POLITECNICO DI MILANO

Scuola di Ingegneria Industriale e dell'Informazione

Corso di Laurea Magistrale in Ingegneria Informatica

PharmaHosp
for the e-prescribing management in the hospital environment

Supervisor: **Prof. Francesco Pinciroli**

Co-supervisors: **Ing. Sara Marceglia**

Ing. Lorenzo Margola

Degree thesis of:

Marco Pasini 764225

ACADEMIC YEAR 2012-2013

TABLE OF CONTENTS

LIST OF FIGURES

1. ABSTRACT

Managing pharmaceutical products is a sensitive topic for healthcare providers.
From production to administration, the item's life cycle is threatened by human errors and frauds.
Health ministry's effort concentrates in achieving a reliable control over this process by monitoring production, tracing logistic, obtaining data about usage and providing withdrawal guidelines.
In 2004, the Italian government approved a law enabling this control. The decree only delineates handling outside healthcare structures, not inside them.

PharmaHosp main purpose is to achieve the same control inside healthcare providers.
The result relies on searching successfully among pharmaceutical data, tracing movements through operational units, providing interfacing and integration for the administrative and the prescribing worlds.
Structuring the pharmaceutical domain emphasized the issue of querying data derived from heterogeneous sources. Observing ontologies, as Rx-Norm, provided an idea of the core attributes and entities of a drug and the resulting decision to implement a search procedure capable of weighting terms. In time, these weights will be tuned with respect to the history of searches.
PharmaHosp speaks the language of the management and of the prescriber; hence, it provides an interface that works by codes and commercial names and by active ingredients and dosages.
Monitoring is defined as recording every movement in the domain and finding products by means of batch and expiration date. Labeling items with unique identifiers and using barcode readers reduced human errors and introduced a reliable monitoring.
Being part of wHospital™ Framework, a healthcare software developed

by Laserbiomed™, PhamaHosp objective of integration lies in communicating with therapy's modules, with administrative modules and eventually external software.

PharmaHosp is configurable for two types of providers: PharmaHospital adopts a hospital-like logic while PharmaHospice tunes its behavior for homecare scenarios, which are in mobility and involve a stricter user to products relation.
Three healthcare providers are actually beginning trials of the latest release of PharmaHosp.

1.1 ESTRATTO IN ITALIANO

Ricerche preliminari

La gestione di prodotti farmaceutici è da sempre uno degli argomenti più delicati per il sistema di assistenza sanitaria. Dalla produzione alla somministrazione, il ciclo di vita di un farmaco è costantemente minacciato dalla possibilità di errori umani, frodi e furti. Seguendo l'esempio di nazioni estere, il ministero della salute sta concentrando i propri sforzi per ottenere un grado affidabile di controllo sull'intero processo che includa opere per supervisionare la produzione, per mantenere la tracciabilità di tutte le movimentazioni che avvengono dalla fabbrica all'azienda sanitaria, per conservare informazioni riguardo le singole somministrazioni e le statistiche sull'utilizzo nei vari territori, ed infine per realizzare linee guida che gestiscano il ritiro dal commercio quando insorgono eventi eccezionali. Nel 2004, il governo italiano ha sviluppato assieme all'Agenzia Italiana del Farmaco ed al Ministero della Salute una serie di decreti legge che adotteranno attraverso varie fasi di sviluppo questo livello di controllo. Il Progetto Tracciabilità del Farmaco, sebbene proponga requisiti dettagliati per tutta la parte di gestione che avviene al di fuori delle strutture, non tratta la gestione interna all'azienda sanitaria. Infatti è richiesto alle strutture solamente di comunicare l'approdo di un prodotto nel loro magazzino ed il momento in cui questo ne esce come somministrazione o come consegna ad un paziente.

L'obiettivo principale di PharmaHosp è dunque quello di completare questa fase del processo e di raggiungere il necessario livello di controllo e gestione anche all'interno delle aziende sanitarie. Per ottenere questo obiettivo è necessario conseguire tre risultati principali: ricercare con successo nel database dei prodotti farmaceutici, tracciare con precisione tutta la movimentazione

attraverso le varie unità operative, fornire integrazione e interfacciamento sia con il mondo amministrativo dell'azienda sanitaria che con quello delle prescrizioni.
Un altro obiettivo di PharmaHosp sarà la possibilità di essere configurabile per due tipologie di strutture: PharmaHospital assumerà le logiche di un'azienda ospedaliera e quindi gestirà la movimentazione attraverso unità operative, armadi di reparto e carrelli di corsia. PharmaHospice invece cambierà determinate logiche per rendere possibile la gestione di strutture di assistenza domiciliare che introducono una relazione più stretta tra operatori sanitari e prodotti in loro gestione, e devono supportare scenari di funzionamento in mobilità.

Metodi

1) **Modellare il dominio farmaceutico** ha posto in evidenza il problema di lavorare con una fonte dati composta da informazioni di natura eterogenea. Tale dominio infatti non è solamente composto da farmaci, ma anche da dispositivi medici, parafarmaci, sostanze e altre tipologie, ed ognuna di queste è correttamente definita da attributi diversificati. Per esempio i farmaci sono identificabili per principi attivi, grammatura e forme farmaceutiche, mentre i dispositivi medici dipendono più dal loro nome tecnico e dalle loro dimensioni.
Ne consegue che organizzare le informazioni a disposizione e studiare dei metodi per ricercarle è un aspetto chiave del prodotto in sviluppo.
Un caso di studio utile a questo processo è stato Rx-Norm™, un dizionario di terminologia medica localizzato per gli Stati Uniti d'America. Essendo implementato tramite UMLS™ implementa le logiche di una rete semantica e realizza una possibile strutturazione dei soli dati riguardanti i farmaci in senso stretto.
A livello italiano, la sorgente dati fornita da FarmaData™ non implementa soluzioni di questo tipo ma fornisce un tracciato completo di tutti i prodotti utilizzati all'interno delle aziende sanitarie, quindi non

solo di natura farmaceutica.
Scopo dello studio sarà quindi fare incontrare questi due differenti approcci alla gestione del dato farmaceutico.
Per poter implementare la tracciabilità dei prodotti all'interno di una struttura è necessario definire la precisione con cui si vuole identificare gli stessi. Per esempio, si potrebbe identificare univocamente ogni singolo prodotto ma questo introdurrebbe replicazione dei dati e un lavoro insostenibile per il magazziniere che dovrebbe etichettare e schedare ogni singolo oggetto. L'analisi del dominio ha evidenziato che i farmaci possono considerarsi equivalenti una volta raggruppati per il loro codice AIC, ovvero di Ammissione in Commercio fornito da AIFA™, per il loro lotto e la loro scadenza. Il ritiro dal commercio avviene sempre tramite questa tripletta e adottando questa soluzione a livello gestionale si può introdurre più elasticità sia nel processo di etichettatura che di smistamento: infatti non è necessario spedire ad un unità operativa la precisa scatola, ma si può spedire una delle scatole del determinato lotto.
Uno degli ostacoli nel definire un'**ontologia dei prodotti farmaceutici** riguarda la forte contestualizzazione dei dati rispetto alla nazione di appartenenza. Ne deriva la necessità di concepire un ontologia che esuli da contesti e regolamentazioni di natura nazionale, e di collegarsi a questa con un'ontologia di dominio specifica per le realtà di utilizzo.

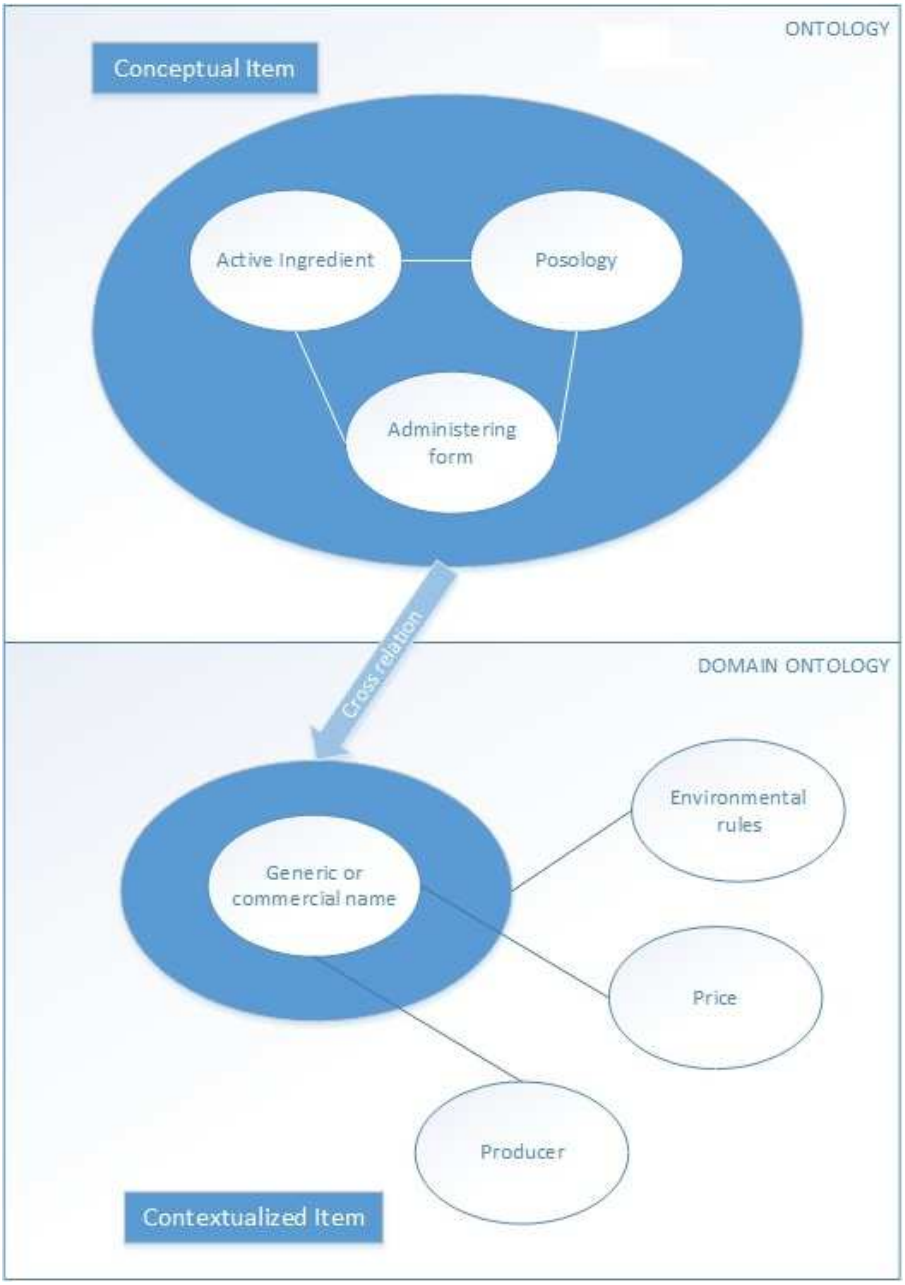


Image 1 – Bozza dell'ontologie di dominio

2) Il progetto PharmaHosp si cala nella realtà del **framework wHospital™** sviluppato da Laserbiomed™. Caratteristica di questo software è di essere composto da molteplici moduli capaci di essere adattabili e personalizzabili per le richieste del cliente. wHospital™ può prendere in carico tutta la gestione ospedaliera e della cartella elettronica, oppure parte di essa attivandolo in specifiche unità operative e può essere esteso mediante moduli di gestione delle terapie, oppure delle farmacie con PharmaHosp. E' attualmente

installato in dodici aziende sanitarie nella regione Lombardia e fa parte del progetto di ristrutturazione del servizio sanitario nazionale dello stato della Georgia.

Durante lo sviluppo di PharmaHosp si è tenuto conto del tipico **ciclo di sviluppo del software** composto da:

- Analisi dei requisiti.
- Fase di design.
- Implementazione.
- Fasi di test.
- Eventuali passi evolutivi seguenti che faranno riiniziare il processo.

L'analisi dei requisiti avverrà secondo le necessità evidenziate da Casa di Cura del Policlinico di Milano™ per quanto riguarda la modalità ospedaliera e da quelle proposte da Fondazione Maddalena Grassi™ per la modalità domiciliare.
Ottenuti tutti i dati, anche tramite studi delle logiche utilizzate attualmente nelle strutture, questi saranno uniti in uno studio di fattibilità che analizzerà le scelte più idonee sia a livello di costi che di tempi di sviluppo. A questo punto si tradurranno le conclusioni in un *design* che fisserà il più possibile le decisioni e i parametri di sviluppo. *La fase implementativa* sarà affiancata nelle fasi avanzate da *alfa test* e *test di sistema* per verificare l'integrazione in wHospital™. In seguito PharmaHosp verrà *distribuito ai clienti* per collaudo e beta test per poi entrare in fase di produzione e passare a fasi di manutenzione e di studio di sviluppi futuri.

Risultati

Due risultati principali sono stati conseguiti durante lo sviluppo di PharmaHosp.

1) Una astrazione del database dei prodotti farmaceutici che permetta la sua gestione e la sua interrogazione in maniera funzionale e rapida.
2) Lo sviluppo della suite PharmaHosp che racchiude gli aspetti di:

2.a) Il tracciamento preciso di tutta la movimentazione attraverso l'intera struttura ospedaliera o di assistenza domiciliare.

2.b) L'integrazione con il mondo amministrativo dell'azienda sanitaria per la gestione economica dei prodotti e con il mondo delle prescrizioni e somministrazioni per collegare i dati di disponibilità dei farmaci con gli effettivi consumi in reparto.

Il **modello dei dati sui farmaci** è stato strutturato identificando gli otto attributi del farmaco ricorrenti per ogni prodotto e quindi estraibili come entità associate all'entità principale:

- Tipo di farmaco.
- Forma farmaceutica.
- Confezionamento.
- Via di somministrazione.
- Attributi di conservazione.
- Codice ATC.
- Produttori.
- Principi attivi.

Questa definizione permette la virtualizzazione dei prodotti farmaceutici e fornisce la possibilità agli attori del mondo della prescrizione come medici e infermieri di prescrivere principi attivi e poi poterli trasformare all'atto della somministrazione in prodotti effettivi.

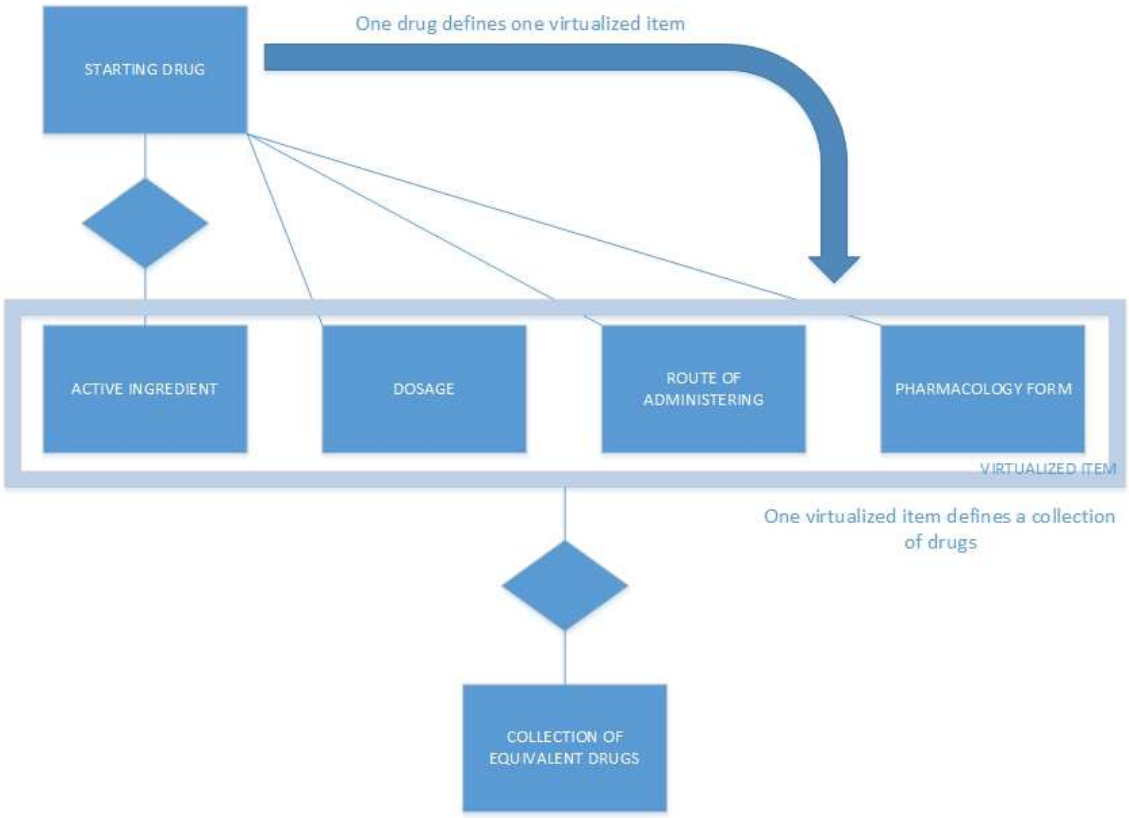


Image 2 – Processo di virtualizzazione dei farmaci

Il modello dei dati della gestione del magazzino invece pone le sue fondamenta sull'entità unità operativa. Questa infatti rappresenta il fulcro di riferimento nella movimentazione degli oggetti all'interno della struttura. Internamente a questa entità si definiscono quindi armadi di reparto e carrelli di corsia ottenendo così uno smistamento più dettagliato e coerente con l'attuale logistica di una azienda sanitaria. Il funzionamento del magazzino si basa sul concetto dei flussi: non esiste un dato persistente che fotografi lo stato di un magazzino, ma viene sempre ottenuto come differenziale fra prodotti entrati e prodotti usciti. Inoltre il flusso degli oggetti procede solamente verso l'alto, dal magazzino fino ai carrelli, mentre il flusso delle richieste viaggia in senso contrario dalle unità operative fino al magazzino. Esiste una procedura eccezionale che consente al prodotto di tornare indietro alla farmacia in modo da gestire eventuali resi. Rispetto allo scenario ospedaliero, quello domiciliare necessita di una struttura più rigida: gli operatori con le loro borse diventano i reali

riceventi dei prodotti che poi li smistano nelle borse dei pazienti, ovvero i prodotti lasciati presso il domicilio. Questa relazione più stretta fra operatori e prodotti a loro disposizione deriva dalla necessità di rendicontare e monitorare più precisamente le attività dei singoli attori del sistema piuttosto che quelle dell'intero reparto.

Lo **sviluppo di PharmaHosp** è partito invece da una *analisi dei requisiti* prodotta dai due clienti ha generato i seguenti casi d'uso essenziali per il funzionamento del sistema sia in modalità PharmaHospital che PharmaHospice:

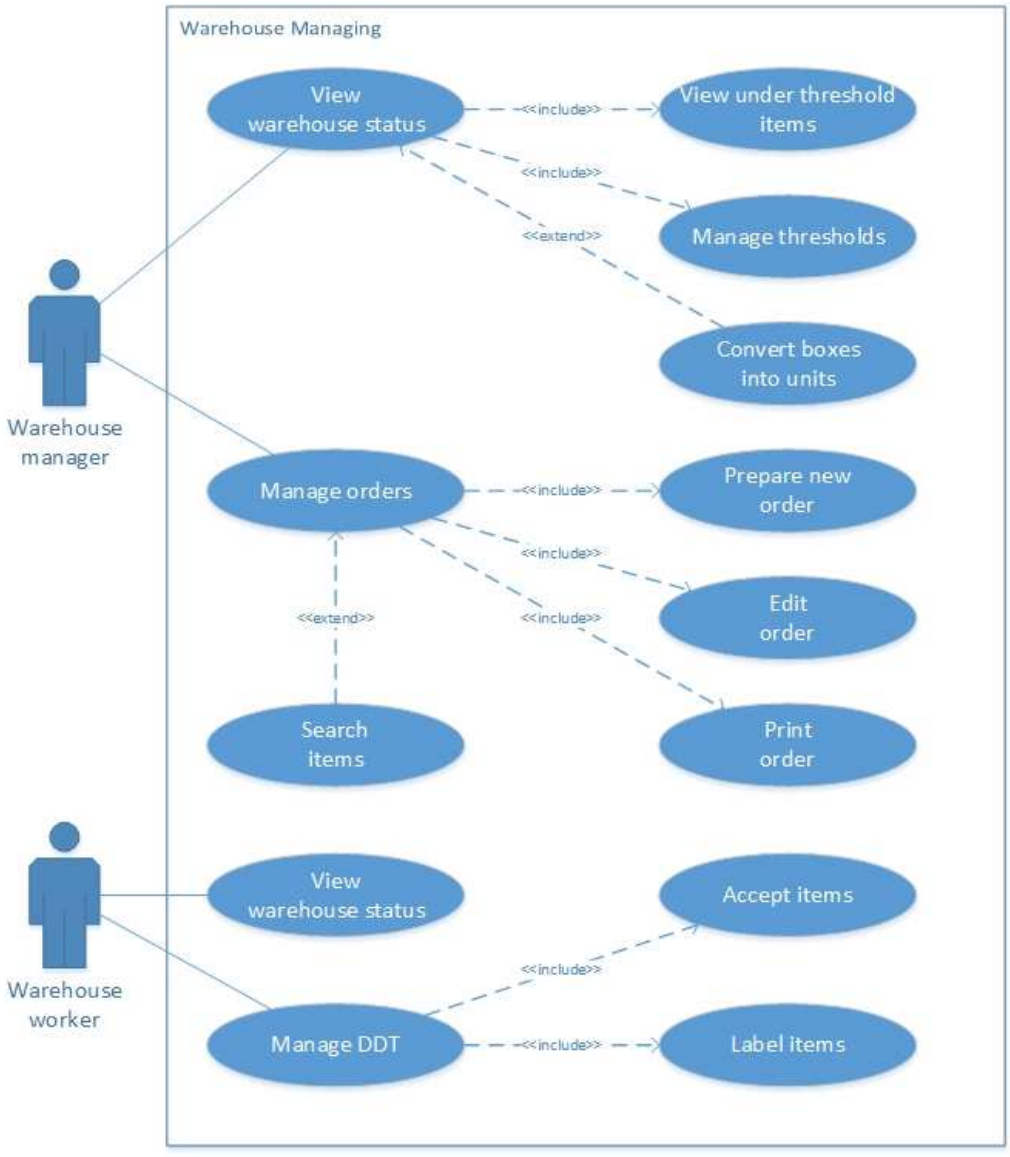


Image 3 – Caso d'uso: Gestione del Magazzino

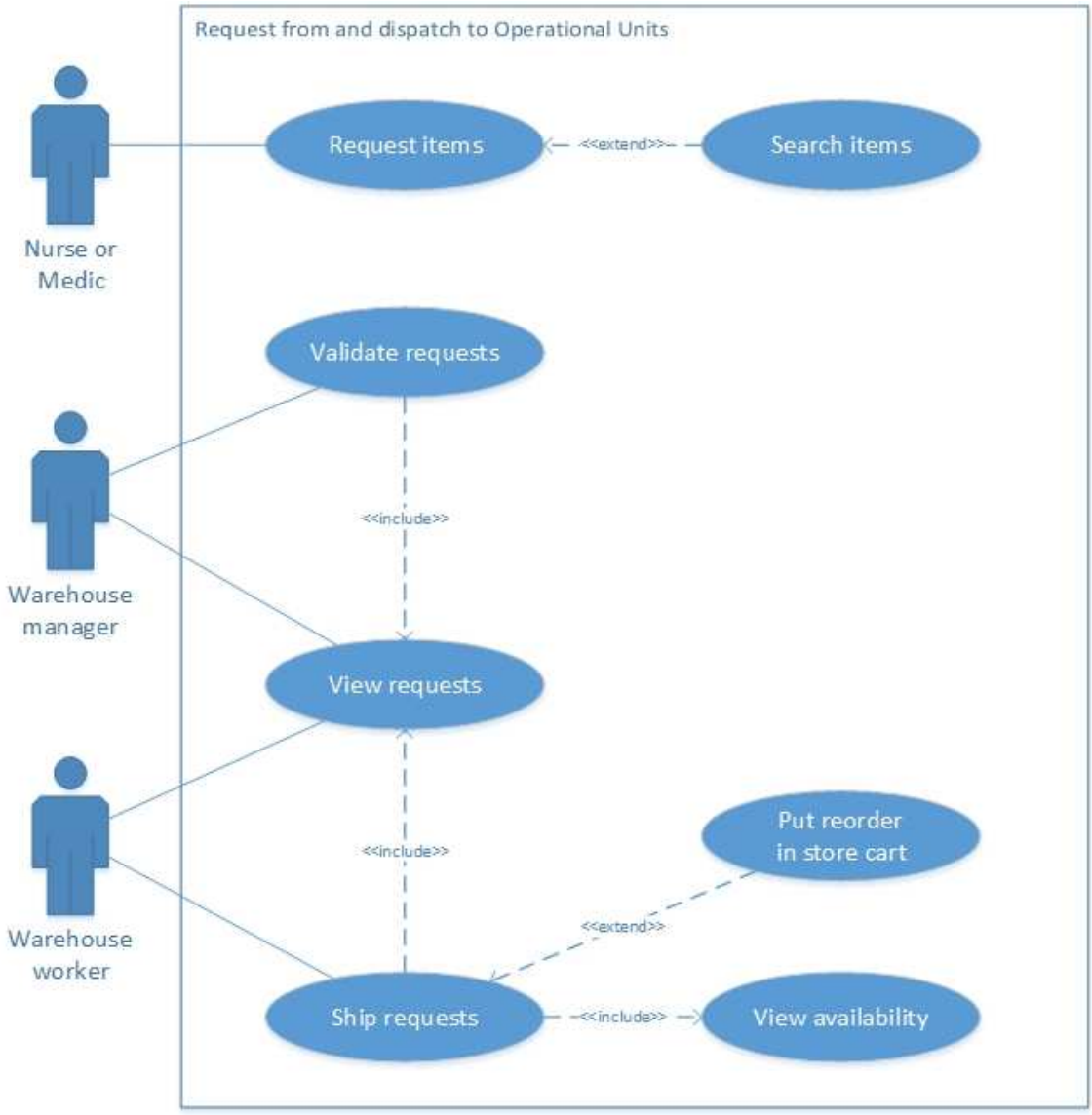


Image 4 – Caso d'uso: Richieste da e Invii alle Unità Operative

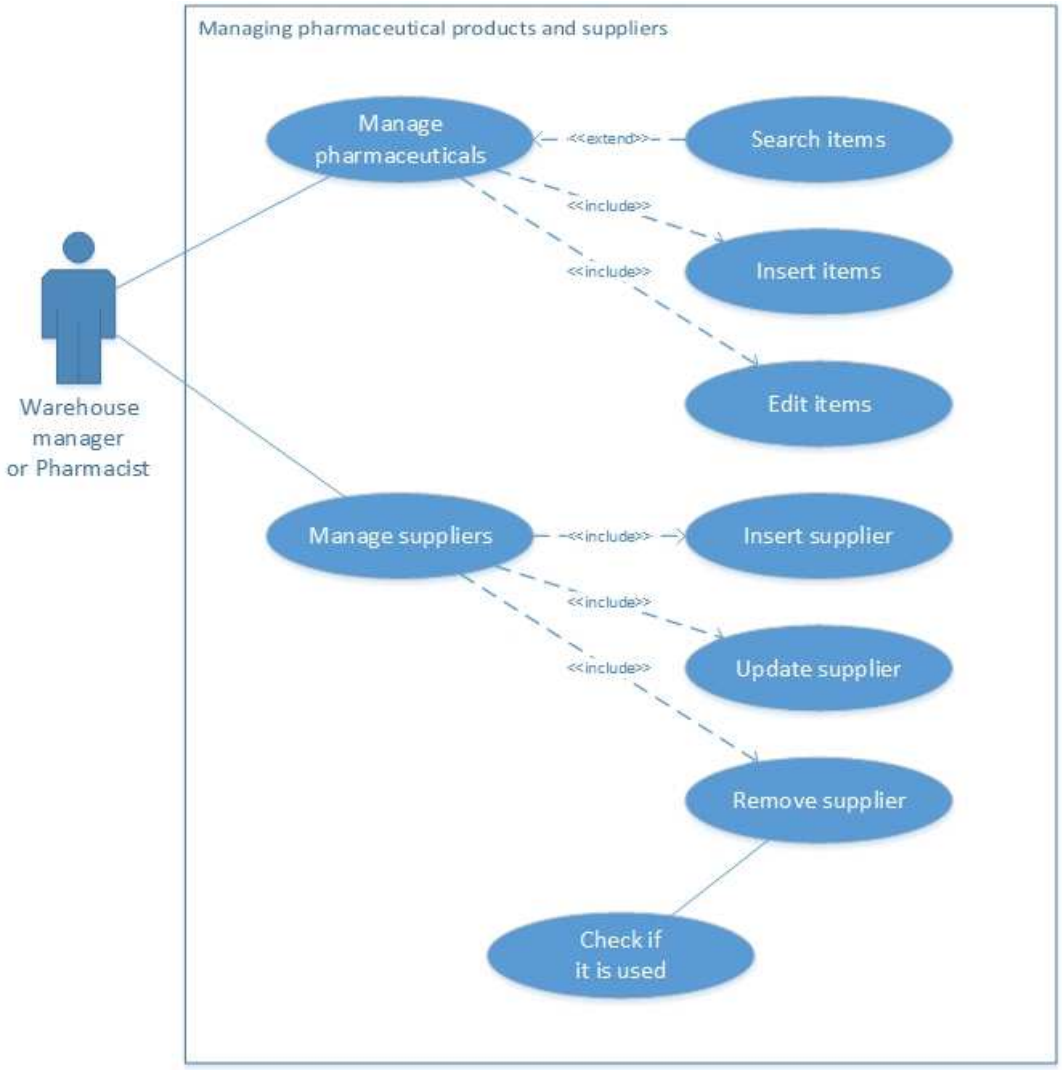


Image 5 – Caso d'Uso: Gestione dell'anagrafica dei farmaci e dei fornitori

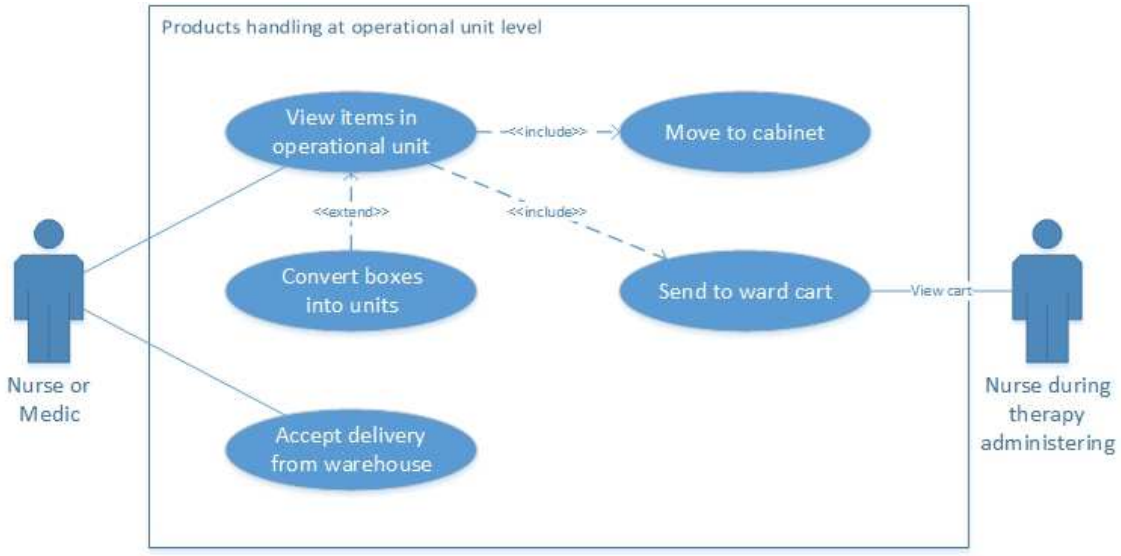


Image 6 – Gestione dei prodotti all'interno dell'Unità Operativa

Per il corretto funzionamento della modalità PharmaHospice devono essere presenti anche i seguenti casi:

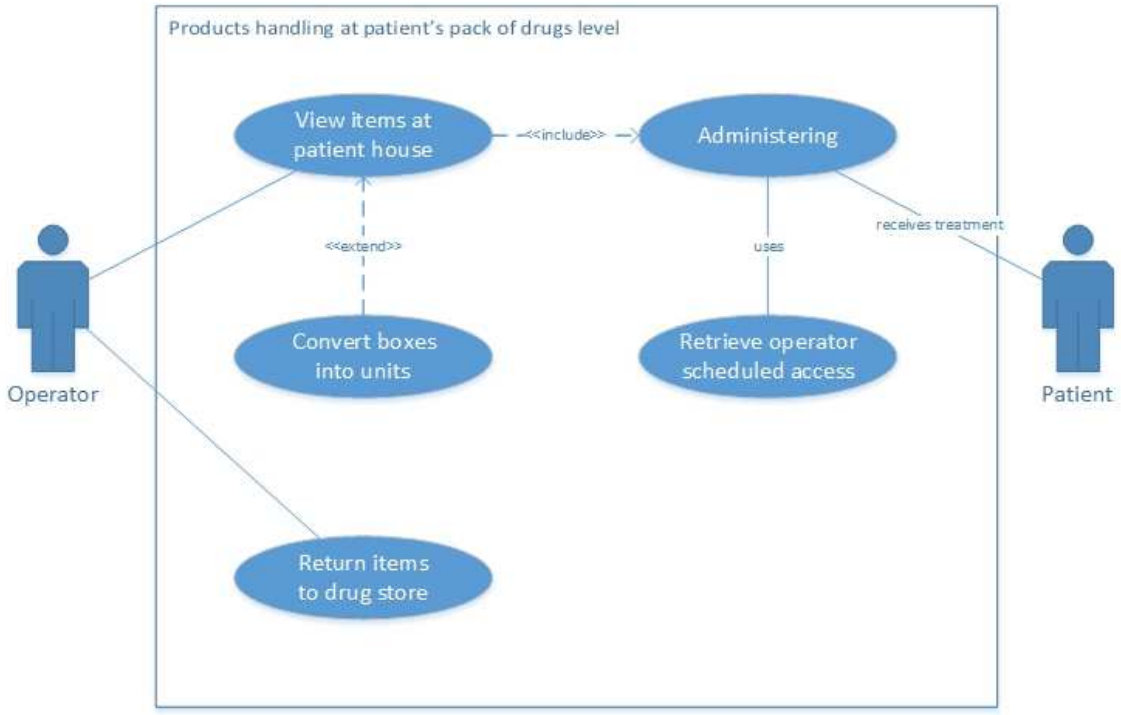


Image 7 – Gestione dei prodotti in modalità PharmaHospice nella cura del paziente

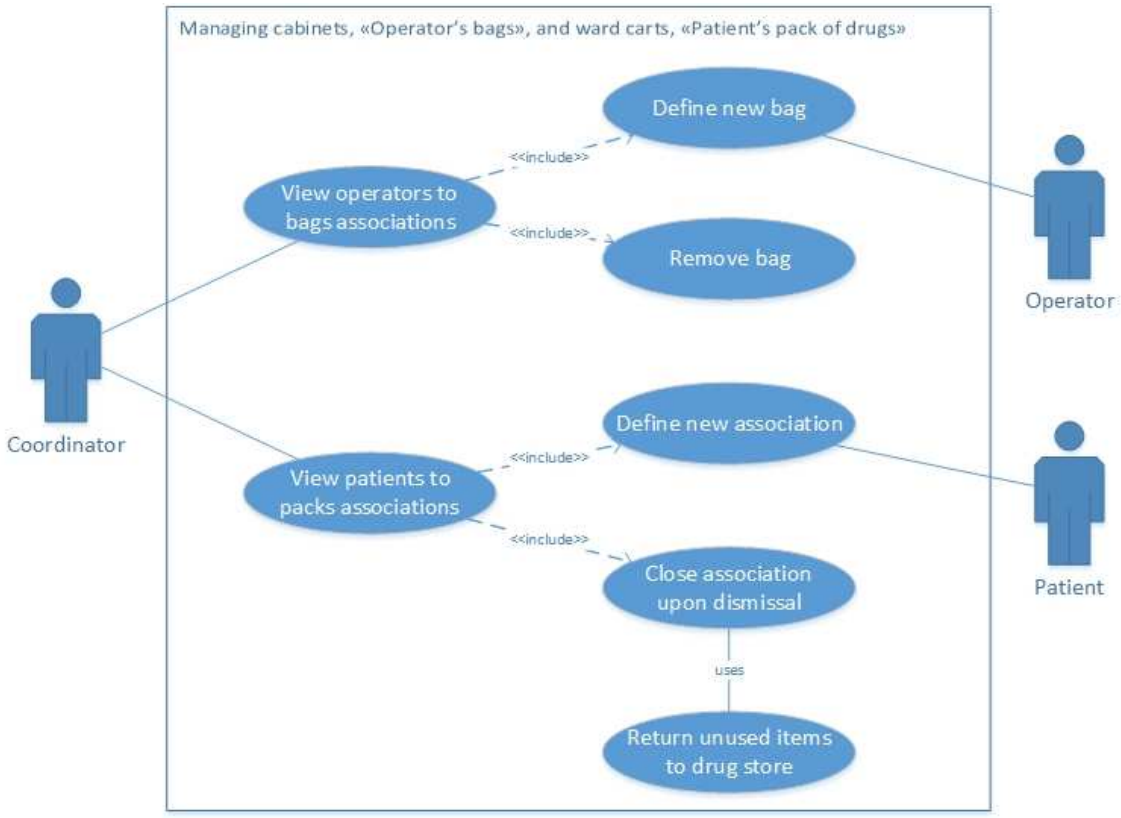


Image 8 – Definizione in modalità PharmaHospice delle borse degli operatori e dei pazienti

Infine, trasversalmente al progetto devono essere disponibili le seguenti funzionalità:

- Filtrare per prontuari di prodotti e tipologie di farmaci
- Definire per ogni prodotto una quantità minima al di sotto della quale venga sollevata un notifica che avverta l'utente della necessità di riordino. Questo valore può essere accompagnato dalla quantità suggerita per il riordino ed il multiplo minimo di prodotti che possono essere ordinati.
- Gestire prodotti come scatole e poterle trasformare all'occorrenza in singole unità di somministrazione.
- Correggere quantità in magazzino, solamente per i supervisori.
- Utilizzare lettori di codici a barre per identificare mediante codici generati da PharmaHosp i prodotti con la precisione del codice AIC, del lotto e della scadenza.

I punti critici dello *studio di fattibilità* riguardano principalmente il dettaglio che si vuole ottenere sia nelle logiche di smistamento che in quelle relative alla tracciabilità. Infatti potrebbe aver senso movimentare i prodotti per singole unità di somministrazione invece che per scatole. Similarmente, il sistema di tracciamento potrebbe monitorare tutte le scatole in maniera univoca, oppure trattare come equivalenze tutti prodotti con lo stesso AIC, lotto e scadenza. Il primo è principalmente un problema di usabilità: il processo di etichettatura diverrebbe insostenibile senza processi automatizzati e in aggiunta, data la dimensione e la fragilità delle singole unità di somministrazione, sarebbe da studiare un sistema di re-confezionamento dei prodotti. Inoltre non tutti i prodotti sono oggettivamente divisibili in unità, per esempio sciroppi o garze possono essere usati parzialmente ma senza sapere a priori in che porzioni. Di conseguenza la scelta è stata di gestire primariamente per scatole, tranne per certi prodotti, definibili in configurazione, che potranno essere trattati in unità. Inoltre la possibilità di convertire i prodotti in

unità in qualsiasi momento permetterà in fase di somministrazione di associare la corretta quantità.
La scelta di gestire prodotti per lotti e scadenze deriva invece dallo studio dei casi reali. Infatti per medici, infermieri e magazzinieri non esistono evidenti differenze che facciano prediligere un prodotto ad un altro con stesso lotto e scadenza. Inoltre anche AIFA™ ritira prodotti per lotto e scadenza e quindi risulterebbe superfluo introdurre un maggior grado di complicazione nel sistema.
Il sistema di ricerca deve funzionare per codici, per testo e nel caso di dispositivi medici in particolare, combinando entrambe le opzioni. Infatti è parso evidente che se i farmaci sono facilmente identificabili per codice AIC o principio attivo, i dispositivi medici vengono prodotti da tantissimi fornitori diversi e una strategia vincente per trovarli passa tramite la ricerca per codice interno REF e nome del produttore.
Un'altra decisione sulla logica del sistema riguarda la gestione delle richieste dalle unità operative: il dualismo è fra la possibilità di effettuare singole richieste per vari prodotti oppure effettuare molteplici richieste ognuna di un singolo prodotto. Seppure la prima scelta si sposa con l'idea di fare un rendiconto settimanale dello stato dei carrelli e degli armadi di reparto ed inviare un'unica richiesta, la gestione per singole richieste rende tutto il processo più elastico e dinamico. Infatti è possibile per esempio, richiedere prodotti al bisogno invece che aspettare di non averne a sufficienza per fare una richiesta.
Inoltre dal punto di visto del magazzino esistono altri vantaggi: è possibile far attendere richieste ed accettarne altre immediatamente senza l'obbligo di creare stadi intermedi di accettazione. Si tratta di bilanciare tre attributi del software: usabilità, complessità ed rigidità.
L'ultimo punto critico dello studio di fattibilità riguarda il processo di validazione. Questo passo aggiuntivo rallenta il paradigma di funzionamento ma essendo filtrabile per prontuari di farmaci rappresenta un'opportunità per un controllo più rigoroso della gestione di determinati prodotti. Considerando la sensibilità di alcuni dati da

trattare, il sacrificio in termini di rapidità del processo appare una perdita accettabile nell'ottica di aumentare la sicurezza sistema.

