

POLITECNICO DI MILANO

Corso di Laurea in Ingegneria Informatica
Dipartimento di Elettronica e Informazione



**Definizione degli scenari di emergenza e Web 2.0:
SocialEMIS**

Relatore: Ing. Pierluigi PLEBANI

Tesi di Laurea di:

Fabio PARONI, matr. 725033

Anno Accademico 2010-2011

*...ad Andrea, Mamma e Papà,
a tutta la mia famiglia
e a tutti gli amici.*

Abstract

Le tematiche relative alle calamità ambientali e alla gestione delle emergenze sembrano aver assunto, negli ultimi anni, un'importanza ed un'attenzione sempre maggiore. I media, in questo ambito, hanno svolto e continuano a svolgere un ruolo importante: avere immagini e video in tempo reale dei danni e dei disastri causati da un evento calamitoso, oltre a dare un'idea chiara dell'entità del danno, fa sì che la sensibilità verso queste problematiche sia notevolmente più alta.

Le nuove tecnologie IT hanno rivoluzionato e tutt'ora continuano a cambiare l'approccio e il metodo per la gestione dell'emergenza, che nel corso degli anni ha mantenuto costante però una caratteristica: l'enorme difficoltà di gestione e le inevitabili difficoltà di previsione. Le sale di emergenza delle protezioni civili utilizzano da tempo diversi strumenti hardware e software per poter gestire al meglio le calamità e i rischi legati all'ambiente e al territorio.

Il presente lavoro di tesi si colloca in questo contesto e mira allo sviluppo di un prototipo di EMIS (Emergency Management Information System) rivolto ad esperti di dominio che operano all'interno delle sale di emergenza. L'EMIS sarà un'applicazione software che andrà ad integrare (ed in alcuni casi potrebbe sostituire) alcuni strumenti utilizzati dalle protezioni civili; sarà utile come punto di integrazione di alcuni flussi di dati provenienti da enti e strumenti hardware distribuiti sul territorio e utilizzerà questi stessi dati per poter creare e gestire degli Scenari di Rischio.

Il presente documento di tesi vuole essere il risultato finale del lavoro di progettazione e sviluppo della web application che prende il nome di SocialEMIS volendo sottolineare proprio l'aspetto della politica 'sociale' che l'applicazione segue.

Indice

1	Introduzione	8
1.1	Obiettivi del progetto	9
1.2	Overview di SocialEmis	11
1.2.1	Perché un EMIS	13
1.2.2	Perché Social	13
1.2.3	Perché una WEB Application	14
1.3	Struttura del documento di tesi	14
2	Stato dell'arte	16
2.1	Gestione dell'emergenza	16
2.1.1	Il ciclo di vita dell'emergenza	17
2.1.2	Il Metodo Augustus	19
2.2	Scenari di rischio	21
2.3	Emergency Management Information Systems	23
2.3.1	Caratteristiche generali	24
2.3.2	Flessibilità, affidabilità ed interoperabilità	25
2.3.3	Evoluzione degli EMIS	25
2.3.4	Progetti interessanti	27
2.4	Conclusioni e considerazioni	29
3	Strumenti tecnologici	30
3.1	Vaadin	30
3.2	RESTful Web Services	31
3.2.1	REST	31
3.3	JSON	32
3.3.1	Strutture del formato	32
3.3.2	JSON e XML	33
3.4	KML	34
3.5	JPA	35

Indice

3.6	Google Maps	36
4	SocialEMIS	37
4.1	Pianificazione e approccio allo sviluppo	37
4.1.1	Panoramica delle funzionalità di SocialEMIS	37
4.2	Raccolta e analisi dei requisiti	42
4.2.1	Raccolta dei requisiti	42
4.2.2	Specifica dei requisiti	43
4.2.3	Design Scenario di Rischio	46
4.2.4	Ricerca Scenari di Rischio	47
4.3	Analisi dettagliata	48
4.3.1	Definizione dei Moduli	48
4.3.2	Progettazione dei dati	49
4.4	Progettazione	53
4.4.1	Architettura SocialEMIS	53
4.5	Algoritmi di Similarity	56
4.5.1	Algoritmo di ricerca per Emergenza	56
4.5.2	Algoritmo di ricerca per Emergenza / Luogo	57
4.5.3	Algoritmo di ricerca Multirischio	58
5	Implementazione	60
5.1	Sviluppo Front-End di sistema	63
5.2	Sviluppo Back-End di sistema	65
6	Conclusioni	68

Elenco delle figure

2.1	Ciclo di vita dell'emergenza	17
2.2	Esempio di Scenario di Rischio per il caso del Comune di Premana	22
3.1	Architettura web application sviluppata con Vaadin	30
3.2	Triangolo paradigma REST	32
3.3	JSON object	33
3.4	JSON array	33
3.5	JSON values	33
3.6	Esempio del contenuto di un file kml	35
3.7	Google Maps	36
4.1	Schematizzazione funzionalità di Design Scenario	39
4.2	Schematizzazione funzionalità di Ricerca Scenario	40
4.3	UML - Use Case Diagram	44
4.4	Moduli software di SocialEMIS	49
4.5	Diagramma ER	51
4.6	Schema dei flussi di dati	52
4.7	SocialEMIS - Architettura di sistema	54
4.8	Interfaccia utente	55
4.9	Diagramma di flusso per l'algoritmo di ricerca	57
4.10	Screenshot dei risultati di una ricerca di scenari simili	59
5.1	Architettura del progetto SocialEMIS	61
5.2	Elenco dei progetti creati	62
5.3	Layout applicazione	64
5.4	Code snippet della classe Hospital Entity	66
5.5	Elenco dei metodi forniti dalla classe Controller	66
5.6	Schema di definizione dei progetti implementati	67

Elenco delle tabelle

2.2	Confronto tra i diversi EMIS	29
4.1	Corrispondenza Requisiti, Use Case e Attori	45

1 Introduzione

Le tematiche relative alle calamità ambientali e alla gestione delle emergenze sembrano aver assunto, negli ultimi anni, un'importanza ed un'attenzione sempre maggiore. I media, in questo ambito, hanno svolto e continuano a svolgere un ruolo importante: avere immagini e video in tempo reale dei danni e dei disastri causati da un evento calamitoso, oltre a dare un'idea chiara dell'entità del danno, fa sì che la sensibilità verso queste problematiche sia notevolmente più alta.

Le nuove tecnologie, nuovi dispositivi hi-tech e la rapidità con cui è possibile trasmettere informazioni e contenuti, hanno permesso questa concentrazione collettiva dell'attenzione verso questo tipo di problemi. La gestione dell'emergenza sembra essere diventata qualcosa che non è più compito esclusivamente degli esperti e degli enti specializzati, come la Protezione Civile, ma è qualcosa in cui il cittadino comune può dare un contributo, magari non attivamente sul campo, ma sicuramente come fonte di informazione. Basti pensare che persino le più famose testate giornalistiche fanno ampiamente uso di contenuti che, per esempio, trovano caricati su internet da user web generici, per documentare nelle loro pagine di eventi calamitosi o di particolari emergenze. Dopo il terremoto che colpì Haiti nel gennaio del 2010, la prima segnalazione dell'evento fu postata su Twitter; i primi video amatoriali dell'accaduto furono caricati su YouTube poco tempo dopo la violenta scossa e in brevissimo tempo Facebook segnalò la creazione di numerosi gruppi, con migliaia di iscritti, per poter documentare gli eventi successivi al disastro ambientale, per segnalare la scomparsa di persone e per organizzare aiuti umanitari.

L'emergenza sembra così un tema di attualità, qualcosa di cui prima non ci si interessava e che adesso copre le prime pagine dei giornali. La verità è che la gestione delle emergenze ambientali è sempre stato un tema molto affrontato e discusso anche negli anni passati.

Le nuove tecnologie IT hanno rivoluzionato e tutt'ora continuano a cambiare l'approccio e il metodo per la gestione dell'emergenza, che nel corso degli anni ha mantenuto costante però una caratteristica: l'enorme difficoltà di gestione e le inevitabili difficoltà di previsione. Le sale di emergenza delle protezioni civili utilizzano da tempo diversi strumenti hardware e software per poter gestire al meglio le calamità e i rischi legati

all'ambiente e al territorio.

Questi strumenti ed in particolare gli applicativi ed i sistemi software, come riportato in [1], utilizzati per diverse funzioni ed in particolare per il monitoraggio del territorio, hanno il difetto di essere strumenti completamente scollegati tra loro. I dati raccolti, che risultano essere numerosi e diversificati, non sono però integrati tra di loro: gli applicativi risultano strumenti indipendenti e scarsamente interoperabili.

Il presente lavoro di tesi si colloca in questo contesto e mira allo sviluppo di un prototipo di EMIS (Emergency Management Information System) rivolto ad esperti di dominio che operano all'interno delle sale di emergenza. L'EMIS sarà un'applicazione software che andrà ad integrare (ed in alcuni casi potrebbe sostituire) alcuni strumenti utilizzati dalle protezioni civili; sarà utile come punto di integrazione di alcuni flussi di dati provenienti da enti e strumenti hardware distribuiti sul territorio e utilizzerà questi stessi dati per poter creare e gestire degli Scenari di Rischio.

Il presente documento di tesi vuole essere il risultato finale del lavoro di progettazione e sviluppo della web application che prende il nome di SocialEMIS volendo sottolineare proprio l'aspetto della politica 'sociale' che l'applicazione segue. Al momento l'opera di condivisione si limita agli esperti di dominio, ma l'approccio punterà successivamente a permettere anche ai 'cittadini' di poter fornire informazioni e di poter fruire informazioni dal sistema.

1.1 Obiettivi del progetto

Come già anticipato, il progetto SocialEMIS mira alla creazione di una web application, utilizzabile da esperti nel campo della gestione delle emergenze ambientali e del monitoraggio del territorio. Il tool deve essere uno strumento di ausilio per la fase di Preparedness del ciclo di gestione delle emergenze. Entrando più nello specifico, deve permettere all'esperto di dominio di creare e gestire dei cosiddetti Scenari di Rischio o Scenari di Emergenza avendo a disposizione, all'interno del tool stesso, tutte le informazioni necessarie per il design e la successiva gestione di uno specifico scenario.

Ogni esperto di dominio attraverso SocialEMIS avrà la possibilità di creare i propri scenari di rischio, di poterli salvare e modificare in caso di necessità, ma avrà anche la possibilità di avere accesso a tutti quegli scenari che sono stati creati da altri esperti di dominio e che si riferiscono ad altre emergenze. Quest'aspetto della condivisione dell'informazione e della conoscenza è un punto molto importante ed una caratteristica centrale del progetto stesso.

1 Introduzione

Un altro aspetto importante e che è stato determinante nella definizione degli obiettivi del progetto è stata la carenza, di cui si è già accennato nell'introduzione, di integrazione tra i dati che vengono gestiti all'interno di diversi centri operativi delle sale di emergenza e a volte anche all'interno del medesimo centro. Si è già accennato come esistano diversi applicativi indipendenti tra loro che si occupano di collezionare una grande quantità di dati dal territorio, necessari alla gestione dell'emergenza, ma che non sono integrati tra loro. Invece di avere un set di applicativi fortemente separati tra loro SocialEMIS, seguendo una linea completamente diversa rispetto a quella attualmente in uso, cerca di integrare all'interno di una singola applicazione più flussi di dati provenienti dal territorio e di integrarli all'interno del sistema per utilizzarli nella creazione degli scenari di emergenza, basandosi su un'architettura a servizi.

Uno di questi flussi di dati è relativo alle informazioni che provengono dalle Wireless Sensor Network. Le reti di sensori che sono installate sul territorio monitorano e immagazzinano dati relativi a determinate misurazioni. Il contributo, da un punto di vista della qualità e dell'aggiornamento dei dati, relativo alla gestione dell'emergenza è notevole. Infatti, le reti di sensori hanno avuto un fortissimo sviluppo ed un impiego sempre maggiore negli ultimi anni; sono strumenti essenziali in tutte le diverse fasi del ciclo di vita dell'emergenza. Per questo motivo un obiettivo del progetto è proprio quello di sfruttare al meglio questo flusso di informazioni, integrandolo all'interno del sistema per far sì che i dati provenienti dalle Wireless Sensor Network possano essere utilizzati adeguatamente dalle diverse funzionalità del sistema SocialEMIS. In particolare, come vedremo in dettaglio nei prossimi capitoli, attraverso l'utilizzo di web services sarà possibile far confluire all'interno del sistema i dati pubblici di sistemi eterogenei come le reti di sensori attraverso servizi RESTful.

Una delle idee portanti che sta alla base di SocialEMIS è proprio quella di poter creare una web application che si appoggia, attraverso una Service Oriented Architecture, su uno strato di informazioni condivise tra tutti gli utilizzatori dell'applicazione stessa: un certo scenario di rischio, creato da un certo user, non è proprietà esclusiva del solo proprietario, ma secondo la politica 'social' del progetto, diventa una risorsa ed una fonte informativa per tutti coloro che utilizzano o utilizzeranno l'applicazione. Gli utilizzatori pur lavorando indipendentemente ai loro progetti e ai loro scenari diventano automaticamente collaboratori e vanno a creare, in modo del tutto trasparente, una rete sociale che gravita intorno alla gestione dell'emergenza e, nella fattispecie, alla creazione degli scenari di emergenza; gli scenari, le informazioni, i processi, i passi del piano di emergenza creati per il caso specifico, diventano automaticamente soluzioni ad uno o più casi non direttamente riferiti al caso di studio originale.

Accanto ad una serie di servizi utili per il design e il salvataggio dello scenario, SocialEMIS mette a disposizione un servizio di Ricerca di Scenari attraverso il quale l'esperto di dominio, impostando determinati parametri di ricerca, può andare a ricercare, e successivamente consultare, tra tutti gli scenari che sono stati precedentemente salvati da tutti gli utilizzatori dell'applicazione. Avere un patrimonio informativo condiviso, costituito da tutti gli scenari che sono stati creati attraverso la web application, è la caratteristica più importante dell'intero progetto SocialEMIS: si ritiene infatti che la possibilità di avere accesso alle informazioni, ai dati e alle strategie di response delineate da altri esperti di dominio nei propri scenari di rischio, aumenti la qualità e il livello di efficacia dei nuovi scenari creati.

Questo lavoro di tesi si inquadra all'interno del progetto INTERREG-MIARIA (Monitoraggio Idrogeologico Adattivo a Supporto del Piano di Rischio Integrato Alpino) che, tra i suoi diversi obiettivi, punta a coordinare e migliorare l'efficacia delle procedure di analisi e gestione e dei relativi interventi volti a migliorare il complessivo sistema di gestione del rischio e delle emergenze.

1.2 Overview di SocialEmis

Il progetto SocialEMIS nasce con l'intento di creare uno strumento di ausilio alla gestione delle emergenze ambientali. Nella sezione precedente sono stati delineati, ad alto livello, i principali obiettivi del progetto, quindi cosa dovrebbe fare l'applicazione e che obiettivi dovrebbe raggiungere la sua implementazione reale. In questa sezione si affronta il discorso in maniera più concreta presentando una panoramica di SocialEMIS: vediamo come la web application può raggiungere gli obiettivi prefissati attraverso la presentazione delle caratteristiche più importanti. Alcune delle caratteristiche che verranno elencate saranno solo accennate perché una trattazione più completa sarà fornita negli appositi capitoli e sezioni.

Le considerazioni fatte nell'introduzione e gli obiettivi delineati nella sezione precedente hanno portato allo sviluppo di un tool con le seguenti caratteristiche principali:

- *Interoperabilità tra applicazioni*: l'interoperabilità è la prima caratteristica che identifica SocialEMIS. SocialEMIS vuole essere un'applicazione che trova una soluzione ai problemi di integrazione ed interoperabilità riscontrati negli applicativi attualmente in uso nelle sale operative delle protezioni civili, attraverso due ingredienti fondamentali: un modello di architettura orientata ai servizi e l'utilizzo dei Web Services.

1 Introduzione

- *Condivisione del patrimonio informativo generato*: SocialEMIS è un sistema che genera nuova informazione integrando diversi flussi di dati provenienti da diverse fonti. Tutte le informazioni e i dati che vengono creati da un utilizzatore, cioè da un esperto di dominio, sono informazioni che vengono memorizzate e rese disponibili a tutti gli altri utilizzatori di SocialEMIS. Il patrimonio informativo diventa uno strumento utilizzabile e una risorsa consultabile da tutti gli users della web application.
- *Definizione degli scenari di rischio*: gli scenari di emergenza sono lo strumento centrale per la gestione della fase di Preparedness e un' importante base di partenza per la creazione dei piani di emergenza da attuare in fase di Response. Sono un potente strumento che concentra in un unico oggetto tutte le informazioni che sono necessarie per avere un quadro completo del territorio e dell'eventuale evento calamitoso. Lo scenario di rischio creato e gestito da SocialEMIS non è una fotografia statica di un certo istante, che fissa la realtà in una determinata configurazione, ma un oggetto dinamico che in ogni momento può essere aggiornato e modificato per poter tener traccia degli eventuali cambiamenti sul territorio.
- *Ricerca degli scenari di rischio*: la definizione degli scenari di emergenza e la loro condivisione all'interno di un patrimonio informativo comune, permette all'utilizzatore di andare ad effettuare delle ricerche tra gli scenari memorizzati. L'esperto di dominio, a seconda dei casi, potrebbe essere interessato alla ricerca di una serie di scenari simili per entità e tipologia dell'emergenza, allo scenario che sta definendo lui stesso, per poterli confrontare o per prenderne spunto. L'esperto, prevedendo l'eventualità di una certa situazione di emergenza potrebbe essere interessato alla ricerca dello scenario più simile alla situazione che potrebbe verificarsi a breve. Il sistema, per soddisfare questi requisiti, implementa una serie di funzioni di *ricerca per similarità* tra i diversi scenari di rischio.
- *Utilizzo di dati in tempo reale provenienti dal territorio*: come già accennato, lo Scenario di Rischio è un strumento che si compone di una serie di dati provenienti da diverse sorgenti e che sono utili per riprodurre al meglio una rappresentazione del territorio e dell'ambiente preso in considerazione. La dinamicità dell'oggetto 'scenario di rischio' è data proprio dal fatto che tutti questi flussi di dati sono fruibili in tempo reale; sono quindi aggiornabili e rappresentano fedelmente gli inevitabili cambiamenti che avvengono nell'ambiente. Un esempio sono i dati collezionati e salvati provenienti dalle Wireless Sensor Network, che in SocialEMIS sono una im-

portantissima fonte di dati, utile per misurare delle grandezze fisiche e per prevedere eventuali emergenze ambientali.

1.2.1 Perché un EMIS

L'applicazione sviluppata rientra nella categoria dei sistemi cosiddetti EMIS e cioè Emergency Management Information Systems che per definizione sono: "Information Systems designed to collect, analyze and share information in support of emergency management activities". Gli EMIS rientrano tra gli strumenti software più importanti che vengono utilizzati da coloro che lavorano nel campo delle emergenze. Questi strumenti vengono utilizzati in tutte le fasi del ciclo di gestione dell'emergenza perché sono sistemi semplici e veloci che possono essere sviluppati per controllare al meglio le diverse attività di 'risk management'. Esistono EMIS che si occupano di gestire le risorse nella fase di Response di un'emergenza ambientale; esistono EMIS che invece vengono utilizzati in fase di Mitigation per documentare le problematiche che emergono dopo un certo evento calamitoso.

Gli EMIS sono strumenti che si possono adattare a diverse funzioni e diversi scopi. Nel caso descritto in questo documento l'intento è proprio quello di sviluppare un primo prototipo di sistema informativo che ha come dominio ovviamente quello della gestione dell'emergenza e come scopo quello della creazione e condivisione di scenari di rischio. Il lavoro di sviluppo, come vedremo in maniera dettagliata nel seguito, si è concentrato quindi su sistema informativo collocato in un contesto non semplice come il mondo delle emergenze.

1.2.2 Perché Social

Una delle idee alla base del lavoro di progettazione di SocialEMIS è quella di creare una vera e propria rete sociale per connettere tra di loro un gruppo di esperti che si occupano di gestione delle emergenze: oggetto centrale e condiviso della rete sociale è lo Scenario di Rischio. L'applicazione oltre ad essere uno strumento per le attività di raccolta dati dal territorio e generazione di nuovi scenari di rischio, diventa un vero e proprio mezzo per condividere, all'interno della rete sociale di esperti, una serie di informazioni utili per la corretta gestione di eventi calamitosi.

