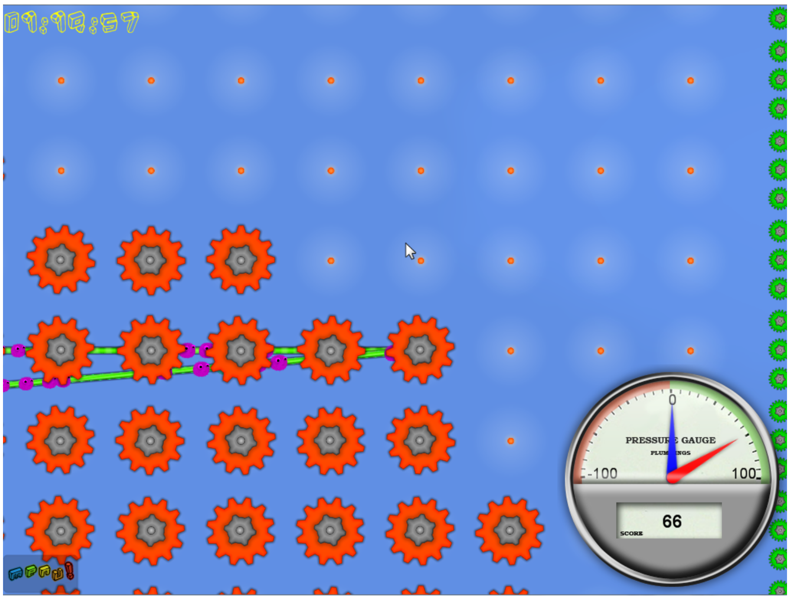
Intuitive Decisions tasks are related to combinatorial optimization tasks like packing problem that are known to be NP-hard [71]. It has been proven [72] that humans are able to solve by intuition even complex tasks belonging to this category of problems, thus human computation can be employed as a mean to use mental abilities of the participants to find solutions or algorithms able to solve combinatorial problems disguised as puzzles.

- **FoldIt** Foldit [18] is an online video game that casts protein structure manipulation as a puzzle solving competition.The game tries to predict naturally occurring protein structures and to design novel proteins not previously seen in nature. In order to achieve this goal, the game gives players the ability to manipulate and optimize protein structures while competing and collaborating with other players to discover the best structures, using various tools provided within the game.The highest scoring solutions are analysed by researchers, who determine whether or not there is a native structural configuration (or native state) that can be applied to the relevant proteins, in the "real world". Scientists can then use such solutions to target and eradicate diseases, and creating biological innovations.

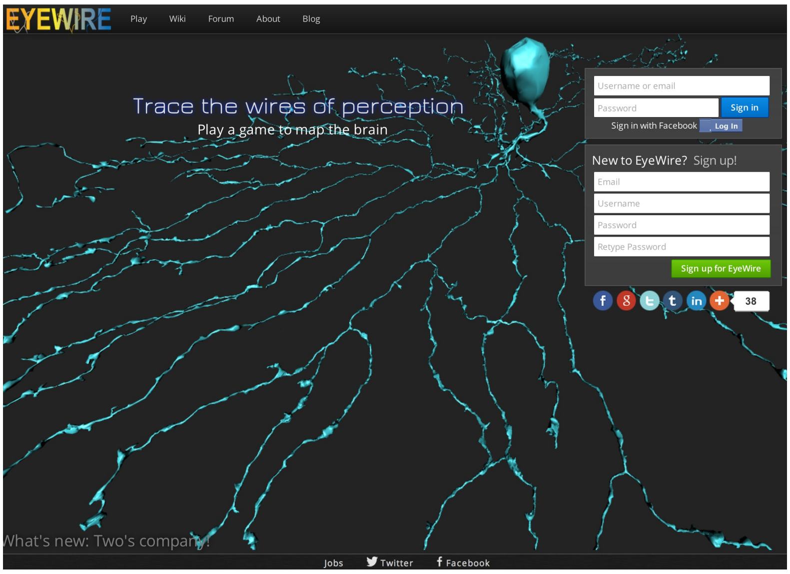


The most remarkable result obtained by the game has been the discovery of the crystal structure of the Mason-Pfizer monkey virus (M-PMV) retro-viral protease, an AIDS-causing monkey virus. While the puzzle was available to play for a period of three weeks, players produced an accurate 3D model of the enzyme in just ten days. The problem of how to configure the structure of the enzyme had been an unaccomplished goal of scientists for 15 years.

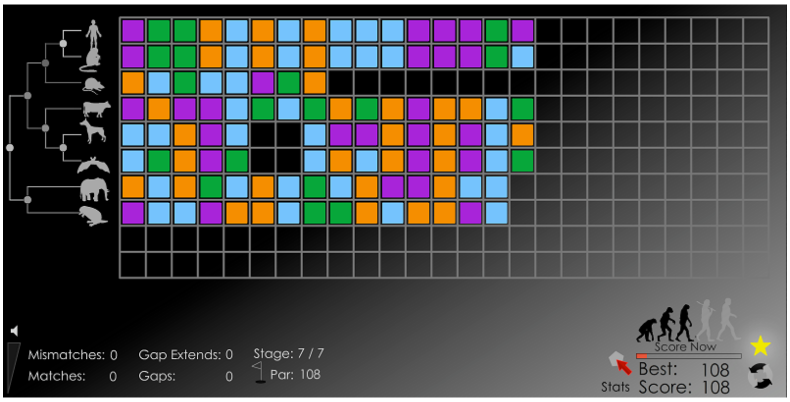
- **Plummings** In Plummings [19] players try to solve the FPGA placement problem, which consists of determining how to place a set of logic clusters on an array of Field Programmable Gate Array (FPGA) tiles such that the critical path of the circuit is minimised. The game allows a player to freely swap clusters on the array; however, details of cluster swapping and the critical path are abstracted away from the player in the form of a game. Each swap will either improve or degrade the resulting speed of the circuit based on the critical path length. This is reflected in a score, and players attempt to improve the critical path.




- **EyeWire** In EyeWire [73], players are challenged to map branches of a neuron from one side of a cube to the other, in a 3D puzzle fashion. Players scroll through cubes (measuring about 4.5 microns per side and reconstruct neurons in volumetric segments with the help of an artificial intelligence algorithm developed at Seung Lab. To sustain continous contributions from the players, the game offer challenges in which players compete for bonuses, profile icons, unique chat colors and even neuron naming rights. Players level up in EyeWire by beating the Starburst Challenge, unlocking the right to map difficult starburst neurons and earn double points. Advanced players participate in Hunts, where they scour completed cells looking for mergers or mistake branches that need to be “scythed” away by an in-game character. No results of the approach have been made publicly available yet.




- **Phylo** Phylo [74] is a citizen science framework to solve multiple sequence alignment problems for a set of dna coming from different vertebrate species. The problem is translated into a puzzle, by using regions with low confidence scores (i.e. likely to be misaligned); the player has to move horizontally the blocks representing nucleotides in order to find a configuration that maximizes conservation properties across columns while minimizing the number of gaps.




### 3.4.2 Aesthetic Judgment

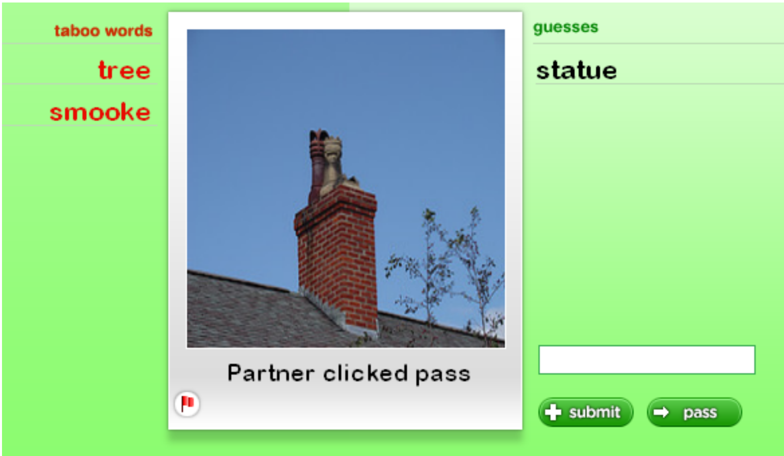
The design and implementation of computational systems capable of having human-level perception and understanding of aesthetics, like the quality of an image, a piece of music or the proportions in a picture is still an unsolved challenge, while humans have a natural instinct for judging and estimating such features, thanks to their superior abstraction capabilities and senses.

- **Tree Modeler** The Exploratory Tree Modeler [22] is a modular software system designed to test exploratory modeling techniques and used to prototype design tools for spaces of trees, humans and bidirectional reflectance distribution functions. In the twelve months after the release of the tool to the general public, more than 6000 new models of tree were created.
- **Picbreeder** Picbreeder [75] is an online service that allows users to collaboratively evolve images. Like in other Interactive Evolutionary Computation programs (IEC), the images are evolved based on the users' choice over the most appealing ones to produce a new generation. The novelty of Picbreeder is the possibility of evolving others' images, by branching them with the use of an online tool. This has been possible thanks to the NEAT algorithm, that guarantees that as images are branched further, new directions will be created for the users to explore. The strength of the system architecture that supports Picbreeder relies on the fact that it is not limited to evolving images but could be used to evolve any artifact, including music, voices and intelligent agents.

### 3.4.3 Contextual Reasoning

Most cognitive processes are contextual in the sense that they depend on the environment, or context, inside which they are carried on. Human Reasoning tasks that are not feasible for the machines often involve semantic understanding and abstraction capabilities typical of human beings. Examples for the application domain of contextual reasoning regards tasks such as image and audio annotations.

- **ESP Game** The ESP Game [17] has been the first Game With a Purpose ever created and was one of the most famous ones of the original GWAP suite developed by Louis Von Ahn. It was created to crowdsource the collection of tags that describes images for indexing purposes. The ESP game is an example of a GWAP paradigm called "output-agreement"[4], in which two players get the same input and are rewarded when their outputs agree.



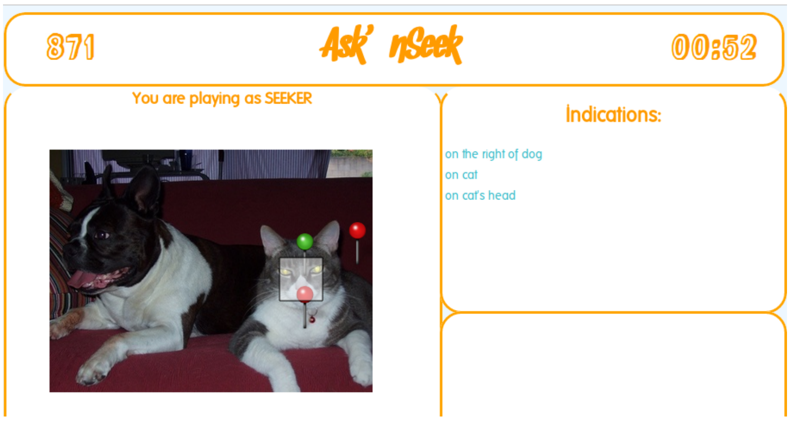

- **Tag a Tune** Tag A Tune [76] is an online game developed to collect tags for music and sound clips. It introduce another fundamental validation mechanic for GWAP, called “input agreement” in which players are provided with either the same or a different object and asked to describe that object to each other. Based on each other’s descriptions, players must decide whether they have the same object or not. In particular, in this GWAP players are providing textual annotations that should meaningfully describe the audio object; based on these annotations, the players have to guess if they are listening the same multimedia content or not.


- **Verbosity** Verbosity [21] is an online game developed to create a database of “commonsense facts” true statements about the world that are known to most humans. It is a two players game in which one player, the "Narrator"is given a secret word and has to hint the other player to type that word by providing clues. The clues are provided in the form of sentence templates with blanks to be filled in; the Narrator can fill in the blanks with any word except for the secret one.


- **Peekaboom** Peekaboom [77] is an online game that aims at collecting image metadata related to object recognition. Two players take part in a gaming session under two different roles, "Peek" and "Boom". Peek starts the round with a blank screen while Boom is provided with an image and a word related to it. The goal of the game is for Boom to reveal circular areas of the image to Peek as an hint for him to guess the associated word. Peek has to enter guesses of what Boom's word is, aided in the task by Boom's indication on wheter the guess is close or not to the solution. When Peek successfully guess the solution, the players' roles are switched. The game provides an incentive for Boom to reveal only the necessary area by granting him more points the less pixel are revealed in case of Peek's successful guess.


- **OntoGalaxy** OntoGalaxy [78] is a fast-paced action-oriented science fiction game comparable to games like Asteroids or Starscape. It has been the first GWAP to provide a storyline and progressive gameplay elements such as levels, different kind of enemies and upgrades for the players' spaceship. The game was used to populate an ontology with common words of the English language, and did so by providing missions to the players on the line of "Collect all freighter ships that are labeled with a word that is a synonym for the verb X"The underlying engine of the game was able to establish arbitrary relations between two elements, may them be words, images or audio files.

- **AskNSeek** Ask'NSeek [79] is a two-player web-based guessing game that aims at both detecting and labeling objects within an image. The users need to guess the location of a hidden region within an image with the help of semantic and topological clues. The collected annotations are then combined with the results from content analysis algorithms to feed a machine learning algorithm to outline the most relevant region within the image and their names.



# Chapter 4

# Gamification

Gamification as a term originated in the digital media industry. The first documented usage dates back to 2008, but the gamification term assumed wide-spread adoption just in the second half of 2010, when several industry players and conferences popularized it.

In Section 4.1 we will try to define what "Gamification"means and why it derives from the term "Game"; later in Section 4.2 we will understand how Gamification motivates the users to act within a system. Finally, in Section 4.3 we will present some successful stories of companies that have obtained significative benefits by applying Gamification in their business.

## 4.1 What is gamification

The field of Gamification is still young and developing quickly, so there are numerous opinions as to what Gamification exactly is. One of the most general and used definition is:

*" Gamification is the use of game elements*
*and game-design techniques in non-gaming contexts."*[80]

To fully understand this definition, it is important to specify each term in a deeper and detailed way.

**Game Elements**

To precisely define the scope, it is important to state that we are talking about games, not of play. While games are usually played, play represents a

different and broader category than games.
Play can be performed in any way and only one own's imagination can set boundaries. In that sense, play as an activity is always more open ended. Games, however, are constrained by rules and have often objective to reach by overcoming challenges.

So the difference between play and games is the amount of constraint and authorship the player has over the experience. The more authorship the player has over the application, the more it becomes like a play. The more the player is the actor following strict guidelines dictated by the application, the more it becomes like a game. Gamified applications use elements of games that do not spawn entire games. Of course, the boundary between games and applications with game elements can be very blurry, because often this boundary is personal, subjective and social. Self representation with avatars, three-dimensional environments, narrative context, feedback, reputations, ranks, levels, marketplaces and economies, competition under rules that are explicit and enforced, team parallel communication systems that can be easily configured, and time pressure are all game elements.

To generalize, using a very liberal interpretation, elements are characteristic to games and are found in most (but not necessarily all) games and possess a significant role during the gameplay.

**Game-design techniques**

It seems not so difficult to take a game element such as a point system, transferring it into a website or putting a leaderboard to show who is the first, but the users often get burnt out by the endless treadmill of points accumulation and thus abandon the system since many users do not find points very interesting. Even new users may arrive with high hopes, only to abandon the system when they see the top of the leaderboard immensely far above them. So, it is not so simple to decide which game elements to put, where and how to make a successful gamified experience and where the game-design techniques should be included. Game design is a mixture of science and art and a lot of analysis of successful past experiences is required to accomplish satisfactory results. The game-design techniques encompass different level of abstraction, as shown in Figure 4.1.

| Level | Description | Example |
|---|---|---|
| *Game interface design patterns* | Common, successful interaction design components and design solutions for a known problem in a context, including prototypical implementations | Badge, leaderboard, level |
| *Game design patterns and mechanics* | Commonly reoccurring parts of the design of a game that concern gameplay | Time constraint, limited resources, turns |
| *Game design principles and heuristics* | Evaluative guidelines to approach a design problem or analyze a given design solution | Enduring play, clear goals, variety of game styles |
| *Game models* | Conceptual models of the components of games or game experience | MDA; challenge, fantasy, curiosity; game design atoms; CEGE |
| *Game design methods* | Game design-specific practices and processes | Playtesting, playcentric design, value conscious game design |

Figure 4.1: Design levels of abstraction

**Non-game contexts**

Gamification uses game elements for purposes which are different than the normal expected use for entertainment. Likewise, joy of use, engagement, or more generally speaking, improvement of the user experience represent the currently predominant use cases of "gamification"Ṡince each of these use cases are strictly related to real-world business or social impact goals, it is of utterly importance to keep in mind that the users should not fall into a fantasy world in which they completely lose the perception of the reality, but they should just be engaged more deeply with the gamified product in order to participate, share and interact in target activities or communities.

There are three particular non-game contexts: internal, external and behavior change.

In the first case, the companies use gamification to improve productivity within the organization in order to foster innovation, enhance camaraderie or to otherwise derive positive business results through their own employees. Internal gamification is something called "enterprise gamification" and in this case there are two distinguishing scenarios. In the first one, the players are already part of a defined community, the company knows them and how they interact with each other on a regular basis. The scenario derives from the first but motivational dynamics of gamification must interact with the firm's

existing management and reward structures. Internal gamification can work for core job requirements, but it is not always applicable, thus there must be some novel motivations.
External gamification involves customers or prospective customers and such applications are generally driven by marketing objectives. Gamification in this case, attempts to improve the relationships between businesses and customer, producing increased engagement, identification with the product, stronger loyalty and ultimately higher revenues.

Finally, behavior-change gamification aims at creating new beneficial habits among the population; this may range from encouraging people to live in a healthier way, to study more, to maintain a sustainable and eco-friendly behavior and so on [80]. Behavior change programs are often run or sponsored by nonprofits and governments, but they can also produce results that are beneficial even for private institutions.

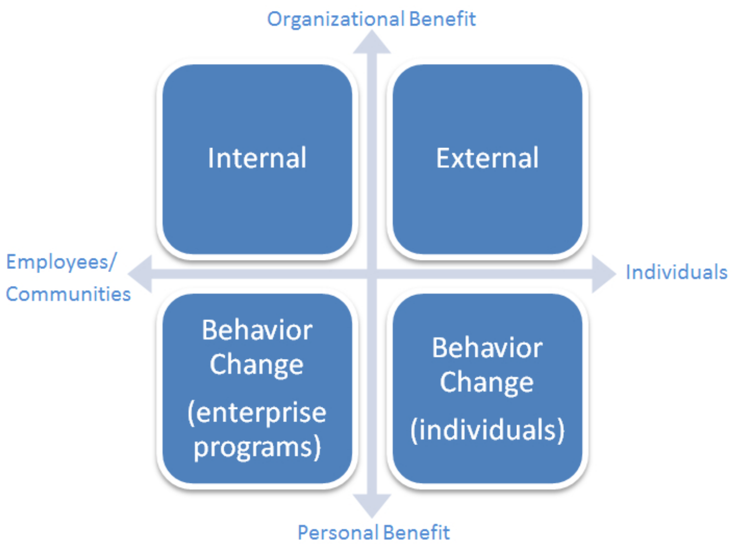



Figure 4.2: Relationships between different Gamification contexts

### 4.1.1 Understanding the players

Gamification is a strategy for influencing and motivating the behavior of people, whether being customers, employees, students, fans, constituents, patients, etc… . While gamification uses techniques from game design, it is not a new way to reach the gamer market. The audience for gamification is anyone you want to engage repeatedly in order to elicit a particular behavior.

One of the most popular theory follows the work accomplished by Richard Bartle in understanding player types [81]. In this seminal work, developed by studying players of MMOGs (massively multiplayer on-line games), Bartle identified four types of players.


Figure 4.3: Bartles player types.

**Explorers**
Explorers delight in having the game expose its internal machinations to them. In a sense, for them, the experience is the objective. One example of a game suited to the explorer player type is any type of Action Game in which a player has to play different missions in different environments to find every hidden level behind every pipe and block, and bring that knowledge back to her peers for glory.

**Achievers**
Achievers regard points-gathering and rising in levels as their main goal, and all is ultimately subservient to this. They drive a great deal of projects, services, and brands. The problem with designing exclusively for this player type is that its difficult to develop a system where everyone can win and achieve. And for achievers, losing at the game will likely cause them to lose interest in playing it.

**Socializers**
Socializers play games for the opportunity of having a social interaction.

Games focused on socializers comprise some of the most enduring games throughout history like dominoes, bridge, mahjong, poker; the thread tying them together is that each is an extremely social experience. To be clear, it isn't that socializer don't care about the game or winning, they do but to them the game is just a backdrop for meaningful long-term social interactions. It's the context and catalyst, not the end in itself.

**Killers**

Killers make up the smallest population of all of the player types, however, they are important to understand. They are similar to achievers in their desire to win; unlike achievers, winning is not enough. They must win and someone else must lose. Moreover, killers really want as many players as possible to see their supremacy over the others, and for their victims to express admiration/respect.

A player can have characteristics of all four types at the same time. However, most people do not express more than one trait. The result of the Bartle Test is the "Bartle Quotient" , which is calculated based on the answers to a series of 30 random questions in the test, and totals 200% across all categories, with no single category exceeding 100%.
For the average person, the breakdown might look something like this:

*80% socializer*
*50% explorer*
*40% achiever*
*20% killer*

The vast majority of people (as much as 75%) are probably socializers. Explorers and achievers each make up about 10% of the population, and killers account for 5%.

## 4.2 Why gamification

A particularly compelling, dynamic, and sustained gamification experience can be used to accomplish a variety of business goals, and to reach such results the key factor is users' motivation.

To be motivated is to be encouraged to do something. All students study several and different subjects in their scholastic life, but they are not always interested to study them. So why do they do? Many are the underlying motivation, but a simple distinction is between those who want to study and those feel like they have to study. The will to do something is called *intrinsic* motivation because, for the person involved, it lies inside the activity; it thus refers to motivation that is driven by an interest or enjoyment in the task itself and does not rely on external pressures or a desires for reward. On the other hand, if someone feels that she has do something even against her will, then *extrinsic* motivation arises, because the motivation lies outside of her desires; the motivation to perform an activity lies in the need to attain an outcome, whether or not that activity is also intrinsically motivated.

Intrinsic motivation has been studied since the early 1970s. Students who are intrinsically motivated are more likely to engage in the task willingly as well as work to improve their skills, which will increase their capabilities. Students are likely to be intrinsically motivated if they attribute their educational results to factors under their own control (autonomy) or if they believe they have the skills to be effective agents in reaching their desired goals (self-efficacy) or are interested in mastering a topic, not just in achieving good grades. These are intrinsic motivations. We are motivated to do something by reasons that come from outside your enjoyment or engagement with the activity.

Common extrinsic motivations are rewards (for example money or grades) for showing the desired behavior, and the threat of punishment that follows misbehavior. Competition is also an extrinsic motivator because it encourages the performer to win and to beat others, not simply to enjoy the intrinsic pleasures of the activity. A cheering crowd and the desire to win a trophy are also extrinsic incentives.

There are two important theories related to "How to motivate the users"Ḃehaviorist thinking suggested that extrinsic motivation was the way to encourage people to do things. A reward or punishment, systematically

applied, would condition and reinforce responses in anticipation of further rewards or punishments.
Against this behavioral approach stands the Self-Determination Theory (SDT), that suggests that human beings are inherently proactive, with a strong internal desire for growth, but that the external environment must support this; otherwise, these internal motivators will be thwarted. Rather than assuming, as the behaviorist approaches do, that people only respond to external reinforcements, SDT focus on what human beings need to allow their innate growth and well-being tendencies to flourish[80]. SDT suggests that these needs fall into three categories; competence, relatedness and autonomy.

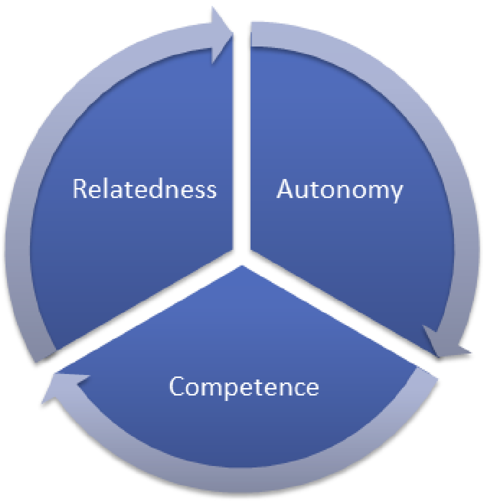



Figure 4.4: Self-Determination Theory (SDT)

*Competence*, or mastery, involves being effective in dealing with the external environment: pulling off a difficult deal.
*Relatedness* involves social connections and the universal desire to interact with and be involved with family, friends and others.
Finally, *Autonomy* is the innate need to feel in command of one's life and acting for what is meaningful and in harmony with one's values. Tasks that involve one or more of these innate human needs will tend to be intrinsically motivated. Game are perfect illustrations of the lesson of SDT. Why do people play? As we have already said, no one forces them to. Even simple games like Sudoku activate intrinsic needs for autonomy ( which puzzle I solve and how I solve it is entirely up to me), competence (I figure it out!) and relatedness (I can share the achievement with my friends)[80].

In the same way, gamification uses the three intrinsic motivators to generate powerful results. Levels and the accumulation of points can all be markers of competence or mastery. Giving players choices and a range of experiences as they progress feeds the desire for autonomy and agency. Social interactions such as Facebook sharing or badges you can display to friends respond to the human need for relatedness.

## 4.3 Successful Case Studies

In the following section are presented five successful case studies in which the business objectives are solved with the application of gamification techniques.

The TV show Psych, broadcast on NBC channels in 2010, launched its own gamified website with the scope to expand audience and deepen engagement to increase impressions on-line and on-air, leveraging existing content and fans' social networks to drive market and revenue growth.

The gamified approach induced fans to earn points by watching videos, reposting content, playing games and browsing photo galleries, rewarding fans who checked in before, during or after the show. The platform allowed them interact each other through multiple social platforms by integrating a chat and allowed the redemption of badges and points displayed on leaderboards to spur competition.

As a result, more than 30.000 fans registered on the site in the first year and the website experienced a 30% increase in overall site traffic, a 40% increase in share among 18-34 year old and a 47% increase in on-line merchandise sales that was subject to discounts gathered through the platform.

Adobe System Incorporated is a software house famous above all, for video and digital graphics. What it needed was to increase revenue by converting more free trials of Adobe Photoshop in sales, finding a quick way to teach users a few simple tasks to engage them.

The solution was the creation of "LevelUp for Photoshop" , a gamified adoption program that onboards new customers quickly so that they felt confident enough in their abilities to buy the product. The application did so by taking an existing curriculum and organizing it into a series of missions involving the completion of simple tasks, such as removing the "red eye" effect from a photo. It balanced tasks from easy to difficult and added incentives to encourage completion to unlock the next level, making it easy to share the corresponding earned badge on social networks.

As results, Adobe experienced a 400% increase of sales derived from free trials as new users learned the product and existing ones developed new skills.

ELOQUA
THE POWER TO SUCCEED.

Eloqua Corp. is a marketing automation Saas company, a subsidiary of Oracle, that had the need of making a successful on-line community even better with more sustained activities and deeper connections between members.

For their solution they used a platform called Jive Advanced Gamification Module, that helped in the definition of levels that community members could unlock through activities, added reward points for interaction and contribution and created competition through leaderboards.

As result, an already active community experienced a sustained 55% increase in average active users and technical support requests decreased as members found quality answers on community boards.

bluewolf™

Bluewolf, a global business consulting, had the objective to create an engaging community for their employees, to grow the company's visibility and showcase their expertise in innovative business process and technologies by motivating distributed employees to effectively use social and collaborative platforms.

The adopted solution was to use Salesforce Gamified solutions to measure the baseline employees' engagement on social networks and creating a social resource center to share solutions to common problems with concise explanations and training. The employees were awarded with points for completing their profiles, sharing content and receiving inbound clicks; the platform offered virtual and tangible rewards for accumulated points, including physical badges and e-stores gift cards.

As results, Bluewolf doubled the usage of their platform, accelerated traffic from social media sites by 20% month-over-month and increased internal collaboration by 57%.

Microsoft

Traditional business environments use specific language Service providers to perform translation operations, and then a secondary Service provider to assess the quality of the work done. The challenge was that, for some languages and locales, finding two independent language translation service providers could be difficult and expensive. The objective for Microsoft was to release high quality translations for their application by exploiting native language speakers from different nationalities and employed within Microsoft.

To address this business objective the Language Quality Game was developed to encourage native speaking populations within Microsoft to do a final qualitative review of the Windows user interface and help identifying any remaining language issues. The goal of the game was to improve translation accuracy and clarity for a series of screenshots and dialogs that were submitted to the employees, that in turn were gaining points for every grammar correction done. As a result more than 4600 players were able to report more than 6700 grammar errors. Success in the game was defined as the amount of coverage of screenshots across the 36 languages tested; incredibly, most languages had several reviewers' feedback per screen.

To summarize this analysis of gamification, we try to find an answer to the following question: "Why should a practice based on games be taken seriously in business?"

There are three particular reasons:

- Engagement
- Experimentation
- Results

**Engagement**
The same human needs that drive engagement with games are present in both the workplace and the marketplace. Gamification can be considered as a tool to design systems that motivate people to do things. Anything that makes customers and employees want to strengthen their relationships with the company, to buy company's products or to engage with the goals of the company can be gamified to bring even more benefits to a company.
Engagement has business value itself. If workers aren't fully engaged in their jobs, this undoubtedly affects not only their performance but their happiness. People know they should exercise more, eat better, use less energy, study more, but the hard part is being sufficiently motivated to do so. And for consumers, engagement is what leads them to initiate a transaction that may bring indirect benefits [80].
**Experimentation**
A second powerful aspect of game-based motivation is to open up the space of possibility. Mastering a game is all about experimentation. It is possible to face some failures, but because the game can always start over, failures do not feel so daunting. In most games, we can win, but we seldom lose in a definitive way. If a game is not too difficult and not too easy players are continually motivated to strive for improvement [80].
**Results**
Despite the novelty of the practice, a number of companies have seen significant positive results from incorporating game elements into their business processes, as we have shown in the five case studies detailed before. Unfortunately, the processes and the data to sustain the thesis that gamification helps to reach the defined objectives are not documented, thus the evaluation process they have applied cannot be validated through meaningful statistical tests.

# Part III

# Design

# Chapter 5

# GWAP Design

In this Section we introduce an abstract representation of the activities involved in GWAP development, with a hint at the precedence constraints and input-output relationships. A possible approach to this end is comparing and integrating the reference models representing the software and game development workflows.

The software development process is a well-known abstraction of the workflow employed in the construction and maintenance of a software product [82]. Different variants of the lifecycle have been proposed, adapted to different development strategies and/or application domains. In the domain of interactive applications, [83] introduced the process model shown in Figure 5.1, which abstracts the development of applications that have a strong focus on the structure and navigation of the front-end, such as Web and mobile applications; we will adopt such a model as a first ingredient for deriving a schematization of the GWAP and gamified application lifecycle.

The second ingredient should be a reference model of the game design workflow. However, such a model is not readily available from the literature. Although game development is conducted as an industrial process in the entertainment sector, the workflow of game design activities has not been described by means of a reference process model yet; the same lack of formalization carries over to gamified applications. To provide a basis for reasoning on how software and game development activities blend in GWAP and gamified application development, in Section 5.1 a possible process model from the game development is "reverse engineered" from the best practices detailed by Crawford [84] and Fullerton [56]. The resulting game development model is then

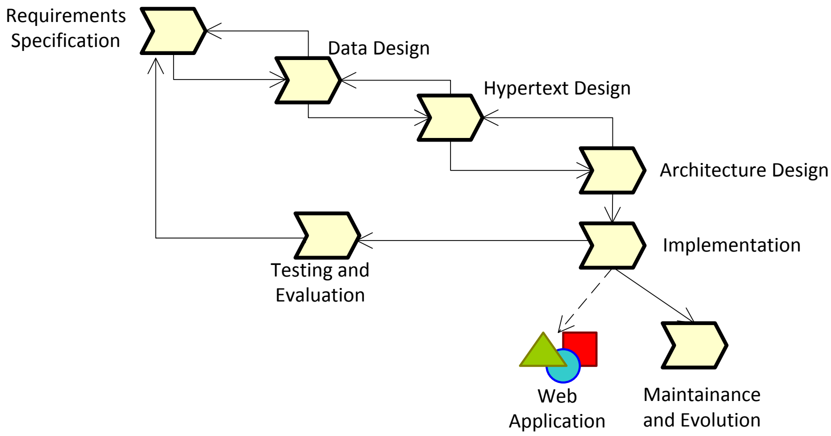



Figure 5.1: Process model for Web application development

compared and fused with the reference software lifecycle of Figure5.1, to yield a development process model that could represent the production workflows of GWAP (Section 5.2).

## 5.1 Traditional Game Development Process

The literature on game design does not prescribe a structured development process, yet companies and designers have distilled their experience into best practices and guidelines useful for organizing game production. One justification advocated for such an informal approach is that "*game design is primarily an artistic process and reliance on formal procedures is inimical to creativity*" [84]. However, with the transformation of games into a consolidated industry with time and budget constraints comparable to those of business software products, some authors [56, 85] have claimed that iterative and rapid software methods, most notably agile methodologies such as "Scrum", may be adequately applied also to the development of games.

Figure 5.2 shows a possible representation of the game development process, obtained by modelling the guidelines and practices suggested by widely recognized designers, such as Chris Crawford and Stacy Fullerton [84][56].

In the following, each phase of the process model is briefly described.

- **Player Experience Definition** pinpoints the goals of the game, the players' interactions, and the emotions induced. The output is a narrative document defining the game concept at a high level.

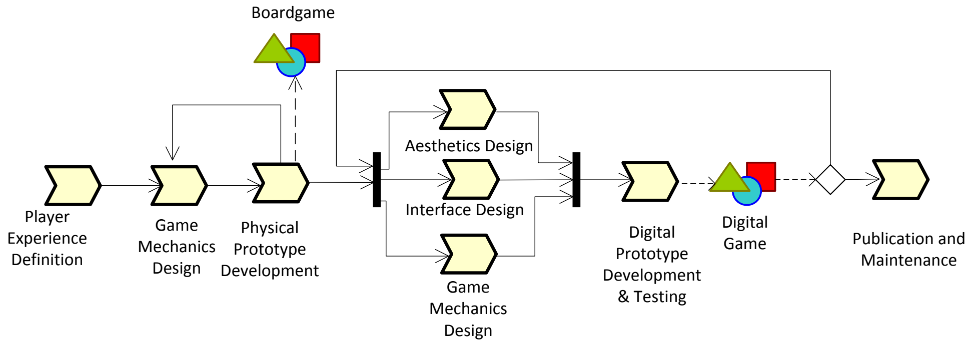



Figure 5.2: Process model for game development

- **Game Mechanics Design** defines the actions, challenges, and rules of the game. The output is a document outlining the translation of the game concept into the actual game dynamics.
- **Physical Prototype Development** involves the creation of a simplistic model that can be used to play and refine the game mechanics. Physical prototypes are typically created using slips of paper, cardboard, and household objects with hand-drawn markings, allowing the designer to focus on gameplay rather than technology and to make rapid changes in the design. Output of this phase is a playable physical game and a preliminary document, *The Game Design Document*, that defines the rules, the challenges and the components of a game, and the workflows of activities and actions that can be performed, better known as *Gameplay*.
- **Aesthetics Design** creates the visual and aural characteristics of the game, including the general look&feel (e.g., cartoon, futuristic, or historical), graphical resources (color palettes, indoor and outdoor scenarios, graphical resources, etc), the graphic models of characters and objects, and the sound themes and effects. The output comprises graphical and audio resources.
- **Interface Design** represents the user's viewpoint of the game, i.e., the display of the game status and of the controls that allow the user to play.

The iterative refinement of the Aesthetics, Interface and Game Mechanics design phases produce as output a final version of the Game Design Document, detailing the gaming experience as a whole.

- **Digital Prototype Development and Testing** involves the implementation of the design specifications into a digital product for testing purposes. This phase is conducted in subsequent iterations until the testing with the target audience achieves the intended goals. Iterations may affect also the prior steps, as shown in figure 5.2.
- **Publication and Maintenance** happens when the game has reached a consistency that allows it to be published and distributed to the whole audience; after publication a game may require maintenance in terms of bug fixes and possible refinements in the game mechanics. Novel functionalities can be added by following the entire process from the beginning, which normally results in a new edition of the game.

## 5.2 Gwap Development Process

Unlike the gamification of an application, the main goal of a Game with a Purpose, despite common beliefs, is not the engagement of its user but to involve humans in the computation process for tasks that are still too complex to be accomplished by machines; without a task to be solved, a GWAP would be just a traditional digital game. On the other hand, a GWAP that does not offer an interesting experience for its players will fail to accomplish its goals, since there will not be enough performer to solve the defined problems since this kind of application rely on the human desire to be entertained.

For this reason, the development process for a Game with a Purpose involves the definition of activities that has to be delegated to human performers and their integration within a game (existing or novel). Figure 5.3 shows the development process of a GWAP based on the experience of the authors and the design guidelines defined in [66].

### Requirement Specification

The requirement specification phase for the design of a GWAP is focused on collecting the information necessary for the definition of a task, a unit of work performed by human worker in the process of solving computational problems that cannot be resolved by AI. Cropping the silhouette of the models in a picture, recognizing and identifying the people contained in a set of images,

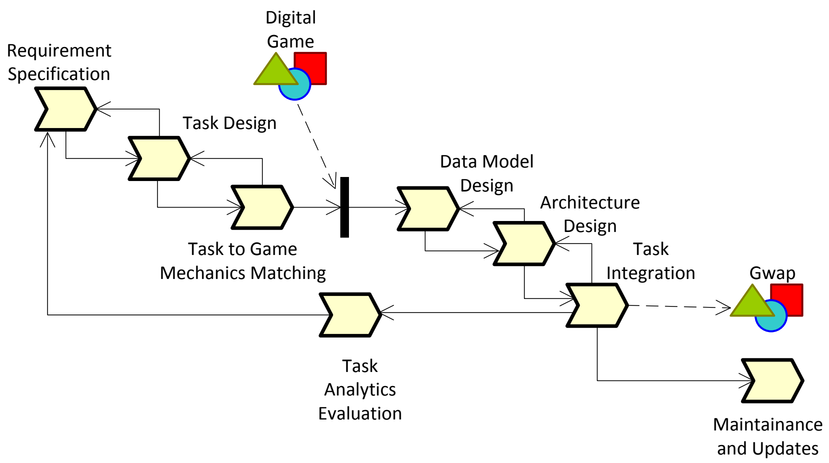



Figure 5.3: SPEM Model for GWAPs Development

collecting labaled data for training a are examples of what a task involving user of a GWAP may look like. A task can be compared to an algorithm, defined in [86] as "a finite set of rules which gives a sequence of operations for solving a specific type of problem". A task is defined by a *description*, generally a textual one, of the goal that has to be reached and a set of *admissible operations* that represent the mean with which this particular task can be accomplished.

### Task Design

Based on the output of the requirement specification, the task design phase involves the design of the algorithm able to solve the defined problem, for which well known methods, based on the decomposition of the problem in operations and controls, are known [87].

Given the fact that the algorithm has not to be performed by a silicon processor but by a human performer, two properties has to be kept in mind when designing it: finiteness and effectiveness. *Finiteness* requires the answer to be provided and verifiable in a known and finite number of steps. *Effectiveness* requires each operation of the algorithm to be executed by a human performer that has no prior knowledge on how to solve that particular problem.

Since the computation is performed by humans, the obtained results are typically quite noisy, and beyond the setting of very simple tasks there is often a need to aggregate information into a collective choice. The design of a task must take into consideration also this aspect, thus an *aggregation strategy* must be defined, for instance by referring to established voting rules defined in the field of *computational social choice* such as Plurality vote,Borda Count, Kemeny[88] or Maximin[89].

If a voting mechanism does not suffice for the problem that has to be faced due to the complexity of the unstructured data generated, then the problem must be further analyzed and a tailored aggregation strategy defined. For instance, the problem of aggregating hand drawn shapes cannot be faced by using a voting strategy. The use of aggregation strategies permits to sacrifice the quality of the data with respect to quantity since the outliers are automatically dealt with.

### Task Matching

Task matching involves the analysis of the operations that have to be performed to solve the task and the identification of known game mechanics used in the gameplay of an already existing game that involve similar actions. The result obtained by performing an action during the game should produce the same results as if the player would perform the specified task manually, following the description of the task and applying the admissible operations, yet it does not have to be recognized as work but as an integral part of the gaming experience.

The matching phase is again a creative process that so far has never been extensively studied and depends on the experience of the designer; to search for the best match, works in literature and specialized websites that provide a list of existing games divided by genre, such as [90] have to be consulted. Possible game mechanics are defined in Section 5.3.

In particular, when referring to Games with a Purpose, three different patterns have been identified based on the players interaction and the winning condition: *input agreement*, *output agreement*, *inversion problem games* [66]. If a known game mechanic cannot be found, then a novel game has to be designed to propose the task to be solved as a conflict within the gameplay.

Once the game mechanics have been defined, a working prototype of the game must be created following the steps defined in 5.1, before committing to the integration phase or, if the existing game allows modifications or the source code is present, the integration will be performed on the existing project. A list of examples of possible tasks and the associated game mechanics or genre is provided, to give an example on possible ways in which a task can be matched:

**Image Labeling** Given an image, the users should provide a descriptive textual label as output. A possible matching conflict requires the player to reach an agreement by submitting the same label in order to score points. [67]

**POI Collection** The users are asked to define the location and provide pictures for the Points of Interest in their area. A possible matching mechanics involve resource acquisition, in which the players of the game are asked to take photos in the real world to become the owner of that particular location in a virtual world. [1]

**Protein Folding** Given as input a protein, return the folding structure needed to make it active. Folding rules can be used to define conflicts under the form of puzzles to be solved that require the users to reach a particular protein structure to progress into the game [91]

**Image clustering** Given as input a set of image, group them in clusters based on the similarity of the content. The game mechanics for the task could be borrowed from "memory"game genres in which the players are asked to match pair of images with the same portraied pictures.

### Data Model Design

The data model design involves the definition of schemas needed to define the input and output of the task to be performed, along with the ones used to represent the state of the game and its players.

### Architecture Design

It involves the definition of the hardware and software components that has to be used to create the game and the backend used to sustain the data necessary to accomplish the tasks.

### Task Integration

In this phase, the set of operations defined to solve a particular task are implemented within the game: the retrieval and visualization of input data within

[1] http://www.ingress.com/

the game are coded as the initial condition of a challenge to be solved during the gameplay, the admissible operations for the task are implemented as gameplay actions that a player could perform and the validation techniques on the provided output are coupled with the algorithms of the game to provide immediate feedback to the users. Output of this phase is a first prototype of the *GWAP* that will be further improved in subsequent iterations if it fails to solve the planned task or in case of balance issues or poor designed game mechanics.

### Task Results Evaluation

In this phase, it is verified if the output of the game maps properly to the particular inputs that were fed into it. Since this cannot be done automatically without having a ground truth to which the new results can be compared to, this check has to be performed with the help of human volunteers.

## 5.3 Game Mechanics Definition

Game mechanics represent the artificial conflicts and interaction means that are introduced in a GWAP or in a traditional game to drive the behaviors of players. One of the greatest issues that a GWAP designer could face is the difficulty in finding the right mechanics that have to be applied in a specific context. In [66] a list of structured templates for the design of GWAPS, namely input-agreement, output-agreement and inversion problem is defined. These templates alone, even though they are fundamental for what concerns the validation of the submitted results, are not sufficient for creating a gaming experience.

Defining a list of possible mechanics that have been used in traditional games could hint a novel designer on the available choices that she could exploit; for these reason based on the best practices described in [92][56], a list of possible game mechanics suitable to be applied to a GWAP is provided, along with examples.

- ***Agreement*** Players are requested to reach an agreement over a question or a topic based on some hints provided by the game. Agreement is one of the most widely used mechanics in GWAP, being the foundation on which templates like input-agreement and output-agreement rely on to be able to automatically validate the contributions of different players. The ESP game for instance requires two players to agree on the same submitted tag by using as the only hint a common image.

- ***Tile(Resource)-Placement*** Tile Placement games feature placing a piece to score points, with the amount often based on adjacent pieces or pieces in the same group/cluster, and keying off non-spatial properties like color, feature completion, cluster size etc. The visual nature of the mechanic is particularly suited for exploiting the capabilities of humans to visually identify patterns through abstraction and intuition. Placing multimedia assets that share some commonalities spatially near each other allows for easy human clustering tasks. In Phylo, players solve pattern-matching puzzles that represent nucleotide sequences of different phylogenetic taxonomies to optimize alignments over a computer algorithm.
- ***Line Drawing*** Games that make use of this mechanics involve drawing lines in one way or another. Line Drawing is a mechanic that allows to identify regions of interest in images and thus to solve object recognition problems. Squigl was a GWAP in which two users where asked to draw the contour of the same object and were judged based on how close their outlines were. Sketchness is a Draw-And-Guess game similar to Pictionary in which one player is given an image and an object to segment by drawing the contour, while the other players, without being able to see the image, have to guess the underlying object based just on the drawn contour.
- ***Memory*** Games that use the Memory mechanic require players to recall previous game events or information in order to reach an objective. By using just their memory, which is likely not to be able to recall all the details of a multimedia asset but just the salient features, the player can be asked to cluster assets by creating implicit mental relationship between object. FliptIt used this mechanic to cluster images that were portraying the same subjects. The players were presented with tiles hiding images and were requested to clear the board by pairing two tiles, picked up sequentially and removed just if the content of the image was the same or similar.

- ***Betting/Wagering*** Involves games that encourage or require players to bet resources or commodities on certain outcomes within the game. Often the values of the commodities are continually changing throughout the game, and the players buy and sell the commodities to make money off of their investment. No known GWAP make use of this mechanic so far, nonetheless it could help develop games involving preference elicitation or human judgment.
- ***Pattern Building*** Players place game components in specific patterns in order to gain specific or variable game results. The objective of FoldIt is to fold the structure of selected proteins as well as possible, using various tools provided within the game. The highest scoring solutions are analyzed by researchers, who determine whether or not there is a native structural configuration (or native state) that can be applied to the relevant proteins.
- ***Bluffing*** In games with the bluffing mechanic, players need to hide their true intent or actions by using bluff, lies or deceiving. In Disguise, a GWAP used to evaluate the capabilities of different color blending algorithms, some of the enemies in the game are semi-transparent in order to disguise themselves among useful resources; it is duty of the player to exploit her perceptions to identify the intruders.
- ***Trivia*** Games that make players answer questions based on their knowledge. In Verbosity, one player is giving textual clues related to a particular word or subject to be guessed by the other player, in order to obtain meaningful semantic annotations.
- ***Area Enclosure*** Players try to surround or reveal an area to score points or to gain other advantages. Similarly to Line Drawing, this mechanics allows to identify regions of interest in images. PeekABoom used this techniques by allowing one player to unveil part of an hidden image that contained salient information regarding the object within the image that another player was asked to identify. The least the area unveiled, the more were the points received by the first player. The traces submitted by the players are aggregated in order to build bounding boxes identifying the position of a particular object.

| Task Type | Category | Task Description | Human Performer Operations |
|---|---|---|---|
| Object Recognition<br>Object Identification<br>Object Detection | Decision<br>Generative | Recognize one or several pre-specified or learned objects together with their 2D positions in the image or 3D poses in the scene. Recognize an individual instance of an object. Recognize specific condition or anomalies | Given a specific object, identify it in the image or environment with an annotation which selects a subset of samples with a particular meaning. Their representation depends on the context and the dimension of the space that is being considered. (2D,3D, 4D...) |
| Clustering | Decision | Task of grouping a set of objects in such a way that objects in the same group (called cluster) are more similar (in some sense or another) to each other than to those in other groups (clusters). | Define a (subjective) similarity measure to compare the input data with and group objects into clusters based on it. |
| Ordering | Decision | Arranging items of the same kind, class, nature, etc. in some ordered sequence, based on a particular criteria. | Define a (subjective) evaluation criteria to compare the input data and order the objects based on the chosen criteria. |
| Natural Language Processing (NLP) | Decision<br>Generative | Performing various operations related to natural language understanding and manipulation, such as Summarizing, Question answering, Sentiment Analysis, Speech recognition... | Performing the requested tasks by exploiting humans' ability to understand natural language |
| State Exploration | Decision | Problems for which the solution can be measured and evaluated but exploring the whole solution space is intractable. Exploring the set of all possible points of an optimization problem that satisfy the problem's constraints, potentially including inequalities, equalities, and integer constraints, to obtain the best solution. | Intuitively recognize optimization patterns that may lead to the best solution for the problem at hand |
| Content Generation<br>Content Submission | Generative | Generating novel content for the problem at hand, respecting the constraints or providing content based on particular requests | Use one's own ability to generate the requested content or choosing the best content to be provided based on personal judgment |
| User Preferences<br>Opinion Elicitation | Decision<br>Generative | Gathering synthesis of opinions of authorities of a subject where there is uncertainty | Submit an opinion or a preference related to a particular topic |

Table 5.1: Most meaningful multimedia refinement tasks for GWAPs

## 5.4 Tasks and Mechanics matching

A Human Computation Task (HCT) is a “unit of work” assigned to a user of a Human Computation system; removing duplicates or inappropriate content, cropping the silhouette of the models in a picture or recognizing and identifying the people contained in a set of images are examples of what a task involving a GWAP's player may look like and its goals may greatly vary based on the business objectives that have to be met. A generic task can be designed by specifying the different components shown in figure 5.4.