La *fase di implementazione* pone al centro del suo sviluppo il sistema di ricerca dei prodotti farmaceutici e la gestione del flusso dei prodotti dal magazzino alle unità operative.
La ricerca utilizza uno schema di soluzione basato su una ricerca testuale seguita da un primo filtraggio dei risultati per tipo di prodotto e successivamente per un eventuale prontuario. Il tipo di testo presente determina la scelta di ricercare per codici, per testo libero o eventualmente di combinare le due ricerche. La ricerca per codice, ovviamente più rapida, si concentra prima sul trovare l'AIC corrispondente e poi in caso non abbia successo esamina EAN e REF, codici non univoci e quindi più lenti da indicizzare. La ricerca per testo libero invece avviene attraverso i campi definiti essenziali per interrogare i dati farmaceutici: nome, principio attivo, forma farmaceutica, confezionamento, produttore. Un sistema di punti assegna un valore ad ogni risultato in modo da evidenziare i prodotti ritenuti più rilevanti.

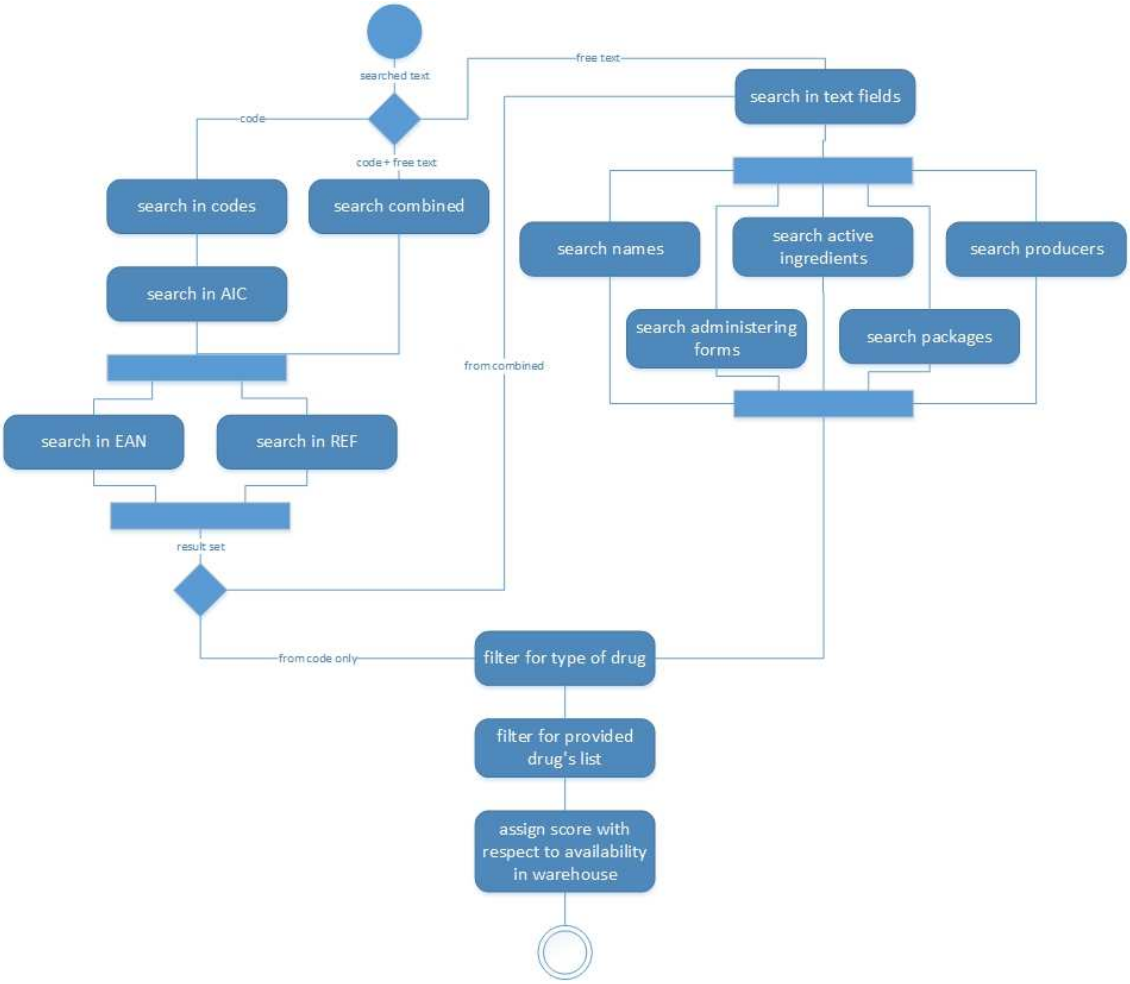


Image 9 –Diagramma attività della ricerca

Alla fine di questa prima ricerca, un ulteriore filtro verifica lo storico dei prodotti trattati dall'azienda e nel caso si trovino corrispondenze il punteggio fin qui accumulato viene nuovamente incrementato. Alla base di questa scelta risiede il principio base per cui i prodotti trattati dall'azienda verranno nel breve tempo trattati nuovamente. In aggiunta, i valori che determinano il peso dei campi di ricerca possono essere perfezionati col tempo, mediante un'analisi dello storico delle ricerche.

I primi *test* che hanno seguito le fasi finali di implementazione si sono concentrati sulle prestazioni della logica di ricerca, le quali risultano essere più che accettabili nel caso del testo libero, dove il carico di lavoro per paragonare i campi di test è più intensivo, risulta sostenibile per l'architettura solamente fino a quattro termini, e i risultati seppur

correttamente ordinati risultano numerosi. Sono invece ottimali nel caso della ricerca combinata dove ricercando per codici si itera su indici ottimizzati e, visto che questa ricerca riduce notevolmente l'insieme dei risultati risulta superfluo cercare per più di un termine chiave, normalmente il produttore. Questa modalità risulta anche più mirata in quanto riduce notevolmente l'insieme dei risultati.

Image 10 – Prestazioni della ricerca per codice e testo

L'attuale implementazione della logica del magazzino e dello smistamento si basa invece su un sistema a tre livelli:

- Il primo rappresenta il punto di ingresso dei prodotti e quindi il magazzino, questo accetta dai fornitori, riceve le richieste dalle varie unità operative e smista verso di esse.
- Il secondo è quello della logica delle unità operative che spostano i prodotti dagli armadi ai carrelli per poter gestire le somministrazioni ai pazienti.
- Il terzo è un livello nascosto all'utente che gestisce tutta la parte di eliminazione di prodotti usati, scaduti o rovinati.

Quest'ultimi evidentemente devono uscire dai flussi dei magazzini ma non possono essere cancellati fisicamente per motivi di rendicontazione e controllo.

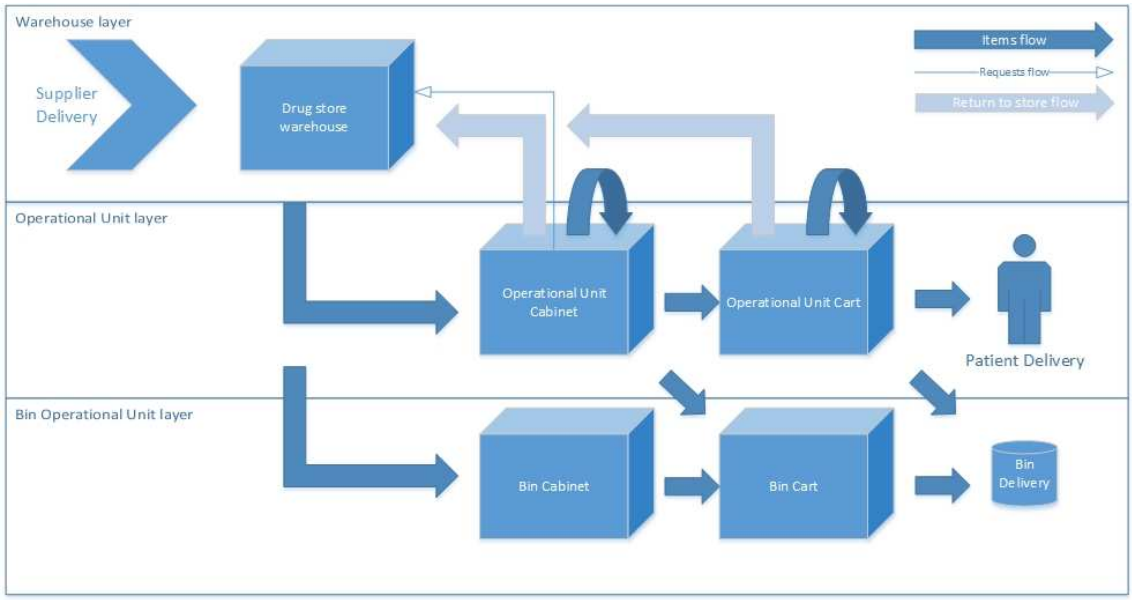


Image 11 – Flussi dei prodotti e delle richieste

La soluzione implementata prevede dunque di creare una unità di cestino parallela al mondo delle unità operative verso cui si mandano tutti i prodotti scartati. Data la logica del sistema in cui gli oggetti si muovono sempre verso l'alto, ovvero dal magazzino all'armadio, poi al carrello, infine al paziente; ognuno dei vari livelli scarterà i prodotti verso il cestino al livello superiore. L'architettura del progetto prevede l'utilizzo di una base di dati dei prodotti farmaceutici popolata con i valori forniti dalla sorgente FarmaData™; su di essa interagisce una libreria dedicata di PharmaHosp, accessibile anche dagli altri moduli di wHospital™ per fornire metodi di integrazione, che viene chiamata da pagine ASP.net e da web services all'interno della applicazione web. I moduli di PharmaHosp sono sviluppati in Silverlight™.

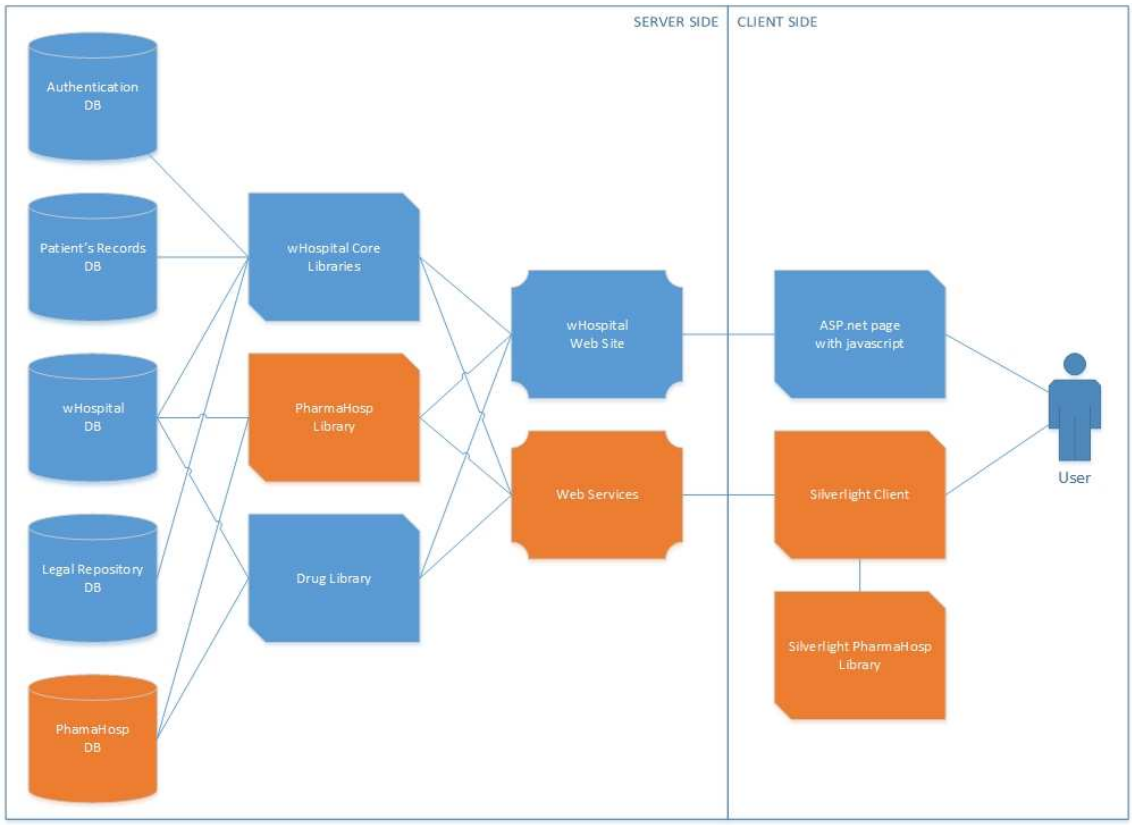


Image 12 – Architettura di PharmaHosp in wHospital

Questa tecnologia permette di sviluppare rapidamente interfacce di visualizzazione e raggruppamento di dati strutturati facendo uso delle tecnologie di data binding ma, siccome funziona come un applicativo client, richiede la replica degli oggetti della libreria anche lato client all'interno dell'applicativo Silverlight™ e lo studio di un meccanismo di serializzazione e de-serializzazione dei dati per realizzare il passaggio dal web service lato server all'interfaccia client.

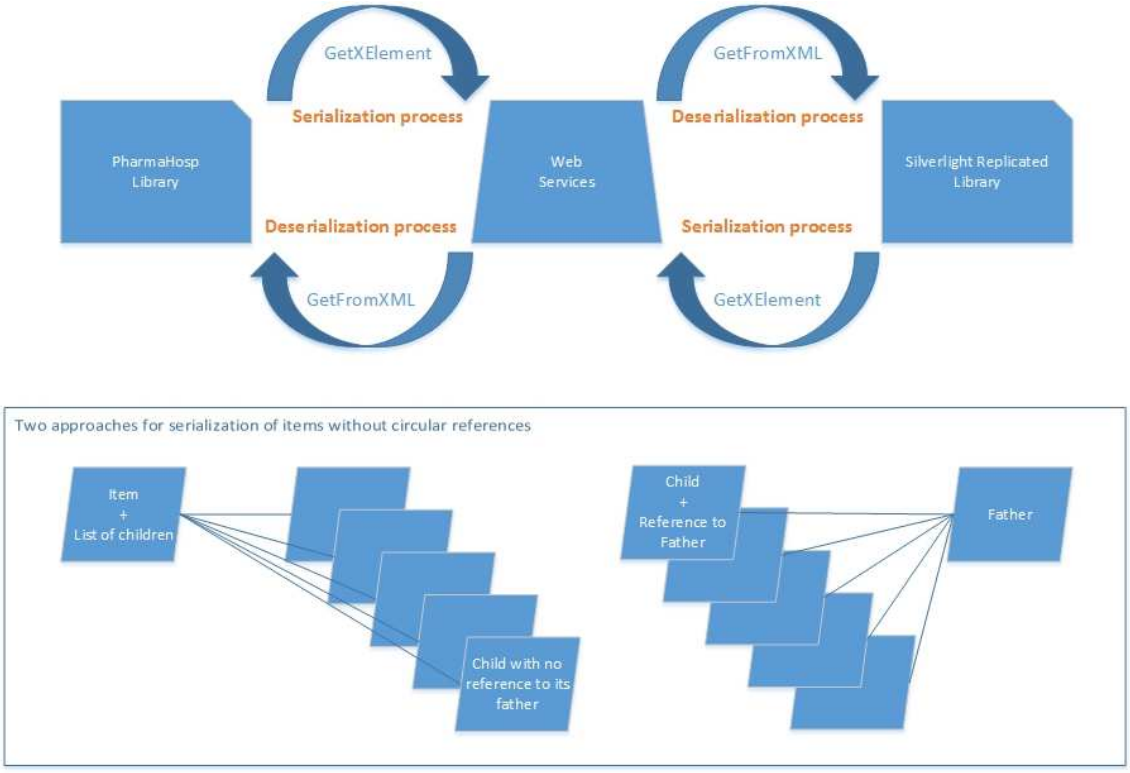


Image 13 – Processo di serializzazione dei dati

Questo processo è critico quando si trasmettono contenitori di collezioni di oggetti in quanto bisogna evitare riferimenti circolari. La struttura adottata prevede il trasferimento di liste di oggetti figli con un riferimento al padre, contro l'equivalente approccio di un padre con la collezione di figli, i quali non contengono l'oggetto padre. Si è preferito il primo approccio in quanto l'interesse sui dati a nostra disposizione è normalmente concentrato sulla collezione piuttosto che sull'oggetto che li contiene. Inoltre nonostante si introduca replicazione del dato padre durante la serializzazione, dato il disaccoppiamento fra l'interfaccia e la logica di business è possibile decorare l'oggetto Silverlight™ con attributi che aumentino la qualità del dato, ad esempio campi calcolati che riportano lo stato attuale di una richiesta.

Sei attori interagiscono con PharmaHosp: fornitori, magazzinieri, responsabili di magazzino, infermieri o operatori, medici o coordinatori, pazienti. Il primo e l'ultimo sono attori esterni al sistema. I moduli messi a disposizione sono:

- Gestione magazzino: che rappresenta la vista per la farmacia sullo stato del magazzino, delle richieste e degli ordini, mentre per le unità operative sullo stato degli armadi e delle consegne.
- Gestione carrelli: rappresenta la vista sui carrelli e nello specifico della modalità PharmaHospice delle borse in consegna nelle case dei pazienti.
- Statistiche: permette di ottenere i primi dati statistici su volumi e valori all'interno dei vari magazzini
- Prontuari: gestisce l'anagrafica dei farmaci e tutti i prontuari
- Configurazione di Armadi e Carrelli: nel mondo statico della modalità PharmaHospital è una configurazione di struttura iniziale, nel mondo più dinamico di PharmaHospice invece gestisce le associazioni per le borse operatori e le borse pazienti.
- Gestione Fornitori: permette la gestione dell'anagrafica dei fornitori.

Conclusioni e Sviluppi Futuri

Uno degli aspetti di innovativi di PharmaHosp è affrontare il problema della tracciabilità e gestione dei prodotti farmaceutici non solo da un punto di vista amministrativo ma anche considerando gli aspetti legati al loro reale utilizzo in ambito medico.
La tracciabilità sulla movimentazione dei prodotti introduce la possibilità di controlli qualitativi come notifiche quando alcuni prodotti sono in scadenza oppure quando la distribuzione dei prodotti non è coerente con il consumo, in aggiunta a quella di base di poter ritirare prodotti all'occorrenza sapendo esattamente dove reperirli.
Esistono dei punti che richiederanno futuri sviluppi, in particolare per quanto concerne la qualità della fonte dati nel caso dei dispositivi medici: infatti la ricerca per termini risulta a volte difficoltosa poiché le definizioni sono per contrazioni di termini o per equivalenti in lingue straniere. Per risolvere il problema si valuteranno database di aziende concorrenti come CodiFa™ oppure l'implementazione di un glossario

dei termini che estenda le ricerche tramite sinonimi del valore inserito.
PharmaHosp si concentrerà anche sull'analisi dei dati raccolti a regime con l'obiettivo di migliorare la gestione e lo spreco dei prodotti.
A livello di sviluppo, verrà completata l'integrazione coi moduli di terapia di wHospital™ permettendo non solo di somministrare prodotti effettivamente disponibili ma anche di pianificare le richieste rispetto alle prescrizioni inserite.
Nel momento in cui l'interfaccia assumerà una struttura consolidata dall'utilizzo si procederà a tradurla con linguaggi meno immediati ma più adatti agli scenari di mobilità come HTML5™ e jQuery™ oppure, mantenendo il legame con le tecnologie Microsoft™, mediante applicazioni per Windows 8™ che girino anche sulle architetture ARM™ tipicamente usate nei tablet.

Attualmente PharmaHosp è in fase di collaudo presso tre strutture della regione Lombardia:

- Casa di Cura Privata del Policlinico™ che la utilizza in modalità PharmaHospital.
- Fondazione Maddalena Grassi™ che utilizza la modalità PharmaHospice.
- Associazione Il Mantello™ Onlus, attualmente concentrata nella gestione del semplice magazzino farmaceutico.

2. BACKGROUND

2.1 Drug representation models

The pharmaceutical world can be approached by two parallel point of views. One which reflects correct management, monitoring and dispatching of items, which is the way managers and warehouse employees understand products, and the other one, closer to doctors and nurses, of the correct ways of searching and administration. Both aspects are central in an environment that aims to keep a precise and detailed trace of every movement inside the healthcare structure. Those first core requirements introduced the need for a deep analysis of the data source and structure available.
The analysis of this domain must consider three different aspects:

- How to organize the information.
- How to search among it.
- The correct meanings to retrieve it and display it.

The pharmaceutical environment is a wide domain composed by very heterogeneous items. In fact, there can be drugs, over-the-counter medications, materials, substances, medical devices and more. All of these have their relevant and distinguishing information: for example, core data regarding drugs are the pharmaceutical dosage form or the posology whereas medical devices are mostly defined by their sizes or dimensions.
There is, though, one concept that can be applied to all items: different elements with same name, intended as unique identifier, same batch and same expiration date are conceptually equivalent. This is because nothing else in this abstraction of the environment differentiates elements with this exact triplet. For example, when a product is withdrawn from the market, it is commonly identified by its name, its

batch and its expiration date.
Hence, grouping drugs inside the pharmacy by this triplet was considered the starting key point leading towards a correct management application.
Nevertheless, although this constraint will be at the core of all further considerations, the need to organize all the additional information for each item was equally crucial. Identification of a product does not entail understanding its usage or purpose.
In fact, each pharmaceutical product has its key values and descriptions. Over-the-counter products have pharmaceutical dosage forms, packages and routes of administration. Drugs have in addition posology and active principles. Medical devices are defined in a simpler way correlated to their sizes.
Therefore the need to structure these data by keeping all the information common to each item together and structuring the other ones specifically for each type arose as a core requirement as well. This process entailed that, however the data was structured, it would be necessary finding a clever way to search among it. Drugs cannot be searched only by name and even if the search by the triplet (name, batch and expiration date) can have some sense, it is useful only when the user is aware of the specific element he needs, not when the search is focused on a product among many which are suitable.
In the healthcare environment, the entry point and the dispatch among operational units relies on knowing exactly what users are sending, this entails using the triplet defined. But other actors, meaning doctors, nurses and everyone who requests items for their medical purpose, unaware of the element at disposal in storage units, simply cannot rely on this information but need to find the desired product in alternative ways.
At this stage, analyzing different patterns to interrogate our data source became the third core requirement. For example, a nurse requesting a drug to be administered will search it by its name,

commercial or generic, but due to the vastness of the database he will possibly too many matches and will lose time identifying the correct item.

Therefore it is important to enable searching by other meaningful fields regarding the type of product. For instance, in the case of drugs a possibility can be searching by active principles and pharmaceutical dosage forms. In the case of medical devices, producers, technical names and sizes would play a key role.
However an ideal scenario, which does not diminish these considerations, would focus as far as possible on working by means of codes and identifiers since they would reduce vastly human mistakes. There are three main identifiers in our source database:

- AIC Code: the code for admission into commerce given by the Italian government, which is unique but assigned only for drugs and over-the-counter medications. Our source data uses it also for all the other items but as a unique key in the database and not a code which can be found on the products and therefore it is rather useless outside the first two groups.

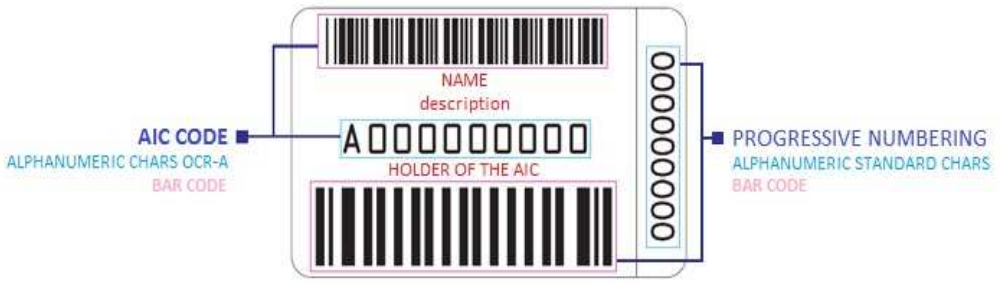


Image 14 - Italian drug label

- EAN Code: it is the code defining the International Article Number, originally European Article Number. It is given for over-the-counter medications and usually medical devices, but it is not mandatory to show it clearly on a product.

- REF Code: it is the internal code used by the producing factory. Being used for internal reference, it is not unique but it give a mean of identification especially for medical devices, which are not fully covered by any of the previous two codes.

Introducing these codes eased the job of searching among products since it gave a fast way to retrieve data when in possess of the code, a list of codes or the product itself. It is almost failsafe when drugs are involved. At this step, an issue is obviously the easy retrieval of a code since no person can be reasonably asked to memorize them and if a box or a list is not at hand, there is no way to know them. A clever way to interrogate the data source is still needed.
Two main strategies were studied:

- Due to their more tree-like structure, drugs and medications need to be queried by different means as active principles, pharmaceutical dosage forms, routes of administration and more.
- Medical devices do not have this kind of structure and have the tendency of being similar products realized by multiple companies, hence it is more relevant to concentrate on producers and internal codes, and when not in possess of the latter, by specifying names related to the range belonging to the product.

Both these approaches were successful in reducing the result set to couples of items, tens in worst cases, thus allowing the retrieval of a requested item with an acceptable time consumption and without setting multiple filters.

2.1.1 RX-NORM

RX-NORM™ is the name of a US-localized medical terminology that contains all pharmaceutical products and medications available on the US market. It is part of the UMLS, Unified Medical Language System, terminology and the National Library of Medicine™ responsible for its maintenance.

UMLS makes use of a semantic network: it consists of a set of broad subject categories, defined as semantic types, which provide a defined and consistent categorization of all the concepts inside the UMLS Metathesaurus, a glossary of all the terms from the source vocabularies. A set of relationships, known as semantic relations, interconnects different semantic types (U.S. National Center for Biotechnology Information, 2009).

The purpose of RxNorm is to provide normalized names for clinical drugs (both generic and branded), realizing a unique structured source of data that supports semantic interoperation between drug terminologies.

It also poses itself as a container for all the heterogeneous drug vocabularies commonly used in pharmacy management (U.S. National Library of Medicine, 2013).

The need for Rx-Norm comes mainly as a consequence of this latter aspect. In fact, hospitals, pharmacies and whatever actor connected with drugs in their working processes needs to interface with different systems that can possibly use different sources for the data. This issue is addressed by normalizing names and unique identifiers and then providing an efficient and unambiguous source of information.

A limit of this system is the fact that it considers clinical drugs and drug packs but not all those items usually managed only in healthcare and hospital environments such as radiopharmaceuticals, contrast media, food, dietary supplements or medical devices.

Using the set of semantic types and relations given by Rx-Norm brings

new possibilities in querying a data source.
Wrapping items with all its defining concepts creates the chance for more refined and potentially useful queries which can enhance the quality of the results.
For instance, it is not only possible to search items by name or active ingredients, but given the dose form it is possible to obtain all the products defined by it no matter what route of administration or pharmaceutical dosage form they possess. It enables a more precise concept of virtualization of drugs: by defining key terms that implement the concept of "virtual", for instance posology and drug form, rather than route of administration, it is possible to filter and interact with products virtualized, not using anymore the precise item until it is actually needed but using an abstraction that includes it and all the equivalents.

2.1.2 AIFA and FarmaData

AIFA™, Agenzia Italiana del Farmaco, is the public office that regulates pharmaceutical products in Italy. It keeps track of all the processes that a product must endure: from registration and admission into commerce to compliancy and quality tests in all the factories that realize them, from safety checks to pricing, from attributing eventual refund ranges to the Health Technology Assessment activities.
A strong aspect of this institution is that, by realizing all the decisional process inside the agency, it is guaranteed the unicity of the pharmaceutical system and the equity towards citizens in accessing it when health is involved.
In addition, it works close to counties which distribute products and the production layer in order to avoid wasting and govern expenses with the objective to avoid exceeding the annual spending quota decided by

the government.
In details, it grants:

- Access to pharmaceutical products, granting safety and correct use in healthcare.
- Unicity of the drug environment on a national level.
- Control over pharmaceutical expenses and competitiveness.
- Innovation, efficiency and ease of usage of all the registering procedures, especially for innovative drugs or rare diseases' medications.
- Strengthen the relationships with foreign countries' agencies and international organizations.
- Supports and rewards Research and Development investments.
- Promotes the knowledge and culture derived from pharmaceutical products by gathering and evaluating international best practices.
  (Agenzia Italiana del Farmaco, 'L'autorizzazione all'immissione')

The main process entrusted to AIFA is to grant the admissibility into commerce of a product. This authorization is provided once all experimental trials and studies have been completed. The task is taken upon by a technical and scientific commission with support from experts, both internal and external the agency, and the ISS, Istituto Superiore di Sanità, which examines the results of the producer. It then must pass a series of evaluations regarding chemical, biological, toxicological and clinical nature (Agenzia Italiana del Farmaco, 'Procedura Nazionale (AIC)').
Once the AIC, Admission into Commerce, code is granted, the pharmaceutical product obtains a precise identity given by:

- the name
- its composition
- the description of the production methods
- therapeutic indications, counter indications, adverse reactions

- posology, pharmaceutical dosage form, route of administration
- precaution and safety measures for drug storage and administration
- a summary of the product characteristics
- a model of the box
- the drug leaflet
- an evaluation of the risks for the environment

It is clear that for every change in the drug properties a new authorization from the AIFA is mandatory.

In order to uphold citizens health, the agency manages also all cases of withdraw or suspension of a product, usually this includes prohibition to sell new batches of the product and, or, the withdrawal of all or a subset of the batches already in commerce that are suspected to be unsafe.

In this scenario, FarmaDati™, which is our presently available data source, along with other competitors, tries to pose itself as a middle layer between the AIFA work and the need of health care environments to manage all these information.
It implements a structured data source with some basic concepts similar to the RX-Norm case but more aimed towards the completeness of the information available, in terms of quantity, rather than focusing on helping queries by implementing semantic networks.
For instance, FarmaData™ does not only enlist pharmaceutical products but also medical devices, veterinary products, raw materials and health services.
This obviously introduces the issue of populating a structured database with very heterogeneous types of information.

Strong points of this data source are the precision and promptness in spreading news and informations after changes to a product or the introduction of new ones, the guarantee of the correctness of the data,

the conservation of the history of a product with validity periods in order to be able to recreate its evolution and eventually determine whether it is still sellable or not.

2.2 The pharmaceutical products delivery process

Traceability is defined as the ability to verify the history, the location or application of an item by means of a documented recorded identification.

This verification can be applied with different degrees of granularity, meaning being able to tag single items or specific groups of them. In logistics, traceability refers to the possibility to trace goods along a distribution chain by some given information such as batch or series numbers. Some examples are the automotive or food industry where this monitoring is used in order to grant safety and possibility to recall products.

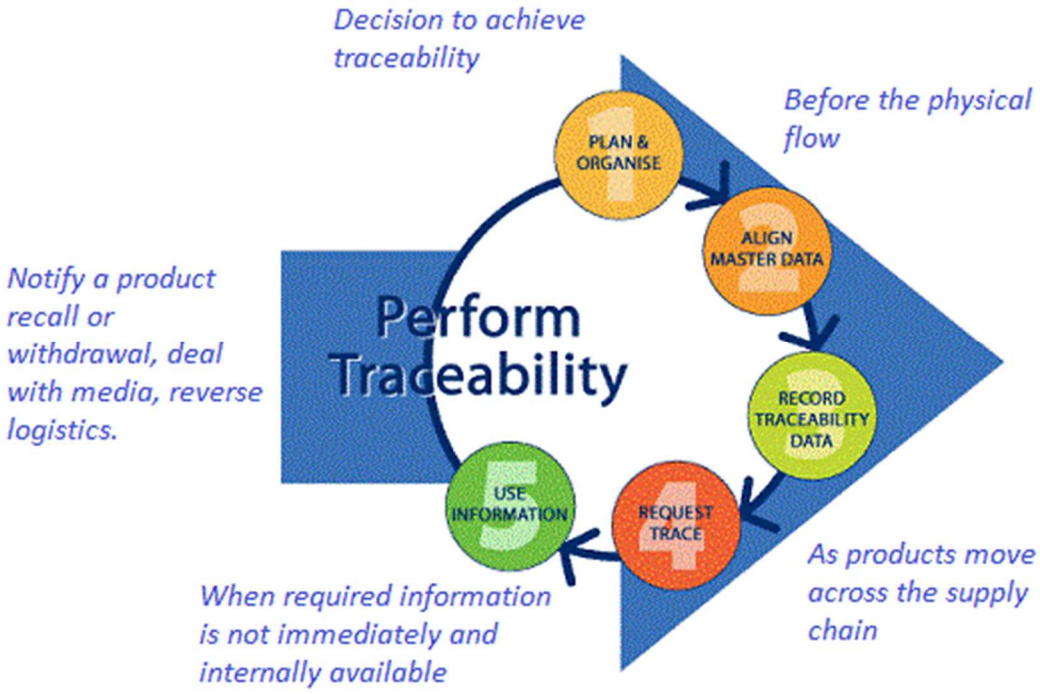


Image 15 – Organizing monitoring

In the pharmaceutical world, traceability is mandatory and precisely organized during the process that brings a drug from being realized to being delivered to a certain customer, as a pharmacy or a hospital. Until 2010, once the drug entered the structure, it was needed to keep track of what is administered to a patient but not to precisely know where something was stored, or when it had been used. Recently the project enlarged towards tracing also prescriptions and internal usage and handling and therefore the need for structures to cover this process has begun.

2.2.1 From the producer to the health-care provider

Pharmaceutical products leave a trace along all the paths from a producer to a client. In fact at every step each box is constantly monitored by the barcode written on it.
This detailed control activity is possible thanks to the actuation by the Ministry of Health and the AIFA of the "Progetto Tracciabilità del Farmaco" (Drug Traceability Project).
It is a safety measure towards public health that takes place by adopting a labeling system, based on uniquely numbered labels which identifies every single type of a drug, and a central database (Agenzia Italiana del Farmaco, 'La tracciabilità del farmaco').
At every step of the production and supply chain, the labels are counterchecked, as in the factory, once arrives to the seller, to the distributor and so on, and their position is transmitted to the central database.
The project aims to obtain the final result of knowing in every possible moment where a specific box of pills or bottle of syrup is, even when it has reached a structure such as a hospital, a pharmacy, or even the patient home.
A secondary objective is to be a way of granting the pertinence of the

prescriptions and drugs usage, in order to contrast frauds, illegal trafficking and easily identifying boxes of counterfeit drugs.
In addition, with the data at hand the agency will be able to better understand the health condition of the country, by translating the analysis of the data related to the drugs' selling into more valuable knowledge regarding evolution of pathologies in the territory, for example if there are variations with respect to past years or other counties. This information could also be useful in order to highlight quickly eventual emergency situations and consequentially adopt more opportune and timely decisions.
The central database for traceability is active in Italy from June 2005. It contains all information about boxes: code, with batch and expiration date, delivery place, date and time of the delivery, quantity delivered and economic value when purchased from the SSN (Sistema Sanitario Nazionale or National Healthcare System) structures.
From this set of data is possible to know:

- The daily number of boxes which enters each drugstore of the territory and addressed to be bought on SSN expenses.
- The daily number of boxes which enters each hospital drugstore or ASL (Azienda Sanitaria Locale or Local Healthcare Structure) and its relative buying prices.

From October 2007 those data had been integrated with information provided by counties and public administrations about usage within SSN structures. This more detailed granularity is realized in an incremental process. In 2010, phase 3 begun: it requires to provide details at a single prescription level with some exceptions for certain structures (for instance, prisons). These data are provided to the database on monthly basis.
In addition, from the same year, data about consumptions of drugs bought from SSN structures and used internally, meaning for outpatient or hospital purposes, is being provided, too (Ministero della

Salute, 'Monitoraggio della distribuzione dei medicinali in Italia'). This information comes in terms of boxes used for ward in a month and monitors deliveries from hospital drugstores towards operational units, considering returns and disposal, both by quantity and economic value. Also these data are provided monthly to the central database. The AIFA drafts periodically the national OsMed report that allows analyzing all data about usage of drugs in Italy on the population, in terms of expenses, volumes and typology. This analysis offers opportunities to understand how pathologies develops on the territory and the consequent usage of specific pharmaceutical products. From this information is possible to interpret the principal reasons that condition the differences in prescriptions for same diseases in different counties (Agenzia Italiana del Farmaco, 'Rapporti OsMed').

2.2.2 Within the health-care provider

In Italy, monitoring the handling of drugs inside a healthcare structure has begun only recently.
As far as today it is mandatory to provide AIFA the following information:

- When a drug enters the structure.
- When a prescription for a specific drug is made.
- When a drug is administered inside the structure.

The first point it is handled when the delivery takes place, but from this moment to the other two there is no specific methodology or approach regarding the traceability issue. The agency needs only to know what enters and what leaves the hospital drugstore.
Obviously, a lot of opportunities comes from tracing drugs inside the structure. In fact, even if the primary reason is to evaluate consumptions and evolution of diseases, at a single structure level

implementing a tracing system opens to the possibility of precisely knowing at every time where a specific batch of a drug is located, being able to prioritize usage by expiration dates, or eventually collecting faulty items at once.
In addition to this immediate vantage points, it will be possible in the long term to examine these data and determine in advance an approximation of what will be needed in the future, reducing to a minimum eventual wasting of products and useless storage occupation.

2.3 Ontologies

Multiple considerations lead to the definition of ontology. Modern systems have developed each one their peculiar business' logic, web services and, usually, data representations. Therefore these systems are poorly integrated one with the other and they often deliver redundant capabilities. Moreover data accessing for not integrated systems can be hard and foster errors, even without considering that this rigidity prevents most secondary uses of data (Smith, 'What is an Ontology').
This heterogeneous situation rise the following questions:

- How to find data
- How to understand it
- How to proficiently use it
- How to compare and integrate

Retrieval is a problem of every domain due to massive non interoperable data models and it is worsening since the same approach is adopted each time a new system is created.
Trying to subvert this trend lead to the definition of semantic technology: each system creates a controlled vocabulary of the commonly used terms and, above it, creates an ontology, binds it to

its data and makes it available on the web. Common terms link different ontologies implementing a web of data.
Ontologies can be defined as standardized labels designed for use in annotations to make the data cognitively accessible to human beings and algorithmically accessible to computers.
In the pharmaceutical scenario an ontology is needed in order to abstract and define the domain at hand and proposing a model that can be sufficiently elastic to adapt to different situations and at the same time capable of allowing intercommunication between data sources.
Since the drugs' environment is strictly related to country laws and restrictions it is difficult to propose a solution of international nature but this does not mean that it could not be a possibility to structure this reality on a level which is above the specific context of a nation.
It requires to provide an ontology that abstracts terms and understandability of pharmaceutical products above laws and languages and, underneath it, proposes another one shaped for a specific context.
An over-domain ontology for shaping the definitions and a domain related one for actual usability.