Con SocialEMIS si cerca di implementare un'applicazione che non è rivolta al lavoro indipendente del singolo utilizzatore; al contrario, questo strumento ha tra gli obiettivi principali proprio quello di essere un punto di contatto tra diversi enti, tra diversi esperti di emergenze che lavorando separatamente, in diverse sedi e a diversi progetti, in verità

collaborano in modo del tutto trasparente per creare uno strato di informazione condivisa e disponibile per tutti. Gli utenti che utilizzano SocialEMIS, generano e gestiscono così Scenari di Rischio condivisi: uno scenario creato da un esperto di dominio può essere utilizzato da altri per la creazione di nuovi scenari o per la ricerca di scenari simili. Come vedremo in dettaglio nel capitolo centrale, la creazione e la ricerca di scenari saranno due delle funzionalità principali del progetto.

1.2.3 Perché una WEB Application

Si può dire che la scelta dello sviluppo di un'applicazione web sia stata una scelta obbligata, nel senso che è stata dettata proprio dagli obiettivi iniziali del progetto; la natura e le specifiche dell'applicazione portano ad una scelta ovvia che vede lo sviluppo di un tool che sia fruibile e che fornisca i suoi servizi attraverso la rete internet. Non sarebbe stato in alcun modo possibile sviluppare un tool con le stesse caratteristiche e le stesse funzionalità implementando un'applicazione desktop; quantomeno non con la stessa facilità ed i vantaggi evidenti che uno sviluppo di un'applicazione web comporta. L'utilizzo della rete internet risulta fondamentale per i seguenti requisiti applicativi di:

- interoperabilità con altri sistemi
- condivisione di dati ed informazioni
- utilizzo di un'architettura orientata ai servizi (SOA)
- dati e flussi informativi provenienti dal territorio real-time
- facilità e velocità di utilizzo ed accesso al tool

Inoltre l'utilizzo del web implica i tipici vantaggi delle Web Application:

1. SaaS: Software as a Service che permette di usufruire di un'applicazione in modo semplice, quando se ne ha bisogno secondo una politica della richiesta/offerta di servizi on-line.
2. Update costante e trasparente dell'applicazione.
3. Compatibilità cross-platform: il tool viene eseguito nei più comuni web browser.

1.3 Struttura del documento di tesi

Il primo Capitolo prosegue con una presentazione ad alto livello dell'intero progetto SocialEMIS, descrivendone brevemente gli obiettivi e dando una panoramica ad alto

1 Introduzione

livello delle caratteristiche principali e delle motivazioni che hanno portato alla scelta di queste caratteristiche.

Il secondo Capitolo invece sarà un capitolo più tecnico, legato alla presentazione dello stato dell'arte in ambito di gestione dell'emergenza. In questa sezione verranno presentate le quattro fasi del ciclo di vita dell'emergenza ed il Metodo Augustus; successivamente si descriveranno gli scenari di rischio in particolare sottolineando l'importanza di questi strumenti e la loro collocazione all'interno del ciclo dell'emergenza. Il capitolo si concluderà con una sezione sugli EMIS ad oggi disponibili sul web, per presentare le caratteristiche di questo importante strumento.

Il terzo Capitolo sarà interamente dedicato ad una presentazione dei principali strumenti tecnologici che sono stati utilizzati nello sviluppo dell'applicazione SocialEMIS tra cui, il framework Vaadin, i RESTful Web Services, la notazione JSON, lo standard KML, le Java Persistence API ed infine il famoso servizio di web mapping Google Maps.

Il quarto Capitolo sarà il nucleo dell'intero documento: in questa parte verrà presentato il progetto SocialEMIS nel dettaglio, passando in rassegna l'approccio allo sviluppo, gli obiettivi e i requisiti del progetto. Dopo di che, da un punto di vista più tecnico, verrà presentata l'architettura della web application e le funzionalità che essa mette a disposizione.

A conclusione del documento il quinto ed il sesto Capitolo presenteranno rispettivamente una schematizzazione della fase di implementazione del progetto con alcuni esempi del codice sorgente di SocialEMIS ed infine le conclusioni, le utime considerazioni e gli eventuali sviluppi futuri.

2 Stato dell'arte

2.1 Gestione dell'emergenza

La gestione delle emergenze che hanno un impatto sull'ambiente e sull'uomo, coinvolge diversi enti e persone specializzate in diverse aree di conoscenza. L'emergenza e la gestione del rischio sono materie che negli ultimi anni hanno ricevuto un'enorme attenzione da un punto di vista mediatico, da un punto di vista scientifico-tecnologico e culturale. Le notizie e i bilanci dei morti per emergenze ambientali (ma anche di diversa natura) come il terremoto e il conseguente tsunami in Giappone del 2011 o, per rimanere vicini, il terremoto che colpì l'Aquila nel 2009, hanno profondamente mutato il rapporto della società nei confronti di questi avvenimenti.

La lotta alle emergenze ambientali, la prevenzione e la preparazione agli eventi calamitosi sono aspetti che non si riferiscono più solo agli addetti ai lavori, ma sono diventati un pensiero comune, un fenomeno sociale. I media in questa trasformazione hanno certamente avuto un ruolo importantissimo: avere immagini e video quasi in tempo reale dei disastri e della distruzione di certi eventi non può non toccare la coscienza comune. È proprio questo l'aspetto che emerge di più nel leggere le linee guida dei progressi e delle novità in ambito di 'emergency management'. Forse nella concezione comune, nell'idea generale sull'emergenza e sul rischio, non c'è ancora un quadro completo.

Il concetto di emergenza ambientale, ad esempio, viene identificato con il periodo di tempo successivo allo scatenarsi di un qualsiasi evento catastrofico (terremoto, uragano ecc.), ma non tiene conto delle fasi precedenti ad esso e di quelle che si protraggono nel tempo dopo che, come si suol dire, il peggio è passato. La letteratura sulla gestione dell'emergenza riporta invece, ormai da parecchio tempo, uno schema ben preciso sull'intero *ciclo di vita dell'emergenza*, composto da quattro fasi differenti e ben distinte come rappresentato in Figura 2.1 .

2.1.1 Il ciclo di vita dell'emergenza

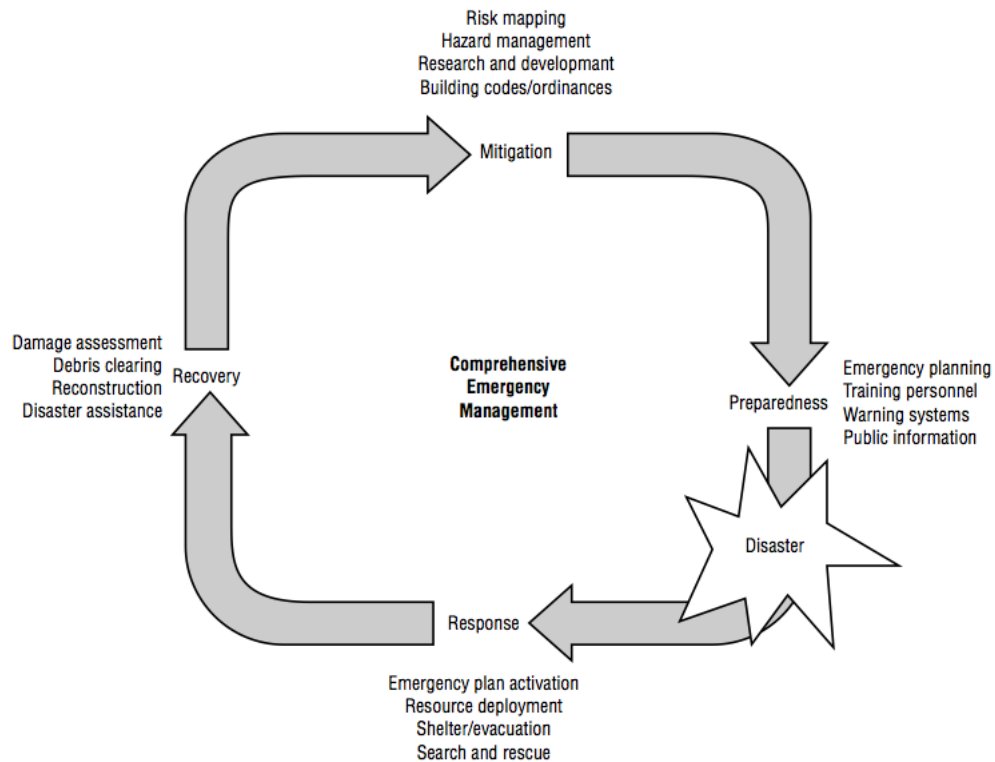


Figura 2.1: Ciclo di vita dell'emergenza

Preparedness

Con la fase di Preparedness si identifica in generale tutta la fase che precede l'eventuale emergenza. In questa parte del ciclo si raggruppano tutte quelle azioni e quei processi che vengono svolti per essere pronti nel caso si verifichi l'emergenza. È una fase principalmente di studio e pianificazione in cui sia gli esperti di dominio, che gli enti e in parte la popolazione civile, si impegnano nel prepararsi adeguatamente. Le procedure tipiche della fase di preparedness possono essere elencate in questo modo:

- identificazione e studio della potenziale emergenza (tipologia, entità, ecc.)
- studio del territorio e dell'ambiente interessato (area, popolazione, strutture)
- previsioni dell'impatto sul territorio e sulla popolazione

2 Stato dell'arte

- creazione e gestione di *Scenari di Rischio*
- progettazione di piani di emergenza che sintetizzino la risposta futura all'evento calamitoso.

Nella prossima sottosezione si vedrà più in dettaglio uno strumento utilizzato dalla Protezione Civile italiana nella fase di Preparedness che prende il nome di Metodo Augustus.

Response

La fase di Response ha inizio dallo scatenarsi dell'emergenza e la sua durata non è ben definibile a priori. Generalmente si dichiara conclusa la fase di Response nel momento in cui è terminata l'emergenza in senso stretto. La fase di Response identifica quindi i momenti più critici che seguono l'evento calamitoso. In questa fase specifica si attuano i piani di emergenza stilati in fase di Preparedness, si traducono in azioni reali i passi previsti durante l'emergency planning: in particolare si attivano tutti gli organi e gli enti che hanno a che vedere con il primo soccorso alla popolazione e si cerca di gestire al meglio tutte le risorse necessarie per il contenimento del danno. La tempestività della fase di Response può essere un fattore cruciale e ha un forte impatto sui danni che l'emergenza può provocare.

Un aspetto da tenere sempre presente è che, ovviamente, le previsioni fatte in fase di Preparedness non saranno mai accurate e perfettamente combacianti con la realtà quindi spesso, in fase di response, gli scenari di emergenza riporteranno delle discrepanze con la situazione reale e i piani di emergenza dovranno essere modificati al momento.

Recovery

La fase di Recovery può essere identificata con la fase di ricostruzione della situazione di quiete, il ripristino della normalità. È la fase di ripartenza e di ricostruzione del paesaggio e delle aree più colpite e danneggiate. In questa fase, in cui l'emergenza è ormai superata, gli enti interessati si occupano dell'assistenza alla popolazione e del ripristino delle attività del luogo. Tipicamente è proprio durante il Recovery che iniziano le opere di ricostruzione.

Mitigation

Infine la fase di Mitigation indica la lunga fase che comprende tutte le misure adottate affinché non si ripresenti una certa emergenza. È la continuazione a lungo termine della fase di Recovery. Le soluzioni adottate precedentemente all'emergenza vengono ricalibrate e migliorate per avere, nell'eventualità, una risposta sempre più adatta all'evento

calamitoso. La fase di Mitigation fa da punto di contatto tra la fase di Recovery e la fase di Preparedness, chiudendo così il ciclo di vita dell'emergenza.

2.1.2 Il Metodo Augustus

Il Metodo Augustus è un progetto nato nel 1998 da uno studio del geologo Elvezio Galanti che ideò questo strumento, utile soprattutto nella fase di Preparedness per la pianificazione dell'emergenza. Galanti, che intuisce subito la difficoltà del contesto, sviluppa un approccio semplificato per la gestione delle attività principali e di supporto all'emergenza. Il concetto che secondo Galanti deve stare alla base della gestione del rischio è la cooperazione sinergica ed il coordinamento delle forze utilizzate per contrastare l'emergenza. Galanti riconosce che le difficoltà principali risultano non tanto da un'insufficienza di mezzi, quanto da un utilizzo non coordinato e non metodico di esse. Ad esempio avere una buona cooperazione tra i diversi livelli amministrativi della Protezione Civile è un aspetto chiave nella lotta alle emergenze ambientali. composto di tre passi fondamentali:

1. Definizione dello Scenario: una fotografia della zona interessata, sulla quale sarà attuato il piano di emergenza.
2. Individuazione dei rischi e delle eventuali emergenze
3. Attuazione delle 14 funzioni di supporto, messe a disposizione dai centri operativi che definiscono ruoli e compiti.:
 - a) *F 1* - Tecnica e di pianificazione: Gruppi di ricerca scientifica (CNR) - Istituto Nazionale di Geofisica - Regioni - Dipartimento PC - Servizi Tecnici Nazionali 21
 - b) *F 2* - Sanità, assistenza sociale e veterinaria: Responsabile C.O. 118 - Regione / AA.SS.LL - C.R.I. - Volontariato socio-sanitario
 - c) *F 3* - Mass-media e informazione: RAI - Emittenti TV / radio private: nazionali e locali - Stampa
 - d) *F 4* - Volontariato: Dipartimento PC - Associazioni locali, provinciali, regionali, nazionali
 - e) *F 5* - Materiali e mezzi: C.A.P.I. - Ministero dell'Interno - Sist. Mercurio . FF.AA - C.R.I. - Aziende pubbliche e private - Volontariato
 - f) *F 6* - Trasporti, circolazione e viabilità: FF.SS. - Trasporto gommato, marittimo, aereo - ANAS - Soc. Autostrade - Province - Comuni - ACI

2 Stato dell'arte

- g) *F 7* – Telecomunicazioni: Ente Poste - Ministero delle Telecomunicazioni - Acconto di Telecomunicazioni
- h) *F 8* - Servizi essenziali: ENEL - ENI - Gas - Acquedotto - Aziende Municipalizzate - Sistema bancario - Distribuzione carburante - Attività scolastica
- i) *F 9* - Censimento danni a persone e cose: Attività produttive (ind., art., comm.) - Opere pubbliche - Beni culturali - Infrastrutture - Privati
- j) *F 10* - Strutture operative: Dipartimento PC - VV.F. - Forze Armate - C.R.I. - C.C. - G.D.F. - Forestale - Capitanerie di Porto - P.S. - Volontariato - CNSA (CAI)
- k) *F 11* - Enti locali: Regioni - Province - Comuni - Comunità montane
- l) *F 12* - Materiali pericolosi: VV.F. - C.N.R. - Depositi e industrie a rischio
- m) *F 13* - Assistenza alla popolazione: Forze Armate - Ministero dell'Interno - C.R.I. - Volontariato - Regioni - Province - Comuni
- n) *F 14* - Coordinamento centri operativi: Collegamento con i centri operativi misti - Gestione delle risorse - Informatica

2.2 Scenari di rischio

Come è stato anticipato nelle sezioni precedenti, gli Scenari di Rischio sono utili strumenti che vengono utilizzati in fase di Preparedness del ciclo di vita dell'emergenza e la loro definizione e creazione rappresenta la prima fase del Metodo Augustus. Nella fase di Preparedness gli scenari di rischio sono il passo successivo ad uno studio approfondito sul territorio e sull'emergenza che può eventualmente verificarsi.

Secondo la delibera della regione Lombardia numero 46001: *“Con il termine “scenario” si definisce una descrizione sintetica, accompagnata da una cartografia esplicativa dei possibili effetti sull'uomo o sulle infrastrutture presenti in un territorio di evenienze meteorologiche avverse (piene ed inondazioni), di fenomeni geologici o naturali (frane e valanghe), di incendi boschivi, oppure di incidenti industriali o a trasporti di sostanze pericolose”*. In Figura 2.2 è riportato un esempio di scenario di rischio relativo ad uno studio geologico effettuato dal Dott.Giovanni Gavazzi per il comune di Premana. Lo scenario riporta una serie di fotografie del territorio in considerazione, su alcune delle quali sono state proiettate le specifiche dell'emergenza in esame.

Lo scenario di rischio è lo strumento intermedio tra lo studio sul territorio e il vero e proprio piano di emergenza. Attraverso lo scenario si definisce una fotografia accurata del paesaggio e delle sue strutture, sui quali viene proiettata l'eventuale emergenza. Lo scenario diventa quindi la rappresentazione futura dell'ambiente e delle sue strutture nel momento in cui vengono colpite dall'evento calamitoso definito nello studio sul territorio. Tipicamente per poter rappresentare al meglio una situazione che deve ancora verificarsi è meglio non utilizzare un solo ed unico scenario, ma è buona norma definire un set di scenari possibili che comprendano anche il cosiddetto 'worst case'. Lo scenario di rischio diventa allora il punto di partenza per definire i piani di emergenza: in base al territorio considerato, alle strutture che sono coinvolte, all'entità del danno si stila il piano di emergenza più idoneo, che contiene tutte le contromisure relative alla specifica situazione ipotizzata nello scenario di riferimento.

Scenario di rischio e piano di emergenza sono due strumenti di altissima importanza nella gestione del rischio, si può dire che l'intero ciclo di vita dell'emergenza si fonda proprio su questi due strumenti.

2 Stato dell'arte

Lo scenario di rischio è un strumento dinamico, che necessita di continui aggiornamenti e di strumenti che forniscano in tempo reale le informazioni più sensibili sul territorio. In questo modo gli scenari non restano solo delle vecchie fotografie di un periodo passato, ma risultano delle rappresentazioni fedeli della realtà che seguono i cambiamenti del territorio e si adattano bene al contesto altamente dinamico della gestione delle emergenze.

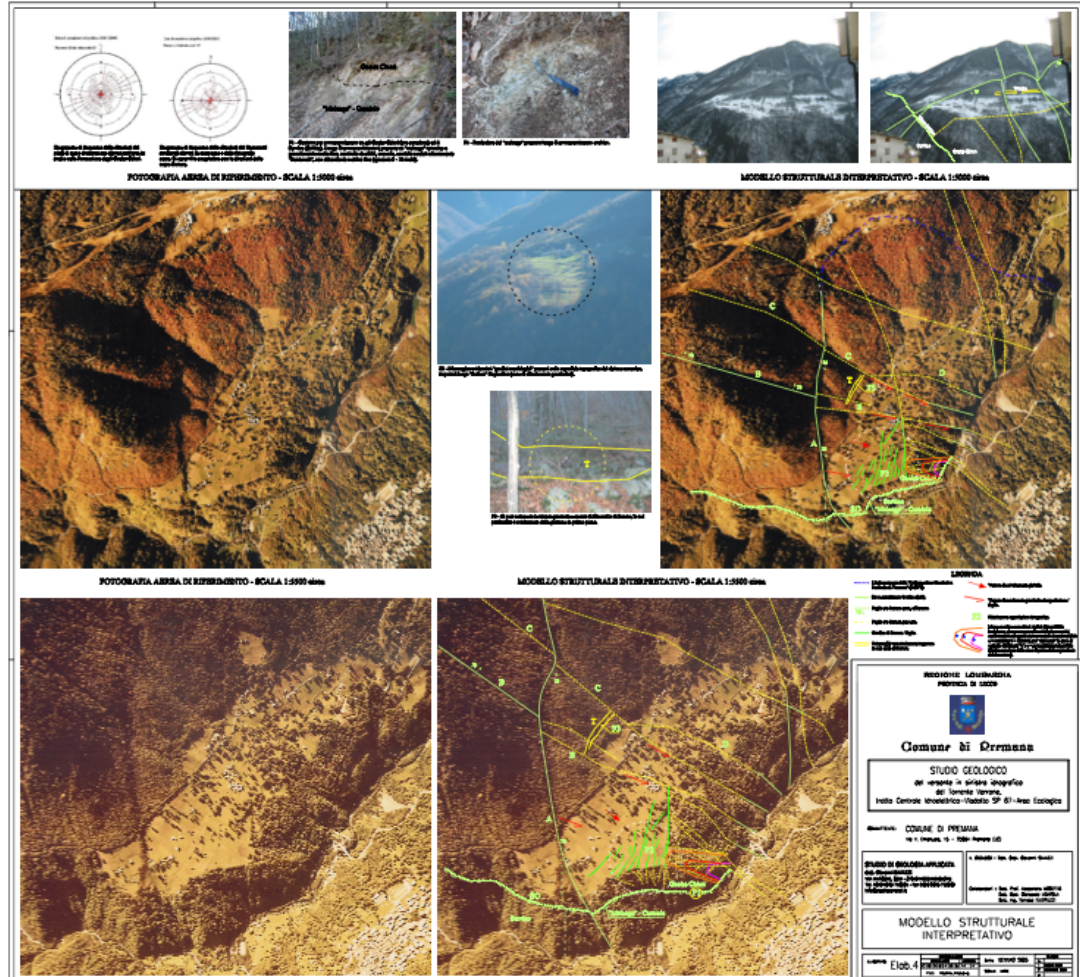


Figura 2.2: Esempio di Scenario di Rischio per il caso del Comune di Premana

2.3 Emergency Management Information Systems

Gli Emergency Management Information Systems (EMIS) sono definiti come: “*Sistemi informativi progettati per raccogliere, analizzare e condividere informazioni per il supporto alle attività di gestione dell'emergenza*”. Nella definizione sono ben sintetizzate le principali caratteristiche che identificano un EMIS¹. Innanzitutto sono Sistemi Informativi quindi implementano al loro interno tutte le caratteristiche peculiari di questa categoria di sistemi software: la completa gestione del patrimonio informativo (in questo caso legato alle tematiche ambientali/territoriali e dell'emergenza) e la capacità di fornire agli utenti, nel modo più semplice possibile, un set di servizi e tools utili.