A task is defined by a *description*, generally a textual one, of the goal that has to be reached and a set of admissible operations that represent the mean with which this particular task can be accomplished by the user. Usually a

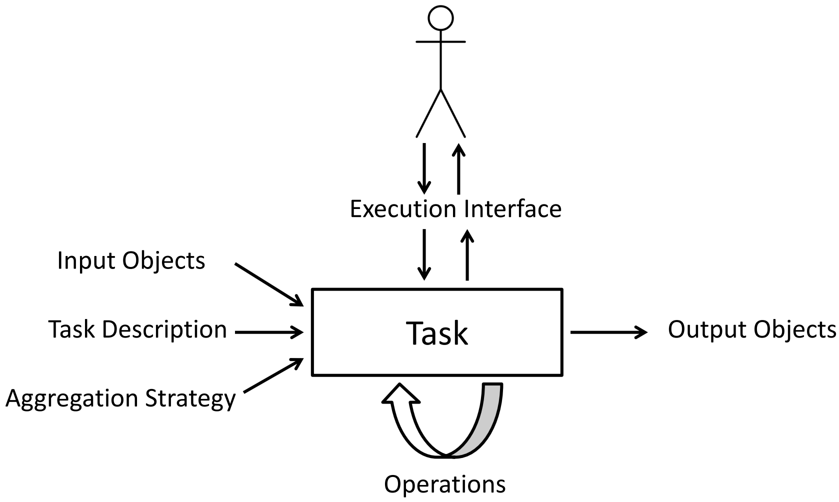



Figure 5.4: Components of a Task

HCT is created with the purpose of creating or modifying multimedia content or its annotations. For this reason, a task has to be defined not only based on the operations that can be performed but also on the data that will be manipulated and produced, the input and output objects, that may be represented as structured data or multimedia content. Depending on the specific nature of the task, it may also be useful to have a certain number of users of the platform to perform the same task several times, to achieve the redundancy needed to overcome inaccurate responses or personal biases. In such a case, the output objects may require further processing to be able to retrieve a meaningful result and thus it may be necessary to define an *aggregation strategy* associated to the task, like ranking, clustering or majority voting[93]. The *operations* are the activity that the user is requested to perform in order to accomplish a specific objective. By analyzing existing works in literature [70] and typical AI hard problems[12], the most common operations that could be performed effectively by human players within a GWAP have been collected, taking into consideration the context of multimedia meta-data refinement, and have been reported in Table 5.1. They may fall into two broad categories:

- *Generative Tasks* include tasks which aim at generating new artifacts as the solution of the problem at hand, e.g. Labeling, Segmenting, Ontology linking...
- *Decision Tasks* are related to decisions that the users has to perform over already existing data, e.g. Preferences elicitation, Ordering...

| **Game Mechanic** | **Task Type** | **Significative Examples** |
|---|---|---|
| Agreement | Object recognition, Clustering, Ordering, NLP | ESP Game, TagATune |
| Tile(Resource)-Placement | Clustering, Ordering | Phylo |
| Line Drawing | Object Recognition | Sketchness, Squigl |
| Memory | Clustering | FlipIt |
| Betting/Wagering | User Preferences/Opinion Elicitation | N/A |
| Pattern Building | State Exploration | FoldIt, Eyewire |
| Bluffing | Ordering, Object Identification | Disguise, SearchWar |
| Trivia | Natural Language Processing | Verbosity, WebPardy |
| Area Enclosure | Object Recognition | PeekABoom, Ask'nSeek |

Table 5.2: Game Mechanics to Task Type matching

Choosing the right mechanics and the possible instantiations that have to be applied in a specific context is the central task of gamification design and it is a matter of experience of the designer of the application. The mechanics presented in 5.3 can be combined together to produce a variety of experiences that form the structure of what can be recognized as a game; the list is not exhaustive, since game design is fundamentally a creative process, but can be seen as a starting point able to cover most of the experiences provided by the GWAP that has been developed so far. The results are shown in Table 5.2, in which the mechanics are paired with the tasks most suited to them.

## 5.5 Validation Templates

As described by Wang et al. [94] one of the great problems related to human computation is the adversarial behavior of some users. In this situations users do not execute the required tasks or feed the system with malicious data. This can be due to many reasons.

- **Maximize Rewards**

  Users of active systems are payed at the completion of the task, while passive systems ones gain points and reputation. In this situation malicious users try to complete the largest number of tasks in the least amount of time, simply by randomly answering or trying to fool the system and complete the task in a fast way.

- **Real Life Advantages**

  In even worse situations users try to poison data in order to gain advantages in the real life. An example are false reviews in recommendation systems.

- **Just For Fun**

  In *Sketchness* [95] (a GWAP designed to assign labels to localized regions in an image, described in 9), users write obscene phrases, instead of fulfilling the task.

As the main purpose of GWAP is the retrieval of meaningful data as a contribution from its players, even though the game itself has not the capability of inspecting the data to verify if the submission is a valid one (otherwise there would be no need for human contribution, since the game would be able to solve the problem on its own), to avoid bad quality annotations a GWAPs rules should encourage players to correctly perform the necessary steps to solve the computational problem and, if possible, involve a probabilistic guarantee that the games output is correct, even if the players do not want it to be correct. This goal can be achieved in two different steps:

- Designing game mechanics that, beforehand, reduce the possibility for the players to submit inconsistent data

- Estimating, offline, object features trough the aggregation of such crowdsourcing annotations, even with the presence of a high number of malicious users.

In Section 5.5.1 are described the validation mechanics that can be used, within the game, to try to ensure correctness of the submitted annotations.
In Section 5.6 are presented various techniques used to solve the adversarial behavior by redundant annotations.
Finally in Section 5.7 a novel aggregation technique is detailed, able to reduce up to 70% the number of annotations required to reach a given accuracy level and how it gives acceptable results even with up to 75% of malicious users.

### 5.5.1 Validation Mechanisms Templates

Most works on GWAPs focus on embedding a speciifc problem solving task into an enjoyable user experience and on evaluating the quality and quantity of output produced by players. The classification of alternative game design patterns, based on different input-output templates, discussed in [66], is the first attempt to generalize GWAP design principles. Three are the main validation mechanisms that are described, namely Output Agreement, Input Agreement and Inversion Problem games.

#### Output Agreement

In this mechanism, two players with the same role share a same input (typically, an image or a tune) and are requested to produce some description of their common input: the objective is to reach an agreement as quickly as possible, by submitting the same descriptive label. Output agreement games are useful for annotating multimedia content assets, as humans are induced to produce semantic annotations that describe as accurately as possible the input.

The rules of the game states that that players should try to produce the same output as their partners; it is usually not required for players to produce it at the same time, but within a timespan that typical represent the time of one "round"a portion of the entire gaming session. Players cannot see one anothers outputs or communicate with one another, thus the easiest way for both to produce the same output is by entering something related to the common input.

This mechanism is able to accomplish several goals: the best strategy for the players is to produce outputs related to the only shared hint, that is their common input. Given the fact that the output domain is typically very large and unique for all the participants, finding a consensus among the players partially verifies that the submitted output is correct. The most significative example of GWAP that uses this mechanism is the ESP Game [67]

### Input Agreement

In this mechanism, two players with the same role are given inputs (e.g., a tune) that are known by the game, but not by them, to be the same or different. Players are requested to describe the object they received and guess whether the objects assigned by the system coincide; they see only each other's outputs and the round terminates when both players correctly determine whether they have been given the same item or not. It is in the players' interest to provide the best accurate outputs to describe their individual inputs with the tools at their disposal (may it a textual description or a selection among a limited domain of entries) to achieve the winning condition.

The mechanism is particularly suited to induce a variance in the annotations provided by the players: since they are not required to agree on their own annotations, players are incentivized to submit rich and precise descriptions of the object under their scrutiny, while mechanisms like Output Agreement have at their core the submission of the most common and easily recognizable output as their best strategy. On the other hand, since the response on which the players have to agree upon is binary, this allows them to try a random guessing strategy and for this reason, mechanisms to strongly penalize incorrect guesses must be introduced (e.g. a suitable scoring system). The most significative example of GWAP that uses this mechanism is Tagatune [96]

### Inversion Problem

In this mechanism at each round, one player assumes the role of the "describer"and the other one that of the "guesser". The describer receives an input (e.g, an image, or a word) and based on it, sends suggestions to the guesser to help her identify some feature of the original input. The goal is, for the guesser, to produce the input that was originally given to the describer. Inversion problem games take advantage of both of the strenghts of Output Agreement and Input Agreement: the players are successful in reaching the goal of the game if the describer has been able to provide enough outputs for the guesser to guess the original input, thus the structure encourages players to submit correct information, that do not have to be limited to the most trivial ones. Blind guessing is also hard to accomplish, due to the fact that if the domain to which the input belongs to is considerably large, finding the solution without relying on sound hints provided by the describer would be a daunting task.

Inversion Problem games may allow the describers to see the replies of the guessers, to make the role less boring and to allow promptly corrections in the submitted annotations and they usually alternate the roles of the players among rounds, to give the possibility to all the users involved to submit annotations or validate them. The most significative example of GWAP that uses this mechanism is Verbosity [21]

## 5.6 Aggregation Techniques

In this Section various techniques used to solve the adversarial behavior by redundant annotations are presented. These techniques aggregate the annotations in different ways in order to obtain better results.

Many of these techniques are tailored to binary annotations, labeling or classification. The algorithm proposed in Section 5.7 is not tailored to a specific kind of annotation even though some of the techniques we are going to present in this Chapter can be seen as a particular instantiation of it.

In literature we can identify two main classes of methodologies:

1. **Non-iterative**

   uses heuristics to compute a single aggregated value of each question separately [97]. Examples of these techniques are **majority voting** (see Section 5.6.1) and **a priori quality checking** (see Section 5.6.3).

2. **Iterative**

   performs a series of iterations, each consisting of two updating steps: (i) updates the aggregated value of each question based on the expertise of workers who answer that question, and (ii) adjusts the expertise of each worker based on the answers given by him [97]. Examples of these techniques are **expectation maximization** (see Section 5.6.4) and **iterative learning** (see Section 5.6.5).

### 5.6.1 Majority Voting

One of the most simple techniques used to solve the problem is majority voting. It is also known as **majority decision**.

> *"Majority Decision (MD) is a straightforward method that aggregates each object independently. Given an object $o_i$, among $k$ received answers for $o_i$, we count the number of answers for each possible label $l_z$. The probability $P(X_i = l_z)$ of a label $l_z$ is the percentage of its count over $k$; i.e. $P(X_i = l_z) = \frac{1}{k}\sum_{kj=1}^{k} \mathbf{1}_{a_{i,j}=l_z}$. However, MD does not take into account the fact that workers might have different levels of expertise and it is especially problematic if most of them are spammers."*
>
> Hung at al. [97]

This method is based on mainly two assumptions:

1. The number of cheaters is less than the number of good annotators.
2. A great number of annotations per object is available.

The two assumptions are required to have such a high probability that the consensus of the users is the right one.

#### Pros

- If the assumptions are respected it generally gives good results.
- Does not require complex aggregation algorithms.
- Does not require any knowledge about the user related to the annotation.
- Does not require any knowledge about the dataset.

**Cons**

- It requires a strong assumption with respect to the number of good users.

As described by Sheng at al. [98] this technique is mainly used in binary or classification tasks.

The binary task consists in choosing between two possible answers YES or NO; once the annotations from the users have been collected, it is just required to choose the answer that has the greatest consensus among them.

The classification task consists in choosing one class from a set of possible classes, once the annotations from the users have been collected, it is just required to choose the class that has the greatest consensus among them.

As explained in [98], majority voting does perform well when the probability $p$ of obtaining the right answer from a single users is greater that 50%. In this situation the probability of obtaining the right answer using majority voting increases with the number of users, the higher is $p$ the faster it tends to 100%. On the contrary when $p$ is less than 50% majority voting fails. In this situation the probability of obtaining the right answer decreases when the number of users increases, the lower is $p$ the faster it tends to 0.

These considerations were done under the assumption that all the users had the same quality (probability to give a good answer); in [98] the situation in which users with different quality are used leads more or less to the same results.

In [98] another extension of the method is presented for a particular class of classification called "soft" labeling that obtains better results due to the multiset nature of the annotation.

Okubo at al. [99] present a small variation of majority voting that exploits information coming from previous answers in order to assign tasks to more thrustfull users. After the assignment, the annotations are aggregated in the exact same way as normal majority voting. Even though this version of the algorithm obtains better results it requires more knowledge related to users and the dataset, knowledge that is not always available.

Tsai at al. [100] present a variation of majority voting that requires the users to communicate in order to reach a consensus before assigning the final annotation. It obtains good result when the users engage in a profitable debate.

### 5.6.2 Honeypot

The technique proposed by Lee at al. [101] and extended to the aggregation case by Hung at al [97] is in between majority voting and a priory quality checking.

It uses a technique coming from the computer security field that is commonly used to identify malicious agents and avoid attacks.

> *"In principle, Honeypot (HP) operates as MD, except that untrustworthy workers are filtered in a preprocessing step. In this step, HP merges a set of trapping questions $\Omega$ (whose true answer is already known) into original questions randomly. Workers who fail to answer a specified number of trapping questions are neglected as spammers and removed. Then, the probability of a possible label assigned for each object oi is computed by MD among remaining workers. However, this approach has some disadvantages: $\Omega$ is not always available or is often constructed subjectively; i.e truthful workers might be misidentified as spammers if trapping questions are too difficult."*
>
> Hung at al. [97]

### 5.6.3 A priori quality check

Another technique used to solve the problem is to do an a priori quality check. Also known as *majority voting with gold standard* or *expert label injected crowd estimation.*

> *"Expert Label Injected Crowd Estimation (ELICE) is an extension of HP. Similarly, ELICE also uses trapping questions $\Omega$, but to estimate the expertise level of each worker by measuring the ratio of his answers which are identical to true answers of $\Omega$."*
>
> Quoc Viet Hung at al. [97]

Given the expertise level of each worker it is possible to weight differently the different workers. It allows to filter out random annotators (not reliable) and even exploit spammers (always give the wrong answer) by negatively weighting them.

This approach generally gives better results than majority voting as demonstrated by Vuurens at al. [102].

An example can be found in [103] where NLP tasks has been assigned to a crowd of non-experts. In this paper it has been used a gold standard coming from experts in order to evaluate the quality of the crowd.

This method allows to obtains even better result by further analysis.

> *"It estimates the difficulty level of each question by the expected number of workers who correctly answer a specified number of the trapping questions. Finally it computes the object probability $P(X_i = l_z)$ by logistic regression that is widely applied in machine learning. In brief, ELICE considers not only the worker expertise ($\alpha \in [1, 1]$) but also the question difficulty ($\beta \in [0, 1]$). The benefit is that each answer is weighted by the worker expertise and the question difficulty; and thus, the object probability $P(X_i = l_z)$ is well-adjusted. However, ELICE also has the same disadvantages about the trapping set $\Omega$ like HP as previously described."*
>
> Hung at al. [97]

### Pros

- Good performance.
- Robust against random and malicious annotators.

### Cons

- Requires a ground-truth of sufficient size in order to estimate correctly the goodness/expertise of the annotators.
- Requires the ability to inject the ground-truth inside the normal workflow.
- Requires a method to uniquely identify the user that has generated an annotation.

- Requires a greater number of annotations with respect to other methods, because some of them are not directly used in the aggregation, they are just used to estimate the user goodness/expertise.

  This requires more time, and higher costs if it is used with a paid crowdsourcing system.

Ertekin at al. [104] propose a modified version of a priori quality checking that allows to reduce the required annotations. In this version the tasks are assigned to just a subset of the crowd, this subset is identified at runtime.

### 5.6.4 Expectation Maximization

Expectation maximization is an approach based on a probabilistic model, as presented by Dempster at al. [105] and Whitehill at al. [106].

> *"The Expectation Maximization (EM) technique iteratively computes object probabilities in two steps: expectation (E) and maximization (M). In the (E) step, object probabilities are estimated by weighting the answers of workers according to the current estimates of their expertise. In the (M) step, EM re-estimates the expertise of workers based on the current probability of each object. This iteration is repeated until all object probabilities are unchanged. Briefly, EM is an iterative algorithm that aggregates many objects at the same time. Since it takes a lot of steps to reach convergence, running time is a critical issue."*
>
> Hung at al. [97]

This method outperforms a priori quality checking and is more robust to the presence of spammers as demonstrated by Vuurens at al. [107] and Raykar at al. [108] even though it is sensible to the initialization. Different starting points can lead to different solutions.

#### Pros

- Does not require a ground-truth.
- Robust against random and malicious annotators.

**Cons**

- Sensible to starting point.
- Requires a method to uniquely identify the user that has generated an annotation.
- Iterative and therefore computational heavy

A similar technique for annotator quality estimation is proposed by Ipeirotis at al. [109]. It has been tailored to multiple choice question and uses “soft” labels instead of hard ones during the estimation of both object probability and worker quality score.

> *“The score separates the intrinsic error rate from the bias of the worker, allowing for more reliable quality estimation. This also leads to more fair treatment of the workers.”*
>
> Ipeirotis at al. [109]

### 5.6.5 Iterative Learning

As explained by Kerger at al. [110], Iterative Learning is a belief-propagation-based method for annotation aggregation.

As suggested by Hung at al. [97] it can be even used to estimate question difficulty.

> *“Iterative Learning (ITER) is an iterative technique based on standard belief propagation. It also estimates the question difficulty and the worker expertise, but slightly different in details. While others treat the reliability of all answers of one worker as a single value (i.e. worker expertise), ITER computes the reliability of each answer separately. And the difficulty level of each question is also computed individually for each worker. As a result, the expertise of each worker is estimated as the sum of the reliability of his answers weighted by the difficulty of associated questions. One advantage of ITER is that it does not depend on the initialization of model parameters (answer reliability, question difficulty). Moreover, while*

*other techniques often assume workers must answer all questions, ITER can divide questions into different subsets and the outputs of these subsets are propagated in the end.”*

Hung at al. [97]

**Pros**

- Does not require a ground-truth.
- Robust against random and malicious annotators.
- Simpler model with respect to *expectation maximization* and *belief propagation.*

**Cons**

- Requires a method to uniquely identify the user that has generated an annotation.
- Iterative and therefore computational heavy

The algorithm presented in Section 5.7 is based on this approach.

## 5.7 General Purpose Aggregation

The aim of this Chapter is to present and formally define the proposed algorithm. In the first part we will propose a general framework for annotation aggregation regardless to the kind of annotation. This framework is a generalization of the one proposed by Karger et al. [111] and presented in [112], we will go beyond the specific case and try to identify a general version. In the second one we will analyze a particular kind of annotation, that is aggregation of the Binary Vectors that has been used to aggregate ROI for the GWAP presented in Chapter 9. For each of them we will take in account common/naïve algorithms and an instantiation of the proposed algorithm.

### 5.7.1 Preliminaries

Let $\mathcal{O}$ be a set of objects, let $O_i$ denote the $i$-th object in the set, and let $F_i \in \mathcal{F}$ be a feature associated to this object where $\mathcal{F}$ is the space on which the feature is defined.

Let $\mathcal{A}$ be a set of users, called annotators, let $a_j$ denote the $j$-th annotator in the set and $F_{i,j} \in \mathcal{F}$ the annotation provided by annotator $a_j$ for the feature $F_i$ of the object $O_i$.

Let $\mathcal{A}_i$ denote the set of the annotators who provided an annotation for the object $O_i$.

Similarly, let $\mathcal{O}_j$ denote the set of objects annotated by $a_j$.

Under ideal circumstances, $F_{i,j} = F_i$. However, due to noise intrinsic in the annotation process, $F_{i,j} \neq F_i$, this require to aggregate the annotations coming from more users, in order to reduce/eliminate the noise.

Let $\hat{F}_i \in \mathcal{F}$ denote the aggregated annotation for the feature $F_i$.

The goal of the algorithm is to minimize the distance (or maximize the similarity) between the real feature $F_i$ and the estimate $\hat{F}_i$.

### 5.7.2 General Algorithm

In order to find the estimate $\hat{F}_i$, the available annotations need to be aggregated.

Depending on the specific kind of annotation there are already known algorithms generally based on the computation of an average or a median, like majority voting, which share a common property, they assign the same weight to all the annotations.

Our algorithm goes beyond by assigning different weights to the different annotations. That is:

$$\hat{F}_i = f(\{\langle F_{i,j}, w_{i,j}\rangle | a_j \in \mathcal{A}_i\}) \tag{5.1}$$

The weights $w_{i,j} \in \mathcal{W}$, $\mathcal{W} \equiv [0, 1] \subset \mathbb{R}$ capture the quality of annotator $a_j$ to annotate object $O_i$. The challenging aspect lies in how to automatically determine these weights without any prior knowledge about the quality of the annotators.

To this end, we propose an iterative algorithm that is able to accomplish this task while relying only on the available annotations.

Following an approach similar to [111], the algorithm seeks the solution iteratively by alternating two steps:

- For each object $O_i$, given the available annotations $F_{i,j}$, $a_j \in \mathcal{A}_i$, and some knowledge about the reliability of each annotator $w_{i,j}^{(k)}$ available at iteration $k$, compute $|\mathcal{A}_i|$ different estimates $\hat{F}_{i,j}^{(k)}$, $a_j \in \mathcal{A}_i$. Each estimate is obtained by aggregating all annotations but the one given by $a_j$. That is:

$$\hat{F}_{i,j}^{(k)} = f(\{\langle F_{i,j'}, w_{i,j'}^{(k)} \rangle | a_{j'} \in \mathcal{A}_i \setminus \{a_j\}\}) \tag{5.2}$$

  where $f()$ is an *aggregation function* that computes a weighted consensus among the available annotations.

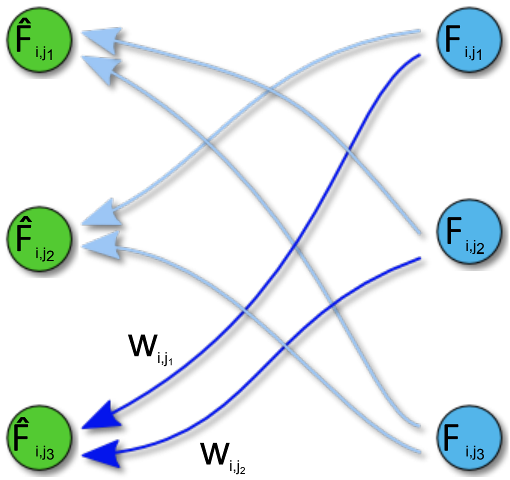



Figure 5.5: Aggregation Step

- For each annotator $a_j$, given the available annotations $F_{i,j}$, $O_i \in \mathcal{O}_j$, and the current estimate $\hat{F}_{i,j}^{(k)}$, compute $w_{i,j}^{(k+1)}$, i.e., the quality in annotating each object $O_i$, by measuring the coherence between the annotation $F_{i,j}$, and the current estimate $\hat{F}_{i,j}^{(k)}$ obtained by using all the annotations but the one related to $O_i$. That is:

$$w_{i,j}^{(k+1)} = g(\{\langle F_{i',j}, \hat{F}_{i',j}^{(k)} \rangle | O_{i'} \in \mathcal{O}_j \setminus \{O_i\}\}) \tag{5.3}$$

  where $g()$ is a *coherence function* that given a set of pairs $\langle F_{i',j}, \hat{F}_{i',j}^{(k)} \rangle$ computes the weight associated to the annotation $F_{i,j}$.

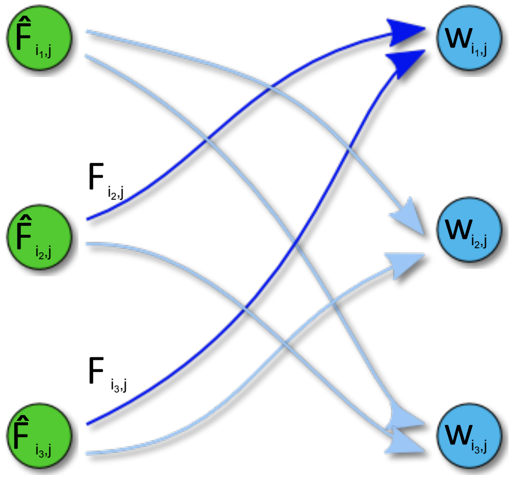



Figure 5.6: Coherence Estimation Step

Note that the description of the algorithm is general and that it does not impose any constraint on the nature of the feature $F_i$ for which the annotations are available. Indeed, the only requirement is the possibility to specify:

*i)* an aggregation function $f()$ defined in Equation (5.2)

*ii)* a coherence function $g()$ defined in Equation (5.3)

Therefore, the proposed algorithm significantly extends the original work in [111], which was specifically tailored to work with features associated with a single binary label, whereas we are able to deal with objects that are associated with features of any kind.

The algorithms iteratively executes the two steps until the relative change of the weights falls below a threshold $\tau$, i.e.:

$$\frac{\sum_{i,j} |w_{i,j}^{(k+1)} - w_{i,j}^{(k)}|}{\sum_{i,j} |w_{i,j}^{(k)}|} < \tau \tag{5.4}$$

In our experiments we set $\tau = 10^{-6}$, and the algorithm converged in 6-7 iterations on average. The theoretical analysis of the convergence properties of the algorithm is left to future work. Upon convergence, the final estimate of the feature is computed according to Equation (5.1), in which the weights are set equal to those computed in the last iteration of the algorithm.

### 5.7.3 Aggregation Function

In Section 5.7.2 we have presented the aggregation function $f()$ that more rigorously can be defined as:

$$f : (\mathcal{F} \times W)^{|\mathcal{S}_i|} \rightarrow \mathcal{F} \tag{5.5}$$

where $|\mathcal{S}_i|$ is the size of the input set.

In many situations $f()$ can be rewritten in the following way

$$\begin{aligned} f(\mathcal{S}_i) &= m'(\varphi(\mathcal{S}'_i)) \\ \mathcal{S}'_i &= \{\langle x_{i,j}, w_{i,j}\rangle | x_{i,j} = m(F_{i,j}), \langle F_{i,j}, w_{i,j}\rangle \in \mathcal{S}_i\} \\ \mathcal{S}_i &= \{\langle F_{i,j}, w_{i,j}\rangle | a_j \in \mathcal{A}_i\} \end{aligned} \tag{5.6}$$

where:

- $m()$ and $m'()$ are two mapping function that allow to map an item of the feature space to and from an item of a convenient intermediate space $\mathcal{X}$ where linear operations can be defined (ex: $\mathbb{R}^n$). Their rigorously definitions are:

$$m : \mathcal{F} \rightarrow \mathcal{X} \tag{5.7}$$

$$m' : \mathcal{X} \rightarrow \mathcal{F} \tag{5.8}$$

- $\varphi()$ is an aggregation function that works in the intermediate space $\mathcal{X}$. Its rigorously definitions is:

$$\varphi : (\mathcal{X} \times \mathcal{W})^{|\mathcal{S}_i|'} \rightarrow \mathcal{X} \tag{5.9}$$

Under these assumptions we can replace $\varphi()$, that is defined over $\mathcal{X}$, where linear operations exist, with a weighted average of the mapped annotations:

$$\varphi(\mathcal{S}'_i) = \frac{\sum_{\langle x_{i,j}, w_{i,j}\rangle \in \mathcal{S}'} w_{i,j} \cdot x_{i,j}}{\sum_{\langle x_{i,j}, w_{i,j}\rangle \in \mathcal{S}'} w_{i,j}} \tag{5.10}$$

### 5.7.4 Coherence Function

In Section 5.7.2 we have presented the coherence function $g()$ that more rigorously can be defined as:

$$g : (\mathcal{F} \times \mathcal{F})^{|\mathcal{S}_j|} \rightarrow \mathcal{W} \tag{5.11}$$

where $|\mathcal{S}_j|$ is the size of the input set.

In many situations this definition of $g()$ can be too general. Often $g()$ can be defined as the average of the coherence computed on the pairs $\langle F_{i,j}, \hat{F}_{i,j} \rangle$. In this situation $g()$ becomes:

$$g(\mathcal{S}_j) = \frac{1}{|\mathcal{S}_j|} \sum_{s \in \mathcal{S}_j} \sigma(s) \tag{5.12}$$

$$\mathcal{S}_j = \{ \langle F_{i',j}, \hat{F}_{i',j} \rangle | O_{i'} \in \mathcal{O}_j \setminus \{O_i\} \}$$

where $\sigma()$ is a function that computes the coherence of a single pair. More rigorously it can be defined as:

$$\sigma : \mathcal{F} \times \mathcal{F} \to \mathcal{W} \tag{5.13}$$

### 5.7.5 Binary Vector

Under this kind of annotation can be grouped all the ones that can be represented as a stream of bits.

We will analyze in the specific case the Regions Of Interest (ROIs) in an image.

#### Feature Space

For binary vector annotations the feature space is:

$$\mathcal{F} = \{-1, +1\}^N \tag{5.14}$$

In the specific case of the ROIs $N = r \cdot c$ the number of pixels in the image. Every item in the bit stream represents that the corresponding pixel in the image is part of the ROI or not.

#### Common Aggregation Algorithm

For binary vector annotations the most common aggregation algorithm is **majority voting**, which formal definition is:

$$\hat{F}_i = \text{sign} \left[ \frac{1}{|\mathcal{A}_i|} \sum_{a_j \in \mathcal{A}_i} F_{i,j} \right] \tag{5.15}$$

where $\text{sign}(x) = \pm 1$, depending on the sign of $x$, and we arbitrarily set $\text{sign}(0) = +1$ to break ties.

#### Proposed Aggregation Function

For binary vector annotations we choose to use a modified version of majority voting, known as (**thresholded**) **weighted average**, executed on each item of the vector, which formal definition is:

$$\hat{F}_i = \text{sign}\left[\frac{\sum_{a_j \in \mathcal{A}_i} w_{i,j} \cdot F_{i,j}}{\sum_{a_j \in \mathcal{A}_i} w_{i,j}}\right] \tag{5.16}$$

Under the framework proposed in Section 5.7.3 this can be defined even in the following way

$$\begin{aligned} \mathcal{X} &= [-1, 1] \subset \mathbb{R} \\ m(F) &= F \\ m'(x) &= \text{sign}(x), x \in \mathcal{X} \end{aligned} \tag{5.17}$$

#### Proposed Coherence Function

For binary vector annotations and in the specific case ROIs we choose to use the framework presented in Section 5.7.4 and define $\sigma()$ using the **Jaccard's similarity** proposed by Paul Jaccard in [113] [114]:

$$\sigma(F_1, F_2) = \frac{|\{x | F_1(x) = +1 \wedge F_2(x) = +1\}|}{|\{x | F_1(x) = +1 \vee F_2(x) = +1\}|} \tag{5.18}$$

### 5.7.6 Real Vector

Under this kind of annotation can be grouped all the ones that are based on a vector of real numbers, like a key-point descriptor or a bounding box.

#### Feature Space

The real vector feature space can be defined as a set of vectors composed by $N$ natural or real numbers:

$$\mathcal{F} = \mathbb{R}^N \tag{5.19}$$

#### Common Aggregation Algorithm

Two of the most common aggregation algorithms are the **average** and the **median**.

- **Average**

  Which formal definition is:

$$\hat{F}_i = \frac{\sum_{a_j \in \mathcal{A}_i} F_{i,j}}{|\mathcal{A}_i|} \tag{5.20}$$

  **Pros**

  - Easy to implement
  - Easy to parallelize
  - Linear complexity in the number of annotations

  **Cons**

  - High sensitivity to outliers (spammers)

- **Median**

  In the particular we use a median or each component of the vector:

$$\hat{F}_i(x) = \texttt{median}\{F_{i,j}(x) | a_j \in \mathcal{A}_i\} \tag{5.21}$$

  **Pros**

  - Low sensibility to outliers (spammers)

  **Cons**

  - Not linear complexity in the number of annotations

### Proposed Aggregation Function

As aggregation function we have chosen to use two modified versions of the previously proposed algorithms that take in account the quality of the annotations.

- **Weighted Average**

  The weighted average takes in account the quality of the annotations weighting them in a different way:

$$\hat{F}_i = \frac{\sum_{a_j \in \mathcal{A}_i} w_{i,j} \cdot F_{i,j}}{\sum_{a_j \in \mathcal{A}_i} w_{i,j}} \tag{5.22}$$

Under the framework proposed in Section 5.7.3 this can be defined even in the following way:

$$\mathcal{X} = \mathbb{R}^N \tag{5.23}$$
$$m(F) = F$$
$$m'(x) = x$$

- **Weighted Median**

  As you can see from Figure 5.7 the weighted median is a modified version of the median that weights the elements in a different way as proposed by Edgeworth, F.Y in[**edgeworth1888new**]:

$$\hat{F}_i = \texttt{wmedian}\{\langle F_{i,j}, w_{i,j}\rangle | a_j \in \mathcal{A}_i\} \tag{5.24}$$

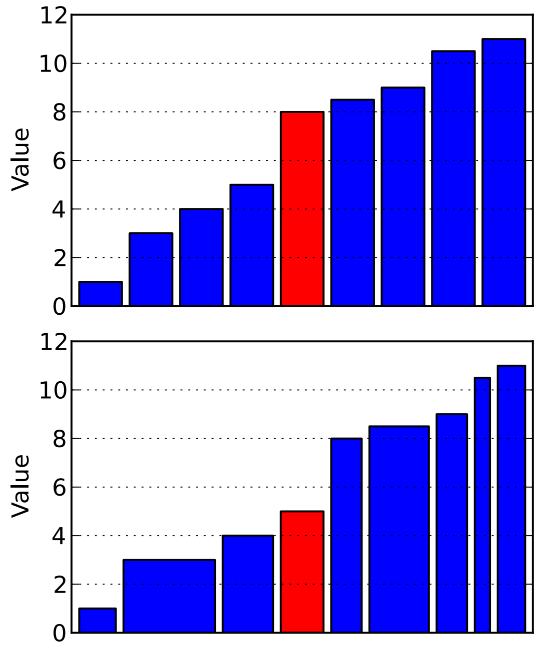



Figure 5.7: Regular Media vs. Weighted Median [**WMedian**]

### Proposed Coherence Function

As coherence function we have chosen to follow the framework proposed in Section 5.7.4 using a $\sigma()$ function based on the **Chebyshev distance** proposed by James Abello et al. [**Abello:2002:HMD:779232**]:

$$D_{Chebyshev}(p, q) = \max(|p_i - q_i|) \tag{5.25}$$

The Chebyshev distance gives a value in $\{x|x \geq 0 \wedge x \in \mathbb{R}\}$ this value can be mapped in a value of the space used by our framework (see Section 5.7.2) in the following way:

$$\sigma(\langle F, \hat{F} \rangle) = \frac{1}{1 - D_{Chebyshev}(F, \hat{F})} \tag{5.26}$$

### 5.7.7 Ranking

Under this kind of annotation can be grouped all the ones that are based on the **ranking/sorting** of a finite set of items.

#### Feature Space

The ranking feature space can be defined as a set of vectors composed by the first $N$ natural numbers without repetition:

$$\mathcal{F} = \{\langle I_1, ...I_N \rangle | N \in \mathbb{N}^+ \wedge \forall_{k \in [1,N]} I_k \leq N \wedge (\forall_{k,l \in [1,N]} k \neq l \iff I_k \neq I_l)\} \subset {\mathbb{N}^+}^N \tag{5.27}$$

#### Common Aggregation Algorithm

The most common ranking aggregation algorithm is the **median rank aggregation** presented by Ronal Fagin et al. [**Fagin:2003:ESS:872757.872795**].

Let $\rho(I_k, F_{i,j})$ be the location given by annotator $a_j$ to item $I_k$ in $F_{i,j}$.

We compute $\mu_i'(I_k)$ as the median over all the $\rho(I_k, F_{i,j})$:

$$\mu_i'(I_k) = \texttt{median}(\{\rho(I_k, F_{i,j})|a_j \in \mathcal{A}_i\}), k \in [1, N] \tag{5.28}$$

Ordering the $\mu_i'(I_k)$ we can obtain a permutation $\mu_i$ that can be used to create the estimate $\hat{F}_i$

$$\begin{aligned} \mu &=< \mu_i'(I_k), ..., \mu_i'(I_l) > |\forall_{k,l \in [1,N]} k < l \iff \mu_i'(I_k) \leq \mu_i'(I_l) \\ \hat{F}_i &=< I_k, ..., I_l > |\forall_{k,l \in [1,N]} k < l \iff \mu_i'(I_k) \leq \mu_i'(I_l) \end{aligned} \tag{5.29}$$

#### Proposed Aggregation Function

As aggregation function we have chosen to use a modified version of the **median rank aggregation**, that instead of using a simple median use a

**weighted median**. The only step in the algorithm that changes is Equation (5.28) that becomes:

$$\mu_i'(I_k) = \texttt{wmedian}(\{\langle \rho(I_k, F_{i,j}), w_{i,j}\rangle | a_j \in \mathcal{A}_i\}), k \in [1, N] \tag{5.30}$$

### Proposed Coherence Function

As coherence function we have chosen to follow the framework proposed in Section 5.7.4 using a modified version of the **Spearman's rank correlation coefficient** to compare the ranking pairs, as proposed by Charles Spearman [115].

$$\begin{aligned} s(\langle F,\hat{F}\rangle) &= 1 - \frac{6\sum_{i\in[1,N]} d_i(\langle F,\hat{F}\rangle)^2}{N(N^2-1)} \\ d_i(\langle F,\hat{F}\rangle) &= \rho(I_i, F) - \rho(I_i, \hat{F}) \end{aligned} \tag{5.31}$$

Where $d_i$ is the distance between the position of the i-th item in the two rankings. The coefficient is a value in $[-1, +1]$ where $+1$ means that the two rankings are exactly the same (maximum correlation), 0 no correlation and $-1$ the rankings are exactly one the opposite of the other. In order to map this in a value in the range $[0, +1]$ we have decided to compute a saturation to 0 of the negative values.

The $\sigma()$ function becomes formally:

$$\sigma(\langle F,\hat{F}\rangle) = \begin{cases} s(\langle F,\hat{F}\rangle) & s(\langle F,\hat{F}\rangle) \geq 0 \\ 0 & elsewhere \end{cases} \tag{5.32}$$

### Convergence

While for the binary version of the algorithm it can be proven that converges in a finite number of steps as in [111] for the others we have not analyzed the convergence property of the algorithm in detail. We can state that in all our tests the algorithm converges after a small number of iterations. The number of required iterations grows when the number of bad users grows or in case they follow a common pattern.

## 5.8 Techniques for Evaluation & Comparison

One of the major differences between digital games and traditional software is related to the fact that the former cannot be evaluated just by means of the utility of the expected results. In a game, the target objective would be to maximize the "fun"of the player.

According to Johan Huizinga, fun is "an absolutely primary category of life, familiar to everybody at a glance right down to the animal level"[116] Toys, games and activities perceived as fun are often challenging in some way and when a person is challenged to think consciously, overcome challenge and learn something new, they are more likely to enjoy a new experience and view it as fun; at the core of this perception is a change from routine activities, that are often repetitive and requiring limited conscious thinking.Nonetheless, given the subjective nature of the activity, the word "fun"has a distinctive elusiveness that is difficult to be represented in a quantitative way.

Games with a Purpose, given their very own nature as a sub-genre of Serious Games, and thus having clear objectives that can be measured, may be subject to quantitative approaches able to state if the GWAP has been successful or not. The higher the number of players, the higher the number of contributions, the better a GWAP is successful with respect to its target goal. Nonetheless, such a result could be derived from factors other than the quality of the game itself, for instance due to marketing campaigns.

How is it thus possible to compare two different GWAP from a pure qualitative point of view? How is it possible to relate the design of the game, the aspect more tied to the "fun"term described before, to the effectiveness of it? This section investigate both of the aspects by providing:

- Quantitative methods that can be used to compare the results of two different GWAP
- Qualitative methods that can be used to compare the entertaining capabilities of digital games.

### 5.8.1 Quantitative Evaluation

In the case in which we could consider games as if they were algorithms, efficiency would be the most immediate metric of evaluation. As there are different possible algorithms for a given problem, as there could be different possible GWAP suitable to solve it. But if for traditional algorithms are classified in terms of "Big-O"notation for what concerns efficiency, taking into account computational steps, a "Human Computation Step"in a GWAP is less clear, thus it is necessary to define efficiency in a different way. The efficiency of a GWAP can be related to the number of tasks solved by a user in a specific timeframe.

In [66], the *throughput* of a GWAP is defined as the average number of problem instances solved, or input-output mappings performed, per human-hour. This is calculated by examining how many individual inputs, or images, are matched with outputs, or labels, over a certain period of time. It is worth noting that the number of task instances solved should be unrelated to the number of actual players available in the platform, since that derives from factors other than the problem to be solved and its integration in the game-play, but it is related to the qualitative perception that a player has on the game. More meaningful would be a metric able to define the throughput of a single player in a particular GWAP; we call this measure the Individual Player Throughput (IPT). Since all games require at least a training phase and players' skills may vary based on how much the game has been played, we first compute the throughput for every single user by dividing the number of tasks performed in a gaming session by the length of the session normalized over a humanhour period. The average for all the throughputs computed for a single player is then calculated and finally averaged among all the players of the game to obtain the IPT.

This measure is a meaningful one to compare the efficiency of a GWAP to solve a particular task and can be used to compare GWAP that are trying to solve similar tasks, but does not take into account the ability of a GWAP to retain its users. A GWAP with a high IPT score that has few or no players at all is not able to solve the problems for which it has been designed. The real measure of the utility of a GWAP is thus a combination of IPT and time spent playing online; in [66] this measure is called "Average Lifetime Play"(ALP) and it is used as a proxy to measure how enjoyable a game is. ALP is the overall amount of time the game is played by each player averaged across all people who have played it.

Given the average number of contributions per hour, per player (IPT) and the expected amount of time that a player, on average, spend on the game, normalized over the hours (ALP), it is possible to assess each player's expected contribution (EXP), that is:

$EXP = IPT * ALP$

This measure is not able to capture phenomena such as popularity gains through spread of mouth, but metrics that are usually applied in Gamified applications, such as Daily Average Users (DAU), Churn and Retention rate could be also considered as meaningful ways to analyze in a quantitative way the results of a GWAP and objectively compare it against others even taking this aspect into consideration. They are described later in Section 6.4.

The EXP is a meaningful measure that allows the comparison among different GWAP that could be, otherwise, impossible to relate in a quantitative way, but it has to be noted that throughput and amount of time spent are meaningless if the results provided by the player via the GWAP are unusable. The EXP has thus to be paired with an accuracy measure able to assess if the aggregated results of a GWAP are meaningful solutions for the problem at hand. This can be achieved in several ways: running the game over a previously known and annotated dataset, comparing the output produced in the game to outputs generated by paid participants (rather than game players) or having the results of the game rated by independent subject evaluating their quality.

### 5.8.2 Qualitative Evaluation

Qualitative Evaluation of a videogame is a broad and multi-faceted topic that refers to a collection of methods, skills and tools used to uncover how a player perceives a videogame before, while and after having interacted with it. It is a non-trivial process due to the fact that "fun"and enjoyability, as defined in the previous section, are subjective, context-dependent and dynamic over time and in order to make the evaluation successful, it is necessary to select the right methods, dimensions and target playerbase on which to test. To handle the qualitative evaluation of GWAP, we have not developed techniques and methods ourselves but we have relied on the material and research available in literature. Since a good GWAP is able to hide a computational task in its gameplay in a seamless way, from a player's perspective it should be just another videogame. Focus of this section is thus to provide an overview of the methods and criteria for qualitative evaluation that have been used in the industry or in academia.

Many researchers have presented different techniques to evaluate videogames. In [117] Malone presents a set of heuristics to develop enjoyable interfaces for games, categorized into challenge, fantasy and curiosity. In a similar way, Garzotto in [118] also propose heuristics for educational games evaluation that included contents (length, integration), fun (attention, goal clarity, challenge, immersion) and social interaction (group cooperation, competition) as main facets.

Heuristic Evaluation for Playability (HEP) [119] are a set of methods that can be applied to analyze the dramatic and formal elements of games. They are divided into four categories: *gameplay*, the set of problems and challenges a user must face to win the game; *game story*, that includes all plot and character development; *game mechanics*, that involve the programming that provides the structure by which units interact with the environment; and *game usability* that addresses the interface and encompasses the elements the user utilizes to interact with the game (e.g. mouse, keyboard, controllers). These heuristics have been further expanded in [120].

In [121] the work of Malliet et al classifies the existing playability methods, along with ones that focus on strictly formal aspects of game content, methods related to user experience and methods that evaluate the interaction between content and players by means of biometric or psychophysiological measures. On the same line, Koeffel et al in [121] tried to create a comprehensive list of heuristics from those identified in literature and compared their effectiveness against video game reviews (even though reviews are highly subjective for their very nature), while Livingston [122] uses particular heuristics to retrieve meaningful considerations out of several game reviews, as a starting point to weight the severity of issues that should have been identified during an usability evaluation.

Biometric and pshychophysiological methods have been used in [123] to relate the design and the content of a game with the player experience. The collected data has proven useful in evaluating aspects such as level design and the relationship between the input system and the flow. In this work a practice-oriented model focused on describing playtesting in game development is also provided.

Finally the Game Experience Questionnaire (GEQ) and its Kids adaptation (KidsGEQ), described in [124] and [125] are self report instruments used to assess in-game experiences in young children (8-12 years old). They are tools that can provide a measure that is able to capture the full spectrum of digital game experiences, robust, agnostic to the type of game, platform or gamer and non-disruptive to the gameplay itself.

# Chapter 6

# Gamified Applications Design

## 6.1 Gamifying over Game Development

Even though Gamification is often misleadingly confused with Game Design, the two have different and deep differences:

When creating a game, it is common to to start with a basic idea. It may just be a theme to explore, it could be an interesting mechanic to flesh out into a full game or it could be the whole game concept from the beginning to the end. However the idea starts life, the rationale beyond its development is the perspective of making it enjoyable and fun for other players. You then start to put the idea together into something coherent. You prototype the basic mechanics and game-play elements. Next you experiment with how they fit together, why dynamics appear out of what combinations. You work out the themes and the story. Basically you put the meat on the bones of game, then the polish. Along the way, depending on how you want to manage the game, you will consider collecting metrics from the game. This may be part of a continual improvement plan, it may be part of a monetisation plan. Eventually after play testing and multiple iterations you have a final game ready for the mass population to play. You measure the success by how much people enjoy the game. Depending on the scale of the game, you will also have to measure sales.

When creating a gamified system, you start with an objective. This may be employee engagement, it may be increasing sales of a product. However, the goal is to meet that objective.

Next, depending on how you feel you can best meet that objective, you start to design your system. First and foremost in many systems will be the metrics you need to collect. The metrics are what will allow you to know if you are on target to meet the objective or not.
You consider what gamification elements and mechanics will best help you achieve the goal and start to put them into your system.
You will probably take into account how different user types react to different elements and experiment with them on test groups of users. Using the metrics you are collecting you will balance the system to drive the best and most efficient results you can. After multiple iterations you release the product. You measure success by how many people reach your original objective. At least this is how all game designers think we do things sadly in many cases, they are right.