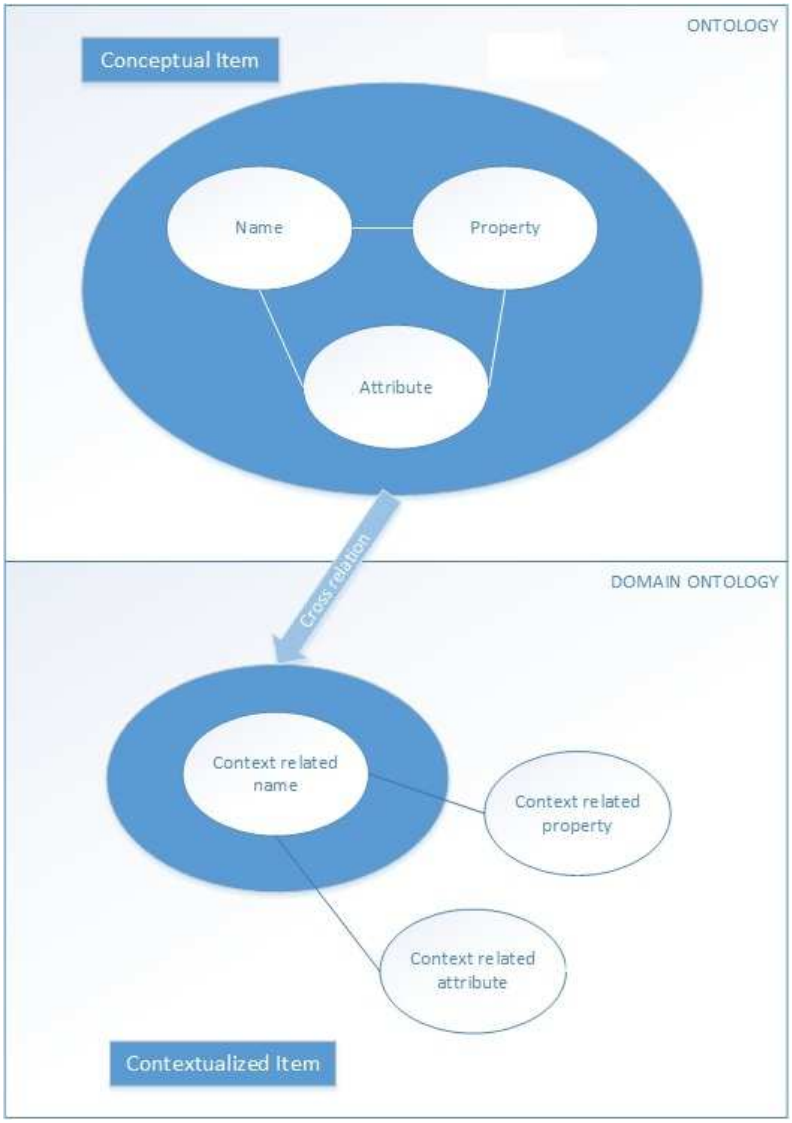


Image 16 – Domain ontology

The pharmaceutical related over-domain ontology can be structured by defining its core concepts and relations.
It depends on the creation of dictionary of terms around which the network of relations will be defined. Since free databases about life sciences such as PubMed™ contains multiple publications about pharmacology, this ontology can be used as a connecting point for all ontologies that defines undelaying or overlaying aspects of the drug environment, for instance genome, chemistry or medical ones.
Actual publications focus on resolve specific scenarios such as creating ontologies for drug recommendations. Searching among PubMed™ articles showed that recent drug related ontologies are focused on connecting pharmaceuticals with therapy and healing processes. The work proposes by Doulaverakis, 2014, about a drug ontology in order to address the issue of drug-to-disease interaction discovery and by Farrish, 2013, to use a similar approach in order to coupling it with a complexity evaluator and determining complexity in polypharmacy prove this trend.
There exists one article written by Hanna, 2013, that proposes a drug ontology for RxNorm and other sources due to the need for reusing drug information from different existing databases.
Their effort abstracts RxNorm definitions in order to re-use them.

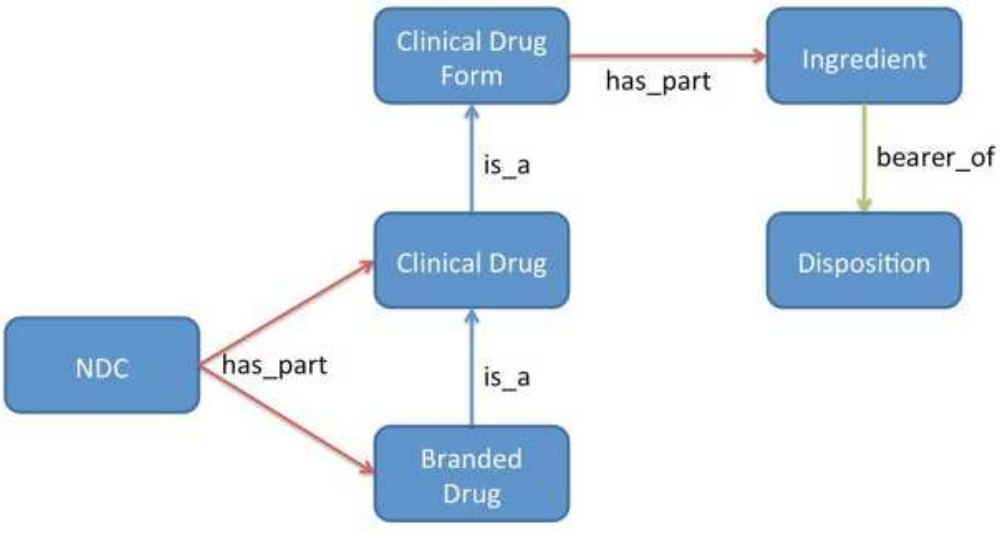


Image 17 – Rx-Norm drug ontology

By being an over-domain ontology, it is a model of objects that can be applied to a wide range of domain ontologies and defines a core glossary.
Basic Formal Ontology is a framework which applies this idea of different levels of granularity in defining ontologies (Smith, 2012).

A domain ontology requests to shape the defining terms of the pharmaceutical world and their relations with the objective of supporting actual data, for instance from a specific country, which can be queried and retrieved by systems and actors.
This means defining not only the concepts related to the objects but also the aspects related only to the scenario such as methods of identification and eventual usage guidelines.
It follows that it could be implemented a core ontology, which will be discussed in the next paragraph, to which this domain related one can bind by extending it and adding the needed adjustments for actually being used.
Biomedical ontologies are surging as recent years trend can show

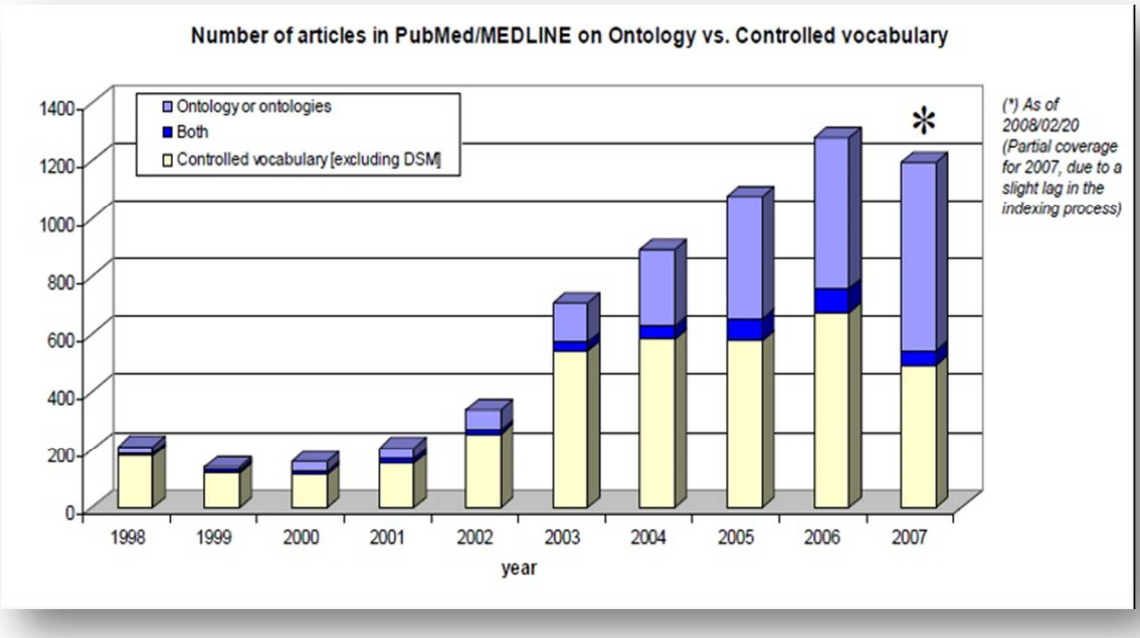


Image 18 – Number of articles about ontology in PubMed

even if most of them are gene research's related.

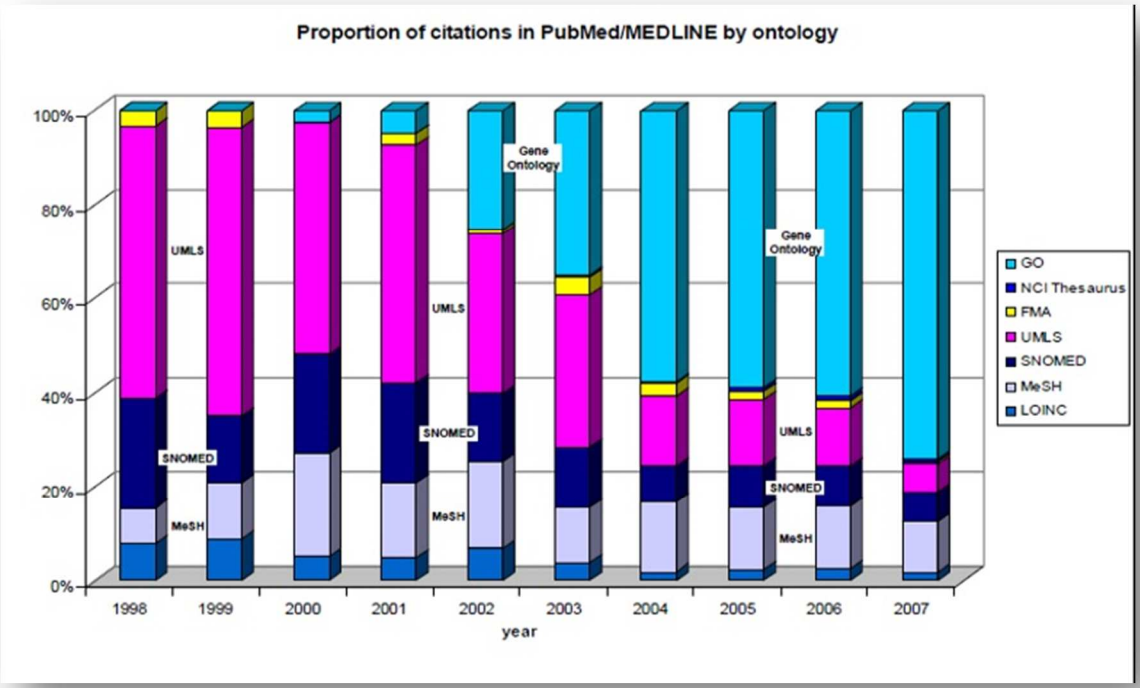


Image 19 – Citations of ontology in PubMed per argument

At a domain related level, some ontologies have been already developed, as showed by the example of RX-Norm for the United States of America.
On the contrary, in Italy this has not yet happened at least outside the academic environment.

2.4 wHospital Framework

The PharmaHosp project is part of a wider product named wHospital™. Developed by Laserbiomed™ S.r.l., this is intended as a framework composed by multiple modules whose aim is to be adaptable and customizable for every need of the customer.
It is actually installed among twelve healthcare providers in Lombardia

county and as part of the national healthcare program in Georgia.

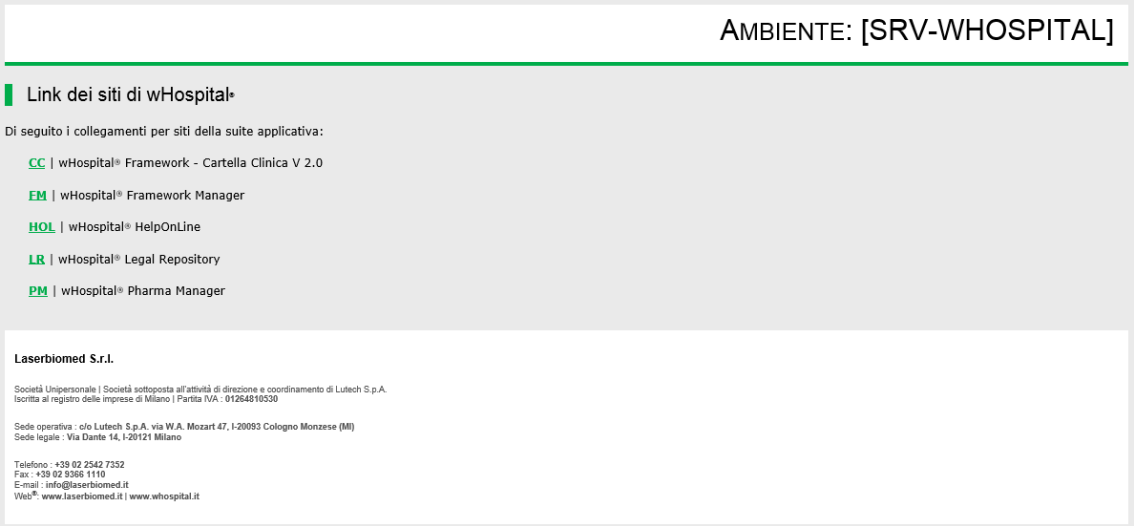


Image 20 – wHospital homepage

wHospital™ is composed by a core module which implements the definition of Health Care Record and manages all data about patients, doctors, nurses and operational units.

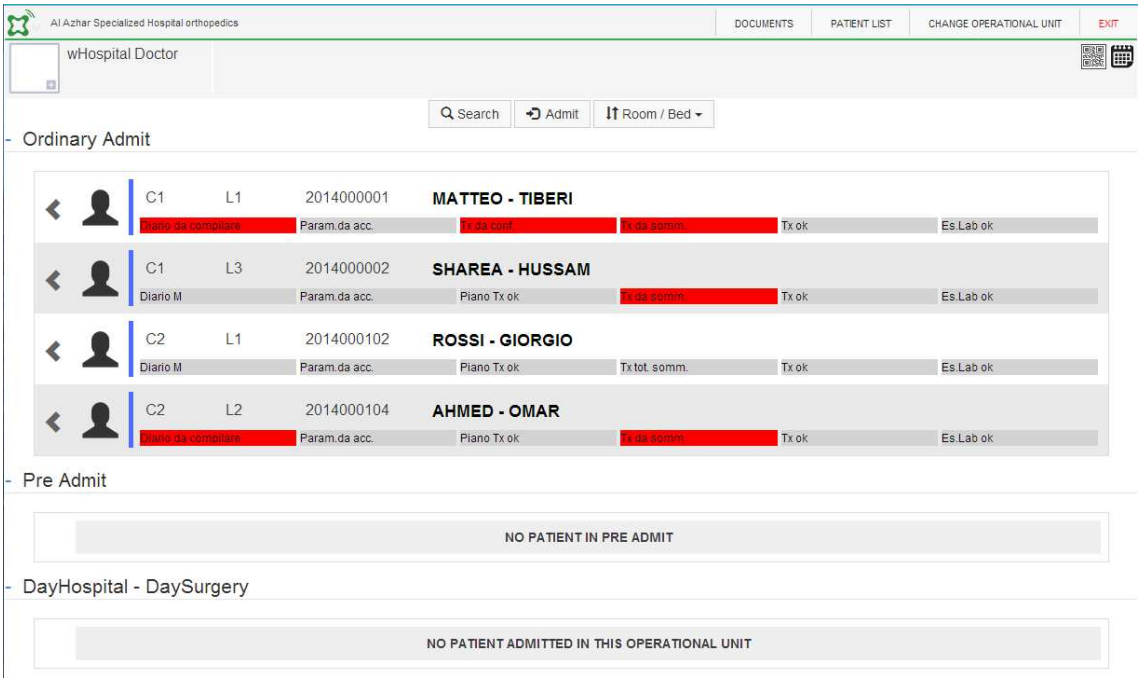


Image 21 – wHospital patient list

To this central pillar is possible to attach and configure a vast series of modules which covers the needs of the health care structure. For example, if a client needs only to cover two operational units one

with modules for the therapy and another one with the PharmaHosp ones, it's possible to install the requested features only for the part of the system that needs them and giving the right to use them only to determined roles.
This gives to the wHospital™ Framework the ability to be fully customizable and to level all installation and usage costs to the minimum of what is actually installed and required by the customer. In addition to that each module, by means of its configuration, can reproduce different behaviors giving an enormous elasticity to the whole product.

A couple of examples could be the possibility to size the investment for the servers to the actual pool of users and data needed. A structure with two operational units, one as a drugstore and the other as a therapy of some kind, will have a much lighter data load with respect to an entire hospital with multiple operational units with modules for therapy, administration, admission, drug managing and more. The reason for having modules that are customizable itself can be, for instance, the possibility for an operational unit to treat drugs by means of active ingredients, which gives the ability to prescribe equivalent drugs, but other operational units could have a so detailed list of drugs available that it would be no use to pass through the process of finding the equivalents and therefore it is possible to work directly by commercial names.

There exists three different types of modules in wHospital™: EHR, health record related, ones, horizontal ones and configuration ones. The firsts includes all the modules that works in a context where a patient with its relative health record is selected. For instance, modules about therapies or administration belongs to this typology since they

can work only when a patient is selected.

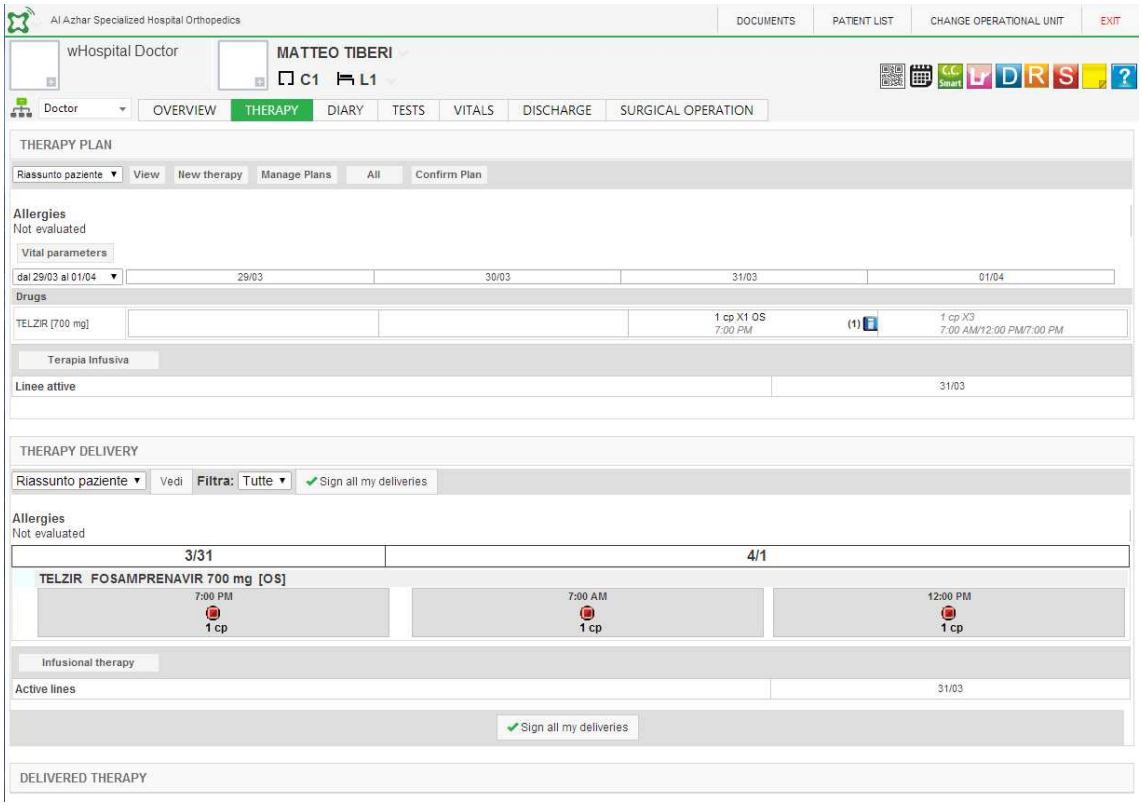


Image 22 – wHospital therapy module

Horizontal modules do not need a patient in their working session, they are generally about the operational unit in a more transversal meaning as it can be the organization of a ward and assignation of patients to rooms.

Configuration modules are technically not so different with respect to horizontal ones, but the operations they implements are of a nature focused on setting up the environment and configuring it. For instance, the definition of the types of therapy lines that will be used in the specific EHR module.

PharmaHosp is mainly a horizontal module, since it works on a managing level, with some configuration modules that helps setting it up.

In this scenario, PharmaHosp will become a main actor in organizing and managing drugs inside the wHospital™ in which it is installed. Almost every module has to work with drugs and lists of drugs available

inside an operational units. This specific module gives the possibility to set all this informations, correct or update them and dispatch all items inside the structure.
Therefore it represents a central aspect of the product in terms of reaching a complete control of the status of a health care structure with a detail close to the single item.
PharmaHosp will be adaptable with respect to the provider which will use it and will provide a different configuration for homecare providers named PharmaHospice mode and for healthcare ones named PharmaHospital mode.

3. OBJECTIVES

As today, healthcare providers across Italy face the issue of managing and tracing pharmaceutical products inside their structures. This is a consequence of a reform process that is planning to monitor all the handling process from the producer to the provider and, when exiting the structure, to the patient. This reform has not yet given precise directives about management and monitoring once products are inside the structure itself.
In this scenario, PharmaHosp main goal is to fill-in this existing gap in managing and tracing pharmaceutical products inside healthcare providers.

Image 23 – PharmaHosp objective

PharmaHosp, as a wHospital module, will grant a precise and reliable control over this processes, and three aspects are key to this achievement:

- Abstraction and querying of pharmaceutical products.
- Managing of all pharmacy operations.
- Providing documented exit patterns for products, used or thrown away, at each level of the process and more importantly when administration.

Abstraction will focus on a correct integration between the system and the data source, providing the list of all pharmaceutical products, by

implementing an independent structure, based upon the example set by international actors such as Rx-Norm, in order to achieve an acceptable level of independence with respect to the data source itself and enabling the possibility to change or adopt new sources, both in the Italian scenario and in a foreign context, without any need for re-factorization of the process.
Additional studies will concentrate on optimizing searches aiming to obtain the best compromise between reducing waiting time and increasing correctness of results.
Managing all pharmaceutical products' related operations will pose its attention mostly in implementing a tracing mechanism sufficiently adaptable to the customer's necessities but capable to lower the risk of human mistakes when handling and dispatching items.
The process must be faultless in order to have a secure monitoring and to enable the opportunity to create reports capable of highlighting potential areas of waste or mismanagement.
The integration of PharmaHosp with the wHospital Framework and its modules will tie together the e-prescription world and the logistic one. By allowing data access from other points of the system, it will be possible to consult pharmaceutical products in the database and to obtain availability of items inside a storage point in an operational unit.
The process in its wholeness must cover the item route from its arrival and acceptance inside the structure until it is administered or used to treat a patient. In doing so, it is crucial to never loose at any time location and movement's history, the people who handled it and the patients who received it.
PharmaHosp latest objective will be configurability: the module will provide different behavioral opportunities depending on the structure that will install it.
One configuration will be focused on home care treating and therefore will treat and save data tying together users and items, while in the

second mode this constraint will loosen providing a more free management for operational units of a healthcare provider.

4. METHODS

This chapter will focus on the technologies and methodologies adopted in this study: from those which determined the process modeling and design to those related to behavioral choices as the user interface. The tasks addressed spread from modeling the environment surrounding and defining the project, structuring a data representation suitable for our endeavors and proposing a way to present this information to the final customer in a way as user friendly and intelligible as possible.

4.1 Process Modeling

Modeling the process was realized by using UML, Unified Modeling Language. This approach has its strong point in precision of definition and understandability of the implementation even for interlocutors not expert in the information technology language.
This grants a layer of common ground which enables discussion of the process and of the explication of the client requests without being too abstract with respect to the actual developing approach.
UML combines techniques from multiple data and logical shaping processes such as data modeling and their entity relationships diagrams, work flows from business modeling, object modeling from class modeling used in information technology and component modeling.
It is suitable at each stage of the developing life cycle since it poses itself as a map of what has been done, a documentation of the product, is been doing, a proof of the work realized so far, and will be done, a common agreement of the project undertaking.
Above this, by being not a language per se and being more abstract than an actual implementation design, it can be used across different

implementation technologies. (Larman, 2005)
The graphs and diagrams implemented include:

- Activities
- Actors
- Business processes
- Database schemas
- Logical components
- Programming language statements
- Reusable software components

It keeps strong relationships with the software development processes since it is born by synthetizing notations of the Booch method, Object-modeling technique (OMT) and Object-oriented software engineering (OOSE) and fusing them into a common language widely usable. The Booch method is an object modeling language developed by Grady Booch. He is considered one of the fathers of graphical representation of models and has defined some of the notations adopted by UML.

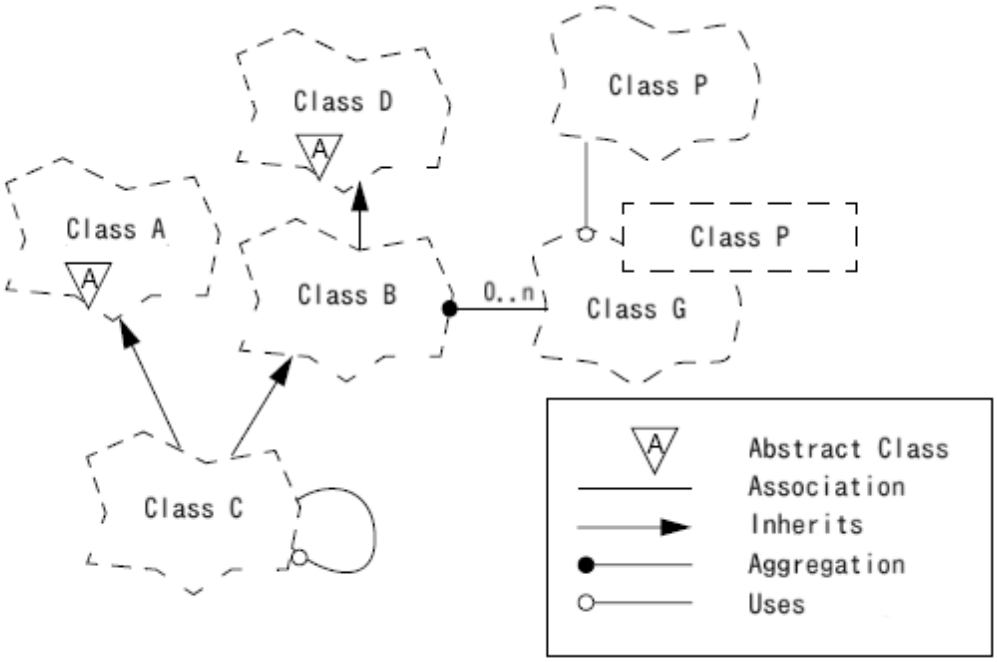


Image 24 – Booch notation

It works both for concurrent and distributed systems. Hence, it has not limits in representation of modern processes and communications' scenarios.

A powerful tool in helping actual development is the possibility to transform automatically UML into other representations and programming languages such as Java™. Nevertheless, UML is not a development method. Using UML considers two different approaches: one that defines the model, with its related written documentation, and another which realizes the set of diagrams of a system, a partial representation of the dynamics of the model in a graphical way. These two approaches defines two separate views:

- Static or structural: focused on object, attributes, relationships and operations.
- Dynamic or behavioral: focused on showing collaboration among objects and changes in their internal states.

The first one are implemented mainly by class diagrams and use cases. The latter by sequence, activity and state machine diagrams. The latest UML implementation, version 2.2, defines fourteen types of diagrams divided in the two categories specified in the previous paragraph.

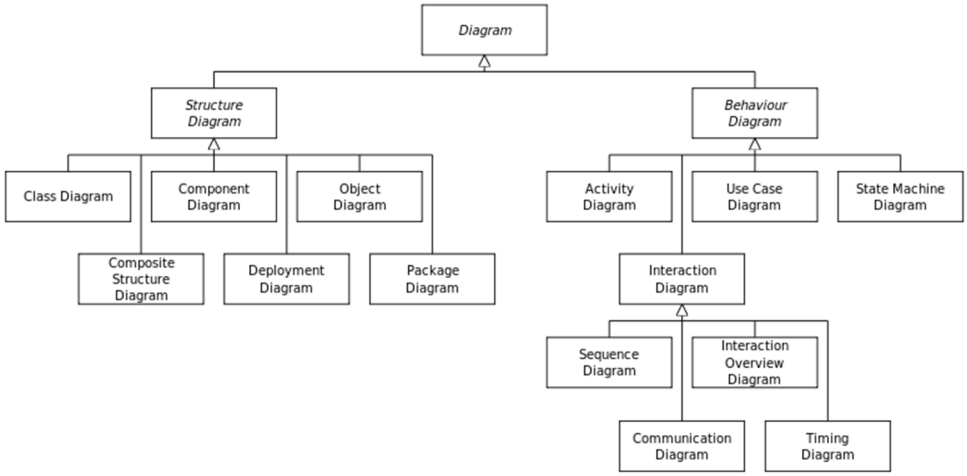
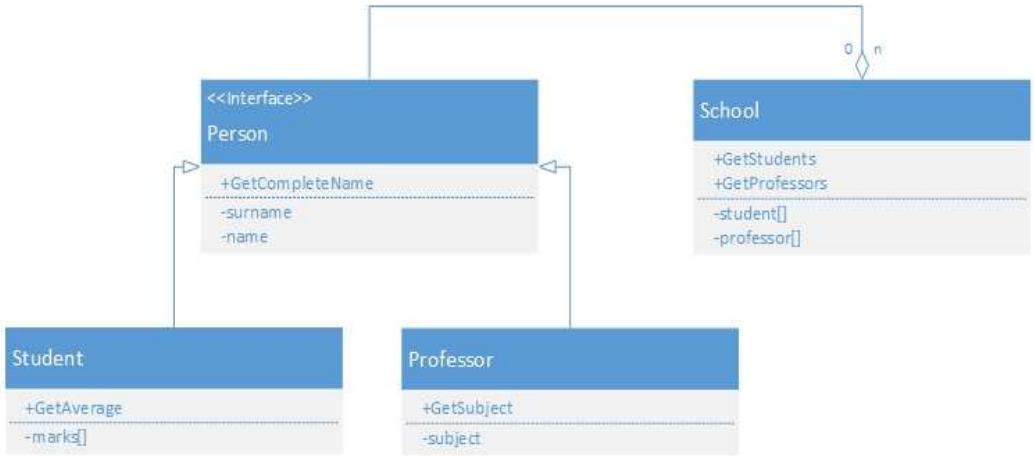


Image 25 – UML Diagrams structure

Structure diagrams emphasize objects and relations of the modeling system. They produce part of the documentation typical for software architecture.

In the following, the static diagrams presented will be:

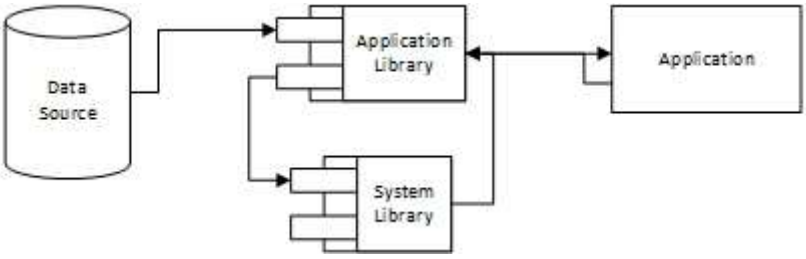
- Class diagrams: structural representation of the actual classes with attributes, relationships and methods.
  It defines also initialization values, scope and visibility.



- *Image 26 – UML class diagram*

  By means of relationship links, it is possible to associate, aggregate, compose entities, and for each one specify its multiplicity.

- Component diagrams: describes how components are wired together in order to realize the system.

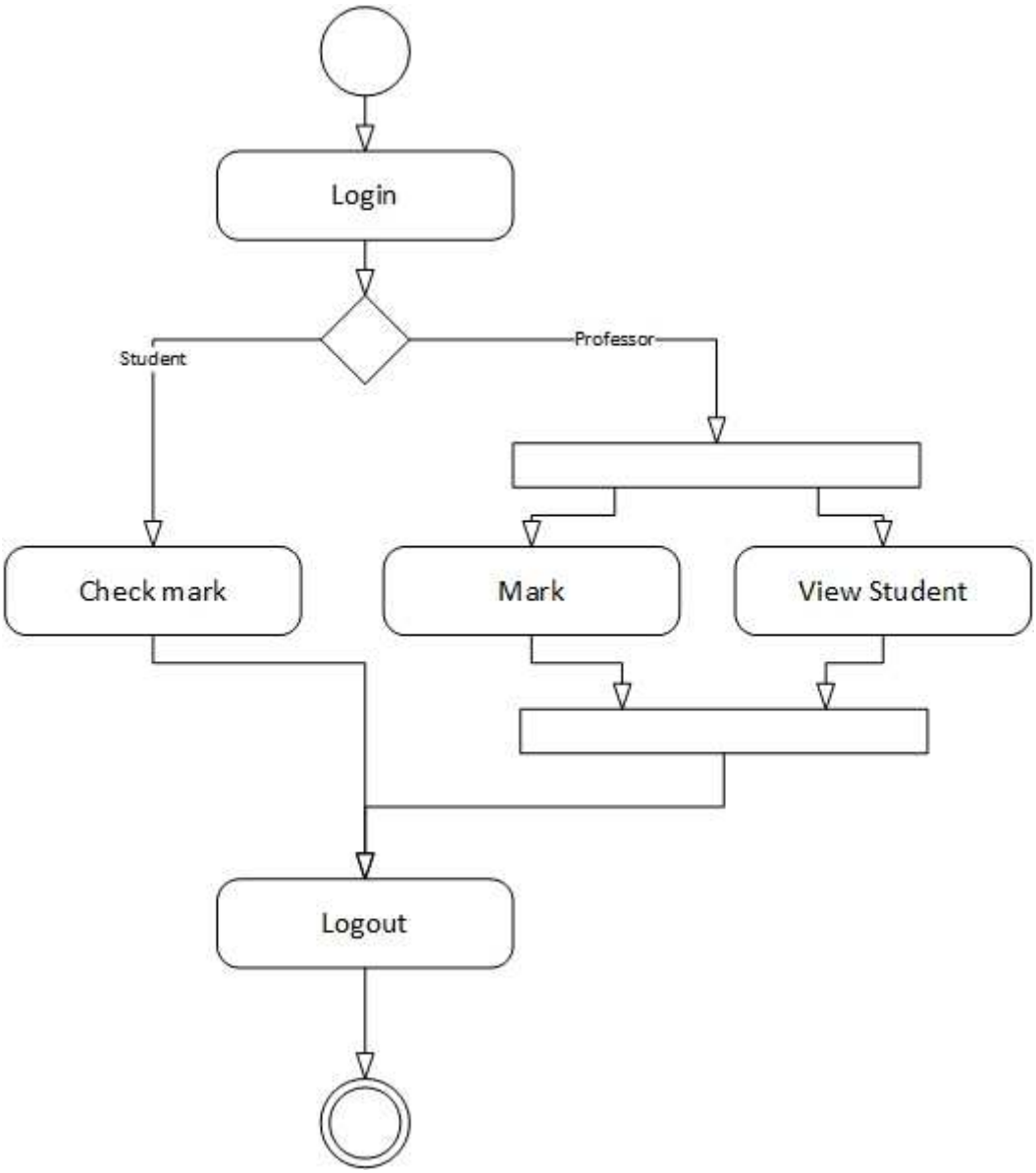


- *Image 27 – UML Component diagram*

  A component is a needed portion of code required to execute a function. The connection is realized by interface of the first one through the provided one of the second and more in general it is based on the paradigm that a component implements a functionality required from another one in order to work.

The behavioral ones will be:

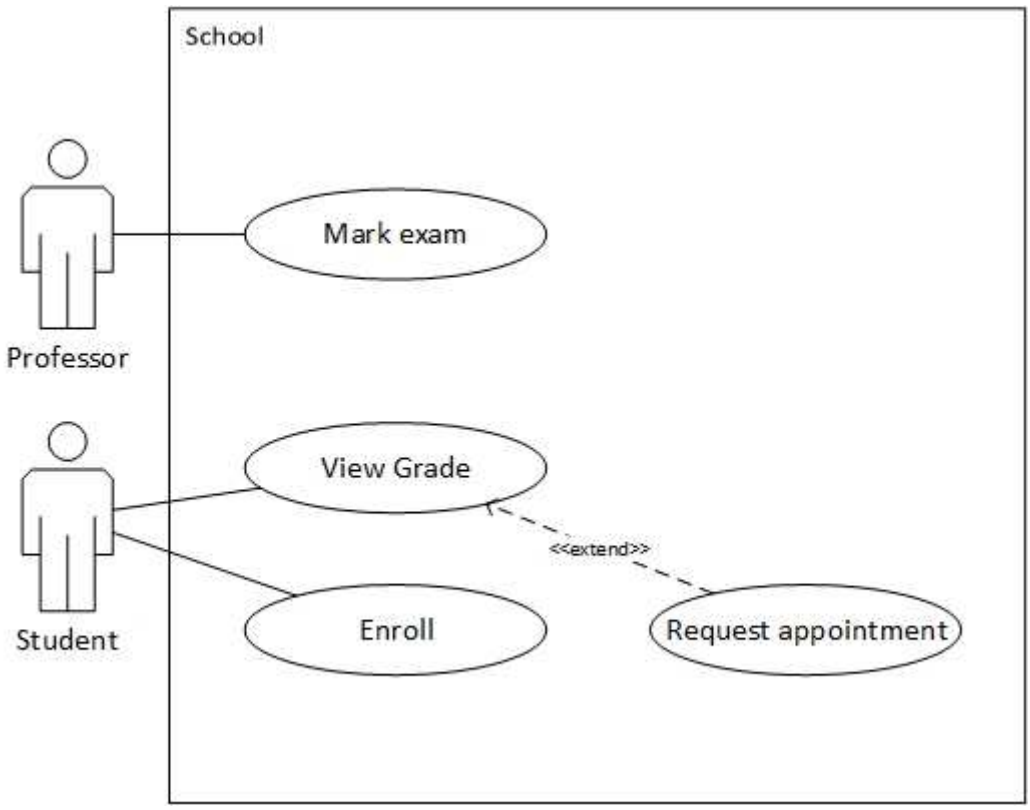
- Activity diagrams: emphasize the business and operational workflow of a system.



- *Image 28 – UML Activity diagram*

They support explication of choices, iterations and concurrencies and hence they can be both computational and organizational processes. Rounded rectangles represent actions, diamonds represent decisions, bars start and end of concurrent activities, circles the start state and the final state of the activity.

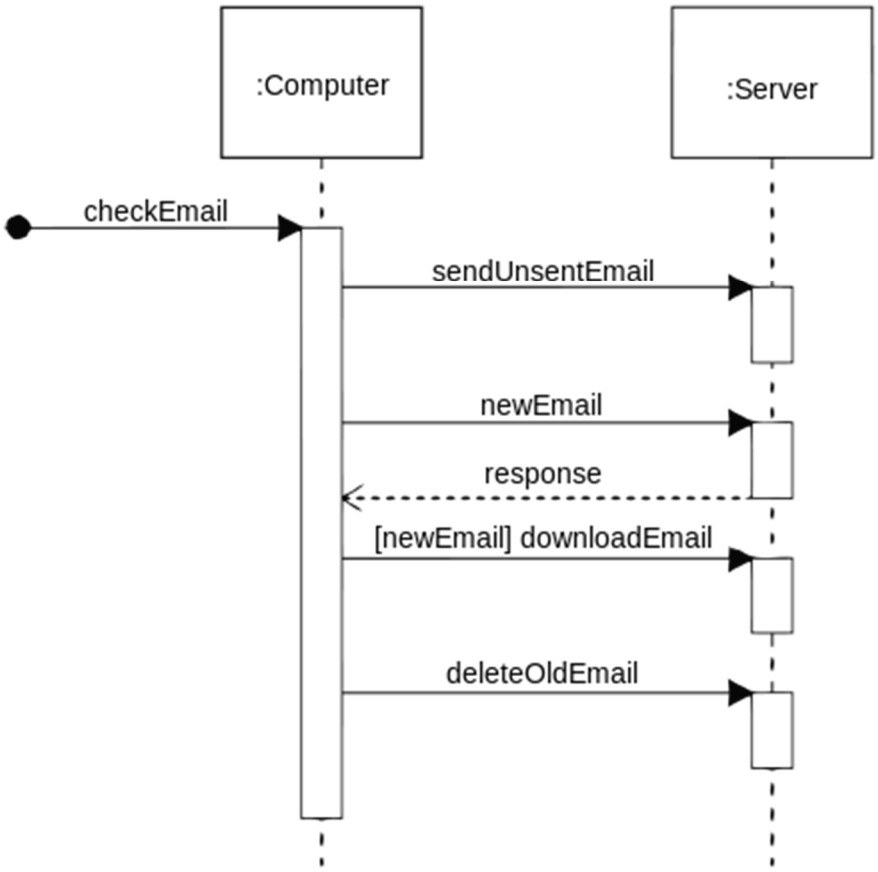
- Use case diagrams: describe the functionality of a system in terms of actors and goals.