La prima funzione, essenziale per la natura stessa di questo tipo di strumenti, è la raccolta delle informazioni. La gestione delle emergenze necessita di una grande quantità di informazioni che vanno dai dati di base sul territorio, alle mappe, ai dati storici e ancora a tutti quei dati che è essenziale avere in tempo reale, legati soprattutto al monitoraggio dell'ambiente e del mondo esterno. L'EMIS risulta essere allora un vero e proprio punto di contatto tra diverse fonti di informazioni e sorgenti dati che possono essere identificate per esempio con i diversi enti che fanno parte del ciclo di gestione dell'emergenza e che sono tipicamente indipendenti tra loro. L'EMIS può essere visto come il nucleo centrale di una rete complessa di enti che forniscono le proprie informazioni; un punto di accumulo che integra con i dati del sistema, le informazioni provenienti da più sorgenti.

L'analisi e l'elaborazione dei dati è il secondo importante aspetto che caratterizza gli EMIS. Il patrimonio informativo che sta alla base del sistema deve essere rielaborato e utilizzato per generare nuova informazione e nuovi servizi utili nella gestione dell'emergenza. Infine il terzo aspetto essenziale è la condivisione.

Le nuove informazioni e i nuovi dati che sono stati collezionati, immagazzinati e rielaborati internamente dall'EMIS sono riportati verso l'esterno, ricondivisi con la rete che gravita intorno al sistema e alla gestione dell'emergenza. Proprio quest'ultima caratteristica degli EMIS, la visione del sistema come nodo di contatto in una rete di enti, l'apertura verso la condivisione dell'informazione, insieme alle possibilità offerte dal web 2.0, sono state delle caratteristiche che hanno avuto un forte impatto sullo sviluppo del progetto legato al lavoro di tesi.

¹EMIS non è l'unico acronimo utilizzato in letteratura per identificare i sistemi informativi per le emergenze. Spesso si possono ritrovare sotto il nome di DEMIS (Dynamic Emergency Management Information Systems) o ERMIS (Emergency Response Management Information Systems) ecc. Nel documento faremo riferimento a tutti questi strumenti con l'acronimo più generico EMIS.

2.3.1 Caratteristiche generali

Categorie

Esistono diverse tipologie di EMIS, con diversi obiettivi, diverse architetture e caratteristiche. È difficile poterne dare una categorizzazione proprio perché non esistono di fatto dei parametri o delle caratteristiche comuni, tali da permettere di classificarli e raggrupparli per tipologie. È possibile stabilire una differenziazione in base alla collocazione di un EMIS in una particolare fase delle quattro che costituiscono il ciclo di vita dell'emergenza. Ci sono EMIS per la fase di *Preparedness* che tipicamente sono dedicati alla generazione di contenuti, previsioni future e conseguenti piani di emergenza. Sono sistemi che fanno ipotesi sul futuro, immaginando scenari possibili e risposte da attuare all'eventuale emergenza.

Gli EMIS che invece si collocano nella fase di *Response* sono sistemi che principalmente sono utili agli enti coinvolti nella risposta all'emergenza dopo che si è verificato l'evento calamitoso. Sono tutti quei sistemi che permettono una gestione più semplice e tempestiva dei piani di emergenza e una gestione efficace delle risorse impiegate.

Esistono poi EMIS coinvolti nella fase di *Recovery* e che sono utili nel momento in cui è terminata la fase centrale dell'emergenza e si cerca di tornare ad uno stato di "normalità" possibilmente nel minor tempo possibile. Infine vi sono EMIS che nella fase di *Mitigation* sono strumenti utili per evitare il ripetersi dell'emergenza.

Utilizzatori

Gli EMIS sono entrati a far parte della strumentazione che gli esperti di emergenze utilizzano per far fronte ai disastri e alle minacce legate all'ambiente e al territorio: le Protezioni Civili, le sale di emergenza e gli esperti di dominio sono i principali utilizzatori di questo tipo di sistemi. C'è quindi una buona parte di sistemi che sono rivolti a specialisti e professionisti del settore. Gli utilizzatori di questo tipo di EMIS lavorano quotidianamente con questi sistemi e con tutta una serie di altri mezzi necessari per la gestione dell'emergenza.

Esiste però un'altra parte di sistemi che è invece rivolta all'utente generico, quindi ad un pubblico non specializzato e non professionista, secondo la politica del *crowdsourcing*². Le persone comuni diventano parte integrante e fonte di nuova informazione per la gestione dell'emergenza. Tipicamente gli EMIS di questo tipo sfruttano gli strumenti web e la rete internet per poter offrire i loro servizi alle persone. Concludendo, gli

²la definizione di Jeff Howe si adatta bene in questo caso: "il Crowdsourcing è l'atto di prendere un lavoro tradizionalmente svolto da un impiegato designato e demandarlo a un indefinito, generalmente ampio gruppo di persone."

EMIS, nella gamma di sistemi offerti, sono sistemi potenzialmente rivolti a tutti. Sia l'esperto che la persona comune può, a modo suo, essere parte attiva nel ciclo di gestione dell'emergenza.

2.3.2 Flessibilità, affidabilità ed interoperabilità

Gli Emergency Management Information Systems, a prescindere dalla fase dell'emergenza a cui fanno riferimento ed a prescindere dall'utilizzatore finale, devono risultare soluzioni *flessibili, affidabili ed interoperabili*. La flessibilità può essere vista come una misura della robustezza esterna del sistema. È il grado di adattabilità, malleabilità in senso lato, dell'EMIS ed è stato riconosciuto come requisito 'critical' di ogni sistema informativo avanzato. Un EMIS flessibile non è fissato su una particolare implementazione o su una rigida configurazione, ma mantiene un livello di genericità sufficiente per adattarsi a nuovi scenari e nuove riconfigurazioni. Il livello di flessibilità del sistema sommato al livello di affidabilità, visto come robustezza interna, danno come risultato il livello generale di robustezza del sistema.

Un altro requisito importante è l'interoperabilità vista come la capacità dell'EMIS di cooperare con altri sistemi tramite la condivisione di informazione e servizi. L'interoperabilità, come verrà descritto più dettagliatamente nei prossimi capitoli, è garantita tramite lo sviluppo di un architettura orientata ai servizi (SOA) e tramite l'utilizzo di web services.

2.3.3 Evoluzione degli EMIS

Nel corso degli anni le tendenze, che portano a evoluzioni e cambiamenti nella struttura e nella progettazione degli EMIS, sono cambiate radicalmente. Ha senso chiedersi, in questo particolare periodo storico e con le attuali tecnologie a disposizione, quali sono i principali trends che veicolano i cambiamenti nella gestione dell'emergenza e che impattano sullo sviluppo degli EMIS.

- da una politica *reattiva* ad una *proattiva*: il terrorismo in primis, ma anche le recenti catastrofi ambientali che hanno colpito diverse nazioni causando migliaia di vittime, hanno spostato l'attenzione e aumentato l'enfasi verso la fase di Preparedness del ciclo di vita dell'emergenza. Giocare d'anticipo su problematiche di tipo ambientale non è semplice, basta vedere l'altissima difficoltà nel riuscire a prevedere i terremoti. Comunque, essere ben preparati per le eventualità e curare al massimo le fasi di previsione e predizione, massimizza il livello di prevenzione e di sicurezza.

2 Stato dell'arte

- nuove tecnologie al servizio dell'emergency management: il web 2.0, le wireless sensors networks, il cosiddetto 'Human sensing'³, l'enorme sviluppo delle tecnologie mobile e wireless, la geolocalizzazione, il GPS sono solo alcuni esempi di nuovi mezzi tecnologici che vengono oggi utilizzati nella gestione delle emergenze. Il loro utilizzo va dal monitoring ambientale alle fasi più complesse di risposta ad una particolare emergenza.
- utilizzo di tools e sistemi collaborativi: la gestione delle emergenze coinvolge un gran numero di enti specializzati che si occupano di fornire i propri servizi per contrastare l'emergenza. Spesso si tratta di enti indipendenti tra loro, ma che svolgono il proprio lavoro per un obiettivo comune. Risulta allora fondamentale la presenza di mezzi adeguati che permettano una corretta collaborazione.
- geolocalizzazione e crisis mapping: l'utilizzo di mappe digitali è da tempo entrato nell'uso comune. Le mappe interattive sono uno degli strumenti principali utilizzati nelle sale di controllo delle protezioni civili per monitorare il territorio. Questo potente strumento non è rivolto solamente ad esperti di dominio, infatti negli ultimi anni è diventato essenziale anche nella vita comune, basta pensare agli itinerari di Google Maps o ai navigatori satellitari. Gli utenti del web sono sicuramente diventati abili utilizzatori di questi mezzi. Quindi conciliare questi due aspetti è diventato naturale: persone comuni che utilizzano degli strumenti diventati oggi di uso comune per essere parte integrante nella gestione delle emergenze. Si descriveranno tra poco degli esempi reali di soluzioni e realtà di questo genere.
- crowdsourcing e aspetti 'Social' della gestione dell'emergenza: non è sempre detto che un numero maggiore di persone che si occupano di un problema di difficile soluzione possano risolverlo meglio rispetto ad un numero ristretto di persone. È vero anche che un numero alto di persone non specializzate non è in grado di dare lo stesso apporto di un gruppo ristretto di persone esperte di dominio. Questo discorso può valere in generale per diversi ambiti, ma nel caso dell'emergency management invece, creare una rete sociale intorno ad una particolare emergenza, avere un alto numero di persone che possono dare il loro contributo, per esempio per dare testimonianza tempestiva di situazioni pericolose o per avvisare altre persone di un pericolo imminente, può essere di vitale importanza. È su questo concetto e

³sfruttando smartphones, fotocamere e videocamere le persone possono diventare sensori che istantaneamente monitorano un particolare evento e lo rendono disponibile alla comunità.

su questi trends che si basano progetti reali come il Crisis Mappers Network⁴ o il progetto Ushahidi⁵.

2.3.4 Progetti interessanti

Ushahidi

Ushahidi è un progetto che nasce da una compagnia no-profit specializzata nello sviluppo di software open-source per la raccolta, la visualizzazione e il mapping interattivo. I tools che sviluppano mirano a fornire alla popolazione degli strumenti per poter creare report e raccontare storie di eventi realmente accaduti, al fine di testimoniare⁶ delle emergenze sul territorio. Ogni report, dopo una valutazione della veridicità e dopo una fase di filtering, viene pubblicato e geotaggato su una mappa interattiva, mappa che di volta in volta si arricchisce di nuovi tags che testimonieranno nuovi eventi e nuove storie raccontati dai cittadini. La compagnia mette a disposizione diversi strumenti che possono essere utilizzati per creare veri e propri portali legati ad una particolare emergenza o ad un fatto che ha in qualche modo creato disagi alla popolazione e al territorio. Basti pensare che la prima implementazione di Ushahidi nasce proprio dall'idea di un piccolo gruppo di sviluppatori che creano un'applicazione web, dopo le elezioni in Kenya del 2008, per testimoniare le violenze avvenute nella popolazione. L'applicazione web registrò in poco tempo l'utilizzo da parte di circa 45.000 utenti, segno della forte necessità di un mezzo rivolto a tutti per testimoniare in modo semplice le problematiche del territorio e utile come fonte informativa per gli aiuti alla popolazione.

Sahana

Il progetto Sahana⁷ nasce e viene sviluppato da un gruppo di membri della Sri Lankan IT Community dopo il terremoto e il successivo tsunami del 2004 nell'Oceano Indiano che colpì l'intero sud-est asiatico fino ad arrivare alle coste orientali dell'Africa. La Sahana Foundation ha progettato e sviluppato un insieme di tre tools per il disaster management ognuno con uno specifico compito:

- Eden Project: piattaforma per la gestione e la creazione di soluzioni efficaci per la gestione degli aiuti umanitari prima o durante un periodo di crisi.
- Vesuvius Project: progetto utilizzato nelle fasi di Preparedness e Response volto all'aiuto della comunità medica e alla riunificazione delle famiglie.

⁴<http://crisismappers.net/>

⁵<http://www.ushahidi.com/>

⁶Ushahidi in Swahili significa proprio 'testimone'

⁷<http://sahanafoundation.org/>

- Mayon Project: è un progetto in via di sviluppo che sarà disponibile a fine 2011. Esso mira a fornire una soluzione altamente scalabile per la gestione del personale e delle risorse coinvolti nelle emergenze.

DM-OPEN&IPAWS

La FEMA (Federal Emergency Management Agency) è l'organo del dipartimento di sicurezza nazionale degli Stati Uniti che si occupa, come descritto nella mission sul sito ufficiale⁸, del supporto ai cittadini per dare una rapida risposta per la preparazione, la protezione, l'azione e la mitigazione verso tutti i tipi di rischi e di emergenze. La FEMA si è occupata di creare il Disaster Management Program, un progetto volto alla completa gestione di qualsiasi tipo di disastro ambientale o emergenza. Il programma comprende diverse architetture come DM-Quadro e DM-OPEN(Disaster Management Open Platform for Emergency Networks). DM-OPEN è stato pensato per creare una base comune per lo scambio di informazioni e la collaborazione tra sistemi di emergency management già esistenti. Il progetto comprende quattro web service attivi e disponibili per gli utilizzatori:

1. Admin Interface: interfaccia web service per la gestione degli utenti.
2. CAP (Common Alerting Protocol) SOAP interface: interfaccia SOAP⁹ del protocollo per l'invio la ricezione di messaggi d'emergenza.
3. HazCollect NWEM¹⁰ interface: interfaccia per l'invio di messaggi CAP specializzati creata appositamente per sistemi HazCollect.
4. EDXL DE¹¹ Interface - WSDL: interfaccia per l'invio di messaggi EDXL-DE.

Il progetto DM-OPEN verrà sospeso a fine agosto 2011 per essere sostituito da un nuovo progetto: IPAWS (Integrated Public Alert and Warning System) un nuovo sistema di allerta efficiente, affidabile, integrato e flessibile al servizio della popolazione degli stati uniti.

⁸<http://www.fema.gov/about/index.shtm>

⁹Simple Object Access Protocol: protocollo leggero inteso per lo scambio di informazioni strutturate in un ambiente decentralizzato e distribuito.

¹⁰Non-Weather Emergency Message: messaggio CAP specializzato per i sistemi HazCollect.

¹¹Emergency Data Exchange Language Distribution Element: formato per lo scambio di messaggi XML per le emergenze

2.4 Conclusioni e considerazioni

Il quadro generale della gestione dell'emergenza è ampio e molto complesso. La sola presenza di quattro diverse fasi per la corretta gestione di situazioni rischiose e dannose per l'ambiente e per l'uomo, lascia già intendere come sia importante attuare in modo intelligente dei piani di emergenza secondo la politica del *divide et impera*: focalizzarsi sulla singola fase e dividere l'emergenza in sottoproblemi è necessario per attuare piani efficaci nel momento del bisogno.

Ogni singola fase presenta caratteristiche e problematiche specifiche che vanno affrontate con i mezzi adeguati e con una conoscenza del contesto dettagliata e multidisciplinare. Le quattro fasi del ciclo di vita dell'emergenza hanno lo stesso grado di importanza, e la buona o cattiva gestione dell'una si riflette inevitabilmente sulle altre.

Tutta la letteratura, lo storico degli eventi calamitosi del passato, le tecniche e gli strumenti efficaci nella gestione dell'emergenza, come il Metodo Augustus, sono un patrimonio importante sia da un punto di vista culturale, che da un punto di vista di efficienza nella risposta ai danni provocati dall'ambiente e dall'uomo.

Gli Scenari di Rischio e gli EMIS si collocano in questo difficile contesto e sono strumenti utili in tutte le quattro fasi del ciclo di vita dell'emergenza. In questo capitolo sono stati presentati nel modo più generico possibile, in particolare descrivendone le caratteristiche di base, le linee evolutive e alcuni casi reali per fissarne meglio il concetto astratto. Dei progetti reali descritti si riporta in Tabella 2.2 un confronto delle caratteristiche principali.

	FASE DEL CICLO DI EMERGENZA	UTENTI	FUNZIONALITÀ
Ushahidi	Response e Mitigation	Cittadini	Generazione di report
Sahana	Response	Organizzazioni e Cittadini	Tools di gestione aiuti umanitari e ricerca dispersi
DM-OPEN	Response	Organizzazioni e Cittadini	Backbone per sistemi per scambio di informazioni
SocialeMIS	Preparedness e Response	Esperti di emergenze	Definizione e ricerca scenari di emergenza

Tabella 2.2: Confronto tra i diversi EMIS

3 Strumenti tecnologici

3.1 Vaadin

Vaadin¹ è un framework per lo sviluppo di applicazioni AJAX lato server, con la peculiarità di fornire allo sviluppatore un nuovo modo di costruire applicazioni implementando solo ed esclusivamente classi Java, anche per la parte di presentation layer. La libreria Vaadin offre allo sviluppatore un set di componenti standard pronti per l'uso e permette la creazione di componenti aggiuntivi customizzabili e personalizzabili. La creazione di una web application tramite Vaadin solleva lo sviluppatore dall'utilizzo e la conoscenza di tecnologie lato client come l'HTML o il CSS o AJAX stesso. Il nuovo concetto nello sviluppo di applicazioni web tramite Vaadin è proprio l'idea che lo sviluppatore dimentica la componente web e implementa il codice dell'applicazione come se stesse sviluppando una normale desktop application. In questo modo lo sviluppatore non deve preoccuparsi di dover ricercare e successivamente utilizzare le tecnologie web e si può concentrare esclusivamente sulla logica di business essendo più produttivo e più veloce.

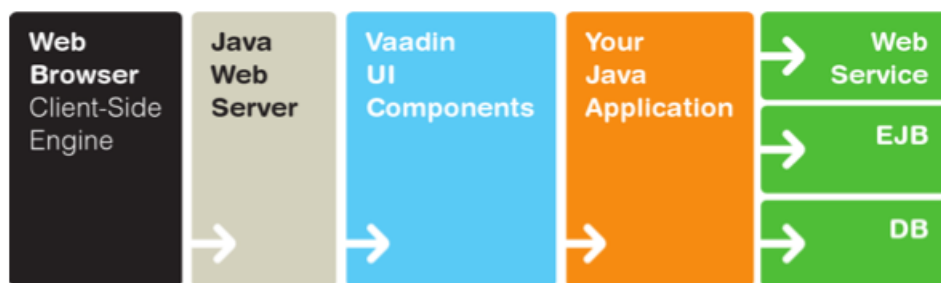


Figura 3.1: Architettura web application sviluppata con Vaadin

Nella Figura 3.1 è schematizzata l'architettura di base di una web application costruita con Vaadin. Vaadin struttura l'applicazione in questo modo: da una parte il Client-side

¹<http://vaadin.com/home>

engine che gira sul browser come programma Javascript e che definisce la user interface, renderizzata tramite GWT(Google Web Toolkit) e riporta la user interaction al server. Dall'altra parte il framework server-side che riceve le azioni dell'utente e si interfaccia con la logica di business e la logica dati. Il web browser può essere visto come un thin client che si occupa solo ed esclusivamente del presentation layer.