It looks from this like there is no middle ground at all. Game design starts from the desire to make something that people will enjoy. In Gamification design, you are making something that will achieve a particular goal.
In game design, metrics are not always a main focus of a game at least at the initial conception. In gamification design, metrics are what your system will live and die for.
In game design you use mechanics, themes and more to help to make the game more enjoyable. In gamification design you add things that will help drive the user towards your business objective.
Game design does have an objective the objective is to create a game that is enjoyable, even if it is only you who finds it so. So everything you do is driven by this goal. You add and remove ideas as you find they work or dont work.
Gamification design is no different. The goal may not be fun, but it is to make something less difficult or tedious to do. Gamification is often about lowering a barrier to achievement in some way.
How this goal is achieved is a discussion for the following chapters, in which the basic mechanics for a gamification system are described

## 6.2 Gamification Development Process

The gamification process addresses the development or the revamping of an application offered to the general public, with the purpose of improving participation through game design techniques. Business and functional requirements are pre-existing to the gamification effort, because the business goals are supposed to be already defined in the design of the original application.

Figure 6.1 shows the representation of the development process of a gamified application as derived from the best practices defined in [126][127] and integrated with the activities usually followed in the development of traditional web applications.

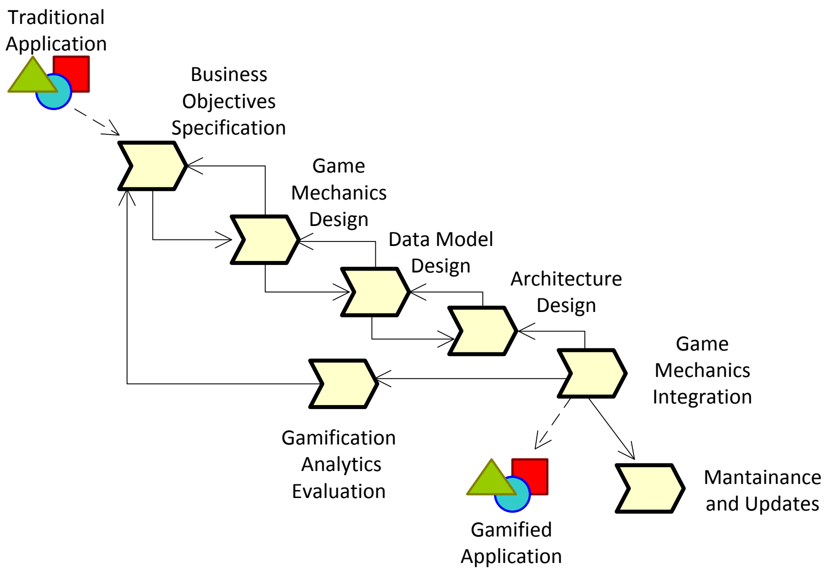



Figure 6.1: Process model for gamified application development

- **Gamification Requirements Specification** complements and integrates the requirements specification activity, by defining the drivers for the introduction of gaming elements to improve application usage. The most important part of the analysis is thus to *identify the business objectives* that have to be met by describing the processes supported by the application and why they are not reaching the expected results. An example could be the need of easing the learning curve of a complex authoring tool, such as a CAD, for which the majority of the users are employing just the basic functionalities and for which the use of traditional tutorials have failed.

  Once the business objectives have been stated, there is the need to *identify Target Players*, selecting a subset of the possible users of the application not only among the "external" ones, such as the customers, but also among the "internal" users, such as the employees or the developers.

In the CAD example, the target players are just the traditional users of the software, both new and experienced ones.
The next step is to *delineate target behaviors*, what the users of the system are expected to do and why these actions could improve the capability of the system in achieving the expected results. An example of target behaviors expected for the CAD scenario could be testing all the tools that the software has to offer in a guided example.
Lastly the nonfunctional requirements for the application are defined; nonfunctional requirements are fundamental for accomplishing the business goals but not specifically related to the functionalities of the application. Some of the most important nonfunctional requirements for a gamified application, given its peculiar dynamic and asynchronous nature, include: Response Time, Localization, Usability, Virality.

- **Game Mechanics Design** Once the requirements have been specified, the following step involves the enhancement of an existing application with the definition of game mechanics and game elements to be able to increase the participation of the users; they are detailed in Section 6.3. Not all the business objectives are suitable for a gamification approach. Ideal candidates for gamification are processes that depend on motivation, offer interesting challenges that are easily coded into rules, and reinforce existing reward systems. Once suitable business objectives have been identified, the cause that lowers the motivation of the users must be identified and paired with mechanics able to address these issues. Self Determination Theory [128] has been used in gamification to describe three needs of human beings that can influence motivation: *Competence* - being effective in dealing with the external environment and solving problems -, *Relatedness* - the desire of being involved in social interactions - and *Autonomy* -the innate need to feel in command of ones life-
This phase involves also the *Aesthetics design* and *Interface design* that has been described in 5.1. It is necessary to state that the design of game mechanics is still a rather creative process that relies on the experience of the designer; the game mechanics presented in Section 6.3 defines just the basic structure of the actions that can be performed to improve the engagement of an application, their real usage and implementations are

dependent on the target audience and the specific application that has to be defined. In the CAD example, the issue is related to the lack of Competence of the players. Defining challenges involving exploration and modification of a 3D environment with the use of the authoring tool has been successfully applied in [129]. To summarize, the game mechanics design phase involves: identifying the source of demotivation, pair suitable game mechanics, design and tailor the mechanics for a specific application and user base.

- **Data Model Design** The data model design, as it happens in traditional applications, focuses on the definition of sub-schemas describing the core elements of the game mechanics, their interconnection and their relationship to existing data sources that could be of interest.
- **Architecture Design** It involves the definition of the hardware and software components that has to be introduced to implement the new functionalities. It may involve the use of external gamification platforms or ad-hoc solutions.
- **Game Mechanics Integration** It involves the mapping of the new data sources and architectural components to the existing platform to introduce the new gamification functionalities. This mapping phase is well-established discipline for which a number of effective methods, tools and technologies can be borrowed from literature on software engineering and known as System Integration. Output of this phase is an iteration of the final *gamified application* that will be refined in subsequent iterations if the business goals are still unreached.
- **Analytics Evaluation** In this phase, the performance of the application with respect to the desired results is analyzed by using traditional web analytics related to page statistics, session time, number of views, coupled with social metrics such as Daily Active Users, K-Factor, E-Score defined in [130][127] and thus obtaining a quantifiable comparison of the application of gamification techniques over the traditional application.

## 6.3 Gamification Mechanics

### 6.3.1 Points

*Points* or *Player scores* are numerical values that represent a measure of the skill of a player. When a player obtains a badge, he is often rewarded with an amount of points that depend on the difficulty of the task performed; the player score is the sum of all the points a player has received in his gaming history. We often see points used to encourage people to do things by collecting them. The assumption is that people will buy more widgets or will work harder in exchange for points. This is a simple approach that occasionally works to motivate those people who like collecting things ("Look at how many points I just received!") or for those who like competing against others ("No one else has 1,000,000 points!") But points can be used in many other ways, and we need to understand how the humble point can serve many functions. Points are used in gamification for a number of reasons:

1. **Keeping score** This is the typical way they are used in gamification systems. Points tell the player how well he or she is doing. Someone who has earned 32,768 points has been playing longer or more successfully than someone with 24,813 points. Points can also demarcate levels. For example, "You need 10,000 points to reach Level 5, at which point you unlock the 'super player'achievement and get access to new content."In this case points represent the true "play space"of a game because they define progress from the beginning of the game to its objectives.

2. **Determine the winning state** when it is necessary to define a winning condition for the players, points can be an easy mean to achieve the result.

3. **Connecting progression and rewards** Many gamified systems offer some real-world prizes for reaching certain levels or for redeeming virtual points: 1,000 points gets you a set of steak knives and 1,000,000 points gets you a round-trip ticket to Tahiti. Club Psych takes this approach, but its common in all manner of marketing and promotional devices that have been used for years.

4. **Provide clear feedback** Explicit and frequent feedback is a key element in most good game design, and points provide feedback quickly and easily. Points are among the most granular of feedback mechanisms. Each point gives the user a tiny bit of feedback, saying that he or she is doing well and progressing in the game.

5. **Provide quantifiable data** The points that users earn can easily be tracked and stored. This allows the designer to analyze important metrics about the system. For example, how fast are users progressing through the content? Do they seem to be falling off or stalling out at certain junctures?

By understanding the nature of points, you can use them in ways that meet the objectives of your gamified system. Do you want to encourage competition? Then use points as scores. Do you want your users to be hooked by the dopamine drip of constant feedback? Then use points to give them a sense of mastery and progression, without showing them how others are doing. And so on. Bear in mind that points are very limited. They are uniform, abstract, interchangeable, and well, pointlike. To put it another way, a point is a point. Each additional point simply indicates a greater magnitude, and nothing more. This is one reason why badges are often found in conjunction with points systems.

### 6.3.2 Achievement and Badges

An *Achievement* is a set of tasks, defined by a designer, for the player to fulfill so to achieve a milestone and progress in the game. A *Badge* is an artifact associated to the completion of an achievement and given to a player after completing the achievement, or, in gaming terms, after "unlocking the achievement".

In literature, the distinction between achievements and badges is blurred. For this reason, we will use just the term "Badge"from now on.

Badges are a chunkier version of points. A badge is a visual representation of some achievement within the gamified process. Some badges simply demarcate a certain level of points. Fitbit is a gamified system that allows people to use a wireless pedometer to track the number of miles they walk or run. The system displays a badge when the user exceeds certain point thresholds, such as 50 miles in a week or 10,000 steps in a day.

Other badges signify different kinds of activities. Foursquare, a service that engages users with local businesses by encouraging them to check in to a location with their cellphone, has numerous badges for all manner of achievements. Users unlock the Adventurer badge as soon as they check into ten places registered with the Foursquare system, and they receive the Crunked badge for checking into four bars in one night. (No one said that badges need to be socially responsible.)

Achievements are now so popular in the gaming culture that the reasons for which they have been introduced are often overlooked; however to make a reward system effective, it is necessary to keep in mind the purpose for which they have been developed.

As stated by Björk and Holopainen [131], "Games do not work without incentives for the players to perform actions and to strive towards their goals", while Juul [132] claims that "Players play for personal goals, are aware of the goals of other players, and the shared understanding of intentionality makes game actions socially meaningful".

Achievements range from simple actions that the player would do anyway, as common gameplay actions, to more difficult challenges even against other players, to a recognition for sharing contents among a community.
They are defined by a name, a visual representation in the form of an icon and a description of the tasks to be performed. The player can analyze the description of an achievement even before starting the gaming experience, thus knowing in advance what to expect from it and gather knowledge about secret features hidden in the story or in the gameplay.

Every achievement must have one or more completion criteria, that can be defined through event-condition-action (ECA) rules [133]. The event may be a player action, a system event, the occurrence of a specific condition of the gamestate or a combination of the three that may trigger the achievement completion.

The condition is the set of pre-requirements on the present state or on past actions that must hold in order for the completion of the achievement to be attributable.

The action is the unlocking of the achievement, which entails the generation of a badge for the user that has completed the achievement and the assignment of digital or real world rewards.The acquired badges implicitly store valuable user information that can be used for profiling purposes, such as his favorite games and genres, his mastered skills and past gaming history.

The objective of an achievement is thus to set players expectations, lead them to fun parts of the game they may not otherwise discover on their own, and instruct them about the possible actions that can be performed within a system. These features alone could be obtained just with the smart use of game mechanics, but the benefits go much further; achievements allow others to recognize what the player has attained and enhance games by providing lasting rewards. This leads to a sense of affirmation given by the fictional status that the player has created for himself and the expectation that others will look with admiration someone who has undertaken the action stated in the achievement.

It is important to remark that an achievement system is not just a set of rewards given to the players for a specific game. Scores associated to a badge are an important aspect that has marked the Xbox Live! Achievement System as a novelty.

The gamer score is a synthetic means for quantifying a player's skill. While the obtained badges can represent the specific game mechanics that a player has been able to master, the numerical score is an immediate indication of the gamer's experience, the one most recognizable by the other players and the one that turned the Xbox Live! Achievement System into a multiplayer game based on the gamer score leaderboard.

The last component needed to a fully operational achievement system is a statistical information dossier about the player. In recent years, games are rarely played in isolation, as players often discuss online their mastery of the game, including any goals they have completed. A detailed dossier of the gaming history of a player, including the game he has played, the badges he has obtained, the level he has achieved and his score along with a friendlist and the social gaming groups he is subscribed to is therefore crucial, because it is the feature that enables the social part of the game rewards.

The design of achievements in a game is an aspect often overlooked by game designers but is one of the key factor that is needed to motivate a player.

In the following, a taxonomy of rewards will be provided along with meaningul guidelines to be followed when designing achievements able to provide satisfying and not alienating experiences.

### Achievements Taxonomy

In order to develop a meaningful categorization, several existing platforms and gaming communities such as Xbox Live!, Playstation Network, Steam, Kongregate, the Facebook's Achievement System have been taken in consideration with respect to their achievement system features.

In particular, the analysis has been carried on six popular games representative of each platform, listed in Table 6.1 along with an example for each achievement category; even considering the heterogeneity of platforms and genres, the patterns found across the analyzed titles were almost identical.

In addition to this research, the opinions and preferences of players has been gathered from several online website centered on the topic, such as Xbox360achievements [134] and PS3Trohpies[135].

The research has been backed up also with comparisons with the existing literature in game design, in particular Salen & Zimmerman [136], Fullerton and Hoffman[137], Jesper Juul [138] to assure coherence with existing terminology and to group achievements following established game mechanics paradigms.

The rest of this Section reports the proposed taxonomy of achievements resulting from he described research.

**Instructors**
: Instructors are used to show to the players the core mechanics of the game and help them to improve their skills.

**Quests**
: Quests are awarded upon the completion of a “level” or any other significant task.

**Modes Exploration**
: Modes exploration achievements are awarded as incentives to try all that the game has to offer. They usually require one to play in a specific game mode, try specific gaming features or components of the game, including the use of in-game menus.

**Socializers**

Socializers are awarded to reward social behavior. Examples include achievements for making custom content, for reaching the maximum number of players in a game or on a server, for giving items to another player, for assisting another player through a level, or otherwise interacting with the other players to enhance the gameplay experience.

**Secret Chests**

Secret Chests achievements are awarded for finding hidden items, special areas, completing collections, and so on. They encourage the player to explore every facet of the game, and as such will enhance value (and replay value).

**Grinders**

Grinders are a type of achievement in which the task is to perform the same action repeatedly, with little or no variance between each repetition such as reaching “1000 kills”, earning “100,000 gold” etc...

**Herculean Tasks**

Herculean Tasks are rewarded when a player is able to perform exceptional actions within the game. They are usually difficult, non repetitive tasks that only few committed players will be able to reach.

**Trophies**

Trophies are achievements that, by their very nature, can be acquired by only a few top players in the world. Being the top player on a permanent leaderboard is one example; winning an online tournament is another.

**Red Marks**

Red Marks are “awarded” for negative actions in the game, such as losing or being humiliated.

**Loyalties**

Loyalties are achievements given to players in order to reward customer loyalty. For instance, they can be given if a player partecipates to a real world event like a convention or if the player buys special editions of the game.

| Game Title | Ghost Recon: Advanced Warfighter | Heavy Rain | Team Fortress 2 | World Of Warcraft | Tyrant | The Sims Social |
|---|---|---|---|---|---|---|
| Platform | Xbox 360 | Playstation 3 | Steam (PC) | PC | Kongregate | Facebook |
| Instructors | Completed tutorial level on any skill level | Prologue - Complete the drawing + Set the table + Play with kids | Use Jarate to reveal a cloaked Spy | Learn how to transform into a dragon and carry an ally. | N/A | Plant something in two empty garden patches |
| Quests | Secure the US president | Paparazzi - Leave home without being spotted by the journalists | Win 2Fort with a shutout | Get caught in 10 consecutive land mine explosions without landing. | Location 2 reached | Complete the quest: Roominating |
| Modes Exploration | Win 5 matches on each original Multiplayer Map | See all endings | While watching a replay, press space bar to enter the editor | Explore Duskwood, revealing the covered areas of the world map. | N/A | Break The Ice - Automatically Earned |
| Socializers | Win 30 co-op matches with at least 6 gamertags in the room | N/A | Achieve 100,000 YouTube views for your movie | N/A | N/A | Post to news feed and have 3 friends click your message |
| Secret Chests | N/A | Crime Scene - Find all clues related to the Origami Killer in the scene | Get to Loot Island and claim your reward! | N/A | 52 unique cards collected | N/A |
| Grinders | Get a total of 10,000 kills in multiplayer | Lexington Station - Knock down at least 50 passers-by | Do 1 million points of total fire damage | Find 100 common artifacts. | N/A | Earn 600 simoleons from cooking |
| Herculean Task | Get 4 kills in 4 seconds or less in multiplayer | Perfect Crime | Get a melee kill while sticky jumping | Fish up Old Crafty in Orgrimmar. | Nexus campaign completed | N/A |
| Trophies | Climb to the top of the universal leaderboard | N/A | N/A | First person on the realm to achieve level 80. | N/A | N/A |
| Red Marks | N/A | The Butterfly - Give up or fail the Butterfly Trial | N/A | N/A | N/A | N/A |
| Loyalties | N/A | N/A | N/A | Owner of the Wrath of the Lich King's Collector's Edition Frost Wyrm Whelp pet. | N/A | N/A |

Table 6.1: Table of the analyzed games and sample achievements

### Achievements Design Guidelines

In the following we present some guidelines on achievements design based on the best practices provided by Greg McClanahan, achievement designer at Kongregate, in [139] and tailored on the taxonomy of Section 6.3.2 along with the comments and preferences of users stated in the aforementioned forums; they can be used to avoid designing achievements that are not appealing for the players or that are not meaningful for the gaming experience.

**Achievement Evaluation**
: When designing an achievement, it is fundamental to keep in mind that if a player is motivated by the need of completing all the achievements that the game has to offer, he will always try to use the most efficient method to earn it; taking into consideration this aspect, an achievement has to be designed by evaluating this strategy, balancing difficulty and means by which the achievement is earned to avoid creating a repetitive and alienating task.

**Highest Difficulty**
: If several achievements have been designed to end the game at different difficulty levels, it is good practice to acknowledge the highest difficulty at which something has been accomplished plus all the implied easier achievements. If someone completes the game in "hard mode", he should not be forced to do the same also in easier settings.

**Always Earnable**
: Achievements should always be earnable without compromising the game progression; a player should not be forced to restart the entire game from the beginning just because he has missed an achievement that can be no longer obtained during the current game situation.

**Not so secret**
: Achievements hints must be findable; players must be able to get know that the game contains secret features or side quests and be accompanied in their exploration of the gameplay to get to them.

**Look what happened**
: Strange or unlikely situations that can happen during the gameplay

should be rewarded with a special badge in order for the player to remember the moment. It should be always kept in mind that these tasks have not to be random or too difficult to be obtained, otherwise committed players will just spend hours trying to achieve that particular result.

**Elitism is bad**

It is better to avoid designing achievements that reward getting the highest ranking among a group of people, a region or worldwide, reaching a top spot on some leaderboard or winning a tournament. Only few selected and possibly excessively committed players will be able to obtain them, while the others will feel like they will not be able to fully complete the game with the risk of abandoning it.

**Nothing Hidden**

It is better to avoid designing "hidden" achievements that do not state in a clear way which tasks the player has to perform. Achievements are well stated goals that the player has to reach, if the tasks are not given the player will just obtain an achievement by chance.

**The Noob**

"Rewarding" players with badges because they have suffered negative actions, such as losing a certain number of matches or suffering a brutal death is bad design that has to be avoided. Players are not satisfied when losing or being considered low skilled and remarking this fact in their gaming history may have a negative effect.

**Grinders Tips**

Grinders are used to encourage deep exploration into modes or certain game mechanics, but even if they are apparently the easiest ones to introduce in a game, an improper design of them can disrupt the game experience. Grinders should fit with an action that the player would naturally do anyway. If completing a specific task in the game would be considered as a normal behavior even without an achievement attached to it, then an achievement can be designed around it. Otherwise the achievement forces a player to change his behavior, which goes against the principle that an achievement should contribute to a better experience for the player.

**Not that easy, not that hard**

The balance of difficulty across the achievements should be planned with care. Too many trivial achievements can induce the players to become bored because their skills are not challenged. Too many hard achievements can, on the contrary, induce anxiety and frustration during the whole gameplay experience. As a rule of thumb, the design should consider the percentage of achievements that an average player is expected to be able to earn globally with the game. A range between 70 and 90 percent is a reasonable option, leaving a small percentage of badges for skilled players, and even fewer (1-2 at most) for highly skilled players.

### 6.3.3 Leaderboards

A *Leaderboard* is an ordered list of players, with the scores they have obtained in a specific game, It can be considered as the early ancestor of the achievement concept. Leaderboards are problematic gamification elements: On one hand, players often want to know where they stand relative to their peers.

A leaderboard gives context to progression in a way the points or badges cant. If performance in the game matters, the leaderboard makes that performance public for all to see. In the right situation, leaderboards can be powerful motivators. Knowing that its just a few more points to move up a slot or even to emerge on top can be a strong push for users. On the other hand, leaderboards can be powerfully demotivating. If you see exactly how far you are behind the top players, it can cause you to check out and stop trying.

Leaderboards can also reduce the richness of a game to a zero-sum struggle for supremacy, which inherently turns off some people and makes others behave in less desirable ways. Several studies have shown that introducing a leaderboard alone in a business environment will usually reduce performance rather than enhance it. There are various ways to make leaderboards work for your gamified system. A leaderboard need not be a static scoreboard, and it need not only track one attribute.

In gamification, leaderboards can track any feature or features the designer wants to emphasize. Theres nothing wrong with multiple leaderboards measuring different things or leaderboards that arent universal for all participants.

Leaderboards can also be tied to social networks to provide more contextual, and less troubling, information about how players are faring.

## 6.4 Techniques for Evaluation and Comparison

After the definition of the game mechanics that can be introduced in a gamified application, another fundamental aspect is the definition of what are the metrics by means of which we can say if the Gamification approach contributed positively or not.
While the term game metric has become something of a buzzword in game development in recent years, metrics have arguably been around for as long as digital games have been made even though he application of game telemetry and game metrics to drive data-driven design and development has expanded and matured rapidly just in the past few years across the industry[140]. Game metrics start with raw telemetry data, which can be stored in various database formats, ordered in such a way that it is possible to transform the data into various interpretable measures, e.g. average completion time as a function of individual game levels or revenue per day.
The game metrics used more often are:

- ARPU
- Churn
- Retention
- DAU
- MAU
- DAU/MAU
- Cohort
- Engagement
- Re-Engagement

- Entry Event
- Exit Event
- Viral rate/K Factor
- Lifetime Network Value

In the following, each of these metrics will be explained, defining the advantages of using it and how to calculate it.

**ARPU**

Average revenue per user (sometimes average revenue per unit) usually abbreviated to ARPU is a measure used primarily by consumer communications and networking companies, defined as the total revenue divided by the number of subscribers[141].

Method of Calculation: To calculate the ARPU, a standard time period must be defined. Most telecommunications carriers operate using the month as a measure. The total revenue generated by all units (paying subscribers or communications devices) during that period is determined. Then that figure is divided by the number of units. Because the number of units can vary from day to day, the average number of units must be calculated or estimated for a given month to obtain the most accurate possible ARPU figure for that month.

The ARPU can also be calculated according to diverse factors such as geographic location, user age, user occupation, user income and the total time per month each user spends on the system.

Also related to this measure is the ARPPU (Average Revenue Per Paying User) which is calculated by dividing up the revenue amongst the users who paid anything at all. This yields a figure that is significantly larger than ARPU. For example in the case of a subscription game (that has a free play version), the ARPPU, measured by accounts, is the subscription price, diluted slightly by free trials[141].

**Churn**

The turnover rate (or attrition rate) of social games active players. The noise level in casual gaming is extremely high, which means social games have a user base that is constantly changing as gamers abandon the game. Churn refers to this constant loss and gain of members [142].

In its broadest sense, is a measure of the number of individuals or items moving out of a collectivity over a specific period of time. It is one of two primary factors that determine the steady-state level of customers a business will support. The term is used in many contexts, but is most widely applied in business with respect to a contractual customer base. For instance, it is an important factor for any business with a subscriber-based service model, including mobile telephone networks and pay TV operators.The term is also used to refer to participants turnover in peer-to-peer networks. Churn rate is an important input into customer lifetime value modeling, and can be part of a simulator used to measure Return on Marketing Investment using Marketing Mix Modeling.

Churn rate, when applied to a customer base, refers to the proportion of contractual customers or subscribers who leave a supplier during a given time period. It is a possible indicator of customer dissatisfaction, cheaper or better offers from the competition, more successful sales or marketing by the competition, or reasons related with the customer life cycle[143]. For customers the formula can be simply:

$$\left(\frac{\text{subscribers lost}}{\text{starting subscribers}}\right) \times 100$$

Others choose to base their churn rate off the number of subscribers at the end of the period instead of the beginning of the period.

$$\left(\frac{\text{subscribers lost}}{\text{ending subscribers}}\right) \times 100$$

In some business contexts, churn rate could also refer to employee turnover within a company. The company size and industry also play a key role in attrition rate. An acceptable attrition rate for a given company is relative to its industry. Regardless of industry or company size, attrition rate tends to be highest among the lowest paying jobs, and lowest for the highest paying jobs[143].

Churn rate can also describe the number of employees that move within a

certain period. For example, the annual churn rate would be the total number of moves completed in a 12-month period divided by the average number of occupants during the same 12-month period.
For employee the formula can be:

$$\textit{Attrition rate (\%)}= \left(\frac{\text{No. of employees resigned during the month}}{\text{Average no. of employees during the month}}\right) \times 100$$

Where:

$$\textit{Average no. of employees during the month}= \frac{(\text{No. of employees at the start of the month + No. of employees at the end of the month})}{2}$$

**Retention**

It can be considered as the opposite of churn. Retention is how well you maintain your user base. The term "retention rate" is used in a variety of fields, not only in games or gamified application, including marketing, investing, education, in the workplace and in clinical trials. One of the most mathematically accurate formula described by [144] as:

$$\textit{Retention rate (\%)}= \left[\frac{(CE-CN)}{CS}\right] \times 100$$

Where:

- CE = number of customers at end of period
- CN = number of new customers acquired during period
- CS = number of customers at start of period

In a practical example, if we start the (week/month/year/other period you choose) with 200 customers, we lose 20 of them, but we gain 40 new, at the end of the period we have 220 customers. So, applying the formula:

$$\textit{Retention rate (\%)}= \left[\frac{(220-40)}{200}\right] \times 100 = \textit{90\%}$$

**DAU (Daily Active Users)**

Social gaming companies use this number to understand their active users in much more granular detail. It is the count of the number of active users on any given day. It is used to track the active nature of the gaming app.
1 million daily active users versus 1 million monthly or active users are very

different numbers and indicate very different success rates for the games. DAUs are used to calculate your revenue trajectory as a game company because it captures the smallest calibration of spending activity. For many social games, if a unique user does not spend money in their first day playing the game, they will likely never spend money[145].

**MAU (Monthly Active Users)**
Social gaming companies use this number to understand their active users in much more granular detail. It is the count of the number of unique active users in any given month. It is used to track the active nature of the gaming app. 1 million monthly active users versus 1 million active users can mean two different numbers[145].

**DAU/MAU**
The DAU/MAU ratio is one of the hot metrics in social games derived from the previous two[142]. Daily Active Users (DAU) to Monthly Active Users (MAU) ratio is a popular metric many consumer startups are being judged by. The ratio is used to find out how many of the active users are logging in on a daily basis. This metric is very important when determining how sticky the application is. In other words, it is a way to measure the applications retention. Giving an example, if we have 500,000 daily users and 1 million monthly users, the DAU/MAU is .5, translating to the average user logging in 15 days per month.
The DAU to MAU ratio is a very powerful metric that every consumer company needs to track. But what is important is to defining the active user, this is really the key to using it effectively. Logging in to the application is a great indicator of which users are engaged but it isnt the only measure. Stay flexible with the definition of active and experiment with the scope of your product to really make the DAU to MAU ratio work for your product.

**Cohort**
In statistics and demography, a cohort is a group of subjects who have shared a particular event together during a particular time span (e.g., people born in Europe between 1918 and 1939; survivors of an aircrash; truck drivers who smoked between age 30 and 40)[146].

In social gaming metrics, cohorts are used for analyzing retention.
By organizing users in groups such as "everyone that visited on June 10th" and analyzing the percentage that revisit, you can pinpoint what promotions are having the greatest effect.
For instance, we could group customer by how they were originally referred to their business and track how much money they spent over time.

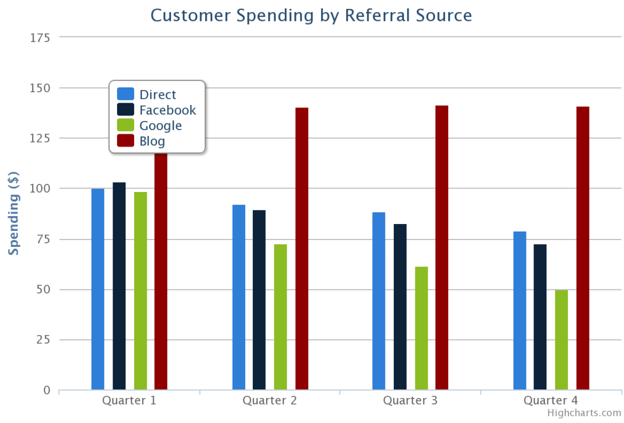



Figure 6.2: Example 1 of Cohort chart

In Figure 6.2 we see that customers referred by the blog deliver strong, consistent long-term spending. Search engines and other channels, however, refer customers who spend a decreasing amount over time.

Perhaps the most popular cohort analysis is one that groups customers based on their "join date," or the date when they made their first purchase. Studying the spending trends of cohorts from different periods in time can indicate if the quality of the average customer being acquired is increasing or decreasing in over time.

In Figure 6.3 the average customer in newer cohorts is spending less as time goes on. This would be a red flag for many investors or acquirers because it implies that the value of recently-acquired customers is less than those acquired in the past.

### Entry Event

An entry event is the first action a user performs when they enter the game[142]. Online social games can track every action you perform, and the *Entry Event Distribution* is one of the more important metrics to follow. What do your users do first? Which entry events are the most effective at

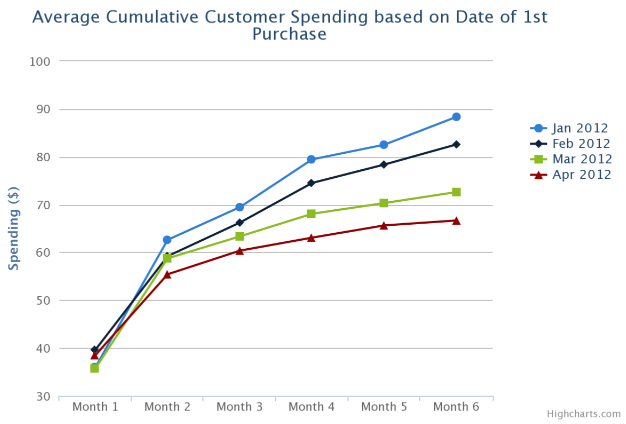



Figure 6.3: Example 2 of Cohort chart

bringing people back?

By determining the more popular entry events, we can push more resources towards them, thus increasing retention,engagement and re-engagement.

### Exit Event

The opposite of entry events. Exit events are the last actions a user performs before exiting the game[142]. Tracking the *Exit Event Distribution* helps show why users are disengaging with the game.

### Viral rate/K-Factor

Measured by K-Factor, the Viral Rate shows how much your users are promoting, evangelizing and spreading the application/game. Because of this, social games are increasingly built around cooperation, competition and the constant addition of new features, which increase virality. Every feature is a source for growth, whether its "liking", Facebook notifications or tweets for example.

The formula of K-Factor is:

$$(\text{Infection Rate}) * (\text{Conversion Rate})$$

Where:

- An Infection Rate is how much a given user exposes the game to other players, such as through status updates or email invites;

- A conversion rate, as marketers know, is when that infection results in a new sign up (or install).

In a rather simple consideration we can sey that a K-Factor of 1 means every member is bringing you one additional member. A high K-Factor is treasured by social game publishers, because it becomes a very effective vehicle for bringing in new players.

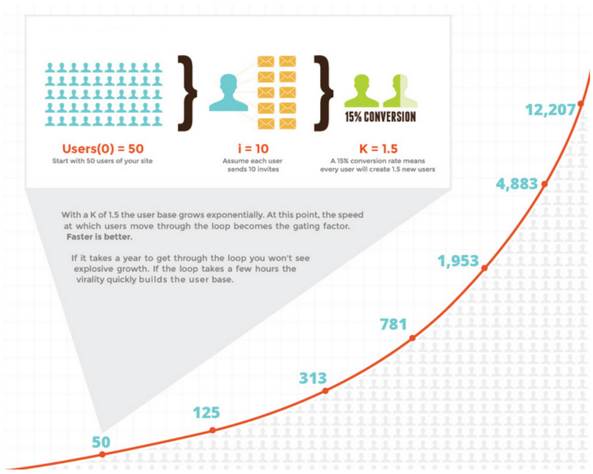



Figure 6.4: K-Factor to evaluate the Virality Rate

**Engagement**

The term engagement, in a business sense, indicates the connection between a consumer and a product or service[147]. There is no single metric on the Web or in mobile technology that breaks down or sufficiently measures engagement. Page views and unique viewers dont quite answer the question of who is engaging with our products, services, ideas, websites, and businesses as a whole. We would be better off thinking of engagement as being comprised of a series of potentially interrelated metrics that combine to form a whole.
[148]
These metrics are:

- Recency: How long ago did customers visit?
- Frequency: How often did customers come back?
- Duration: How long did customers stay?

- Virality: How many people have customers told about company? (Explained before)
- Rating: What did customers explicitly say when asked about company?

Collectively, they can be amalgamated as an E (or engagement) score. The relative proportion, or importance, of each of these metrics will vary depending on the type of business.
Substantially, engagement measures how long they spend playing your game. How many features do they access? Are they spending hours or seconds? How many pages does the average user view? What percentage are returning visitors?
An example, for Facebook the engagement metrics are translated how: *Interactions* (Like, Comments, Shares, Replies, Rewteets, and so on), *Reach* (the percentage of fans that have seen your post from your Page) and *Engagement Rates* calculated with the follow formula:

$$\frac{\text{(Likes + Comments + Shares) on a given day}}{\text{Total fans on a given day}} \times 100$$

Even if the previously formula is related to Facebook, it can be generalized finding the useful actions related to the company and dividing for the number of active users.

**Re-Engagement**

Gamers stop playing eventually. Re-engagement is how you get them back. It includes re-engaging gamers who have been signed off for an hour, a day, a month, or more[142]. Theres a lot of competition out there, so implementing and tracking re-engagement practices is a must.

**Lifetime Network Value**

Also called LTV, its the value a user provides to your network over the course of their entire lifetime on the network. For instance, is the user contributing to viral effects? Evangelizing the game? Contributing positively to ARPU? This is compared to the User Acquisition Cost, or how much it costs (via marketing and viral efforts) to bring in new members.
LTV metrics is composed by six key shown in Figure 6.5.

In the following, each of the LTV's keys are detailed:


Figure 6.5: LTV's keys

- Monetization: of course the most obvious component of LTV is direct monetization. "Premium"game purchases and pay-to-play transactions are the ideal type of monetization. Influencing a non-paying player to pull out their wallet increases their likelihood of spending in the future and becoming an invested, engaged player.
- Marketplace Exposure: rankings drive free exposure and organic installs, providing ancillary value for each new installation. Consistent exposure is key to driving long-term organic installs. Reviews, and more so ratings, influence users likelihood of tapping the install button.
- Virality: as said before, each player has the potential to drive new user adoption through face-to-face recommendations, online word-of-mouth, or more formal viral loops. These socialites often re-engage existing and churned players through these organic notifications.
- UGC and Community: new, fresh content is key to keeping an engaged user base; however, this is typically the most costly investment for game creators. Games that support UGC not only benefit from free content but also create a meta-game that extends the life of their game, particularly

for the most engaged, elder players. Community interaction in forums, leaderboard rankings, and multiplayer/co-op experiences also contribute to Value creation. Although a player may never provide LTV in other areas, their participation in multiplayer matches helps ensure players always have competition to quickly match up against and provides social proof that other are playing the game.

- Loyality: loyal players tend to associate themselves with brands they love, promoting them on their sleeve or through online channels. They also thirst for behind-the-scenes info and sneak-peeks of new content or titles, often signing up for newsletters and other communication channels for future re-engagement.

- Feedback: explicit feedback is provided through support emails, social network posts, forums, reviews, and in-game surveys. As creative and experienced as one may be, some of the best ideas come from players. Arguably more important is understanding what players actually do through implicit feedback. User behavior analytics (when are players churning? what content is selling?) and crash reports (which devices and OS are experiencing issues?) provide empirical data on areas of improvement. Each player can contribute but statistical significance is required before making any strong conclusions.

# Chapter 7

# Human Computation Architecture

One of the key challenges for the development of a socially enabled human computation platform, and thus also GWAP, or a gamified application is the design of a unified data model for representing the relevant aspects of users, their social ties and activities, the communities where they are active, the actions they can contribute, and the contents which are the object of such actions.

As it has been detailed in [149], currently there is no universal data model or standard able to embrace all the facets of the personal and social contribution of users. The envisioned model must convey in an integrated manner the *profile features* and *social links and roles* of users [150], the characteristics of the *content objects* they produce and consume [151] and the elementary *actions and tasks* they perform in virtual or real contexts of interest. Such actions and tasks are organized into *processes* to meet some global, community-wide goal, or special-purpose aspects, as required, e.g., when special tools like *gaming applications* are exploited to better engage users and foster their participation or exchange of opinions.

Such a data model should also be capable of (1) expressing the *uncertainty* of data, which is introduced by the automated collection procedures that are normally used to harvest user's features, and (2) managing arisen *conflicts* due to approximate feature extraction algorithms, contradictory data, or conflicting user's actions.

Fig. 7.1 depicts a bird's eye view on the main sub-models that compose our

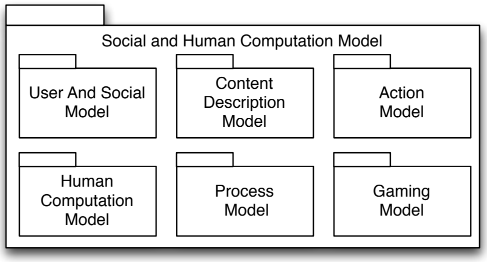



Figure 7.1: The Social and Human Computation Model

community management framework, and that will be described in details in the next sections. The *Process* model focuses on the global aspects of coordinating a set of human and automatic actions to achieve a specific purpose; as it does not differ from the workflow models of popular business processes and service orchestration languages (e.g. BPEL or BPMN[152]), its description is therefore omitted.

## 7.1 The User & Social Model

The *User and Social* model is devoted to the representation of humans in the context of human computation, by expressing the roles they can play as social actors, content producers and consumers. Furthermore, it describes the embedding of users in social networks by modeling their relations to communities and the most common properties that characterize a social activity profile.

### 7.1.1 Modeling Users

Fig. 7.2 depicts the user taxonomy at the core of the *User and Social Model*. The main concept is the *User*, which specializes into *Administrator*, *Content-Provider*, and *End-User*. *Administrators* and *ContentProviders* denote roles that serve an internal purpose in the specific human computation platform: the former controls the system, whereas the latter provides *ContentObjects*. These internal roles can be extended, to cater for a taxonomy of internal roles depending on the application domain. The *End-User* entity represents social users that interact with the platform consuming or producing resources.

When registered in a human computation platform, end-users are further distinguished into:

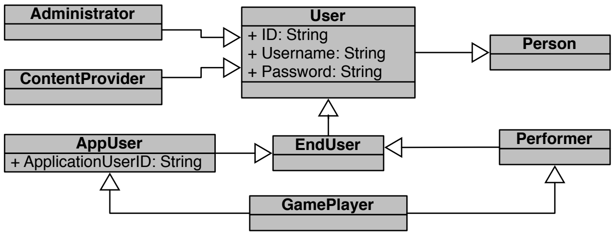



Figure 7.2: User Taxonomy Model

- *AppUsers*, who interact through an *Application*, e.g. the *Fashion Trend Application* of the running example; they can be characterized by application-specific properties (e.g., an ApplicationUserID and application-dependent profile data). An *End-User* not registered to the human computation platform can also provide useful information, e.g., by implicitly boosting the relevance of a given content object through share or re-post actions.

- *Performers*, end users that are registered explicitly for contributing to managed tasks; they have work-dependent attributes, e.g., their work history and quality data (e.g., error rates and other performance indicators).

Games are treated as a special class of applications for implicitly solving a human computation task. Therefore, a *GamePlayer* specializes both *AppUser* and *Performer*. More information about how *AppUsers*, *Performers*, and *GamePlayers* are related with the other entities can be respectively found in the *User and Social* Model (described next), in the *Action* model, in the *Gaming* model, and in the *Conflict Resolution* model described later in this Section.

### 7.1.2 Modeling Users' Social Space

Fig. 7.3 depicts the model in charge of representing users' relationships and interactions in the social space. The model refers to the topmost user type of the taxonomy in Fig. 7.2, so to represent people regardless of their affiliation to the human computation platform. Users are also related to each other through *UserRelationships* of a given *UserRelationshipType* (e.g., friendship, geographical proximity).

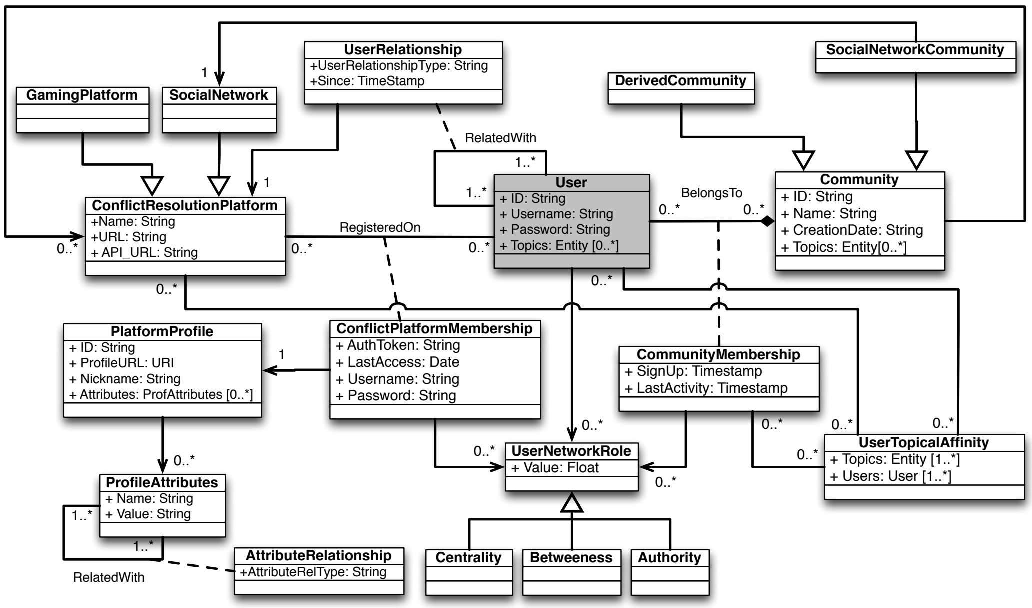



Figure 7.3: User and Social Model

The social space consists of *ConflictResolutionPlatforms*, i.e., platforms where users can perform tasks. For example, *SocialNetworks* (e.g. Facebook, Google+, LinkedIn, Twitter) and *Gaming Platforms* (e.g. Apple's Game Center, Microsoft's Xbox Live!) are specific types of *ConflictResolutionPlatforms.*

A *User* can be subscribed to zero or more *ConflictResolutionPlatforms*; each subscription goes with a *ConflictResolutionPlatformMembership*, i.e. an entity that contains the main authentication credentials to the social platform, plus some metadata information that describes the User in the platform. Examples of such metadata are: (i) a *PlatformProfile*, i.e. the set of personal user's details stored within the platform, which includes also an open set of *SocialProfileAttributes* (e.g. birthdate, hometown, preferences); in our model, profile attributes are represented as a flat list of properties, but the adoption of more expressive representations (e.g. graphs) is supported. (ii) A set of *UserNetworkRole*s, that represent a measure of the importance of the *User* within the social network space; examples are classical indicators such as *centrality*, *prestige*, and *authority.* (iii) A set of *TopicalAffinities*, i.e., relationships with topics. An affinity link is represented by the *UserTopicalAffinity*, which embodies pointer to topics described as *Entities* of the *Content and Content Description* model.

Another central concept is that of *Community*, defined as a group of interacting people, living in some proximity (i.e., in space, time, or relationship) and sharing common values. A *Community* is characterized by a *Name*, and by a set of *Topics* that define the common values that tie together its members (*Topics* are described by entities).

A *CommunityMembership* denotes the metadata about the community members, including: (i) a set of *CommunityMetrics* i.e., measures of the importance of the *User* within the *Community*; (ii) a set of *TopicalAffinities*, i.e., topical relationship with a given (set of) topics.

Users may have affinities only to a sub-set of the *Topics* that describe the community, and such affinity can involve other *Users*. Communities can be real, that is, proper subgroups of a *SocialNetwork* (denoted as *SocialNetwork-Communities*) or *DerivedCommunities*, i.e. communities that span multiple social platforms according to some criterion (e.g., the union of the Facebook and G+ groups of Star Trek fans). The *User and Social Model* allows for a definition of *GlobalNetworkMetrics*, i.e., metrics that define the aggregate importance of a *User* across both social networks and communities.

## 7.2 The Content Description Model

The *Content Description* model contains the concepts that denote the assets (e.g., blog posts, tweets, images, videos, entities of interest) associated with the user's activity, the metadata (i.e. *annotations*) that describe such objects, and their associations with the users that produced them. The *Content Description* model is inspired by several existing content representation format such as RuCOD [153] or MPEG-7 [154], on which it can be trivially mapped.

The model element *ContentObject* shown in Fig. 7.4 denotes an abstract entity representing a piece of information that can be accessible through some kind of storage system (e.g. relational, no-sql, or graph databases). Each *ContentObject* is defined by (i) an *ID*entifier, to uniquely refer to a piece of content; and (ii) a *URI*, a string that unambiguously identifies the location of the *ContentObject* in the storage system (e.g. `http://en.wikipedia.org/wiki/File:Mpeg.gif`).

*ContentObjects* can be related to each other; such a relationship, that materializes in *ContentRelationship* objects, may be motivated by the presence of existing or created physical or logical relationships: for instance a video object can be related to the HTML page object that contained its description; likewise, a video object can be related with the thumbnails (i.e. image objects) automatically generated from its keyframes. The *ContentRelType* attribute identifies the type of relationships (e.g. *CrawledDescription*, or *DerivedObject*).

The *Content Description* model also comprises a set of entities and relationships that express knowledge about a *ContentObject*. This knowledge, typically expressed as a metadata *Annotation*, can be automatically or manually produced, and helps describing the *ContentObject* for search and retrieval purposes.

A *ContentObject* can feature zero or more *ContentDescriptions*, where each *ContentDescription* is characterized by a unique *ID* and by a *Name* that help identifying the scope of the description (e.g., the same content can be described multiple times by several parties)[1].

*Annotation*s express metadata that composes a *ContentDescription*, and they describe the *ContentObject* as a whole. They are characterized by an *AnnotationScheme* (which uniquely identifies the type of annotation according, for instance, to the annotation component that generated it), a *Name*, a *CreationTimeStamp*, and a *textual Description* (or the DescriptionURI that points to an external description). There exist several kind of *Annotation*s; for instance: (i) *TextAnnotation*s contain textual values in a given Language; (ii) *Low-Level Features* contain array(s) of numerical values, typically representing the result of a numerical analysis of the content item; (iii) *Entities* are semantically defined metadata that correspond to real world objects or occurrences as described in ontologies (i.e. DBPedia or Yago[155]).

An *Annotation* is typically created by automatic or human *Actions*, as described in Section 7.3. Annotations can belong to an *AnnotationAggregate*, to denote the fact that multiple annotations have been created by the same *Action* (e.g., a set of image tags created by an instance of a GamePlay with an image tagging GWAP), or that there exists a logical or functional dependency

[1]Please notice that content descriptions typically also include information about temporal of physical *segments* of the described objects. For the sake of brevity we omitted the description of this important content description aspect, although fully supported by our model.