- *Image 29 – UML Use Case diagram*

It is usually combined with a textual narration of the scenario and it is used in order to explain actions available to users in the system.

- Sequence diagrams: explicates the sequence of messages between objects or processes, and their lifespan.



- *Image 30 – UML Sequence diagram*

It shows as parallel vertical lines different processes or live objects and as horizontal arrows the exchange of messages happening between them providing a graphical representation of runtime scenarios.

Those examples are a reinterpretation of the work by Fakhroutdinov, 2009 in explaining UML diagrams.

4.2 Drug Abstraction

4.2.1 Data Base

The database structure is designed and shaped using an entity relationship model.
ER models are used in order to describe data and information of business domains, or eventually its process requirements using an abstract representation that can be ultimately be implemented in a relational database.
The main components of these models are entities and the relationships that exists in between them.
There exists three layers of models that can be developed:

- Conceptual data model: it possess the least granular level and establishes the overall scope of the environment to be represented. It is usually generated during documentation of data architecture.

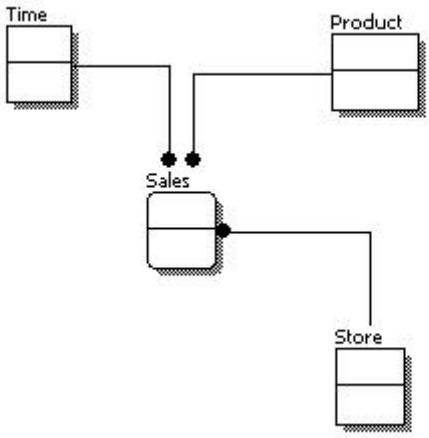


Image 31 – ER conceptual model

- Logical data model: does not require the conceptual part since it defines the development of distinct information system. It contains more details with respect to a conceptual model by establishing both the details of each entity and the relationships between them. It remains independent from the technology

which will implement it.

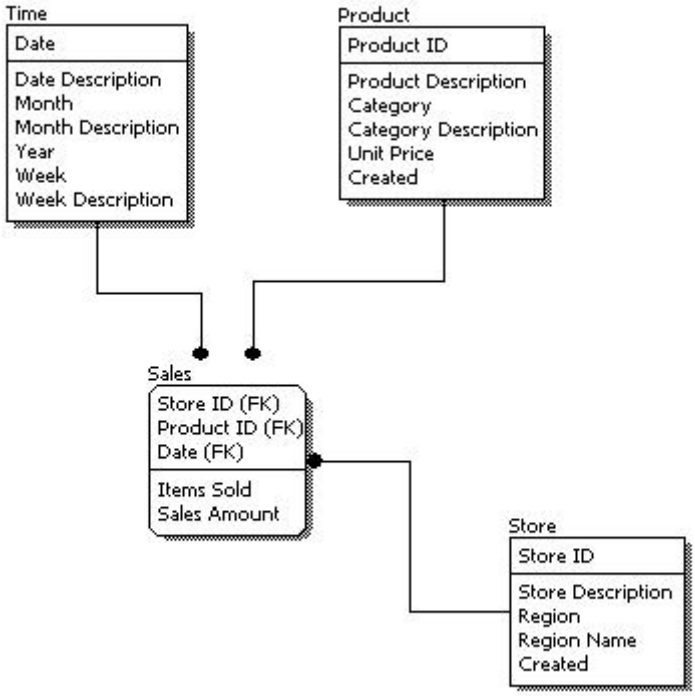


Image 32 – ER logical model

- Physical data model: it is typically the actual instantiation of a database and therefore it is not unique for a logical model but represents one of the possibilities. It follows that it is technology dependent.

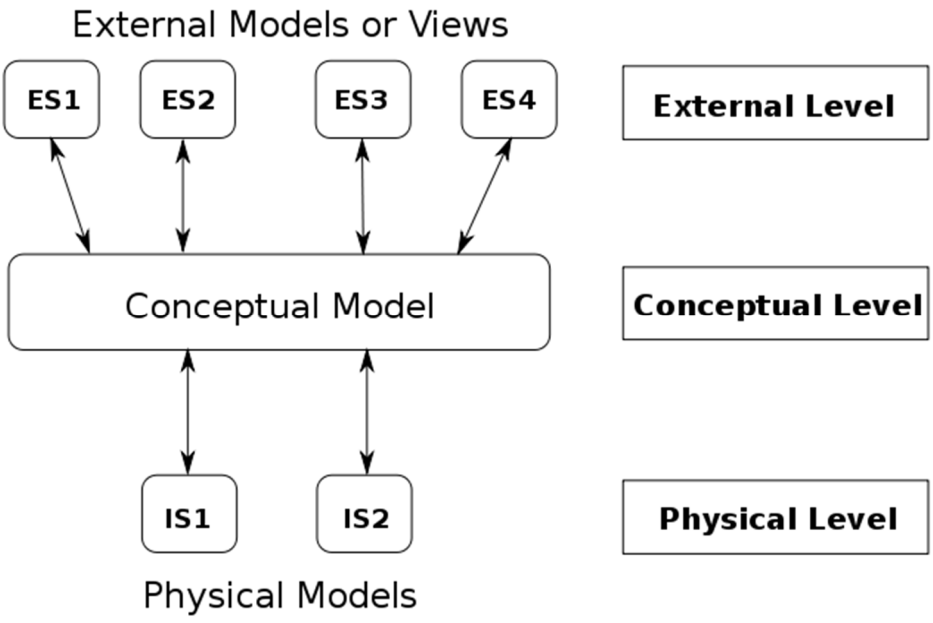


Image 33 – ER layers

The first use of this representation is normally during requirements analysis in order to describe the information needed and eventually their type on a conceptual level.
It can also be used to describe any ontology as an overview and classification of used terms and their relationships for a specific environment.
Design phase is the following step in which ER diagrams are used. In this case it is usually a logical representation which will progress into the actual physical implementation of the database and will realize the last model.
The definition of entity refers to an object which is capable of existing independently and can be uniquely identified, it has to be an abstraction from the complexities of the domain.
Relationships capture the interconnection and interaction aspects of the abstraction, meaning that they represents actions or links between entities. A typical sample of this base objects of database modeling can be nouns and verbs used in any language.
Once modeled the conceptual representation, it is necessary a language which can express these concepts: Structured Query Language, known as SQL, is the language at the base of database development. It is a programming language with a special purpose in mind which is managing data held in relational database management systems, or RDBMS.
Based upon relational algebra and tuple calculus, its scope is data inserting, querying, updating and deleting, schema creation and modification and data access control. In addition it includes procedural elements that extends its possibilities.
SQL is divided into several language elements, including:

- Clauses: constituent components of statements and queries.
- Expressions: producing either scalar or table results.

- Predicates: using a three valued logic or Boolean logic are used in order to specify conditions.
- Queries: retrieving data based on specific criteria
- Statements: which may have persistent effects on data or schema and can control transactions, flows, connections, sessions and diagnostics.

The powerfulness of this language is the ability to produce not only the data stored in tables but also to complete it with valued columns and manipulating it in order to obtain directly the type of data needed for the specific scenario. This allows a more explicit decoupling between data layers and business layers: at this level the data is shaped in the desired form and in the above one is changed, updated or simply viewed without the need to introduce more complexity.

4.2.1 The pharmaceutical ontology

The pharmaceutical environment can be shaped as a two-layer ontology since it has a strong direct interconnection with the country laws and rules.
The first layer would define the world itself on a generic level implementing a core ontology that provides interoperability between every pharmaceutical database throughout the world.
The second layer would be localized for the specific reality that needs it by adding those additional concepts and relations that would not change the nature of the item at the previous layer but would only complete it and refine its understanding in the specific context. Hence it would be an ontology with items implemented ad-hoc. For instance, logical structure of usage guidelines could be different from one country to the other, an ontology would allow not only to define the two different constructs but also to propose an interpretation, or translation, that would allow understanding even foreign policies and

directly show the possible conversion in the scenario at hand. This does not mean to subvert the process of rules provided by national pharmaceutical associations but only to provide an international standard that can lead through differentiated protocols to the definition of an interpretation of a concept in the country that it is actually studying it.

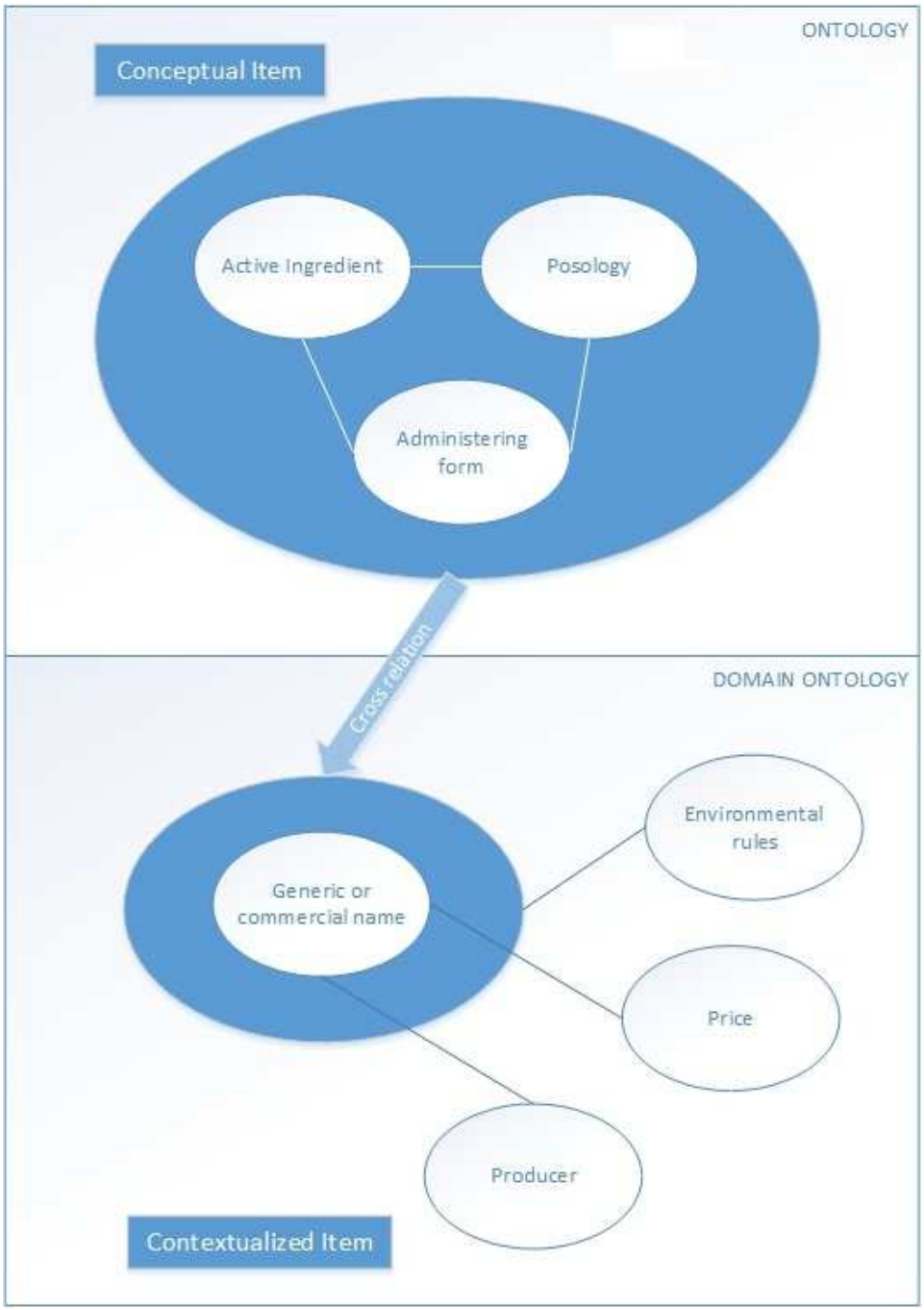


Image 34 – Sketch of drug domain ontology

A conceptual sketch of the implementation would be the core definition connected with domain or national ones refining and completing it. This structure would be updatable in its over-domain definitions as human progress advances and the domain ones as national rules and commissions adopt new policies. In addition, domain ontologies could intercommunicate in order to complete or consult each other definitions. It could be a direct interfacing if structures are similar or with the above layer acting as translator.

4.3 Implementation Approach

4.3.1 User Interface

Designing the interface of the PharmaHosp suite required understanding the way pharmacists, nurses and in general future users of the system proceed during their everyday routines. Providing an interface as intuitive as possible helps in dealing one of the first obstacle that companies encounters while digitalizing their processes. Passing from a methodology based on paper sheets and oral communications to one that manages everything via software messages and representations is a change often underestimated and unwanted by workers since it requires an initial learning process. The Microsoft Silverlight™ technology used in the development of the suite was essential in order to achieve complex behaviors with a rather intuitive and fast to learn interface.
With respect to other context where the focus can be more on content delivering rather than easiness of usage, in this scenario is preferable to create an interface that reminds of something already seen such as the common Windows™ interface. Since there is no need to design something extremely innovative, the main objective was to replicate

patterns using the same aesthetic and with the same functionalities throughout the solution and to represent data using grid views as similar as possible to the typical cells of electronic sheet programs. Microsoft Silverlight™ fulfills abundantly this requirement since it provides easy tools to create forms, display complex data grids, templates for search fields with autocomplete behaviors and contextual menus.

This is possible thanks to the easiness in data binding items to graphic controls.

The basic behavior required is displaying, interacting, updating and inserting informations.

4.3.2 Software Development Cycle

Following the typical cycle of modern software, composed by requirement analysis, design, implementation, testing and evolution (Ghezzi, 2003), the main aspects of PharmaHosp design and implementation will be taken into account and analyzed in order to motivate adequately choices and eventual compromises.

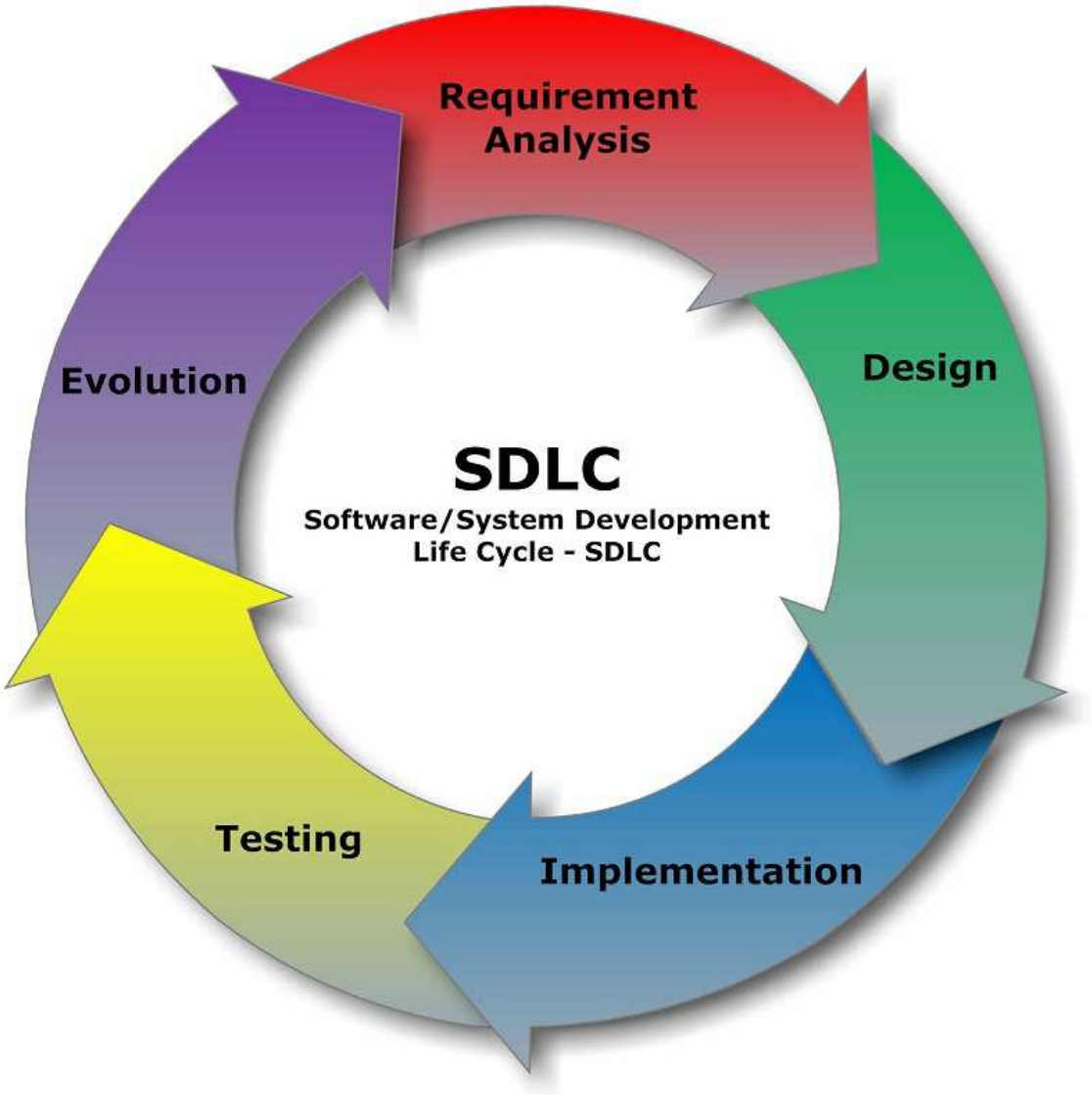


Image 35 – Software Development Life Cycle

Firstly will be taken into consideration how the actual model representing the world, which surrounds pharmaceutical products, has been realized.
Then, there will be a requirements analysis based on what where the expectations arisen from the study of real scenarios such as the hospital environment of Casa di Cura Privata del Policlinico di Milano and of Fondazione Maddalena Grassi homecare structure. These considerations will be merged into a feasibility study in order to propose a solution both cost and effort effective.

Once this phase is discussed, the actual design phase of the software will begin, giving birth to how the application will be actually realized at least at pseudo-code level. This will lead to the development phase that will implicate choosing the proper software to use and the analysis of certain portions of code in order to understand mechanics and performances. Bearing in mind that both time constraints with respect to the customer and previous technology related choices from the developers of wHospital were both opportunities and limitations to the implementation choices.
Testing and then deploy will follow with a brief analysis of the first reactions to actual usage.

4.3.3 Testing

The main reason for testing software is to analyze critical points and use cases in order to avoid issues, especially blocking ones, during usage of the product.
Ideally testing will stress the system in all possible ways, but since it is impossible there exists multiple criteria to judge precision of testing. A test case represent a single situation under which the system might run and consists of entry data, a procedure and a certain expected result.
While developing, four main testing phases take place:

- Unit testing: verifies the consistency of an isolated piece of code or functionality.
- Integration testing: verifies if two intercommunicating portions actually work together.
- System testing: verifies if all components together work.
- Alpha acceptance testing: by defining in-house users it tries all functionality in the same way the customer should do once provided with the product.

After the developing phase, two more tests criteria are used:

- Installation testing: easiness of installation and proper working of the system in the new environment
- Beta acceptance testing: involves real users trying out the product.
  (Marick, 2011)

During requirements analysis is usually redacted a Test Plan document that states all types and phases of testing that will be covered. It includes test objectives, schedules, strategies, test cases and procedures for handling problems.
Two main techniques are used in testing:

- White box: based on code, tests how a program does a feature.
- Black box: based on couples of input and output, tests the behavior of a program.

System testing focuses on evaluate the system as a whole. Once asserted an acceptable degree of correctness of the product, it passes onto stress testing, by using larger than normal parameters in terms of data transactions or users, then execution testing providing performance values, and recovery testing in order to avoid losses upon disasters.
In the healthcare environment, software usually sustains also compliancy and security tests in order to adhere to standards and requirements.

5. RESULTS

The following section will focus on the actual choices that lead to the implementation of PharmaHosp.

5.1 Meta-model

Modelling the world of pharmaceutical products arise the need to acknowledge two separate environments that will continuously communicate in order to properly work.
The starting point is to take into consideration how to actually model a drug which is not a plain object but is characterized by a number of heterogeneous information.
The source of information used in order to realize the model was specific for Italy but the effort was focused in trying to achieve the more abstraction possible, obtaining an international base structure of information with additional properties that would possible to use in actual structures that works with national laws and security standards.
Above this base structure there is the need to model the world of a hospital pharmacy that has to dialog with operational units by means of storing units, requesting actors, patients and flows of items.

5.1.1 Pharmaceutical Products

Pharmaceutical products needs a series of properties in order to be successfully defined, whatever the country in which they are put into commerce. In this context the admission into commerce code is obviously a unique identification property of an item.
There are eight attributes that defines properly a drug:

Image 36 – Drug related attributes

- Its type, meaning to what pharmaceutical category the product belongs. For instance, if it is a drug or a medical device, if it has to be considered a para-pharmaceutical or for veterinary use only.

	DrugTypeID	DrugTypeDesc	DrugTypeShortDesc
1	1	OMEOPATICO USO UMANO	OMEOPATICO USO UMANO
2	2	OMEOPATICO USO VETERINARIO	OMEOPATICO USO VETERINARIO
3	3	PRESIDIO MEDICO CHIRURGICO	PRESIDIO MEDICO CHIRURGICO
4	4	FARMACO DA BANCO	FARMACO DA BANCO
5	5	FARMACO ETICO	FARMACO ETICO
6	6	FARMACO OSPEDALIERO ESITABILE IN FARMACIA	FARMACO OSPED. ESITABILE
7	7	FARMACO GENERICO	FARMACO GENERICO
8	8	FARMACO SOLO USO OSPEDALIERO	FARMACO SOLO USO OSPEDALIERO
9	9	PARAFARMACO ERBORISTICO	PARAFARMACO ERBORISTICO
10	10	PREPARAZIONE MAGISTRALE	PREPARAZIONE MAGISTRALE
11	11	CODICE DI AGGANCIO	CODICE DI AGGANCIO
12	12	MEDICINALE VETERINARIO PREFABBRICATO	MEDICINALE VETER.PREFABBRICATO
13	13	FARMACO ODONTOIATRICO	FARMACO ODONTOIATRICO
14	14	PARAFARMACO USO UMANO	PARAFARMACO USO UMANO
15	15	SERVIZI PRESSO L'ESERCIZIO (COMPRESE PRENOTAZIO...	SERVIZI
16	16	SOSTANZA PRECONFEZIONATA	SOSTANZA PRECONFEZIONATA
17	17	PARAFARMACO TRATTATO ANCHE DALLE SANITARIE	PARAFARMACO SANITARIO
18	18	MATERIA PRIMA	MATERIA PRIMA
19	19	FARMACO VETERINARIO	FARMACO VETERINARIO
20	20	PARAFARMACO USO VETERINARIO	PARAFARMACO USO VETERINARIO
21	21	DISPOSITIVO MEDICO IMPIEGATO ANCHE IN AMBITO EXT...	DISPOSITIVO MEDICO
22	22	PREMISCELA MEDICATA VETERINARIA	PREMISCELA MEDICATA VETERIN.
23	23	DISPOSITIVO MEDICO IMPIEGATO PREVALENTEMENTE I...	DISPOSITIVO MEDICO OSPEDALIERO

Image 37 – Defined types for pharmaceutical products

- Its pharmaceutical shape: it distinguish it by means of capsules or syringes and more.
- Its package: whether it comes as blisters or in other ways.
- Its route of administration: such as oral, injections, etc.
- Its storing attributes: this is taken into account since it unites storage temperatures and expiring periods, it can be a complex structured field composed by days, months, years and temperature conditions.
- Its ATC: the Anatomical Therapeutic Chemical classification system is an international standard that defines through multiple layers of codes the therapeutic and chemical characteristics of the product. It is meaningful when grouping and abstracting drugs. It is composed by seven levels with each one defining a sub-group. For instance the five level code: C-03-C-A-01 defines the active ingredient Furosemide with the following groups respectively to the code: Cardiovascular system, Diuretics, High-ceiling diuretics, Sulfonamides, Furosemide.
- Its producers: this is not really an international aspect, since it takes into account multiple producers and dealers. But especially for medical devices it really defines the product searched and consequentially becomes a fundamental attribute for the modeling.
- Its active ingredients: obviously this applies only to drugs and para-pharmaceutical products. But similarly to producers and medical devices, it really defines a product by explicating what it is used for, in which quantity and also how it is diluted by its excipient.

  All these attributes are related to the central entity which is the drug itself. The properties that enrich it are more detailed and tailored to the reality of the actual source of data from which it is populated, in fact there are fields related to its commerce possibilities and prices, its code for identification, brief summary

descriptions which are needed since not all products possess all the previous attributes. A mandatory attribute is the type since it defines the meaning and way of interpreting all the other ones. This entails that by means of these eight main attributes it is possible to search whatever product on any data source properly connected.

It also introduces the concept of virtualization of a drug. In fact, an actual case scenario is the one where doctors or nurses reason in terms of posology and active ingredients. While prescribing there is no need to know the exact product necessary, an equivalent will suffice, but when it comes to the administration of the product it is inevitable to assign to this more abstract concept a precise object identified with a certain drug.

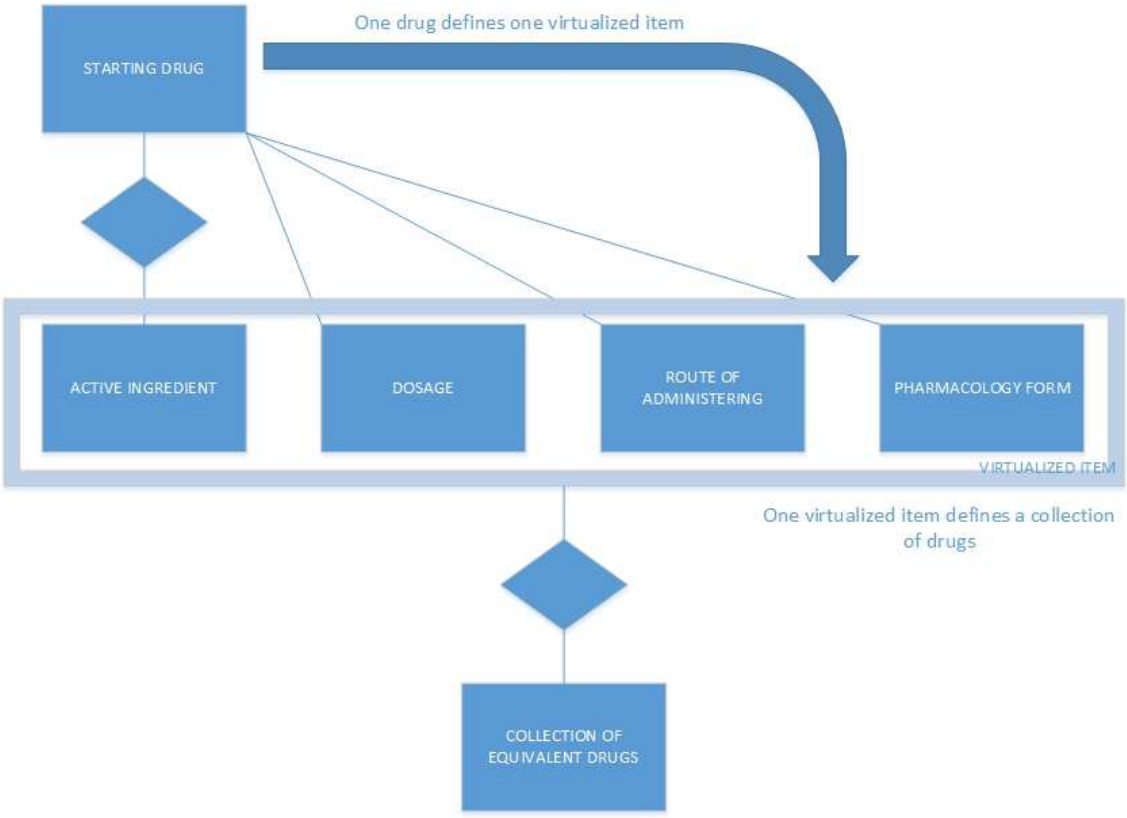


Image 38 – Drug virtualization process

It is an efficient way of grouping certain products by means of type, posology, active ingredients, route of administration and eventually shape in order to make the choice when prescribing clear but not so

strict that the pharmacy, or the nurse handling the administration, has to understand and then decide whether it is possible to use the specified item or if an equivalent will be the same.

5.1.2 Storage Environment

The fundamental entities defining the storage environment are the operational units. These are basically the recipients that collects products moving through the health care structure. By being a core point of the storing procedures they are consequentially a valuable asset that administrations needs to record.

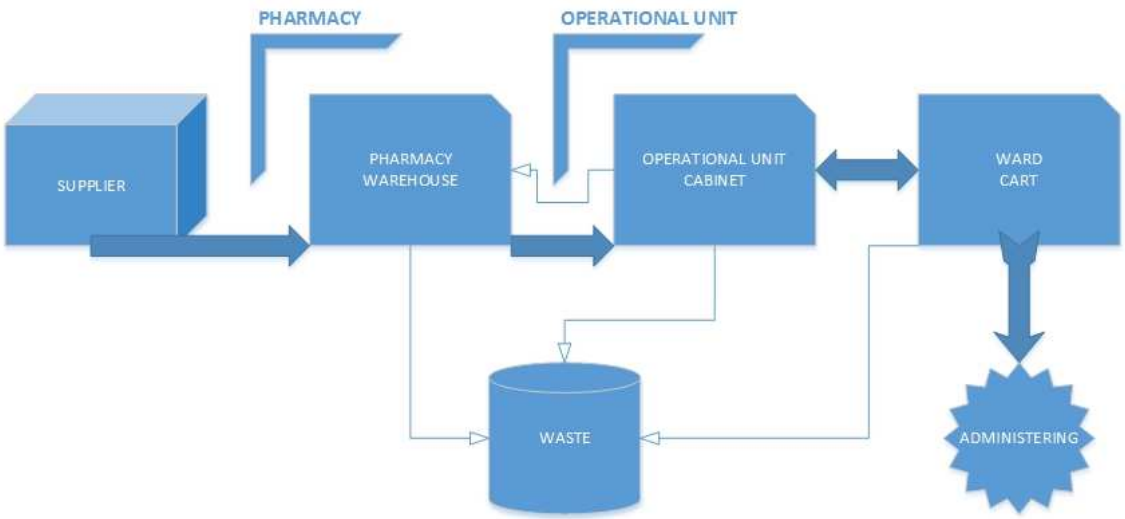


Image 39 – Items flow throughout healthcare structure

Operational units are internally split in sub categories such as cabinets and carts which represent internal dispatching points needed to accomplish the typical flow of a drug from the pharmacy to the patient: the product firstly enters the operational unit with the role of manager of drugs, then it dispatches items to another unit that requests it. As it receives them, products usually are stored temporarily in a cabinet and once prescribed are moved onto a cart from which they will be brought to the patient in order to be administered.

This ideal process must face exceptions, allowing the flow to be reversed at any point for any reason. Thus items will be taken back to the pharmacy where they will restart the process.
The homecare assistance scenario is partially different since it considers operators with their own personal bags and packs of products that are directly tied to the patient home. Anyway they can be considered quite similar once we assert that operators are cabinets and the packs left to the patient are carts. One main difference in this scenario is that accountability is done not at the operational unit level but at the operator one in order to maintain control on what everyone is managing and in what quantity. The role played by users is significantly more central than in the hospital scenario.
This configuration can however happen in hospitals too, since a structure could have a single operational unit on multiple levels and therefore requested to control each level separately in order to maintain both monitoring and accountability.

5.2 Implementation

5.2.1 Requirement Analysis

The study of the requirements was conducted following the instructions given by Casa di Cura del Policlinico di Milano for the PharmaHospital healthcare scenario and by Fondazione Maddalena Grassi for the PharmaHospice homecare scenario.
Their request stated the possibility to manage all drugs' related information: stocking inside the pharmacy warehouse, dispatching and horizontal movements between cabinets and cart of an operational unit and, to conclude the flow, administration to the patient.
This meant the following macro-operations were needed:

- Management of drug information, updating and eventual manual corrections.
- Creation of the pharmacy operational unit with capability to admit, order and dispatch products.
- Possibility to request items and move them through the multiple storing points inside the operational unit.
- Integration with the already existing software for therapies in order to administer products effectively stored in the carts that reaches the patients' beds.

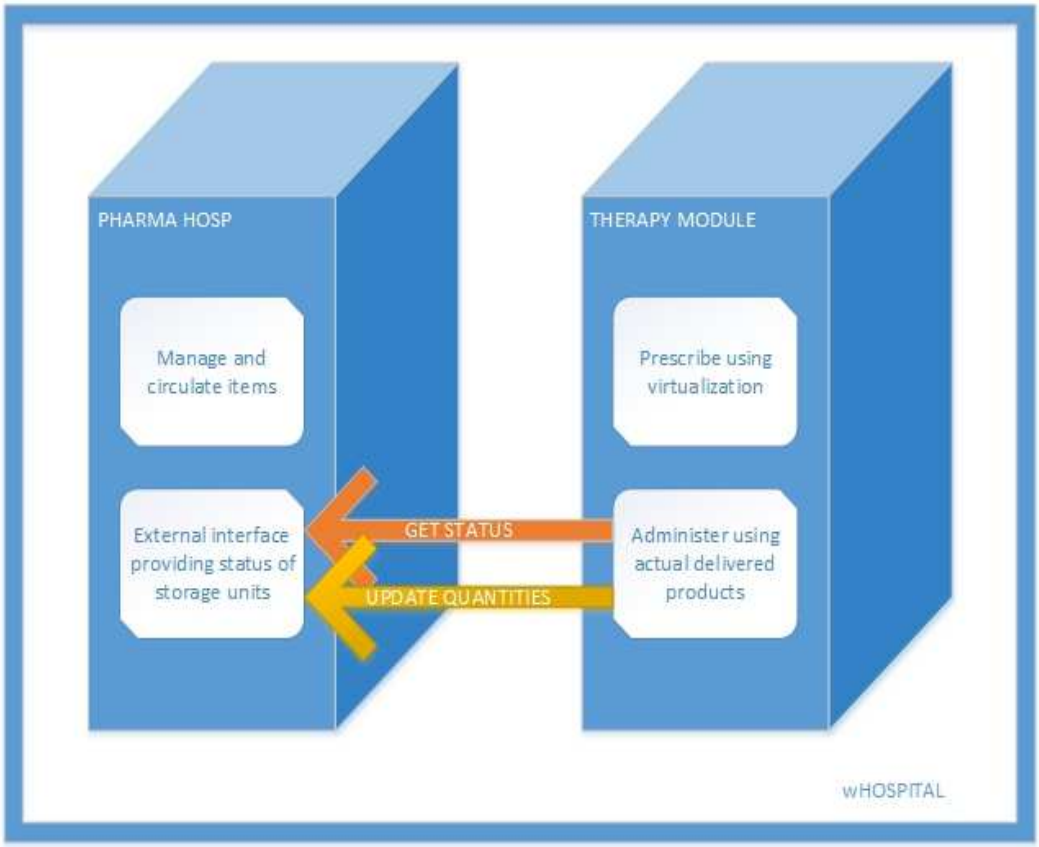


Image 40 – PharmaHosp and therapy module integration

Deeply analyzing these use cases lead to some more precise requests about how information on drugs is treated:

- It must be possible to create custom drugs, both for experimental therapies and for addressing eventual voids in the data source. This entails the possibility to create and add active ingredients in

order to construct a complete posology of a drug, allowing the process of virtualization of a drug in which the medic can prescribe therapies by properties and dosage of an active ingredient and the nurse can administrate the correct product of whatever brand.

- Products must be updatable both through the data source monthly releases and manually in order to complete eventual partial information and to insert certain unique codes that are able to identify unequivocally products inside the healthcare facility. For instance, while AIC codes are always given, some EAN codes and REF codes are not reported from the data source but are necessary when identifying certain categories of items.
- The creation of prontuaries, which are lists of pharmaceutical products, grants the user the ability to select only products that are commonly between the ones ordered by the pharmacy avoiding requesting unavailable or misspelled products.

About the flow of the drugstore:

- It must be possible a bulk load of the warehouse. The main process for insertion of products must use the paradigm of orders followed by "documents of reception", commonly known as D.D.T. But since certain orders can be made asynchronously with respect to the system, for instance due to time constraints in obtaining certain items, it must be possible to accept products unrelated to any order.
- The possibility for each product to define a minimum quantity level under which it is triggered an alert that notifies the user the need for reordering. This value can be coupled with a suggested amount to request and the minimum multiple of items that can be ordered. For instance, some items are not sold in single pieces but in groups of six, eight, ten and more. This introduces third level of encapsulation of products: single pills, or posology units,

boxes, made of a determined number of units, and groups of boxes which defines the minimum multiple for orders. In addition, it must be possible to reason inside the system both as boxes and units by means of configuration, setting a flag for the item, or transformation, converting a boxed item into an unboxed one.

- Trustworthy users, as supervisors, must be granted the ability to correct values in store and eventually delete items.
- A part from working by meaning of requests from operational units, the warehouse manager must be able to see the state of the cabinets of each operational unit due to managing purposes, as monitoring use or actively send restocks.
- A validation process, at least for certain products, for requests coming from an operational unit must be introduced. A supervisor of the drugs' warehouse has to confirm the correctness of a request before enabling the warehouse worker to prepare the products to dispatch.

About the management inside the operational unit:

- Since administration to the patient is in units of posology, the conversion utility must be available also at this stage, in order to obtain single units.
- Products can be returned to the pharmacy, disposed or administered.

In addition, it is requested a functionality horizontal to all units:

- By using barcode readers, it must be possible to identify products with the precision of the triplet composed by drug code, batch and the expiration date. This has to be implemented at every step of the process from the moment a certain item enters the warehouse. For instance, receiving a valid request, the warehouse man must be able to select the item to send by simply

reading the barcode generated and printed on the box and the nurses in the operational unit that requested them should be able to double checking the correct product, while accepting it into the local cabinet, by reading the same barcode.

5.2.1.1 Use Case: Warehouse managing

The warehouse worker has to be able to view the state of every storage point throughout the structure, prepare new orders and accept new deliveries from suppliers.