3.2 RESTful Web Services

3.2.1 REST

REST sta per Representational State Transfer ed è un paradigma che utilizza un'architettura client-server stateless in cui i web services sono visti come risorse e possono essere identificati tramite il loro URL. Il client utilizza di fatto i web services per avere accesso a queste risorse e in particolare utilizza un set ristretto di metodi da utilizzare sulle risorse stesse. Le web application che necessitano di accedere a risorse web possono farlo ad esempio attraverso i metodi GET, POST, PUT e DELETE del protocollo HTTP. L'idea del paradigma REST nasce da Roy Fielding uno dei padri del protocollo HTTP che identificò REST come un sistema per poter descrivere ed identificare delle risorse web. Possiamo allora dire che gli elementi che costituiscono REST sono: le risorse e i metodi per accedere ad esse.

1. *Risorse*: REST mette al centro del suo paradigma il concetto di risorsa. Ogni risorsa sul web è identificata univocamente con un URL, basta pensare alle pagine web.
2. *Metodi*: Get, Post, Put e Delete sono i metodi HTTP utilizzati per accedere alle risorse.

Gli elementi del paradigma REST possono essere schematizzati come in Figura 3.2.

Durante un'interazione tra client e server quello che viene trasferito è una rappresentazione dello stato interno della risorsa. Perché un Web Service sia conforme alle specifiche REST deve avere alcune caratteristiche specifiche :

- architettura basata su client e server.
- stateless. Ogni ciclo di request\response deve rappresentare un'interazione completa del client con il server. In questo modo non è necessario per mantenere informazioni sulla sessione utente, minimizzando l'uso di memoria del server e la sua complessità.

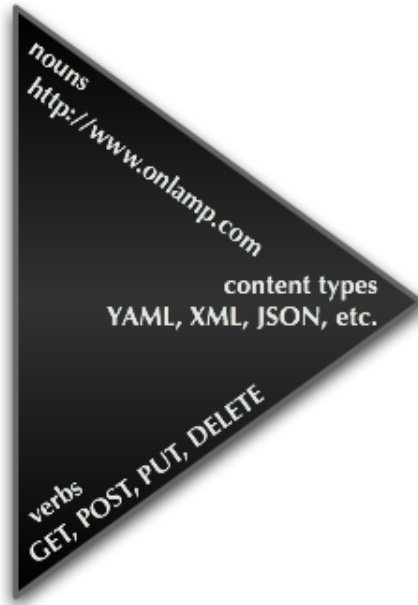


Figura 3.2: Triangolo paradigma REST

- uniformemente accessibile. Ogni risorsa deve avere un'indirizzo univoco e ogni risorsa di ogni sistema presenta la stessa interfaccia, precisamente quella individuata dal protocollo HTTP.

3.3 JSON

JSON (JavaScript Object Notation) è un sottoinsieme del linguaggio JavaScript ed è un formato che viene utilizzato per lo scambio di dati tra applicazioni client-server. È un formato assolutamente indipendente dal linguaggio di programmazione dell'applicazione che ne fa uso e le strutture che compongono il linguaggio sono molto semplici. Queste sue caratteristiche fanno di JSON uno strumento molto utilizzato nello scambio di informazioni e dati.

3.3.1 Strutture del formato

La notazione JSON si basa su due sole strutture:

1. un oggetto: caratterizzato da una serie di coppie *nome:valore* racchiuse tra due parentesi graffe come mostrato in Figura 3.3.

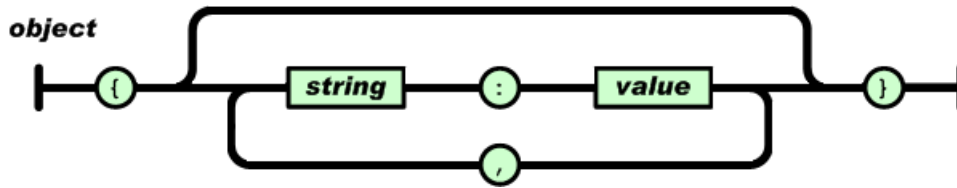


Figura 3.3: JSON object

2. un array: una serie ordinata di valori separati da virgola e racchiusi tra parentesi quadre come mostrato in Figura 3.4.

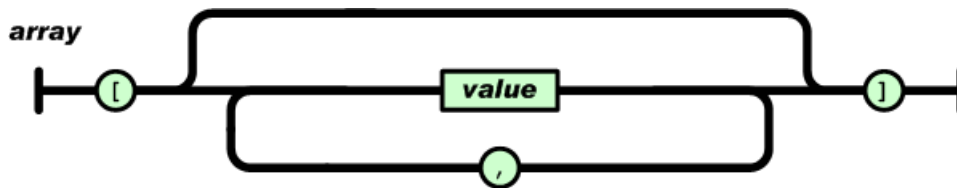


Figura 3.4: JSON array

Un valore può essere rappresentato da una stringa, un numero, un valore booleano, un oggetto o un array, come mostrato in Figura 3.5.

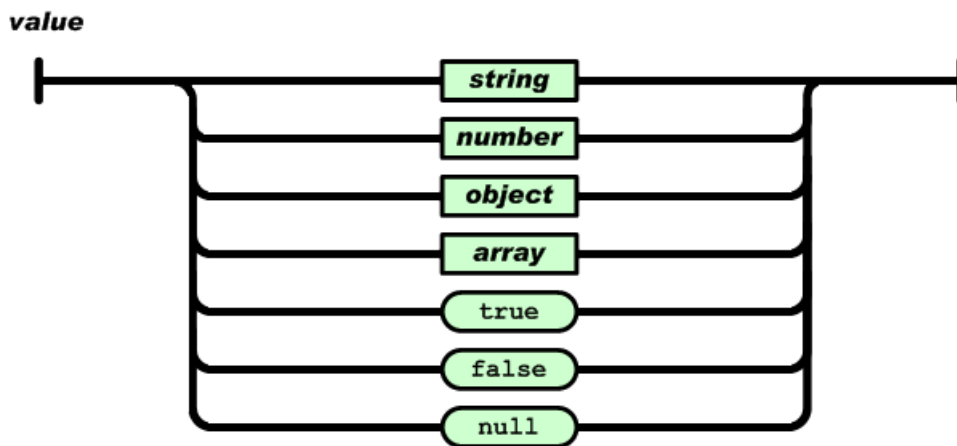


Figura 3.5: JSON values

3.3.2 JSON e XML

XML è stato, negli anni passati, il formato standard per lo scambio di dati tra applicazioni indipendenti. JSON si presenta come una alternativa ad XML che offre gli stessi

vantaggi di interoperabilità e di indipendenza dalle applicazioni, ma con una struttura del linguaggio nettamente più semplice. Infatti JSON ha una grammatica molto più snella e mappa direttamente sulle strutture dati usate oggi nei linguaggi di programmazione. Il linguaggio è molto più leggibile rispetto ad XML perché il linguaggio di quest'ultimo si basa sull'utilizzo di tag che non sono invece utilizzati in JSON. Per di più la struttura del documento XML richiede la traduzione dei dati in un documento. Infine, per precisione, bisogna specificare che XML è uno strumento utile per lo scambio di documenti, mentre JSON è uno strumento utile per lo scambio di dati.

3.4 KML

Il Keyhole Markup Language è una grammatica XML che viene utilizzata per la creazione di file con estensione .kml, utili per memorizzare dati geografici come markers, linee, poligoni, aree georeferenziate. Questi files possono essere visualizzati su Google Maps oppure su Google Earth. I file kml sono creati mediante un linguaggio a tag e vengono interpretati ed elaborati da Google Maps per rappresentare sulla mappa gli oggetti memorizzati nel file. Tra gli elementi che è possibile memorizzare e successivamente visualizzare in Google Maps elenchiamo:

- Placemarks
- Icone
- Poligoni e linee
- Layers

Nella figura 3.5 è stato riportato uno stralcio di un file kml utilizzato per rappresentare uno scenario di rischio, generato attraverso l'applicazione SocialEMIS.

Tutti questi oggetti possono essere customizzati e devono obbligatoriamente contenere le indicazioni di latitudine e longitudine per poter essere correttamente fissati sulla mappa. La versione 2.2 ha ricevuto il consenso dell'OGC (Open Geospatial Consortium) ed è stato adottato come implementazione standard. KML è complementare a molti OGC standards esistenti come GML(Geography Markup Language), WFS(Web Feature Service) e WMS(Web Map Service). Alcuni elementi utilizzati in KML derivano proprio da GML versione 2.1.2 e l'OGC e Google sono d'accordo che ci saranno future armonizzazioni tra KML e GML.