(expressed by the *AggregateType* attribute): for instance one annotation can be created as a *refinement* of another one after being validated and corrected in a human computation task. An *Annotation* can be associated with *AnnotationConfidence* objects that state the level of uncertainty associated with the truth-value of an annotation.

**Uncertain Information Representation** An important issue when dealing with human and automatic computation is the management of uncertain information, because both algorithms and user's contributions are approximate and their trust level can be appraised only probabilistically. Uncertainty can be related to several concepts in the system, and, typically, is the result of an approximate approach to the determination of a given fact. For instance, textual annotations produced by automatic classification algorithms are commonly associated with a trust value, i.e. a number that estimates the correctness of the given classification. Fig. 7.4 depicts how uncertainty is described in our model: under the generic term of *Confidence*, we define the uncertainty degree associated with a piece of information, and we allow such degree to be expressed as a *Confidence Value* (e.g. 0.8), as a *Confidence Interval* (e.g. $[0.6, 0.8]$), or as a *Probability Distribution* of a given type (e.g. normal, Poisson, etc.).

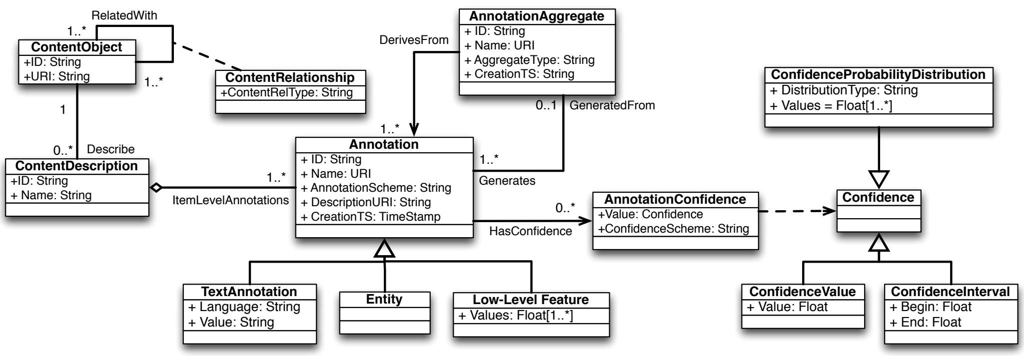



Figure 7.4: Content and Description Model

## 7.3 The Action Model

The *Action* model describes two types of actions that can be performed on content objects: *automatic actions* done by software components like classification algorithms, and *human actions*, performed by users to detect and resolve conflicts, provide relevant feedbacks, etc. These human actions are called *Tasks*, which can be executed with a variety of approaches, from answering a query, to performing work on demand.

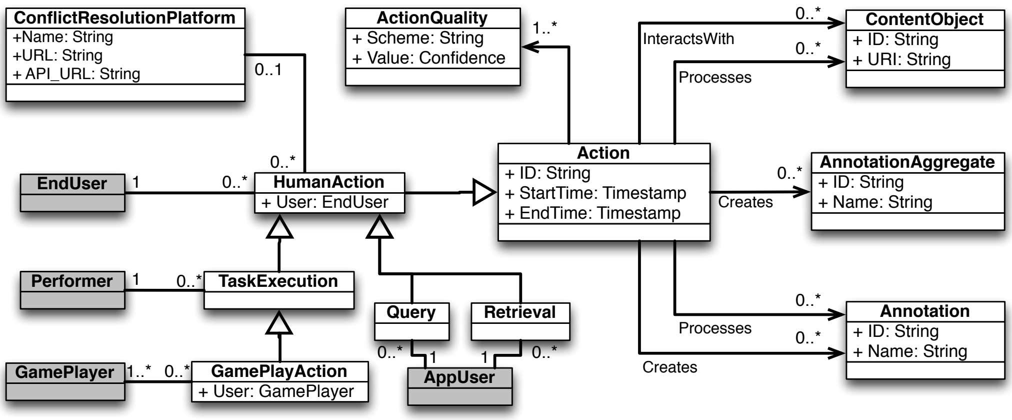



Figure 7.5: The Action Model

The Action Model depicted in Fig. 7.5 is strictly related to the *User and Social* model, as it represents the spectrum of actions that humans can perform on a *ConflictResolutionPlatform.*

We define an *Action* as an event (happening in a given time span delimited by a *StartTime* and an *EndTime*) that involves the interaction with-, the processing, or the creation of-, *Content Object*s, *Annotation*s, and *AnnotationAggregate*s. *Action*s can be associated with one or more *ActionQuality* values, i.e. values that denote the correctness or the completeness of an action. For human computation platforms, we distinguish *HumanActions* that fall under three main archetypes: *Retrieval*, *Query* and *TaskExecution* actions.

The first two are examples of interactions performed in applications, and they involve the querying, consumption, or collection of content items. *TaskExecution*s, instead, relate specifically to human problem solving activities (e.g. rating, tagging, disambiguating, recognizing) and, therefore, are executed by *Performers*. A *GamePlayAction* is a specific type of *TaskExecution* that leverages the entertainment capabilities of online games in order to exploit *Game Players* to solve human computation tasks. More details about games are described in the *Gaming* model of Section 7.5.

## 7.4 Human Computation Model

The *Human Computation* model, depicted in Fig. 7.6, expresses the uncertainty arising from automatically computed social data and content objects' metadata. It also deals with conflicting opinions that may arise when humans are requested to perform a piece of work that may entail judgement or errors. The *Human Computation* model is related to the *Action* model, as *conflicts* are the source of tasks for human solvers, and it revolved around the concept of *Conflict*, i.e. a situation during the analysis of a given *ContentObject* where absence of contradictions about facts recorded in *Annotations* may arise.

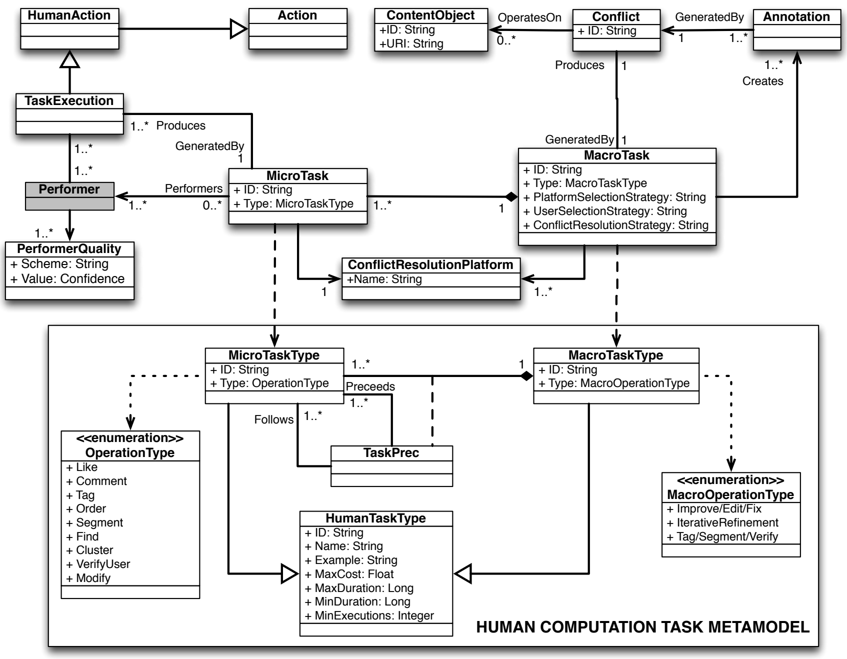



Figure 7.6: Human Computation Model

*Conflicts* typically happen in the following scenario [156]:

- **Missing Annotation**: during a given annotation *Action*, the performer (an annotation component or a human), is not able to find a suitable *Annotation* for the analyzed content;

- **Uncertain annotation**: during a given annotation *Action*, the performer creates *Annotation* having an *AnnotationConfidence* within a given interval of confidence, thus leaving uncertainty about the truthfulness of such a value. For instance, referring to the running example, the annotation component for garment recognition reports the presence of a "poncho" in an image containing a "hat". Uncertainty might also arise when the annotation *Action* has been performed within a given interval of *ActionQuality*, thus raising doubts about the actual quality of the annotation activity. For instance, this scenario may occur when the actions, performed by a *Performer*, have been marked (automatically by the system, or manually by another user) as poorly executed.

- **Inconsistent Annotations**: a conflict may happen in merging the *Annotations* between two different annotators for the same *ContentObject*. For example, some *Annotation* could be associated with a high *AnnotationConfidence*, however, they may lead to a wrong conclusion when they are put together, or they may contradict a fact that the system might not know yet.

According to the definition above, a *Conflict* is therefore characterized by an *ID* (identifier), by a (set of) *ConflictualContentObjects*, and by a (possibly empty) set of related *ConflictualAnnotations*, i.e., the set of *Annotations* that generated the conflict.

When a *Conflict* occurs, it might need to be managed by human-enacted activities. Such activities are defined in a *MacroTask* that consists of a set of human-enacted atomic *Task*s. *MacroTask*s and *Task*s are instances of human computation activity archetypes defined in the Human Computation Task Metamodel of Fig. 7.6.

A *HumanTaskType* is the abstraction of a piece of work that needs to be completed by a specific amount of performers in a given period of time. A *HumanTaskType* is described by an *ID* (identifier), by a *Name*, and an *Example* (a textual description of the activities associated with the type of task). A *HumanTaskType* typically details some constraints about the execution of the type of task; for instance, the *MinDuration* and *MaxDuration* allowed for the overall task execution, the *MaxCost* allocated for the task, etc.

*HumanTaskTypes* specialize into *MicroTaskType* and *MacroTaskType*. A *MicroTaskType* represents a unit of human computation activity performed by one or more *Performers*; in the human computation literature, *MicroTaskType* can be defined as highly fractioned tasks that do not require specialized skills and can be completed in a small amount of time. A *MicroTaskType* is characterized by an *OperationType*, i.e. a specific human computation activity type. Examples of *OperationType* are preference tasks and data manipulation tasks [157]. The former correspond to typical social interactions (like, dislike, comment, tag), while the latter (create, order, complete, find, cluster) abstract simple and classical primitives of relational query languages that are common in human computation and social computation activities.

A *MacroTaskType* represents an aggregation of one or more *MicroTasks*, organized in a workflow in order to achieve a high-level goal. An example of *MacroTaskTypes* is the human computation "Tag/Segment/Verify" pattern that is used in the game described in the running example. Within a *MacroTaskType* aggregation, *MicroTaskTypes* present precedence relationships that define their order in the workflow. The "Tag/Segment/Verify" pattern, for instance, can be instantiated by pipelining three micro tasks.

A *MacroTask* has a given *MacroTaskType*, and it is typically executed on one or more *ConflictResolutionPlatforms* (e.g. social networks or human computation frameworks). The selection of the involved *ConflictResolutionPlatforms* is done through the application of a given *PlatformSelectionStrategy*, i.e. a numerical, logical, or heuristic method that decides which are the best platforms to adopt for the solution of a *Conflict*; for instance, if the conflict resolution task involves the evaluation of fashion pictures, then the system may decide that the best platform to tap for human computation is Facebook rather than Amazon Mechanical Turk. Likewise, a *UserSelectionStrategy* is a method that decides, for the selected *ConflictResolutionPlatforms*, which are the best performers to involve in order to satisfy the constraints defined in the *MacroTaskType* definition; for instance, in the running example *Pippa* is chosen over *Michael* to play a game since it is one of her passions. Finally, the *ConflictResolutionStrategy* decides how to split the execution of *MicroTasks* among the selected *Performers*, deciding, for instance, which conflictual *Annotations* or *ContentObjects* will be assigned to each *Performers*; moreover, the *ConflictResolutionStrategy* dictates the result aggregation method (e.g.

Majority Vote) to use for creating the final output of a *MacroTask*, which, typically, consists of one or more new *Annotation* objects

The decisions undertaken by the *PlatformSelectionStrategy*, *UserSelectionStrategy*, and *ConflictResolutionStrategy* are directly mapped into the *MicroTask*s that compose the given *MacroTask*, as each *MicroTask* is executed on a *ConflictResolutionPlatform*, by a (possibly singleton) set of *Performers*, operating on a (possibly overlapping) subset of the *ContentObject*s and *Annotation*s.

As a *MicroTask* can be assigned to several *Performer*s, each running instance of a Microtask to be executed is associated with a *TaskExecution*, a type of *HumanAction* that contains information about the *StartTime*, *EndTime*, and *QualityMetrics* of the work performed by the single *Performer*, plus reference to the *Annotation*s created during the specific execution and related to other *Annotation*s or *ContentObject*s.

An *Annotation* (or a set thereof) created during a *MacroTask*, can be, by definition, conflictual, and thus be the source of a new *Conflict* within the platform. The decision about the right course of action to undertake is typically related to the selected *ConflictResolutionStrategy.*

## 7.5 Gaming Model

The *Gaming* model focuses on a specific class of tasks deployed in the form of a game with a purpose (GWAP) and expresses the engagement and rewarding mechanisms typical of gaming (including gaming scores, leaderboards, and achievements). The *Gaming* model is related to *Action*, *User & Social* and *Human Computation* models to denote the assignment of a gaming session to a player for solving a *Task*.

The Gaming Model is depicted in Fig. 7.7. A *Game* is an entertainment application described by a *Title* and characterized by a *Genre* (e.g. Puzzle, Educational), a *Mode* (Single Player, Multi Player), and a *Theme* (e.g. Abstract, Comic, Crime, Science Fiction). A *GamePlay Action* is a human computation action that the user has performed while playing a *Game* during a specific session of that game, the *Gameplay.* Since the *Gameplay* tracks all the actions performed by different players during a specific running game, it is possible to retrieve social information regarding the relationship among the

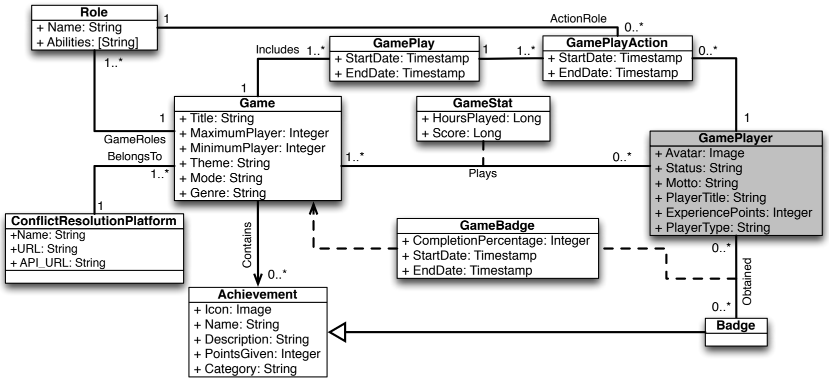



Figure 7.7: Gaming Model

gamers. A *Game* may also have a list of available *Roles* that a *GamePlayer* may assume during a specific *GamePlay*; in the running example, the roles for the player can be the Sketcher and the Guesser. A *Role* can be described with a *Name* and a list of *Abilities* that define which are the allowed actions in the game for a particular role. A *Game Player* is a type of user described by customization attributes (e.g. *Nickname*, *Motto*) and accomplishments: for instance the *PlayerLevel* attribute represents the proficiency and the experience of a player; the *PlayerTitle* is a special recognition given to the player for his actions (e.g., a chivalry role); the *PlayerType* (e.g. Achiever, Explorer, etc.) is used to associate the player with a particular cluster of gamer type. *GameStats* are stored in order to keep track of the *HoursPlayed* by a player on a specific *Game*, the *Score* he has obtained or other meaningul variables. Games have *Achievements*, i.e. means to foster an entertaining experience for users and a way to profile them. An *Achievement* is a specific challenge or task that the player can perform in order to get a reward in terms of points or other special features (in-game items, artworks, behind the scene videos); it is defined by a *Category* that specifies which kind of task the achievement was associated with such as Quests, Socializers, Grinders and the like as they have been defined in [158] and a *PointsGiven* attribute which contains the amount of points to be given if the requirements for the achievement have been met. Once a player reaches the goals of a listed achievement, she will gain a *Badge* related to that specific achievement. A *GameBadge* is used to relate a player with the achievement she has obtained, and it is described by a *CompletionPercentage* attribute that shows how much the player has already achieved in order to complete a specific task.

# Part IV

# Case Studies

In the following, two detailed case studies that make use of the development process, the mechanics and the data structures defined in the previous sections are described. Both of the applications has been developed and tested along with a consistent number of users to achieve two different results: the first one is an Enterprise gamification platform that has been used to increase the participation level for existing users in order to achieve desired business objectives that were not met in everyday usage; the second one is a gwap that has been integrated in a human computation framework that was developed for fashion trend analysis in order to provide a mean to gather human contributors in image segmentation tasks.

# Chapter 8

# Gamified Application: Webratio Headquarters

In this chapter, we present the implementation of an application gamified following the game design techniques explained before. Section 8.1, will introduce the environment of the company while Section 8.2 will detail the requirements specification that lead to the development of a Gamification approach. In Section 8.3 the focus is on the Gamification design and on the game elements included to solve the business goal defined by the company and Section 8.4 describes the architecture of the application, starting from the analysis through UML Diagram, Data Model, the integration with the existing modules and Hypertext Model in which well describe the IFML Model used to develop the application.

Finally, Section 8.5 of the chapter will be entirely dedicated to the evaluation of the application developed, based on data collected after a phase of Beta testing, finding if the Gamification approach was right to solve the requirements specification.

## 8.1 Background

WebRatio is a development tool used to design complex data-intensive Web Application with the use of Web IFML (Interaction Flow Modeling Language), a modeling language dedicated to the definition of "User Interaction" dynamics between an application and its user.

Given the fact that Web IFML is not widespread as other languages, the only source of information and training is the material offered by the company, through its forum and tutorials. Before the introduction of the gamified application, the components that the company used to interact with its current and potential users were: an on-line portal with information about the company and the product, a forum in which users can seek for information, a store to download extensions for the tool and an e-learning center. **on-line portal** The institutional website is the first entry point to the world of WebRatio. In this sense, the website was built to be a corporate portal, suited for users who do not know the product but would like to get information. On the website, the user can download the free version of the tool and can register herself not only to activate it but also to access the other applications (Forum, Store and E-learning center). **Forum** It is an essential element for any community, used to share ideas, suggestions and ask question related to the use of the tool and web development in general. **Store** Is an application that allows users to expand their personal set of "units" the building blocks of a Webratio application or downloading new components or style templates to enhance their application from a functional or visual perspective. The store offers also the possibility for the advanced users to publish new components to sell. **E-learning center** The Learning Management System (LMS) is a newly developed platform with the aim of training the users by teaching them how to interact correctly with the building tool through a series of articles, videos and short quizzes.

## 8.2 Objectives of the Gamification Effort

**Business Objectives** The main problem that the company faced and for which required an intervention in the form of a gamified solution was related to the fragmentation and heterogeneity of their online tools, most of which were unknown or not used to their potential. For this reason, the most important business objective to reach was the creation of a unique entry point, an integrated platform for each user that used Webratio and capable of improving the connection and functionalities among the various existing resources; this would led to a unique and complete user experience, allowing and encouraging the users to use all the tools to their full extent.

Active participation of the users in a community able to span different components, including not only the Forum, but also the Portal, the Store and the E-Learning center was the second most impelling issue: by increasing the participation, the company aimed at creating a group of loyal users and aficionados to involve in all the proposed activity of the company, effectively self-sustaining the community itself. At the same time, taking care of such a community could lead to a word of mouth effect that would be beneficial for the market share of Webratio, not only by mean of acquiring new users, but also by attracting the interest of new business firms that could judge Webratio as a competitive and professional tool sustained by a lively userbase. **Target Players** Being born as a Spinoff of an academic research project, students and university professors have always been the target for Webratio in order to expand the popularity of the tool, but using it for personal or educational reason is not the approach that can sustain am entire business.

For this reason, the target userbase for the application should include also clients that use the tool to generate applications for their own personal business or Webratio's partner that promote and share the use of the tool as a new methodology to build fast and reliable web applications.

Since we are focusing on users that will take advantage of the new gamified platform, it is worth noting that also Webratio's employees would make use of the Forum, the E-Learning platform and the Store to increment their know-how, thus they have to be taken into consideration during the design phase. **Target Behaviors** To reach the stated business objectives, several could be the actions and the behavior to encourage:

- Registering to the platform, in order to track the users and all the activities performed within the system and being able to profile him.
- Registering the serial of a product, in order to distinguish between partners, firms or common users and apply different strategies tailored to the specific needs.
- Logging into the platform, to keep the sense of a lively community.
- Posting or Replying to questions in the forum, in order to mantain a self-sustained, user centered technical support tool.

- Downloading or Publishing content to the store, to extend the features of Webratio without the need of employing workers belonging to the company.

- Training with the use of the E-Learning platform, to increase user awareness of the user on the platform, reducing the technical questions and improving the overall quality of the applications created with Webratio

## 8.3 Gamification Design

To fulfill the defined business objectives, the gamified web application has been designed with the goal of managing all the possible operations performed by the users using a common and integrated interface, thus solving the fragmentation problem of the tools and creating a unique and homogeneous user experience. All the actions necessary to fulfill the business objectives were already in place within the system, thus there was no need to introduce new ones. Goal of the implementation of a gamified application is thus encouraging the proper use of the existing actions, by introducing mechanics able to rewards the players for their contributions.

To accomplish this result, we opted for a Point, Badges, Leaderboard system (PBL), augmented with the introduction of digital quizzes, physical rewards and filter to enhance competition among new players without discouraging them due to the presence of experienced users that were able to amass considerable amounts of points.

Users inside the new gamified community can earn points through operations that take place in one of the modules of the system, for example, by posting a question in the forum, completing their profile data, purchasing some products from the store and so on. Relating actions in the system to particular gains in terms of points is something that should be customizable: our gamification platform will reflect this principle, by allowing the "gamification designer" a special role in the company, to decide which actions to gamify, how much points to assign and grouping them based on different themes and goals that have to be enforced.

A detailed list of all the actions that has been considered meaningful to be rewarded are listed in Table [TODO] which has been part of the design effort of the gamified application. The point values have been assigned at first intuitively, by rewarding with higher amount of points the actions that were contributing more to the objectives considered to be most relevant; the numbers have then been adjusted during the testing stage of the application when balance issues where found.

To better distinguish the ability and contributions of the users within the system and rewarding users that are not only knowledgeable but also active, we have decided to assign points to actions by dividing them in two different categories: participation and reputation actions.

The concept of reputation is adopted in order to express, through a numerical value, the ability of an user and use it as an extrinsic motivator. Ability and proficiency are measured in the platform when completing lessons, trainings, questionnaires, quizzes and exams in the learning center or by participating to contests and special events that require reasoning or skills from the user side. The higher the reputation, the greater is the ability of the user, so through this parameter the users in the community can identify which are the experts; in this way their contribution will be valued more and external firms, partners or even Webratio itself may contact them to offer employment positions. For these reasons, introducing reputation as a mechanic increase the user's desire of boosting it to become one of the leaders in the community. Reputation points are never shown in the application: once a certain amount of reputation points are gathered for a particular group, they are converted into badges or physical rewards. Figure 8.1 shows all badges that have been designed for the initial implementation of the application.


Figure 8.1: WebRatio Community badges

By analyzing the four different separated components (Portal, Forum, Store and E-Learning), we have identified four different areas of reputation: certificates (purple badges), forum activities (blue badges), store contributions (orange badges) and Learning Management System Rewards (yellow badges).

After the user has performed a number of actions and achieved the required score, he automatically earns a badge that proves her abilities and skills in a particular module of the WebRatio world. The set of actions required to achieve one particular award are obviously all the actions classified as reputation relevant, because their fulfillment require an objective capability in the use of WebRatio.

The second type of points that have been designed are grouped under the “Participation” category. Users acquire participation points, used as an intrinsic motivator, by playing “an active role” within the community, for example by filling their personal information on their profile, reading an article in the E-learning center, logging into the platform and other similar activities. The participation score is an numerical value that can grow up very quickly and thus of difficult interpretation; for this reason, the participation is displayed as a percentage, by using a progress bar, in relationship with the user with the greatest participation score (the first in the participation rank).

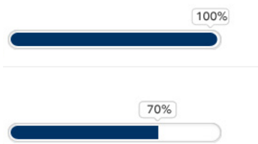



Figure 8.2: WebRatio Community progress bar

With this purely visual strategy, the user can perceive how her participation increase or decrease in real time, thus committing to an active community participation in order to fill the progress bar as quickly as possible. The sum of all the participation points acquired corresponds to an equal number of *WebRatio Credits*, virtual coins that the user can spend to buy components from the store, to obtain gadgets or professional technical support directly from the company. The introduction of these virtual goods is a powerful methodology of reward that promotes and encourages the participation into the community, making it lively and active.

The gamified application has to be composed by three elements: a set of *leaderboards*, used to make comparisons among the players, a *user's dashboard* and a set of *accessibility elements* for all the modules.

Leaderboards are shown in the home page related to the community, by displaying only the top positions, and in a dedicated page in which all the possible rankings are displayed and which contains:

- A monthly ranking where only participation points acquired in the current month are considered.
- A global ranking where it is possible to visualize both the participation and the reputation rank (by progress bar and badges).
- The company ranking, where companies and universities are listed and ordered based on the sum of all the scores of their employees or students.

Even though the company ranking may seem less important, it is instead a key indicator: no company would like to be at the bottom of a leaderboard or worse behind a direct competitor. For this reason, an active community is going to be composed by companies' employees or universities' students/professors pushed by their own institutions.

The ranking pages allow also to have a customized view by filtering the rankings based on several parameters. Among all the filtering possibilities, a meaningful one is the possibility to rank the results based on the geographical region. By default, the ranking are calculated based on all the world, but the customers can choose to see only Europe, Latin America, North America and Asia, Africa and Oceania. In this way the user can decide to see only the specific rank of his geographical area; this choice has been done due to the fact that some people may dislike to compare themselves with people from a different country. If a user is logged in while accessing the home page, the leaderboard shows her position and the players which are immediately before and after her, without the need of comparing against the top users. As it has been described in the previous chapters, this design decision foster competition even for new players, without discouraging them with impossible goals, since reaching the top of the leaderboard could be extremely time consuming.

The user's dashboard is a summary page in which all the information related to a user are shown, along with the badges she has acquired, the level of

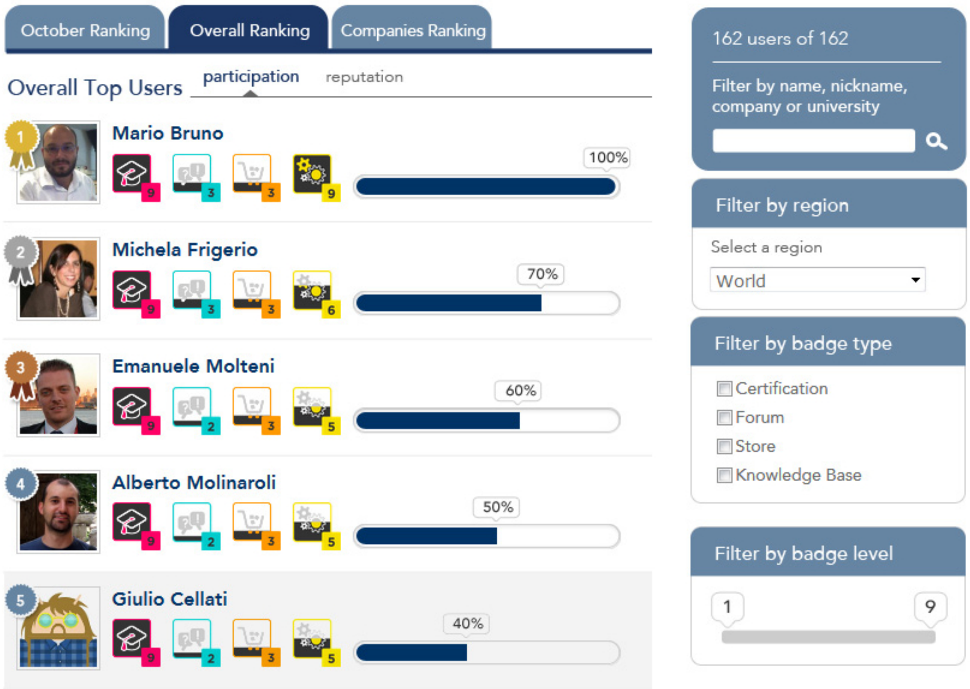



Figure 8.3: WebRatio Community leaderboards

participation, the latest post where the user interacted and so on; every action performed by the user can be shown or hidden, based on the gamification designer policies and the privacy settings of the user. It may be displayed in two different modes: public and private.

The public dashboard is composed by the social information of the user, the highest level area badge she has acquired, her participation and reputation score, both with a rank number to show her position in the overall leaderboard, and a section that shows the most important certificate she has acquired.

The private dashboard offers the same functionalities as the public dashboard, plus the possibility to view the complete history of the acquired badges and certifications, along with a detailed list of all the actions that have been performed in order to reach such results; for this reason, the private dashboard is a complete log of the entire community life of the user, data that will be extremely useful to profile the users offline.

The third component present in the dashboard pages and in the home page is a set of accessibility elements that show three boxes related respectively to the Forum, E-learning center and Store; they also shows the topics/articles/components with which the user has recently interacted with and

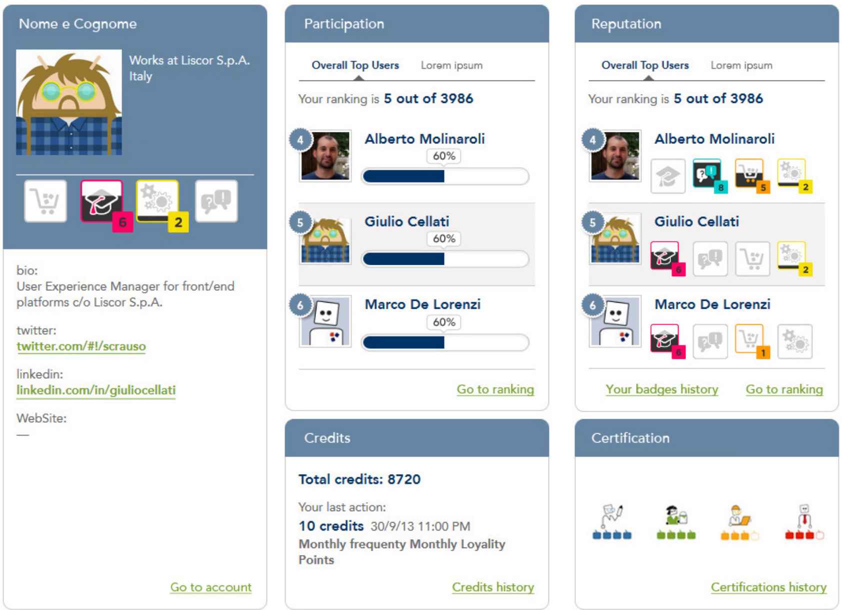



Figure 8.4: WebRatio Community dashboard

topics/articles/components that could be interesting for him, thus solving the fragmentation issues that were present prior to the introduction of the gamification platform.

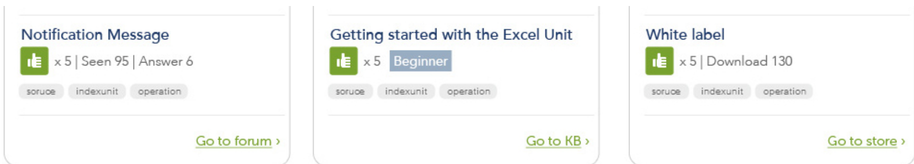



Figure 8.5: WebRatio Community collector element

## 8.4 Architecture

After having examined all the business goals, the idea of creating a centralized entry point for all the Webratio's public activities was born. The company needed an application able to unify all the other modules, already present on the web (Forum, Store, LMS and Institutional Website), by creating an unified user experience. According to these goals the best choice that could fit all the requirements was a gamified web community, called **WebRatio Headquarters**.

Thanks to it, all the customers using Webratio would obtain visibility and improved engagement, while also increasing attractiveness towards new users and potential clients. All the existing modules were developed within WebRatio and the community will also be created with the web modeling software of the company. WebRatio offers, through IFML (Interaction Flow Modeling Language), a visual standardized modeling tool to create applications. Thanks to a dedicated modeling language it is possible to define the user interaction dynamics between an application and its user.

Before explaining in depth the architecture of the gamification platform, it is necessary to define some of the desired properties of the application:

- Usability, the extent to which a product can be used by specified users to achieve specified goals with effectiveness, efficiency, and satisfaction in a specified context of use. A usable application should allow the users to accomplish basic tasks the first time they encounter the design and perform them quickly. When users return to the designed application after a period of not using it they should easily reestablish proficiency with it. The application should also be designed to minimize errors and be pleasant to use.

- Performance, measured by the output behavior of the application. A good performing web application is expected to render a page in around or under one second (depending on the complexity of the page). It is especially important to maintain good performances even under load.

- Extensibility, a system design principle that takes future growth into consideration. It is a systemic measure of the ability to extend a system and the level of effort required to implement the extension. Extensions can be obtained by introducing new functionalities or by modifying existing ones; this is easily accomplished in Webratio thanks to the customization offered by the units.

Provided with these non-functional requirements, we can now describe the functionalities and architectural details of our gamified application.

### 8.4.1 Use case

Given the requirements described in the design phase, in Figure 8.6 we describe a Use Case diagram showing the most important actions that a user can perform inside the community:

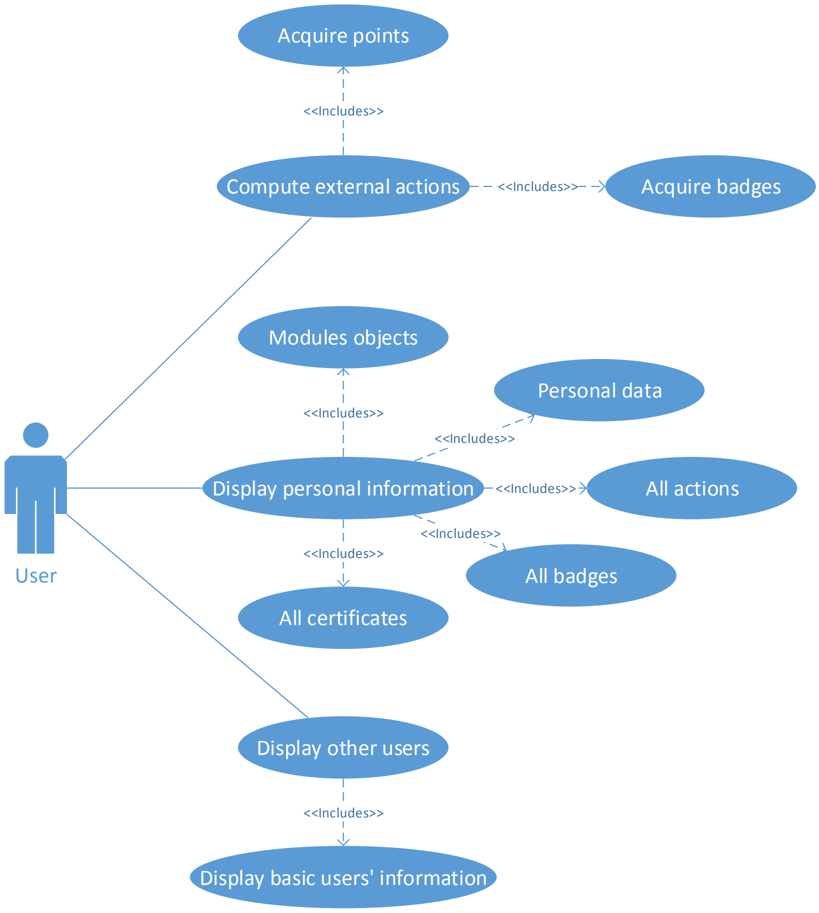



Figure 8.6: Use case diagram of WebRatio Community

- Compute external actions: consumers can perform several of operations in the various components, such as posting a new question inside the Forum, reading an article in the LMS, acquiring WebRatio certifications,etc.... Albeit all these actions take place outside the physical pages of the community, they are strictly linked to it. In fact each action contributes to increase the participation of the user, that will be prone to share her knowledge and experience. Every time that a new action is performed, the customer receives points regarding participation, reputation and credits, based on the activities that have been defined by the designer. Moreover the system checks, through specific gamification logic, if the user is entitled to receive an achievement and, if it is the case, it assigns the badge to the user.

- Displaying personal information: users can display all their information, from the classic personal data (first name, last name, company name, social accounts), to all the actions performed (in chronological order) that they have performed, to the history of the badges acquired with the date of acquisition and the certificates of completion for the LMS.

The certificates and the badges are shown through a graphical representation that shows even the achievements that the user has yet to obtain, in this way the gamification mechanics can fulfill their motivational role. The customers can also navigate through the LMS articles read, courses done, video watched, Forum post where in which they took part, the components downloaded from the Store and an expansive set of new object suggested in each modules.

- Displaying other users: Any user can consult the public profile of the other users in the community. Even the latest objects and activities performed in the other components are visible.

### 8.4.2 Data model

A crucial point in all software developing process is the planning of suitable data models able to support the software's functions. For these reasons, to manage all the mechanics (points, leaderboards, achievements and badges) and features we have designed a data model based on the main principles of gamification and the considerations done in Chapter 6.4

As shown below, the structure of the community database is developed to support all the gamification requirements and to be as independent as possible with respect to the other WebRatio's components. For each of the tables of the database, a description its structure and its usage within the application is provided.

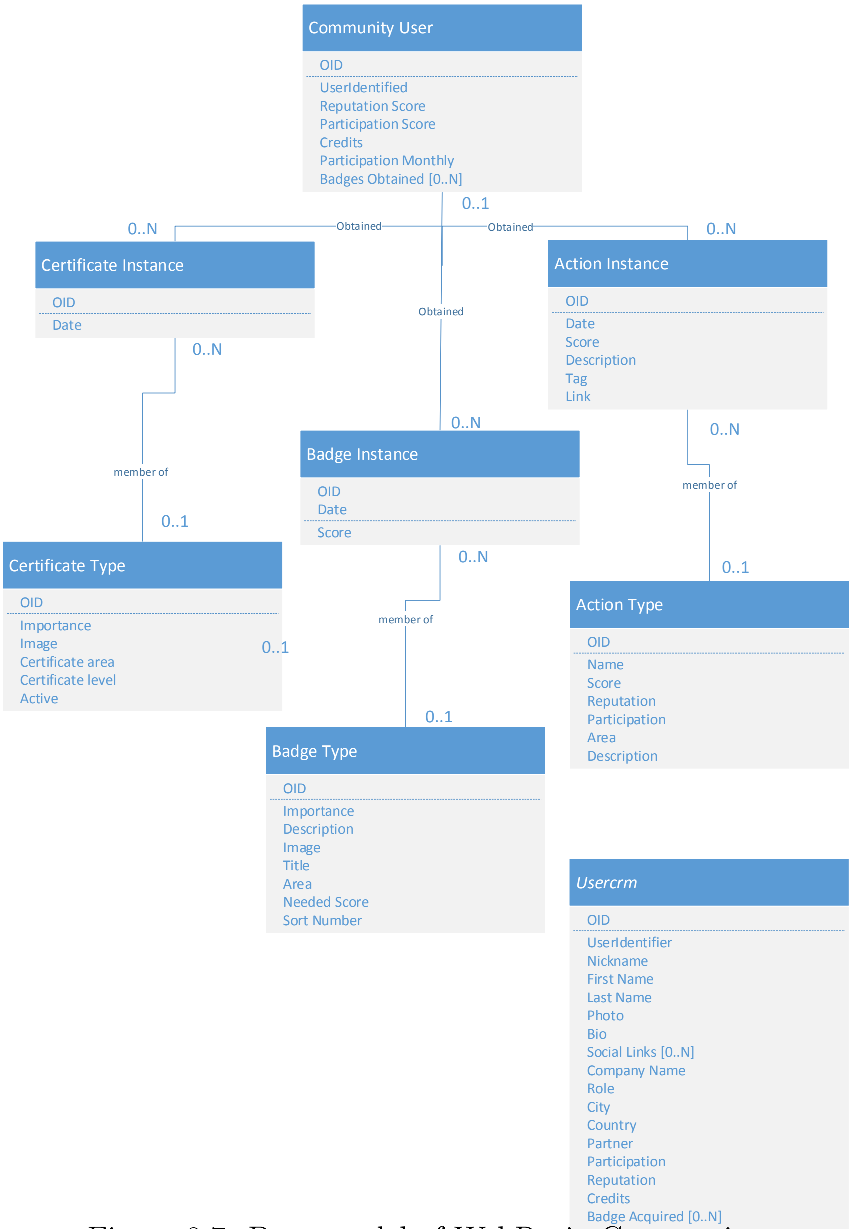



Figure 8.7: Data model of WebRatio Community

**Community User**

Is the table in which all the users of the community are stored. For each one, personal data (biography, social accounts, photo, first name, last name, nickname, etc.) and all the points (participation, participation monthly, reputation and credit) with the most important badges acquired are gathered.

The nickname, if set, is the attribute that identifies the user within the community, otherwise the first name and last name are shown; the table is filled the first time the user performs an action.

**UserCrm**

The Community User table is strictly connected to a database view which has a fundamental role inside the community architecture. UserCrm is a cross-database view that has been created with dblink, a module which supports connections to other PostgreSQL databases from within a database session. It is used to link the table Community User, related to the community database, with the Contact table related to the WebRatio CRM (Customer Relationship Management) database. In this way the community can access all the WebRatio existing userbase without using directly the business database of the society.

In general the big difference between the Community User table and the UserCrm view is that the first one is used to store in a persistent way all the customers who have participated actively within the community and to allow the creation of relationships among the users and all the table representing gamification features in the community database (Action Instance, Badge Instance, Certificate Instance). Thanks to the view it is not necessary to import all the personal data regarding each user a second time, thus fulfilling the integration requirement with the existing infrastructure.

There are three tables are used as dictionaries: Action Type, Badge Type and Certificate Type; they represent the key features of their objects while leaving to the gamification designer the possibility to extend them to introduce custom features.

These table are created to be the keys of their objects, are developed to be completely customizable by allowing the system administrator to define custom features.

**Certificate Type**

The Certificate Type table encompasses all the WebRatio academy certificates divided by area, level and importance.

The certificates are written official statements that certify the abilities and the knowledge in the use of the WebRatio.

This table has been created just for maintenance reasons, because it was necessary to have a schema where all the type of certificates were described and categorized, to allow easy modifications by the gamification designer without requiring interventions on the code. It should be filled by the designer once, prior to the deployment of the gamification platform and seldomly updated afterwards. Table 8.1 shows a possible entry for such a table.

Table 8.1: Information model of Certificate Type

| OID | Key | Importance | Image | Checked Image | Area |
|---|---|---|---|---|---|
| 1 | area.level | 1 | cert-01-no.jpg | cert-01.jpg | Certification |

| Level | Active | Sort Combination | HD Checked Image | HD Image |
|---|---|---|---|---|
| Trainee | False | 1 | cert-01.jpg | cert-01-no.jpg |

**Action Type**

The Action Type table is where all the activities that can be performed in the community are described; in this way all the actions that are related for the gamification mechanics are grouped and can be easily added or modified with respect to the needs of the company.

All the actions are described by the area of WebRatio where the event can take place, the score to assign to the user, if it affects reputation or participation, the action's name and a brief description.

The table is managed by the gamification designer who compiles it prior to the deployment of the system, but he can manage and monitor it even afterwards to guide the community behavior and modifying points or actions to balance the system. Table 8.2 shows a possible example of action in the system.

Table 8.2: Information model of Action Type

| OID | Name | Score | Reputation | Participation | Area | Description |
|---|---|---|---|---|---|---|
| 32 | Post a question | 10.00 | False | True | Forum | Post a question |

**Badge Type**

The Badge Type table is used to describe all the badges that the community offers to its users. For each badge, it is defined one of the four areas in which it can be gained, the score required to obtain it and its relative importance in relationship to all the other achievement grouped by area. As the two previously described tables, it is created for the management of the badge by the gamification designer; thanks to its flexibility it allows to apply several tweaks to the gamification rules even while the community is running to balance or adjust them. Table 8.3 shows a possible example of badge for the system.

Table 8.3: Information model of Badge Type

| OID | Importance | Description | Checked Image | Image | Needed Score | Area | Title | Sort Number | Key | HD Checked Image | HD Image |
|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | 1.00 | First answer | for-01.jpg | for-01-no.jpg | 100.00 | Forum | Starter | 1 | area.level | HD for-01.jpg | for-01-no.jpg |

The following three dictionary tables are used to store instances of the previously defined objects: Certificate Instance, Badge Instance and Action Instance.

**Certificate Instance**

Is the table in which all the certificates acquired by the users are stored, along with the date and time of acquisition, and it is used every time a user gains a certificate of knowledge. Table 8.4 shows a possible certificate instance within the system.

Table 8.4: Information model of Certificate Instance

| OID | Date | Certificate Type OID | Community User OID |
|---|---|---|---|
| 1 | 2013-02-03 16:00:00 | 10 | 4516 |

**Action Instance**

Is the table that stores all the actions performed by the users, along with date and time values; every time an event happens a new row is inserted in the Action Instance table. Two attributes deserve a careful explanation: *Tags* identify a set of meaningful words to associate to the performed action and *Link* relates to the hyper-textual link of the web page where the action occurs. Table 8.5 shows a possible action instance within the system.

Table 8.5: Information model of Action Instance

| OID | Date | Score | Action area | Reputation | Participation | Name | Description | Tag | Link | Action type OID | Community User OID |
|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | 2013-02-03 16:00 | 400 | Forum | False | True | Answer to a post | How do I compare strings... | Java,String | 345 | 33 | 2341 |

**Badge Instance**

Is the table in which are stored all the badges assigned in the community with date and time of acquisition; it is used every time a user gains a badge thanks to the points acquired inside the community.
The score attribute is registered from the current status of the player and not imported from the Badge Type table, in this way it is possible to keep track of contingent score points variation during the community evolution. Table 8.6 shows a possible badge instance within the system.

Table 8.6: Information model of Badge Instance

| OID | Date | Score | Badge Type OID | Community User OID |
|---|---|---|---|---|
| 3 | 2013-02-03 16:00:00 | 400 | 1 | 2341 |

### 8.4.3 Integration

The gamified community has been created to be the main core component of all Webratio online relationship with users and customers. The centrality is a key point of the platform since all the existing modules are put in relationship with this new "heart" .

To integrate the new platform within the legacy system, we have developed a set of "bridge applications"(also generated with WebRatio, to smoothen the integration), to create a connection between the community and each individual module.

These bridge applications are developed in a way that allows the systems to remain autonomous and independent, introducing modularity features that could allow future improvement and additions in the case in which business objectives will change.

For each of the WebRatio modules, the bridge application is able to detect when a new relevant event (for the purposes of Gamification action) happens and, as a response, it alerts the Gamification core of the community that handles the occurrences according to the application rules.

The bridges are able to send notifications thanks to an automatic trigger that updates the tables every time a meaningful action is performed. In order to interact with the "bridges", the gamified community provides a set of Web Services that can be invoked by all the modules to which they want to send some actions.

The Web Services are software system designed to support inter-operable machine-to-machine interaction over a network and they have been developed with WebRatio by complying with the SOAP protocol.

SOAP, originally defined as Simple Object Access Protocol, is a protocol specification for exchanging structured information among Web Services in computer networks.