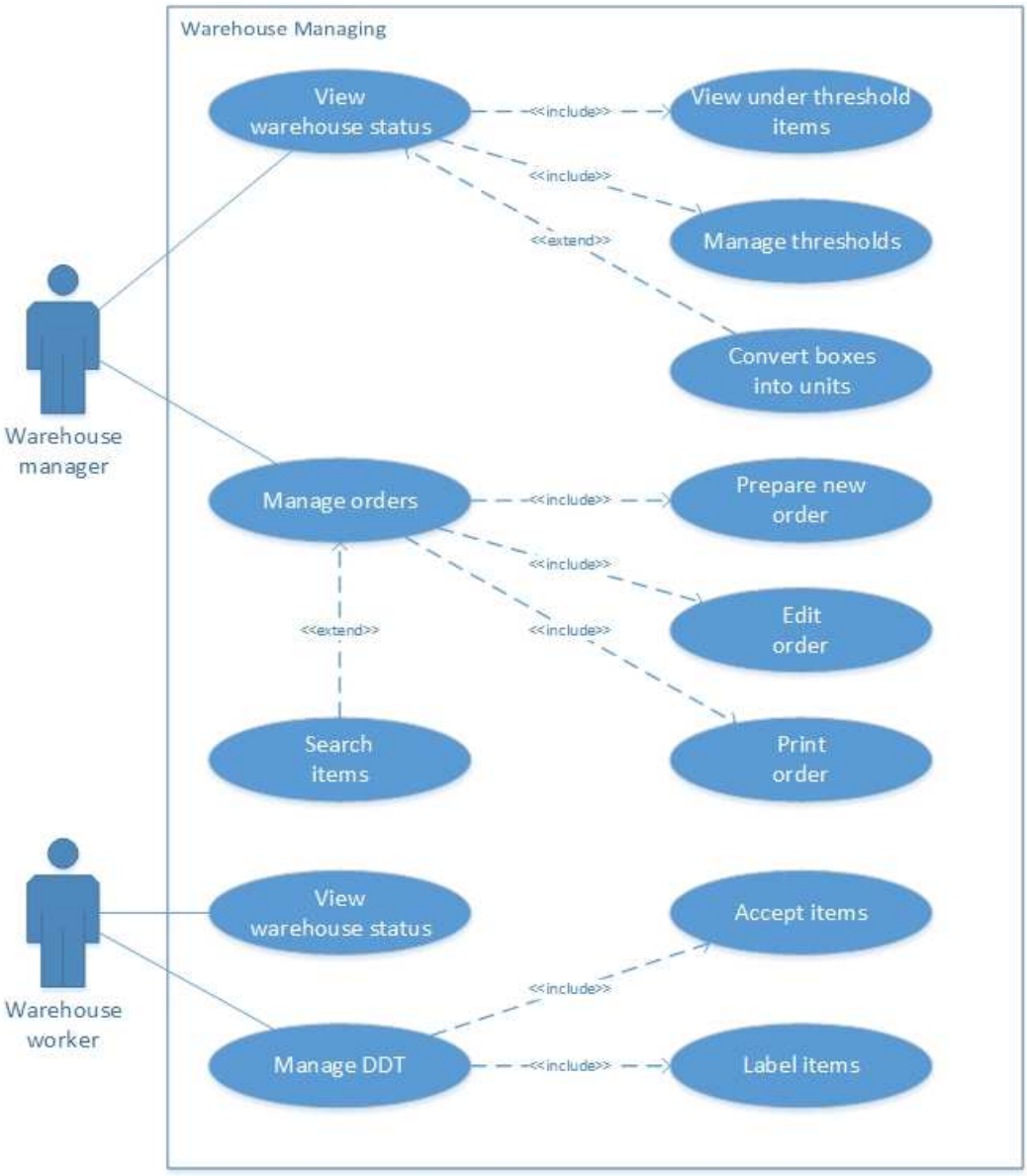


Image 41 – Use case 1

5.2.1.2 Use Case: Request from and dispatch to operational units

Nurses in operational units can request items to the drug store, items will be validated by the warehouse manager and then dispatched to the requesting unit by the warehouse worker.

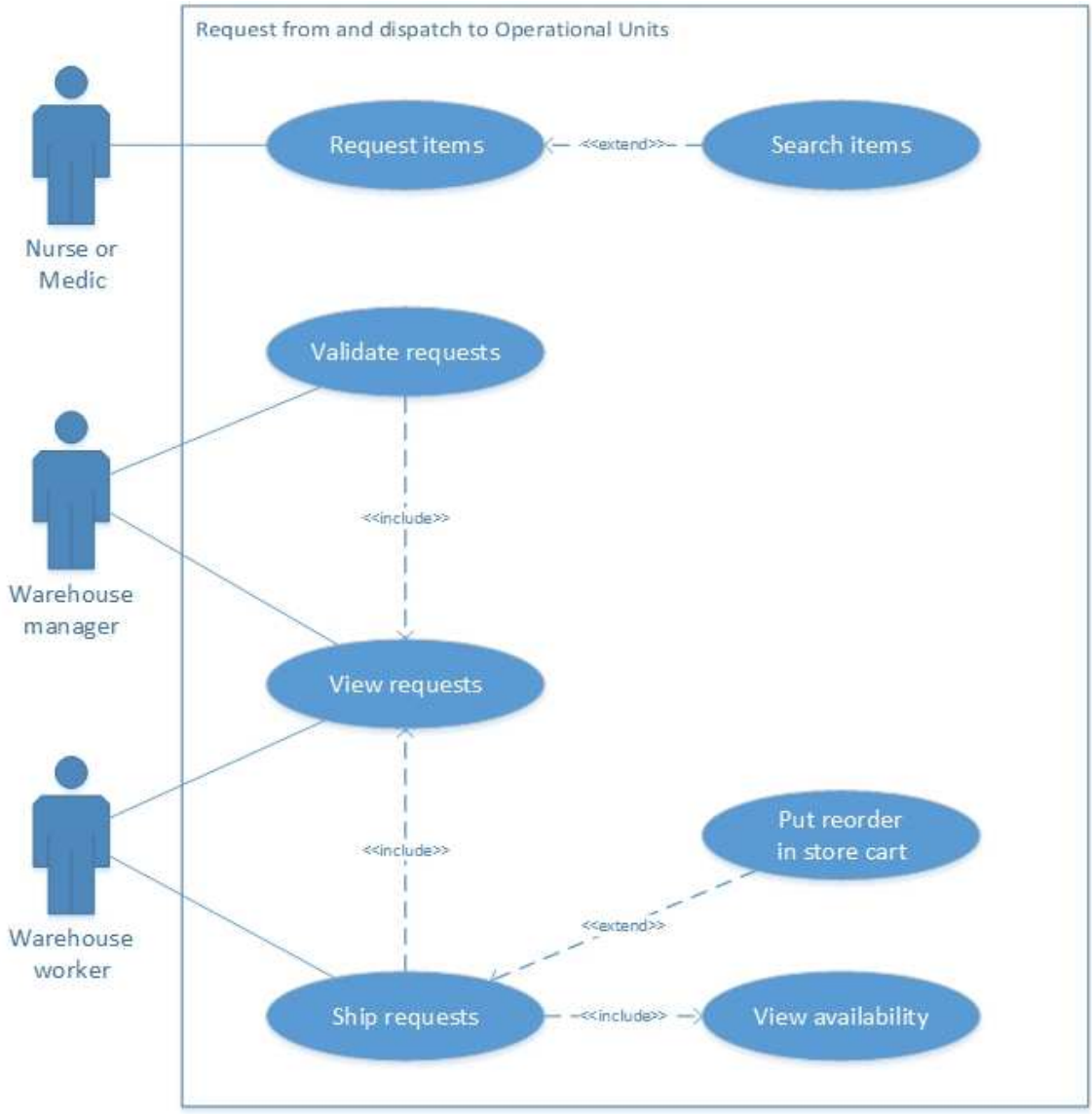


Image 42 – Use Case 2

5.2.1.3 Use Case: Managing pharmaceutical products and suppliers

A supervisor or a warehouse manager has to update or introduce items in the database records.

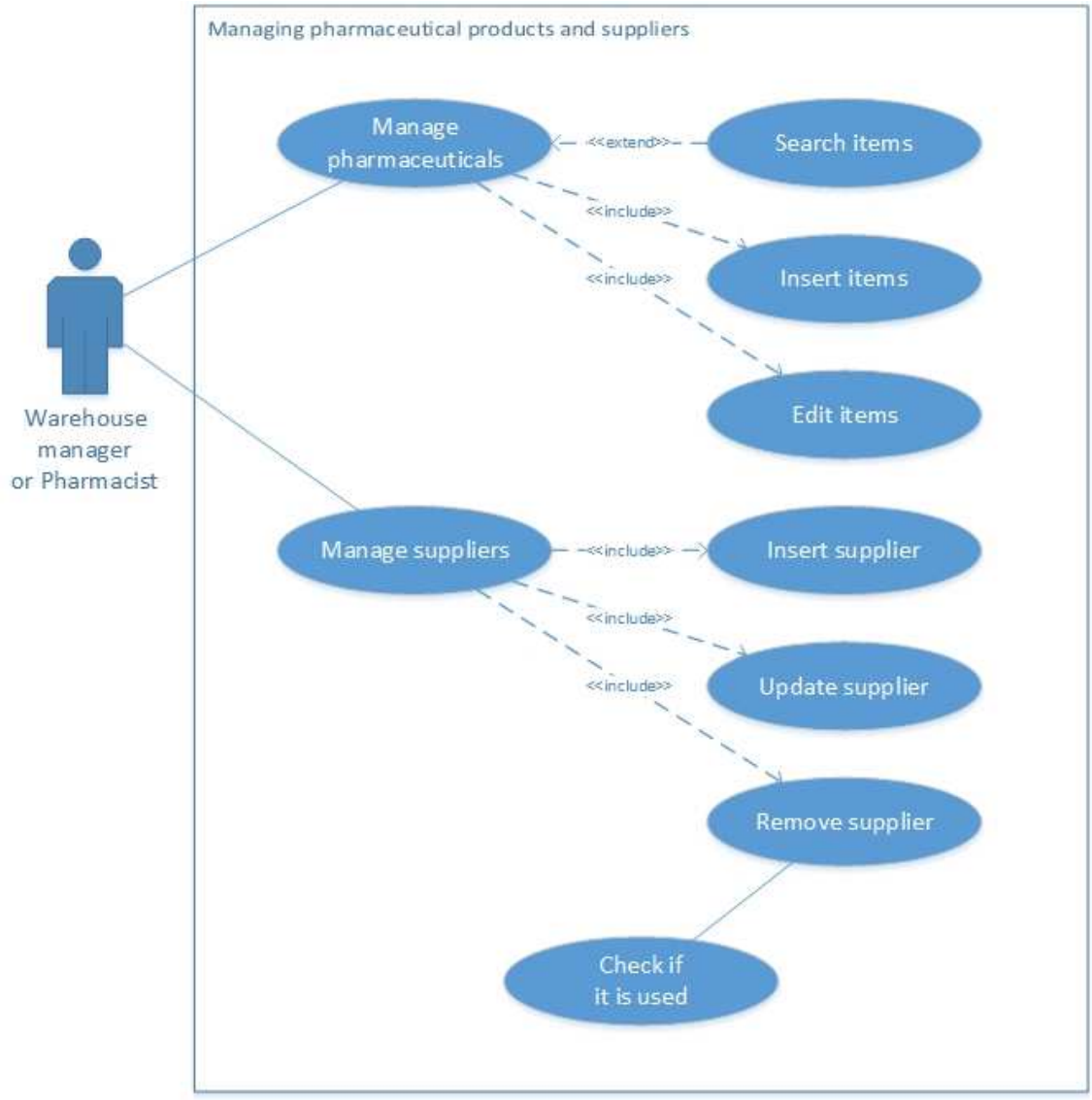


Image 43 – Use Case 3

5.2.1.4 Use Case: Products handling at operational unit level

Nurses and medics has to handle product once they reach them. They must accept products, move them through cabinets and ward carts, manipulate them by splitting boxes into units.

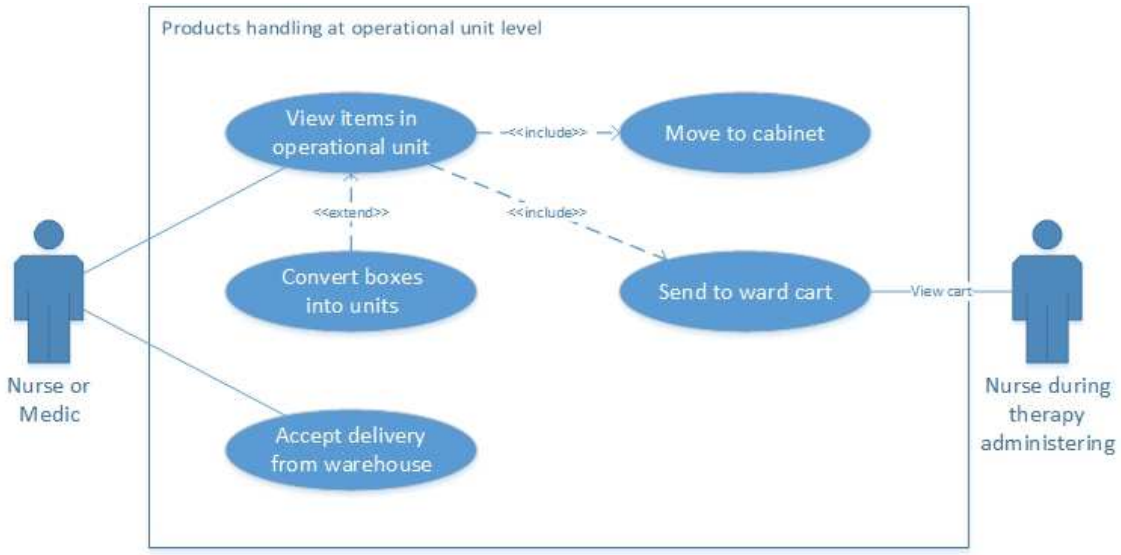


Image 44 – Use Case 4

5.2.1.5 Use Case: Products handling at patient's pack of drugs level

This use case is only related to the PharmaHospice mode, in which the operator brings items to the patient's home. In this scenario, administration is a functionality requested of PharmaHosp and happens by tying it together to the access scheduled in the operator agenda. Therefore this case must provide functionalities in order to view the actual list of items at the patient's house and to manipulate, administer, throw away or return items to the drug store.

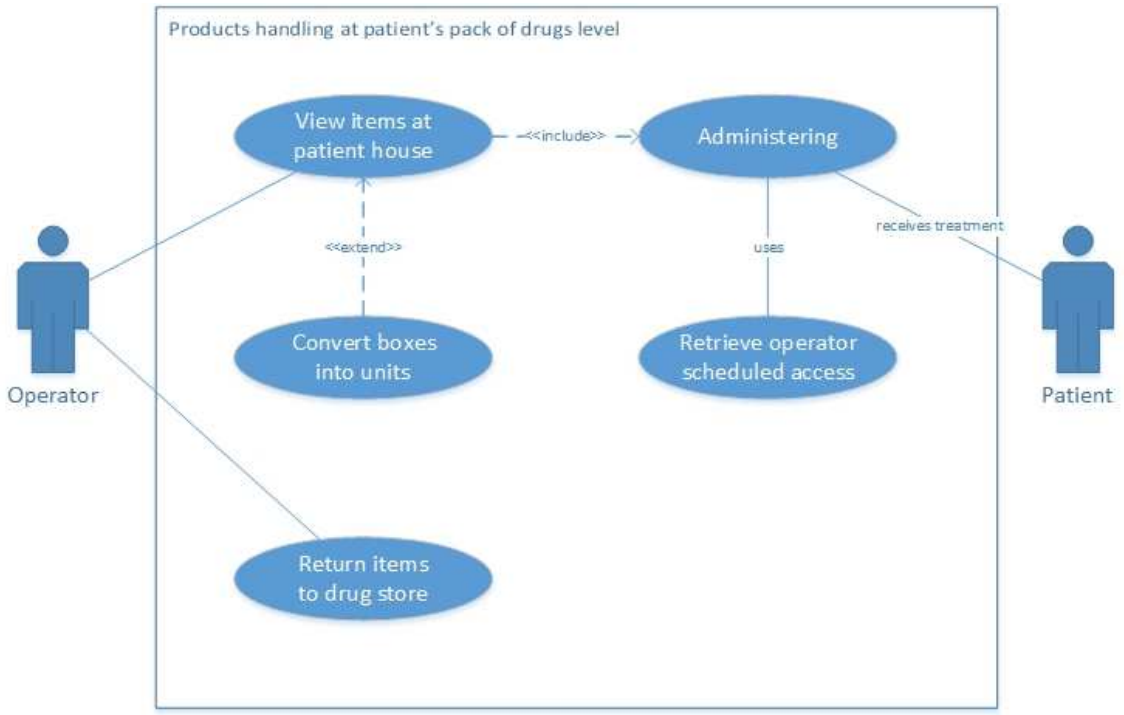


Image 45 – Use Case 5

5.2.1.6 Use Case: Managing cabinets and ward carts

This configuration operation is more essential in the PharmaHospice modality rather than the PharmaHospital one. In fact, managing cabinets and carts in a hospital means defining a structure, usually static in time. In the homecare scenario since operators and cabinets are strictly tied and the same happens for patients and carts, in fact they are renamed "Operator's bag" and "Patient's pack of drugs" this is a more dynamic process that has to be updated by a coordinator each time a new operator or a new patient enters the provider records.

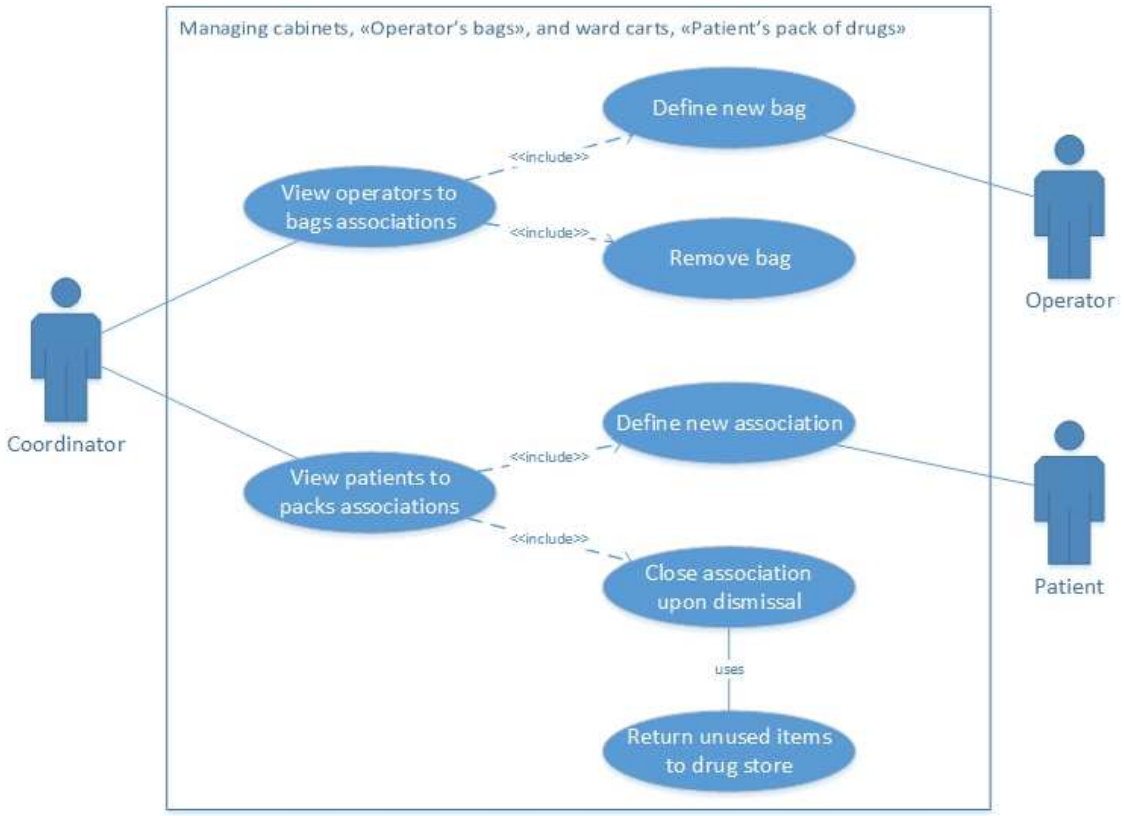


Image 46 – Use Case 6

5.2.2 Feasibility study

Once completed the requirements' analysis, certain aspects were highlighted as crucial for the whole process both in terms of correctness and sustainability.
In first place, it was necessary to further investigate the precision needed to achieve for the unique recognition process focusing between:

- Single dose of administration versus blister or box precision
- Uniquely identify every box versus identification by meaning of the triplet composed by drug code, batch and expiration dates.

The first problem is mainly an issue in terms of usability. In fact, for this kind of precision to be achieved, it would be necessary to label every single pill or unit of administration.

Clearly, without an automated process, it can be an evident wasting of man-hours since, considering an average of thirty pills per box, and an approximation of at least four hundred products in stock in a pharmacy, which is a little one, and already there are twelve thousand units of administration that need to be marked in some way. Moreover, it is out of the question to attach the label to the drug itself, both for space and contamination issues, therefore also a way to repackage items has to be considered.

Lastly, not all drugs are pills or distinguishable units. How should the software proceed with syrups and similar items? This matters of practicability and uniformity lead to the decision to achieve the precision of the single box and since by requirement it has to be possible to transform boxes into units, a functionality which is better to allow only to determined users, a best practice suggested was to label boxes and use this splitting method only when the unboxing operation actually happens, typically close to administration.

Medical devices bring additional considerations: the sizes of these products are quite variegate and in some cases it is necessary to reason in terms of units. For instance if we consider catheters, they are shipped to the warehouse in numbers of hundreds but they are dispatched in tens or less, it is therefore quite mandatory in this case to label the single unit.

Regarding the second choice to be made in managing items, whether to identify uniquely each box or the triplet composed by drug code, batch and due date, most of the reasoning has been already discussed in chapter two.

Usability is not involved in this case and there are no evident constraints limiting choosing one or the other approach. The choice is more based on studying the real world environment and the way users mostly deal with products. For a medic or a nurse there is no evident difference between two items with the same drug code, and therefore same exact posology and active ingredients. Adding the batch and

expiration date is compulsory due to safety and monitoring reasons. The decision came by answering the question on how to retrieve specific products in case of necessity. It appeared evident that was not only a sufficient condition but also a necessary one to know these two additional fields.
The drug administration panel, A.I.F.A. (Agenzia Italiana del Farmaco) in Italy, calls for retirement of certain products in terms of drug code, batch and expiration dates. If a single item is for any reason compromised, only the person handling it would be able to recognize the issue, and it would have to toss it right away, but it is not possible for a person outside the field of perception of the product to acknowledge that a certain product is compromised, unless it is due to some mismanagement during production or delivery and hence it would entail all the batch in question.

Analyzing the search methodology was also required. As already explained, the world of pharmaceutical products is not made only by drugs, it contains also the heterogeneous group of products of medical devices. This includes prosthesis, bandages, syringes, catheters and more. Each possesses its unique features and is made similarly by a vast group of producers. It is of difficult inquiry since there is not a common language defined in describing and grouping these products nor by AIFA™ nor by FarmaData™, our data source.
Therefore even if the objective at full functionality is to search all products needed by means of unique codes, abstracting from the need to inquiry the data source, at least in a start phase and every time a new product needs to enter the structure it is necessary to develop a search engine capable to obtain the requested items from the data source with an acceptable amount of precision.
Therefore it must be possible to search among multiple fields defining the drug's entity: requests must query for type of products, names, descriptions, eventual shapes and packages, number of units and

producers.
In addition, observing samples of medical devices showed a useful pattern in searching: being able to enquiry by both code and free text results in a considerable optimization of search times. This happens by using internal factory codes which are provided by our data source for medical devices. Unfortunately it is not a unique value and therefore needs to be filtered mostly by means of producers or gamma of the product. A certain assert about factory codes is that they are not replicated for a specific producer.

Observing the communication process between the drugstore and the operational units highlighted another choice to be done: whether to allow the requester to create single requests made of multiple items or strictly binding items and requests, meaning that one request is made for only one item in multiple quantities.
From a feasibility point of view, this issue drives toward balancing three attributes of a software: usability, complexity and elasticity of the flow. The user imagines that by making one request for all the items his workload would decrease. This is a reasonable scenario for an operational unit if we consider to do a sort of inventory every specific day of the week at the same hour. On the other hand, if we consider that with the single item approach it is possible to fulfill a request in the exact moment the necessity for an item is realized, a strong advantage point arises: with this pattern a product can be dispatched as soon as it is in stock.
This advantage point happens also in the warehouse, where once the request has been validated, the worker can ship each single item only by considering possible unavailability. In the case of multiple items for request, the order would not be fulfilled until all products are available. It would be possible to implement half process stages that allows to dispatch partial requests, but this would introduce a process less controllable and more passible of mistakes.

The validation process is requested mainly when handling special products that for matters of price or safety cannot be handled without permission. Validating products meant not only to flag with a go or block a request. The requirements specified the possibility to select the amount to validate in a four state process: request, validation, delivery and acceptance. The risk of slowing too much the whole process appeared concrete.
However there are a couple of opportunities introduced by this feature. Firstly, it allows a quality control that lowers the risks of requesting and dispatching products not usually stocked by the pharmacy or with certain rules. Secondly, if structured accordingly, it could not affect too much the flow since it can be easily filtered by lists of drugs, and therefore by defining a list of all the "to be validated" items, it should be possible an automated validation process for items outside that specific list.
From a user experience point, the validation process slows the reactiveness of the single item for single request paradigm, but in a healthcare scenario the need for control and monitoring is a priority.

The last part of this analysis is about how to effectively integrate the suite PharmaHosp with the suites already in place and in use of the wHospital Framework.
PharmaHosp is a particular suite, its environment is mostly self-contained since it handles pharmaceutical products from the entry in the warehouse until they leave it and it request few session data from the framework a part from identification of operational units. The first integration has to be for the hospital environment in which dispatched products ends up being administered to patients by means of a relative therapy in course.
This type of integration is quite direct since PharmaHosp would grant the tracing of all movements until the bed of the patient, and the therapy module would need in its logic only to counter check, before

administration a drug, if in this moment the product is available in a specific cart or cabinet and lower its remaining value with respect to the administered quantity.
The second scenario takes place in the homecare environment. This case does not have a module that treats administration properly, the structure reasons in terms of accesses to a patient's home but has no specific tracking of current therapies since they must follow prescriptions from the hospital that dismissed the patient. Therefore it is demanded to PharmaHosp to be able to take the process one step further and keep track from the pack of drugs left to the patient home of what is used at each access, what it has been used in the past and what is being returned to the warehouse once the treatment ends. The connection step will be the link between the scheduled accesses and the administered drugs, meaning that the only real intercommunication operation is a request from PharmaHosp to the module handling accesses in order to retrieve dates and identifications of the operator's schedule, the rest of the logic will be a PharmaHosp prerogative.

5.2.3 Design Phase

During design phase, the approach focused mostly on two aspects:

- Designing an effective object for pharmaceutical products that can be quite understanding by itself but it is surrounded by a list of attributes that completes its definition and designing above this structure a search process that can be as user-friendly as possible and effective.
- Define the pharmaceutical dispatching environment in terms of actors, entities, process flows and separation layers.

Shaping the actual entity representing pharmaceutical products does not pose real obstacles by itself. The main objective is to create a core

entity that possess all the essential information peculiar to a specific item. This entails the list of the three types of codes defined (AIC, EAN and REF), information about marketability such as prices and taxes, and generic informations given by commercial names and a brief descriptions. Technically, it will possess all the non-recurrent attributes of a pharmaceutical product.
Eight sub-entities will surround it, giving a more detailed view and categorization of an item:

- Type
- Shape
- Route of Administration
- Package
- Storage informations
- Active Ingredients
- ATC Class
- Producers and Dealers

These are the recurrent entities of the data source, they are not specific to a single item and are better qualifier of the category to which a product belongs. Three of these are crucial in the process of finding and virtualize products: the type of drug must be always given since it is essential to the definition of sub-categories that can speed up and refine searches. The same can be asserted for producers but mostly when handling items of the medical devices' type.
Active ingredients are on the other hand essential in order to successfully pass from a drug to its equivalent, for example when subscribing a cure and not an exact product. Obviously information about posology alone is not sufficient to obtain and exact equivalent, since there exists products which possess the same amount of active ingredients but have different shapes or routes of administration. Therefore all these methods must be configurable in order to tune the best categorization for the consumer needs.

Since the software must allow the creation of new drug's items, for instance if it lacks a certain experimenting product, it is essential to define how a custom product has to be defined in order to be successfully used. A custom drug for instance needs some constraints: a part from basic details such as a name and a drug code, which can be invented, it is indispensable for drugs to have defined their posology and active ingredients in order to admit them into the process of virtualization. In addition, due to wHospital constraints and completeness of the process, a custom item must be related with all operational units which intends to use it and at least with a list of drugs of the system in order to appear in fast searches both in the PharmaHosp suite and in other modules such the therapy one. This constraint is due to the necessity to keep custom items confined to the environment that uses it which can be only one operational unit of an entire hospital. New items that are actually in commerce, hence usable everywhere, will be added from FarmaData™ data source in scheduled update sessions.

Designing the search process must start from analyzing the requirements given by the customer. The objective is to provide a fast inquiry form, not overwhelmed by many filters to set and able via free-text searches to obtain the best result sets possible.

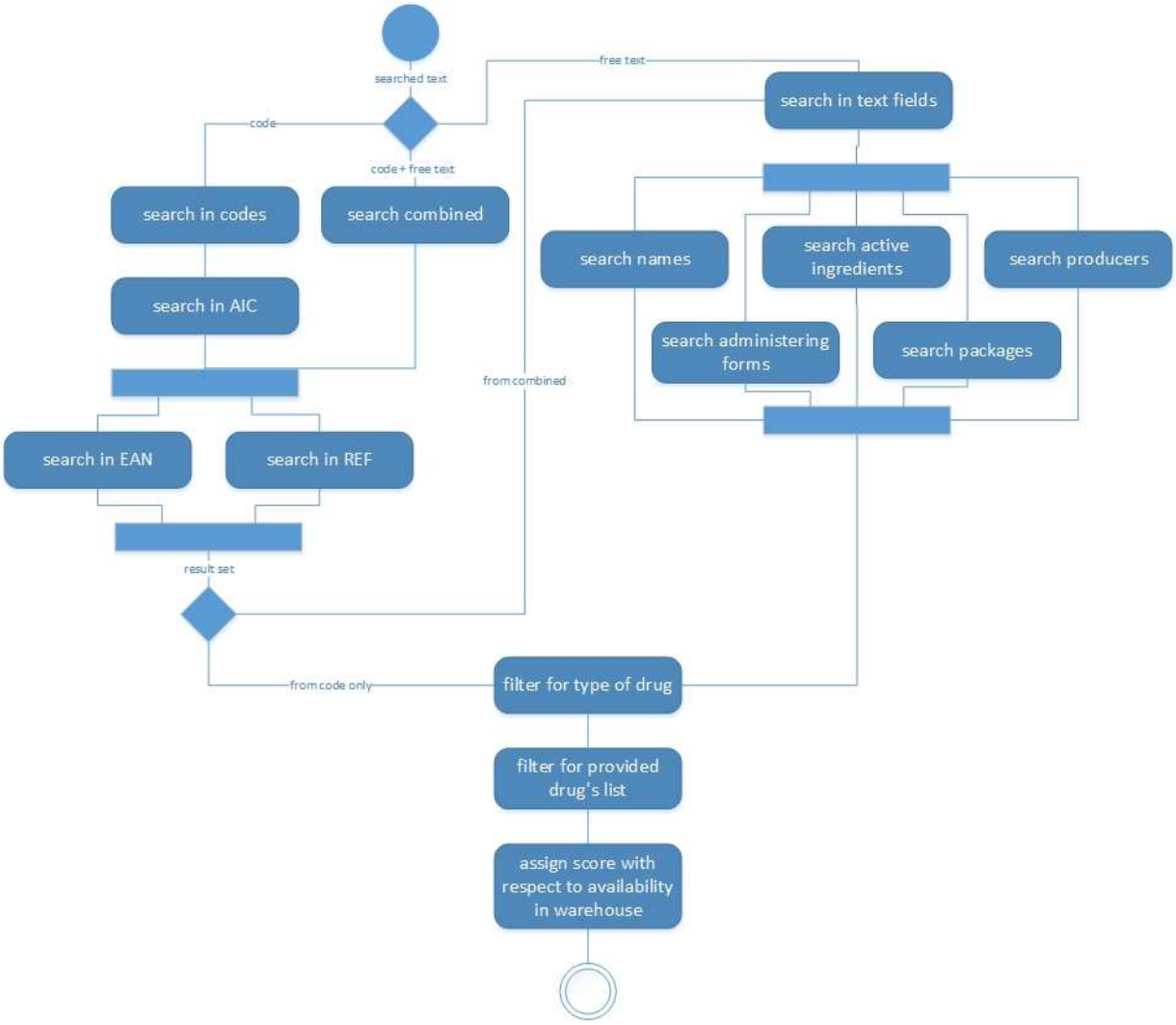


Image 47 – The search activity diagram

This implies the necessity to build a scoring mechanism for every field of the data source involved in the search process in order to being able to boost up or down results with respect to their relevancy. In addition this scoring mechanism allows a further tuning on the process based on searches history. Understanding in time the way medics and nurses interact with the suite and their way of processing pharmaceutical products will actually refine the search process. Since however a direct mean of interrogation of the database has to be provided by using codes, the design is focused on a two-way search with two input fields. The first allows only codes and via a structured query first searches for drug codes, previously also called AIC, which are unique, and then further searches into EAN and REF codes. Initially

EAN codes were reputed unique, and technically they are, but since they exists both in thirteen digits and fourteen digits form and our data source provides only the first ones the assertion of uniqueness was unsafe, it can happen that FarmaData™ decides to provide the fourteen codes as truncated thirteen ones in order to not rebuild their structure, at least as a temporary measure. Therefore the option is discarded but to the user is given the possibility to add via PharmaHosp these codes in their complete form.
REF codes, as already states, cannot considered unique since they are internal only to the factory and since they do not have a common schema and can have dashes or similar characters which for instance are not encoded into barcodes, they have to be searched also for partial matches.
The second way of searching pharmaceutical products has to combine free text hits between the following main attributes: name, description, shape, route of administration, posology, active ingredients and producers.
Ideally the intention was to not include the description since it usually is a contraption of posology, shape and route of administration but medical devices use this attribute vastly since they are not mapped as a drug.
Technically, the search can be again split in two ways: one for medical devices and their complementary set. In fact, medical devices rely for their identification on the commercial name, the description, the number of units and the producers, when drugs and similar rely on the other hand more on shape, route of administration, active ingredients, posology and of course the commercial name.
Once results have been filtered by text, the idea was to introduce the possibility to assign a score to each item and to tune the result again for numbers found in the free text string introducing also the possibility to find the more specific item quantity or posology in the requested unit of administration.

This set of results, only in PharmaHosp, will finally endure a last score procedure that is based upon the history of the drug store in question. If an item is currently in stock or was in stock, it will receive a bonus, of different amounts, that will boost it on top of queries. This intuition was based on the fact that usually after the first load of the warehouse of the drug store, the items ordered and requested are mostly with a high percentage of confidence the same, at least until new ones are released which typically does not happen all together.

One objective is to deliver a product that can suits both the hospital environment and the homecare one. This mean modeling this two world as much as possible as one single in order to maintain a degree of elasticity and avoiding as much as possible replication. In general, the both worlds can be represented as three inside layers surrounded by two more, one at the beginning of the process and one at the end.

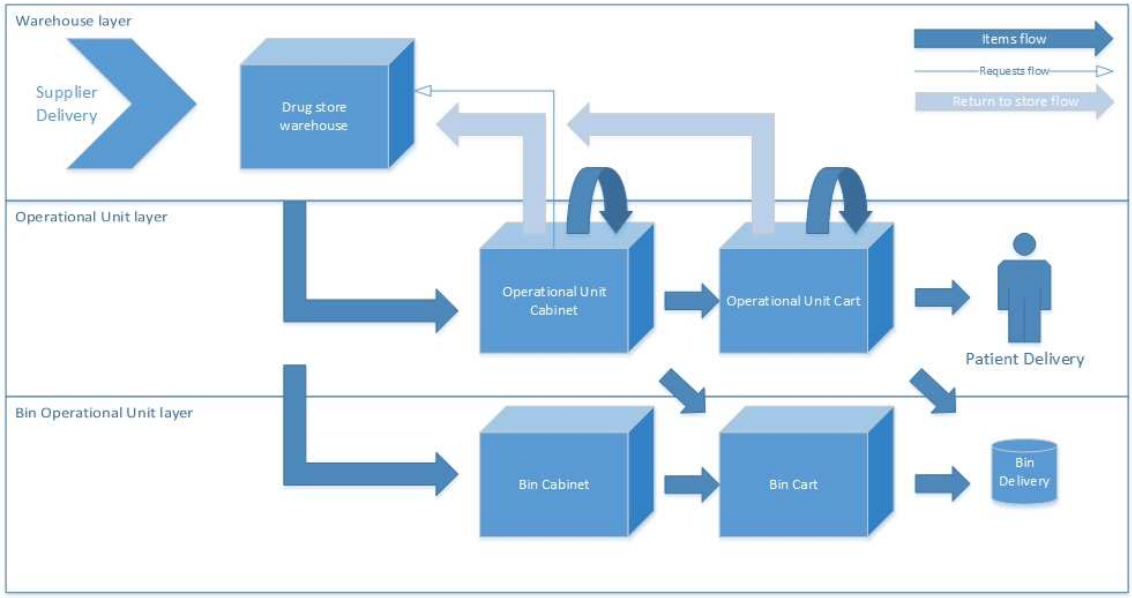


Image 48 – Flows of products and requests

There are two principal flows, the one of products and the one of requests. The first one always proceeds upwards in the hierarchy, and on the contrary the second one goes always backwards. There is the point of entrance of data, represented by external

suppliers who restock the drug store. The warehouse and its related operations belong to the first layer of this abstraction. This layer admits movements inside from the suppliers and outside towards operational units. It also considers internal movements in case of multiple stocking places, for instance warehouses placed on different levels. Items that exits the drug store, enters the operational unit world which is represented in the hospital environment by cabinets and carts and in the homecare one by bags assigned to moving operators and packs of drugs left at the patient's house. At this abstraction level the overlaying of this two worlds is almost perfect. The main process can be applied to both of them and it starts from a fulfilled request to the drug store. This means that we have items in our first internal layer, the cabinet or bag. At this point the only choice is to forward items towards carts and then to the outside layer of patients, and the respective integration with outside suites of the wHospital Framework. Requirements from the homecare client stated that inside movements between cabinet and carts must be forbidden, while in the hospital environment they are allowed as far as they are inside the same operational unit, this is because of the conceptual different nature of the couples cabinets/cart and bags/packs. The first ones belongs to an operational unit which can have multiple of them, the second are strictly bind to physical actors of the system, a bag belongs to an operator, who has only that one, and a pack belongs strictly to a patient. This entails that this requirement will have to be a configurable parameter of this design.

There are six actors interacting with the software:

- Suppliers: they receive order created by the drug store and sends items and documentation to it.
- Warehouse men: they are the workers of the drug store's warehouse, they stock items and prepare shipment for operational units.

- Drug store managers: they supervise the warehouse men work and can validate requests, edit drug information, create new items and accept returns from operational units.
- Nurses / Operators: nurses request new items to the drug store and manage movements inside the operational unit, operators manage their personal bag and delivers and administrate pack of products to the right patient.
- Medics / Coordinators: medics are much more related to the prescription phase in whospital, however they can do the same operations of nurses and can oversee the process. Coordinators are related to the homecare scenario, they are only supervisors that associate patients to operators and verify movements.
- Patients: they are the last step of the process, they simply are associated to a pharmaceutical product, technically in its dose unit form, in the moment the item is administered to them.

The design of how to maintain the exact status of a certain stage of the process, the warehouse, a cabinet or a cart, is focused on the concept of flows.
There is no database table that makes an instant picture of a warehouse by persisting the current status. Every place is dynamically calculated by subtracting to what has reached that particular stage what has left it. This means that actually there is no keeping of the state of a warehouse but is persisted the flow of requests and deliveries.
For instance, if our central drug store warehouse lists that we have three boxes of a certain item it could be because ten were accepted from two different orders (one of six and the other of four boxes) and then three were dispatched to a certain operational unit, three to another and one was thrown away because it had passed its expiration date.

This choice introduces two aspect to take in consideration in order to have a sustainable process:

- How to throw away items that are not going to be administered or will not keep by the store.
- How to send unused items back to the drug store.
- How to avoid excessive computational effort in order to obtain a status when in the database there will be a considerable amount of rows.

Since it has been stated that items can go only upwards, and requests only downwards. The idea behind managing products that are thrown away was to create a side operational unit, not really present in wHospital, with one cabinet and one cart, which is referenced every time a product is trashed, as showed in picture 4.2.3.2.
The item will not be sent to a stage belonging to the operational unit but to one outside it, but since it will be a delivery from it, the process is able to subtract it from the status of the storage center in observation.
This creates an actual flow control of dropped items that can be controlled and represents what is used or wasted in the structure at each level.
The return of items to the drug store is actually the only exception to the process. Even if it is based firmly on the principle of always moving upwards. The concept is that the item is delivered upwards in the flow with a special tag that highlights is removal state, and since it has to be reintroduced it actually makes its entrance back to the warehouse as a delivery from a supplier, marked in a way that it will not appear as a normal acceptance in the drugstore but will be visible only in the returns section.
Lastly, controlling the inevitable growth of computational effort, suggested to consider an operation of maintenance. Since while managing of a common warehouse there exists an operation called

"inventory" which is done regularly and in specific moments, this intuition lead to designing a function of the software that can historicize all actual movements and persists the actual status of the warehouse. This obviously entails losing the immediate knowledge of the provenience of a product but since data is not thrown away but kept in parallel history tables it can be retrieved as needed.