3 Strumenti tecnologici

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<kml xmlns="http://www.opengis.net/kml/2.2" xmlns:gx="http://www.google.com/kml/ext/2.2" xmlns:atom="h
  <Document>
    <Placemark>
      <name></name>
      <description>Map</description>
      <Point>
        <coordinates>12.517011165618896,41.94422606500535</coordinates>
      </Point>
    </Placemark>
    <Placemark>
      <name>Scuola</name>
      <description>Struttura</description>
      <Style>
        <IconStyle>
          <scale>0.0</scale>
          <heading>0.0</heading>
          <Icon>
            <href>http://google-maps-icons.googlecode.com/files/school.png</href>
            <refreshInterval>0.0</refreshInterval>
            <viewRefreshTime>0.0</viewRefreshTime>
            <viewBoundScale>0.0</viewBoundScale>
          </Icon>
        </IconStyle>
      </Style>
      <ExtendedData>
        <Data name="Taken">
          <value>taken</value>
        </Data>
      </ExtendedData>
      <Point>
        <coordinates>12.517697811126709,41.944122324095034</coordinates>
      </Point>
    </Placemark>
    <Placemark>
      <name>Piscina</name>
      <description>Struttura</description>
      <Style>
```

Figura 3.6: Esempio del contenuto di un file kml

3.5 JPA

Nello sviluppo di applicazioni e software informatici è necessario utilizzare soluzioni semplici e rapide per la gestione della persistenza dei dati. Per lo sviluppatore è essenziale avere un framework che implementi le funzioni CRUD necessarie per gestire la persistenza. JPA² (Java Persistence API) trasferisce l'idea dell'Object Orienting sui database relazionali attraverso un object/relational mapping, astruendo dal particolare database o dal linguaggio di querying e fornendo un'interfaccia standard per l'interazione con i dati. Il funzionamento di JPA prevede il mapping delle entità contenute nel database in *Entity Classes*, cioè classi Java leggere che identificano l'oggetto e che contengono solamente un costruttore di default e i metodi getter e setter. La gestione di queste classi avviene attraverso l'utilizzo di classi *Controller* o *Manager* che mettono a disposizione tutti i metodi necessari per l'interazione con il database.

²definita nella specifica JSR-000317 del 10 dicembre 2009

3.6 Google Maps

Google Maps è il famosissimo servizio di web mapping offerto da Google. Le sue funzionalità e le sue caratteristiche sono ormai note e chiunque abbia, almeno una volta, provato a navigare tra le strade di Google Maps avrà sicuramente potuto giovare della semplicità, della velocità e della qualità di questo servizio. L'interfaccia diventata ormai famosa ed inconfondibile è riportata in Figura 3.7. Il progetto viene lanciato nel febbraio 2005 ed è stato il pioniere di questo tipo di web application. Sono stati tanti i progetti sviluppati per emulare Google Maps, infatti è facile trovare sul web delle buone alternative ad esso. È un progetto in continuo sviluppo che anno dopo anno continua ad offrire sempre nuovi servizi. Una delle caratteristiche più apprezzate, soprattutto dal mondo degli sviluppatori è l'ampia scelta di API³ con cui è possibile inserire le funzionalità di Google Maps all'interno dei propri siti e delle proprie web applications, dalle Static Maps API, ai Web Services alle API per Javascript e Flash. Grazie alle API anche la comunità di Vaadin, ha potuto sviluppare un Add-on per il proprio framework che copre la maggior parte delle funzionalità presenti in Google Maps.



Figura 3.7: Google Maps

³<http://code.google.com/intl/it-IT/apis/maps/index.html>

4 SocialEMIS

Dopo aver inquadrato il progetto SocialEMIS da un punto di vista funzionale e tecnologico, in questo capitolo si descrive, da un punto di vista ingegneristico, il lavoro di pianificazione e progettazione della web application. In particolare nella prima sezione del capitolo verranno descritte, ad alto livello, le funzionalità offerte del sistema analizzando SocialEMIS proprio dal punto di vista dell'utilizzatore finale. Nella seconda sezione invece, spostando il focus della descrizione dall'utilizzatore del sistema al sistema vero e proprio, verranno raccolti in maniera testuale e successivamente analizzati e specificati, attraverso il linguaggio di modellazione UML, tutti i requisiti di sistema. Nella terza sezione verrà fornita una descrizione dei moduli software e i dati, attraverso un diagramma Entità-Relazione, che i moduli utilizzano. La quarta sezione, più tecnica, alla luce delle considerazioni fatte precedentemente, presenterà le diverse fasi di progettazione del sistema attraverso la descrizione dell'intera architettura SocialEMIS. L'ultima sezione del capitolo sarà invece completamente dedicata alla descrizione degli algoritmi implementati per le diverse tipologie di ricerca di scenari di rischio, valore aggiunto di SocialEMIS.

4.1 Pianificazione e approccio allo sviluppo

4.1.1 Panoramica delle funzionalità di SocialEMIS

Il prototipo SocialEMIS deve fornire all'utilizzatore finale, quindi nella fattispecie all'esperto di dominio, un tool con le seguenti macrofunzionalità:

1. **DESIGN DI SCENARI DI RISCHIO:** la funzionalità di design permette all'utilizzatore di creare e memorizzare uno o più nuovi scenari di rischio o eventualmente di apportare delle modifiche ad uno scenario che ha precedentemente salvato. L'esperto di dominio con questa funzionalità ha a disposizione un mezzo per poter modellare, attraverso un'applicazione web, una particolare situazione reale, uno scenario che potrebbe verificarsi tra quelli che ha ipotizzato durante lo studio di un particolare caso di emergenza ambientale. Abbiamo visto nel capitolo relativo allo stato dell'arte, nella sezione dedicata agli Scenari di Rischio, l'importanza

che svolge ai fini di una corretta gestione dell'emergenza, lo strumento 'scenario di rischio'.

È stato evidenziato proprio l'enorme vantaggio che si ha nel poter disporre di diversi scenari di emergenza, per poter ipotizzare e così essere preparati, ad un certo evento calamitoso che si verifica in una determinata zona. SocialEMIS propone all'utilizzatore un modo semplice per poter creare da zero gli scenari di emergenza necessari per un certo caso di studio: in pochi click e attraverso strumenti di uso comune come mappe interattive e markers l'esperto può definire una 'fotografia' digitale della realtà.

La funzionalità di design prevede anche la possibilità di poter aggiornare nel caso ce ne fosse la necessità, gli scenari precedentemente salvati. Questo fa sì che lo scenario di rischio soddisfi il requisito di essere un elemento dinamico e non fisso su una determinata configurazione, requisito necessario proprio perché l'oggetto rappresentato dallo scenario, quindi l'ambiente ed il territorio, è un elemento fortemente dinamico e soggetto a continui cambiamenti.

La creazione di un nuovo scenario di rischio può essere sintetizzata nei seguenti semplici passi:

- identificazione sulla mappa dell'area che potrebbe essere interessata da un'emergenza ambientale.
- caricamento delle strutture presenti nell'area selezionata (come scuole, fabbriche, ecc.) e posizionamento tramite markers sulla mappa
- richiesta di informazioni utili, ai servizi partner e agli enti coinvolti nel ciclo di vita dell'emergenza
- creazione e localizzazione dell'emergenza ambientale, tramite marker e superficie colpita, sulla mappa
- identificazione e segnalazione delle strutture sul territorio che hanno subito danni in base all'emergenza definita

Gli scenari di rischio generati dall'utilizzatore sono memorizzati dal sistema e sono proprietà dell'utilizzatore stesso, nel senso che l'utilizzatore è l'unico che ha i permessi per poterli modificare o eventualmente eliminare. Lo scenario di rischio salvato però, entra a far parte di un sistema di *condivisione di scenari di rischio*: infatti secondo la politica Social definita nel primo capitolo del documento, tutti gli utilizzatori di SocialEMIS possono usufruire di uno strato di informazioni comuni che è rappresentato proprio dall'insieme di tutti gli scenari salvati dai diversi

utilizzatori di SocialEMIS. SocialEMIS oltre a fornire un client semplice per la gestione degli scenari di rischio mette anche a disposizione un architettura che si basa su un patrimonio informativo condiviso intorno al quale collaborano, si potrebbe quasi dire 'involontariamente', tutti gli utilizzatori dell'applicazione; infatti questo approccio porta alla formazione di una vera e propria rete sociale che ruota intorno al concetto di scenario di rischio. Questo aspetto dell'applicazione SocialEMIS è molto importante anche relativamente alla seconda macrofunzionalità del sistema. In Figura 4.1 è schematizzata la funzionalità di Design Scenario.

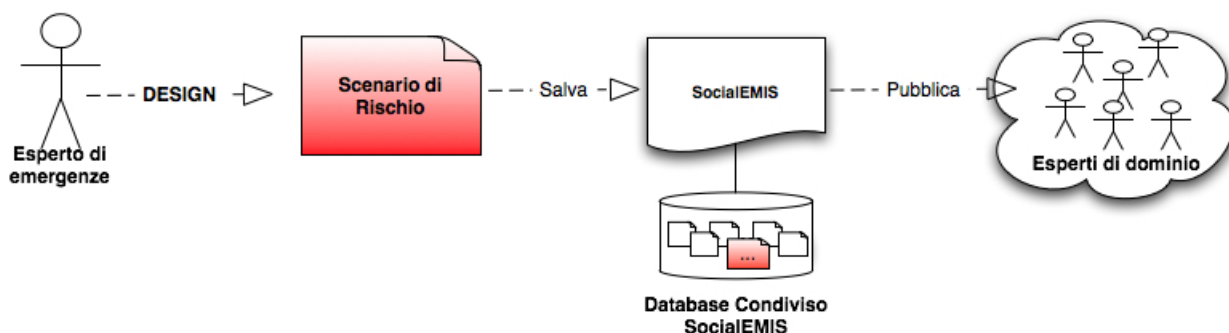


Figura 4.1: Schematizzazione funzionalità di Design Scenario

2. **RICERCA DI SCENARI DI RISCHIO:** la seconda funzionalità che offre SocialEMIS è la possibilità di poter effettuare delle ricerche tra tutti gli scenari salvati dagli utilizzatori dell'applicazione. La funzionalità di ricerca è possibile proprio grazie alla politica 'social' di cui è stato discusso precedentemente e che permette ad un esperto di dominio di poter visualizzare come risultati di una ricerca, degli scenari selezionati tra tutti gli scenari creati tramite SocialEMIS. Questa funzionalità ha un duplice obiettivo:

- a) è utile in fase di design come strumento di consultazione nello storico degli scenari, per poter definire un nuovo scenario e prendere così spunto ed avere come riferimento gli scenari di rischio definiti da altri esperti di dominio.
- b) è utile nel momento in cui si vuole avere il set di scenari più *simili* ad una data situazione che si potrebbe verificare, ad esempio se si è prossimi ad una eventuale emergenza: in questo caso l'utilizzatore, definendo attraverso l'interfaccia del client SocialEMIS alcuni parametri di ricerca, può ottenere come risultato una selezione degli scenari di rischio più simili.

Uno schema della funzionalità di ricerca è riportato in Figura 4.2.

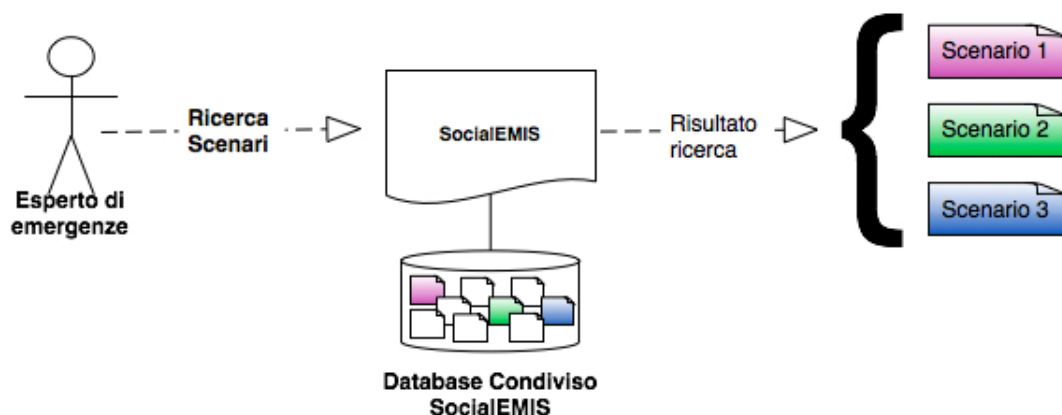


Figura 4.2: Schematizzazione funzionalità di Ricerca Scenario

Un aspetto importante delle macrofunzionalità di ricerca di scenari è che essa è suddivisa secondo tre diverse tipologie di ricerca; ogni tipologia implementa un diverso *algoritmo di similarity* per selezionare gli scenari più simili all'interno dell'intero pool di scenari. All'utilizzatore infatti sono rese disponibili tre modalità differenti di ricerca, questo per far sì che la ricerca stessa sia il più possibile adatta ai bisogni e alle necessità dell'esperto di dominio.

- *Ricerca con singolo parametro*: la prima tipologia di ricerca permette all'utilizzatore di impostare una ricerca, sulla base di un singolo parametro e cioè la tipologia di emergenza. L'utilizzatore in base a delle considerazioni che possono essere legate al particolare caso di studio oppure ad una rilevazione della rete di sensori, definisce attraverso l'applicazione un'emergenza. L'utilizzatore può così invocare la funzione di ricerca di scenari simili che, attraverso l'opportuno algoritmo di ricerca, andrà a selezionare un gruppo di scenari che possono essere simili, e quindi nei quali è definita la stessa emergenza definita dall'utilizzatore. Questo tipo di ricerca, facendo riferimento al ciclo di vita dell'emergenza, è utile in fase di *Preparedness*. Proprio in questa fase infatti, l'esperto di dominio che si dedica alle operazioni di definizione di un nuovo scenario, volendosi documentare con scenari già definiti da altri utenti, che riportano emergenze in zone non necessariamente vicine a quella definita dall'esperto, può trarre ispirazione e seguire le linee guida da trasferire sul suo

scenario e sulla definizione del piano di emergenza.

- *Ricerca con doppio parametro*: la tipologia di ricerca con doppio parametro segue la stessa struttura della ricerca precedente, ma utilizza un algoritmo di ricerca differente. L'algoritmo è a doppia passata, nel senso che genera una prima selezione di tutti e soli gli scenari che riportano la stessa emergenza definita dall'utente (primo parametro) e successivamente in una seconda fase seleziona tra questi, un set degli scenari più vicini geograficamente allo scenario definito dell'utente. La distanza è il secondo parametro di ricerca. La ricerca in questo caso è indirizzata al ritrovamento di scenari simili a quello definito dall'utente. L'idea è che l'esperto di dominio abbia già definito una serie di scenari di emergenza per un dato caso di studio. Con il presentarsi, o con la possibilità che si possa presentare l'emergenza del caso in esame, l'esperto vuole conoscere quali tra gli scenari che ha definito sono quelli che si avvicinano di più allo scenario di riferimento. Questo secondo tipo di ricerca, facendo riferimento al ciclo di vita dell'emergenza, si colloca o nella fase di *Preparedness* o nella fase di *Response*, in base al fatto se si sia già verificata o meno l'emergenza in esame. La ricerca in questo caso, a differenza della ricerca con singolo parametro, è molto più precisa proprio perché la necessità dell'utilizzatore è quella di trovare lo scenario che si avvicina il più possibile a quello di riferimento. Per questo tipo di ricerca i dati provenienti dalla Wireless Sensor Network svolgono un ruolo molto importante. Lo scenario di riferimento su cui verrà successivamente impostata la ricerca nel database dovrà essere definito in base alle ultime misurazioni percepite dai sensori della rete. I dati che provengono dalla sensor network, letti ed interpretati dall'esperto di dominio, portano all'impostazione della ricerca per una emergenza, di un certo tipo, di una certa intensità ed in un certo luogo. Anche in questo caso la possibilità di avere a disposizione un flusso dati preciso e sempre aggiornato proveniente da un sistema indipendente come una Wireless Sensor Network aumenta la qualità e la precisione delle funzionalità offerte dal sistema.
- *Ricerca 'Multirischio'*: il multirischio è definito tramite l'identificazione, per un certo scenario, di eventuali emergenze secondarie che sono generate dall'emergenza iniziale, l'emergenza primaria. Sono infatti frequenti casi in cui presenta questo tipo di situazione e casi addirittura in cui le emergenze secondarie hanno un impatto ancora più distruttivo rispetto all'emergenza che ha dato loro origine. Per questo la ricerca multirischio implementa un algoritmo che seleziona gli scenari, dal pool di scenari condivisi, avendo come parame-

tri di ricerca le diverse categorie e tipologie di emergenze definite dall'user. Quest'ultima tipologia di ricerca, esattamente come nel caso precedente, può essere utile sia in fase di *Preparedness* che in fase di *Response* del ciclo di vita dell'emergenza.

La macrofunzionalità di ricerca offre allora un set di sottofunzionalità in cui si possono effettuare ricerche all'interno del pool di scenari condivisi, con diversi gradi di precisione e con diverse finalità. La ricerca con singolo parametro è una ricerca più generica che tipicamente può essere utilizzata dall'esperto che ha necessità di trovare situazioni simili a quella ricercata, ma che non necessariamente si trovano nella stessa zona. Una ricerca di questo tipo è utile per poter visualizzare per esempio che piani di emergenza sono stati progettati per situazioni che ipotizzano la stessa emergenza. La seconda tipologia di ricerca invece è sulla falsa riga della prima, ma l'algoritmo di ricerca è più selettivo, restringe il campione di scenari possibili in quanto specializza e raffina la ricerca inserendo come parametro anche la posizione dell'emergenza. Infine la ricerca multirischio è utile per le situazioni più complesse, quelle dove l'emergenza iniziale può generare altre emergenze a effetto domino.

Queste sono le funzionalità che caratterizzano il sistema, che ne identificano l'utilità e che specificano gli obiettivi del sistema. Accanto a queste ci sono ovviamente tutte le funzionalità accessorie tipiche delle web application.

4.2 Raccolta e analisi dei requisiti

4.2.1 Raccolta dei requisiti

In questa parte verranno descritti in forma testuale le diverse tipologie di requisiti che sono stati raccolti prima della fase di progettazione e di sviluppo di SocialEMIS. I requisiti saranno suddivisi in requisiti iniziali, che descrivono ad alto livello che caratteristiche deve avere SocialEMIS per raggiungere gli obiettivi definiti nel capitolo introduttivo del documento. Si passerà poi alla definizione dei requisiti funzionali che descrivono le funzionalità che devono essere implementate dal sistema. Infine verranno elencati i requisiti architetturali per specificare come è stata strutturata la web application.

- *Requisiti iniziali*: SocialEMIS mira ad essere una applicazione rivolta ad un gruppo di utenti ristretto, cioè gli esperti di gestione delle emergenze come gli operatori delle sale di controllo delle Protezioni Civili. SocialEMIS dovrà essere uno strumento che si aggiunge alla strumentazione attualmente in uso nelle sale di emergenza.

L'applicazione dovrà essere progettata ed implementata cercando di superare alcuni limiti e di colmare alcune mancanze che caratterizzano gli strumenti software utilizzati attualmente dalle sale di emergenza, come l'integrazione di diversi flussi dati provenienti da altri sistemi e l'interoperabilità con questi stessi sistemi.

- *Requisiti funzionali*: le funzionalità dell'applicazione SocialEMIS ruoteranno tutte intorno allo strumento Scenario di Rischio. SocialEMIS dovrà poter fornire all'utilizzatore tutte le funzionalità necessarie per poter creare, gestire e lavorare con questo potente strumento che, come è stato precedentemente descritto, svolge un ruolo importante nel ciclo di vita dell'emergenza. Come descritto in forma testuale nella precedente sezione, le due funzionalità principali che fornirà il sistema riguardano il design e la ricerca di scenari attraverso l'applicativo. Inoltre il sistema dovrà interagire con altri sistemi di back end per poter fornire le proprie funzionalità all'utilizzatore finale e per poter così soddisfare i requisiti di integrazione dei dati ed interoperabilità dei sistemi.
- *Requisiti architetturali*: come si evince dal nome, l'applicazione rientrerà nella categoria degli EMIS. L'applicazione sarà fruibile attraverso la rete Internet dall'utente attraverso un client, per potersi interfacciare con il sistema. Il front end del sistema, accessibile con tutti i più comuni browser, offrirà un'interfaccia semplice ed usabile per poter eseguire tutte le funzionalità offerte dal sistema. Il back end del sistema oltre a gestire tutta la logica di business sarà progettato in modo da rendere il sistema il più possibile interagente con altri sistemi. L'intero sistema dovrà seguire così l'approccio orientato ai servizi, partendo dal modello di riferimento presentato dalla Service Oriented Architecture (SOA).

4.2.2 Specifica dei requisiti

Nella fase di raccolta dei requisiti è stata presentata una panoramica di alto livello dei principali requisiti del progetto. In questa parte invece si passerà da una presentazione testuale, quindi più libera, ad una presentazione più accurata e nella quale si farà uso di alcune tecniche di modellazione e specifica offerte dal linguaggio UML (Unified Modeling Language). In particolare verrà utilizzata la notazione dei *diagrammi dei casi d'uso*, di cui si descriveranno le diverse componenti in maniera dettagliata.

- **USE CASE DIAGRAM**

In Figura 4.3 è riportato lo Use Case Diagram.

Si definiscono così i due elementi che costituiscono il diagramma:

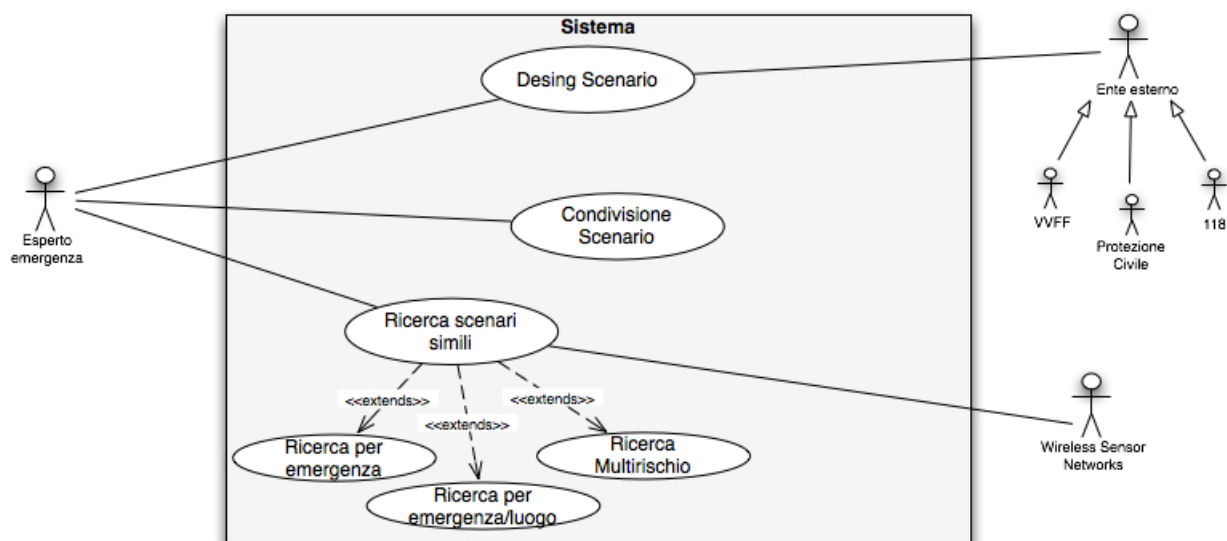


Figura 4.3: UML - Use Case Diagram

- *Attori*: all'interno dello Use Case Diagram definito per il sistema SocialEMIS devono essere identificate due tipologie di attori; la prima tipologia sono gli utenti che interagiscono in modo diretto con il sistema e che nel caso di SocialEMIS sono gli esperti di dominio o gli operatori delle sale di emergenza delle protezioni civili. La seconda categoria comprende invece tutti i sistemi informativi che forniscono un qualche servizio al sistema e che nella fattispecie sono i Vigili del Fuoco, sedi della Protezione Civile, gli ospedali e tutti quegli enti che rientrano in qualche modo nel ciclo di gestione dell'emergenza. Tra questi anche tutti quei sistemi come le Wireless Sensor Network che sono utili per poter integrare nel sistema dati dal territorio in tempo reale. Questi ultimi a differenza degli utilizzatori principali sono gli attori automatizzati.
- *Casi d'Uso*: i casi d'uso, seguendo l'ordine utilizzato nella panoramica delle funzionalità del sistema sono: il *Design e la Condivisione di Scenari di Rischio*, la *Ricerca di Scenari di Rischio* secondo diverse modalità e la *Generazione di Web Services*.

I casi d'uso, come già specificato, sono emersi e sono stati definiti in base all'analisi dei requisiti e alle funzionalità che il sistema deve esporre. Per esplicitare come i requisiti iniziali e i requisiti funzionali sono stati soddisfatti dal sistema si inserisce nel seguito la Tabella 4.1 esplicativa della relazione che esiste tra un particolare requisito, il caso d'uso

4 SocialEMIS

che lo soddisfa e l'attore che è direttamente collegato allo use case.

REQUISITI	USE CASE	ATTORI
SocialEMIS dovrà poter fornire all'utilizzatore tutte le funzionalità necessarie per poter gestire gli Scenari di Rischio.	Design Scenario di Rischio	Esperto di emergenze
L'utilizzatore dovrà poter condividere con gli altri esperti i propri scenari di rischio.	Condivisione Scenario di Rischio	Esperto di emergenze
SocialEMIS dovrà implementare una funzionalità di ricerca di scenari simili in base al parametro 'emergenza'.	Ricerca per emergenza	Esperto di emergenze
SocialEMIS dovrà implementare una funzionalità di ricerca di scenari simili in base ai parametri 'emergenza' e 'luogo dell'emergenza'.	Ricerca per emergenza/luogo	Esperto di emergenze
SocialEMIS dovrà implementare una funzionalità di ricerca di scenari simili in base al 'multirischio'.	Ricerca Multirischio	Esperto di emergenze

Tabella 4.1: Corrispondenza Requisiti, Use Case e Attori

4.2.3 Design Scenario di Rischio

Attraverso le funzionalità che compongono questo use case l'utente ha la possibilità di progettare un nuovo scenario di rischio. L'esperto di dominio potrebbe essere interessato nella fase di studio di un particolare caso di emergenza ambientale, alla creazione di una serie di scenari che rappresentano delle situazioni possibili relativamente al verificarsi di un certo evento calamitoso. Attraverso il sistema è possibile generare nuovi scenari utilizzando una mappa interattiva identica alla mappa di Google Maps, ma con in più la possibilità di arricchire la mappa con tutta una serie di markers customizzabili dall'utente e che identificano tutti gli elementi necessari per descrivere al meglio lo scenario di rischio e l'emergenza che si vuole prevenire. Nella fase di design l'utente inquadra sulla mappa l'area di interesse, con il giusto livello di zoom e carica sulla stessa mappa una serie di markers che identificano le strutture presenti sul territorio. Sempre in questa fase, si definisce un altro marker che invece identifica l'evento calamitoso: questo può essere adeguatamente configurato e come il resto dei markers georeferenziato sulla mappa. Ogni marker posizionato sulla mappa in fase di design può essere invocato per ricevere le informazioni relative ad esso e per aggiornare i dati forniti. La fase di design si conclude nel momento in cui l'utente decide di voler salvare lo scenario creato. L'attore riferito a questo use case, come già detto, è l'esperto di dominio; il design dello scenario di rischio è collegato con gli attori automatizzati 'Enti esterni' che rappresenta i sistemi informativi che generano Web Services invocabili dal sistema. Nella fattispecie questi attori sono i Vigili del Fuoco, sedi della Protezione Civile, gli ospedali e tutti quegli enti che rientrano in qualche modo nel ciclo di gestione dell'emergenza. La connessione tra questo use case e gli attori automatizzati è fondamentale per sottolineare la caratteristica principale di tutti i markers che posizionati sulla mappa costruiscono lo scenario di rischio, e cioè il fatto che i dati forniti dai markers sono forniti in tempo reale attraverso l'utilizzo dei Web Services. In questo modo il design dello scenario sfrutta i servizi esposti da altri attori e ha la possibilità di avere sempre, costantemente dati aggiornati e aggiornabili in tempo reale. Strettamente collegato allo use case di Design Scenario c'è anche lo use case di *Condivisione di Scenari di Rischio*: lo scenario di rischio creato dall'utente può essere salvato attraverso un semplice form che rende lo scenario condiviso. La proprietà dello scenario è ovviamente di chi l'ha progettato, ma l'architettura di SocialEMIS fa sì che tutti gli scenari salvati, siano condivisi tra i diversi esperti di dominio che possono visualizzare e consultare anche scenari che non sono stati creati direttamente da loro. La condivisione degli scenari avviene automaticamente attraverso il salvataggio dello scenario stesso. Il sistema si occupa di inserire lo scenario all'interno di un database condiviso. L'attore relativo a questo use case è ancora l'esperto di emergenze.

4.2.4 Ricerca Scenari di Rischio

Questo use case, insieme al 'Design di Scenari di Rischio', contiene alcune delle funzionalità più importanti dell'intero sistema. Proprio in questo use case sono concentrate le funzioni di ricerca di scenari simili tra tutti gli scenari condivisi e mantenuti in memoria da SocialEMIS. Questo use case si specializza in tre use case differenti che implementano tre tipologie di ricerca differenti. L'attore è per tutti i casi l'esperto di dominio.