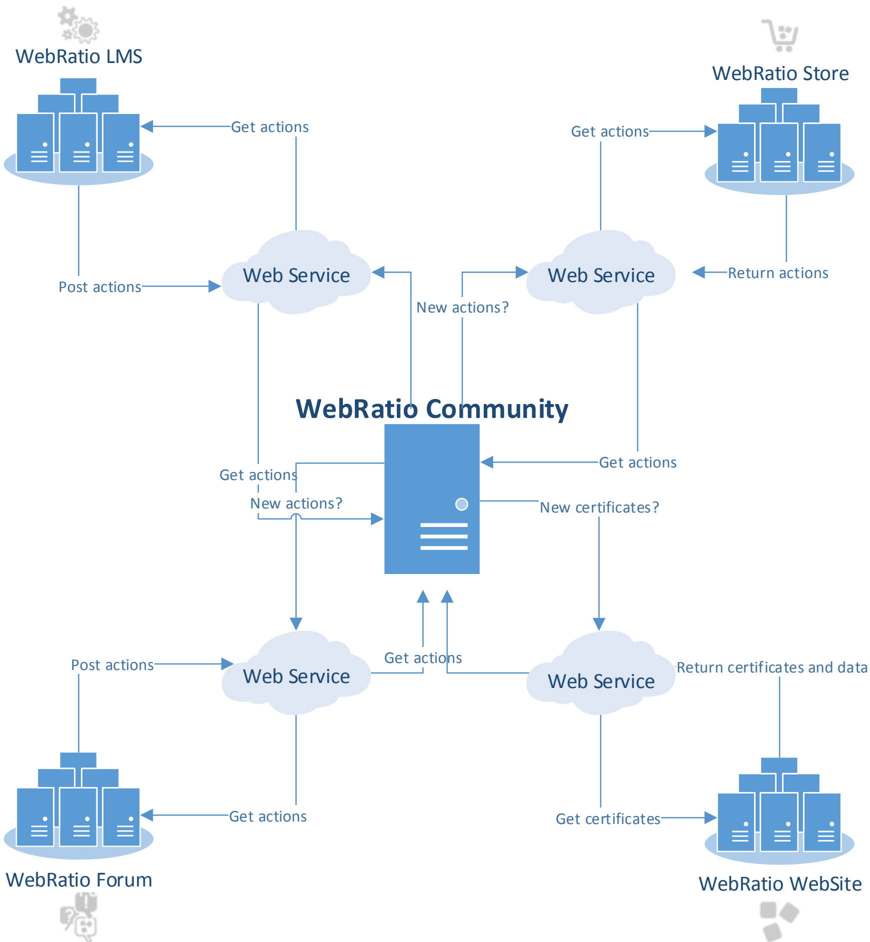



Figure 8.8: Integration of WebRatio Community with WebRatio's modules

Figure 8.8 shows the architecture of the integrated gamified community: all the modules communicate with the Webratio Community of the application and, if needed, the community is able to send messages to the modules; the interactions are ruled as follows:

- For the Store: the community periodically controls if there is a new user, who downloads, uploads, votes or reviews components and with these data the application can assign points to the user that has accomplished those operations in the store area.
- For the Forum: the procedure is the opposite of the previous scenario: it is the Forum itself that advise the community that something happened. The Forum invokes the Web Service of the community by sending the information related to the action executed (create post, answer, vote up, subscription...).
- For the Learning Management System: as for the Forum, the LMS invokes directly the community by providing all the information regarding the activities performed by the user.
- For the Web Site: the website calls the Web Service of the community that assigns points every time a user logs in, updates her data or registers to the platform. The community takes advantage of the "bridge application"that periodically scans the website database to identify if there are users who have acquired new WebRatio Certifications or if there are new users who have activated WebRatio's licenses.

Before the deployment the "bridge applications"need to perform an initial import of all the previous data belonging to each module, in this way all the historical data are preserved and made available to the community.

### 8.4.4 Sequence diagram

To explain in a better way how the Web Services and the bridge applications are related, we describe a sequence diagrams that represent a typical instance. The diagram in Figure 8.9 depicts the situation where the WebRatio module (LMS, Forum, Site) itself is able to register the user action and consequently call the community to alert it.
The user executes an action in one of the portals which are able to inquire the community. For instance, she could read an article about Webratio knowledge, she could create a new Forum post or, in general, perform any of the actions available in the LMS, Forum and WebRatio Site (with the exception of the

certificates acquisition).
When the user has executed the action, the module calls directly the Web Service of the community, that receives the action with all the necessary data (action name, descriptions, user nickname, area of the portal, date, time, tags, and link).

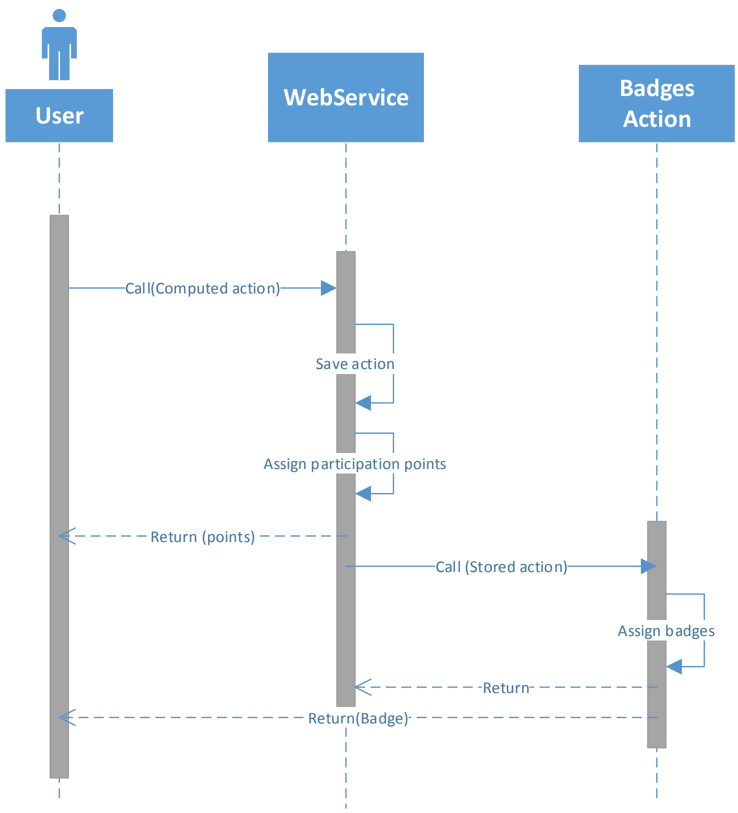



Figure 8.9: Sequence diagram 1 of WebRatio Community

From this moment the community is in charge of computing the points and the possible reputation influence which will be assigned to the user. After the points and credits are assigned, the gamification logic in charge of assigning badges is called.

The community checks if the user has acquired a sufficient number of points to acquire a badge, and, if it is the case, the user will receive one or more badges to certify her success.

In the opposite case, the second diagram (Figure 8.10 ) depicts the situation in which a bridge application has to actively look for new actions to send to the community Web Service. The service inquires every three minutes the databases of the WebRatio modules (Store, Certificates) and parses the data. If it finds out that new actions have been executed, it prepares the data in the right format and sends them at the community Web Service. From this point onwards the performed activities are the same one of the previous sequence diagram: first the community saves the actions and assigns the relative participation points, reputation and credits; afterwards the gamification logic controls whether the user has enough points to acquire a badge and, in such a case, it assigns it to her.

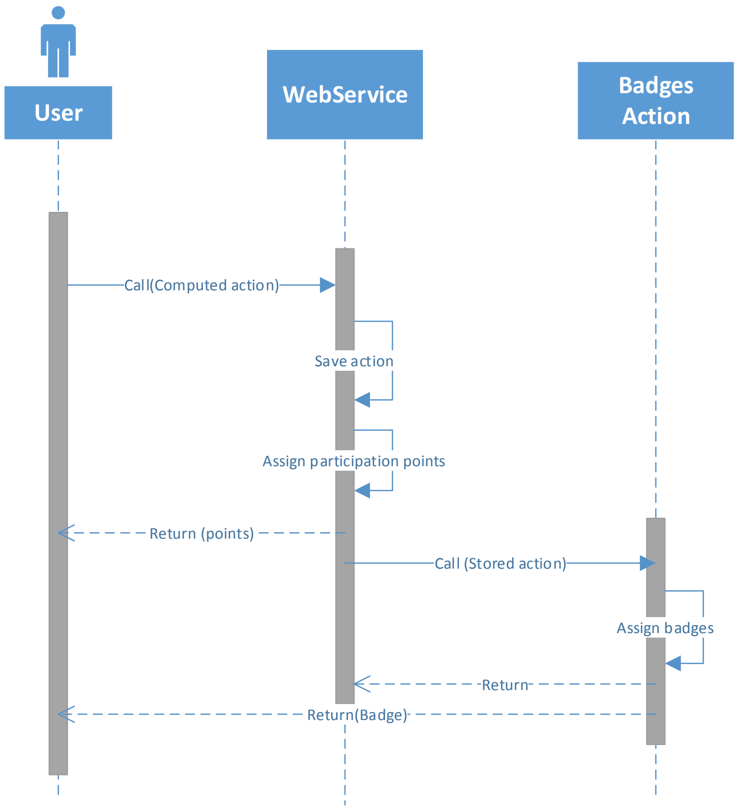



Figure 8.10: Sequence diagram 2 of WebRatio Community

In both the cases, some actions sent to the community are not stored because the gamification logic considers them not valid. For instance, the community assigns participation point for every user's login if and only if each login has been registered at least one hour after the previous one.

## 8.5 Results Evaluation

After the realization of all the phases of design and development of the product and before launching the community into production and make it

accessible to customers, the company has decided to run a test phase of the application.
This decision, as well as allowing a check regarding the stability and quality of the developed software, made possible a first control on the effectiveness of the introduction of the new gamification mechanics; this has allowed also an initial comparison between the results obtained respect to the business goals of WebRatio it has been performed.
WebRatio by following the best practices of software testing decided, along with the developing team, to perform the Alpha-Beta testing, in order to assure the quality of the product before releasing it to the general public. The alpha phase of the release life cycle is the first phase to begin software testing. In this phase, developers generally test the software using white box techniques. A White box technique is a method for testing software that verifies internal structures or execution of an application, as opposed to its functionality. Additional validation is then performed using black box or gray box techniques, by another testing team. Black box testing is a method of software testing that examines the functionalities of an application (e.g. what the software does) without peering into its internal structures. Moving to black box testing inside the organization is known as alpha release. Alpha software can be unstable and could cause crashes or data loss. The alpha phase usually ends with a feature freeze, indicating that no more features will be added to the software. At this time, the software is said to be feature complete.
The Beta phase generally begins when the software is feature complete. Software in beta phase will generally have many more bugs in it than completed software, as well as speed/performance issues and may still cause crashes or data loss. The focus of beta testing is reducing impacts to users, often incorporating usability testing. The process of delivering a beta version to the users is called beta release and this is typically the first time that the software is available outside of the organization that developed it.
The users of a beta version are called beta testers. They are usually customers or prospective customers of the organization that develops the software, willing to test the software without charge, often receiving the final software free of charge or for a reduced price. Beta version software is often useful for demonstrations and previews within an organization and to prospective

customers.
Developers release either a closed beta or an open beta; closed beta versions are released to a restricted group of individuals for a user test by invitation, while open beta testers are from a larger group, or anyone interested.
Beta testing comes after alpha testing and can be considered a form of external user acceptance testing. Releases of the software, known as beta versions, are distributed to a limited audience outside of the programming team. The software is released to groups of people so that further testing can ensure the product has few faults or bugs. Sometimes, beta versions are made available to the open public to increase the feedback field to a maximal number of future users. During the Alpha testing, the community has been tried by all members of the development team, analyzing bugs and problems. In this phase was deemed unnecessary to collect data about gamification aspects because the number of users was small.
Once this first phase was completed, the software was stabilized, and all the main functions have been finalized. So after the Alpha testing, the Beta testing could begin. WebRatio opted for a closed test in fact it decided to open the test to all employees of each business area and some outside consultants (experts on graphics, usability, accessibility and game mechanics).
Although the test was not open to a very large number of people, it has allowed the collection of several data which have made possible some important considerations and comparisons through the calculation of mathematical metrics.
The community, by its nature, is designed to collect a lot of data about user behavior, while not all the existing portals stored all the events related to the customer. For this reason, some comparisons were not possible due to the lack of data coming from the past.
Through a set of database queries, it has been possible to retrieve several information about the past user activities which were then used as comparison.
Goal of the experimental setting is to understand if business goals clearly stated by means of gamification mechanics, in the form of badges and participation points for user comparison through leaderboards, could produce a positive gain. Being able to compare one own participation level with others and having a clear indication of what contributes to the participation level requested by the company, improves the amount of activities done on the

community platform that has been created. To verify the achievement of corporate objectives, before proceeding with the investigation, a number of hypotheses have been drawn up.
Following the success certified by the results on this preliminary phase, the Gamification Platform has been released to the general public and has become one of the assets that the company is selling as a service, as part of their tools.
The following table lists the various hypotheses that we analyze in detail later, the second parameter identifies whether as a result of the calculations, the hypothesis benefited from the Gamification.

Table 8.7: Hypothesis table

| Hypothesis | Positive results |
|---|---|
| **Hypothesis 1**: Gamification mechanics can improve self-powered customer support through the Forum | |
| HP 1.1 Participation points increase the use of the community forum to seek for aid (community driven help) | YES |
| HP 1.2 Participation points increase the will of the user to help other users in the community | YES |
| HP 1.3 Participation points improves the quality of the answers provided in the Forum | PARTIALLY |
| HP 1.4 The use of badges can induce users to post more meaningful answers | PARTIALLY |
| **Hypothesis 2**: Gamification mechanics can improve the ability of the company to identify the customers that use their product and increase the company image by showing more and more companies using WebRatio | |
| HP 2.1 Participation points can increase the number of users that register within the system | YES |
| HP 2.2 Participation points can increase the details provided by the users on themselves (photo, social media accounts, biography) | NO DATA |
| HP 2.3 Participation points can increase the involvement of existing enterprise users and identify them (customer/partner serial registration) | NO DATA |
| **Hypothesis 3**: Gamification mechanics can increase user retention | |
| HP 3.1 Participation points may induce users to login more often | NO |

| | |
|---|---|
| **Hypothesis 4**: Gamification mechanics can induce users to produce and select high quality extensions for WebRatio | |
| HP 4.1 Improved feedback for the components present in the Store | YES |
| HP 4.2 Increased the number of components available | NO |
| HP 4.3 Increased the awareness of the users with respect to the extensibility of the platform | YES |
| **Hypothesis 5**: Gamification mechanics can induce users to participate and read more tutorial article about the use of WebRatio | |
| HP 5.1 Increased the number of WebRatio tutorial article read by the users | YES |

After the data retrieval and the choice of the hypothesis to test, we studied how to proceed with the analysis.
Having different hypotheses to be compared for the same scenarios (the number of different actions performed by users in the same month, before and after the introduction of game mechanisms), we decided to use the Student's T-Tests, since we have only two groups to contrast, and we used it as a statistical tool to infer differences between small samples based on the mean and standard deviation.

For all our experiments a level of significance of $\alpha = 0.05$ (5%) has been chosen; since we want to verify that the gamification and all the game mechanisms have introduced some benefits, we will use a one-tailed test because we need to prove the hypothesis only in one direction, that is if the introduction of gamification components has brought benefits based on the business objectives to be reached. We compare the results that we have collected during a period of 6 months, from October to March of two consecutive years; in the first year the collected results are related to an environment that was not gamified while on the second year the collected data derived from the full gamified community, already in place with target users. For a one-tailed test with 181 degrees of freedom (given the 6 months period, 182 days), the t-critical value for $\alpha$ =0.05 is :

$$tcrit = 1.645$$

This value will be used to all the practical experiments illustrated in the next chapter, comparing the t-critical value with the t-value of the observation and if:

$$\text{t-obs} > \text{t-crit}$$

this means that our calculated value is greater than critical one, so the value of the experiment falls inside the accepting area and we can reject the null hypothesis and accept the alternative hypothesis; in other words, the results of the experiments are not due to chance.
If $t - obs$ is smaller than the $t - critical$ value it would be impossible to reject the null hypothesis, because the calculated value falls inside the middle area of the student distribution; in such case the results could be due to chance or are congruent values with respect to the population mean.
Each hypothesis that has been reported in the previous table has been considered individually to check whether the assumptions made during the design phase were feasible, by detailing the data collected from the experiments and the calculated metrics. A hypothesis can have more actions associated to it so to clarify the procedure, before performing the testing phase, the summary table with the data and the graph is reported; since the data has been obtained over a period of 182 days, we cannot report each data for space and clarity reasons and the daily results have been represented as monthly aggregates in the graphs.

### 8.5.1 Self Powered Customer Support

- Hypothesis 1: Gamification mechanics can improve self-powered customer support through the Forum.
  - HP 1.1 Participation points increase the use of the community forum to seek for aid (community driven help).
    *Post a question analysis*

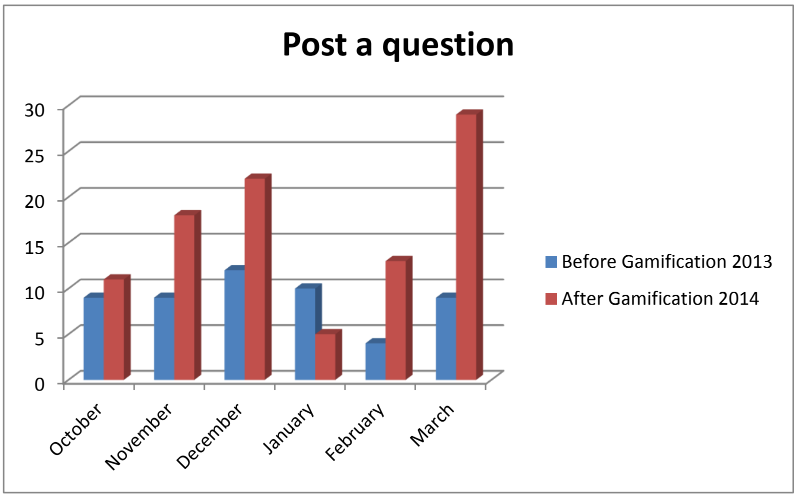



Figure 8.11: Post a question graph

In the Forum, the company would like to improve the participation of its users, making the WebRatio tool more clear.
If the forum is well populated, a customer who uses WebRatio will be confident in asking his questions or problems because he will perceive that the Forum is a place where it is possible to learn and get support for free in a quick and easy way.
To prove that gamification has led the users into a greater desire to publish questions, thus making the Forum attended and full of active "life" regarding WebRatio, we want to compare the amount of daily questions created from October to March 2013 against October to March 2014.
The data have been collected daily and thirty days after several metrics were calculated and reported in Table 8.8; Table 8.9 reports the t-Test results for our hypothesis. In detail were calculated: the total sum of questions published, the monthly average, variance, standard deviation and asymmetry of the data collected.
To verify the effects caused by gamification, we performed a Dependent T-Test for Paired Samples. As illustrated in the previous chapter, the dependent t-test for paired samples is used when we want to compare the samples of two groups which are paired. This implies that each individual observation of one sample has a unique corresponding member in the other sample. In our case, for each

month, each day has a unique matching to the one in the following year.
From the T-Test theorem, we have to decide which is the null hypothesis ($H_0$), and which is the alternative hypothesis ($H_1$). Since we want to prove that the gamification has introduced an increase in the number of questions created in the Forum, we chose as null hypothesis:

$$H_0 : \mu = 0; \text{differences in number of questions are due to chance.}$$

and as alternative hypothesis:

$$H_1 : \mu > 0; \text{the induction of gamification bring a benefit.}$$

| Post a question | Before Gamification | After Gamification | Difference |
|---|---|---|---|
| Sum | 53 | 98 | 45 |
| Mean | 0,291 | 0,538 | 0,247 |
| Variance | 0,351 | 0,780 | 0,429 |
| Standard deviation | 0,592 | 0,883 | 0,291 |
| Asymmetry | 2,388 | 2,070 | 0,317 |

Table 8.8: Post a question data

Table 8.9: Post a question t-Test results

| | |
|---|---|
| Pearson Correlation | -0,037 |
| Hypothesized Mean Difference | 0 |
| df | 181 |
| t Stat | 3,083 |
| P(T<=t) one-tail | 0,001184 |
| t Critical one-tail | 1,6533 |

Since tobs > tcrit our calculated value is larger than the tabled critical value at $\alpha$=0.05, so we reject the null hypothesis and accept the alternative hypothesis, namely, that the difference in number of question published is likely the result of the gamification effects and not the result of chance variation: the game mechanics that have been introduced have encouraged users to take an active part in the

Forum life. From the p-value is also possbile to infer that the test is valid even for $\alpha = 0.01$, a strong indication that the gamification approach was the reason behind such an improvement.
The benefits for the company are clear: a lively community indicates a product that creates interest among the users and well supported also.

– HP 1.2 Participation points increase the will of the user to help other users in the community.
*Post an answer analysis*

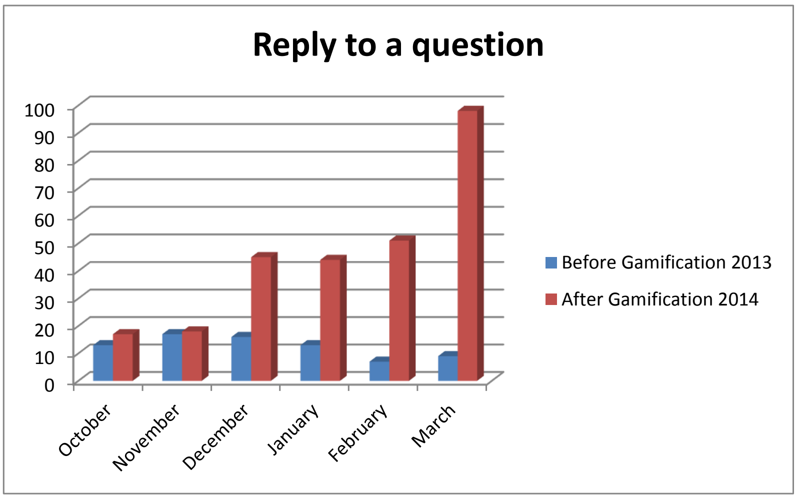



Figure 8.12: Post an answer graph

| Post an answer | Before Gamification | After Gamification | Difference |
|---|---|---|---|
| Sum | 75 | 273 | 198 |
| Mean | 0,412 | 1,472 | 1,06 |
| Variance | 0,53 | 3,941 | 1,256 |
| Standard Deviation | 0,728 | 1,985 | 1,256 |
| Asymmetry | 2,042 | 2,315 | 0,272 |

Table 8.10: Post an answer data

Table 8.11: Post an answer t-Test results

| | |
|---|---|
| Pearson Correlation | -0,104 |
| Hypothesized Mean Difference | 0 |
| df | 181 |
| t Stat | 6,546 |
| P(T<=t) one-tail | $2,93 * 10^{-10}$ |
| t Critical one-tail | 1,6533 |

Webratio created the community in an attempt to raise the number of feedbacks and the collaboration among the users, in this case in the Forum.
According to the gamification theories, if a user is hooked within the game mechanisms, she would be prone to collect points to win awards; this encourages her to participate more, providing her skills and knowledge to others.
Since we want to prove that the community has driven the people to actively contribute, in this case through the number of replies in the Forum, we chose as null hypothesis:

$$H_0 : \mu = 0; \text{any differences in number of questions is due to chance.}$$

and as alternative hypothesis:

$$H_1 : \mu > 0; \text{the induction of gamification brings a benefit.}$$

Since tobs > tcrit our calculated value is larger than the tabled critical value at $\alpha$=0.05, so we reject the null hypothesis and accept the alternative hypothesis, namely, that the difference in number of replies published in the Forum is likely the result of the gamification effects and not the result of chance variation. Tobs is significatively higher than tcrit also, which means that our assumptions holds even for $\alpha = 0.01$: the introduction of gamification mechanics has actively boosted the self-sustained support through the Forum thanks to an increased participation of its users.

– HP 1.3 Participation points improves the quality of the answers provided in the Forum.

*Forum Upvote analysis*

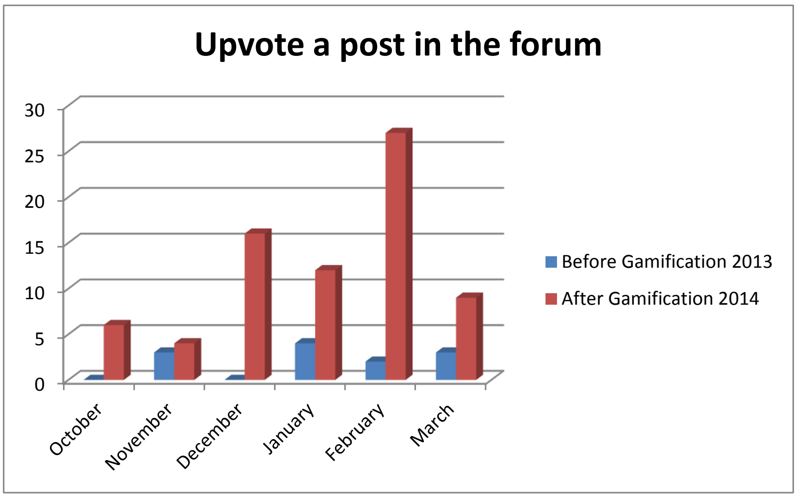



Figure 8.13: Forum Upvote graph

| Forum Upvote | Before Gamification | After Gamification | Difference |
|---|---|---|---|
| Sum | 12 | 74 | 62 |
| Mean | 0,066 | 0,406 | 0,34 |
| Variance | 0,084 | 0,706 | 0,622 |
| Standard deviation | 0,289 | 0,84 | 0,55 |
| Asymmetry | 4,80 | 2,5 | 2,30 |

Table 8.12: Forum Upvote data

Table 8.13: Forum Upvote t-Test results

| | |
|---|---|
| Pearson Correlation | -0,042 |
| Hypothesized Mean Difference | 0 |
| df | 181 |
| t Stat | 5,101 |
| P(T<=t) one-tail | $4,22 * 10^{-7}$ |
| t Critical one-tail | 1,6533 |

The community has the goal help the company to spread the word about WebRatio to many potential customers, but above all to

improve the quantity and quality of the content published on its platforms.
For this reason, it is believed that the number of upvotes, applicable to both questions and answers in the Forum is an important measure to discover "Quality Content" both qualitatively and quantitatively. Since we want to prove that the community has improved the quality in all the activities which the users perform inside the WebRatio platforms, we chose as null hypothesis:

$$H_0 : \mu = 0 \text{any differences in number of upvotes is due to chance.}$$

and as alternative hypothesis:

$$H_1 : \mu > 0; \text{game mechanics increased the number of upvotes.}$$

Since tobs > tcrit our calculated value is bigger than the corresponding critical value at $\alpha$=0.05 and also at $\alpha$=0.01, we can reject the null hypothesis.
Upvoting was a feature seldom used before the introduction of the gamified forum, probably because it was not that well known and also because it requires deep understanding of all the possible answers that have been submitted in order to choose the best one. Most of the users seemed to be lazy or reluctant to do so; gaining points for such an action created a meaningful boost in the usage of this feature, improving the overall quality of the submitted content by highlighting valuable contributions.

*Answer approved analysis*

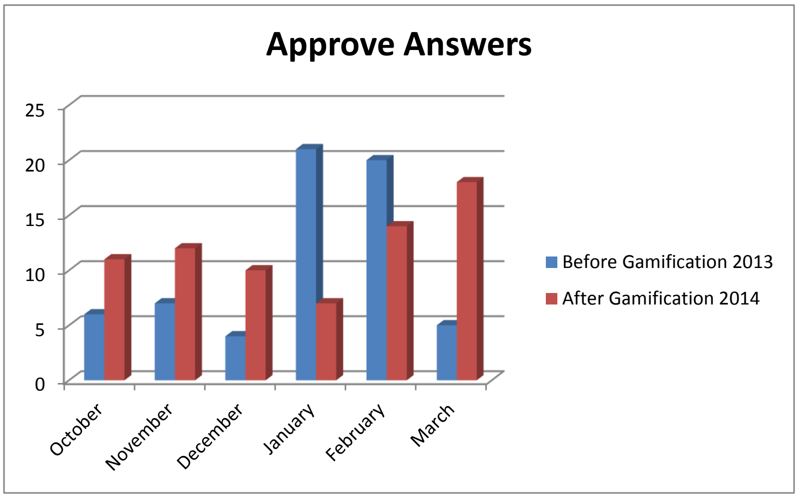



Figure 8.14: Forum answer approved graph

| Forum answer approved | Before Gamification | After Gamification | Difference |
|---|---|---|---|
| Sum | 63 | 72 | 9 |
| Mean | 0,335 | 0,395 | -0,06 |
| Variance | 0,599 | 0,483 | 0,116 |
| Standard deviation | 0,774 | 0,695 | 0,079 |
| Asymmetry | 2,802 | 1,875 | 0,927 |

Table 8.14: Forum Answers approved data

Table 8.15: Forum Answers approved t-Test results

| | |
|---|---|
| Pearson Correlation | -0,032 |
| Hypothesized Mean Difference | 0 |
| df | 181 |
| t Stat | 0,77 |
| P(T<=t) one-tail | 0,22 |
| t Critical one-tail | 1,6533 |

When an user sees one of her responses approved on the Forum, she gains participation and especially reputation points; answer approval is, for this reason, one of the most important actions to discover meaningful contributors in the community.

Unfortunately, the data obtained for this particular case are not positive, as we can see from the $p - value$ of 0,22. As shown in Figure 8.14, there is no clear indication that the users are willing to approve answers due to the introduction of gamification mechanics and on the contrary, on two months the non gamified platform outclassed all the other months even when the gamification platform was already introduced. A possible explanation for the data may be related to the fact that users, realizing that approving answers brings points to their competitors, prefer to not accept them tactically (no given points to the other users). If this phenomenon was confirmed, the system administrator should act by accepting correct answers and the designer in charge of the gamification platform should modify the mechanics in order to cope with the situation.

– HP 1.4 The use of badges can induce users to post more meaningful answers.

*Best Users Posted Answers*

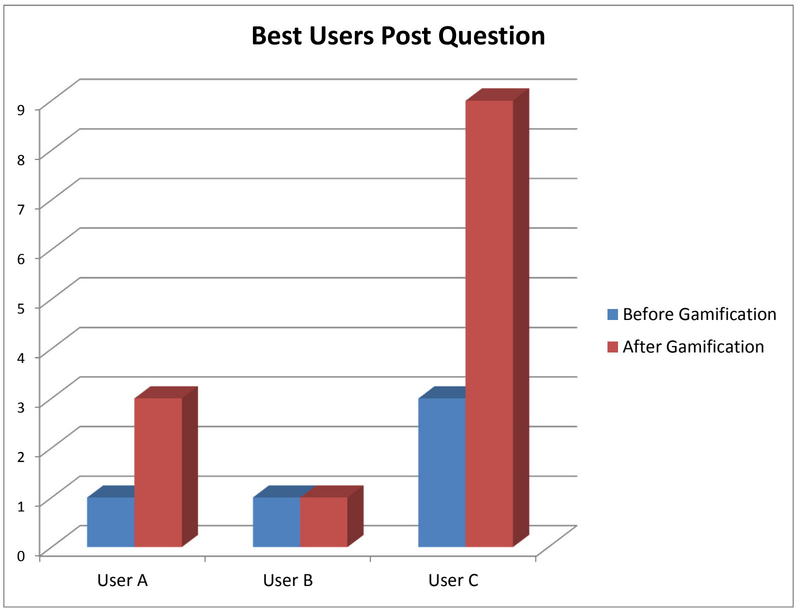



Figure 8.15: Best User Forum Questions graph

| User A | Before Gamification | After Gamification | Difference |
|---|---|---|---|
| Sum | 1 | 3 | 2 |
| Mean | 0.033 | 0.1 | 0.066 |
| Variance | 0.033 | 0.093 | 0.133 |
| Standard deviation | 0.182 | 0.305 | 0.365 |
| Asymmetry | 5.477 | 2.809 | 0.924 |

Table 8.16: Best User Forum Questions User A information table

| User B | Before Gamification | After Gamification | Difference |
|---|---|---|---|
| Sum | 1 | 1 | 0 |
| Mean | 0.033 | 0.033 | 0 |
| Variance | 0.033 | 0.033 | 0.068 |
| Standard deviation | 0.182 | 0.182 | 0.262 |
| Asymmetry | 5.477 | 5.477 | 0 |

Table 8.17: Best User Forum Questions User B information table

| User C | Before Gamification | After Gamification | Difference |
|---|---|---|---|
| Sum | 3 | 9 | 6 |
| Mean | 0.1 | 0.3 | 0.2 |
| Variance | 0.093 | 0.355 | 0.51 |
| Standard deviation | 0.305 | 0.595 | 0.714 |
| Asymmetry | 0 | 1.906 | 0.899 |

Table 8.18: Best User Forum Questions User C information table

The objective of these tests is to study the behavior of users which were already present in the WebRatio world.
We study how the mechanism of rewards based on the badges system has amplified the participatory attitude of the users who usually perform more actions in the Forum area (that is, posting and approving answers). In the tables above are illustrated the data collected during the Beta testing phase for the three users who have performed more actions in the Forum.
With respect to the previous data analysis, we have available

data just for one month, since the platform was not previously able to keep track also of the actions performed by each single users, feature that was introduced afterwards even if ahead of the integration with the gamification platform. We have three users, User A, B and C; User B has performed the same number of actions both prior to the introduction of the gamification platform and during the beta phase, thus she has not been affected from the presence of badges.

For the other two users, who have a positive difference between the number of actions performed in the two consecutive years, we can proceed with the t-test.

Since we want to study if there was a positive change of user behavior after that the introduction in the community of a badge system, we choose as null hypothesis:

$$H_0 : \mu = 0; \text{behavior change is due to natural trends}$$

and as alternative hypothesis:

$$H_1 : \mu > 0; \text{increased participation due to gamified elements}$$

Table 8.19: Forum Answers User A t-Test results

| | |
|---|---|
| Pearson Correlation | -0,061 |
| Hypothesized Mean Difference | 0 |
| df | 29 |
| t Stat | 1 |
| P(T<=t) one-tail | 0,162 |
| t Critical one-tail | 1,699 |

Table 8.20: Forum Answers User B t-Test results

| | |
|---|---|
| Pearson Correlation | -0,034 |
| Hypothesized Mean Difference | 0 |
| df | 29 |
| t Stat | 0 |
| P(T<=t) one-tail | 0,5 |
| t Critical one-tail | 1,699 |

Table 8.21: Forum Answers User C t-Test results

| | |
|---|---|
| Pearson Correlation | 0,392 |
| Hypothesized Mean Difference | 0 |
| df | 29 |
| t Stat | 1,98 |
| P(T<=t) one-tail | 0,028 |
| t Critical one-tail | 1,699 |

For the User A $t - obs < t - crit$ for $\alpha$=0.05, so we can not reject the null hypothesis;for User C $t - obs > t - crit$ so we can reject the null hypothesis and declare that there is $\alpha$=0.05 statistically significance that the badges system introduce benefits. Overall, we would have expected that the more active users wanted to maintain their reputation in the system and show it via the new gamification means used to certify their status. This seems not to be the case, probably also because a old known users will be such even without digital recognition; it has to be noted though that we didn't had enough data to make an in-depth analysis for this specific behavior.

### 8.5.2 Customers Identification

- Hypothesis 2: Gamification mechanics can improve the ability of the company to identify the customers that use their product and increase the company image by showing more and more companies using WebRatio,

– HP 2,1 Participation points can increase the number of users that register within the system.

*User Registration analysis*

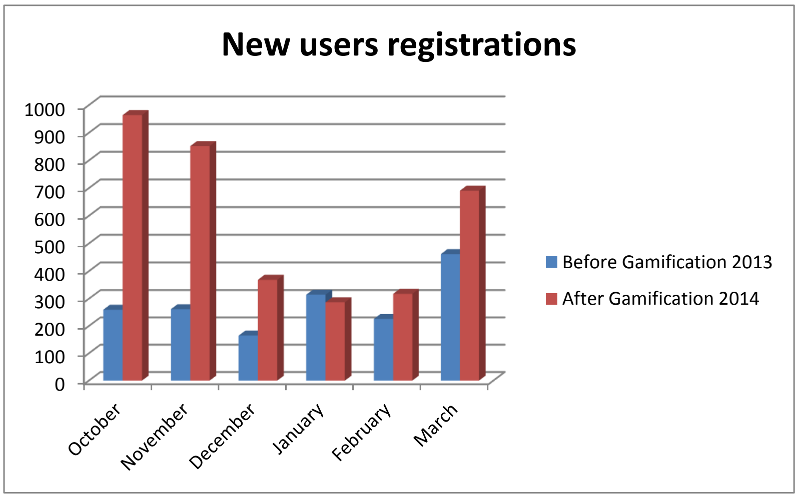



Figure 8.16: User Registration graph

| User Registration | Before Gamification | After Gamification | Difference |
|---|---|---|---|
| Sum | 1690 | 3481 | 1791 |
| Mean | 9,15 | 18,95 | 9,80 |
| Variance | 116,12 | 421,87 | 305,75 |
| Standard deviation | 10,77 | 20,53 | 9,76 |
| Asymmetry | 1,36 | 1,72 | 0,36 |

Table 8.22: User Registration data

Table 8.23: User Registration t-Test results

| | |
|---|---|
| Pearson Correlation | 0,0684 |
| Hypothesized Mean Difference | 0 |
| df | 181 |
| t Stat | 5,8622 |
| P(T<=t) one-tail | $1,060 * 10^{-8}$ |
| t Critical one-tail | 1,6533 |

The number of registered within the WebRatio portal is an important feature, because one of the main goals of the company is to increase the number of its clients, increasing the popularity of the product among industry professionals,
Table 8.22 illustrates the total and average monthly number of registered users in the platform prior and afterwards the integration with the gamification components, As we can see immediately there is a consistent boost in the number of registrations, that have increased by 105%,
Since we want to prove that gamification has transformed all the WebRatio portals, making them more attractive to new users, we chose as null hypothesis:

$$H_0 : \mu = 0 \text{nothing has changed with respect to the past,}$$

and as alternative hypothesis:

$$H_1 : \mu > 0; \text{the gamification strategy has attracted more users,}$$

Since $t-obs > t-crit$ our calculated value is larger than the critical value at $\alpha$=0,05 and $\alpha$=0,01, so we reject the null hypothesis and accept the alternative hypothesis, namely, that the difference in number of registration is likely the result of the gamification effects and not the result of chance variation,
This real world result reached an important goal for the company but also for us: many works in literature have described the benefits of introducing gamification elements in an enterprise scenario, without providing data or statistical analysis supporting this thesis, Thanks to this experiment, it was possible to prove that the engaging feature of the new portal was able to attract an unexpected amount of users, doubling the registrations with respect to the previous year,

- HP 2,2 Participation points can increase the details provided by the users on themselves (photo, social media accounts, biography),

– HP 2,3 Participation points can increase the involvement of existing enterprise users and identify them (customer/partner serial registration), *User Information Data analysis*

| Action | Before Gamification | After Gamification |
|---|---|---|
| n,Photo | No numerable | 877 |
| n,Bio | No numerable | 1865 |
| n,Twitter | No numerable | 1865 |
| n,Linkedin | No numerable | 1846 |
| n,Website | No numerable | 1865 |
| n,Newsletter | No numerable | 1862 |
| n,Company/University | No numerable | 1862 |
| n,Clients/Partners | No numerable | 183 |
| Tot | No numerable | 12728 |

Table 8.24: Updating Information data

Unfortunately, all kind of information related to the data that the user could set in his account, before the creation of the community, were not tracked by the system so it is not possible to know when an user had set or change his profile data,
For this reason we have not a previous history with which to compare the number of actions which occurred after the introduction of gamification elements,
We can note that the figures provided attest that, according also to the data provided in Table 8.22, most of the new users and part of the old users has modified in some way or another their profile, The old users in particular may have updated and enriched their personal information to receive points which the community assigns for these types of action,
A key to understanding of the need of completing its own profile, for both newcomers and already registered users is undoubtedly the desire to increase their own scores to climb the ranks of the community,

### 8.5.3 User Retention

- Hypothesis 3: Gamification mechanics can increase user retention,
    - HP 3,1 Participation points may induce users to login more often.

*Forum Login analysis*

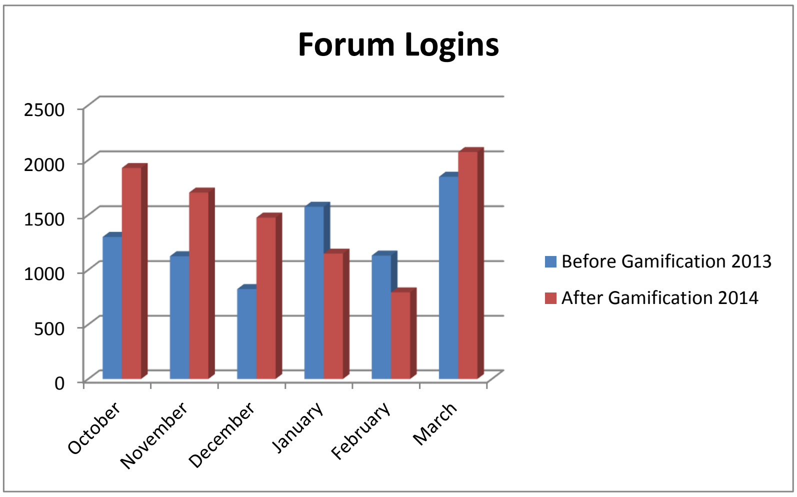



Figure 8.17: Forum Logins graph

| Forum Logins | Before Gamification | After Gamification | Difference |
|---|---|---|---|
| Sum | 7802 | 9124 | 1322 |
| Mean | 42,74 | 49,95 | 7,21 |
| Variance | 2864,79 | 1935,94 | 928,85 |
| Standard deviation | 53,52 | 43,99 | 9,53 |
| Asymmetry | 2,24 | 2,62 | -0,38 |

Table 8.25: Forum logins data

Table 8.26: Forum Logins t-Test results

| | |
|---|---|
| Pearson Correlation | -0,0489 |
| Hypothesized Mean Difference | 0 |
| df | 181 |
| t Stat | 1,372 |
| P(T<=t) one-tail | 0,0858 |
| t Critical one-tail | 1,6533 |

One of the key points of the community is to involve users in the WebRatio world, It has the aim of creating a kind of virtuous circle that causes people to be often on the platform, in a consistent manner, To verify that this objective has been achieved, it is necessary to control the number of logins that users have made in the Forum, so as to determine whether, in general they changed their habits and how this trend has changed,

Since the aim is to prove that the participation points assigned by visiting the community, encourage the users to participate more and to increase their visits on the portals, the chosen null hypothesis is:

In this case $t - obs < t - crit$ in this situation we can not reject the null hypothesis ($H_0$) then it is not possible to assert that the increase of number Forum login is statistically significant,

According to the t-test results this small increase could be also due only to a random event, The result is a bit surprising, since we have seen that the amount of registered users has increased, but it let us do an important consideration: since the amount of users logging in within the forum has not changed significantly in the two periods taken into consideration, this means that the introduction of gamification has been successful in promoting active participation of users that were already in the system but had no incentive to act.

*Best User Forum Login analysis*

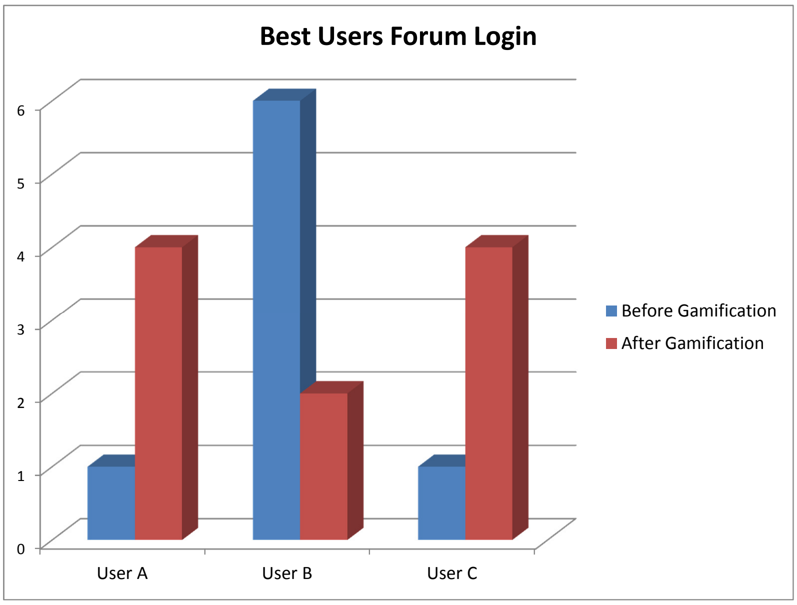



Figure 8.18: Best User Forum Login graph

| User A | Before Gamification | After Gamification | Difference |
|---|---|---|---|
| Sum | 1 | 4 | 3 |
| Daily Mean | 0,033 | 0,133 | 0,1 |
| Weekly Mean | 0,25 | 1 | 0,75 |
| Variance | 0,033 | 0,119 | 0,162 |
| Standard deviation | 0,182 | 0,345 | 0,402 |
| Asymmetry | 5,477 | 2,272 | 0,883 |

Table 8.27: Best User Forum Login User A information table

Table 8.28: User A Forum Logins t-Test results

| | |
|---|---|
| Pearson Correlation | -0,073 |
| Hypothesized Mean Difference | 0 |
| df | 29 |
| t Stat | 1,360 |
| P(T<=t) one-tail | 0,092 |
| t Critical one-tail | 1,699 |

| User B | Before Gamification | After Gamification | Difference |
|---|---|---|---|
| Sum | 6 | 2 | -4 |
| Daily Mean | 0,2 | 0,066 | -0,133 |
| Weekly Mean | 1,5 | 0,5 | -1 |
| Variance | 0,303 | 0,064 | 0,326 |
| Standard deviation | 0,55 | 0,253 | 0,571 |
| Asymmetry | 2,758 | 3,659 | -2,405 |

Table 8.29: Best User Forum Login User B information table

Table 8.30: User B Forum Logins t-Test results

| | |
|---|---|
| Pearson Correlation | 0,148 |
| Hypothesized Mean Difference | 0 |
| df | 29 |
| t Stat | 1,273 |
| P(T<=t) one-tail | 0,105 |
| t Critical one-tail | 1,699 |

| User C | Before Gamification | After Gamification | Difference |
|---|---|---|---|
| Sum | 1 | 4 | 3 |
| Daily Mean | 0,033 | 0,133 | 0,1 |
| Weekly Mean | 0,25 | 1 | 0,75 |
| Variance | 0,033 | 0,119 | 0,093 |
| Standard deviation | 0,182 | 0,345 | 0,305 |
| Asymmetry | 5,477 | 2,272 | 2,809 |

Table 8.31: Best User Forum Login User C information table

Table 8.32: User C Forum Logins t-Test results

| | |
|---|---|
| Pearson Correlation | 0,473 |
| Hypothesized Mean Difference | 0 |
| df | 29 |
| t Stat | 1,795 |
| P(T<=t) one-tail | 0,041 |
| t Critical one-tail | 1,699 |

An interesting analysis which allows to study how the introduction of game mechanisms have changed the behavior of users was made, This research has been made by tracking the number of Forum login of the three users (participants in beta testing) who in the past have carried out the higher number of Forum logins over all other users,
In the tables above are illustrated the data collected during the testing phase for the three users more involved (User A, User B, User C),
With respect to the previous data analysis, once more we have available data just for one month, since the platform was not previously able to keep track also of the actions performed by each single users, feature that was introduced afterwards, even if ahead of the integration with the gamification platform,
We can note that the User B did not increase the number of logins executed in the Forum, but decreased it, thus for User B the t-test is superfluous, as probably the small number of actions performed during the month of beta testing is due to chance, For the other two users, who have a positive difference between the number of logins performed, we can proceed with the t-test,
Since we want to study if there was a positive change of user behavior after that the community has been created, we chose as null hypothesis:

$$H_0 : \mu = 0; \text{behavior change is due to natural trends}$$

and as alternative hypothesis:

$$H_1 : \mu > 0; \text{increased forum attendance due to gamified elements}$$

Just for User C tobs > tcrit at $\alpha$=0,05, so we reject the null hypothesis and accept the alternative hypothesis, namely, that the gamification has increased the number of logins for her,
Once more, we cannot make consideration that are truthful in this case, due to the limited number of data at hand, but we cannot see a

true benefit of introducing gamification to increase the participation of users that were already active on their own.