PharmaHosp must be configurable and adaptable to different customers' needs. Therefore it will need some configuration parameters that decide its behavior:

- Enabling grouping by batch and expiration date.
- Forcing prints whether a movement outside a storage unit happens
- Printing unique labels that identifies better all products.
- Enabling the possibility to users to edit pharmaceutical products' informations.
- Deciding the working paradigm between hospital mode and homecare mode.
- Setting the level for validation: all, none or by drugs' list.

	Drug Store	Operational Unit
Batch and Due Date	■	■
Cabinet Manager		■
Creating Drug Stores	■	
Editing Drugs	■	■
Mapping Drug Codes	■	
Must Print on Event	■	■
Linked to Accesses		■

Image 49 – Map of PharmaHosp configurations

Some configuration parameters regard operational units above the drug store.

All prints are handled using XSLT templates that are transformation tools that allow converting and styling XML documents (Bolchini D., 2006) such as the one generated from the serialization process.

5.2.4 Development Phase

The following are the actual implementation choices for the PharmaHosp suite. Technologies were obviously reduced by the already in place wHospital Framework.
The database is accessed via SQL Server Management™ and therefore is in sql and t-sql, developed following directives of the book on database design written by Carter J., 2002.
The business logic and class libraries are written in C# on Visual Studio™ and the interface is implemented via Silverlight applications and the contemporary use of Visual Studio™ and Microsoft Expression™.
The choice of an interface in Silverlight can be largely discussed since it is a plugin that can't work properly on every device that can access the internet, but it was a choice by the knowledge base of these tools that Laserbiomed was able to give and of easiness in fast implementing quite complex interfaces.

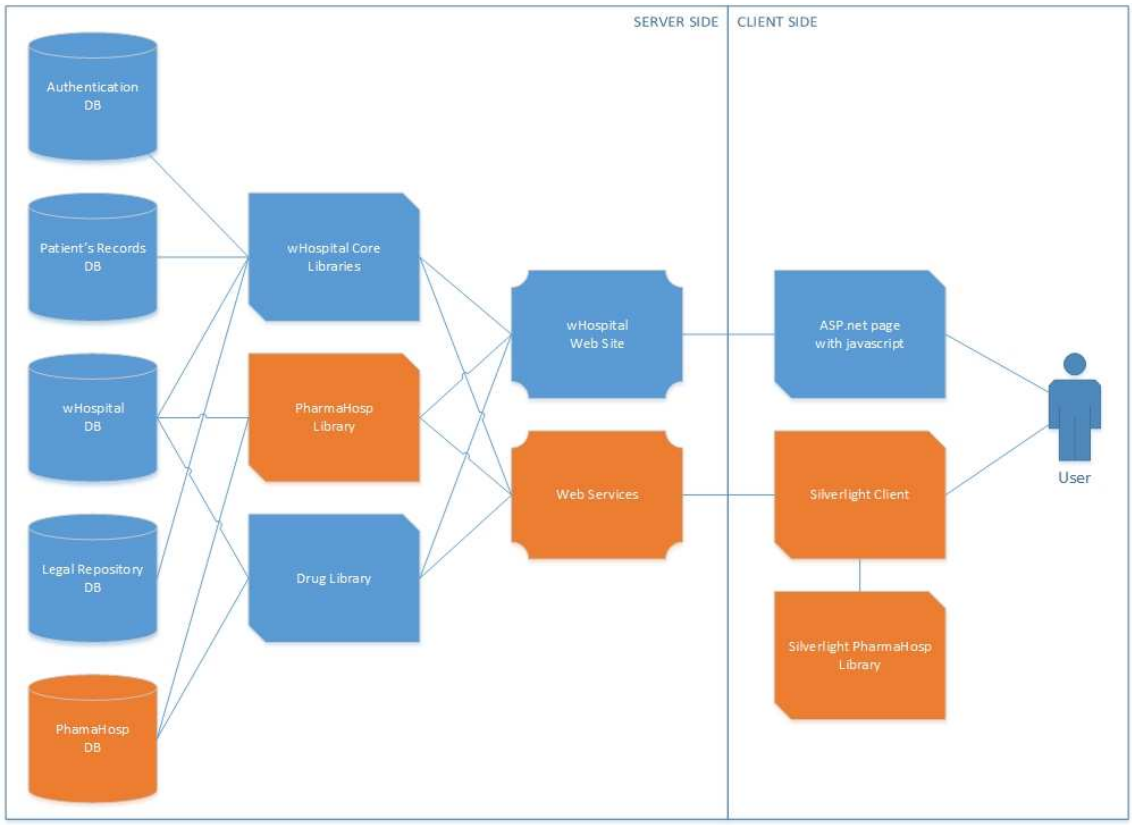


Image 50 – wHospital architecture

In picture, a schematic representation of the wHospital architecture with highlighted in orange the portion of the system involved in developing PharmaHosp.
Time constraints in releasing a software to test run lead to the choice of the Silverlight client, but further developments, especially in case of new requests from customers, considers the option to rebuild, in progressive steps, the interface in HTML 5 and J-Query in order to be more accessible in mobility such as on tablets and other devices.

5.2.4.1 Data Base Layer

The structure of the database is divided in two large schemas: one for the drug structured data and the other for the store informations and flows.
There is also a support schema that handles non environment related

operations such as the history of searches on pharmaceutical products and the queries to retrieve wHospital patients and users. The drug related part, named wH_RealDrug maps all drug related entities from the already described core attributes for a pharmaceutical product to the relations for drugs' lists and operational units.

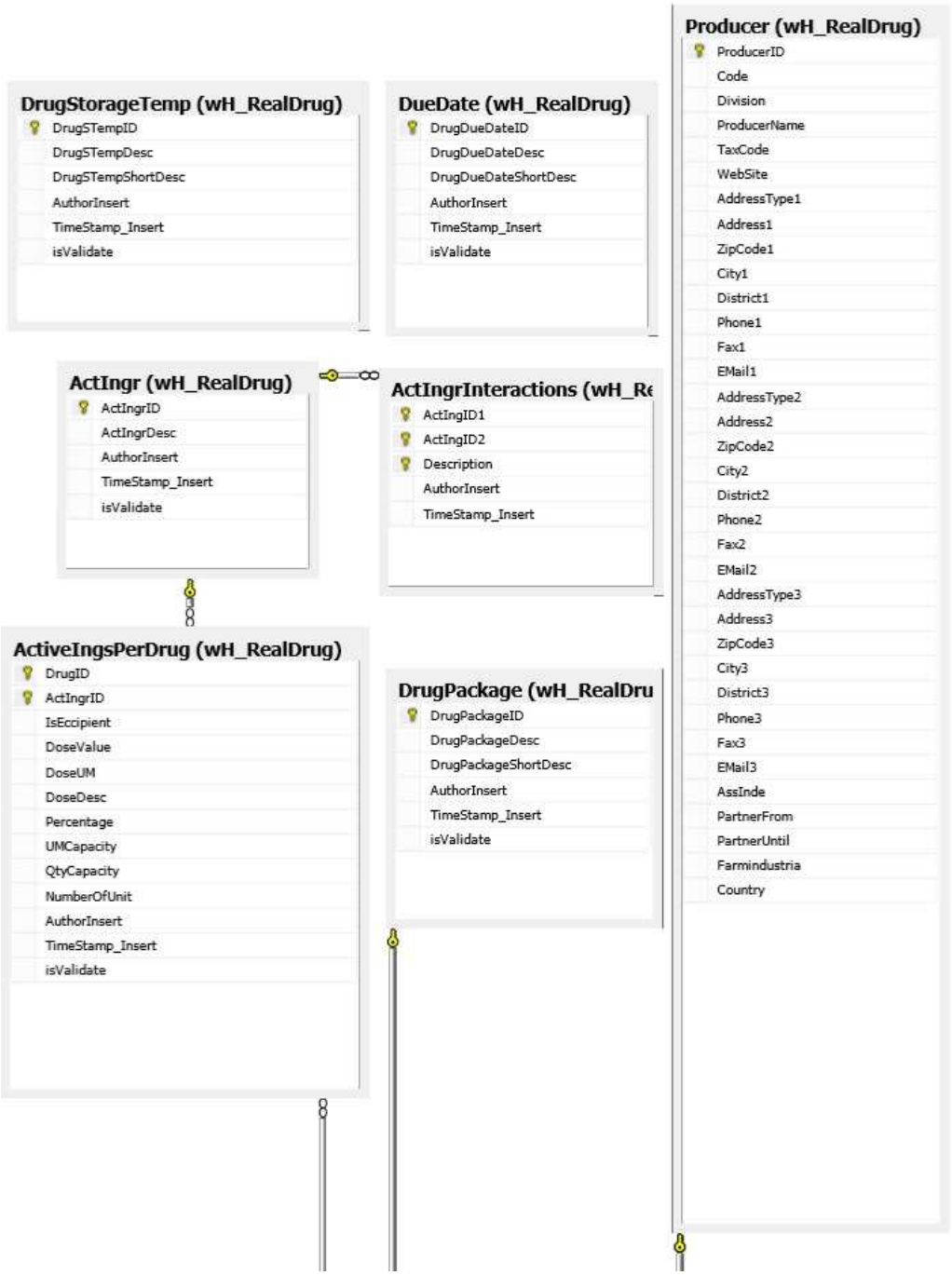


Image 51 – Drug class diagram part 1

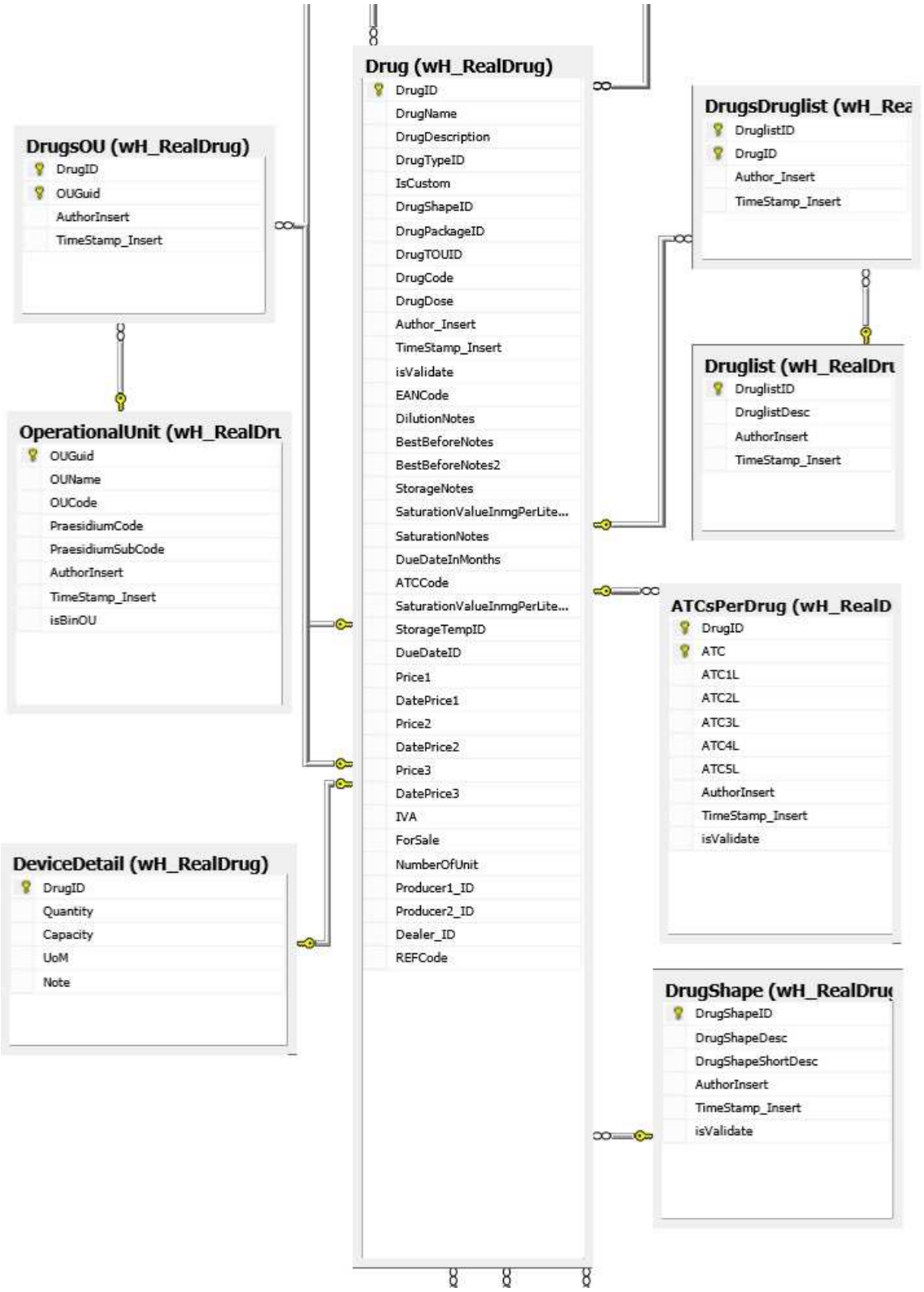


Image 52 – Drug class diagram part 2

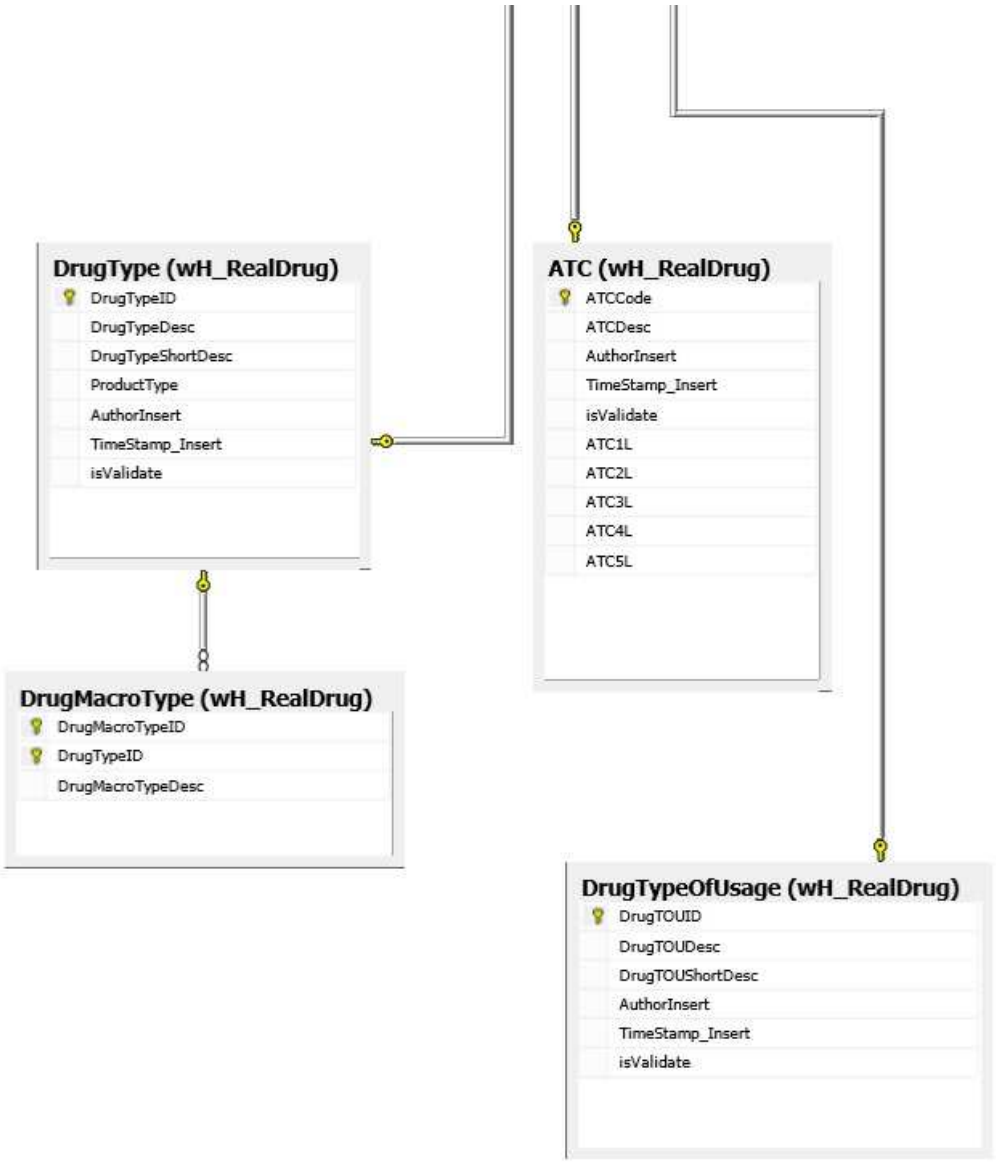


Image 53 – Drug class diagram part 3

There are two main entities:

- Drug: the core of the data retrieved from the data source. It is surrounded but all the relations about type, shape, package, route of administration, ATC, Producer.
- Active Ingredient: obtained also from the data source, essential in order to complete the drug related information.

This group of entities realize the complete information about pharmaceutical products at our disposal.

This data is integrated by three entities and their respective association entities that helps the workflow and precision of the process:

- Operational Unit: used both in the drug store management process and in order to understand where eventual custom drugs belong.
- Druglist: this lists of commonly used pharmaceutical products inside the healthcare environment are essential in order to reduce search times and correctness of the searched items.
- MacroDrugType: this association with types is in order to have a separation layer between different items, such as drugs and medical devices, without having to know the exact type of a product. For instance, for the data source, there is not the drug type but there is a more specific item as a "drug only for hospitals" or "generic drug" or "etic drug". It can not be asserted that a user will know this differences and therefore it is needed to create a layer above this distinction which is more generic such as "drug", "medical device", "para-pharmaceutical product".

  Stock is the schema name for the drug store management portion of the database. A part for the OUSetting table which defines how the PharmaHosp suite will behave, entities in this schema define the implementation of the flow designed in the previous paragraph.

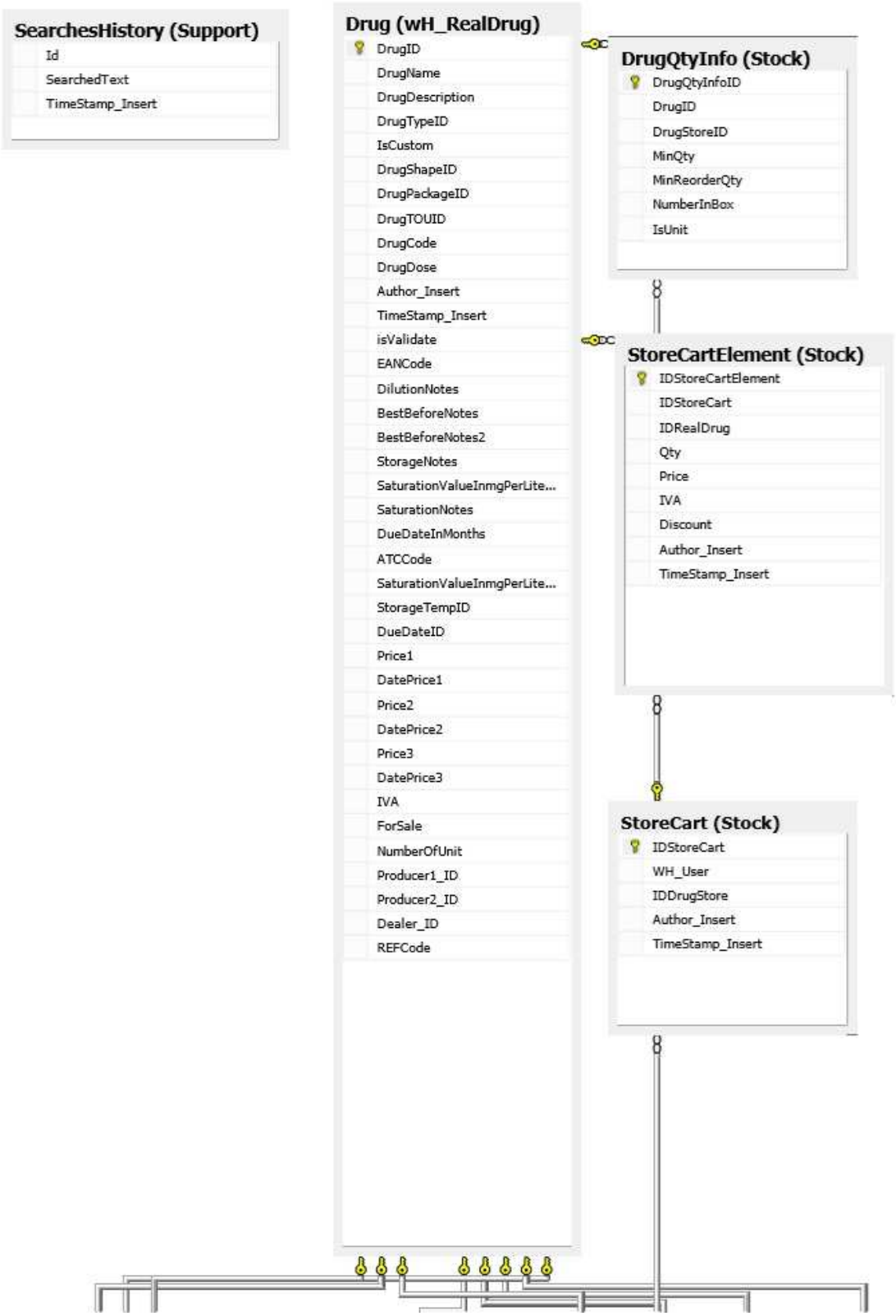


Image 54 – Stock class diagram part 1

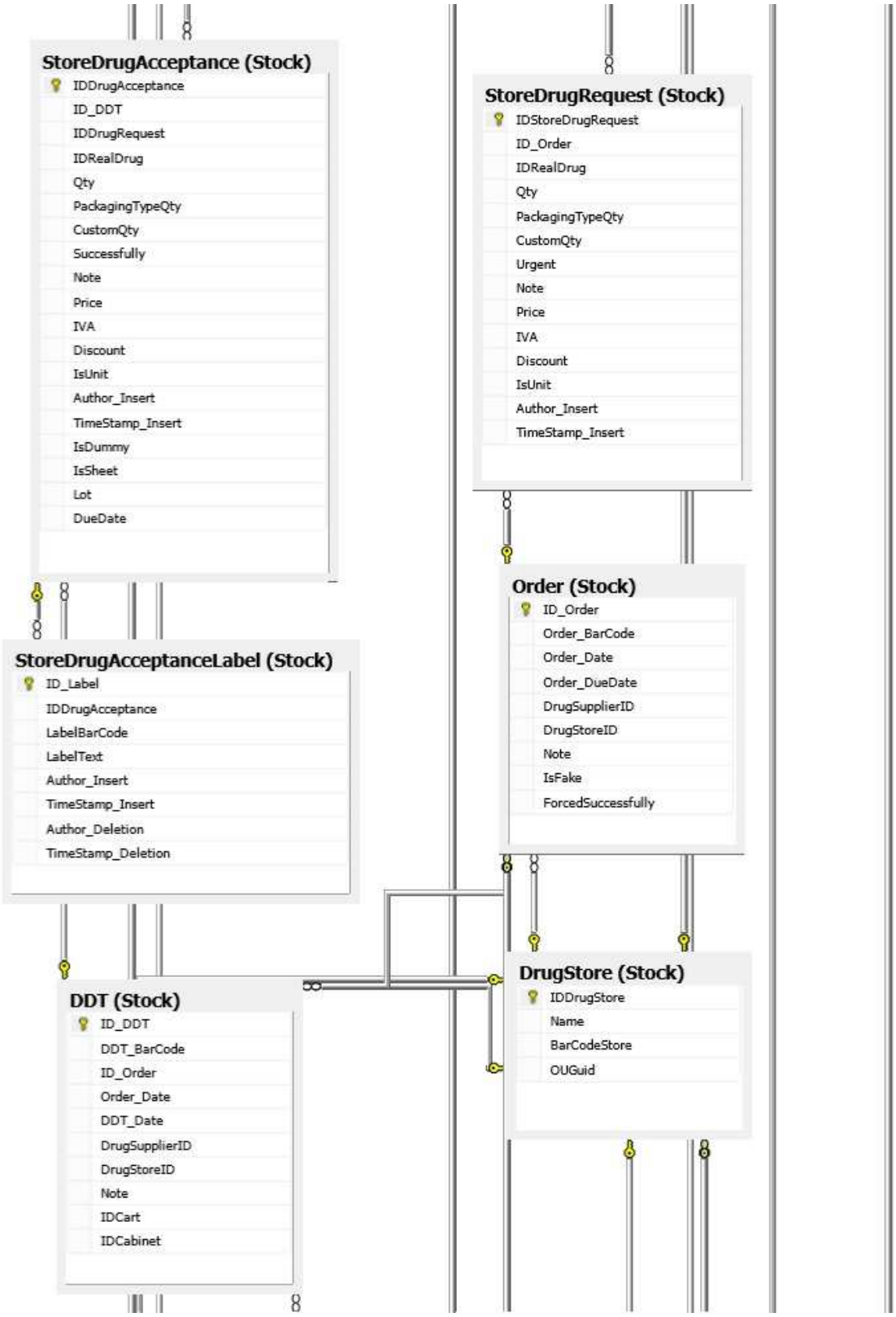


Image 55 – Stock class diagram part 2

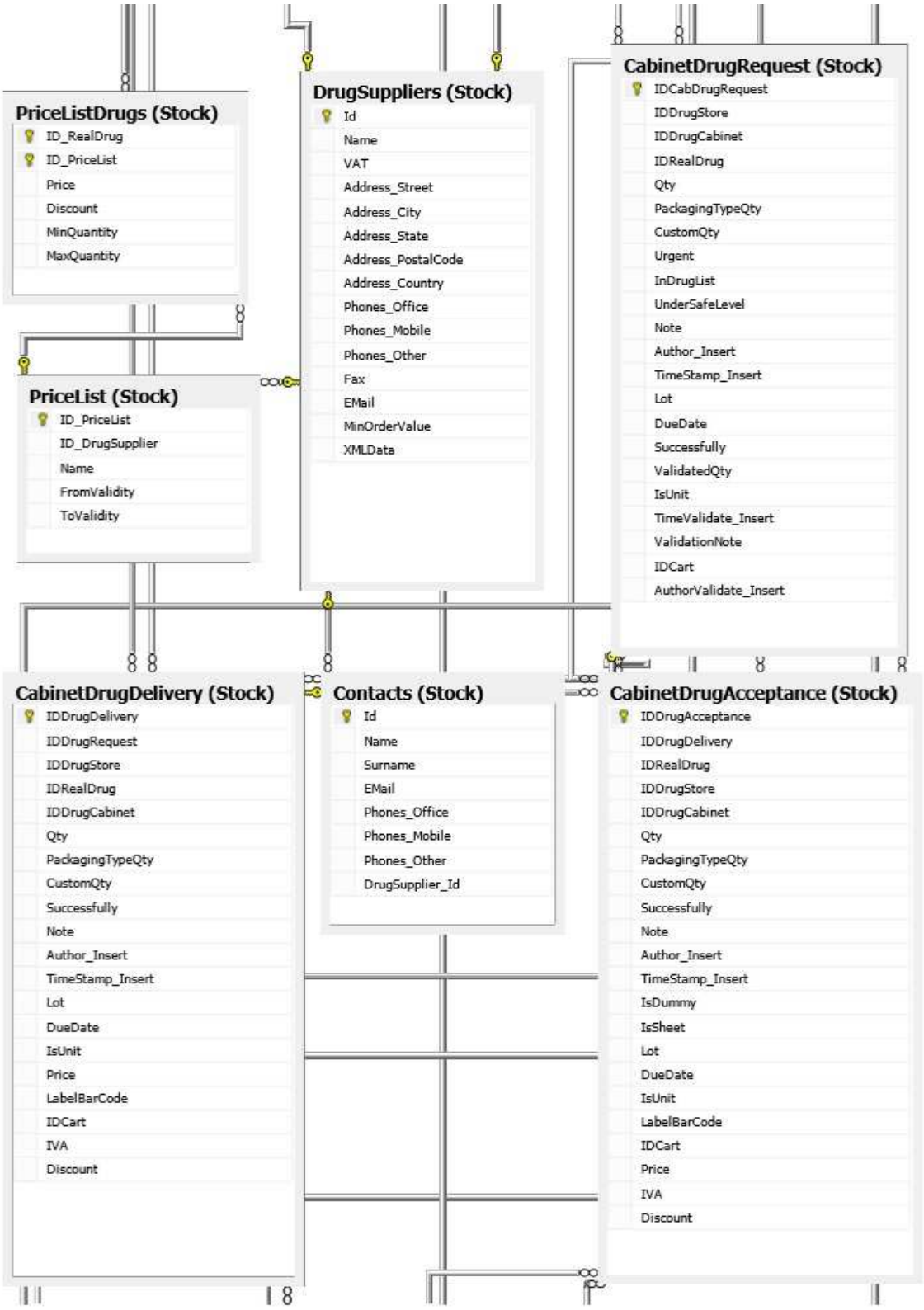


Image 56 – Stock class diagram part 3

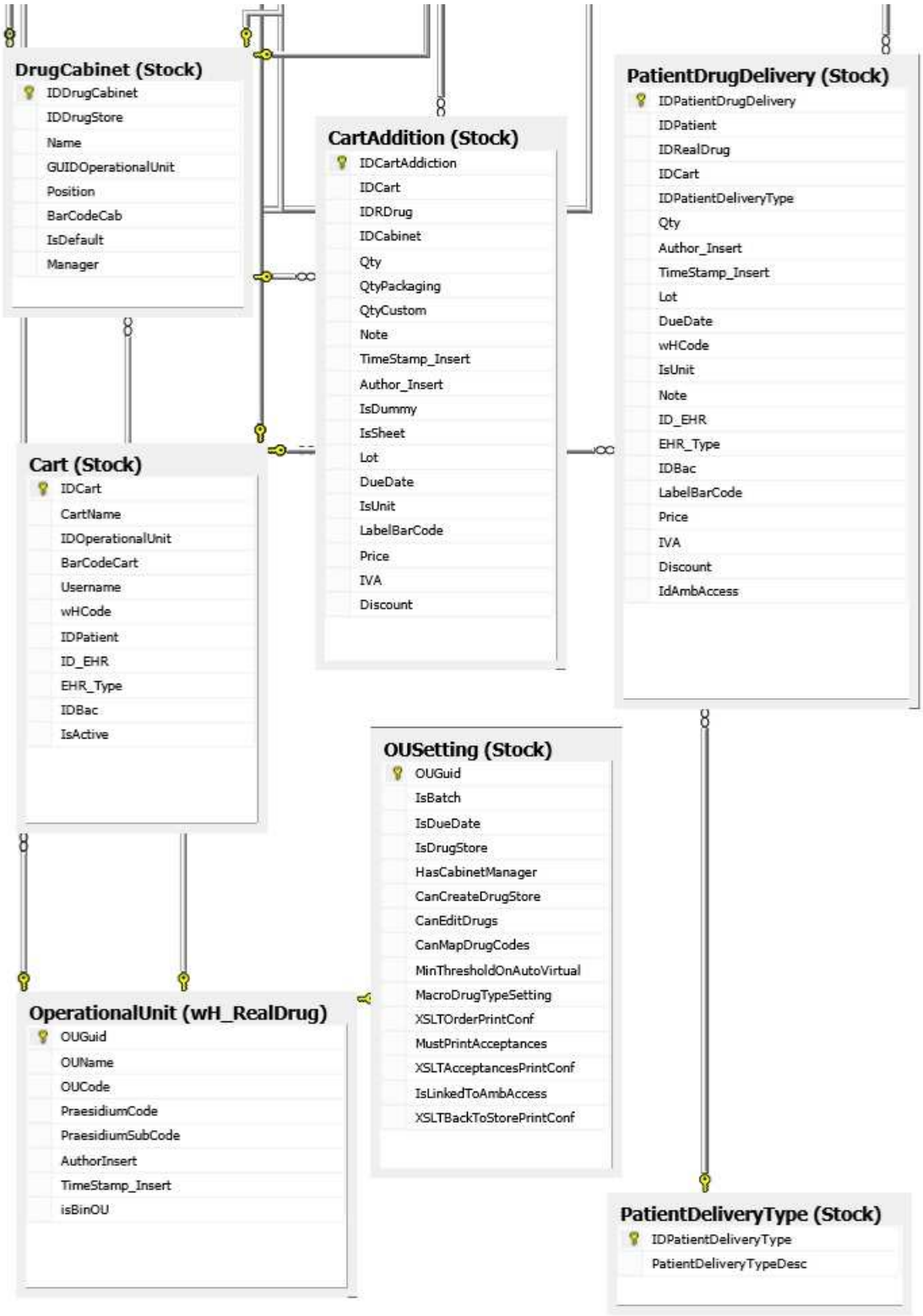


Image 57 – Stock class diagram part 4

Therefore we have some data coordinates that defines the healthcare structure by means of where it is sending the items:

- DrugCabinet: these are the first point of arrival inside an operational unit. In the homecare scenario they represent the bag of an operator.
- Cart: these are the last layer before administration of a product to a patient. In the homecare scenario they are the pack of drugs left at the patient home.
- Order and DDT: these do not represent the healthcare structure per se but are containers of requests and delivery from outside the environment, both physical and wHospital one. In fact, inside the suite the process is for single items but commonly communications with outside actors are kept by reasoning in terms of request composed by multiple items.
- Patient: this entity is defined only for the homecare scenario, in which the administration process is handled by the suite.
- StoreCart: this is the container for future orders outside the drug store. It represents the concept of shopping cart.

All these tables communicates one with the other by means of flows of information. These are implemented by a triplet of tables that realize the concept of request, delivery and acceptance, in some cases the last two are identical.
Each step of this flow must bring on several core information such as batch and expiration date of the product, if printed, the reference label, quantity and author.
In the case of a request to the drug store from an operational unit the field quantity is extended with one field more since it must support the validation process. Therefore one field is the mere quantity requested, the other one is the validated amount. The table returns also, as computed fields, the remaining quantity to deliver in case only a portion of it was dispatched and the accepted quantity. This last information is focused for the homecare scenario in which an operator

could accept only a portion of the request in order to deliver it promptly to the patient.

The core of the searching process is implemented by two stored procedure inside this database. The first one basically searches for products by means of the three main codes, AIC, EAN and REF. The second one more interestingly tries to understand the free text submitted by the user and to sort the results with a weighted approach to the significance of the fields queried.
The base approach is to create subsets of the search, one for each field considered. In this first implementation the search happens by means of name, description, producer, active ingredient and shape.
This five sub results are then joined removing redundant entries and summing the score assigned to each result, boosting up results that found hits in more than one search.
This approach starts by the assumption that with few exceptions a word of the string passed will be hit only by one of the searches and not by multiple, or at least in a consistent way.
For instance, searching "aspirin capsule" should trigger results from name and shape but not from the other three fields.
In the case of generic products it will happen that both name and active principle will be triggered by the word in the search string with the active principle but this actually works at advantage since it boosts up kind of products we are actually looking for.
Key to the success of this approach is the score mechanism. In fact an information such as the shape should not trigger results outside the scope of the search, but since it is a union between searches they will be considered. Weighting the shape field less than the name one will allow to reward hits in the name more than shapes and therefore ordering results by means of name plus shape, name, shape.
Deciding how to weight this fields is still in development since it will be

based on user experience in searching products. Actually the priority is set to name, producers, active ingredients, description, shape.