- **Ricerca per emergenza:** la funzionalità è offerta in maniera analoga alla funzionalità di design, nel senso che anche la ricerca avviene attraverso un'interazione diretta con la mappa, che viene ancora una volta customizzata attraverso l'inserimento di marker che identificano una certa emergenza, che potrebbe interessare l'area circoscritta sulla mappa. Attraverso un modulo apposito dell'interfaccia di SocialEMIS è possibile caricare sulla mappa le strutture del territorio ed un marker che identifica l'emergenza che è posizionato nel centro di un poligono e che descrive l'area esatta colpita dall'emergenza. L'utente ha la possibilità di segnalare quali delle strutture del territorio potrebbero essere effettivamente colpite e impostare così una ricerca che trovi, tra tutti gli scenari condivisi, quelli che sono più simili a quello appena ricreato dall'utente. La ricerca viene effettuata dal sistema attraverso un algoritmo che calcola la somiglianza tra lo scenario in esame e quelli salvati basandosi sulla tipologia di emergenza, definita dall'utente, e sulle strutture che sono state colpite.
- **Ricerca per emergenza e luogo:** il secondo use case del caso di 'Ricerca di Scenari di Rischio' affina la ricerca rispetto al precedente use case perché inserisce un nuovo parametro di ricerca che è il luogo in cui si è manifestata l'emergenza. La distanza tra il luogo definito nello scenario di riferimento e i diversi luoghi definiti negli scenari salvati diventa discriminante per tutti quegli scenari che sono stati definiti in aree geografiche completamente diverse rispetto a quella definita dall'utente. La scelta dell'emergenza in questo caso viene definita dall'esperto di dominio in modo strutturato: l'esperto infatti dopo aver consultato ed interpretato, attraverso l'utilizzo dei Web Services, i dati provenienti dalla Wireless Sensor Network relativi alle ultime misurazioni sul territorio, può definire sulla mappa un marker relativo all'emergenza del caso di studio, calibrandone la tipologia e l'intensità proprio grazie ai dati delle ultime misurazioni dei sensori. L'ipotesi sull'emergenza futura viene definita all'interno dello scenario, proprio grazie al fatto che l'utilizzatore può sfruttare attraverso l'uso dei web services relativi alla wireless sensor network, dei dati in tempo reale, continuamente aggiornabili attraverso

l'applicazione SocialEMIS stessa e senza dover fare riferimento ad altri applicativi per il monitoraggio del territorio. Questo giustifica la connessione tra lo use case generale di *Ricerca di scenari simili* e l'attore automatizzato *Wireless Sensor Networks*.

- **Ricerca Multirischio:** lo use case per la ricerca di tipo multirischio, implementa le funzionalità che permettono allo user di poter definire all'interno dello scenario di riferimento una emergenza primaria, cioè l'emergenza che è al centro del caso di studio, ed in più una o più emergenze secondarie che potrebbero essere generate dal verificarsi della prima emergenza. Lo scenario definito dall'esperto di dominio sarà così costituito da più markers che indicano sulla mappa le diverse emergenze che hanno colpito il territorio. In questo caso una ricerca con i parametri visti nei due use case precedenti non sarebbe stata sufficiente; è per questo che è stata pensata una funzionalità appositamente per questo tipo di caso, in cui l'algoritmo di ricerca andrà a recuperare quegli scenari che hanno al loro interno le definizioni di una o più emergenze tra quelle definite nello scenario di riferimento.

4.3 Analisi dettagliata

Attraverso la pianificazione e l'approccio allo sviluppo sono stati definiti i requisiti principali del sistema, gli obiettivi da raggiungere e la descrizione delle funzionalità che SocialEMIS mette a disposizione. È stato definito il sistema da un punto di vista esterno, prendendo come riferimento il punto di vista dell'utilizzatore finale. In questa sezione invece il sistema verrà descritto internamente, cercando di dare una panoramica della struttura del sistema. Inizialmente verranno descritti i passi che hanno portato dalle funzionalità definite nei requisiti, alla progettazione e suddivisione dei moduli software che implementano tali funzionalità. Successivamente si presenterà la fase di progettazione dei dati e delle entità che hanno portato alla creazione della base di dati.

4.3.1 Definizione dei Moduli

Un modulo rappresenta un'astrazione per poter tradurre uno o più use case in un package, visto come un insieme di tutte quelle classi che in fase di implementazione andranno a definire le funzionalità descritte dai requisiti. Il modulo va visto come il primo punto di contatto tra la fase di analisi e la fase di progettazione, poiché qui si inizia a suddividere il sistema in sottosistemi che si occupano di una determinata funzionalità. In Figura 4.4 è stato definito il diagramma dei moduli di SocialEMIS.

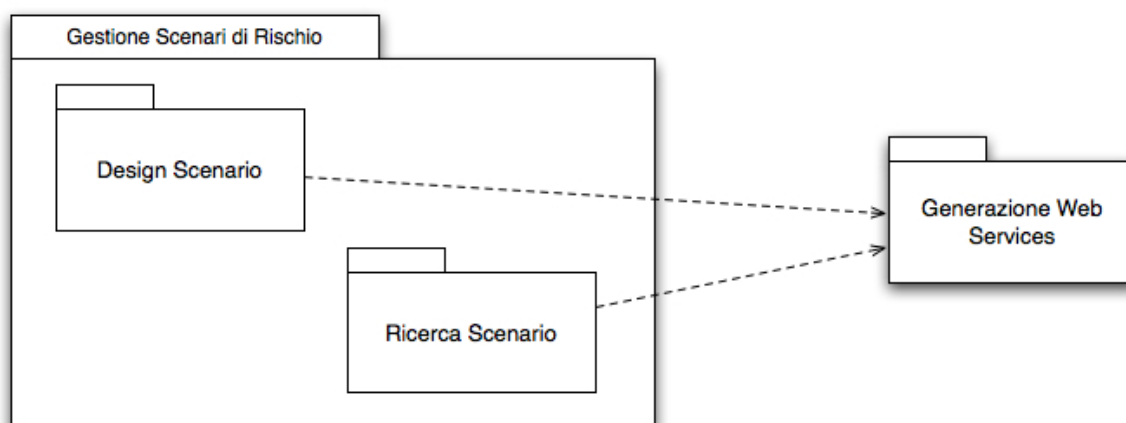


Figura 4.4: Moduli software di SocialEMIS

Il modulo principale è quello di Gestione degli scenari di rischio, di cui fanno parte i moduli di Design e di Ricerca di scenari. All'interno di questo modulo software saranno presenti tutti i package e le classi che implementeranno i metodi necessari per le funzionalità descritte negli use case. Sono stati definiti tre moduli principali che saranno la base di partenza sia per la fase di progettazione del sistema che per la successiva fase di implementazione. Il modulo di Gestione degli scenari contiene tutte le macrofunzionalità fornite da SocialEMIS identificate dai due moduli 'Design' e 'Ricerca' di scenari. Entrambe i moduli sono collegati con il modulo che si occupa della generazione dei Web Services; come definito nei requisiti e nella panoramica delle funzionalità, il modulo software di design e il modulo di ricerca utilizzeranno le classi definite nel modulo esterno per poter richiedere i dati provenienti dai web services.

4.3.2 Progettazione dei dati

Analizzando nel dettaglio la panoramica delle funzionalità del sistema sono emersi i dati principali che SocialEMIS deve gestire. Questi dati sono stati tradotti in entità che andranno poi a costituire le tabelle contenute nel database del sistema. In particolare la fase di progettazione della base di dati è stata suddivisa in due parti: la prima in cui è stato definito il diagramma Entità Relazione, utile in fase di progettazione per poter aver chiari gli elementi che popoleranno il database e come questi dati sono correlati e possono eventualmente venire aggregati. Nella seconda parte invece si presenterà un'analisi degli stessi elementi definiti nel diagramma ER, ma con uno sguardo più rivolto all'architettura del sistema, suddividendo quelli che sono i dati prodotti da SocialEMIS

e salvati nel database del sistema e quali sono invece quei dati che vengono integrati nel sistema attraverso l'utilizzo dei Web Services.

- **DIAGRAMMA ER:**

in Figura 4.5 è riportato il diagramma ER come era stato pensato proprio in fase di analisi del progetto SocialEMIS. L'intero diagramma è centrato sull'entità *Scenario* e tutte le altre entità come la *Mappa*, i *Markers* e le *Emergenze* sono fondamentalmente tutti gli elementi costitutivi dello scenario stesso. Ogni *Scenario* è quindi identificato dalla *Mappa* che indica il luogo esatto esaminato dall'esperto di dominio e dall'emergenza o dalle emergenze che si potrebbe verificare sul territorio. All'entità *Mappa* sono poi collegate tutte quelle entità che fanno riferimento ad elementi che devono essere riportati sulla mappa stessa perchè sono elementi importanti e da tenere in considerazione sul territorio. Tra questi nel diagramma ER sono stati identificati i *Sensori Fisici* che rappresentano i sensori delle Wireless Sensors Network, le *Strutture del territorio* e i cosiddetti *Servizi Partner* che sono tutti gli enti coinvolti nella gestione dell'emergenza come i Vigili del Fuoco, gli ospedali, le strutture di soccorso, le Protezioni Civili ecc. Infine il diagramma ER riporta ancora le entità che identificano l'*User* e cioè l'esperto di dominio che utilizza SocialEMIS per gestire gli scenari di rischio e il *Caso di studio* a cui ogni scenario fa riferimento.

- **SPECIFICA DELL'ARCHITETTURA DATI:**

è bene dare delle informazioni aggiuntive rispetto ai dati che vengono gestiti ed utilizzati dal sistema, avendo in mente le considerazioni riportate nell'analisi dei requisiti architetturali e nella descrizione dello use case relativo alla generazione dei Web Services.

Come è stato detto, l'applicazione SocialEMIS presenta due caratteristiche fondamentali che hanno dato le linee guida per una progettazione di un sistema orientato ai servizi, interoperabile e che integrasse diversi flussi di dati provenienti da altri sistemi. In Figura 4.6 è stato rappresentato un modello dei dati in cui vi è una netta separazione tra i dati che sono generati e salvati direttamente dal sistema e che in figura sono rappresentati dalle entità *User*, *Caso*, *Emergenza* e *Struttura del Territorio*, rispetto a quei dati che invece vengono richiesti attraverso l'utilizzo di Web Services.

Questi dati sono rappresentati in Figura 4.6 dai *Partner Web Service Data* e dai *Wireless Sensors Network Web Service Data*; entrambe rappresentano dei flussi di

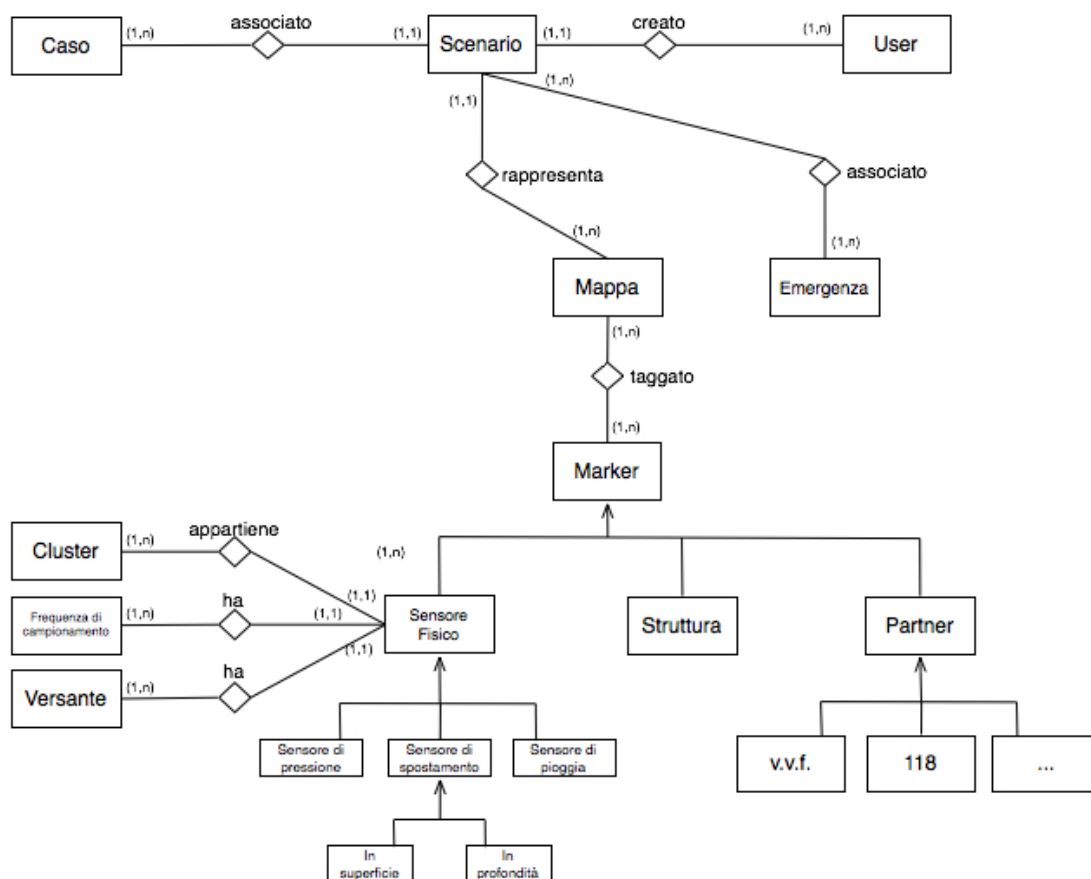


Figura 4.5: Diagramma ER

dati che provengono da sistemi informativi indipendenti da SocialEMIS e corrispondono rispettivamente alle entità del diagramma ER di figura 4.3 *Sensori Fisici* e *Partner*. Sono dati che i cosiddetti sistemi partner mettono a disposizione e che possono essere utilizzati attraverso i Web Services.

- **SCENARI DI RISCHIO IN FORMATO KML:**

L'idea alla base del meccanismo di condivisione del progetto è quella di avere un database condiviso tra gli utilizzatori dell'applicazione, che contenga tutti gli scenari di rischio creati tramite SocialEMIS. Uno scenario di rischio, come si cerca di rappresentare in Figura 4.6, è un insieme di dati generati dal sistema, integrati con altri flussi di dati provenienti dai Web Services. Lo scenario di rischio e tutti i dati di cui è composto, sono salvati all'interno di un file KML (Keyhole Markup Language). Il database su cui si interfaccia SocialEMIS può essere visto come un

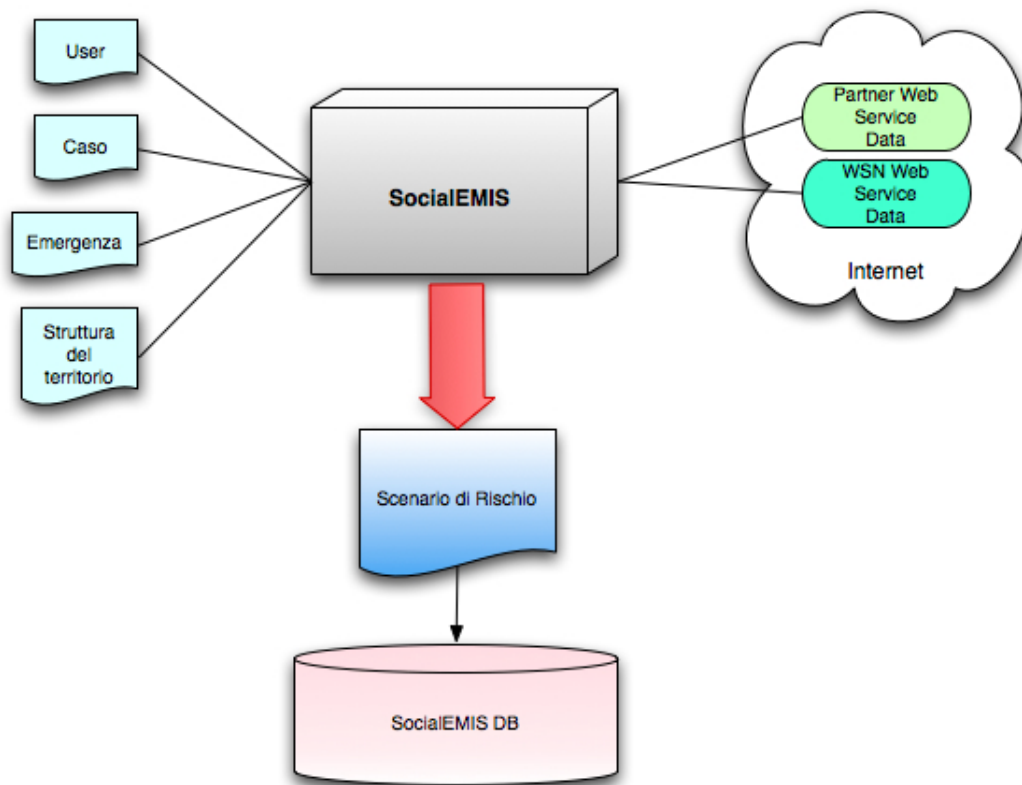


Figura 4.6: Schema dei flussi di dati

raccogliatore di tutti i file KML che vengono generati di volta in volta dagli utenti. Il file KML si adatta perfettamente a contenere uno scenario di rischio: attraverso la sua semplice struttura a tag permette di immagazzinare tutti gli elementi definiti in fase di design dall'esperto di dominio.

4.4 Progettazione

In questa sezione si continua con la fase di progettazione delineando l'architettura definitiva del sistema. La presentazione dell'architettura procederà per livelli analizzando passo per passo i diversi layer di cui è composta.

4.4.1 Architettura SocialEMIS

Nella definizione dell'architettura è importante presentare tutti i moduli principali che compongono il sistema specificando se si tratta di moduli che implementano funzionalità di Front-end o di Back-end.

Mantenere una netta separazione tra front-end e back-end a livello di progettazione e, come vedremo nella prossima sezione, anche a livello di implementazione è importante per avere un giusto approccio ingegneristico allo sviluppo del progetto. Risulta anche un buon approccio dal punto di vista descrittivo perché permette di tenere ben separati i diversi aspetti dell'architettura di sistema. In Figura 4.7 è schematizzata l'architettura di SocialEMIS.

- **SOTTOSISTEMA DI MEMORIZZAZIONE:**

Contiene la base di dati in cui sono ospitati gli Scenari di Emergenza creati e salvati dagli utilizzatori dell'applicazione. Il sottosistema oltre a fornire un meccanismo di archiviazione dello storico degli scenari, permette di avere una struttura di informazioni condivisa tra tutti gli utilizzatori di SocialEMIS. All'interno del sottosistema di memorizzazione è implementata la logica di accesso ai dati.

- **BACK END:**

Il back end dell'applicazione identifica tutta la logica di business del sistema e tutte quelle funzionalità che vengono gestite lato server. Da un punto di vista architetturale la caratteristica più importante del back-end è il *Sottosistema di Interoperabilità*: questo sottosistema è utile per fornire all'utilizzatore una serie di informazioni e diversi flussi dati, che provengono da altri sistemi che decidono di rendere disponibili alcuni dei propri dati, soddisfacendo così il requisito di integrazione dei dati definito nelle precedenti sezioni. Nell'architettura di Figura 4.7 al sottosistema di interoperabilità sono connessi una serie di enti che, coinvolti direttamente o indirettamente con la gestione dell'emergenza, rendono pubblici una serie di dati che possono essere consultati ed utilizzati dall'esperto di dominio durante l'interazione con il sistema SocialEMIS. Il sottosistema di interoperabilità utilizza

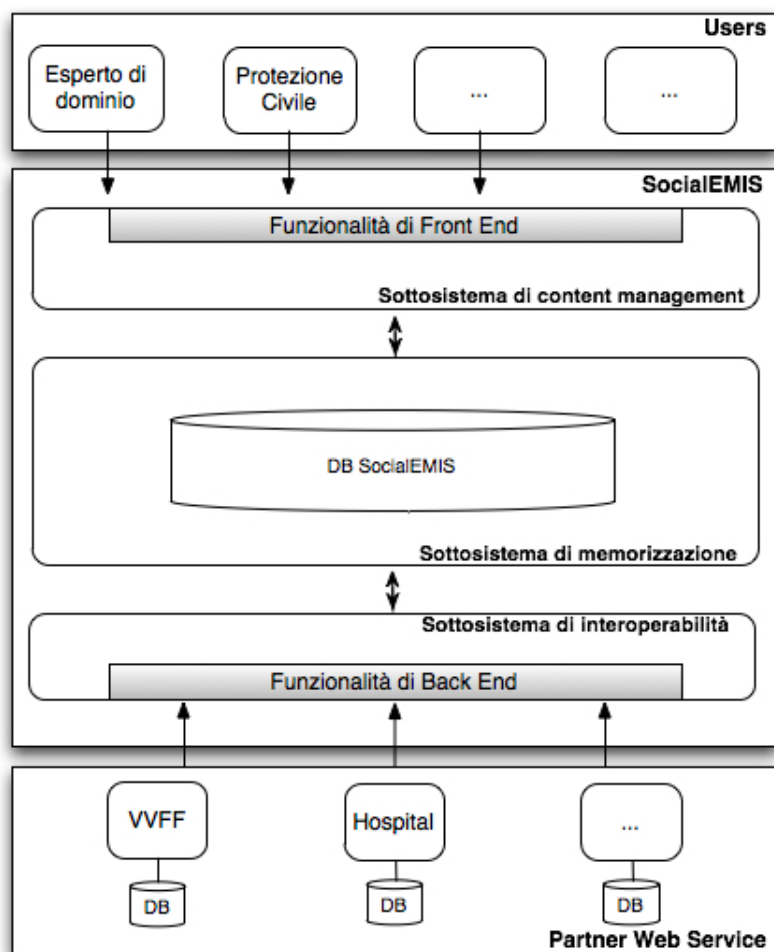


Figura 4.7: SocialEMIS - Architettura di sistema

per implementare questo tipo di comunicazione con sistemi esterni, la tecnologia dei Web Services, il che rende il sistema SocialEMIS una Service Oriented Architecture interoperabile con sistemi eterogenei, importante requisito architetturale definito in fase di definizione dei requisiti. Da una parte la logica di back-end è impiegata per l'interfacciamento del sistema con l'esterno, dall'altra invece si occupa della connessione con il sottosistema interno di memorizzazione.

- **FRONT END:**

Con il Font End si identifica il client dell'applicazione; esso fornisce l'interfaccia grafica di SocialEMIS e consente agli utilizzatori di interagire con il sistema. Fa

4 SocialEMIS

parte del Front End il *Sottosistema di Content Management*: esso identifica quella parte del sistema che permette all'utente di avere tutti gli strumenti necessari per poter gestire i contenuti, che nel caso di SocialEMIS sono gli Scenari di Rischio. All'interno di questo sottosistema è implementata la logica di presentazione di SocialEMIS. L'interfaccia grafica, risultato dell'impementazione del sottosistema di content management, fornisce all'esperto di dominio che interagisce con essa, tutti gli strumenti per poter creare e gestire i propri scenari di rischio e per poter effettuare ricerche di scenari. Il front-end, facendo riferimento ai moduli software definiti in Figura 4.4 si occupa di fornire un'interfaccia comune sia per il modulo di Design Scenario che il modulo di Ricerca di Scenari. Uno screenshot dell'interfaccia grafica di SocialEMIS è visibile in Figura 4.8.

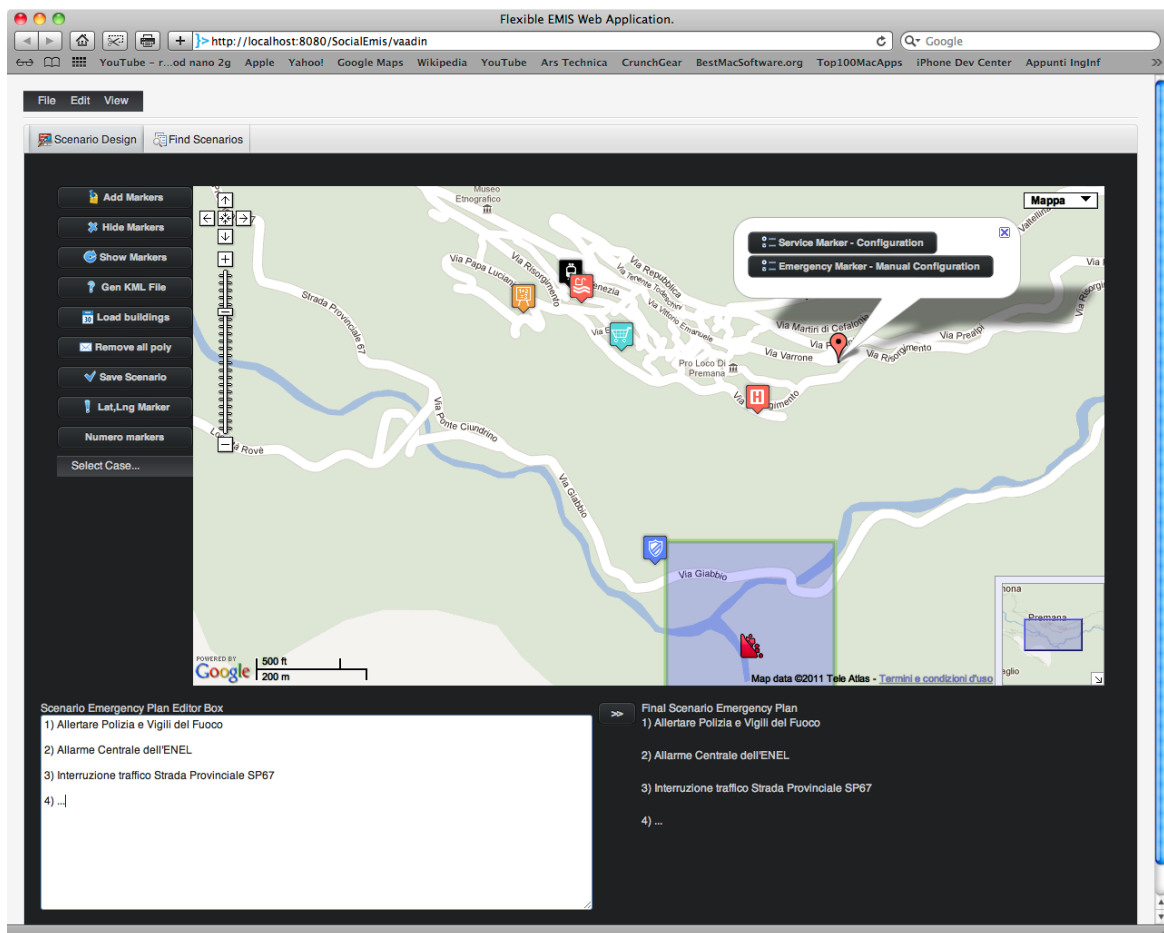


Figura 4.8: Interfaccia utente

4.5 Algoritmi di Similarity

Si conclude il capitolo centrale del documento con una sezione interamente dedicata ad un approfondimento degli algoritmi, che sono stati implementati per poter inserire all'interno del progetto SocialEMIS la funzionalità definita come *Ricerca di Scenari Simili* nella panoramica delle funzionalità. All'interno di questa sezione si cercherà di presentare i tre algoritmi implementati per poter fornire all'utilizzatore le tre tipologie di ricerca, senza entrare nei dettagli del codice, ma cercando di definire in maniera schematica la struttura dei diversi algoritmi: che tipo di ricerca implementano e che differenze hanno l'uno rispetto all'altro. Il focus su questa funzionalità è importante sia per presentare dei prototipi di algoritmi che possono effettivamente essere utili agli esperti di dominio durante le fasi di studio e di ricerca, ma anche per presentare quella che forse è la funzionalità più originale dell'intero progetto. Le tre sezioni seguenti riportano ognuna un differente algoritmo di ricerca.

4.5.1 Algoritmo di ricerca per Emergenza

Il primo algoritmo di ricerca di scenari simili permette all'esperto di dominio di impostare una ricerca tra tutti gli scenari mantenuti nel database di SocialEMIS definendo come unico parametro di ricerca la tipologia di emergenza. Lo scenario tipico, dal punto di vista dell'interazione con il sistema, è ad esempio quello di un esperto di emergenze che, dopo aver monitorato i valori della Wireless Sensor Network, definisce sulla mappa un marker che identifica un' emergenza di una certa natura e di una certa entità. L'esperto può successivamente segnalare sulla mappa le eventuali strutture che potrebbero avere un danno dal manifestarsi dell'emergenza, ad esempio l'allagamento di una scuola o l'incendio di un certo edificio. L'esperto a questo punto potrebbe essere interessato a visualizzare e studiare degli scenari che sono già stati creati da altri esperti e che riportano lo stesso tipo di emergenza che ha definito lui stesso nella mappa. Utilizzando la funzione di Ricerca per Emergenza il sistema invoca l'algoritmo di ricerca associato, la cui struttura è riportata in Figura 4.9. L'algoritmo dopo aver prelevato dal database tutti gli scenari salvati, effettua una prima selezione e in una prima passata mantiene solo ed esclusivamente quegli scenari di rischio che riportano l'emergenza definita dall'utente. Il set di scenari possibili è quindi ridotto ed è stato proprio il parametro impostato dall'esperto a effettuare una prima scrematura dall'insieme degli scenari. Ora l'algoritmo per affinare la ricerca andrà a ridurre ulteriormente il set di scenari mantenendo solo quelli che presentano le stesse strutture colpite dall'emergenza, come definite da lui stesso

nella mappa. Dopo questa seconda selezione, l'algoritmo presenta attraverso l'interfaccia grafica gli scenari risultato della ricerca.

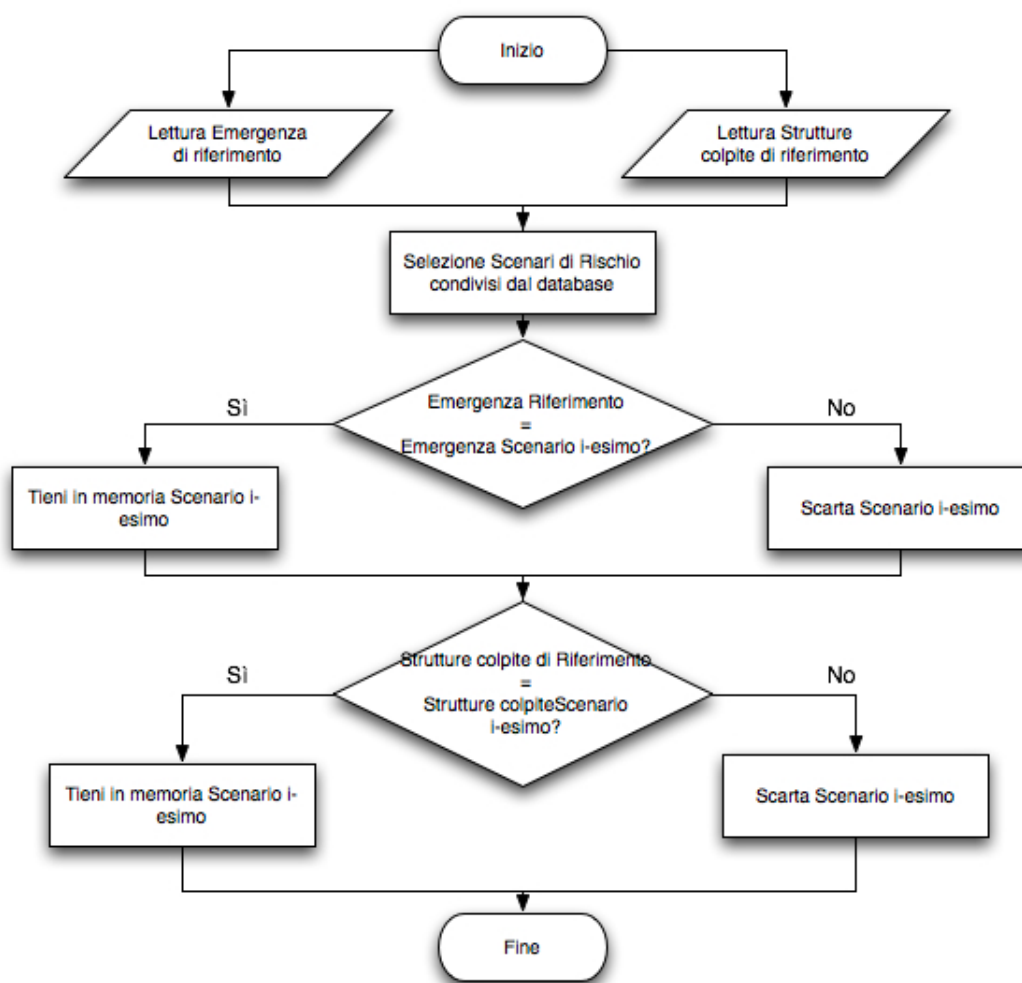


Figura 4.9: Diagramma di flusso per l'algoritmo di ricerca

4.5.2 Algoritmo di ricerca per Emergenza / Luogo

Il secondo algoritmo di ricerca presenta una struttura molto simile al primo, ma inserisce un nuovo parametro di ricerca rispetto all'emergenza. Si tratta del luogo dell'emergenza, cioè della posizione geografica in cui si è verificato l'evento calamitoso. Il nuovo algoritmo risulta così molto più preciso rispetto al primo, riportando una selezione più accurata tra tutti gli scenari salvati. L'introduzione di un nuovo parametro fa variare di poco l'algo-

ritmo di ricerca precedente, perché di fatto introduce solo una 'passata' in più all'interno del codice per poter selezionare gli scenari che hanno l'epicentro dell'emergenza il più possibile vicino a quello definito dall'utente. La ricerca in base all'emergenza definita e la selezione in base alla corrispondenza delle strutture colpite rimane la stessa dell'algoritmo precedente. Si presenta ora il procedimento per la verifica della corrispondenza geografica di due scenari.

- CALCOLO DELLA DISTANZA TRA GLI EPICENTRI DELLE EMERGENZE DI DUE SCENARI:

ogni emergenza definita in uno scenario è georeferenziata tramite una coppia (Latitudine, Longitudine). L'algoritmo deve così andare a calcolare la distanza tra il punto dell'emergenza definita dall'utente e il punto dell'emergenza definito nello scenario salvato nel database. Il calcolo deve essere effettuato per tutti gli scenari del set di scenari, che potrebbero essere simili a quello dato. Dati un'emergenza di riferimento definita dal punto $E1(lat1, lng1)$ e l'emergenza contenuta nello scenario salvato $E2(lat2, lng2)$, il calcolo della distanza tra le due emergenze viene effettuato dall'algoritmo attraverso le due seguenti funzioni matematiche:

$$f = \arccos [\cos (lat1 - lat2) * \cos(lng1) * \cos(lng2) + \sin(lng1) * \sin(lng2)] \quad (4.1)$$

$$distanza(E1, E2) = f * 6360km \quad (4.2)$$

dove i $6360km$ rappresentano la misura approssimativa del raggio terrestre. All'esperto di dominio verranno così presentati come scenari simili a quello definito, solo quelli che presentano la distanza minore tra tutte quelle calcolate.

4.5.3 Algoritmo di ricerca Multirischio

Il terzo ed ultimo algoritmo è quello che permette all'utilizzatore di effettuare una ricerca cosiddetta 'multirischio', nel senso che a differenza delle due tipologie di ricerche viste nelle precedenti sezioni, questo algoritmo non tiene traccia di un'unica emergenza come parametro di ricerca, ma permette all'utente di definire più emergenze che si sono verificate sul territorio. Si ipotizzi infatti il caso in cui l'esperto di dominio si trova nella situazione di dover studiare un particolare scenario in cui il verificarsi di un'emergenza può generare una seconda emergenza o addirittura una serie di emergenze a effetto

4 SocialEMIS

domino. Non sono rari i casi in cui in passato si sono verificate situazioni del genere. È importante per questo motivo implementare un algoritmo di questo tipo che riesca a ritrovare all'interno del database dei particolari scenari che riportino a situazioni in cui, sullo stesso caso di studio, siano state definite emergenze multiple. Dal punto di vista dell'algoritmo la ricerca viene effettuata tenendo dei parametri mutlipli che identificano le diverse emergenze ambientali definite dall'utente.

Tutti gli algoritmi di ricerca danno come risultato finale un set di scenari simili che l'applicazione SocialEMIS riporta attraverso l'interfaccia grafica all'esperto di dominio, il quale potrà ricaricarli per poterli visualizzare e studiare. La Figura 4.10 rappresenta un esempio di interazione con il sistema, che riporta i risultati di una ricerca di scenari simili. In rosso sono stati evidenziati gli scenari risultanti dall'esecuzione dell'algoritmo di ricerca.

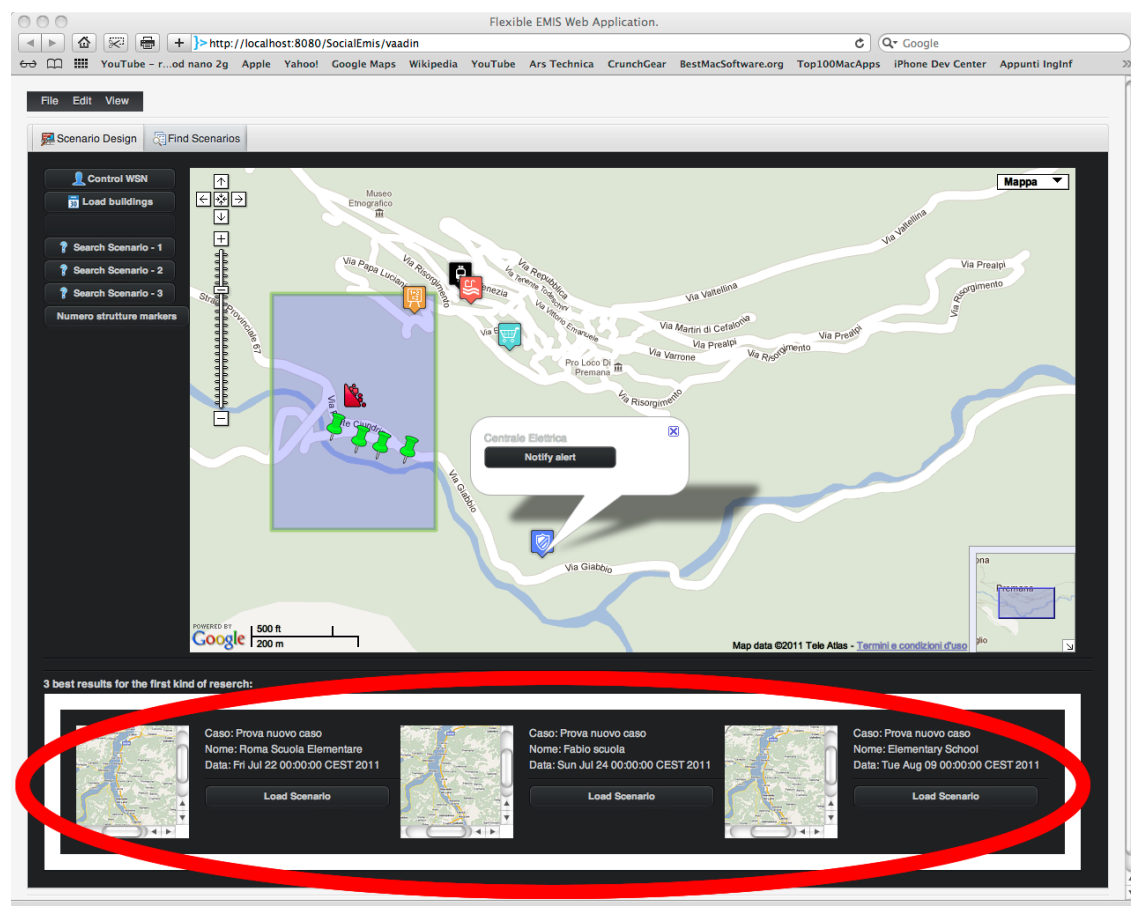


Figura 4.10: Screenshot dei risultati di una ricerca di scenari simili

5 Implementazione

Questo capitolo conclude la descrizione del progetto SocialEMIS ed è dedicato alla presentazione dei passi principali della fase di implementazione del sistema. Si definirà la struttura dei progetti implementati per costruire il prototipo SocialEMIS e si descriverà l'utilizzo degli strumenti tecnologici, elencati nel capitolo 3, nelle diverse fasi di sviluppo.

La descrizione dell'implementazione manterrà la solita distinzione tra la parte di Front-End e di Back-End del sistema, soluzione già adottata anche nell'analisi della progettazione del sistema per riportare l'approccio ingegneristico utilizzato e per rendere l'approccio strutturato e comprensibile.

Lo sviluppo di SocialEMIS è stato svolto attraverso l'implementazione di diversi progetti, tutti sviluppati attraverso il linguaggio Java utilizzando come ambiente di sviluppo l'IDE NetBeans¹. In Figura 5.1 è definita l'architettura che segue l'intero progetto SocialEMIS, e all'interno della struttura sono definiti gli elementi tecnologici impiegati in ogni fase dello sviluppo.

¹<http://netbeans.org/> , IDE attualmente alla versione 7.0.1

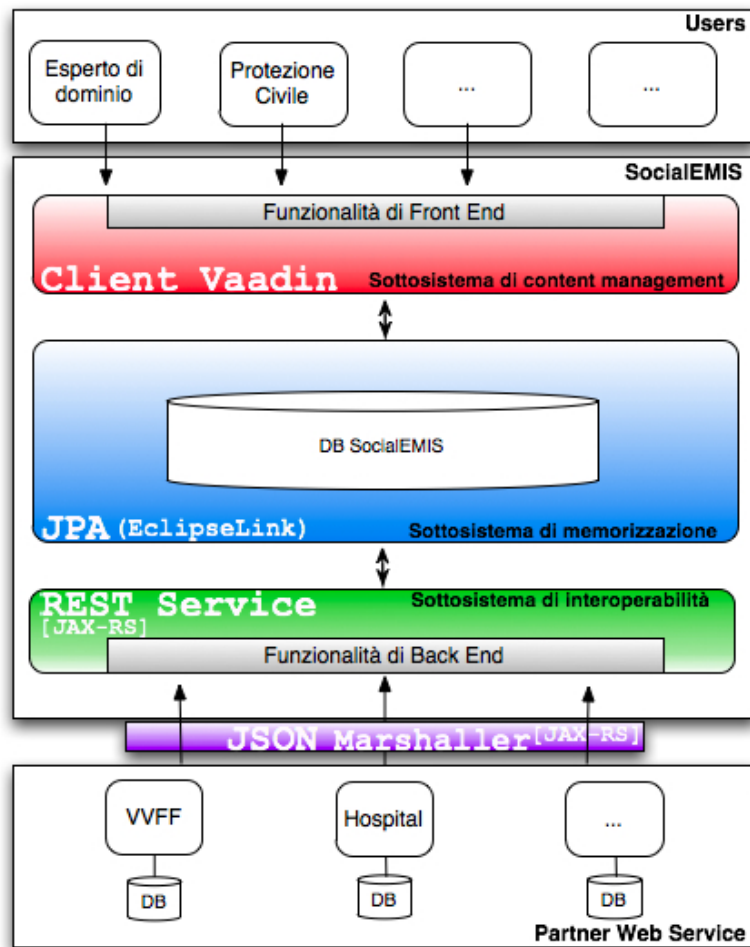


Figura 5.1: Architettura del progetto SocialEMIS

Analizzando la struttura del progetto, gli elementi principali che sono stati sviluppati sono:

- il Client dell'applicazione, che rappresenta l'interfaccia grafica e quindi il Front-End dell'applicazione, che sarà utilizzato dall'user per poter sfruttare le funzionalità del sistema SocialEMIS. Lo sviluppo degli elementi grafici e dell'interazione tra essi e la logica di business del sistema è stata implementata attraverso il Framework Vaadin.
- il modulo per la gestione di Web Services, secondo lo stile architetturale REST che

5 Implementazione

fa uso della JAX-RS² e cioè le Java API for RESTful Web Services. Tale modulo comprende le applicazioni che contengono il codice necessario per la creazione dei RESTful Service e gli elementi che si occupano del marshalling/unmarshalling delle rappresentazioni JSON delle entità richieste tramite i web services.

- il modulo EclipseLink JPA, che implementa una soluzione per la persistenza Java dei dati contenuti nel database dell'applicazione. L'implementazione di questo modulo consente l'interfacciamento con l'ultimo modulo dell'architettura che è il database.
- il Database relazionale, punto finale della parte relativa al Back-End dell'applicazione.

Per poter creare il prototipo SocialEMIS è stato necessario implementare attraverso l'IDE NetBeans una serie di progetti che hanno delle particolari funzioni e che si occupano di definire i diversi componenti che sono stati elencati e che costituiscono l'architettura riportata in figura 5.1.

Sono state definite due tipologie di progetti: da una parte tre Java Project che sono utilizzati nel sistema globale per poter implementare le Java Persistence API e connetterle con il database di sistema; la definizione e i principi legati all'implementazione di questi tre progetti verrà trattata nella sezione di Back-End del sistema.

I progetti restanti sono invece tre Java Web Application, uno dei quali è dedicato alla creazione del client SocialEMIS, che verrà quindi descritto nella sezione di Front-End; i due progetti restanti implementano invece i RESTful Web Services. L'elenco di tutti i progetti che sono stati creati in fase di implementazione è evidenziato in Figura 5.2.

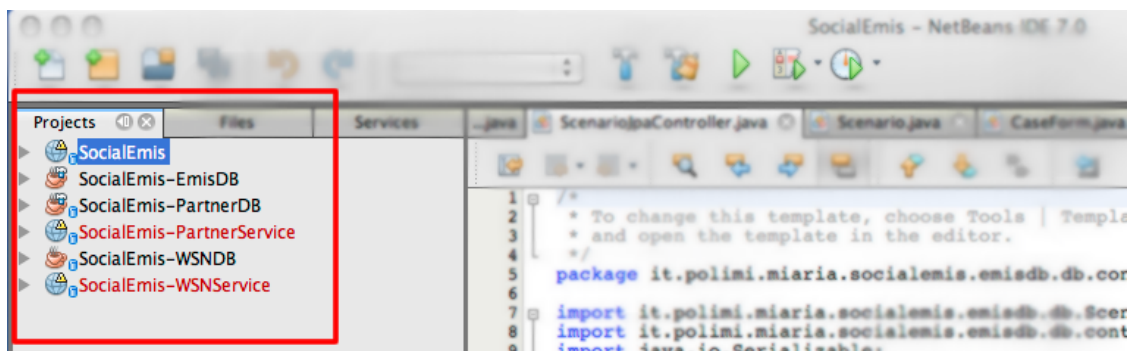


Figura 5.2: Elenco dei progetti creati

²<http://jcp.org/en/jsr/detail?id=311>, la specifica JSR 311 per i RESTful Web Services

5.1 Sviluppo Front-End di sistema

In una prima fase di sviluppo è stato necessario definire come presentare a livello di interfaccia grafica le diverse funzionalità del sistema SocialEMIS; il sistema per poter essere utilizzato da utenti che non sono necessariamente esperti di tecnologia o di informatica aveva il requisito di riportare una presentazione grafica semplice, funzionale ed intuitiva. E' stato deciso di inserire nell'interfaccia gli elementi essenziali per la gestione degli scenari di emergenza con uno stile pulito e semplice.

In una prima fase di design per avere un'idea di base sull'interfaccia da rendere all'utente è stata creata una bozza di interfaccia utilizzando un programma di grafica che permette di disegnare in pochi passi, secondo una politica di drag&drop di elementi grafici, uno schizzo del front end del sistema. Attraverso il tool Balsamiq³ è stata definita in poco tempo quella che sarebbe poi diventata l'interfaccia finale del sistema, la cui bozza preliminare è riportata in Figura 5.3.

Per l'implementazione vera e propria è stato invece utilizzato il framework Vaadin di cui è stato discusso nel capitolo 3. Come già detto, il framework e gli elementi che esso mette a disposizione, hanno permesso di ricreare all'interno della Java Web Application *SocialEmis* un'interfaccia del tutto simile a quella definita tramite Balsamiq.

In particolare per poter lavorare con il framework Vaadin è stato necessario importare all'interno dell'IDE NetBeans la libreria di Vaadin e richiamare la libreria opportunamente all'interno della classe principale del progetto come riportato nel seguente blocco di codice.

³<http://balsamiq.com/>

5 Implementazione