*Website Login analysis*

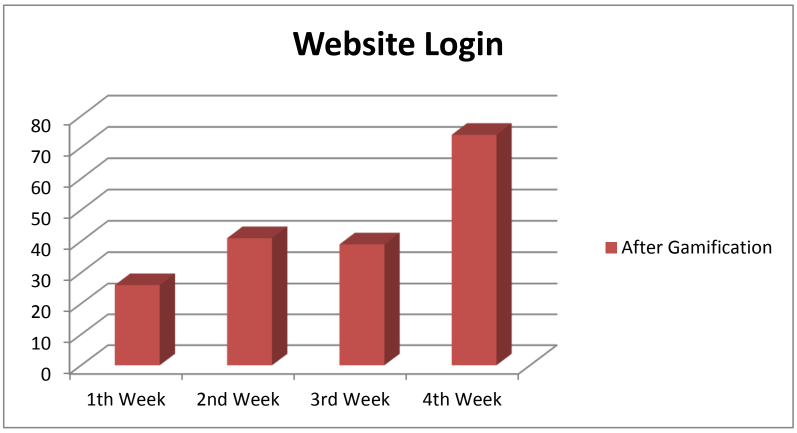



Figure 8.19: Website Logins graph

| Website Login | Before Gamification | After Gamification |
|---|---|---|
| Sum | Not acquired | 550 |
| Mean | Not acquired | 91,66 |
| Variance | Not acquired | 469,06 |
| Standard deviation | Not acquired | 21,65 |
| Asymmetry | Not acquired | -1,09 |

Table 8.33: Website Login data

The number of logins performed in WebRatio website is part of the data that is tracked by the gamification platform, Unfortunately, logins of the users were not recorded in the past, so we can not make a comparison,

### 8.5.4 Component Submission

- Hypothesis 4: Gamification mechanics can induce users to produce and select high quality extensions for WebRatio.

– HP 4.1 Improved feedback for the components present in the Store.

*Rating a component analysis*

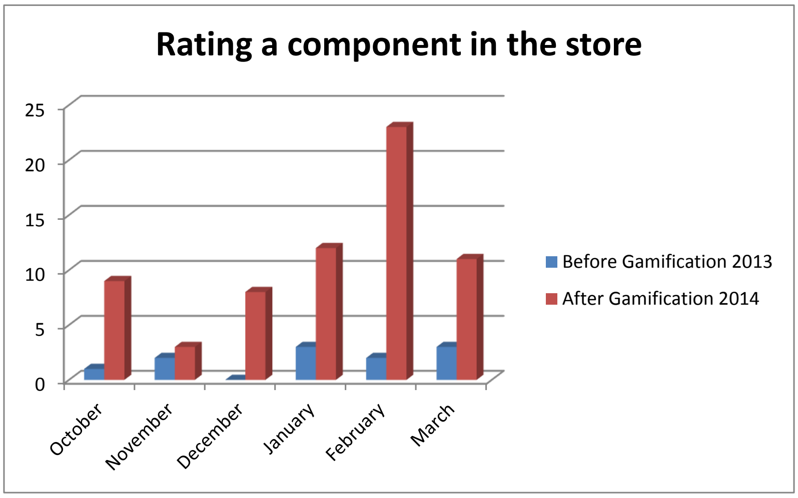



Figure 8.20: Rating a component graph

| Rating a component | Before Gamification | After Gamification | Difference |
|---|---|---|---|
| Sum | 11 | 66 | 55 |
| Mean | 0,060 | 0,36 | 0,30 |
| Variance | 0,079 | 0,519 | 0,44 |
| Standard deviation | 0,281 | 0,720 | 0,439 |
| Asymmetry | 5,103 | 2,549 | 2,554 |

Table 8.34: Rating a component data

Table 8.35: Rating a component t-Test results

| | |
|---|---|
| Pearson Correlation | -0,027 |
| Hypothesized Mean Difference | 0 |
| df | 181 |
| t Stat | 5,22 |
| P(T<=t) one-tail | $2,43 * 10^{-7}$ |
| t Critical one-tail | 1,6533 |

Historically, the rating of a component in the store is an action that users do not practice very often because they tend to download the artifact and, once installed, they do not come back on the portal to leave their rates and comments, unless the component is bad or has problems. This behavior highlights just bad components, leaving the working ones in a neutral limbo of uncertainty.

Promoting rating and review from the users also for components which are used but seldom rated, the gamification platform of the store provide participation points for a submitted content and rating. But has this strategy brought any benefit? To verify it with a t-student test, we choose:

$$H_0 : \mu = 0 \text{any differences in the rating numbers is due to chance.}$$

and as alternative hypothesis:

$$H_1 : \mu > 0; \text{the users' will to rate has increased.}$$

Once more, $t - obs > t - crit$ for $\alpha$=0.05 and $\alpha$=0.01, proving without any doubt that the new strategy has sensibly increased the contributions of the users in terms of feedbacks submitted to the component store.

This is a solid help for the company to scrape garbage software and maintain high quality standard for the entire development platform.

– HP 4.2 Increased the number of components available.

*Published a component analysis*

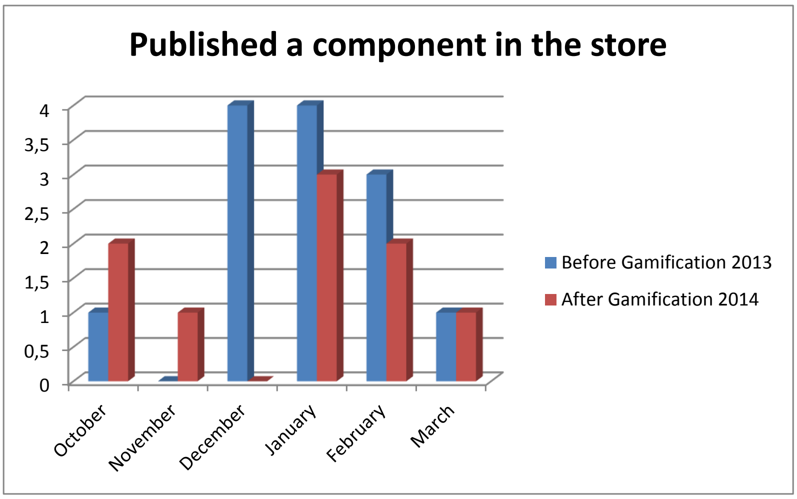



Figure 8.21: Published a component graph

| Published a component | Before Gamification | After Gamification | Difference |
|---|---|---|---|
| Sum | 13 | 9 | 4 |
| Mean | 0,071 | 0,049 | 0,022 |
| Variance | 0,077 | 0,047 | 0,030 |
| Standard deviation | 0,278 | 0,217 | 0,061 |
| Asymmetry | 4,10 | 4,19 | 0,09 |

Table 8.36: Upload a component data

Table 8.37: Published a component t-Test results

| | |
|---|---|
| Pearson Correlation | -0,059 |
| Hypothesized Mean Difference | 0 |
| df | 181 |
| t Stat | 0,815 |
| P(T<=t) one-tail | 0,208 |
| t Critical one-tail | 1,6533 |

An indirect consequence of the diffusion of WebRatio as a commercial product, can be the birth of developers who dedicate their time at developing components that extend and expand the functionality of the tool.
For this reason it is important for the company to track how the creation of new components is evolving, also since Webratio can obtain a remarkable business advantage in having its platform extended for free.
Since we would like to see if the community has introduced more components and extensions with respect to the past because of the new incentives, we chose as null hypothesis:

$$H_0 : \mu = 0; \text{components uploaded increase is due to chance.}$$

and as alternative hypothesis:

$$H_1 : \mu > 0; \text{ gamification increased the components uploaded.}$$

Since $t-obs < t-crit$ for $\alpha$=0.05, we can not reject the null hypothesis. This is, in one sense, reasonable: developing components is a costly and time consuming activity, that can be performed just by few selected programmers. An extrinsic incentive is hardly going to work in this scenario, because the complexity of the activity outclass by far the possible reward got in return.

– HP 4.3 Increased the awareness of the users with respect to the extensibility of the platform.

*Download a component analysis*

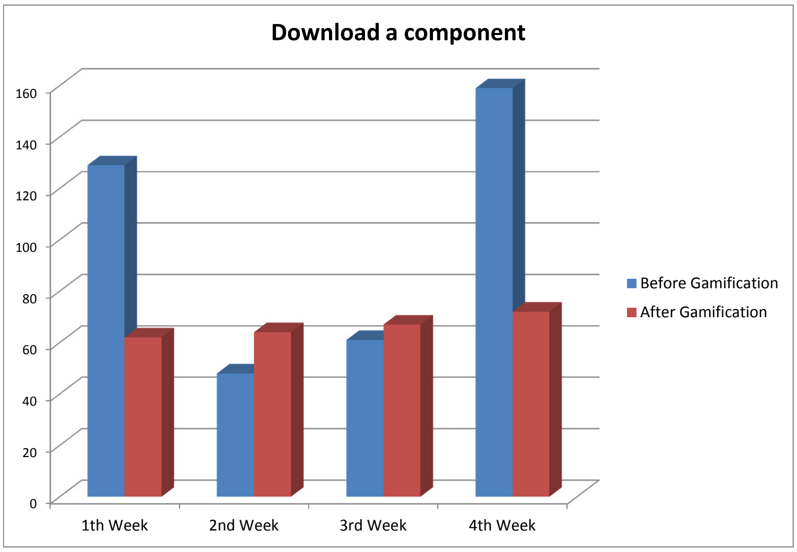



Figure 8.22: Download a component graph

| Download a component | Before Gamification | After Gamification | Difference |
|---|---|---|---|
| Sum | 1597 | 2556 | 959 |
| Mean | 8,632 | 13,813 | 5,181 |
| Variance | 27,637 | 79,246 | 51,609 |
| Standard deviation | 5,257 | 8,902 | 3,645 |
| Asymmetry | 1,151 | 1,513 | 0,362 |

Table 8.38: Download a component data

Table 8.39: Download a component t-Test results

| | |
|---|---|
| Pearson Correlation | 0,078 |
| Hypothesized Mean Difference | 0 |
| df | 181 |
| t Stat | 7,005 |
| P(T<=t) one-tail | $2,341 * 10^{-11}$ |
| t Critical one-tail | 1,6533 |

On the opposite with respect to the previous scenario, we can easily see that gamifying the download of a component from the store, due to its inherent simplicity and considerable gain in terms of point, has brought a boost in the number of components downloaded. This is certified by the fact that $t - obs > t - crit$ for $\alpha$=0.05 and $\alpha$=0.01. Due to the complexity of Webratio as a tool, we cannot be sure that the users truly used the components they requested, but we can state for sure that gamifying the store has increased users' awareness of the possibility of extending the basic functionalities of the tool, whereas previously many users were not even aware of the store's existence.

### 8.5.5 Increased Participation

- Hypothesis 5: Gamification mechanics can induce users to participate and read more tutorial article about the use of WebRatio.
  - HP 5.1 Increased the number of WebRatio tutorial article read by the users.

*Read an article analysis*

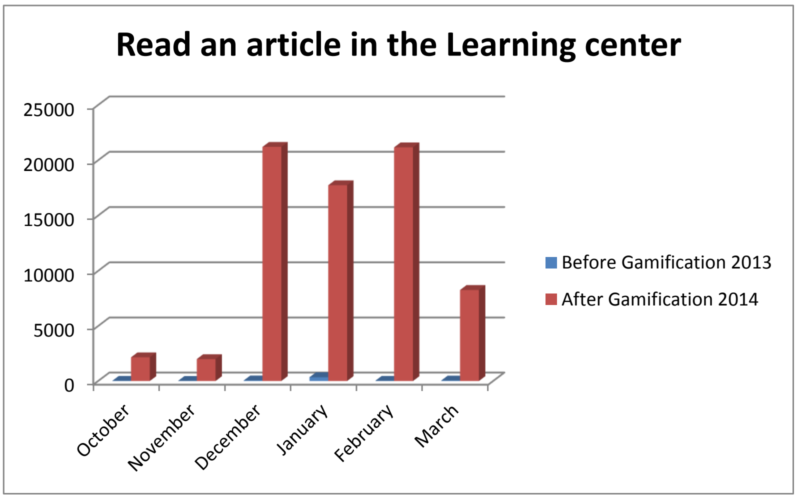



Figure 8.23: Read an article graph

| Read an article | Before Gamification | After Gamification | Difference |
|---|---|---|---|
| Sum | 494 | 72548 | 72054 |
| Mean | 2,714 | 398,615 | 395,901 |
| Variance | 22,448 | 331503,365 | 331480,916 |
| Standard deviation | 4,738 | 575,763 | 571,025 |
| Asymmetry | 2,389 | 3,461 | 1,072 |

Table 8.40: Read an article data

Table 8.41: Read an article t-Test results

| | |
|---|---|
| Pearson Correlation | 0,0787 |
| Hypothesized Mean Difference | 0 |
| df | 181 |
| t Stat | 9,282 |
| P(T<=t) one-tail | $2,57 * 10^{-17}$ |
| t Critical one-tail | 1,6533 |

The new gamified Learning Center is an upgrade of the Knowledgebase platform that was created in August 2012, and at that time the only feature was to provide a set of educational articles and tutorials related of the use of WebRatio tool.

As we can see at first glance, the results are extremely positive, just by looking at the difference in number of page visited: 72054 more with respect to the same period of the previous year, with a gain of 14458%. For the purpose of completeness, we still perform the t-test. Since we want to prove that the competition among the users that the community has created, push the customers to be more prepared and more interested to learn as much as possible the WebRatio features, we chose as null hypothesis:

$$H_0 : \mu = 0; \text{the amount of article read is due to chance.}$$

and as alternative hypothesis:

$$H_1 : \mu > 0; \text{competition increased the number article read.}$$

Since tobs > tcrit for $\alpha$=0.05 and $\alpha$=0.01 we reject the null hypothesis and accept the alternative, namely, that the difference in number of read article is the obvious result of the gamification effects and not the result of chance variation. Moreover the t-value calculated is also widely greater than the critical t-value at $\alpha$= 0.0005 (tcrit = 3.6594), thus there is evidence, at the 0.05% level of significance that the gamification has introduced an increase of article read in the WebRatio learning system.
As widely expected, the t-test showed a statistically significant increase. More trained users implies better software produced, an increased ability to reply to technical questions in the support forum and a better overall image for the company.

### 8.5.6 Qualitative Evaluation

The target goal of a gamified application is primarily to augment business objectives with the introduction of game mechanics and retention techniques like the ones that have been described in the previous sections. If the effectiveness of the application of gamification have been quantitatively described, showing how the Webratio portal was able to increase the interest towards the platform, motivating users in acquiring a deeper knowledge, and self sustaining a user-centered customer support, the only way to actually measure qualitatively if such a change has been satisfactory is inquiring the company itself.

For this reason, an interview with Stefano Butti, CEO and Co-founder of WebRatio, has helped lighting shed on the matter. Stefano has been responsible for setting the overall direction and business development strategy for the company during the years. In 2001 he was one of the founders of WebRatio, and has served in different roles since: WebRatio analyst, sales engineer, application development manager, software factory manager and sales manager. Stefano's opinions on the contribution that the CUbRIK Gamification platform has brought to tackle with the previously silent community is summarized as follows:

"*When perception changes from job to entertainment, the increased users' experience turns into a company gain in performances. Interaction with users is a pillar for a successful company; at Webratio we had hundreds of users that interacted with us through several heterogeneous applications, but the effort was not properly channeled since the users did not felt a common user experience. A Community was built to foster discussion and comments, but simply bringing customers together was not the best way to turn them into active participants, and the community quickly lost traction, becoming deserted and unused.*
*We wanted to provide a better user experience and a healthy community. This is very important for us because when a new potential customer come to us and sees a healthy community is much more motivated in wanting to adopt Webratio as a solution. Feedbacks, suggestions and solutions to common problems that the user experienced were also another key factor of having active users.*
*Ultimately, higher quality data are the ones that consumers volunteer to do; providing an incentive to them to do it in an entertaining and synergistic way could have changed everything.*
*Thus we started to consider about gamifiying their experience on our platform to collect details of the actions users perform within the Portal (Website , Forum, Learn, Shop) and self empowering active participation.*
*The CUbRIK gamification platform was able to tackle with our problem in an effective and cheap way thanks to its openness and for being supported by solid background researches.*
*The openness of the integration layer allowed us to combine all our heterogeneous components without having to change the underlying existing infrastructure to build up a unified gamification system.*
*After just 8 months we were able to start an internal beta testing phase with a*

*small set of users; the gamification platform proven to be effective at increasing the interaction with the existing platform by a factor of 10.*
*These results are especially relevant for two reasons: although our product is innovative and one of the leading solutions in the model driven development field, we are still not a well known company in the software development world, thus we have to prove to our potential prospect and customers that we have a large community, a large user base that help us in improving our product by providing feedback, suggestions and willing to self-sustain a community based support able to solve issues on its own.*
*One of the best means to reach these objectives is to show that there is a lively community already in place; gamification is the key to this process and the right incentive to our customers to show to their peers what Webratio is capable to offer. Will this solution be effective on the long run? Only time will tell. For now, given the numbers, the results achieved so far and the presence of users coming from all over the world and competing in our gamified community, we are proud and satisfied to have adopted such a solution.*"

## 8.6 Summary

In this chapter we described the implementation of a novel Gamification Platform for an enterprise business case, Webratio Headquarters. The platform has been created by following the design guidelines, principles, datamodels and methods described thorough this work and has been able to exceed the target business objectives both quantitatively and qualitatively, as it has been reported.

The chapter has described a real world problem from a software development company, Webratio, and decomposed it into the necessary steps to solve it with the gamification design process. The company had several heterogeneous components that were used to interact with its users: a institutional website, entry point for the world of the business application, a Forum in which users can discuss about best practices and issues related to the software, an online store in which the users could share and sell their custom plugins and an E-Learning platform that could be used as a training mean to improve one's own skills.

The Gamification platform is a collector for all the actions that can be done in the system and able to solve the fragmentation of the tools at disposal of the company; it introduces the concept of reputation and participation scores in order to distinguish between the ability and competences of the users and the contribution of the users in the platform, assigned each time a particular action has been performed by a user. The platform allows a full customization over all the gamification elements that can be introduced, like badges, achievements, leaderboards and scoring mechanisms, that has been tailored by the company following their needs.

After the implementation of the platform, to verify the feasibility of the chosen approach, an internal beta test was conducted, followed by a public test. The results of the tests have been reported and analyzed by comparing the data collected in 6 months of two consecutive years(2013-2014): the first year's data concerned the use of the old platform, without any gamified elements, while in the second year the data were collected from the gamified business application that was already in place. The data collected was used to verify 5 research hypothesis related to gamification and related to the company's business objectives, namely the ability to improve self-powered user customer support, improving the image of the company, increase user retention, facilitate the use and development of extensions and incentivate the use of the Learning Center.

If the qualitative considerations of the CEO of the company were more than positive, the results obtained by comparing the data over two years using Student T-Tests provided one of the first successful examples of gamification applied in a business case backed up by statistically relevant considerations. In particular, the introduction of the gamified platform was able to increase the number of users registrations by 105%, the number of e-learning articles read by over 10000%, to create a lively forum even while maintaining the same number of monthly users that were accessing the platform and increase the awareness of the users of the presence of custom components that could be downloaded from the store, reaching the main business objectives that the company defined at the beginning of the project.

# Chapter 9

# Game with a Purpose: Sketchness

## 9.1 Background

The *Social Fashion Trend Intelligence* is one of the application demonstrators of the CUbRIK [159] project. Its purpose is to provide fashion trend analysis tools for small and medium businesses in the fashion industry, exploiting and integrating machine, human and crowd-sourced tasks. It allows to identify insights into consumer preferences and behaviour. The CUbRIK Fashion Trend application is the second demonstrator developed in the CUbRIK project and is devoted to providing SMEs working in the fashion sector with a search application for their innovation.

The tool provides Fashion SMEs with information about the opinions and feedback of their potential customers, their preferences, what they like about clothes and current trends, in terms of colours and textures. In this application, primarily existing content from social networks is used (e.g. fashion pictures and keyframes extracted from YouTubes videos which are crawled from Twitter) in order to make it an efficient tool for market analysis. The application is accessible at the address: `www.fashiontrendanalysis.eu`. The goals to be achieved by the application are manyfold:

**Fashion Images Crawling** by crawling fashion related content from social networks and further processing images, it is possible to extract segments representing different garments and their features. To achieve these results, automatic segmentation of fashion garments is required, yet no algorithm available in literature has proven to be accurate enough, as it will be presented in the following sections.

**Trend Analysis** , that is the ability of analyzing clothing preferences based on the clothing category the SME user has selected, by relying on the images extracted from social networks and further processed for features and segments extraction. The problems that arose while trying to solve this phase are the same as in the previous scenario.

**Fashion Matching** , the possibility for a SME user to select an image or uploading one and search within the system for similar images to be used as a base in creating an outfit. Similar images are proposed to the user according to similarity criteria, time and spatial information set by the user. Once again, detecting and discriminating garments automatically is a not an easy task.

One specific aspect concerns the trend analysis by allowing feature-based analysis and identification of similar garments (e.g., by color, texture etc.). Since these task classes currently cannot be solved easily and with satisfactory performance by existing computational approaches, they could benefit from incorporation of human computation. In any human computation approach to problem solving, including the design of GWAPs, a problem is mapped into a set of tasks, which are then assigned to both human and machine executors. The resource allocation is done in a way that optimizes a specific quality criterion on the problem-solving process, e.g., the quality of the solution sought or the resources (in terms of time and/or monetary cost) spent to find it.

The scenario that is being considered is related to the elicitation of fashion trends from images representing people wearing garments. The process will be able to detect those trends by analyzing correlations among color and texture features extracted from different garments. The process will also identify peculiar facial traits of a subject, such as gender and age, to estimate if there are underlying trends based on these features. To do so, it will use off-the-shelf components such as [160, 161] in order to perform upper-body/ lower-body detection and [162] to perform face detection and annotation. However, the outcomes of these algorithms are largely affected by the characteristic of the input image set and how the subjects of the photo have been captured. Missing faces or body parts can drastically reduce the quality of the output, but can, on the other hand, immediately be verified by humans.

The envisaged processing pipeline consists of a sequence of both automated tasks and human computed tasks, as illustrated in Figure 9.1, with the goal of improving the detection of the relevant portions of an image when the automatic algorithms fail. Specifically, the human contribution is exploited at two levels: i) to segment portions of the images by identifying the position of garments or human body parts; ii) to annotate faces in images that portray human beings. In particular, this Thesis focuses on the description of a GWAP that can be used to assist image segmentation.

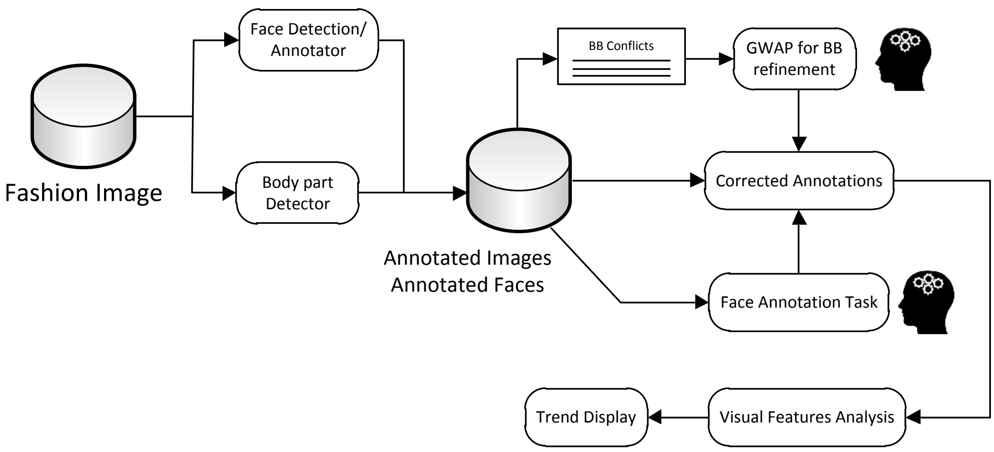



Figure 9.1: The process of the fashion trend mining pipeline

The pipeline receives as input an image, which is processed in parallel by two automatic tasks. A *Body Part Detector* task analyzes the image to obtain bounding boxes of different body parts (e.g., upper body, lower body, etc.), so as to identify the regions of interest where garments might appear. A *Face Detection/Annotator* task detects faces and provides semantic annotations (e.g., gender, age) that can be inferred from the extracted low-level features. The output of these tasks (either bounding boxes or annotations) are associated with confidence measures produced by the automatic algorithms. Whenever the confidence is sufficiently high, these outputs can be readily used for the subsequent steps. Otherwise, the annotated images are given as input to specific tasks that exploit humans. In particular, if the confidence of a bounding box is low, the process will feed the image as the input of a specific GWAP for image segmentation, called Sketchness. By exploiting a GWAP, CUbRIK shows the ability to adopt different approaches to solve human computation and compare the efficiency and quality of the tasks performed using different paradigms. The aggregated results of the game will then form the *Corrected Annotations* that will serve as input of the final steps of the process. These images are processed by the *Visual Features Analysis* task whose role is to extract low-level visual features based on, e.g., color and texture. Finally, the extracted visual features are combined with facial annotations and available timestamps in order to mine similar characteristics and identify possible trends using visual analytics techniques [163].

## 9.2 Objectives of the Gwap Adoption

The key technical principle of CUbRIK is to create a "white-box"version of a multimedia content & query processing system, by splitting its functionality into a set of search processing "Pipelines", i.e., orchestrations of open source and third-party components instantiating current algorithms. Given the nature of the tasks, that is handling and enriching multimedia content, the sole contribution provided by the machine is not sufficient, has it has been stated in the previous section. For this reason, CUbRIK has embraced the Human Computation paradigm following two different approaches:

**Crowd-enabled Applications** are the first kind of application implementing the mechanism of Human in the loop. These Applications are targeted to enable individual and social participation to search processes; the actual Crowd mechanism leverages on the CrowdSearcher Framework [164] implementing distributed work solutions for multimedia search. The Framework is in charge of design, execution and verification of tasks by a crowd of performers; In particular it manages core aspect including but not limited to Human task design, People to task matching, Task assignment, Task execution, Executor evaluation and Output aggregation.

**GWAP Applications** are the other kind of applications implementing the human in the loop mechanism. As for Crowd-enabled Application, even for GWAP a specific GWAP Framework is designed. By leveraging on playing games, the Gaming Framework, actually outsources certain processes steps to humans, in an entertaining way. Typical steps include labelling images to improve web searching, transcription of ancient text and other activity requiring common sense or human experience. Basic mechanism is the training of the system to solve problems mainly related to media understanding and contents interpretation.

This section is related to the analysis of the second type of applications and in particular to a game, called Sketchness, that has been developed for one of the most difficult media refinement problems, that is object identification and segmentation. Following the development process detailed in the previous chapter, the requirement specification for the Gaming Framework is hereby described.

#### Requirements Specification

**Business Objectives** Objective of the Gaming Framework and the GWAP that has been developed within CUbRIK is to improve the metadata associated to multimedia content or creating it in the case in which the content was completely unannotated.

**Target Players** Children to Adults. The broader the audience, the larger will be the results that could be obtained and given the simple nature of the task there are no limitations on the possible users of the game.

**Target Behaviors** Object Identification, Object Segmentation, Inappropriate Content Filtering.

## 9.3 GWAP Design

Object recognition is the task of finding a given object in an image or video sequence. Even though the task may seem easy for humans, given their superior perceptual capabilities, it is still a challenge for computer vision systems in general. Garment detection is a particular case of object recognition problems in which the goal is to identify specific garments and their position within an image. Specialized algorithms exists to tackle with the problem but upon failure, human intervention is needed. Sketchness [95] is a multiplayer puzzle game in which the players take turns to draw the shapes of objects in an image in order to make the other players guess the underlying object. It has been created as a GWAP to solve object recognition and segmentation problems for the fashion domain. The application has been developed following the process described in 5.2; in the following, the task to be solved and the associated game mechanics are described by detailing just the meaningful phases.

### Task Design

The task to be solved requires the segmentation of a stated object within a provided image; given a textual tag related to an object in the image, the expected output is a mask used to recognize its position.

### Task to Mechanics Matching

The manual tracing of the contour of a particular garment in the image is an activity that can be described simply as “drawing”. In principle, restricting the objects domain just to garments, the shape should be a sufficient hint to let a human identify the underlying object, thus validating the task performed.

Based on these two assumptions, it is possible to search in the gaming literature for games involving “drawing”as a main conflict. By doing so, it is possible to find a particular game genre, called “Draw and Guess“games, in which players try to identify specific hidden words hinted by their opponent drawings. Usually the game mechanics rely on the imagination of the player to draw the requested subject, so to match the game with our problem, the rules of the game are modified, requiring the player to use an underlying image to draw the contour of the object to be recognized. Given a word specifying a garment and a fashion image, a player is asked to draw the garment in the image by tracing its contour for the other players to guess it, thus solving the needed task.

### Game Design

*Sketchness* is a multiplayer Game with a Purpose used to obtain segmentations on fashion related images that couldn't be processed automatically by the Body Part Detector component in the CUbRIK's Fashion Trend pipeline. It can be accessed at the address `http://www.sketchness.com/`.
The GWAP can be used to:

- Check if a particular fashion item is present or not within an image by asking for a confirmation to the crowd in the form of a tag; the image can also be tagged in the case in which it was not previously annotated.
- Segment the tagged fashion item within the image by asking to the players to trace the contours of the object. In each round a player is chosen at random to be the Sketcher while all the others will play as Guessers.

Players take turns to draw the shapes of objects in an image in order to make the other players guess the underlying object; if the correct word is identified both the drawer and the players that were able to spot the word receive points, based also on the time the response was submitted. After a certain number of rounds, in which every player is asked to draw on different images or to guess over the images drawn by the others, the winner will be the player that has achieved the highest score.

The game consists of 10 rounds in which one player has access to the image to be segmented and a tag representing the object or the part of the image that has to be selected, while the other players are asked to guess the tag based on the drawings made by the first player. These two roles are defined in the game as *Sketcher* and *Guesser*.

In each round a player is chosen at random to be the Sketcher while all the others will play as Guessers.
The *Sketcher* is given as input an image coming from the annotated images with low confidence. During the round in which a player is the sketcher, (s)he will be the only player with the rights to see the image, while the image will be hidden to the other players. The sketcher is asked to provide a tag for a garment visible in the image, such as "tie" or "trousers" or he will be given a tag generated from previous matches; (s)he will then be asked to draw the given word by tracing the contour of the object specified in the tag over the

image within a limited period of time, typically 120 seconds. The sketcher is also given the possibility to skip the drawing of the image object if (s)he cannot or does not want to play with that particular image, or to mark the image as inappropriate if the content is corrupted or contains wrong material. The Sketcher is not allowed to give hints other than the contour of the image, such as writing the word on the whiteboard or hinting through the chat, but (s)he can draw logos that contain written text (e.g., the name of the brand of the garment) if it is shown in the image. An example of a possible view of a Sketcher's interface is provided in Figure 9.2.


Figure 9.2: An example of the Sketcher's interface.

The *Guesser* is asked to type guesses about the object being drawn in a text box. The Guessers cannot draw on the whiteboard and are able to see only the content that is being drawn by the Sketcher that has been chosen for that round. The guesses of a player are visible to everyone except when the player has typed a word that is close to the requested one or has guessed correctly; in such cases the word is visible only to the player that has typed it and the Sketcher. An example of a possible view for a Sketcher's interface is given in Figure 9.3.


Figure 9.3: An example of the Guesser's interface.

For each round a player receives Score Points based on his/her performance and role. The Sketcher receives 10 points for the first correct guess provided by a Guesser. One point for each additional right guess from other Guessers is awarded, up to a maximum of 5, and a total earning potential of 15 points per round. The first guesser to give a right answer is awarded 10 points. The 2nd guesser to provide the right is awarded 9 points, 3rd guesser gets 8 points and so on, with a minimum of 5 points. The scoring system ensures a high payout for guessing quickly, drawing effectively and getting as many correct guessers as possible. A round will end when any of the following occurs:

- All Guessers type the correct guess.
- The time available to the Sketcher runs out.
- The Sketcher decides to end the round by clicking skip.
- The Sketcher does not provide a tag or draw anything within the first 20 seconds.
- After a set time (usually 20 seconds) following the first correct guess.
- If enough players press the "Warn Player" button when the Sketcher violates any drawing rule.

In order to make the Sketcher respect the rules, the player can "warn" him/her if (s)he is doing something wrong. If all the opponents warn a player, (s)he will skip the turn without getting any point. Before switching to another Sketcher, the underlying image and the drawn contour are revealed to all the players. If a player has not drawn the contour of an object inside the image (s)he can get a warning flag, thus receiving no score points for that round. At the end of the 10th round, the player with the highest score is declared the winner.

### 9.3.1 Task Injection

The purpose of the game is to perform segmentation on the images provided as input, which derive from low confidence outputs of the *Body Part Detector* task in the pipeline. This objective has been achieved by suitably modifying the game mechanics of a well known and appreciated game category, called "drawing and guessing games", to implicitly solve segmentation problems while playing. We call the process of hiding human computation tasks beneath existing game mechanics "Task Injection".

The proposed game differs from the existing drawing and guessing games since, while the traditional game mechanics rely on the imagination of the player to draw the requested subject, in the GWAP that is being developed the player uses an underlying image to draw the contour of the object to be recognized. Players may solve two different tasks during their gameplay: they may tag the provided image by identifying garments and they are asked to segment the picture with the contour of the object stated in the tag. The tag for the image can also be provided as an input by the previous components of the pipeline, in order to specify the garments or areas of the image that the system has not recognized and thus requiring segmentation. The image itself is an immediate hint for the player that will just need to trace a contour of the tagged object within the image to get a representation good enough for the other players to guess it.

In this way the game can build a bounding box to mark the position of the object within the image if the object has not been segmented, or to validate automatically derived bounding boxes. The correspondence between the segmented part of the image and the required one is enforced by the answer of the guessers: if they agree with the tag that is available only to the Sketcher, it can be assumed that the contour drawn by the Sketcher truly represents the target object to be segmented in the image. The game provides the results to the other components of the pipeline through the generation of binary masks corresponding to particular garments. Given the requirements for the game, the presence of a considerable amount of user generated content fed into the system as tasks, the possibility to evaluate the contribution of each user by comparing it to a ground truth is impractical. It is impossible to choose the players to play with beforehand, excluding someone, thus a trust model for the players has not been defined. On the other hand is possible to assign a reputation score to the players based on their contribution.

The novel approach that has been used for Sketchness aims at automatically assigning a reputation score to the user based on his/her past annotations in the system with respect to the contributions provided by all the other participants. Correct contributions are more likely to be similar to each other while malicious users contributions are, for their very own nature, random and dissimilar to the expected result. Once a reputation score for the user has been obtained, his/her contributions can be weighted accordingly in the aggregation phase in order to lower their influence in the generation of the aggregated result. The generation of aggregated masks can be performed thanks to the actions submitted as part of the gameplay; the most meaningful action in the game is the creation of an objects contour by one of the players whose role is the Sketcher, and the validation of such contour by a successful guess by one of the guessers. In case in which no collusion among players takes place, the confidence of the player is automatically elicited through the inversion problem mechanic typical of the game, since he will have submitted a meaningful contour for the other players to guess. In case of cheating, the confidence of the submission will be automatically lowered offline when it will be compared to the other annotations during the aggregation phase.

The detailed description of the algorithm has been provided in 5.7. In general, the collected tracks do not follow exactly the silhouette of the objects. This is due to several factors, e.g.,: i) the different ability to draw with the mouse or with a tablet; ii) the fact that a guesser might identify the correct tag before the sketcher completes the drawing; iii) the presence of malicious players that try to fool the game by, e.g., writing the tag or drawing a simplified sketch of the object which is not overlaid to the silhouette. For these reasons, the same image/tag pair is reused in multiple games with different sketchers, so that tracks can be aggregated together to obtain a reliable segmentation by using majority voting or more advanced techniques like an aggregator that makes use of the Reputation model previously defined.

## 9.4 Architecture

The design of the architecture for Sketchness posed several problems due to special requirements dictated by the requisites of the project and considerations related to the usage of the application itself, in particular:

- The lack of previous guidelines or best practices in the development of Games with a Purpose from an implementation point of view.
- The need of making the entire platform open source and fully customizable as a starting point for the development of other GWAP
- The need of lowering the usage barrier requirements for the application, by making the game accessible by the greatest possible number of users.

The architectural definition for the backend of a game with a purpose is the same as the one of a generic crowdsourcing platform, with the required extra layer used to manage gameplay data and to present the task in a entertaining way for the users. The definition of suitable data structures has already been pursued in Chapter 7. Thus, the backend of the GWAP has followed the same architecture of the CUbRIK Project, which has been specifically designed to handle human computation tasks.

### 9.4.1 The CUbRIK Architecture

CUBRIK is a distributed system layered in four main tiers, as shown in Figure 9.4.

- The **Content and User Acquisition Tier** is responsible for registering content and users into the system. *The Subscription Manager* handles the registration to the platforms of two classes of users: searchers and

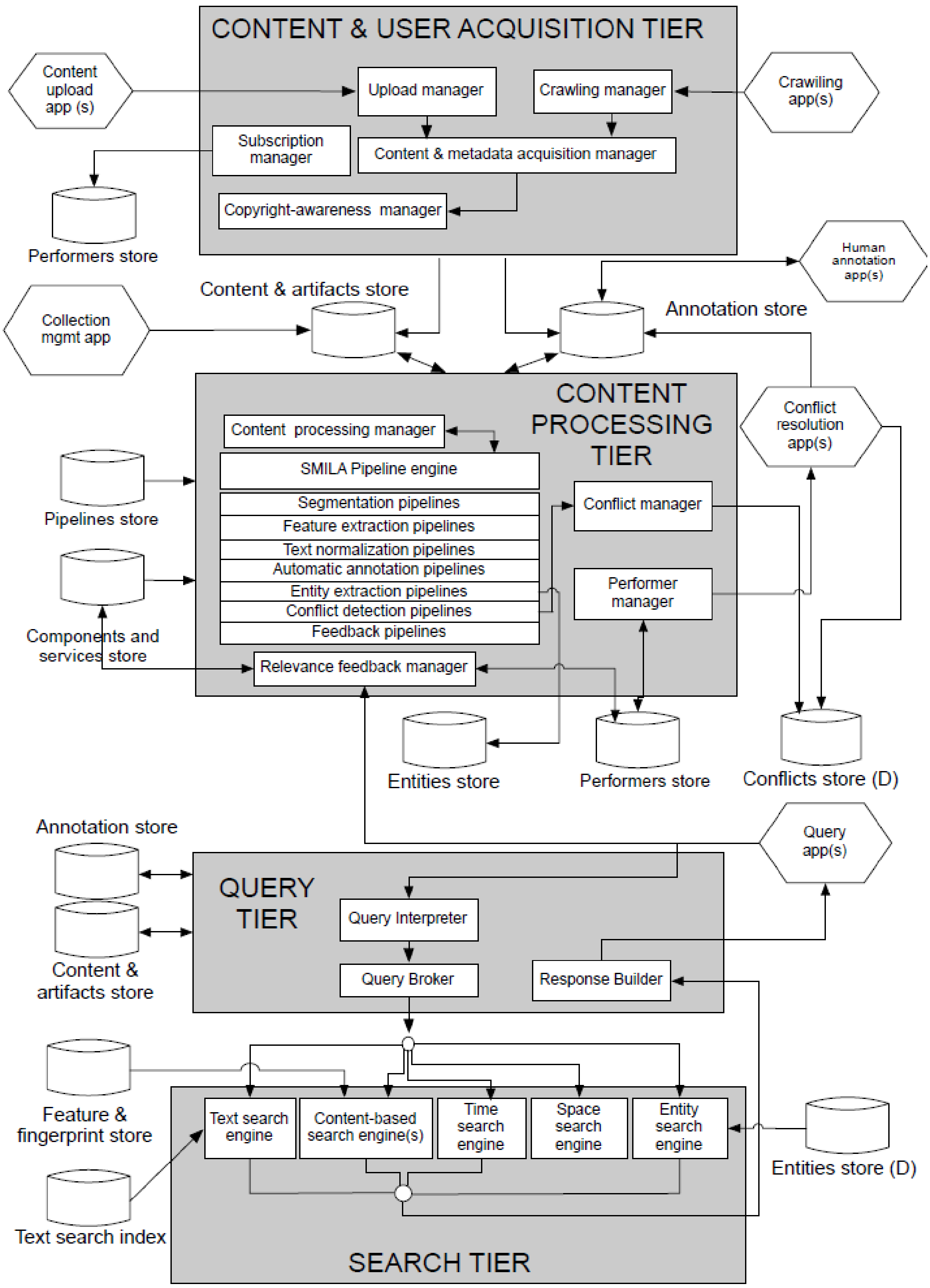



Figure 9.4: The architecture of the CUbRIK Framework

performers. Searchers use CUBRIK applications for finding and interacting with information; they may be exploited to get feedback on query result quality. Performers execute tasks (via gaming or Query&Answer) to provide contribution, semantic annotation, and conflict resolution. Content is added to a CUBRIK platform via upload or by scheduled crawling, handled by the *Upload Manager* and the *Content and Metadata Acquisition Manager*; crawling may import into CUBRIK external metadata in popular formats. Content registration gives content element an internal ID, and then stores the content element in raw format, the associated crawled or manual metadata, and rights information.

- In the **Content Processing Tier**, the *Content Processing Manager* listens to a queue of pending content processing requests and is responsible for starting, suspending, resuming, terminating, and rescheduling tasks. The *Conflict Manager* is the core component for integrating human computation; it manages the set of conflicts and their assignment to applications and performers.
  - In the simplest case, conflicts can be assigned to an application, which manages their allocation to performers, possibly using data provided by the *Performer Manager*. This is the typical case for simple GWAP applications, where the interaction logic is user-independent and only basic profile data, like the skill level of the gamer, are employed to decide which conflict to present.
  - Alternatively, conflicts can be assigned to an application-performer pair: in this case, the association of the performer is managed by CUBRIK and the application routes the conflict to the performer suggested by the platforms. This is the case of more personalized applications, like Q&A, where a mix of history, profile, and trust data of the performer can be used to route the most appropriate questions. A conflict resolution application can also be an existing third-party application (e.g., a crowdsourcing application on top of a commercial platform).

The *Conflict Manager* is responsible for closing a conflict and storing the produced facts. The *Performer Manager* is responsible for keeping data about performers (profile, social network centrality measures, history of solved conflicts, throughput, quality of decision, etc.), which are used to optimize task allocation.

Some pipelines are designed to receive feedback from the user on the results of a query. This feedback is routed to a *Relevance Feedback Manager* module that updates the level of trust of performers (human and automatic) in the component and performer store.

- The **Query Processing Tier** consists of one or more *Query Applications*, which contain the front-end for issuing queries and viewing results. Queries are expressed according to a multimodal query language, serialized and submitted to a CUBRIK platform (through Web services API); results are organized according to an application-dependent result schema, serialized, and returned as responses from CUBRIK to the application. The *Query Interpreter* analyses the query and understands its class. Classes of supported queries are: Keyword, Visual similarity (image and video), aural similarity, and multimodal (keyword + one similarity criterion). The *Query Broker* translates the query into the format expected by the search engine(s) of the Search Tier and dispatches the query or sub-queries to the relevant search engine(s). The *Response Builder* normalizes and fuses the responses from the search engines(s) and creates a single result list, to be returned to the query app.
- The **Search Tier** contains a collection of independent search engines. Each search engine can access the content and annotation store(s) to build/rebuild its indexes. Indexing is independent and asynchronous w.r.t. content processing and acquisition. Each search engine listens to the content processing manager events, in order to understand when to build, re-build, extend, and update its indexes.

### 9.4.2 The Gaming Framework

The CUbRIK project identifies in the fact that games are designed and tailored over the specific task to be solved the major issue of GWAPs. Indeed this can lead to an user experience that may still be perceived as work and not as entertaining. Therefore the investigation on the design of game mechanics and motivation techniques, and the definition of a methodology for the assignment of human computation tasks are needed. These problems are addressed in CUbRIK with the use of a Gaming Framework that provides a set of tools and guidelines that can ease the development of attractive applications able to exploit human contributors.

First of all it is necessary to understand the collocation of the Gaming Framework in the whole platform architecture. The Gaming Framework main contribution in CUbRIK is to bring humans in the loop of the search process, improving the platform services. The main task is solving tasks to be executed with the aid of users in situations in which other processes (pipelines) have failed or for which no known software component can be used. GWAPS are classified as applications and they can be:

- Human Annotation apps, which directly provide the high level metadata that can only be generated by means of human contribution;
- Conflict Resolution apps, which support the Conflict Manager in the Content Processing Tier to resolve the conflicts that machines cannot handle alone. These gaming applications can resolve the problem of potential conflicts by using human resources.

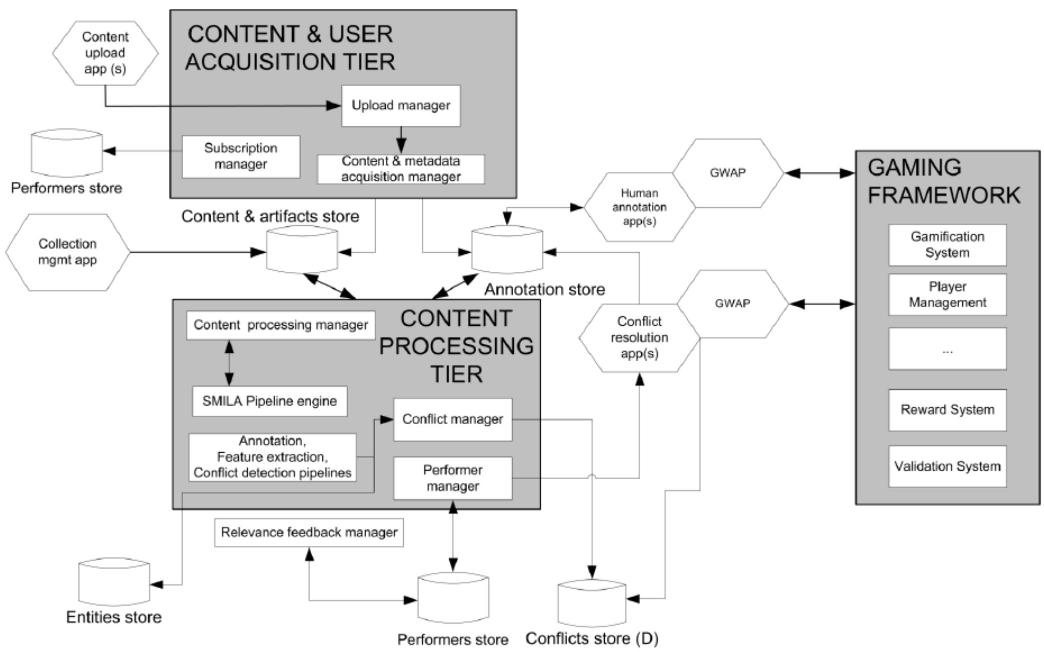



Figure 9.5: GWAP framework in the CUbRIK platform.

Figure 9.6 shows the architecture of the Gaming Framework. Multiple tasks are introduced into the system and, as previously described, they can be either tasks that cannot be performed by other processing units of CUbRIK without the human in the loop, or tasks that are referred by the Conflict Manager Unit and need human judgement to remove the conflict.

The *Gamification system* is the component that handles the logic of the game, presented as a complex state machine to handle the different actions performed by the users in the application. It offers also components to handle human computation tasks to be integrated into the game by maintaining a list of running games, active users and checked out task instances. In this way it uses this information to orchestrate the results and to assign suitable players to running gaming sessions. The other components are, on the other hand, offering accessory functionalities to render the actual developed game more appealing to the users:

- The *Reward System* is responsible for inducing motivation and increasing participation of the users.
- The *Visualization Components* includes Graphical User Interfaces (GUI) and their supporting components.
- The *Player Management* is responsible for registration, management, authentication and authorization of players within the CUbRIKs platform.
- The *Validation System* is responsible for confirming the results of user inputs with other available resources, by using the techniques that have been described in 5.5

### 9.4.3 Sketchness Architecture

Figure 9.6 presents the architecture of the implemented game and the associated content management system used to store images and associated metadata.

The game has to be played online by as many people as possible, without requiring any particular software or equipment other than the ones commonly used to browse the Internet. For this reason, the application has been developed in HTML5 following the MVC software architecture pattern, using the

Play! Framework for the backend and a custom CMS developed in NodeJs to store the annotations submitted by the users.

The game itself has been developed as a complex state machine that has been created in a specular way both backend (Java) and frontend (HTML5 + Javascript). The messages among servers and clients are exchanged by using specialized JSON packets sent via websockets.

The views represent the interfaces of the game that are in charge of emitting events related to the gameplay actions performed by the players.