Stored Procedure for free-text and code search

[wH_RealDrug].[Drug_L_ByCodeAndText]

```
DECLARE @DrugCode varchar(20) = NULL,
@DrugName nvarchar(256) = NULL,
@DrugMacroTypeID int = NULL,
@DrugListID uniqueidentifier = NULL,
@TextSearch nvarchar(256) = @DrugName

-- CTE: Common Table Expression(s)
;WITH Results(DrugID, Score) AS
(
  SELECT [Drug].[DrugID], 4 AS Score
 FROM [wH_RealDrug].[Drug]
 LEFT JOIN [wH_RealDrug].DrugType
  ON [DrugType].DrugTypeID = [Drug].DrugTypeID
 LEFT JOIN [wH_RealDrug].DrugMacroType
  ON [DrugMacroType].DrugTypeID = [Drug].DrugTypeID
 WHERE CONTAINS([Drug].DrugName, @TextSearch)
  AND ([Drug].[ForSale] = 'H' OR [Drug].[ForSale] = 'S' OR
[Drug].[ForSale] IS NULL)
  AND (@DrugMacroTypeID IS NULL OR [DrugMacroType].[DrugMacroTypeID] =
@DrugMacroTypeID)

 UNION ALL

 SELECT [Drug].[DrugID], 3 AS Score
 FROM [wH_RealDrug].[Drug]
 LEFT JOIN [wH_RealDrug].DrugType
  ON [DrugType].DrugTypeID = [Drug].DrugTypeID
 LEFT JOIN [wH_RealDrug].DrugMacroType
  ON [DrugMacroType].DrugTypeID = [Drug].DrugTypeID
 LEFT JOIN [wH_RealDrug].ActiveIngsPerDrug
  ON [ActiveIngsPerDrug].DrugID = [Drug].DrugID
 LEFT JOIN [wH_RealDrug].ActIngr
  ON [ActIngr].ActIngrID = [ActiveIngsPerDrug].ActIngrID
 WHERE CONTAINS([ActIngr].ActIngrDesc, @TextSearch)
  AND ([Drug].[ForSale] = 'H' OR [Drug].[ForSale] = 'S' OR
[Drug].[ForSale] IS NULL)
  AND (@DrugMacroTypeID IS NULL OR [DrugMacroType].[DrugMacroTypeID] =
@DrugMacroTypeID)

  UNION ALL

 SELECT [Drug].[DrugID], 2 AS Score
 FROM [wH_RealDrug].[Drug]
 LEFT JOIN [wH_RealDrug].DrugType
  ON [DrugType].DrugTypeID = [Drug].DrugTypeID
 LEFT JOIN [wH_RealDrug].DrugMacroType
```

```
  ON [DrugMacroType].DrugTypeID = [Drug].DrugTypeID
 INNER JOIN [wH_RealDrug].Producer
  ON [Producer].ProducerID = [Drug].Producer1_ID
 WHERE CONTAINS([Producer].ProducerName, @TextSearch)
  AND ([Drug].[ForSale] = 'H' OR [Drug].[ForSale] = 'S' OR
[Drug].[ForSale] IS NULL)
  AND (@DrugMacroTypeID IS NULL OR [DrugMacroType].[DrugMacroTypeID] =
@DrugMacroTypeID)

  UNION ALL

 SELECT [Drug].[DrugID], 1 AS Score
 FROM [wH_RealDrug].[Drug]
 LEFT JOIN [wH_RealDrug].DrugType
  ON [DrugType].DrugTypeID = [Drug].DrugTypeID
 LEFT JOIN [wH_RealDrug].DrugMacroType
  ON [DrugMacroType].DrugTypeID = [Drug].DrugTypeID
 LEFT JOIN [wH_RealDrug].DrugShape DS
  ON DS.DrugShapeID = [Drug].DrugShapeID
 WHERE CONTAINS(DS.DrugShapeDesc, @TextSearch)
  AND ([Drug].[ForSale] = 'H' OR [Drug].[ForSale] = 'S' OR
[Drug].[ForSale] IS NULL)
  AND (@DrugMacroTypeID IS NULL OR [DrugMacroType].[DrugMacroTypeID] =
@DrugMacroTypeID)

   UNION ALL

 SELECT [Drug].[DrugID], 1 AS Score
 FROM [wH_RealDrug].[Drug]
 LEFT JOIN [wH_RealDrug].DrugType
  ON [DrugType].DrugTypeID = [Drug].DrugTypeID
 LEFT JOIN [wH_RealDrug].DrugMacroType
  ON [DrugMacroType].DrugTypeID = [Drug].DrugTypeID
 LEFT JOIN [wH_RealDrug].[DrugTypeOfUsage] DTOU
  ON DTOU.[DrugTOUID] = [Drug].DrugTOUID
 WHERE CONTAINS(DTOU.DrugTOUDesc, @TextSearch)
  AND ([Drug].[ForSale] = 'H' OR [Drug].[ForSale] = 'S' OR
[Drug].[ForSale] IS NULL)
  AND (@DrugMacroTypeID IS NULL OR [DrugMacroType].[DrugMacroTypeID] =
@DrugMacroTypeID)
)
SELECT TOP 250
  MAX(D.[DrugID]) AS DrugID
 ,MAX(D.[DrugName]) AS DrugName
 ,MAX(D.[DrugDescription]) AS DrugDescription
 ,MAX(D.[DrugTypeID]) AS DrugTypeID
 ,MAX(DT.DrugTypeDesc) AS DrugTypeDesc
 ,MAX(D.[DrugShapeID]) AS DrugShapeID
 ,MAX(DS.DrugShapeDesc) AS DrugShapeDesc
 ,CAST(MAX(CAST(D.IsCustom AS INT)) AS BIT) AS IsCustom
 ,MAX(D.[DrugPackageID]) AS DrugPackageID
 ,MAX(DP.DrugPackageDesc) AS DrugPackageDesc
 ,MAX(D.[DrugTOUID]) AS DrugTOUID
```

```
 ,MAX(DTOU.[DrugTOUDesc]) AS DrugTOUDesc
 ,MAX(D.[DrugCode]) AS DrugCode
 ,MAX(D.[DrugDose]) AS DrugDose
 ,MAX(D.[Author_Insert]) AS Author_Insert
 ,MAX(D.[TimeStamp_Insert]) AS TimeStamp_Insert
 ,CAST(MAX(CAST(D.isValidate AS INT)) AS BIT) AS isValidate
 ,MAX(D.[EANCode]) AS EANCode
 ,MAX(D.[DilutionNotes]) AS DilutionNotes
 ,MAX(D.[BestBeforeNotes]) AS BestBeforeNotes
 ,MAX(D.[BestBeforeNotes2]) AS BestBeforeNotes2
 ,MAX(D.[StorageNotes]) AS StorageNotes
 ,MAX(D.[SaturationValueInmgPerLiterMin]) AS
SaturationValueInmgPerLiterMin
 ,MAX(D.[SaturationNotes]) AS SaturationNotes
 ,MAX(D.[DueDateInMonths]) AS DueDateInMonths
 ,MAX(D.[ATCCode]) AS ATCCode
 ,MAX(D.[SaturationValueInmgPerLiterMax]) AS
SaturationValueInmgPerLiterMax
 ,MAX(D.[StorageTempID]) AS StorageTempID
 ,MAX(D.[DueDateID]) AS DueDateID
 ,MAX(D.NumberOfUnit) AS NumberOfUnit
 ,MAX(D.[Price1]) AS Price1
 ,MAX(D.[DatePrice1]) AS DatePrice1
 ,MAX(D.[Price2]) AS Price2
 ,MAX(D.[DatePrice2]) AS DatePrice2
 ,MAX(D.[Price3]) AS Price3
 ,MAX(D.[DatePrice3]) AS DatePrice3
 ,MAX(D.[IVA]) AS IVA
 ,MAX(D.[ForSale]) AS ForSale
 ,MAX(D.[Producer1_ID]) AS Producer1_ID
 ,MAX(P1.[ProducerName]) AS ProducerName1
 ,MAX(D.[Producer2_ID]) AS Producer2_ID
 ,MAX(P2.[ProducerName]) AS ProducerName2
 ,MAX(D.[Dealer_ID]) AS Dealer_ID
 ,MAX(P3.[ProducerName]) AS DealerName
 ,MAX(D.[REFCode]) AS REFCode
 ,MAX(DD.Capacity) AS Capacity
 ,MAX(DD.UoM) AS UoM
 ,MAX(DD.Note) AS Note
 ,wH_RealDrug.ufn_GetGrammaturaByDrug(MAX(D.DrugID)) AS Grammatura
 ,SUM(R.Score) AS Match
FROM Results R
INNER JOIN [wH_RealDrug].[Drug] D
 ON D.DrugID = R.DrugID
LEFT JOIN [wH_RealDrug].DrugType DT
 ON DT.DrugTypeID = D.DrugTypeID
LEFT JOIN [wH_RealDrug].Producer P1
 ON P1.ProducerID = D.Producer1_ID
LEFT JOIN [wH_RealDrug].Producer P2
 ON P2.ProducerID = D.Producer2_ID
LEFT JOIN [wH_RealDrug].Producer P3
 ON P3.ProducerID = D.Dealer_ID
LEFT JOIN [wH_RealDrug].DrugShape DS
```

```
 ON DS.DrugShapeID = D.DrugShapeID
LEFT JOIN [wH_RealDrug].DrugPackage DP
 ON DP.DrugPackageID = D.DrugPackageID
LEFT JOIN [wH_RealDrug].[DrugTypeOfUsage] DTOU
 ON DTOU.[DrugTOUID] = D.DrugTOUID
LEFT JOIN [wH_RealDrug].[DeviceDetail] DD
 ON DD.[DrugID] = D.DrugID
LEFT JOIN [wH_RealDrug].[DrugsDruglist] DL
 ON DL.DrugID = D.DrugID
WHERE (@DrugCode IS NULL
 OR [EANCode] LIKE @DrugCode
 OR [REFCode] LIKE @DrugCode+'%')
 AND (@DrugListID IS NULL OR [DL].DruglistID = @DrugListID)
GROUP BY R.DrugID
ORDER BY Match DESC
```

This approach proves quite solid for searches on drugs. Looking for medical devices rises more issues: the problem is mainly in how the data source handles this information. The actual useful search fields are three: name, description and producers, this is obvious since this kind of products does not have active ingredients or shapes. The problem is keeping unwanted results down in the boosting process since the data is not formatted in a unique way. For instance, "catheter" could be written as "cat" or "cath", name and producer could coincide and the line of the product could not be present, this means that the results would contain all products of that producer no matter what. Therefore some guidelines in searching medical devices where developed:

- Avoiding common names and looking for commercial ones.
- Always state the producer in order to boost up in searches their related products.
- Jointly search for internal factory codes, REF, and producers in order to select among the possible many results with the same code the one from that particular manufacturer.

Since the look up is on strings of free text it is necessary to search by means of "contains" functions. Name could contain more than just the

mere commercial reference and producers could have acronyms such as SPA or SRL. In addition, this method is expensive from a time and computational point of view. This requested to create indexes for the five fields at the core of the search procedure and the three codes. Consequence of this decision is a space consumption which is relevant but in terms of twenty percent of the dimension of the database, on the other hand waiting times for results have considerably reduced by around sixty percent.
The following three pictures shows the quality of the result set concerning hits and performances.

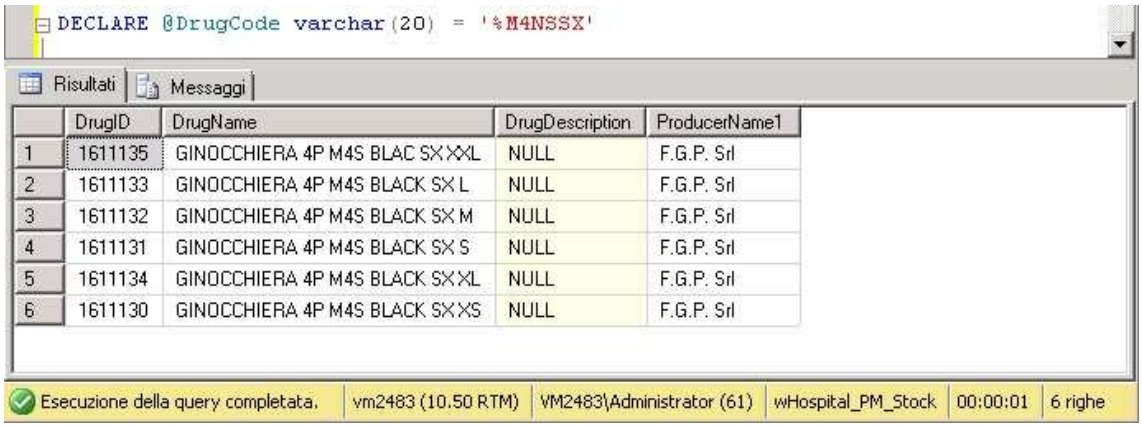


Image 58 – Search by code

When searching for codes different than AIC time consumption in search is quite plain since it is a simple search on a table.

```
DECLARE @DrugName nvarchar(256) = '%aspirina%|%compresse%|%bayer%',
    @DrugMacroTypeID int = 1,
    @DrugListID uniqueidentifier = NULL
```

Risultati | Messaggi

	DrugID	DrugName	DrugShapeDesc	ProducerName1	Match
1	5891	ASPIRINA	COMPRESSE	BAYER SpA	7
2	5897	ASPIRINA C	COMPRESSE EFFERVESCENTI / COMPRESSE SOLUBILI	BAYER SpA	7
3	5904	ASPIRINA	COMPRESSE	BAYER SpA	7
4	5911	ASPIRINA C	COMPRESSE EFFERVESCENTI / COMPRESSE SOLUBILI	BAYER SpA	7
5	5915	ASPIRINA	COMPRESSE/TAVOLETTE MASTICABILI	BAYER SpA	7
6	5921	ASPIRINA	COMPRESSE	BAYER SpA	7
7	5918	ASPIRINA	GRANULATO	BAYER SpA	6
8	5920	ASPIRINA	GRANULATO	BAYER SpA	6
9	5899	ASPIRINA	GRANULATO	BAYER SpA	6
10	41825	VIT.C BAYER	PREPARAZIONE INIETTABILE	BAYER SpA	6
11	41826	VIT.C BAYER	PREPARAZIONE INIETTABILE	BAYER SpA	6
12	77087	KOGENATE BAYER	PREPARAZIONE INIETTABILE	BAYER SpA	6
13	77088	KOGENATE BAYER	PREPARAZIONE INIETTABILE	BAYER SpA	6
14	77089	KOGENATE BAYER	PREPARAZIONE INIETTABILE	BAYER SpA	6
15	77093	KOGENATE BAYER	PREPARAZIONE INIETTABILE	BAYER SpA	6
16	77095	KOGENATE BAYER	PREPARAZIONE INIETTABILE	BAYER SpA	6
17	3540...	EPID COMPRESS...	COMPRESSE	SPECCHIASO...	5
18	3540	EPID COMPRESS	COMPRESSE	SPECCHIASO	5

Esecuzione della query comple... | vm2483 (10.50 RTM) | VM2483\Administrator (64) | wHospital_PM_Stock | 00:00:01 | 250 righe

Image 59 – Search by free-text

Considering the free-text situation, the logic creates a big set of results since it rounds up all elements with at least one keyword. Anyway, the point assigning systems proves able to sort by the requested means. In order to reduce the final set the last selection cuts results to two-hundred and fifty.

```
DECLARE @DrugCode varchar(20) = '%1055022',
    @DrugName nvarchar(256) = '%pharmaplast%',
    @DrugMacroTypeID int = 5,
    @DrugListID uniqueidentifier = NULL
```

Risultati | Messaggi | Piano di esecuzione

	DrugID	DrugName	ProducerName1	Match
1	1698649	PHARMAPLAST CATET NEL CH22 100	CONVATEC ITALIA Srl	4

Esecuzione della query completata. | vm2483 (10.50 RTM) | VM2483\Administrator (71) | wHospital_PM_Stock | 00:00:00 | 1 righe

Image 60 – Search by combining the two methods

The combined search process proves the best in both resulting set and time consumption. It is mostly used for medical devices since they do not possess defining attributes such as active ingredients and are not

coded by AIC.
The search procedure does not end at this stage, in the database layer there is a first process that creates a subset of the pharmaceutical by means of the text searched or the codes or eventually both of them. In the business layer a second process takes place that tries to reason on quantities.

5.2.4.2 Business Layer

The business layer starts with the objects that replicates the data structure. Initially it was realized by creating the object, with its fields and methods for insertion, update and deletion, and its relative collection, entrusted with retrieval of single or lists of items and eventually bulk saves of object's collections.
A refactoring process took this choice to be replaced by using the Entity Framework™ which asks to the developer the creation and personalization of all the methods described but takes charge of the creation of objects, references and collections.
While with the first strategy there was more rigidity towards database related changes, since it would be inevitable to modify manually also the class library, this approach grants the possibility to update the structure automatically. On the other hand it introduces an issue with replication of data and context. The first approach retrieved only what the system was interested in and therefore there was no risk for cycling references. With this one, the object created contains references to the collections that inevitably contains references to the element itself. This would not be a problem in an attached scenario in which the application communicates directly with libraries and database. In a web application this is not possible since while database and libraries are on the servers, the application is on the client side and therefore it is necessary to send the object by serializing it.

This issue would be resolved by not using a plugin like Silverlight, but the reasons behind this choice, mostly time related, were already explained at the begin of this chapter.
The issue of cycling references is resolved with two different approaches determined by the type of data requested at application level: it is possible to cut collections from the serialization of the item to be sent, or to send the collection directly not including the referenced parent object, or only its unique identification. This obviously is for a generic approach, then in case of special needs it is possible to create an ad hoc serialization process.

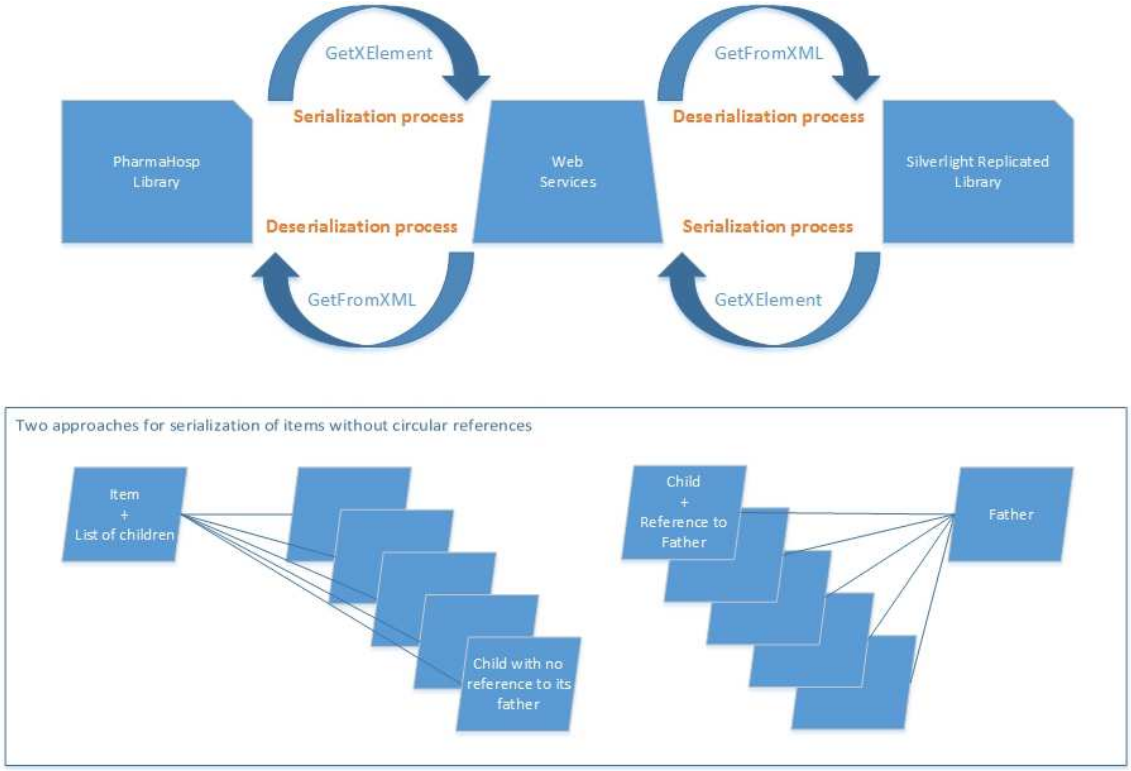


Image 61 – Serialization process

The class library does not only contains the structure replication of the database. It contains the methods that implement the logic of the application and the web services that enables the communication process with the client.
The web services are developed with a stateless approach, as suggested by Fowler, 2003, in order to provide a base that can be used also by other applications that would like to interface with the library

in the future. Being stateless implicates that there is no need for an active context to work on but only for key data passed to the service, for instance the operational unit identification parameter or a drug code. This is intended for example for the case of the therapy suite of the wHospital Framework which in the future is supposed to retrieve from this library the actual list of available products and administer them directly.
In this layer the second phase of the search process is implemented: here the list retrieved from the database is observed for numbers in in sub entities such as active ingredients doses and number of units in packages. This step also determines whether a product is actually or has been previously been used by the drug store referenced. This is a key factor in obtaining a successful search process. In fact a part from filtering for lists' of drugs, which is possible with some parameters, the suite is capable of boosting up results based on their history inside the structure. This works fine because most products are always the same and therefore one they have been introduced for the first time they will be easily retrievable in any moment, even if neither the list of drugs, nor the unique labels are used.

5.2.4.3 Application Layer

Since this is a web application, mostly used in intranet environments, the business layer is works detached from the application one. This entails that objects are replicated also in this layer but here they have practically no logic and they can be personalized for presentation purposes without mixing the business logic with interface matters. Silverlight does not provide the possibility to implement graphics that an ASP.Net™ mixed with javascript™ would not, but it enables them with less time effort and in a more reliable way.

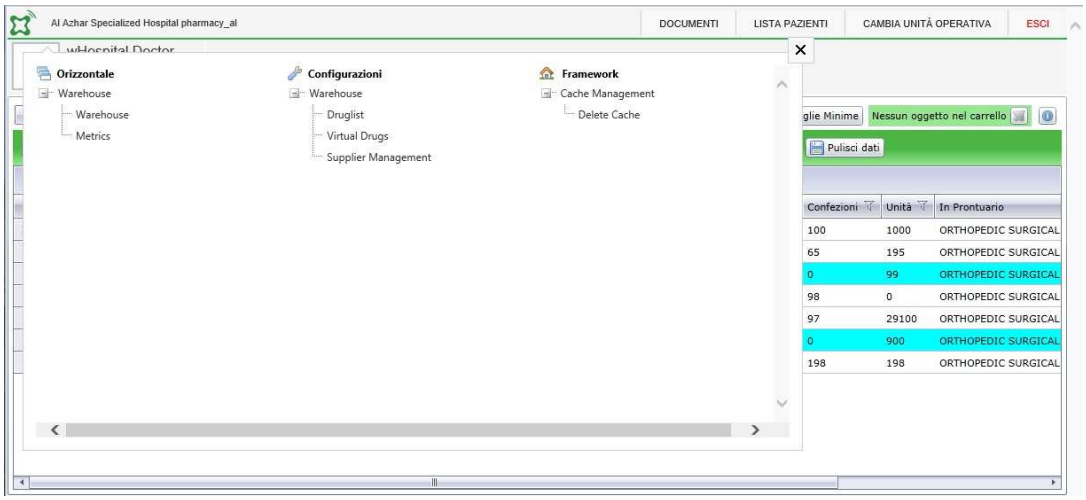


Image 62 – wHospital menu

As explained in chapter two, wHospital provides a view of data from three different point of view: centered on the electronic health record of a patient, horizontal to all records of an operational unit and at a configuration level for the modules themselves. Then the user interface is divided into modules that implements different features. PharmaHosp does not implements electronic health record modules since it whole work paradigm is not based on a single patient. Its core modules are two horizontal ones:

- Stock Management: it implements the view on the drug store warehouse, all the requests and deliveries related to operational units and all the orders and deliveries from suppliers.

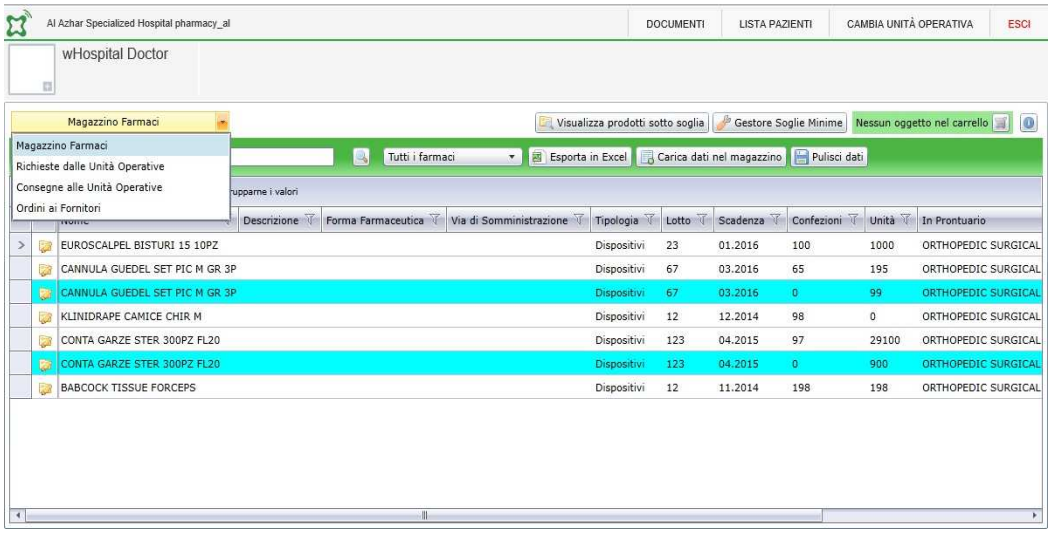


Image 63 – PharmaHosp stock management

- Cart Management: it is the portion developed for the homecare scenario. It is a specific view on the status of items left at a patient place and allows to manage this products by administrating them, throwing them away or returning them back to the drug store.

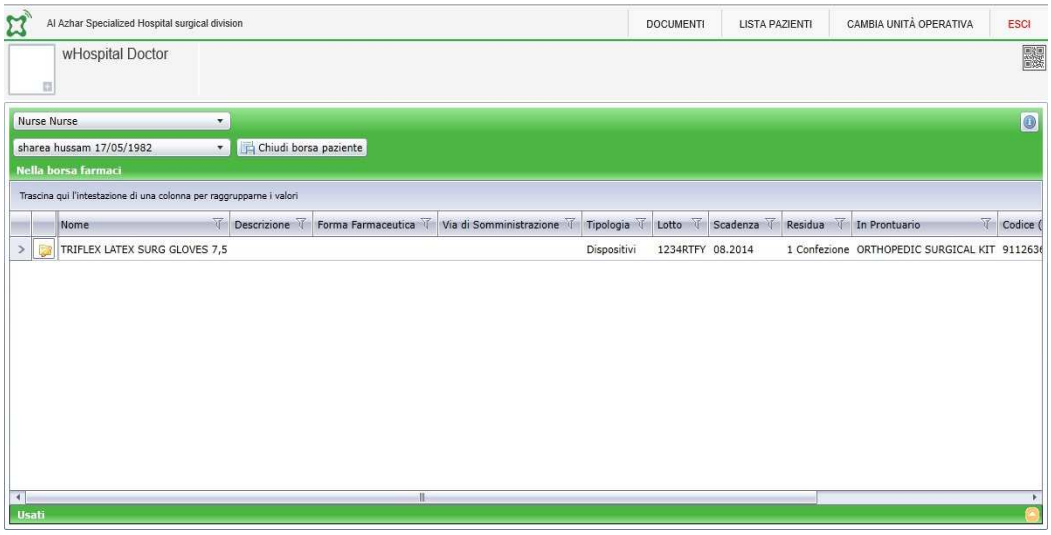


Image 64 – PharmaHosp cart management

- Metrics: it is a module that provides statistical data about usage of the warehouse and the operational units' cabinets. Focus the attention on values and volumes of products inside the structure.

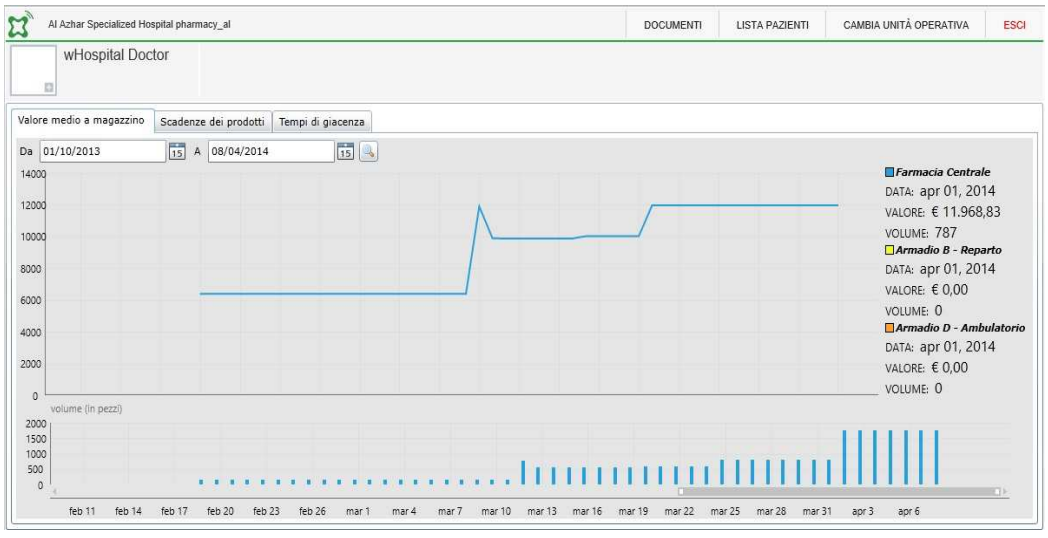


Image 65 – PharmaHosp metrics

This two modules are sided by three configuration modules that provide possibility to update the structure of the suite.

- Drugs' list Manager: it provides a fast tool for enquiry and updating the pharmaceutical products database and create lists of drugs to be used both in the PharmaHosp suite and in the other suites of wHospital.

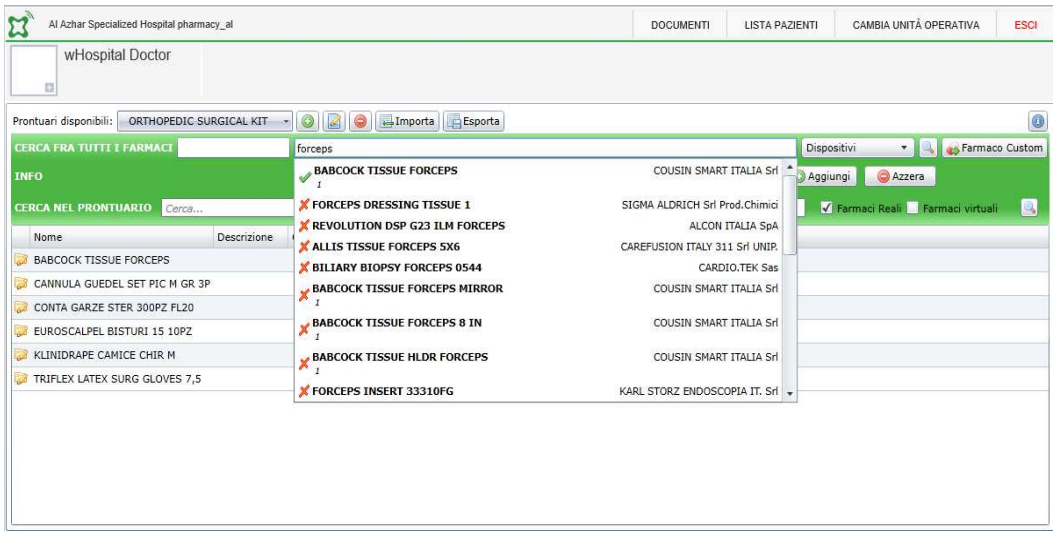


Image 66 – PharmaHosp drugs' list manager

- Cabinets and Carts configuration: it allows the user to create and manage the structure inside its operational unit of cabinets and carts. In the homecare scenario it implements the relation between bags and operators, and between patients and products left at their home.

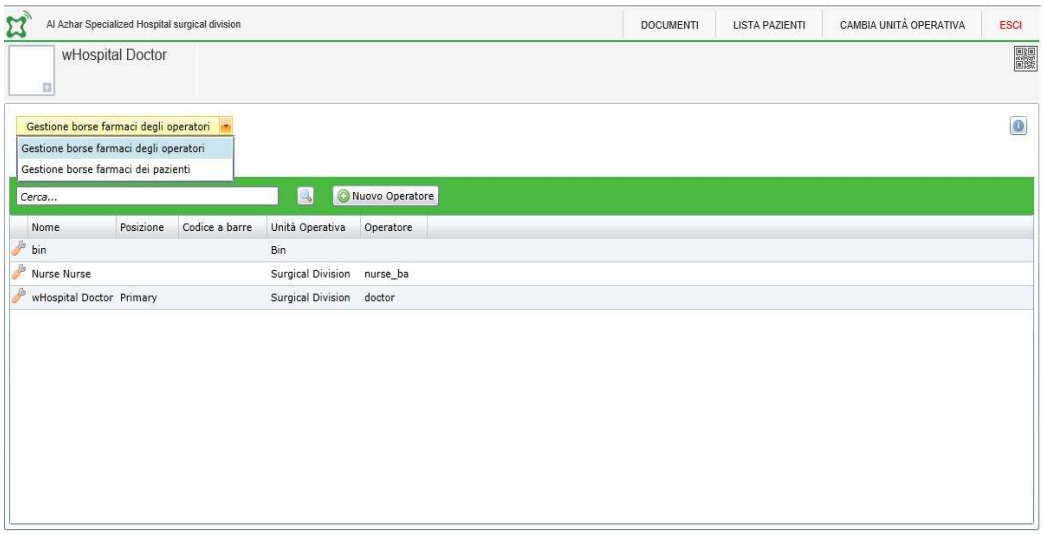


Image 67 – PharmaHosp cabinets and carts configuration

- Supplier's records: it simply manages the suppliers of the drug store. It will be completed by implementing the possibility to add price listings of products handled by the specific supplier.

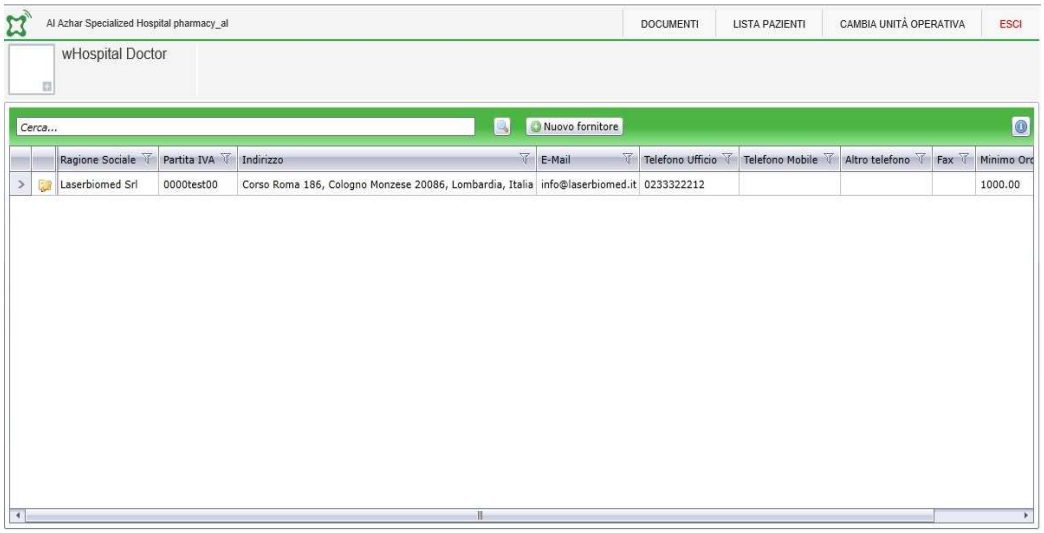


Image 68 – PharmaHosp supplier's records

5.2.5 Testing Phase

Due to time constraints in readying the product for actual demonstrations to customers testing began with developing in later stages. Before deploy on the company's test machine, tests were focused on system and integration testing in order to guarantee a successful installation inside the wHospital framework. Consequently, an alpha acceptance testing, conducted by me and my supervisor, was held inside the company. Once developing reached its latest stages it has been installed on a client real machine for a beta acceptance testing and trial that took place at Casa di Cura del Policlinico di Milano. The black box approach was used for both testing phases since the objective was focused on finding issues in the natural flow of the product. This implied testing the user interface directly looking for exceptions risen by implementation mistakes and moments of the interactions between the actors of the system in which the process was unclear or to refine.

It was not possible to put in place a unit testing phase during the development process due to time constraints.
This rise the issue of developing more stable software in a longer time span or developing something that implements the client requests and eventually perform maintenance on the product once it is taken for test runs and formal trials.
Obiously the first option would be preferable but it is often impossible to apply in order to produce something in strict time periods.
Therefore it becomes essential an approach to developing and code writing that avoids as possible mixing different phases of the process.
For instance, even if the layers separation is quite evident between database, business logic and application logic it is easy to blur these borders. The search process is an example: it would be possible to implement it all at database level but since we developed different procedures to obtain certain data, some of the logic is elevated at business level.
The most important separation has to be kept between server operations and client ones, Silverlight enables to manipulate data more than an ASP.net or web interface, thus risks to retrieve data and manipulate in the client is possible but to avoid. This also enables all the logic and structure to be reusable with very few adaptation to a new user interface in another technology, keeping the suite adaptable to different usages.
The reduced possibility to test the applicative before releasing to clients implicates to develop software with the following features:

- Low rigidity: changes in portions of code does not affect, or affect lightly other parts of the application.
- Flexibility: easiness in modifying processes and behavior of certain portions of code.
- Low replication: avoid same code copied around the software but create ad hoc objects that everyone references or extends.

- Reusability: it is a consequence of low replication, code is reused for other features, eventually by using interfaces or inheriting classes.

This helps during maintenance operations that will be needed even more due to this lack of testing, by reducing the time needed for corrections and by raising the possibility to adapt to new requests.

5.2.6 Production Phase

Production phase is beginning right now with the PharmaHosp suite being installed in the versions of wHospital™ licensed to Fondazione Maddalena Grassi homecare structure and Casa di Cura del Policlinico di Milano healthcare one.
Both systems are not yet in a fully operational phase.
In the first structure the process actually in progress is the creation of lists of drugs representing their dispatching and validation logic. Next phases will include first load of the drug store and beginning moving products with two operators that will be enabled with the technology needed in this scenario that requests mobility.
They will be provided with tablets, supporting Silverlight, and begin reporting accesses to patient houses and administration of products.
The second structure began testing the dispatching process. Initially by moving products inside and outside the warehouse of the drug store towards operational units. It will begin to run the full process once they will understand the therapy processes they need to implement.

6. DEBATE AND CONCLUSIONS

The main objective of this research was to design and develop an application for management of pharmacies inside healthcare providers. Reaching this goal required not only the typical feasibility and requirement examination but also an exhaustive study of the specific domain of application of the project due to the sensible and heterogeneous quantity of information available.

PharmaHosp reaches this goal by being not only a normal management software but also by taking into consideration the pharmacology aspects of the process.

The enormous quantity of data available about drugs and medical devices lead to the implementation of a structured database query-able for information about single items in the system and for the history of all operations executed inside the pharmacy.

The interrogating the set of data highlighted the challenge of successfully and quickly retrieve information from it. The queries realized use a ranking system in order to weight the importance of a hit, based both on the relevance of the field examined and the history of products inside the drug store. In addition, it allows a refining process of search results, as time passes, by analyzing the way users searched on the system. For example, if a user searches mostly for drug name and posology rather than for shapes or producers it is possible to boost up those scores in order to increase the hit ratio.

A second challenge was to provide a rather intuitive and user friendly search form in order to avoid having to set up a list of parameters before being able to actually obtain results. The system can search by means of three different codes usually on pharmaceutical products: AIC (Ammissione In Commercio, stands for admission into commerce), EAN (European Article Number) and REF (which is a production reference code) of which only two are univocal and only AIC is given for every product. It can also search for name, active principle, drug's

shape, producer, posology and quantity inside a box. The final step was to separate codes from free-text in order to obtain a fast search by code but also being able, if needed, to mix codes and free-text searches based on the characteristics of the product.
The results of this approach during tests had been quite satisfying since it gave the possibility to find with a good precision the required object. An open issue is related to the lack of uniformity in terms provided from the data source: when searching for medical devices there is no pattern for words describing the item, it can be an adjective in Italian, in English or a contraption of the term. As a consequence searches are not uniform with respect to the description searched. Possible solutions to address the problem can be the introduction of a translator that iterate search for synonyms of a term or requesting the data source to provide more uniform data.
A brief study of databases from competitors of FarmaData™ has been realized, CodiFa™ represents a possibility since preliminary studies shows better defined way in handling medical devices (Codifa, 'Documentazione tecnica').
Regarding the pharmacy's management the interesting and innovative result is the possibility not only to keep trace of materials' handled but also to achieve a complete monitoring of objects in the system by grouping them not only by code and name, but also by batch and expiration date. The implementation of this more fine granularity allowed different features from alerting the user if something is going to expire, to suggest the consumption of a drug closer to its expiration date, to know exactly where a batch is located in a complex environment such as a hospital: for example when different boxes are scattered among operational units and some are still in the pharmacy. One issue, emerged when visiting real structures, was the difficult communications between the pharmacy and the operational units caused by the fact that the two layers talks a similar but rather different language. The first works with drugs with the perspective of the

administrative and logistic world and the latter uses a prescription point of view. PharmaHosp is one of the first software in the segment that, by being integrated in wHospital™, realizes a strong connection of this two worlds acting in this multi-user healthcare environment.

The application is able to keep the state of each storage site from pharmacy to operational unit cabinets and ward carts. It can also maintain detailed control of every ongoing flow of each considered item.

In order to achieve the required level of security, it uses the paradigm of roles and operations' granted that wHospital implements. By these means, it is possible to associate specific operations only to a specific set of users of the system, for instance in a warehouse there will be three main actors: a worker, a manager and a pharmacist.

A possible limit of the implementation can be the perspective used in designing the search process. In fact, though it can also be a point of strength in this specific situation, the starting point of the development was a study of "how the user actually proceeds" and not a theoretical approach of the problem. That can become, in a different usage scenario, a possible limitation. Helps lowering the rigidity of the implementation the possibility to be able to tune the weight of the search terms to refine the perspective of the searches.

Finding a precisely way to trace items rose the problematic argument of deciding whether to implement the real fine granularity of the single box, if not of the single unit of administration, or the rougher but still quite meaningful granularity of the batch and expiration date for each product. The decision came by studying real cases. Even though systems that can work with the single dosage are in place, it appeared evident that managing the single unit since the entry point in the system was a prohibitive task, if not unsustainable, both for the system and for the warehouse worker, who would be obliged to tag every single unit or blister in stock. In addition, the system grants the possibility to transform boxes into units at every step of the logistic

process. Hence it is possible both to manage specific items as a unit from the entry point, for instance catheters which usually came in boxes of hundreds and go to wards in order of tens, or to handle them as boxes until the point in which it is needed to use the single dose, as what happens when administration of a drug from the cart to a patient happens. This choice does not limit the withdrawal process since directives from AIFA™ are never for single boxes but for entire batches with specific expiration dates.

The PharmaHospice modality, suitable for homecare providers, is the most innovative feature since it plans on working in mobility scenarios and once it is integrated with the agenda and health record modules will provide a fully operative handheld wHospital to operators, allowing them to report back accesses to patients on the go rather than in the office.

Further developments of PharmaHosp are headed towards analysis and data mining of the information collected during usage in order to bring consistent improvements on handling and logistic and preventing wasting of materials.

At a development level, it will be connected to the therapy modules and prescription lines of wHospital. This will allow the system to determine immediately, studying the planning of the week, how each cabinet must be refilled and consequentially how to efficiently move around products inside the structure and when to order new ones. By having a well-defined knowledge of the position of each drug it will also be possible to investigate the very appealing scenario of re-usage and recycling of drugs, for instance documenting all the process to recover a blister of a not completely used box left to a patient at homecare. Once usage will have defined a stable interface it will be redesigned in a more mobility-like technology such as HTML 5 with jQuery or as a Windows 8 App, suitable also for ARM architectures.

A technology improvement could also be passing from barcodes to RFIDs that will make the update of the state of each storage site more

precise and less human dependent. Lastly, the search libraries can be made available also for external users. wHospital implements a library which provide the possibility to call methods exposed by a configuration file with routes. One feature of wHospital is the ability to integrate and interface with other software and provide intercommunication, hence exposing PharmaHosp methods will open scenarios such as interfacing with administrative tools.