```
import com.vaadin.Application;
import com.vaadin.ui.*;
public class HelloVaadin extends Application {
    @Override
    public void init() {
        Window mainWindow = new Window("HelloVaadin");
        Label label = new Label("Hello_Vaadin_user");
        mainWindow.addComponent(label);
        setMainWindow(mainWindow);
    }
}
```

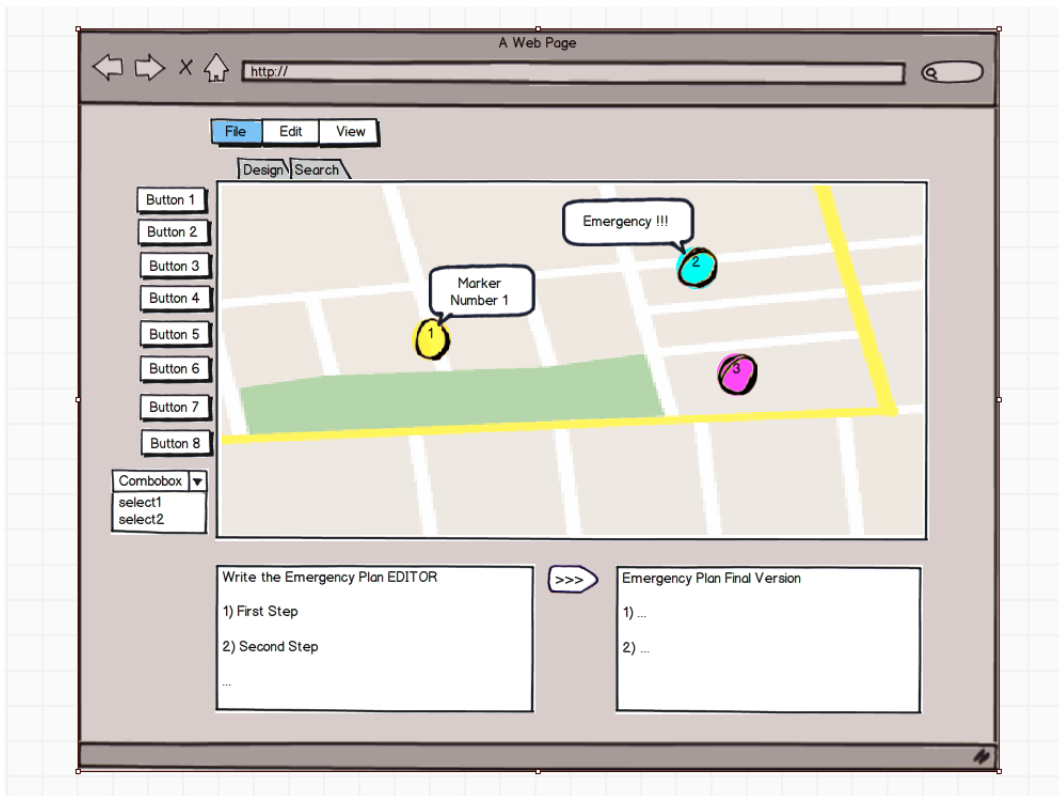


Figura 5.3: Layout applicazione

5.2 Sviluppo Back-End di sistema

I progetti che sono stati sviluppati come Back-End del sistema sono stati definiti in parte per simulare la generazione di Web Services, in parte per gestire l'interazione con il database di SocialEMIS. Come è stato detto nella definizione degli obiettivi del progetto, una caratteristica fondamentale dell'applicazione è quella di essere interoperabile con altri sistemi attraverso l'utilizzo di RESTful Web Services. Nello sviluppo del progetto è stato quindi necessario poter creare un prototipo di Web Services che fossero effettivamente utilizzabili. Per fare ciò sono stati creati i seguenti progetti:

SocialEMIS-WSNDB: Java application che si occupa della gestione dei dati provenienti da un database che contiene le informazioni raccolte da una Wireless Sensor Network installata in una determinata zona per monitorare e memorizzare una serie di misurazioni. Il progetto Java definisce tutte le entità necessarie alla gestione dei nodi e dei sensori della WSN partendo da questo database.

SocialEMIS-PartnerDB: Java application per la gestione dei dati provenienti dai 'partner' del progetto SocialEMIS (ospedali, vigili del fuoco, protezioni civili ecc.) cioè tutti quegli enti che fanno parte del ciclo di vita dell'emergenza e che possono fornire informazioni essenziali per il risk management e per la definizione di scenari di rischio. Il progetto definisce le entità necessarie per identificare i diversi enti e le informazioni che vengono rese pubbliche e quindi fruibili da altri sistemi.

I due progetti appena definiti, interfacciandosi direttamente con i rispettivi database dei dati relativi alla Wireless Sensors Network e degli enti partner, permettono di creare le classi *Entity* e le classi *Controller* per tutte le entità contenute nel database:

- le classi *Entity* creano una corrispondenza uno ad uno tra ogni tabella contenuta nel database e una nuova classe Java che astrae dall'entità contenuta nel database e ne fornisce tutti gli attributi ed i metodi setter e getter associati ed infine una serie di 'named queries' autogenerate, che definiscono una serie di interrogazioni per il database senza dover scrivere a mano alcuna riga di codice SQL. Un esempio di classe entity è rappresentato in Figura 5.4, che riporta solo un esempio di parte del codice.