The controller is in charge of monitoring and filtering only the meaningful received events, forward them to the models and display the updated status of the system through the views. The models are used to handle the gameplay of the game and persist the operations performed by the player with the use of a custom CMS. By exploiting a message bus, the game has been divided in independent modules able to communicate among each other, be turned off or reused in future applications; this architecture founds the basis of the framework that has been used for rapid GWAP prototyping and development.

The GameRoom contains the logic of the game that assigns the roles for the players in each round, associate tasks to be solved, verifies the answers of the players and assigns points to them all by following the finite state machine that has been already described and it is the only module that requires modifications to be able to build a different game.

The other independent modules, are the Chat, that contains the logic for the creation of a chatroom used to exchange messages among the players and the game and the Paint module, which allows the creation of a canvas that can be used to share drawings and images.

The CMS module receives packets from the shared message bus and provides interfaces and methods to handle the storage of actions and annotations within the *Content and Metadata Acquisition Manager* for future usage, in particular to calculate the reputation of users and store the aggregated masks for a particular garment.

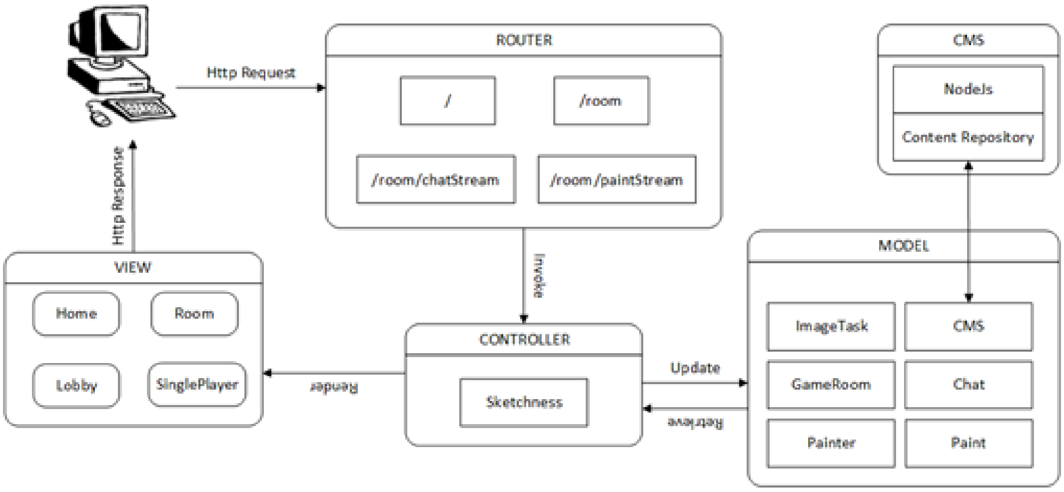



Figure 9.6: Architecture of the Sketchness GWAP

# 9.5 Results Evaluation

## 9.5.1 Task Resolution Results

The problem of image segmentation is not new in the field of Human Computation and Crowdsourcing: several works in the past have tried to tackle with it following three approaches: GWAP, Pure Algorithmic and Crowdsourcing initiatives. In the following, for each of the approaches, the most meaningful examples are described, in order to define a starting point for comparisons with Sketchness. Finally, the results obtained by Sketchness on a small dataset of about 200 images is then compared with state of the art algorithms, in order to prove the feasibility of the GWAP approach.

#### Image Segmentation: GWAP

**Peekaboom** One of the historical Von Ahns GWAP [77]. The game has been played by 14,153 players providing 1,122,998 traces during a month. No experiments have been performed on a real dataset and there is no information regarding the images that has been used. The accuracy of the retrieved bounding boxes has been computed by choosing 50 random images-word pairs that had at least two matches played by two different couples. Peekaboom generated bounding boxes have been compared with the ones provided by 4 volunteers over the same 50 images, generating 200 bounding boxes; these bounding boxes have then been compared with the ones generated by the game using Jaccard Similarity Coefficient. The mean overlap obtained in the experiment was 0.754 with a standard deviation of 0.109. It is not possible to compare directly the approach used in Sketchness with the Peekabooms one since the images that has been used in this game have not been released publicly.

**Squigl** Another game belonging to the original GWAP suit. The game was assigning points to the players based on how close the contours they traced were. It has been mentioned in several articles related to GWAP but no scientific paper or data has been released, the game probably failed to achieve the expected results.

**Ask'n'Seek** A GWAP that asks users to guess the location of a small rectangular region hidden within an image with the help of semantic and topological clues (e.g., "to the right of the bus"), by clicking on the image location which they believe corresponds to (one of the points of) the hidden region [79]. Goal of the experiments of the work describing the game was to estimate a certain amount of clicks necessary to obtain a defined quality level.

The effectiveness of the approach has been validated by creating a synthetic player to replace the real one to perform the segmentation on the dataset and, afterwards, validating the feasibility of the synthetic player against the real gaming traces generated during real games. The traces of the real games involve 50 players, 255 matches and 24 images chosen from the PASCAL VOC2010 dataset,that shows objects divided into 20 classes. The mean Jaccard Index obtained by the game over the 20 classes is 0.5836, while results for each class are shown in Figure 9.7

**Fine Grained Crowdsourcing for Fine-Grained Recognition** The work [165] presents an approach for identifying discriminative features (ROI that distinguish one instance of the same class with respect to the other) within an image with the use of a GWAP. The evaluation has been performed over two datasets, CUB14 and CUB200. The CUB14 dataset contains 14 classes of birds, divided in two subgroups and has 42 pairs of classes to compare. 16336 annotations from 4101 games using 210 traning images. Table 9.1 reports the precision of the approach in recognizing birds.

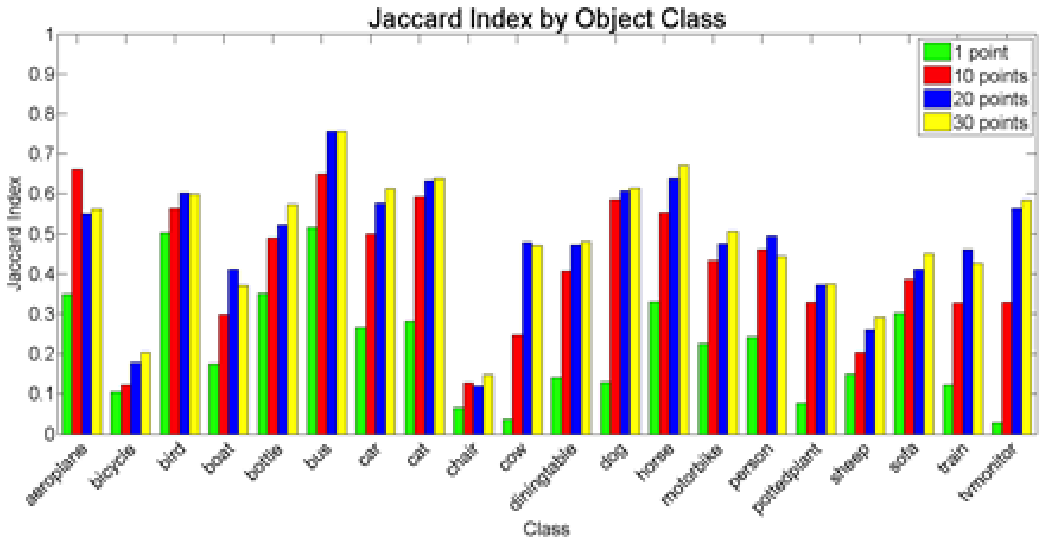



Figure 9.7: Jaccard Index by Object Class for Ask'nSeek

| Method | mAP (%) |
|---|---|
| MKL | 37.02 |
| Birdlet | 40.25 |
| CFAF | 44.73 |
| Authors' Approach (SPM 1x1) | 52.98 |
| Authors' Approach (SPM 1x1, 2x2) | 48.63 |
| Authors' Approach (Random Bubbles) | 43.72 |
| Authors' Full Approach | 58.47 |

Table 9.1: Bird recognition in the CUB14 dataset

The authors also show (Figure 9.8 how the accuracy varies based on the number of bubbles; this is used to understand how many traces are needed to achieve a particular accuracy level, in a similar fashion with the Reputation model that has been applied in Sketchness.

The results over the CUB200 dataset, that differs from the CUB14 just for the number of classes, are presented in table 9.2

**A Collaborative Benchmark for Region of Interest Detection Algorithms** In Photoshoot, the GWAP presented in [166], players are assigned the roles of "target"and "shoot"in turns. In a round, the same image is presented to both target and shoot players. The target player places targets on the image by drawing rectangles over the image using drag-and-drop. Up to

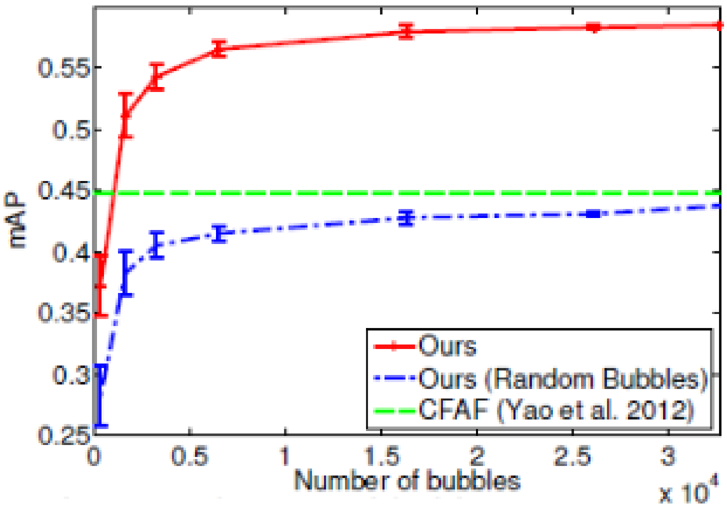



Figure 9.8: Accuracy level over number of bubbles used for objects recognition

| Method | mAP (%) |
|---|---|
| MKL | 19.0 |
| Random Forest | 19.2 |
| Hierarchical Matching | 19.2 |
| Multi clue | 22.4 |
| KDES | 26.2 |
| Tricos | 26.7 |
| Authors' Approach (Random Bubbles) | 26.5 |
| Authors' Full Approach | 32.8 |

Table 9.2: Bird recognition in the CUB14 dataset

five targets can be placed. Without seeing the targets placed by the target player, the shoot players role is to guess where those targets are by shooting at them by clicking on the image. The game has been used to collect the most meaningful region of interest for the images that were supplied to the game. Over the course of one month, the game has been played by 1002 players, 71% of them played more than one session and 36% more than four sessions; of these, more than 20 players played more than 60 times. The images to be annotated were 3000 and 134646 regions of interest were collected, supported by 168352 annotations used to verify them (players were asked to click on the targets). The approach has been benchmark validated against a dataset of 30 images manually annotated by volunteers, and compared also against eye tracking data, in which the region of the image that was stared at most frequently by the users was captured, and has yield to the results shown in Table 9.3

| | **Manually Annotated** | **Eye Tracking** |
|---|---|---|
| Precision | 0.91 | 0.85 |
| Recall | 0.90 | 0.87 |

Table 9.3: Precision and Recall for the Photoshoot GWAP

Even if the results are promising, the approach does not allow full and precise segmentation of a particular object within the image and thus it is not meaningful for making comparisons with Sketchness.

### Image Segmentation: Algorithmic Approaches

**Parsing Clothing in Fashion Photographs** The Fashionista Algorithm [167] is considered as one of the state of the art algorithm for Clothing Parsing, thus it is one of the most meaningful work against which it is possible to test the effectiveness of the GWAP that has been developed. It uses the Fashionista dataset which is composed by 685 photos with 53 possible clothing labels. It contains also 158k un-annotated samples to test against. The results of the algorithm have been validated against a baseline that consider all the regions to be predicted as background and are shown in Table 9.4, where mAGR stands for Mean Average Garment Recall.

| **Method** | **Pixel Acc** | **mAGR** |
|---|---|---|
| Full-a | **89.0 ± 0.8** | 63.4 ± 1.5 |
| with truth | 89.3 ± 0.8 | 64.3 ± 1.3 |
| without pose | 86.0 ± 1.0 | 58.8 ± 2.1 |
| Full-m | 88.3 ± 0.8 | **69.6 ± 1.7** |
| with truth | 88.9 ± 0,7 | 71.2 ± 1.5 |
| without pose | 84.7 ± 1.0 | 64.6 ± 1.8 |
| Unary | 88.2 ± 0.8 | **69.8 ± 1.8** |
| Baseline | 77.6 ± 0.6 | 12.8 ± 0.1 |

Table 9.4: Pixel Accuracy and Mean Average Garment Recall for the Fashionista Algorithm

The general clothing parsing problem, with no prior knowledge about items from metadata yields to 80.8% pixel accuracy. The drawbacks of the approach are related to the fact that it assumes that each superpixel has the same clothing label and encourages over-segmentation to make the assumption nearly true and has severe problems when the body is only partially visible, making the body pose estimator, part of the initialization step, fail.

**Getting the Look: Clothing Recognition and Segmentation for Automatic Product Suggestions** The work [168] describes a novel algorithm for garment segmentation. It is compared against Fashionista by using two different datasets: the original Fashionista dataset and a dataset of fashion images crawled from Yahoo Shopping. Once again the baseline was considered to be all regions belonging to the background. The authors compare themselves against the other algorithm in the scenario in which no prior knowledge about items from metadata is present and the results are shown in Tables 9.5 and 9.6

| Method | Mean Pixel Accuracy | Average Time |
|---|---|---|
| Fashionista | 80.7 | 334 seconds |
| Authors' Approach | $80.2 \pm 0.9$ | 5.8 seconds |
| Baseline | $77.6 \pm 0.6$ | N/A |

Table 9.5: Comparison between Getting the Look and Fashionista

| Clothing Category | Proposed (%) | Random (%) |
|---|---|---|
| Dress | 68 | 10 |
| Skirt | 59 | 2 |
| Blouse | 37 | 4 |
| Top | 55 | 6 |
| Jackets & Coats | 43 | 3 |
| Pants & Jeans | 69 | 12 |
| Boots | 66 | 14 |
| All | 54 | 8 |

Table 9.6: Pixel accuracy of Getting the Look against a random approach

The results are obtained by retrieving all the garment tagged with a particular clothing category within the dataset using their proposed algorithm or by using a random approach; it has been validated against users contribution by collecting 178 image annotations from 11 users. Although the authors are able to obtain similar results than the state of what was considered the state of the art by lowering the computational time, the approach suffers from the same drawbacks: they have failures related to pose estimation or on images presenting perdiodic color changes as (stripes) due to the initial color segmentation approach of the algorithm; the accuracy drops also for high overlapping classes (skin for leggings, shorts for skirt)

**Paper Doll Parsing: Retrieving Similar Styles to Parse Clothing Item** The improved version of the Fashionista algorithm, the Paperdoll [169] algorithm works on a modified version of the Fashionista dataset that contains 229 testing samples and 56 different categories. The performance of the algorithm has been measured in terms of: accuracy, average precision, average recall. The concept of foreground accuracy is also introduced, to show that the approach is not just good at discriminating the background but also to identify elements which appear in the foreground, and the results are shown in Table 9.7

| Method | Accuracy | F.G. Accuracy | Avg. Precision | Avg. Recall |
|---|---|---|---|---|
| CRF | 77.45 | 23.11 | 10.53 | 17.20 |
| 1. Global Parse | 79.63 | 35.88 | 18.59 | 15.18 |
| 2. NN Parse | 80.73 | 38.18 | 21.45 | 14.73 |
| 3. Transferred Parse | 83.06 | 33.20 | 31.47 | 12.24 |
| 4. Combined (1+2+3) | 83.01 | 39.55 | 25.84 | 15.53 |
| 5. Final Parser | 84.68 | 40.20 | 33.34 | 15.35 |

Table 9.7: Paperdoll Accuracy and Precision w.r.t. the state of the art

Even though the algorithm has an accuracy of 84.68% against the 77.45% of the state of the art, the approach suffers from some of the problems that were afflicting also the other algorithms: the approach suffers conflicting items being predicted for the same images, such as dress and skirt or boots and shoes, classifying the same item under different tags and moreover suffers from identification problems for small items or accessories such as rings, bags, necklaces and the such.

### Image Segmentation: Crowdsourcing

**LabelMe** LabelMe [170] is an online tool used to ease the segmentation of images. To validate the tool, annotation experiments have been performed over four object categories: sailboats, dogs, bottles and motorbikes. 18 Sailboats, 41 dogs, 154 bottles and 49 motorbikes were collected through LabelMe and used to train four classifiers. 4000 images for each class were then downloaded using Google, Flickr, Altavista, although not all the images contained instances of the queried objects. The detector trained with LabelMe was used to sort the images returned by the online query tools. To measure the performance, the first 1000 images downloaded from the web for the categories sailboats and dogs were manually annotated to provide the necessary groundtruth used to evaluate the precision of the detector for the ranked images for the two categories, as shown in Figure 9.9

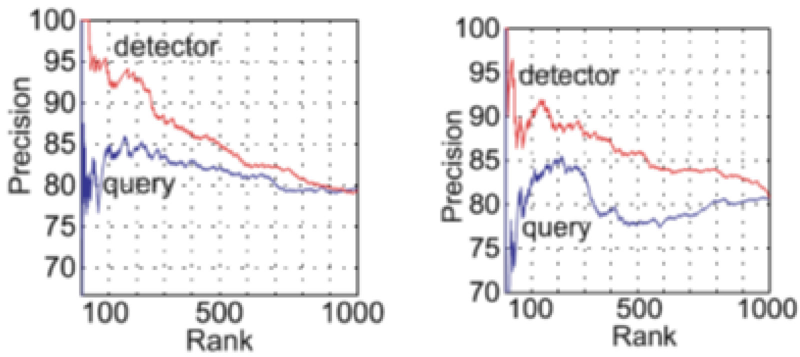



Figure 9.9: Precision of the LabelMe detector for "Sailbots"(left) and "Dogs"(right)

### Sketchness: Comparison with the State of the Art

At the time of writing this section, although the game was still in Beta stage, the data collected within the system were the same as the ones reported in Table 9.8

The accuracy of the system has been evaluated by comparing the results obtained on a subset of the total images fed into the game, obtained from the Fashion-focused creative commons social dataset [171], and processed using the state of the art clothing parsing algorithms, that is Fashionista [167] and Paperdoll [169].

| Type of Data | Value |
|---|---|
| Images Analyzed | 2396 |
| Number of unique users | 844 |
| Avg Number of tags per Image | 9.33 |
| Segmentation Submitted | 15264 |
| Avg Segmentations per Image | 83.01 |

Table 9.8: Summary of the data collected through Sketchness

Figure 9.10 shows on the left a sample image taken form the dataset along with the annotations submitted by the players of the GWAP, on the right the resulting binary mask used for garment segmentation and obtained after the aggregation phase.

Figure 9.10: Fashion-focused Image processed by Sketchness

Figure 9.11 shows on the left the overlay of the binary mask obtained by the GWAP on the original image, on the right the results obtained by applying the Fashionista algorithm on the same image.

All the images that were used have been manually segmented by experts to identify the most meaningful garment in order to create the necessary groundtruth for the comparison of the results obtained by all the algorithms.


Figure 9.11: Comparison of the results for the “Shirt”garment. Sketchness on the left, Fashionista on the right.

The comparison has been done at first using as baseline the Background, as it has been done in the papers describing the algorithms and afterwards by comparing the pixel accuracy of the specific garment for which we had the groundtruth against the masks generated by the three algorithms (Fashionista, Paperdoll, Sketchness). The results are shown in Table 9.9

| **Algorithm** | **Mean (%)** | **Standard Deviation** |
|---|---|---|
| Fashionista | 81.79 | 13.55 |
| Paperdoll | 78.25 | 13.55 |
| Sketchness | 93.27 | 9.26 |

Table 9.9: Sketchness comparison results with Background as a Baseline

Sketchness is able to improve the current state of the art by 12% with a lower standard deviation. It is interesting to note that Paperdoll, which should be a more advanced version of Fashionista, performs worse than the previous version if the background is considered as a Baseline.

When considering the groundtruth, as shown in Table 9.10, and thus the accuracy pixel level of just the garment, Sketchness significatively outperforms

| Algorithm | Mean (%) | Standard Deviation |
|---|---|---|
| Fashionista | 26.78 | 31.49 |
| Paperdoll | 54.21 | 37.05 |
| Sketchness | 81.38 | 24.43 |

Table 9.10: Sketchness comparison results with Groundtruth as a Baseline

the state of the art by 27%. This shows the limitation of automatic segmentation for the garments when considering user generated content, for images in which the pose of the subjects is not completely defined (occlusions of part of the body) and when the image contains small items or accessories, hardly recognized by the pure algorithmic approaches.

### 9.5.2 Malicious Users Detection

The results on the reputation score that have been obtained so far for the players registered in the system are reported in Figure 9.12. To bootstrap the system and to perform the initial experiments, all the images that were fed into the game were annotated by two special users, deemed as Spammer, that is a user who has intentionally provided wrong annotation for all the tasks, and Groundtruth, that is a user who has intentionally provided the most precise annotations that a human could submit.

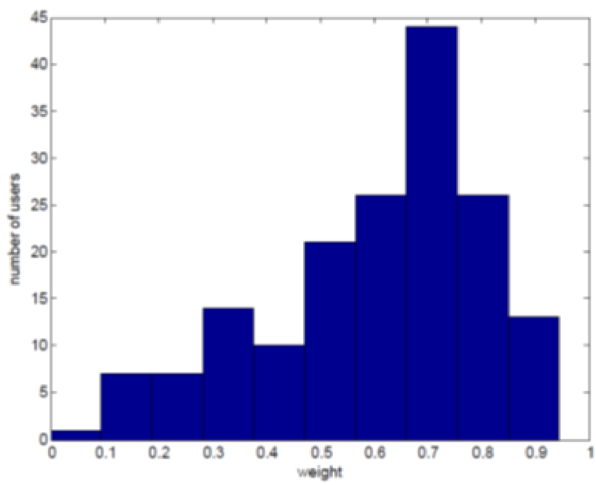



Figure 9.12: Reputation Score for the players of the game.

While there was no incentive for the players to submit good contours for the garments, beside the fact that the other users would have not been able to recognize their contribution, it can be seen that most of the users have a reputation score that settles at around 0.7. It is important to notice that from the players perspective, reaching the maximum accuracy of segmentation is not the optimal strategy, since the more accurate the segments provided, the more time consuming the operation is, the less the other players will have time to guess and that seldom the players will provide the exact same contour for the same garment.

It is also worth noticing that even though one of the players used to bootstrap the content was really accurate in its segmentations, no user has a reputation of 1. This is due to the fact that the reputation even for the "Groundtruth"player is compared against the annotations submitted by all the other players which, although similar, can by no mean be exactly the same. On the other hand, the algorithm has been able to identifying the presence of the Spammer user, giving him a reputation of 0, thus discarding all the contribution that he has made when generating the aggregated masks.

The reputation model for the player has been found invaluable during the aggregation phase with respect to the commonly used majority voting approach, as it can be seen in Figure 9.13 that illustrates the performance of the different aggregation strategies when varying the number n of games per image and settings all the other parameters to their default values. The proposed algorithm achieved TP@1%   80% with as few as n = 3 games per image, halving the number of rounds (and thus the number of players) required to obtain the same accuracy with respect to simple majority voting. It can also be noticed that the algorithm slightly outperforms also the case in which majority voting considers only the good annotations. This is due to the fact that the algorithm not only assigns low weight to cheaters but unequal weights even to good players, depending on their potentially different skills, resulting in a more accurate aggregated region of interest by reducing the effect of imprecise borders of the regions.

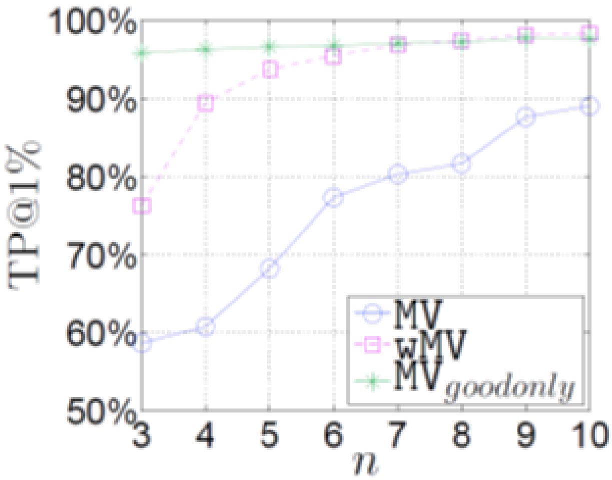



Figure 9.13: TP@1% vs. number of games per image n

Finally on Figure 9.14 is possible to see the resilience of the approaches to an increased percentage q of malicious users, seeing how our approach is able to improve the accuracy of the aggregated results by 10%.

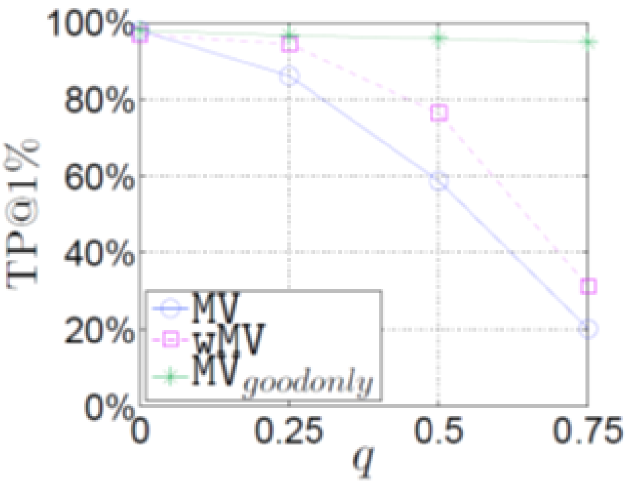



Figure 9.14: TP@1% vs. percentage of spammers q

### 9.5.3 Artificial Players contribution

Sketchness game mechanics rely on the participation of two or more players per game session to create a meaningful experience. However, two-player games present several logistical challenges.

- There may be times when an odd number of people want to play a particular game, meaning at least one of them cannot play.

- When a game is in its infancy, it is difficult for game administrators to guarantee that many people will be able to play at the same time.

For these reasons, it is necessary to give players the possibility to play the game on their own by introducing a single player feature implementing ad-hoc bot players. To design such artificial players, the two different roles they could interpret during the game must be taken into consideration, each one with its own peculiarities.

### Guesser Bot Design

If the current role for a player is the Guesser, he/she is asked to type his guesses in a provided text input box. The guesses are visible to the other players except when the player have guessed correctly. When a player has guessed correctly, his status will be changed (changing the name color in the list of the players) in order to show to everyone that he/she was able to answer correctly. The quicker the player answers correctly, the higher the score he/she will obtain. The guesser bot should be able to emulate the human reasoning process, applied during the guessing round.

To obtain a believable gameplay it is not sufficient to apply the solution exploited by most of GWAPs, which consists in using a set of actions recorded from an earlier game session, such as guesses of each partner along with timing information. The game system knows which is the correct guess, but this is not sufficient to meet two main requirements:

- The guesser bot should fake a human player, which usually tries different guesses before answering correctly.
- If a human player draws useless or wrong segmentations of the cloth, the bot should not guess correctly

These problems must be taken into account to protect the illusion that a real game is taking place. The solution could exploit the fact that the images used during the matches come from a familiar database, so some information about the images and the depicted cloths can be extracted. For example the current drawing position, in relation with the entire image content, is an important information to be exploited. In particular, the current drawing position can reveal the body part in which the cloth to be guessed is located.

The specific task of determining the pose of an object in an image is referred to as Pose Estimation. This requires additional information to be stored for each image, represented by the Pose attribute. The pose object is meant to store the coordinates of the bounding boxes related to the main body parts (head, torso, left arm, right arm, legs, feet). In this way it is possible to check the location of the current drawing point and classify it using the pose coordinates, to identify the current body part.

To summarize, the design of the simulated guessing process includes the following steps:

1. Retrieval of current drawing position.
2. Classification of the current drawing position in a body part.
3. Computation of the current area inside the segmentation traced by the user.
4. Computation of the ratio between the current area drawn and the total area of the body in the image.
5. Given the ratio of the areas, guess the most appropriate cloth in the identified body part.

This process is repeated every n seconds, until the guessing round terminates, either because of time expiration or because the correct guess has been provided by the bot procedure. The body part and the areas ratio are the two inputs of the last activity in the process, which is the final classification, carried out to get the most appropriated guess. In the following the two parallel branches which compute these inputs are analyzed.

**Body part classification** The first key element to intelligently guess cloth names is to retrieve in real-time the current drawing position of the human sketcher, to find out the body part in which the traces are drawn and guess the related possible cloths associated to that body area. For example, if the human player is drawing in the “feet”neighbourhood, possible guesses will be “shoes”or “socks”. The identification of the current drawing position is trivial with the current game mechanics.

Once the current drawing position has been retrieved, it must be classified to get the body part. To do that a new attribute related to the Image object has been introduced: the Pose field. This attribute is used to store the coordinates of the bounding boxes related to the body parts "head", "torso", "left arm", "right arm", "legs"and "feet".

Pose objects are available data which do not need to be computed in real-time. The pose can be retrieved at the beginning of the round, improving the performances of the gameplay. To solve the so called "Pose Estimation problem", which is the task of determining the pose of an object in an image, a part of the algorithm of Clothing Estimation described in [168] has been exploited. Instead of using articulated limb parts as in previous approaches, this new method proposes to use the capture orientation with a mixture of templates for each part.

The output of the pose estimation algorithm is a pose object containing the coordinates of the main body parts.

The bounding boxes which compose the final pose object are computed starting from these output coordinates. Six different body areas have been identified, labelled as: head, torso, left arm, right arm, legs and feet, as shown in figure 9.15

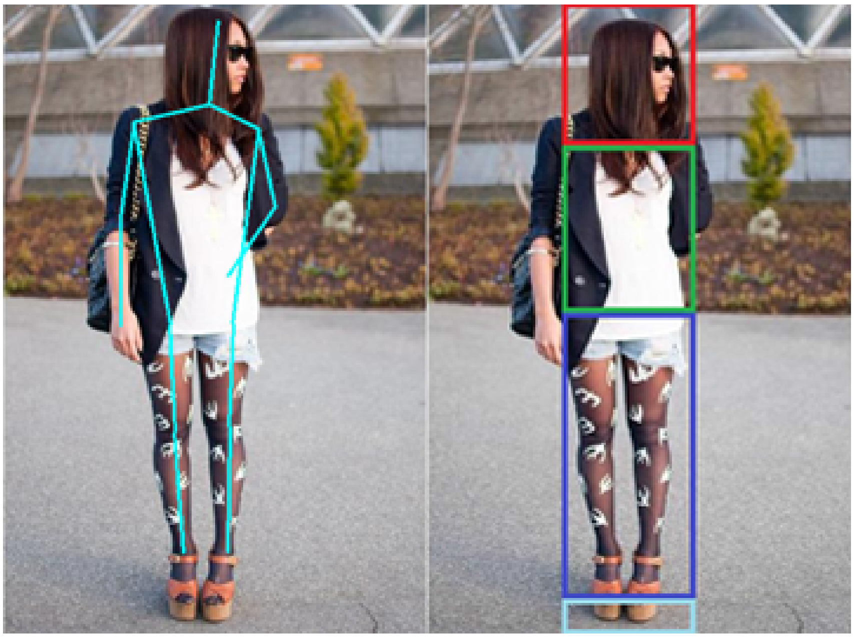


Figure 9.15: Results of the Pose Estimation Algorithm; Bounding Boxes extracted from the pose coordinates.

**Ratio Computation** The final result of this branch of the process is the ratio between two quantities:

- The area of the whole body in the image.
- The area inside the current segmentation traces.

The first value is computed at the beginning of the round and saved to avoid useless calculations during the match and to improve performances. Indeed the area of a body depicted in an image is a constant value that can be computed again exploiting the pose data. In particular the total area is approximated, using the minimum and maximum coordinates of the pose object in the x and y directions, with a rectangular shape.

The computation of the current area requires instead a little bit more of steps. The area is computed using a simple and fast algorithm, which is able to return the area of an irregular polygon, given the arrays of the x and y coordinates of the vertices, traced in a clockwise direction, starting at any vertex.

However there are circumstances where the algorithm will produce the wrong or unexpected results. They all have to do with self-crossing polygons where one side crosses over another side of the same polygon. It is not surprising that this happens, almost all the theorems about polygons also fail with self-intersecting polygons.

To solve this problem a pre-processing step, applied to the vertices coordinates, is required. It consists in identifying the intersections between polygon sides and the creation of separated polygons in order to avoid crossings.

The idea is that, when an intersection is identified, a new polygon is created using all the previous points and the identified intersection point. All the previous points will be removed from the list, while the intersection point will be the first point of the next polygon. In this way the total area is computed as the sum of all the computed polygons, without intersections, using the algorithm previously described. To find the self-crossing points in a polygon the BentleyOttmann algorithm [172] has been used, which is a sweep line algorithm for listing all crossings in a set of line segments.

We decided to use this algorithm because of its low complexity with respect to the nave algorithm that tests every pair of segments: for an input polygon consisting of n line segments with k crossings, the BentleyOttmann algorithm takes time $O((n+k)\log n)$.

**Final Classification** The core component of the guessing process is represented by the final classifier, which will return the guess cloth name, given as input the ratio between the current segmentation area and the total body area. The classifier is composed by five different models, one for each body part. The classifier has been built using the Fashionista Dataset of [167].

To build the classifier, the idea is to use the annotations contained in the Fashionista Dataset, including pose and cloths in images, to create new datasets that will be the input of a machine learning algorithm.

In particular a new dataset has been created, for each body part. The "right arm" and "left arm" categories have been merged in a unique class "arms", since it is not possible, and it has no meaning, to distinguish between them during the classification process.

The new datasets are composed by two attributes, a label identifying the cloth and a number identifying the ratio. To train and test the dataset the Weka workbench (Waikato Environment for Knowledge Analysis), a popular suite of machine learning software, has been used. Starting from the new dataset, training and testing datasets have been created by splitting the sample into a training set and an independent test set, where the former, composed by 75% of the source data is used to develop the classifier and the remaining 25% to evaluate its performance.

The idea is to build a model for each body part and then test the model to verify if the hypothesis of a correlation between the cloth label and its area with respect to the entire body is verified. To improve the performance of the classification algorithms, similar garments belonging to the same body part where grouped together, dividing the same part into several bins corresponding to closely related garments. This is due to the fact that similar cloths have the same dimension (and consequently ratio), and it is really difficult to distinguish among them.

However this is not a problem, since the differentiation of this kind of clothes is not required by the game: even a human player would be in trouble classifying these garments, and the bot should be as close to a human player as possible.

The classification performance over the dataset, tested by using several machine learning algorithms such as Multi Layer Perceptron, Naïve Bayes, Bayes Network, Decision Trees, Random Forest and Random Trees, has yield the results detailed in Table 9.11

| | **Head** | **Torso1** | **Torso2** | **Torso3** | **Arms** | **Legs** | **Feet1** | **Feet2** |
|---|---|---|---|---|---|---|---|---|
| MLP | 58,16% | 19,79% | 33,67% | 44,36% | 61,60% | 52,20% | 23,56% | 83,69% |
| Naïve Bayes | 61,13% | 28,28% | 35,33% | 36,92% | 62,40% | 54,41% | 23,88% | 82.09% |
| Bayes Net | 68,25% | 36,35% | 62,43% | 80,78% | 64,00% | 53,05% | 44,88% | 84,12% |
| J48 | 73,00% | 62,04% | 69,00% | 82,01% | 68,60% | 60,00% | 64,18% | 84,12% |
| Random Forest | 80,11% | 71,22% | 76,88% | 85,12% | 76,00% | 77,12% | 73,03% | 89,23% |
| Random Tree | 81,60% | 72,91% | 78,40% | 86,56% | 77,60% | 80,68% | 74,52% | |

Table 9.11: Classification Performance of machine learning algorithms over garments, divided in different body parts

As it is possible to see, Random Trees produce the best performance overall, strictly dominating all the other approaches. This result may be explained considering some of the features of Random Trees. Random trees do not suffer from overfitting over the training data nor they are sensible to outliers, which are considerable benefits given our dataset. Moreover they also computes proximities between pairs of cases, often used for clustering problems, obtained through the use of the Gini index, a measure of inequality between two classes. Since our dataset pairs the garment with a position and an associated area, Random trees are efficient at splitting dataset into similar clusters before classifying them, which is probably the reason for which the algorithm beats all the other machine learning algorithms.

Figure 9.16 shows a round involving the guesser bot. As the drawing proceeds, the bot guessed the garments labels “belt”, “shorts”, “jeans”, and finally the correct word “pants”.

**Sketcher Bot Design** If the current role for a player is the Sketcher, he/she will be provided with an image coming from a set of annotated or not annotated fashion images with low confidence. He/she will be the only player with the rights to see the image, while the image will be hidden to the other players. The sketcher is asked to provide a tag for a garment visible in the image, such as tie or trousers or he/she will be given a tag generated from previous matches. Once the tag has been added, the Sketcher is asked to draw the

Figure 9.16: Bot trying to guess the contours submitted by the player

selected object by tracing its contour. He/she will be able to see the guesses given by the other players and he/she will also be able to skip the drawing of the image object if he/she cannot/does not want to play for that particular image.

The sketcher bot should be able to emulate the human drawing capabilities. In this case the simulation is easier than the guesser bot, since actions recorded from an earlier game session can be used to reproduce the segmentation with believable results. The timing information is a key element of the pre-recorded data, in order to reproduce correctly a realistic human behaviour. Indeed during the segmentation performed by the user, points are sampled at a constant interval and stored together with the timing information. Reproducing human traced segmentation, also the quality of the draws must be taken into account. Different segmentation actions could be available for the same image, and only the best ones should be chosen to be reproduced by the sketcher bot. This requires an additional information, which is represented by the quality of a segmentation object: it can be for example a number between 0 and 1.

To simulate a round using pre-recorded data, some basic information about the current image must be known:

- The tag, which refers to the cloth object to be drawn.
- At least one segmentation action performed by a user in a previous ses-

sion.

For this reason during a match which involves the use of bots, the tagging task, in which the sketcher is asked to provide a tag for the current image, should be avoided. In this way only annotated images will be used.
Summarizing, there are four main features required for the integration of bots within the game:

1. The use of only annotated images: tags and segmentations must be available.
2. Elimination of the tagging task from the game logic when bots are involved.
3. The introduction of the Pose element associated to the image object.
4. The introduction of the segmentation quality related to a segmentation action.

**Sketchness Turing Like Tests** The Computer Game Bot Turing Test was designed to test a bot's ability to interact with a game environment in comparison with a human player. The idea is to evaluate how is it possible to make game bots, non-player characters controlled by AI algorithms, appear as human as possible.

To perform the Turing Test the game logic has been modified as following. The test consists of a single match with nine rounds. The game requires two players: the first one is the user which has been selected for testing, the second one is a fixed human player, which will play only some of the rounds of the game, while the remaining ones will be played by the bot. The user subjected to the test is not aware about which rounds are played by a human player and which instead are played by the bot. What happens is that the user will play a standard two players game, but, during the different game rounds, his/her opponent could be either another human player or a bot. At the end of each round participants are given few seconds to classify their opponent.

Human participants were of a moderate skill range, with no participant either ignorant to the game or capable of playing at a professional level. The idea of selecting a human player which will alternate with the bot comes from the fact that the test would be compromised if the user knows the level and the language skills of the other player. Without this caution, in most of the cases, the test would have been performed by two players which know each other. This would have not reflected the real game situation, where, when the user is paired with the bot, he/she should think to play with a randomly selected player. In order to perform the test, nine images have been selected carefully to include scenarios as various as possible, such as full body or partial body portrayed in the image, differentiating the tags to avoid duplicate garments. All the images have been processed by 30 different users that were facing either a bot or an expert fixed Sketchness player.

The results of the test, which have proven to be even statistically relevant with a confidence interval of 1%, are shown in Table 9.12

| | |
|---|---|
| Percentage of Correct Answers | 60,74% |
| Percentage of Incorrect Answers | 39,26% |

Table 9.12: Results on submitted responses for the Turing Like test

More detailed results are provided in Table 9.13

| | |
|---|---|
| Experimental Results on Human Players | 109 |
| True Positive (TP) | 89 |
| False Positive (FP) | 45 |
| True Negative (TN) | 75 |
| False Negative (FN) | 61 |
| True Positive Rate | 0,593 |
| False Positive Rate | 0,375 |
| True Negative Rate | 0,625 |
| False Negative Rate | 0,407 |

Table 9.13: Detailed results of the Turing Like Test

In 39,26% of cases, users were not able to distinguish correctly between human and bot players. Since the percentage success rate for the Turing Test is 33%, we can state that the test was successful, since more than 33% of users have confused bot for human and vice versa.

### 9.5.4 Qualitative Engagement Evaluation

To validate the usability and effectiveness of our GWAP from a qualitative point of view, it has been decided to draft the online questionnaire following Hassenzahl's model [173]. It assumes that two distinct attribute groups, *pragmatic* and *hedonic* attributes, can describe characteristic of an interactive product. Pragmatic attributes are connected to the users' need to achieve a particular goal, thus requiring utility and usability; a product that allows effective and efficient goal-achievement possesses perceived pragmatic qualities. In contrast, Hedonic attributes are related to the users' self; they involve stimulation, novelty and challenge for the user and identification as the need to express one's self through object. According to Hassenzahl,system design characteristics that make a design more usable should improve its users' experience in terms of Pragmatic quality and consequently Goodness (overall quality). Hedonic quality and consequently overall beauty should not be affected.

An adapted version of the AttracDiff2 questionnaire [174] was employed to measure perceived pragmatic quality (PQ) and perceived hedonic quality-stimulation (HQS). To measure these quality, thirteen-7-point items with bipolar verbal anchors (i.e., a semantic differential, see Table9.14 and Table **??**) were used. The PQ and HQS scores were calculated by averaging the respective item values per participant. A high PQ score primarily implies high usability, while a high HQS score implies a high degree of perceived novelty, stimulation and challenge. The questionnaire was then integrated with questions and considerations derived from the Game Experience Questionnaire [124] [125] that were also used during the iterative development of the game in order to drive the improvement on both the game and user interface design, but due to their verbose nature are difficult to measure in an objective way.

The questionnaire was published on the webpage of the game and was on a voluntary base, with no incentives, in order to elicit only unbiased and truthful responses. In the one-month period under scrutiny, 30 responses were obtained from players of different ages, including the target group (children between 8-10 years old) and grown ups. A brief summary of the results is provided in **??**. It is interesting to note that the quality values are higher for the target player group with respect to other users, and that young adults, probably the ones that has seen the growth of digital games since their infancy. For all the typologies

Table 9.14: Pragmatic Quality (PQ)

| Scale | Anchors |
|---|---|
| PQ_1 | Technical - Human |
| PQ_2 | Complicated - Simple |
| PQ_3 | Impractical - Practical |
| PQ_4 | Cumbersome - Direct |
| PQ_5 | Unpredictable - Predictable |
| PQ_6 | Confusing - Clear |
| PQ_7 | Unruly - Manageable |

Table 9.15: Hedonic qualitystimulation (HQS)

| Scale | Anchors |
|---|---|
| HQS_1 | Typical - Original |
| HQS_2 | Cautious - Courageous |
| HQS_3 | Conservative - Innovative |
| HQS_4 | Lame - Exciting |
| HQS_5 | Easy - Challenging |
| HQS_6 | Commonplace - New |

of users the quality is way above than the mean, showing that the GWAP has been considered an attractive experience. It is also worth noting that the pragmatic quality measure is higher than the hedonic quality-stimulation: even though it has been spent a considerable effort in improving the user interface of the game, the users that answered the test were more satisfied by its usability and simplicity than the visual appeal. For what concerns the results of the questionnaire that were outside of the scope of the AttracDiff test, Figure **??** shows the results of the questions that were designed following the guidelines defined in the GEQ. The chart shows that the vast majority of players found Sketchness both attractive and easy to be understood.

The results of the questionnaires, paired with the considerable number of contributions that the game was able to gather prove that pipelines that require input from a large number of users can be put in place without the need of relying on monetary incentives.

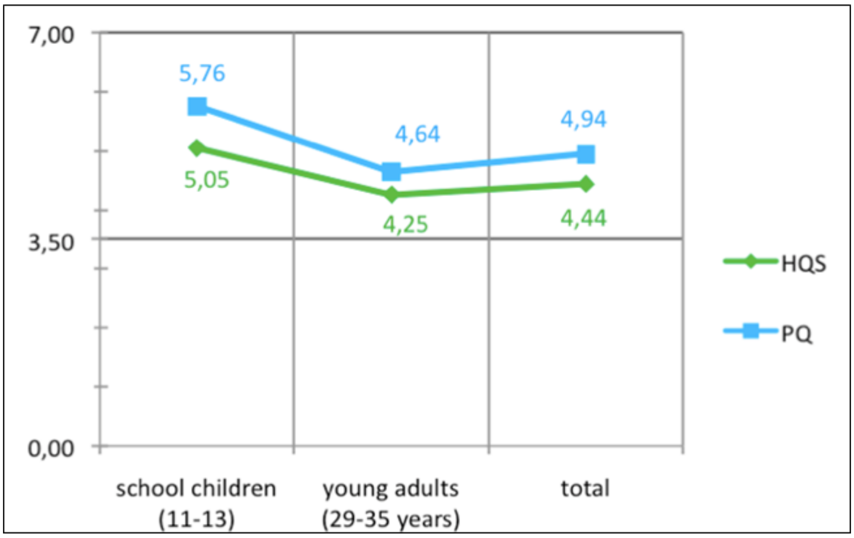



Figure 9.17: Pragmatic Quality and Hedonic Quality Stimulation as gathered from the questionnaires

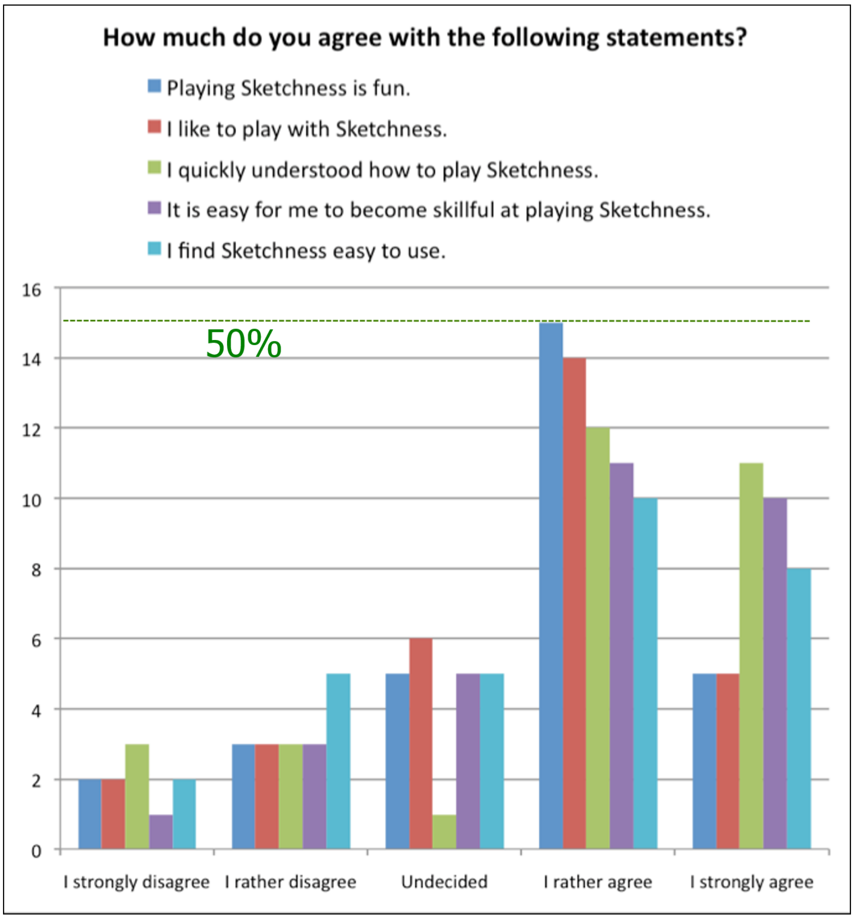



Figure 9.18: Game Experience like questionnaire results

## 9.6 Summary

In this chapter we described the implementation of a novel open GWAP for object segmentation, Sketchness, by following the design guidelines and methods that we have defined thorough this work, able to obtain satisfactory results from both a quantitatively and qualitatively point of view.