```

20 L  */
21  @Entity
22  @Table(name = "hospital")
23  @XmlRootElement
24  @NamedQueries({
25      @NamedQuery(name = "Hospital.findAll", query = "SELECT h FROM Hospital h"),
26      @NamedQuery(name = "Hospital.findByIdHospital", query = "SELECT h FROM Hospi
27      @NamedQuery(name = "Hospital.findByAmbulance", query = "SELECT h FROM Hospit
28      @NamedQuery(name = "Hospital.findByBed", query = "SELECT h FROM Hospital h W
29      @NamedQuery(name = "Hospital.findByVolounteer", query = "SELECT h FROM Hospi
30      @NamedQuery(name = "Hospital.findByOther", query = "SELECT h FROM Hospital h
31  public class Hospital implements Serializable {
32      private static final long serialVersionUID = 1L;
33      @Id
34      @Basic(optional = false)
35      @Column(name = "id_hospital")
36      private Integer idHospital;
37      @Column(name = "ambulance")
38      private String ambulance;
39      @Column(name = "bed")
40      private String bed;
41      @Column(name = "volounteer")
42      private String volounteer;
43      @Column(name = "other")
44      private Integer other;
45
46      public Hospital() {
47      }
48
49      public Hospital(Integer idHospital) {
50          this.idHospital = idHospital;
51      }
52
53      public Integer getIdHospital() {
54          return idHospital;
55      }

```

Figura 5.4: Code snippet della classe Hospital Entity

- le classi *JPAController*, autogenerate dal JPA Persistence Provider che nella fattispecie è EclipseLink JPA 2.0, definiscono tutti i metodi Java per selezionare, aggiornare, inserire ed eliminare dati dal database. Un esempio dei metodi generati è riportato in Figura 5.5. Ad ogni Entity class è associata una classe JPAController.

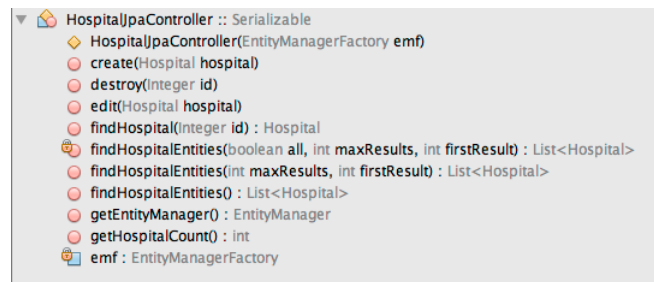


Figura 5.5: Elenco dei metodi forniti dalla classe Controller

5 Implementazione

A partire dai due progetti appena descritti sono stati creati altri due progetti che hanno il compito di generare i Web Service. I due progetti sono i seguenti:

SocialEMIS-PartnerService: Java Web Application che genera i RESTful Web Services partendo dalle classi Entity relative al progetto SocialEmis-PartnerDB. Ad ogni entità è assegnato uno specifico URI che può essere raggiunto attraverso i metodi definiti dal protocollo HTTP.

SocialEMIS-WSNService: Java Web Application che genera i RESTful Web Services per le classi Entity relative al progetto SocialEmis-WSNDB. I nodi e i sensori presenti nel database sono associati ad uno specifico URI che può essere raggiunto attraverso i metodi definiti dal protocollo HTTP.

Ogni URI generato tramite i Web Services rappresenta una risorsa pubblica che può essere richiesta da altri sistemi attraverso l'utilizzo di semplici metodi del protocollo HTTP. Tipicamente i sistemi richiedenti rappresentano i client nello schema tipico dei RESTful Web Services di interazione client e server. Ciò che viene trasferito nella comunicazione è una rappresentazione dello stato interno della risorsa a cui è associato un certo URI.

Lo schema finale dei diversi progetti creati in fase di implementazione e le relazioni esistenti tra essi sono definite in Figura 5.6

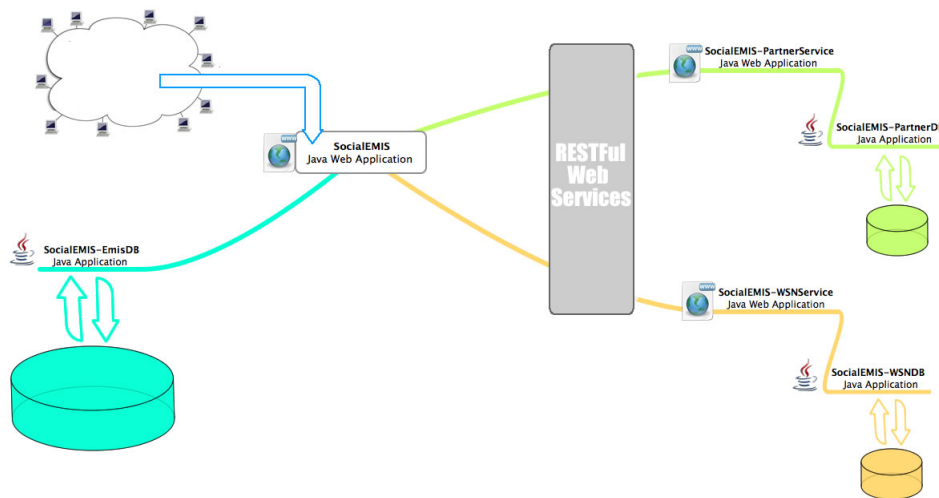


Figura 5.6: Schema di definizione dei progetti implementati

6 Conclusioni

Con questo documento di tesi è stato presentato il lavoro che ha portato allo sviluppo della web application SocialEMIS, un Emergency Management Information System che fornisce gli strumenti necessari per poter gestire degli Scenari di Rischio. Il progetto nasce con l'intento di fornire agli esperti di emergenze uno strumento nuovo ed innovativo che può superare alcune problematiche che riportano gli strumenti software utilizzati attualmente nelle sale di emergenza. Il risultato dell'intero lavoro di tesi è stato un prototipo di Emergency Management Information System che implementa al suo interno le funzionalità di design e di ricerca di scenari di rischio in un ambiente condiviso.

Il progetto SocialEMIS ha come obiettivi principali la creazione di un prototipo basato su un'architettura orientata ai servizi (SOA), in grado di integrare al suo interno diversi flussi di dati provenienti da sistemi remoti indipendenti. Da sottolineare l'importanza, da un punto di vista dell'integrazione dati, di aver implementato una web application in grado di poter fruire delle informazioni provenienti da sistemi hardware come le Wireless Sensor Network, strumenti sempre più indispensabili nella gestione delle emergenze.

Un aspetto che non è stato oggetto del lavoro di tesi, ma che è molto importante da un punto di vista dell'implementazione software, che può essere impostato come lavoro futuro sul progetto SocialEMIS, è la validazione del prototipo da parte di un esperto di dominio. Avere delle indicazioni e dei consigli sull'applicazione e sul suo reale utilizzo è importante per poter migliorare quegli aspetti che solo l'esperienza e le conoscenze di un esperto di emergenze possono far emergere.

Avendo sottolineato più volte all'interno del documento di tesi l'importanza della politica 'Social' e di condivisione dell'informazione, un altro aspetto che può essere approfondito ed integrato in futuro nell'applicazione è il *crowdsourcing*. Dal punto di vista del progetto SocialEMIS, integrare il concetto di crowdsourcing, significa permettere non solo all'esperto di emergenze di poter generare e condividere informazioni, ma inserire come attore principale e come fonte di informazione anche il 'cittadino', cioè l'utente generico dell'applicazione.

Il progetto SocialEMIS, nella fase di definizione dello scenario, integra al suo interno anche la definizione di uno schema di base di piano di emergenza. All'interno della

6 Conclusioni

versione attuale del progetto, il piano di emergenza è gestito tramite un editor di testo, all'interno del quale si può definire liberamente un piano di emergenza. Un interessante sviluppo futuro potrebbe mirare all'integrazione del sistema SocialEMIS con un sistema dedicato per la definizione di piani di emergenza.

Bibliografia

- [1] Carlo Carmagnola, C. Ciminaghi (2010). “Scenari di Rischio e SOA: verso il metodo ‘Augustus 2.0’ ” - Elaborato di Laurea - Politecnico di Milano
- [2] Martin Frassl, M.Lichtenstern, M.Khider, M.Angermann (Maggio 2010). “Developing a System for Information Management in Disaster Relief” - Methodology and Requirements - German Aerospace Center (DLR)
- [3] Franclin S. Foping, I.M.Dokas, J.Feehan, S.Imran (2009). “On Using Software as a Service to Deploy an Early Warning Service” - Cork Constraint Computation Center - University College Cork Ireland
- [4] Fabrizio Parisi, L.Tavellin (2011). “Disaster Management Interoperability Service” - Politecnico di Milano
- [5] Ahmed Sobhi Atteih, S.A.AlQahtani, A.Nazmy “Emergency Management Information System: Case Study” - GM, Unicom for Communication Technologies
- [6] Edward Mahinda, B.Whitworth (2004). “Evaluating Flexibility and Reliability in Emergency Response Information Systems” - New Jersey Institute of Technology
- [7] Sarah Underwood (Febbraio 2010). “Improving Disaster Management” - Communication of the ACM
- [8] Andreas Meissner, T.Luckenbach, T.Risse, T.Kirste, H.Kirchner “Design Challenges for an Integrated Disaster Management Communication and Information System” - Fraunhofer IPSI, Fraunhofer FOKUS, Fraunhofer IDG - Berlin
- [9] Liz Carver, M.Turoff (2007). “Human Computer Interaction: The Human and Computer as a Team in Emergency Management Information Systems” - Communications of the ACM
- [10] Tim Grant (2008). “A Checklist for Comparing Emergency Management Information Systems” - Netherland Defence Academy

Bibliografia

- [11] Xiang Yao, M.Turoff, R.Hiltz (2010). "A Field Trial of a Collaborative Online Scenario Creation System for Emergency Management" - International ISCRAM Conference
- [12] Jovilyn Therese B. Fajardo, C.M.Oppus "Implementation of an Android-Based Disaster Management System" - Department of Electronics Communication and Computer Engineering - Manila University
- [13] Giovanni Savazzi (2005). "Studio Geologico del Versante in sinistra Idrografica del Torrente Varrone" - Comune di Premana
- [14] J.L. Zezere, E.Reis, R.Garcia, S.Oliveira, M.L.Rodrigues, G.Vieira, A.B.Ferreira (2004). "Integration of spatial and temporal data for the definition of different landslide hazard scenarios in the area north of Lisbon" - Centro de Estudos Geograficos - Lisboa Portugal