The results obtained through Sketchness have then finally compared with the most important works in literature, by conducting an extensive experimental campaign involving over 800 users employed to annotate a collection of challenging fashion images thanks to the submission of thousands of annotations. The game has been able to outperform state of the art algorithms in fashion image segmentation by 27% without any constraints on the target userbase employed.

Thanks to the novel aggregation strategies developed and put in place for this GWAP, the aggregation phase of the annotations has proven to be resilient to the presence in the system of malicious users and allowed to reach a target quality level by halving the number of required contributors.

To solve the "cold start" effect and making the game enjoyable even when few players are online, one of the first examples of Artificial Intelligence applied to GWAP has been integrated, drawing best practice on how to deal with the problem in a general way and capable of passing a "Turing-Like" test.

# Part V

# Conclusions

# Chapter 10

# Conclusions

From enterprise business applications to large scale human computation platforms, the lack of motivation is a multi-faceted problem that undermine the quality of the work of their users. Traditional incentives such as payments are often insufficient to guide a worker towards a goal, since she could be motivated by other factors, such as fun, curiosity, altruism or others. The intrinsic interest in the task itself may occur but it is, in general, a rare occasion.

In this dissertation we propose a possible solution to this problem by investigating the design and development process of Games with a Purpose (GWAP) and gamified applications; the former can be used to solve computational tasks that are out of the scope of traditional computational algorithms, while the latter can be used as a mean to increase participation and motivation of users in a platform to achieve target objectives.

With respect to Games with a Purpose, we faced and solved the issues deriving from the lack of guidelines to follow when designing a GWAP, from the design phase, to the architectural development till the application deployment and collection of the results. This has been possible with the definition of a development process for GWAP creation, by defining and identifying the human computation tasks to be solved, its conversion into meaningful game mechanics that could be adapted and integrated within a successful video game instantiation and the aggregation of the obtained results even in presence of bad contributors or cheaters.

Gamified applications were plagued by the same issues: the lacking of a process to follow, the confusion about the existing mechanics and their possible applications, the difficulties in verifying quantitatively the effectiveness of the introduction of gamified elements in a platform were solved by defining a detailed workflow for gamified applications design and implementation.

The initial results obtained in real world case studies in which the processes and techniques were applied demonstrate the feasibility of the application of the GWAP and Gamification paradigms, yet they are not without drawbacks.

Using a GWAP instead of a traditional Human Computation platform pose severe limitations on the computational tasks that could be solved, restricting their application mostly on multimedia refinement tasks for which we were able to find a matching game mechanic. It is very challenging to pose time constraints on the collection of the results, since a GWAP relies on the voluntary and intrinsic desire of the player to engage herself within the platform; in traditional human computation platforms, the process can usually be accelerated by paying more. The players participating in a GWAP are less prone to provide truthful responses and they often use approaches that guarantee the best result with minimum effort by cheating and gaming the system. On the other hand, once set in place, a GWAP could provide unlimited data almost for free, it could be distributed on more known and crowded platforms and even be used to solve tasks others than the ones for which they were originally planned, for instance to verify if the user that is interacting on a website is a bot or a human.

Gamification techniques could be applied to every domain for which an increase of participation and contribution from its users is requested, even in games, but could often mislead them into focusing on activities that are not meaningful for the problem at hand.

### 10.0.1 Summary of the work

In the following, we summarize the research work carried out in this thesis.

#### Games with a Purpose

The research process in the field of Games with a Purpose reported in this thesis was focused on the study of Games with a Purpose for multimedia content refinement. Prior to this work, the design of GWAP was performed in a ad-hoc way, using the designer's past experiences to create a game that could be used to solve a task.

No other work in literature considered a GWAP just as another application; even in literature about game design a formal development process for a videogame was not defined. We started our research from a traditional process model for web applications, while we created a reference model for the game design workflow by reverse engineering the best practices of prominent game designers. The two models have then been fused to yield to a development process model that could represent the production workflow of GWAP.

The analysis revealed that a GWAP is nothing more than a human computation application that is injected in an digital game; one of the problems that we had to face was then related to the formalization of an architectural and data model useful to describe and implement general purpose human computation applications.

Since in literature there was no universal data model or standard able to embrace all the facets of personal and social contribution of the users, along with features that were typical of videogames, the next step involved the creation of a model able to convey the profile features, social links and roles of the users, the characteristics of the content objects they produce and consume and the elementary actions and tasks performed in the context of interest.

In particular the definition of human computation tasks collided with the necessity of hiding them within the gameplay of the GWAP; finding the right mechanics to be applied in specific contexts was still a problem not solved, and fundamental for a GWAP designer in order to make the right choice in the design phase. We analyzed the mechanics available in traditional game design literature in order to find the most suitable ones that could have been applied to the computational tasks identified in the previous step.

These steps allowed us to create a solid background for the collection of annotations-centered tasks for human computation, but one of the great problems related to this field is represented by the adversarial behavior of some users, which do not execute the required tasks or feed the system with malicious data in order to maximize their rewards, to obtain real life advantages or just for the sake of it. Cheating to obtain the maximum scoring is also a frequent behavior among players.

For these reasons, the final steps of the GWAP design research was focused on trying to minimize the aforementioned problem with the identification of suitable validation templates that could avoid collusion and not meaningful annotations. Since validation templates alone were not enough, due to the fact that user submitted annotations are, in general, unreliable, we investigated aggregation techniques already used in literature and we designed a novel aggregation technique able to estimate the reliability of the contributor based on her past submissions. As an unexpected side effect of this technique, we were able not only to detect cheaters and spammers, but also to halve the number of contributions required to reach a target quality level in any GWAP.

To validate all the work, we applied the design process and the techniques described in this thesis in the development of a novel GWAP for fashion images segmentation as part of a fashion trend mining application of the CUbRIK European project[1]. This GWAP, called Sketchness, was also the first instantiation of the technical framework described in the architectural section of the thesis and allowed us to create a game engine for the development of GWAP for multimedia annotation. The game has been able to outperform state of the art algorithms in fashion image segmentation by over 27% without any constraints on the target userbase employed or conditions of the submitted image.

### Gamification

The research process in the field of Gamification focused on enterprise business applications scenarios. This was due to the fact that when there is the need for a gamified application, the focus is not on the entertainment level of the players but primarily in reaching a target objective and, in this sense, gamified applications are similar to GWAP. Business and functional requirements are preexisting to the gamification effort, since the business goals are supposed to be already defined in the design of the original application, thus the only missing part is the introduction of game mechanics able to drive participation of the users in order to reach the specified objectives; the approach is complementary to the one related to GWAP development, in which the starting point was an existing game in which to inject a computational task.

As it happened with GWAP, the systematic development of gamified applications was still an undefined process in literature, leaving designers that

[1] http://www.cubrikproject.eu/

needed to face the problem of applying gamification techniques to an enterprise application just with a couple of blurried “best practices” .

Starting from an already developed application, we outlined the development process of its gamified counterpart. The data model and architecture that we described for the GWAP was found to be suitable also for this particular class of problems. The game mechanics to be used in the gamification problem, on the other hand, were completely different with respect to the ones used in GWAP since the business process was already in place. Mechanics used to increase retention and engagement of the users of a system were thus studied and expanded by borrowing concepts also from traditional game design literature, describing advantages, drawbacks and possible application scenarios for each of them.

Metrics to evaluate quantitatively the benefits of having introduced a Gamified approach in a traditional application were then explained to provide the necessary tools to compare one gamified approach against the other.

To validate all the work, we applied the design process and the techniques described in this thesis in a real world enterprise business case with the creation of Webratio Headquarters. The gamification framework derived from the development process was used to collect and unify all the actions performed by the users in several heterogeneous components that were already in place.

The objective ranged from increasing the participation of novel and existing users to improve the company's image towards new clients, to create a self-sustainable user centered support service to increase the knowledge about the use of the platform.

The data collected from the gamified platform were analyzed in order to assess the feasibility of the gamified approach, a process that in literature have been seldom done in a statistically relevant manner. The results of the study revealed that the gamified platform was able to improve all the performance indicators meaningful for the company, for instance by increasing the number of users registrations by 105% and the number of e-learning articles read by over 10000%.

### 10.0.2 Contributions

The most significative contributions of this work are:

- The definition of development processes for Games with a Purpose and Gamified applications. By following these processes, future designers can have a broader vision of the steps necessary to implement such applications, avoiding common pitfalls by relying on tools and techniques that have been proven effective in dealing with multimedia processing tasks and engagement/retention strategies used to drive users' behavior. Game mechanics to task matching guidelines have been proposed, along with aggregation and evaluation strategies for both kind of applications.

- The definition of a comprehensive model and software architecture able to cater for all the characteristic of human performers in human computation, GWAP and gamified applications. The tasks that can be performed in such platforms have been modeled and described, along with the data types on which they operate; techniques and structures necessary to handle conflicts, orchestrate and assign units of work to the users have also been detailed.

- The formal description of a general purpose aggregation strategy for human computation applications, agnostic with respect to the annotation type. Tests on both synthetic and real datasets deriving from our GWAP proved that the algorithm always outperforms state of the art aggregation techniques, is able to reduce by up to a half the number of contributors to reach a target quality level and is able to automatically assign a reputation score to the users based on the quality of their submitted contributions, thus excluding spammers and malicious users from the system.

- Sketchness, a multiplayer GWAP for fashion images segmentation. The game has been integrated into an automated pipeline for the automatic analysis of fashion trends to substitute automatic algorithms for images that were hard to process due to lighting condition or body occlusion. The game was able to outperform state of the art automatic algorithms for fashion images segmentation, has proven to be resilient to malicious users in the system, was able to gather a considerable number of contributors with respect to other works in literature and considered to be enjoyable from the qualitative tests we have performed.

- A game engine for the development of novel HTML5 GWAP. Most of the works in literature have not shared the technical infrastructure on which their applications were built upon. Our GWAP, Sketchness, has been developed with a custom open source multiplayer game engine that has been based on the data and architectural considerations presented as part of this thesis work; it is released as open source software as part of the CUbRIK Project[2]. It offers a backend for defining and storing human computation tasks for audio, images and textual annotations. It also manages the players and all the actions performed by them during gameplay sessions. The modular frontend components that have been created ease the development of GWAP by offering support to the most common tasks in human computation such as tagging, drawing and textual annotations submission and allows the implementation of the most common validation templates found in literature, by offering shared canvases and a chat service.

- Webratio Headquarters, a gamification platform that has been integrated in a real world business application, Webratio. Working closely with the company, the gamification techniques described in this work have been implemented by the company and integrated in their application. The resulting gamification platform has been used to empower their online community and is now sold as a service which is able to generate fully customizable gamified communities, both in terms of game mechanics and visual appearance. Data collected over several months of usage with the target users were able to show the benefits of introducing gamified techniques by providing statistically relevant results, whereas quantitative evaluations of gamified approaches in enterprise scenarios are scarce in literature.

[2]http://sourceforge.net/projects/cubrik/

### 10.0.3 Ongoing and Future Work

In order for GWAP and Gamified applications to reach their full potential, it is necessary to make them closer and closer to traditional commercial digital games and have a better understanding of what makes them fun, engaging and successful. These two kind of applications are expensive and challenging to develop with respect to traditional serious games, and this work has provided just the basis to make their production less risky. Nonetheless, there are still several interesting considerations that we are leaving as future direction of research:

- During the designing phase of a game, an important aspect to take into consideration is the structure of interaction between a player, the game system and any other players. While most of the GWAP developed so far fall into the pattern "multilateral collaboration" and very few in the "single player vs game" case, several other possibilities of player interactions could be applied to make the game more compelling or open new design paths. GWAP are usually pushing the players in adopting cooperation towards the purpose of the game, but how the rewards schemes and structure of the GWAP would change if we were using a competitive interaction pattern? Would it affect the long term involvement of the players? Would it increase or decrease the quality of the submitted contributions? Would different interaction patterns open different strategies for the players to solve a particular task? Further research on the matter is required and could open new exciting possibilities for GWAP development.

- As the types of interaction among the users and the system grow to reach the full spectrum of possibilities offered by traditional game design techniques, so the validation templates used to promote the submission of good contribution should be adapted to the new scenarios, in order to cope with the different gameplay styles. Classical validation templates such as input-aggregation, output-aggregation and inversion problem mechanics could also become unfeasible, leaving the system unharmed against malicious users or collusion attempts. Further investigation on this field could bring more efficient safeguards techniques that could also be applied retroactively to old GWAP. In such a case, comparison measures to estimate the quality of a validation template should be introduced, whereas nowadays in most GWAP a validation template is a forced choice for a specific problem to solve.

- The GWAP that have been created so far, with OntoGalaxy as the only exception, have always been abstract puzzle games. We know from game design literatures that a variety of different genres, ranging from First Person Shooters (FPS) to Real Time Strategy (RTS) games, exists. The introduction of these new genres could allow us to apply an even more diversified spectrum of game mechanics that would have been otherwise unfeasible in classic GWAP. As a consequence, new computational tasks that would have been impractical with a traditional approach could also be introduced. A 3D FPS is a prime candidate, for instance, to let a player solve tasks in the real world through the use of a robot. Different game genres would also increase the level of immersion with the introduction of background storylines, worlds to explore and choices to make, options that are not inherently related to the task to be solved; this could make the game appealing to a broader userbase, possibly also increasing its longevity.

- Gamification has so far been driven by the use of mechanics and tutorials that were guiding the users towards the desired business objectives or hinting them to perform specific actions within the system. We think that Artificial Intelligence, applied extensively to traditional digital games to increase the engagement of users in single player modes, could prove to be a promising and underestimated approach to increase the interactivity of gamified applications. Virtual opponents tailored over the profile of a specific user may offer challenges related to activities that she consider more interesting; this could increase considerably the level of interaction and participation of the player while providing a tool for the company to differentiate the expected contribution of a user within the system. We plan to apply this idea to gamify a novel betting web-portal, but the possibilities of applying artificial intelligence to gamified applications are endless.

With respect to the ongoing work, the development processes and tools derived from this thesis are now being applied and validated to the problem of crowdsourcing meaningful contributions for environmental sustainability.

- The “Comoleonti project” part of the Proactive Framework [3], is a gamified serious game that aims at collecting geospatial data related to the location and movements of kids to tailor the public transportation services over their needs, while improving the company’s image. Children are requested to submit the data and their commonly used transportation mean as part of a set of activities promoted in a gamified platform that offers also several mini educational games. The effort of the most virtuous classroom and school is rewarded with free school-trips offered by the public transportation company.

Figure 10.1: The Comoleonti Project

[3]http://www.fp7-proactive.eu/class/proactive-project

- Drop!, part of the SmartH2O project [4], is a collection of persuasive games and gamified applications that are used to change water consumers behaviour, both in terms of knowledge of the implications of water production and consumption and in their actual daily practices. The digital game is used as an extension of a boardgame to collect water consumption habits and data related to a particular household through a series of trivia questions. The points collected in the game are also saved in a gamified water bill connected to the portal of the water utility used by the family. The gamified portal, created with Webratio Headquarters, allows the users to have a clear understanding of how their actions and habits influence their water savings. Competitions among neighbors and rewards schemes are used as a mean to maintain sustainable habits over a long period of time.


Figure 10.2: The Drop! game suite

[4] http://www.smarth2o-fp7.eu/

# Bibliography


[1] H. Tibbets. *Is Everything I Know About Crowdsourcing Wrong?* 2011. URL: http://soa.sys-con.com/node/1996041.

[2] VentureOutsource. *VentureOutsource business terms and definitions.* 2007. URL: https://www.ventureoutsource.com/contract-manufacturing/information-center/terms-and-definitions/3/.

[3] J. Howe. “The rise of crowdsourcing”. In: *Wired magazine* 14.6 (2006), pp. 1–4.

[4] J. Howe. *Crowdsourcing: Why the Power of the Crowd Is Driving the Future of Business*. 1st ed. New York, NY, USA: Crown Publishing Group, 2008.

[5] A. S. Blinder. “Offshoring: The Next Industrial Revolution?” In: *Foreign Affairs* 85.2 (2006), pp. 113–128. URL: http://www.internationaltraderelations.com/Blinder.Offshoring(ForeignAffairs,March-April2006).pdf.

[6] J. Horton and L. B. Chilton. “The Labor Economics of Paid Crowdsourcing”. In: *Proceedings of the 11th ACM Conference on Electronic Commerce*. EC ’10. Cambridge, Massachusetts, USA: ACM, 2010, pp. 209–218.

[7] D. Chandler and A. Kapelner. “Breaking monotony with meaning: Motivation in crowdsourcing markets”. In: *Journal of Economic Behavior & Organization* 90.0 (2013), pp. 123 –133.

[8] A. J. Quinn and B. B. Bederson. “Human computation: a survey and taxonomy of a growing field”. In: *29th International Conference on Human Factors in Computing Systems*. 2011.

[9] D. P. Anderson, J. Cobb, et al. "SETI at Home: An Experiment in Public-resource Computing". In: *Commun. ACM* 45.11 (Nov. 2002), pp. 56–61.

[10] D.A. Grier. *When Computers Were Human*. Princeton University Press, 2005.

[11] D.G. Stork. "Character and document research in the Open Mind Initiative". In: *Document Analysis and Recognition, 1999. ICDAR '99. Proceedings of the Fifth International Conference on*. 1999, pp. 1–12.

[12] L. Von Ahn, M. Blum, et al. "CAPTCHA: Using Hard AI Problems for Security". In: *Proceedings of the 22Nd International Conference on Theory and Applications of Cryptographic Techniques*. EUROCRYPT'03. Warsaw, Poland: Springer-Verlag, 2003, pp. 294–311.

[13] T. Gruber. "Collective knowledge systems: Where the social web meets the semantic web". In: *Web semantics: science, services and agents on the World Wide Web* 6.1 (2008), pp. 4–13.

[14] T. W Malone, R. Laubacher, et al. "Harnessing crowds: Mapping the genome of collective intelligence". In: *MIT Sloan Research Paper No. 4732-09* (2009).

[15] Nielsen. *What Americans do Online*. 2010. URL: `http://www.nielsen.com/us/en/insights/news/2010/what-americans-do-online-social-media-and-games-dominate-activity.html`.

[16] Newzoo. *2014 Global Games Market*. 2014. URL: `http://www.newzoo.com/insights/global-games-market-will-reach-102-9-billion-2017-2/`.

[17] L. Von Ahn and L. Dabbish. "Labeling Images with a Computer Game". In: *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. CHI '04. Vienna, Austria: ACM, 2004, pp. 319–326.

[18] F. Khatib, F. DiMaio, et al. "Crystal structure of a monomeric retroviral protease solved by protein folding game players". In: *Nature Structural & Molecular Biology* 18.10 (October 2011), pp. 1175–1177.

[19] L. Terry, V. Roitch, et al. “Harnessing Human Computation Cycles for the FPGA Placement Problem.” In: *ERSA*. Ed. by Toomas P. Plaks. CSREA Press, Nov. 3, 2009, pp. 188–194.

[20] E. Law, L. Von Ahn, et al. “Tagatune: A Game for Music and Sound Annotation”. In: *International Conference on Music Information Retrieval (ISMIR’07)*. Vienna, Austria, 2007, pp. 361–364.

[21] V. L. Ahn, M. Kedia, et al. “Verbosity: a game for collecting common-sense facts”. In: *Proceedings of the SIGCHI conference on Human Factors in computing systems*. ACM. 2006, pp. 75–78.

[22] J. O. Talton, D. Gibson, et al. “Exploratory Modeling with Collaborative Design Spaces”. In: *ACM SIGGRAPH Asia 2009 Papers*. SIGGRAPH Asia ’09. Yokohama, Japan: ACM, 2009, 167:1–167:10.

[23] S. Matyas, C. Matyas, et al. “Designing Location-based Mobile Games with a Purpose: Collecting Geospatial Data with CityExplorer”. In: *Proceedings of the 2008 International Conference on Advances in Computer Entertainment Technology*. ACE ’08. Yokohama, Japan: ACM, 2008, pp. 244–247.

[24] *Survey: Do Office-Based Employees Want Digital Engagement Programs?* URL: `http://technologyadvice.com/gamification/blog/survey-office-based-employees-want-digital-engagement-programs/`.

[25] S. Deterding, D. Dixon, et al. “From game design elements to gamefulness: defining ”gamification””. In: *Proceedings of the 15th International Academic MindTrek Conference: Envisioning Future Media Environments*. Tampere, Finland, 2011.

[26] *Tech Trends 2014 Elevate IT for Digital Business*. URL: `http://www2.deloitte.com/content/dam/Deloitte/us/Documents/technology/us-cons-tech-trends-2014.pdf`.

[27] *Gamification in 2012 Market Update Consumer and Enterprise Market Trends*. URL: `http://www.slideshare.net/wandameloni/gamification-in-2012-trends-in-consumer-and-enterprise-markets-13453048`.

[28] J. Antin and E. F. Churchill. "Badges in Social Media: A Social Psychological Perspective". In: *CHI 2011, ACM, Vancouver, BC.* 2011.

[29] Amazon Research Lab. *Amazon's Mechanical Turk 2012.*

[30] Kathleen T. "GWAPs: Games with a Problem". In: *Foundations of Digital Games 2014.* 2014.

[31] A. Eustace. *A fall spring-clean.* URL: `http://googleblog.blogspot.it/2011/09/fall-spring-clean.html`.

[32] Nielsen. *What americans do online: Social media and games dominate activity.* Tech. rep. 2011.

[33] L. Von Ahn. "Human computation". In: *CIVR* (2009).

[34] A. Parameswaran, A. D. Sarma, et al. *Human assisted graph search: It's okay to ask questions.* Tech. rep. Palo Alto, CA: Stanford University, 2010.

[35] X. Hu, M. Stalnacke, et al. "A mobile game to collect and improve position of images". In: *3rd International Conference on Next Generation Mobile Applications, Services and Technologies.* 2009.

[36] J. Howe. "The rise of crowdsourcing". In: *Wired magazine* 14.6 (2006), pp. 1–4.

[37] D. C. Parkes K. Z. Gajos Y. Chen E. Huang H. Zhang. "Toward automatic task design: A progress report". In: *ACM SIGKDD workshop on Human Computation.* 2010, pp. 77–85.

[38] P. Ipeirotis. *The new demographics of Mechanical Turk.* URL: `http://www.behind-the-enemy-lines.com/2010/03/new-demographics-of-mechanical-turk.html`.

[39] *ICT Facts and Figures 2013.* URL: `http://www.itu.int/en/ITU-D/Statistics/Documents/facts/ICTFactsFigures2013-e.pdf`.

[40] T. Abdelzaher, Y. Anokwa, et al. "Mobiscopes for human spaces". In: *IEEE Pervasive Computing* 6 (2007), pp. 20–29.

[41] A. T. Campbell, S. B. Eisenman, et al. "The rise of peoplecentric sensing". In: *IEEE Internet Computing* 12.4 (2008), pp. 12–21.

[42] P. Dutta, P. M. Aoki, et al. "Common Sense: Participatory urban sensing using a network of handheld air quality monitors". In: *7th ACM Conference on Embedded Networked Sensor Systems*. 2009.

[43] O. Aulov and M. Halem. "Assimilation of Realtime Satellite and Human Sensor Networks for modeling Natural Disasters". In: *AGU Fall Meeting Abstracts*. 2011.

[44] C. Manasseh, K. Ahern, et al. "The connected traveler: using location and personalization on mobile devices to improve transportation". In: *2nd International Workshop on Location and the Web (LOCWEB09)*. 2009, pp. 1–4.

[45] T. Sakaki, M. Okazaki, et al. "Earthquake shakes Twitter users: realtime event detection by social sensorsEarthquake shakes Twitter users: realtime event detection by social sensors". In: *19th International Conference on World Wide Web (WWW10))*. 2010, pp. 851–860.

[46] O. J. Cacho, D. Spring, et al. "Allocating surveillance effort in the management of invasive species: A spatiallyexplicit model". In: *Environmental Modelling & Software* 25.4 (2011), pp. 444–454.

[47] N. Maisonneuve, M. Stevens, et al. "Citizen noise pollution monitoring". In: *10th Annual International Conference on Digital Government Research: Social Networks: Making Connections between Citizens, Data 10th Annual International Conference on Digital Government Research: Social Networks: Making Connections between Citizens, Data and Government*. 2009, pp. 96–103.

[48] S. Kim, C. Robson, et al. "Creek watch: pairing usefulness and usability for successful citizen science". In: *29th International Conference on Human Factors in Computing Systems*. 2011, pp. 2125–2134.

[49] Y. Liu, P. Piyawongwisal, et al. "Going beyond citizen Going beyond citizen data collection with Mapster: A Mobile+Cloud citizen science experiment". In: *IEEE Workshop on Computing for Citizen Science (eScience 2001)*. 2011.

[50] G. Pickard, I. Rahwan, et al. “Time critical social mobilization: The darpa network challenge winning strategy”. In: (2010).

[51] F. Sebastiani. “Machine learning in automated text categorization”. In: *ACM computing surveys (CSUR)* 34.1 (2002), pp. 1–47.

[52] M. Hamilton, F. Salim, et al. “Transafe: a crowdsourced mobile platform for crime and safety perception management”. In: *ACM SIGCAS Computers and Society* 41.2 (2011), pp. 32–37.

[53] J. Stothard, J. SousaFigueiredo, et al. “Investigating the spatial microepidemiology of diseases within a pointprevalence sample: a field applicable method for rapid mapping of households using lowcost gpsdataloggers”. In: *Transactions of the Royal Society of Tropical Medicine and Hygiene* (2011).

[54] E. Horvitz and T. Paek. “Complementary computing: Policies for transferring callers from dialog systems to human receptionists”. In: *User Modeling and User Adapted Interaction* (2007).

[55] L. Von Ahn and L. Dabbish. “General techniques for designing games with a purpose”. In: *CACM* (2008), pp. 58–67.

[56] T. Fullerton, C. Swain, et al. *Game Design Workshop: A playcentric approach to creating innovative games.* Morgan Kauffmann, 2008.

[57] A. J. Stapleton. “Serious games: Serious opportunities”. In: *Australian Game Developers Conference, Academic Summit, Melbourne*. 2004.

[58] S. De Freitas. “Using games and simulations for supporting learning”. In: *Learning, media and technology* 31.4 (2006), pp. 343–358.

[59] E. Clua, B. Feijó, et al. “Game and Interactivity in Computer Science educationVideo Presentations Are Available on the Citation Page”. In: *ACM SIGGRAPH 2006 Educators Program.* SIGGRAPH ’06. Boston, Massachusetts: ACM, 2006.

[60] D. R. Michael and S. L. Chen. *Serious Games: Games That Educate, Train, and Inform.* Muska & Lipman/Premier-Trade, 2005.

[61] V. Narayanasamy, K. W. Wong, et al. “Distinguishing Games and Simulation Games from Simulators”. In: *Comput. Entertain.* 4.2 (Apr. 2006).

[62] M. Hogan, H. Sabri, et al. "Interactive Community Simulation Environment for Community Health Nursing". In: *Proceedings of the 2007 Conference on Future Play*. Future Play '07. Toronto, Canada: ACM, 2007, pp. 237–240.

[63] M. Ma, L. C. Jain, et al. *Virtual, Augmented Reality and Serious Games for Healthcare 1*. Springer Publishing Company, Incorporated, 2014.

[64] B. Sawyer and D. Rejeski. *Serious Games Summit*. URL: www.seriousgamessummit.com/.

[65] T. Greer. "2013-2018 North America Mobile Edugame Market". In: *Ambient Insight Targeted Research Report* (2013), pp. 643–647.

[66] L. Von Ahn and L. Dabbish. "Designing games with a purpose". In: *Commun. ACM* 51 (2008), pp. 58–67.

[67] L. Von Ahn and L. Dabbish. *Labeling images with a computer game*. Ed. by E. Dykstra Erickson and M. Tscheligi. ACM, 2004.

[68] S. Cooper, A. Treuille, et al. "The Challenge of Designing Scientific Discovery Games". In: *Proceedings of the Fifth International Conference on the Foundations of Digital Games*. FDG '10. Monterey, California: ACM, 2010, pp. 40–47.

[69] L. Von Ahn and L. Dabbish. "Designing games with a purpose". In: *Commun.ACM* 51.8 (2008), pp. 58–67.

[70] Markus Krause and Jan Smeddnick. "Human Computation Games: a Survey". In: *EUSIPCO'11 Proceedings of the 19th European Signal Processing Conference*. 2011, p. 6.

[71] A. Lodi, S. Martello, et al. "Two-dimensional packing problems: A survey". In: *European Journal of Operational Research* 141.2 (2002), pp. 241 –252.

[72] J. R. Corney, I. Kowalska, et al. "Crowdsourcing human problem solving strategy". In: *CrowdConf10 Proceedings of the 1st International Conference on Crowdsourcing. San Francisco, CA, USA*. 2010.

[73] H. S. Seung et al. "Eyewire". In: *Available at eyewire. org* (2012).

[74] A. Kawrykow, G. Roumanis, et al. “Phylo: A Citizen Science Approach for Improving Multiple Sequence Alignment”. In: *PLoS ONE* 7.3 (Mar. 2012).

[75] J. Secretan, N. Beato, et al. “Picbreeder: Evolving Pictures Collaboratively Online”. In: *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. CHI ’08. Florence, Italy: ACM, 2008, pp. 1759–1768.

[76] E. Law and L. Von Ahn. “Input-agreement: A New Mechanism for Collecting Data Using Human Computation Games”. In: *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. CHI ’09. Boston, MA, USA: ACM, 2009, pp. 1197–1206.

[77] L. Von Ahn, L. Ruoran, et al. “Peekaboom: A Game for Locating Objects in Images”. In: *In ACM CHI*. ACM Press, 2006, pp. 55–64.

[78] M. Krause, A. Takhtamysheva, et al. “Frontiers of a Paradigm: Exploring Human Computation with Digital Games”. In: *Proceedings of the ACM SIGKDD Workshop on Human Computation*. HCOMP ’10. Washington DC: ACM, 2010, pp. 22–25.

[79] A. Carlier, O. Marques, et al. “Ask’nSeek: A New Game for Object Detection and Labeling”. In: *Proceedings of the 12th International Conference on Computer Vision - Volume Part I*. ECCV’12. Florence, Italy: Springer-Verlag, 2012, pp. 249–258.

[80] K. Werbach and D. Hunter. *For the Win: How Game Thinking Can Revolutionize Your Business*. Wharton Digital Press, 2012. URL: `http://books.google.it/books?id=abg0SnK3XdMC`.

[81] R.A. Bartle. *Designing Virtual Worlds*. New Riders Games Series. New Riders, 2004.

[82] B. W. Boehm. “A Spiral Model of Software Development and Enhancement”. In: *IEEE Computer* 21.5 (1988), pp. 61–72.

[83] S. Ceri, P. Fraternali, et al. *Designing data-intensive Web applications*. Morgan Kaufmann Publisher, 2003.

[84] C. Crawford. *The Art of Computer Game Design*. Washington State University Vancouver, 1982.

[85] A. Rollings and D. Morris. *Game Architecture and Design.* New Riders, 2004.

[86] D. E. Knuth. *The Art of Computer Programming, Volume I: Fundamental Algorithms, 2nd Edition.* Addison-Wesley, 1973.

[87] R. Kowalski. "Algorithm = logic + control". In: *Commun. ACM* 22 (1979).

[88] J. G. Kemeny. "Mathematics without Numbers". In: *Daedalus* 88 (1959).

[89] J. Levin and B. Nalebuff. "An Introduction to Vote-Counting Schemes". In: *Journal of Economic Perspectives* 9.1 (1995), pp. 3–26.

[90] *Boardgame Geek.* URL: `http://boardgamegeek.com/`.

[91] S. Cooper, F. Khatib, et al. "Predicting protein structures with a multiplayer online game". In: *Nature* 466.7307 (2010), pp. 756–760.

[92] S. Bjork and J. Holopainen. *Patterns in Game Design.* Charles River Media game development series. Charles River Media, 2005.

[93] E. Law and L. Von Ahn. *Human Computation.* Synthesis Lectures on Artificial Intelligence and Machine Learning Series. Morgan & Claypool, 2011.

[94] T. Wang G.and Wang, Haitao Zheng, et al. "Man vs. machine: Practical adversarial detection of malicious crowdsourcing workers". In: *23rd USENIX Security Symposium, USENIX Association, CA.* 2014.

[95] L. Galli, P. Fraternali, et al. "A Draw-and-Guess Game to Segment Images". In: *Privacy, Security, Risk and Trust (PASSAT), 2012 International Conference on and 2012 International Confernece on Social Computing (SocialCom).* 2012, pp. 914–917.

[96] E. Law and L. Von Ahn. "Input-agreement: a new mechanism for collecting data using human computation games". In: *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems.* ACM. 2009, pp. 1197–1206.

[97] N. Quoc Viet Hung, N. Tam, et al. “An Evaluation of Aggregation Techniques in Crowdsourcing”. English. In: *Web Information Systems Engineering WISE 2013*. Ed. by Xuemin Lin, Yannis Manolopoulos, et al. Vol. 8181. Lecture Notes in Computer Science. Springer Berlin Heidelberg, 2013, pp. 1–15.

[98] V. S. Sheng, F. Provost, et al. “Get Another Label? Improving Data Quality and Data Mining Using Multiple, Noisy Labelers”. In: *Proceedings of the 14th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. KDD ’08. Las Vegas, Nevada, USA: ACM, 2008, pp. 614–622.

[99] O. Yuki, K. Teruaki, et al. “A Preliminary Study of the Number of Votes under Majority Rule in Crowdsourcing”. In: *Procedia Computer Science* 22.0 (2013). 17th International Conference in Knowledge Based and Intelligent Information and Engineering Systems, pp. 537 –543.

[100] T. Wei-Tek, W. Wenjun, et al. “Cloud-Based Software Crowdsourcing”. In: *Internet Computing, IEEE* 18.3 (2014), pp. 78–83.

[101] K. Lee, J. Caverlee, et al. “The Social Honeypot Project: Protecting Online Communities from Spammers”. In: *Proceedings of the 19th International Conference on World Wide Web*. WWW ’10. Raleigh, North Carolina, USA: ACM, 2010, pp. 1139–1140.

[102] M. D. Smucker. “Crowdsourcing with a crowd of one and other TREC 2011 crowdsourcing and web track experiments”. In: *In Proceedings of the Text REtrieval Conference (TREC*. 2011.

[103] R. Snow, B. O’Connor, et al. “Cheap and Fast—but is It Good?: Evaluating Non-expert Annotations for Natural Language Tasks”. In: *Proceedings of the Conference on Empirical Methods in Natural Language Processing*. EMNLP ’08. Honolulu, Hawaii: Association for Computational Linguistics, 2008, pp. 254–263.

[104] E. Şeyda, H. Haym, et al. “Approximating the Wisdom of the Crowd”. In: *Proceedings of the Second Workshop on Computational Social Science and the Wisdom of Crowds (NIPS 2011)*. 2011.

[105] A. P. Dempster, N. M. Laird, et al. “Maximum likelihood from incomplete data via the EM algorithm”. In: *Journal of The Royal Statistical Society, Series B* 39.1 (1977), pp. 1–38.

[106] J. Whitehill, P. Ruvolo, et al. “Whose Vote Should Count More: Optimal Integration of Labels from Labelers of Unknown Expertise”. In: *Advances in Neural Information Processing Systems 22*. Ed. by Y. Bengio, D. Schuurmans, et al. 2009, 2035–2043.

[107] J. Vuurens, A. P. De Vries, et al. “How Much Spam Can You Take? An Analysis of Crowdsourcing Results to Increase Accuracy”. In: *Proceedings of the ACM SIGIR 2011 Workshop on Crowdsourcing for Information Retrieval (CIR 2011)*. Beijing, China, 2011, pp. 48–55.

[108] V. C. Raykar, S. Yu, et al. “Learning From Crowds”. In: *J. Mach. Learn. Res.* 11 (Aug. 2010), pp. 1297–1322.

[109] P. G. Ipeirotis, F. Provost, et al. “Quality Management on Amazon Mechanical Turk”. In: *Proceedings of the ACM SIGKDD Workshop on Human Computation*. HCOMP ’10. Washington DC: ACM, 2010, pp. 64–67.

[110] D. R. Karger, S. Oh, et al. *Iterative Learning for Reliable Crowdsourcing Systems.*

[111] D. R. Karger, S. Oh, et al. “Budget-Optimal Task Allocation for Reliable Crowdsourcing Systems”. In: *CoRR* abs/1110.3564 (2011).

[112] C. Bernaschina, P. Fraternali, et al. “Robust Aggregation of GWAP Tracks for Local Image Annotation”. In: *Proceedings of International Conference on Multimedia Retrieval*. ICMR ’14. Glasgow, United Kingdom: ACM, 2014, 403:403–403:406.

[113] P. Jaccard. *Lois de distribution florale dans la zone alpine*. Bulletin de la Société Vaudoise des Sciences Naturelles. 1902. URL: `http://books.google.it/books?id=AeRAlgEACAAJ`.

[114] P. Jaccard. “The Distribution of the Flora in the Alpine Zone”. In: *New Phytologist* 11.2 (1912), pp. 37–50.

[115] C. Spearman. "The proof and measurement of association between two things. By C. Spearman, 1904." In: *The American journal of psychology* 100.3-4 (1987), pp. 441–471.

[116] B. Daniels. *Puritans at play : leisure and recreation in colonial New England*. New York: St. Martin's Press, 1995.

[117] T. W. Malone. "Heuristics for Designing Enjoyable User Interfaces: Lessons from Computer Games". In: *Proceedings of the 1982 Conference on Human Factors in Computing Systems*. CHI '82. Gaithersburg, Maryland, USA: ACM, 1982, pp. 63–68.

[118] F. Garzotto. "Investigating the Educational Effectiveness of Multiplayer Online Games for Children". In: *Proceedings of the 6th International Conference on Interaction Design and Children*. IDC '07. Aalborg, Denmark: ACM, 2007, pp. 29–36.

[119] H. Desurvire, M. Caplan, et al. "Using Heuristics to Evaluate the Playability of Games". In: *CHI '04 Extended Abstracts on Human Factors in Computing Systems*. CHI EA '04. Vienna, Austria: ACM, 2004, pp. 1509–1512.

[120] H. Desurvire and C. Wiberg. "Game Usability Heuristics (PLAY) for Evaluating and Designing Better Games: The Next Iteration". English. In: *Online Communities and Social Computing*. Ed. by A.Ant Ozok and Panayiotis Zaphiris. Vol. 5621. Lecture Notes in Computer Science. Springer Berlin Heidelberg, 2009, pp. 557–566.

[121] D. Gouscos. *Proceedings of the 5th European Conference on Games Based Learning*. Academic Conferences Limited.

[122] I. Livingston, N. Lennart, et al. "Critic-Proofing: Robust Validation Through Data-Mining". In: *Playability and player experience: Proceedings of the Fun and Games 2010 Workshop*. NHTV Expertise Series 10. Workshop paper. Leuven, Belgium, 2010, pp. 81–94.

[123] L. Nacke, A. Drachen, et al. "Playability and player experience research". In: *Proceedings of DiGRA 2009*. DiGRA. 2009.

[124] J. H. Brockmyer, C. M. Fox, et al. "The development of the Game Engagement Questionnaire: A measure of engagement in video game-playing". In: *Journal of Experimental Social Psychology* 45.4 (2009), pp. 624 –634.

[125] K. Poels, W. Ijsselsteijn, et al. "Development of the Kids Game Experience Questionnaire: A self report instrument to assess digital game experiences in children". In: *Meaningful Play 2008*. 2008.

[126] K. Werbach and D. Hunter. *For the Win: How Game Thinking Can Revolutionize Your Business*. Wharton Digital Press, 2012. URL: `http://books.google.it/books?id=abg0SnK3XdMC`.

[127] G. Zichermann and C. Cunningham. *Gamification by Design: Implementing Game Mechanics in Web and Mobile Apps*. 2011. URL: `http://shop.oreilly.com/product/0636920014614.do`.

[128] M. Gagn and E. L. Deci. "Self-determination theory and work motivation". In: *Journal of Organizational Behavior* 26 (2005), pp. 331–362.

[129] *Autodesk Apocalypse Trigger*. URL: `http://proc.autodesk.com/apocalypsetrigger/`.

[130] *Fairchild M., The Secret Glossary of Social Games Analytics*. URL: `http://www.wavedash.net/2010/04/the-secret-glossary-of-social-games-analytics/`.

[131] S. Bjork and J. Holopainen. "Patterns in game design". In: (2005).

[132] J. Juul. "In search of lost time: on game goals and failure costs". In: *Proceedings of the Fifth International Conference on the Foundations of Digital Games*. FDG '10. Monterey, California: ACM, 2010, pp. 86–91.

[133] S. Ceri and P. Fraternali. *Designing Database Applications with Objects and Rules: The IDEA Methodology*. Addison-Wesley, 1997.

[134] *Xbox360Achievements*. URL: `http://www.xbox360achievements.org/`.

[135] *PS3Trophies*. URL: `http://www.ps3trophies.org/`.

[136] K. Salen and E. Zimmerman. *Rules of Play: Game Design Fundamentals*. Mit Press, 2004. URL: http://books.google.it/books?id=UM-xyczrZuQC.

[137] T. Fullerton, C. Swain, et al. *Game Design Workshop: A Playcentric Approach to Creating Innovative Games*. Gama Network Series. Elsevier Morgan Kaufmann, 2008. URL: http://books.google.it/books?id=OjIYWtqWxtAC.

[138] J. Juul. *Half-real: video games between real rules and fictional worlds*. MIT Press, 2005. URL: http://books.google.it/books?id=ivdOAAAAMAAJ.

[139] *Achievement 101*. URL: http://www.gamasutra.com/blogs/GregMcClanahan/20091202/3709/Achievement_Design_101.php.

[140] A. Drachen. *What are game metrics?* 2012. URL: http://blog.gameanalytics.com/blog/what-are-game-metrics.html.

[141] Wikipedia. *Average revenue per user*. 2013. URL: http://en.wikipedia.org/wiki/Average_revenue_per_user.

[142] N. Matt. *The Secret Glossary of Social Games Analytics*. 2010. URL: http://www.wavedash.net/2010/04/the-secret-glossary-of-social-games-analytics/.

[143] Wikipedia. *Churn Rate*. 2013. URL: http://en.wikipedia.org/wiki/Churn_rate.

[144] J. Haden. *Best Way to Track Customer Retention*. 2013. URL: http://www.inc.com/jeff-haden/best-way-to-calculate-customer-retention-rate.html.

[145] M. Sheeley. *Defining "Active User" And The DAU To MAU Ratio*. 2013. URL: http://www.makegreatsoftware.com/2013/09/defining-active-and-dau-to-mau-ratio.html.

[146] RJMetrics. *Cohort Analysis 101*. 2013. URL: http://cohortanalysis.com/.

[147] G. Zichermann and C. Cunningham. *Gamification by Design: Implementing Game Mechanics in Web and Mobile Apps*. 2011. URL: http://shop.oreilly.com/product/0636920014614.do.

[148] L. Petersen. *Q&A: Gabe Zichermann on gamification, fun and metrics*. 2011. URL: `http://econsultancy.com/it/blog/7283-q-a-with-gabe-zichermann-on-gamification-fun-and-metrics`.

[149] A. Bozzon, P. Fraternali, et al. “Modeling CrowdSourcing Scenarios in Socially-Enabled Human Computation Applications”. English. In: *Journal on Data Semantics* 3.3 (2014), pp. 169–188.

[150] E. Kapsammer, S. Mitsch, et al. “A First Step Towards a Conceptual Reference Model for Comparing Social User Profiles”. In: (2011).

[151] A. Axenopoulos, P. Daras, et al. “I-SEARCH: A Unified Framework for Multimodal Search and Retrieval”. In: (2012), pp. 130–141.

[152] M. Brambilla, P. Fraternali, et al. “BPMN and Design Patterns for Engineering Social BPM Solutions”. In: *Proceedings of the Fourth International Workshop on BPM and Social Software, BPMS 2011*. 2011.

[153] P. Daras, A. Axenopoulos, et al. “Introducing a unified framework for content object description.” In: *IJMIS* 2.3/4 (2011), pp. 351–375.

[154] P. Salembier and T. Sikora. *Introduction to MPEG-7: Multimedia Content Description Interface*. Ed. by B.S. Manjunath. New York, NY, USA: John Wiley & Sons, Inc., 2002.

[155] F. M. Suchanek, G. Kasneci, et al. “Yago: a core of semantic knowledge”. In: *Proceedings of the 16th international conference on World Wide Web*. WWW ’07. Banff, Alberta, Canada: ACM, 2007, pp. 697–706.

[156] C. Lee and V. Soo. “The conflict detection and resolution in knowledge merging for image annotation”. In: *Inf. Process. Manage.* 42.4 (July 2006), pp. 1030–1055.

[157] A. Bozzon, M. Brambilla, et al. “Answering search queries with CrowdSearcher”. In: *WWW*. 2012, pp. 1009–1018.

[158] L. Galli and P. Fraternali. “Achievement Systems Explained”. In: *Serious Gaming & Social Connect*. Singapore: Springer-Verlag.

[159] P. Fraternali, M. Tagliasacchi, et al. “The CUBRIK Project: Human-enhanced Time-aware Multimedia Search”. In: *Proceedings of the 21st International Conference Companion on World Wide Web*. WWW ’12 Companion. Lyon, France: ACM, 2012, pp. 259–262.

[160] M. Eichner and V. Ferrari. *CALVIN upper-body detector*. URL: `http://www.vision.ee.ethz.ch/~calvin/calvin_upperbody_detector/`.

[161] L. Bourdev and J. Malik. “Poselets: Body Part Detectors Trained Using 3D Human Pose Annotations”. In: *ICCV*. 2009.

[162] Kee Square. *Morpheus SDK*. URL: `http://www.keesquare.com/`.

[163] S. Yamaoka et al. “Cultural Analytics in Large-Scale Visualization Environments”. In: *Computer* 44 (2011), pp. 39–48.

[164] A. Bozzon, M. Brambilla, et al. “Answering Search Queries with CrowdSearcher”. In: *Proceedings of the 21st International Conference on World Wide Web*. WWW ’12. Lyon, France: ACM, 2012, pp. 1009–1018.

[165] J. Deng, J. Krause, et al. “Fine-Grained Crowdsourcing for Fine-Grained Recognition”. In: *Computer Vision and Pattern Recognition (CVPR), 2013 IEEE Conference on*. 2013, pp. 580–587.

[166] T. Huang, K.-Y. Cheng, et al. “A collaborative benchmark for region of interest detection algorithms”. In: *Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on*. 2009, pp. 296–303.

[167] K. Yamaguchi, M.H. Kiapour, et al. “Parsing clothing in fashion photographs”. In: *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on*. 2012, pp. 3570–3577.

[168] K.-Y.. KCheng, L. Kennedy, et al. “Getting the Look: Clothing Recognition and Segmentation for Automatic Product Suggestions in Everyday Photos”. In: *Proceedings of the 3rd ACM Conference on International Conference on Multimedia Retrieval*. ICMR ’13. Dallas, Texas, USA: ACM, 2013, pp. 105–112.

[169] K. Yamaguchi, M.H. Kiapour, et al. “Paper Doll Parsing: Retrieving Similar Styles to Parse Clothing Items”. In: *Computer Vision (ICCV), 2013 IEEE International Conference on*. 2013, pp. 3519–3526.

[170] A. Torralba, B.C. Russell, et al. “LabelMe: Online Image Annotation and Applications”. In: *Proceedings of the IEEE* 98.8 (2010), pp. 1467–1484.

[171] B. Loni, M. Menendez, et al. “Fashion-focused Creative Commons Social Dataset”. In: *Proceedings of the 4th ACM Multimedia Systems Conference*. MMSys ’13. Oslo, Norway: ACM, 2013, pp. 72–77.

[172] J.L. Bentley and T.A. Ottmann. “Algorithms for Reporting and Counting Geometric Intersections”. In: *Computers, IEEE Transactions on* C-28.9 (1979), pp. 643–647.

[173] M. Hassenzahl. “The interplay of beauty, goodness, and usability in interactive products”. In: *Human-Computer Interaction* 19.4 (2004), pp. 319–349.

[174] M. Hassenzahl, Michael Burmester, et al. “AttrakDiff: Ein Fragebogen zur Messung wahrgenommener hedonischer und pragmatischer Qualitt”. German. In: *Mensch & Computer 2003*. Vol. 57. Berichte des German Chapter of the ACM. Vieweg+Teubner Verlag, 2003, pp. 187–196.