Actually, PharmaHosp is deployed and in testing phase in three structures in Milan:

- Casa di Cura Privata del Policlinico™, testing PharmaHospital, the hospital-like scenario.
- Fondazione Maddalena Grassi™, exploiting PharmaHospice, the home care scenario.
- Associazione Il Mantello™ Onlus, using the PharmaHospital modality to focus its efforts in evaluating the aspects of re-usage and recycling of pharmaceutical products.

7. BIBLIOGRAPHY

- AGENZIA ITALIANA DEL FARMACO, '*L'autorizzazione all'immissione in commercio*', http://www.agenziafarmaco.gov.it/it/content/l%E2%80%99autorizzazione-all%E2%80%99immissione-commercio [Accessed 02/04/2014]
- AGENZIA ITALIANA DEL FARMACO, '*La tracciabilità del farmaco*', http://www.agenziafarmaco.gov.it/it/content/la-tracciabilit%C3%A0-del-farmaco [Accessed 30/11/2013]
- AGENZIA ITALIANA DEL FARMACO, '*Procedura Nazionale (AIC)*', http://www.agenziafarmaco.gov.it/it/content/procedura-nazionale-aic [Accessed 22/02/2014]
- AGENZIA ITALIANA DEL FARMACO, '*Rapporti OsMed – L'uso dei farmaci in Italia*', http://www.agenziafarmaco.gov.it/it/content/rapporti-osmed-luso-dei-farmaci-italia [Accessed 02/04/2014]
- BOLCHINI D., MAINETTI L., PAOLINI P., '*Progettare siti web e applicazioni mobili*' 187, McGraw-Hill (2006)
- CARTER J., '*Database Design and Programming with Access, SQL, Visual Basic and ASP*' chapter 2, McGraw-Hill (2002)
- Codifa, 'Documentazione tecnica', http://www.codifa.it/cont/codifa-aziende-ict/documentazione-tecnica/48/documentazione-tecnica-100all1.pdf [Accessed 01/04/2014]
- DOULAVERAKIS C., NIKOLAIDIS G., KLEONTAS A., KOMPATSIARIS I., '*Panacea, a semantic-enabled drug recommendations discovery framework*', J Biomed Semantics (06/03/2014)

- FAKHROUTDINOV K., '*The Unified Modeling Language*', (2009) http://www.uml-diagrams.org/activity-diagrams-examples.html [Accessed 02/04/2014]
- FARRISH S., GRANDO A., '*Ontological approach to reduce complexity in polypharmacy*', AMIA Annu Symp Proc, eCollection 2013 (16/11/2013), 398-407
- FOWLER M., RICE D., FOEMMEL M., HIEATT E., MEE R., STAFFORD R., '*Patterns of Enterprise Application Architecture*' chapter 6, Addison-Wesley (2003)
- GHEZZI C., JAZAYERI M., MANDRIOLI D., '*Ingegneria del Software: fondamenti e principi*', Pearson Italia S.p.a. (2004)
- HANNA J., JOSEPH E., BROCHHAUSEN M., HOGAN W.R., '*Building a drug ontology based on RxNorm and other sources*', J Biomed Semantics Volume 4 (18/12/2013), 44, http://www.ncbi.nlm.nih.gov/pmc/articles/PMC3931349/ [Accessed 12/03/2014]
- LARMAN C., '*Applicare UML e i Pattern: Analisi e progettazione orientata agli oggetti*', Pearson Prentice Hall (2005)
- MARICK B., '*A Manager's Guide to Evaluating Test Suites*', Testing Foundations (19/07/2011)
- MINISTERO DELLA SALUTE, '*Monitoraggio della distribuzione dei medicinali in Italia*', http://www.salute.gov.it/portale/temi/p2_6.jsp?lingua=italiano&id=3116&area=tracciabilita%20farmaco&menu=produzione [Accessed 02/04/2014]
- SMITH B., '*Basic Formal Ontology*', National Center for Ontological Research (18/08/2012)
- SMITH B., '*What is an Ontology and What is it Useful for?*', National Center for Ontological Research
- U.S. NATIONAL CENTER FOR BIOTECHNOLOGY INFORMATION, '*UMLS® Reference Manual*' (05/09/2009), www.ncbi.nlm.nih.gov/books/NBK9679/

[Accessed 02/04/2014]

- U.S. NATIONAL LIBRARY OF MEDICINE, '*RxNorm Technical Documentation*' (05/08/2013), www.nlm.nih.gov/research/umls/rxnorm/docs/2013/rxnorm_index_2013-3.html
  [Accessed 02/04/2014]

APPENDIX A

PharmaHosp
Installation and User Manual

1. Setting up wHospital for PharmaHosp

When installing PharmaHosp for the first time, a few preliminary steps directly on database are necessary.
Assuming to start from the most recent release of the pharmaceutical database the following steps needs to be taken (it also exists a script that follows these steps):

- Defining the MacroTypes in the table wH_RealDrug.DrugMacroType, this table creates more general types above the default drug types, for instance it can create the more general concept of "drug" rather than "drug for hospitals" or "ethical drug".
- Take from the wHospital configuration table wH_Conf.Ou the operational units that will be involved in PharmaHosp environment and define them inside wH_RealDrug.OperationalUnit. Pay attention in copying the exacts presidium codes since they are essential to the system.
- Define a default drug supplier in table Stock.DrugSuppliers, this step can be implemented also when actually defining the records for suppliers in the respective configuration module but until there will be at least one supplier the rest of the suite will be unusable.
- Insert in Stock.DrugStore the name of the operational unit acting as drug store.
- Define the back structure of bins that allows dismiss of items: firstly define a row in wH_RealDrug.OperationalUnit with a generated Guid and the field IsBinOU set to true. Then insert a row in Stock.DrugCabinet with the Guid previously defined and the id of the previous step. Lastly, insert a row in Stock.Cart with the same logic used.

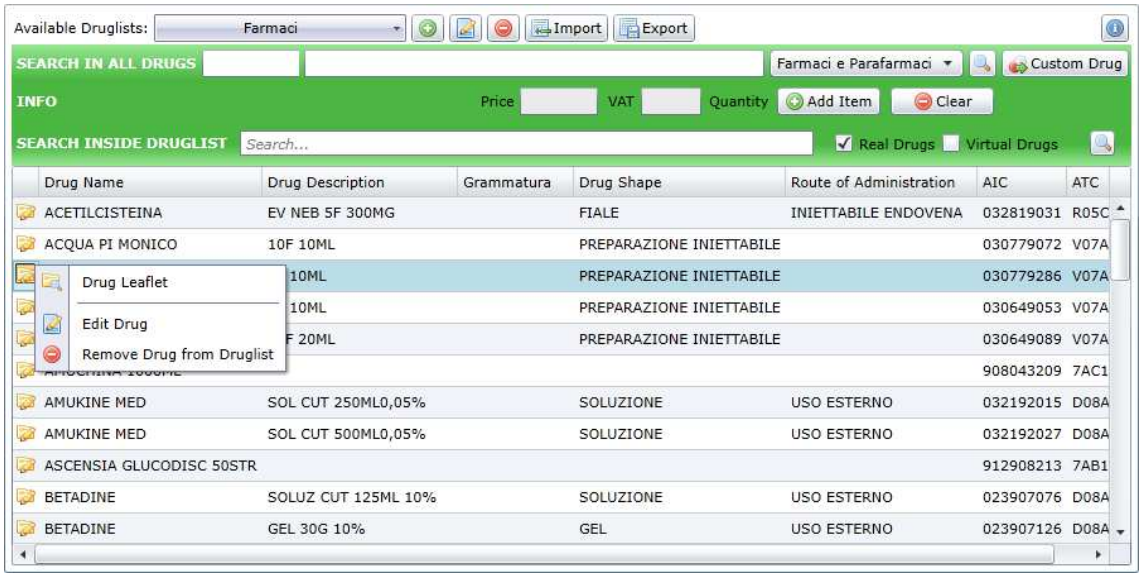
Once defined this preliminary steps, One further configuration one is required still on database level (modules for this part of the procedure is in development).

- Insert in Stock.OUSetting the actual settings desired for the operational units acting in the system.
    - OUGuid reference to the guid in wH_RealDrug.OperationalUnit.
    - IsBatch and IsDueDate work together defining if the operational unit works by means of batches and expiration dates.
    - IsDrugStore sets which unit acts as drug store.
    - HasCabinetManager defines if the structures needs the strict relation between operators and items, set if false for PharmaHospital mode and true for PharmaHospice mode.
    - CanCreateDrugStore allows the related operational unit to define new drug stores.
    - CanEditDrugs enables editing of pharmaceutical product records.
    - CanMapDrugCodes enables triggering of an alert on ordering items which invites user to fill missing codes for products handled.
    - MacroDrugTypeSettings is an XML field with the custom definition for the operational unit of the macro types. If left NULL will use the wHospital definition.
    - XSLTOrderPrintConf contains the XSLT template used in order to print orders.
    - MustPrintAcceptances defines if upon acceptance of products from operational units or operators a file must be printed.
    - XSLTAcceptancePrintConf is the XSLT template related to the acceptance.

- XSLTBackToStorePrintConf is the XSLT for products returning back to the store
- IsLinkedToAmbAccess is false for PharmaHospital mode and true for PharmaHospice mode since it enables the association between administration and operator accesses to a patient’s house.

2. Configuring PharmaHosp

Druglist Management allows creation of drug's lists and management of pharmaceutical product records.



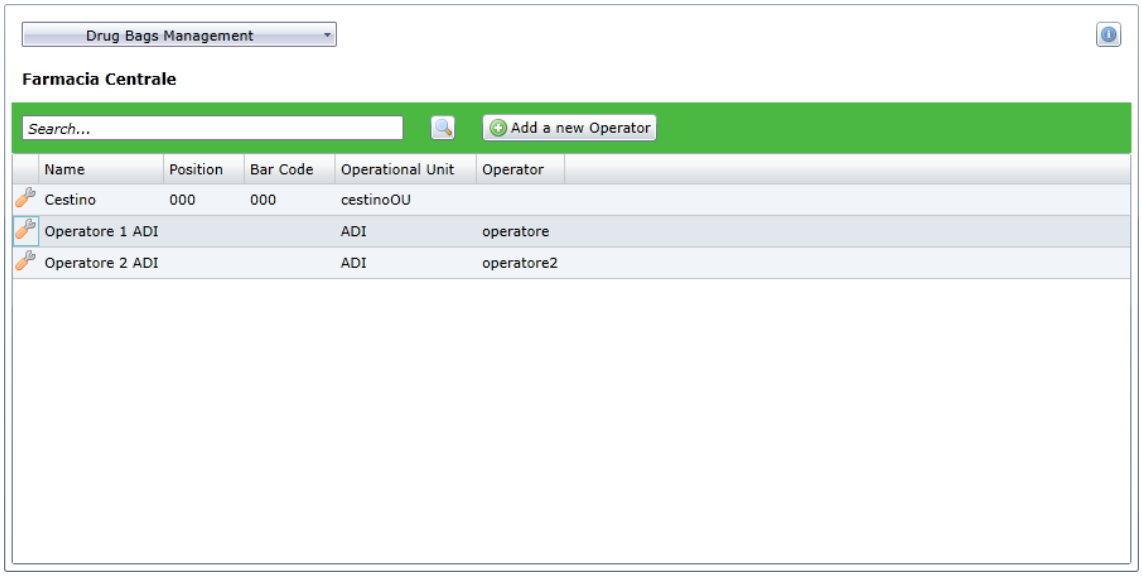
The first row allows selection of drug's lists, creation, editing and deletion. In addition it is possible to export and import data as XML for migration or backup.

The green form represents the search form used throughout all PharmaHosp: the first field text is for codes, the second for free-text and the drop down box is for the macro drug types defined during preliminary steps. The last button allows creation of custom drugs.

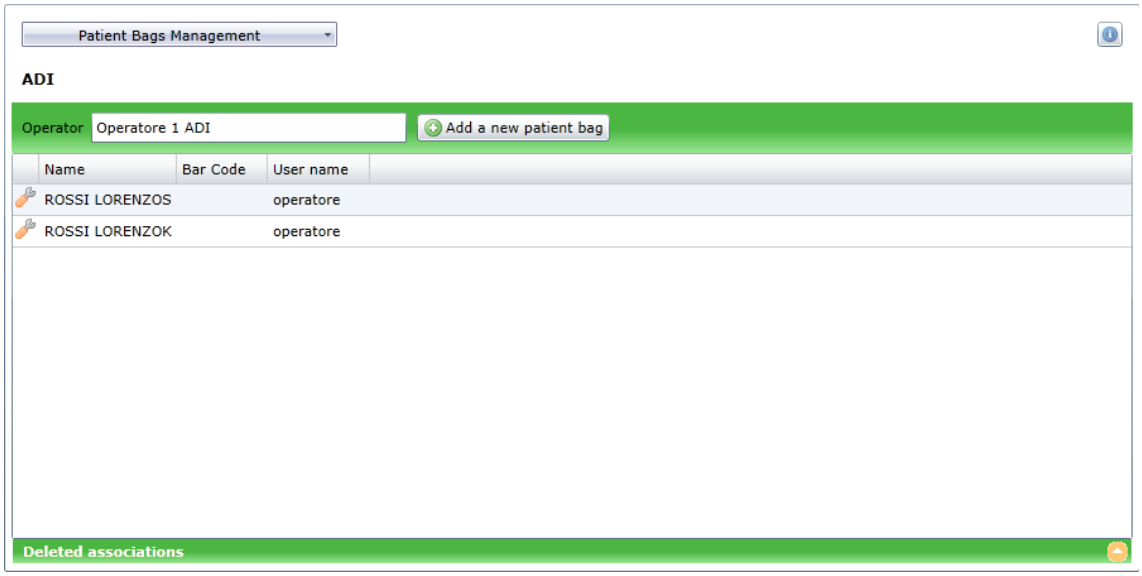
Once selected an item searched its informations will be displayed next to the INFO label and it will be possible to add the item to the selected drug's list.

On items already in the list is possible, by clicking on the folder to view the leafled, edit the drug record or remove it from the drug's list.

Cabinets and Carts Management, or Operator's and Patient's bags Management for PharmaHospice mode allows the creation of the structure internal the operational unit.

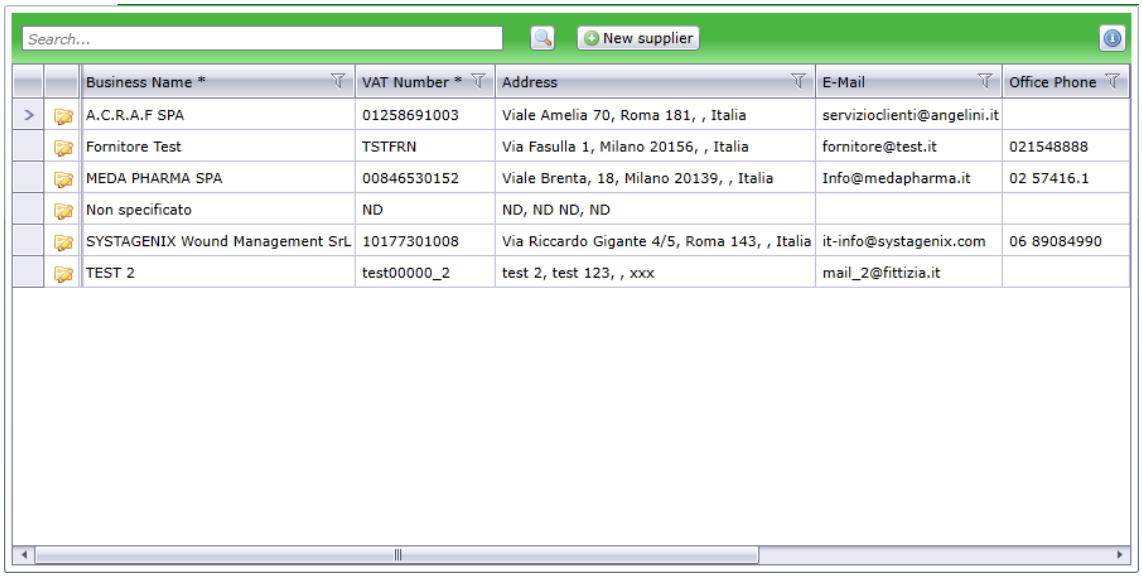


The first page allows to create a new cabinet or link a new operator to its respective bag.



The second page allows by selecting the cabinet or the operator to link the new cart or patient's bag to it.

Lastly, Supplier Management allows to create, edit and update the supplier's records



Business Name *	VAT Number *	Address	E-Mail	Office Phone
A.C.R.A.F SPA	01258691003	Viale Amelia 70, Roma 181, , Italia	servizioclienti@angelini.it	
Fornitore Test	TSTFRN	Via Fasulla 1, Milano 20156, , Italia	fornitore@test.it	021548888
MEDA PHARMA SPA	00846530152	Viale Brenta, 18, Milano 20139, , Italia	Info@medapharma.it	02 57416.1
Non specificato	ND	ND, ND ND, ND		
SYSTAGENIX Wound Management SrL	10177301008	Via Riccardo Gigante 4/5, Roma 143, , Italia	it-info@systagenix.com	06 89084990
TEST 2	test00000_2	test 2, test 123, , xxx	mail_2@fittizia.it	

The modules Stock Manager and Cart Manager requires in wHospital administration to set special framework rights.

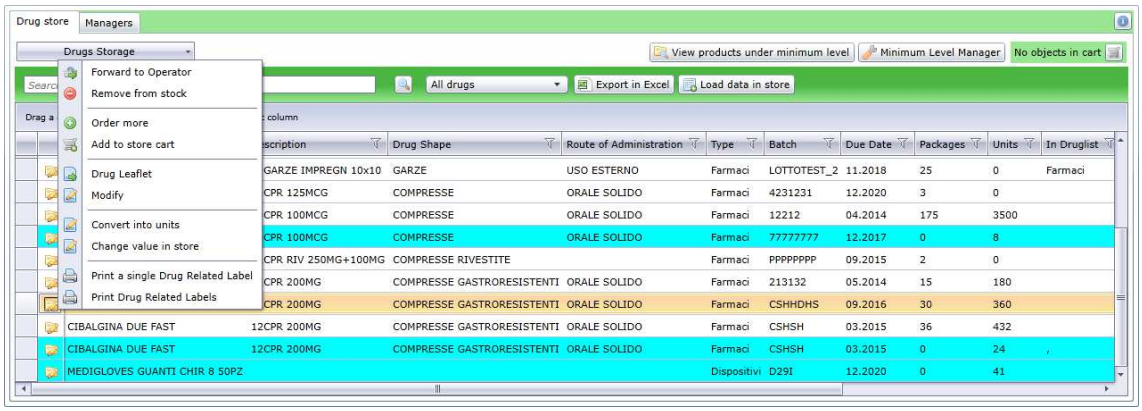
Stock Manager needs the "WarehouseMng" right in order to create orders and validate requests, the "WarehouseOps" right enables reception of documents of delivery and shipping of requests. Cart Manager requires the "Operate" right for users acting as operators, hence requesting for their patients and administration, and the right "Coordinate" for supervisors that manage bags and needs to overview the status of operator's bags.

3. Using PharmaHosp

PharmaHosp manual is described for each intended user of the system.

a) Warehouse Manager

This role has access to the Stock Manager module. Once loaded the main page shows the status of the warehouse filtered by a specific list of drugs.

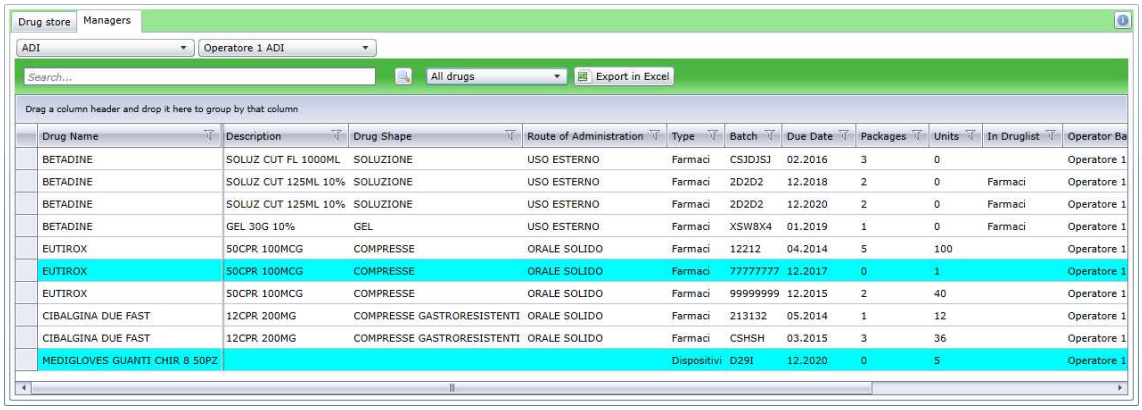


The warehouse manager can view products under minimum levels and manage these respective levels through the buttons in the high right corner. The button next to it is the cart of to be re-ordered items. Items are shown in blue if they are intended as units rather than boxes. For each item a context menu grants a list of functionalities such as:

- Sending to an operator without a specific request
- Remove items from stock
- Re-order items
- Save in the stock cart for an order that will be initiated later
- Read the drug leaflet
- Modify the drug record, this is enabled if the respective OUSetting table record is set to true
- Convert into units allows user to transform boxes into the equivalent number of units
- Change value in store allows manual correction of a value in case of mismanagement

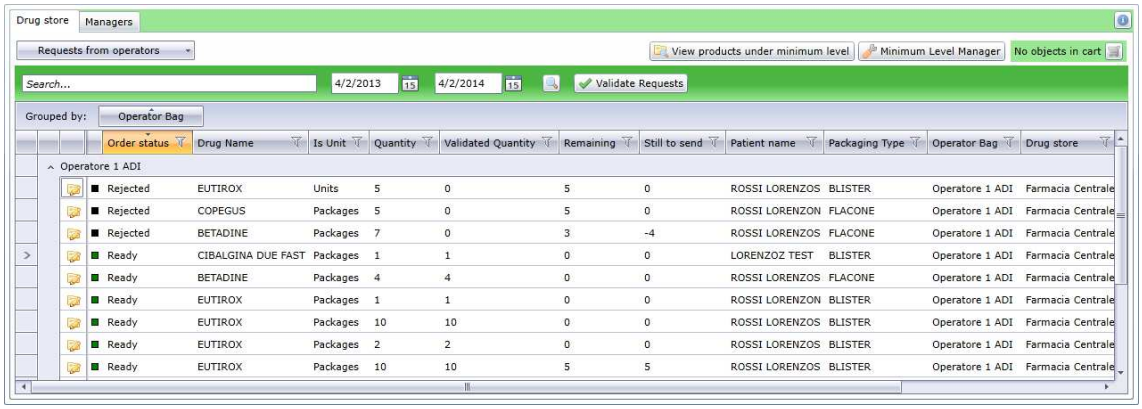
- Print label for single drug and for all drugs prints the label for that exact batch and expiration date

By clicking the tab next to 'Drug store' it is possible to visualize the status of operational unit cabinets, or operator's bags.



Drug Name	Description	Drug Shape	Route of Administration	Type	Batch	Due Date	Packages	Units	In Druglist	Operator Ba
BETADINE	SOLUZ CUT FL 1000ML	SOLUZIONE	USO ESTERNO	Farmaci	CSJDJSJ	02.2016	3	0		Operatore 1
BETADINE	SOLUZ CUT 125ML 10%	SOLUZIONE	USO ESTERNO	Farmaci	2D2D2	12.2018	2	0	Farmaci	Operatore 1
BETADINE	SOLUZ CUT 125ML 10%	SOLUZIONE	USO ESTERNO	Farmaci	2D2D2	12.2020	2	0	Farmaci	Operatore 1
BETADINE	GEL 30G 10%	GEL	USO ESTERNO	Farmaci	XSW8X4	01.2019	1	0	Farmaci	Operatore 1
EUTIROX	50CPR 100MCG	COMPRESSE	ORALE SOLIDO	Farmaci	12212	04.2014	5	100		Operatore 1
EUTIROX	50CPR 100MCG	COMPRESSE	ORALE SOLIDO	Farmaci	77777777	12.2017	0	1		Operatore 1
EUTIROX	50CPR 100MCG	COMPRESSE	ORALE SOLIDO	Farmaci	99999999	12.2015	2	40		Operatore 1
CIBALGINA DUE FAST	12CPR 200MG	COMPRESSE GASTRORESISTENTI	ORALE SOLIDO	Farmaci	213132	05.2014	1	12		Operatore 1
CIBALGINA DUE FAST	12CPR 200MG	COMPRESSE GASTRORESISTENTI	ORALE SOLIDO	Farmaci	CSHSH	03.2015	3	36		Operatore 1
MEDIGLOVES GUANTI CHIR 8 50PZ				Dispositivi	D29I	12.2020	0	5		Operatore 1

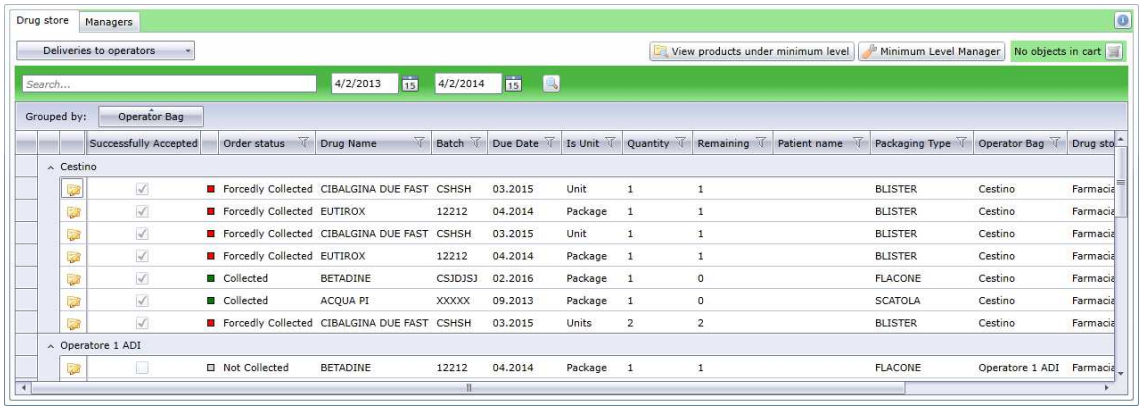
Returning to the drug store tab and using the menu underneath it is possible to view the actual requests arrived to the drug store.



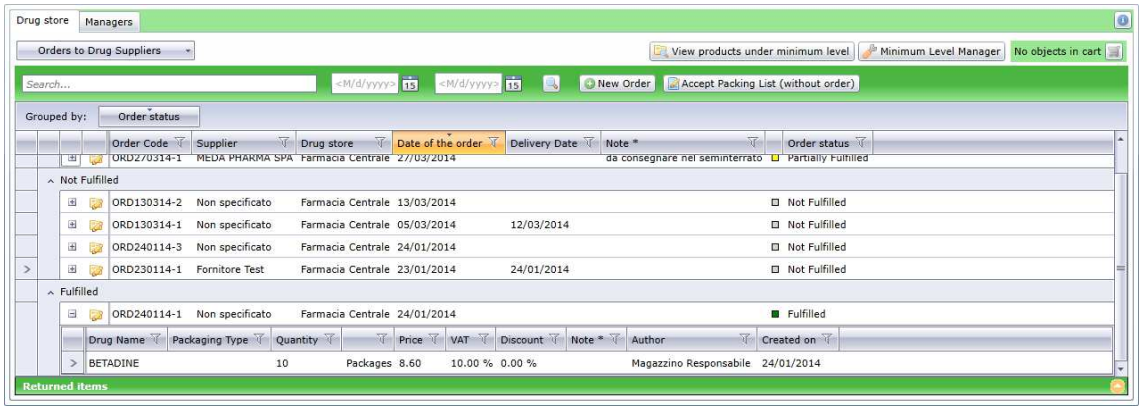
Order status	Drug Name	Is Unit	Quantity	Validated Quantity	Remaining	Still to send	Patient name	Packaging Type	Operator Bag	Drug store
Operatore 1 ADI										
Rejected	EUTIROX	Units	5	0	5	0	ROSSI LORENZOS	BLISTER	Operatore 1 ADI	Farmacia Centrale
Rejected	COPEGUS	Packages	5	0	5	0	ROSSI LORENZON	FLACONE	Operatore 1 ADI	Farmacia Centrale
Rejected	BETADINE	Packages	7	0	3	-4	ROSSI LORENZOS	FLACONE	Operatore 1 ADI	Farmacia Centrale
Ready	CIBALGINA DUE FAST	Packages	1	1	0	0	LORENZOZ TEST	BLISTER	Operatore 1 ADI	Farmacia Centrale
Ready	BETADINE	Packages	4	4	0	0	ROSSI LORENZOS	FLACONE	Operatore 1 ADI	Farmacia Centrale
Ready	EUTIROX	Packages	1	1	0	0	ROSSI LORENZON	BLISTER	Operatore 1 ADI	Farmacia Centrale
Ready	EUTIROX	Packages	10	10	0	0	ROSSI LORENZOS	BLISTER	Operatore 1 ADI	Farmacia Centrale
Ready	EUTIROX	Packages	2	2	0	0	ROSSI LORENZOS	BLISTER	Operatore 1 ADI	Farmacia Centrale
Ready	EUTIROX	Packages	10	10	5	5	ROSSI LORENZOS	BLISTER	Operatore 1 ADI	Farmacia Centrale

Due to its role, the warehouse manager can validate the requests not yet validated by clicking the Validate button.

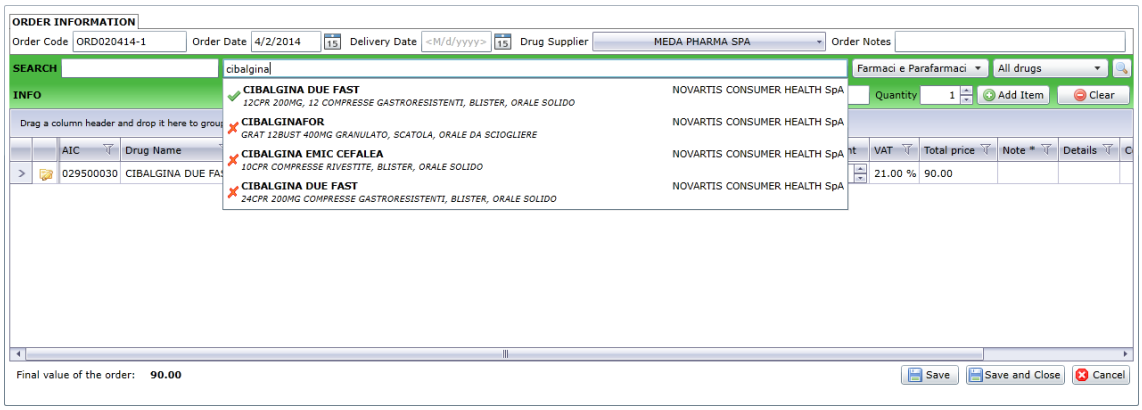
By selecting the voice ‘Deliveries to operational units/operators’ it is possible to see the status of already shipped requests.



The last section of the menu shows the orders sent grouped by orders fulfilled, not fulfilled and partially fulfilled.



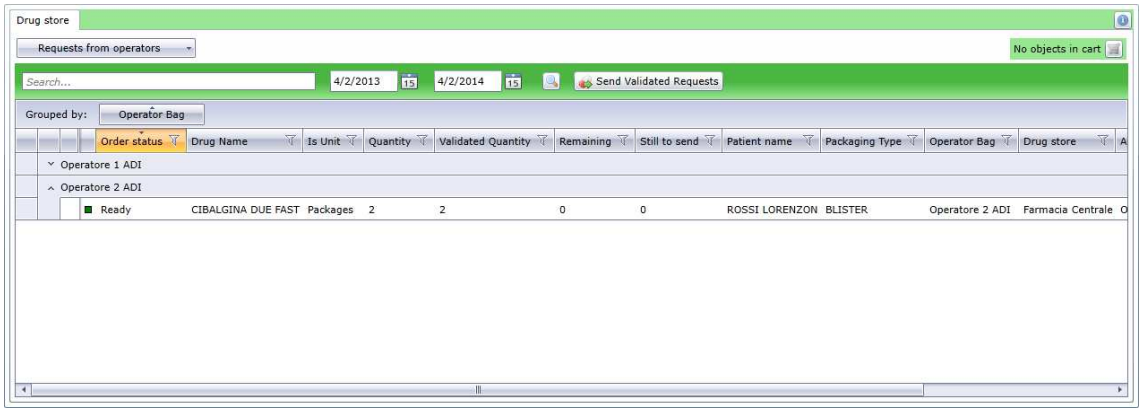
The plus icon for each order allows seeing immediately the products inside the order, the folder next to it allows accepting deliveries, and print the order and view accepted deliveries for already fulfilled orders. The orange icon in the low right corner opens the table of items returned to the drugstore from operational units and allows the manager to accepts them or throw them away. Clicking on the button ‘New order’ opens the page for new orders creation:



This form contains the same searching interface seen in the druglist configuration module. In picture, it is shown that for items already in the database a green V appears whether for items that never entered the structure a red X is shown. When items have been in stock but are not present anymore, a yellow V is shown.
By selecting all the items to order (it is possible to add items to the store cart from the main page and then initiate an order already populated from there) it is possible to assign prices, vats and eventual discounts. The final amount will be automatically calculated.

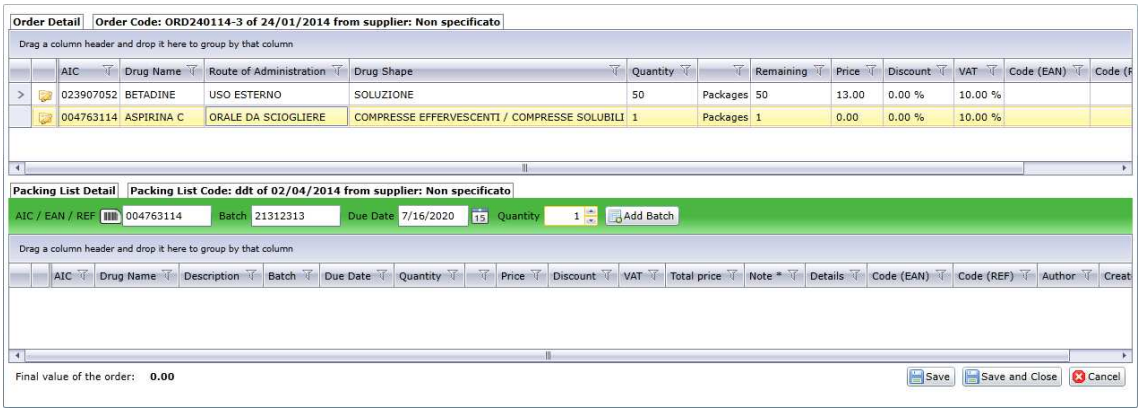
b) Warehouse Worker

The warehouse worker does not possess certain rights such as editing drugs, initiating orders, validating requests, sending items without requests and setting up minimum values.
Its reduced set of operations include sending validated requests and accepting deliveries from suppliers.



As shown in picture, the button for the worker is not 'Validate' anymore but it states 'Send'.

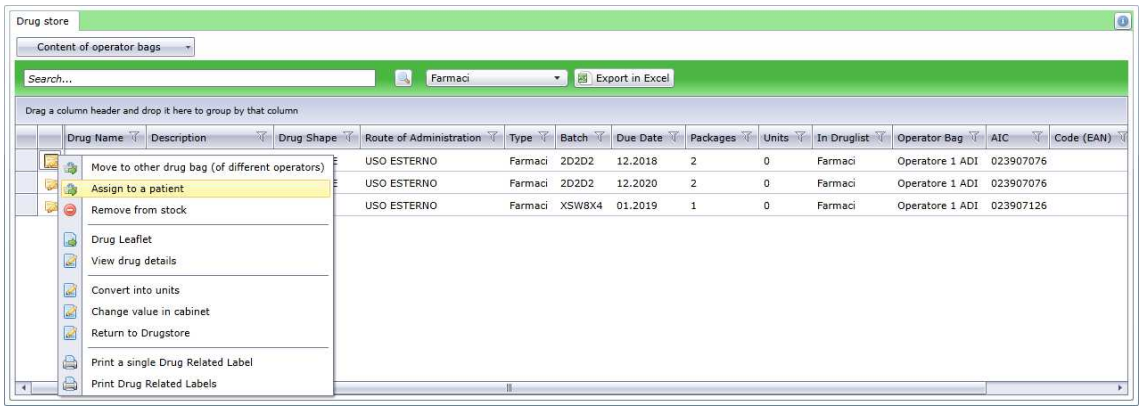
In order to accept deliveries from suppliers it is necessary to find the correct order in the 'Orders from suppliers' page and then click on the folder in the correct row and select 'Accept DDT'. This will open the accepting form.



This form works by barcodes or simple click of the item, this action will select the product and then the user will type batch and expiration date and quantity received. This operation will empty the order and populate the delivery. When the operation is saved the items will enter the drug store.

c) Operator or Nurse

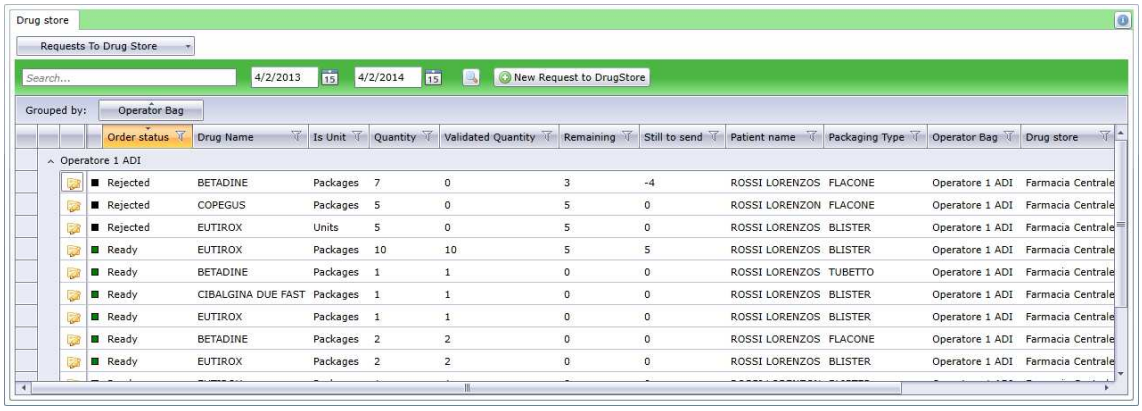
The nurse in the PharmaHospital mode and the operator in the PharmaHospice mode will have access to the Stock Management module. In this case they will not have access to the data of the pharmacy warehouse but they will see the status of cabinets or operator's bags in their referenced operational unit.



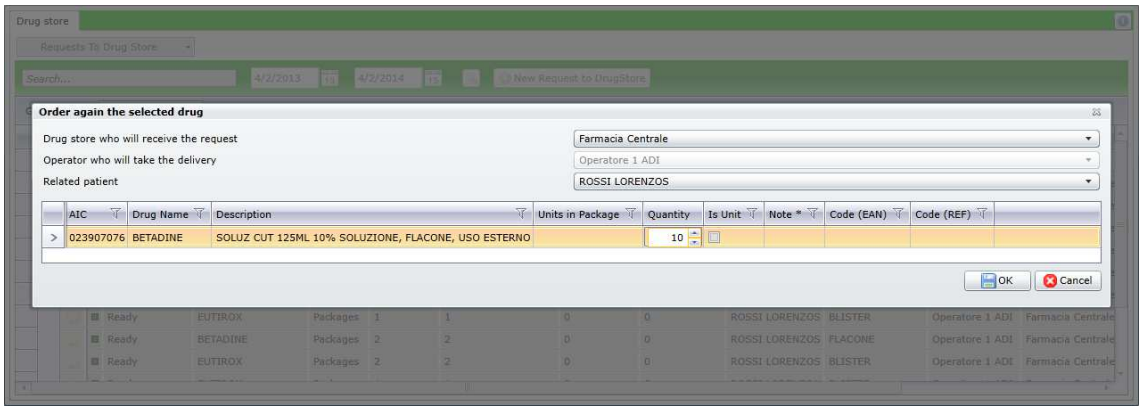
Two new operations are available to this users:

- Assign to patient allows to pass an item from the operator bag to the patient bag at his house, or in the hospital environment to move items to a specific ward cart.
- Return to drugstore will move items back to the store where they will be accepted only when validated.

By selecting 'Requests to Drugstore' from the menu in the left high corner it is possible to see the status of requests, with eventual notes especially when requests are rejected.

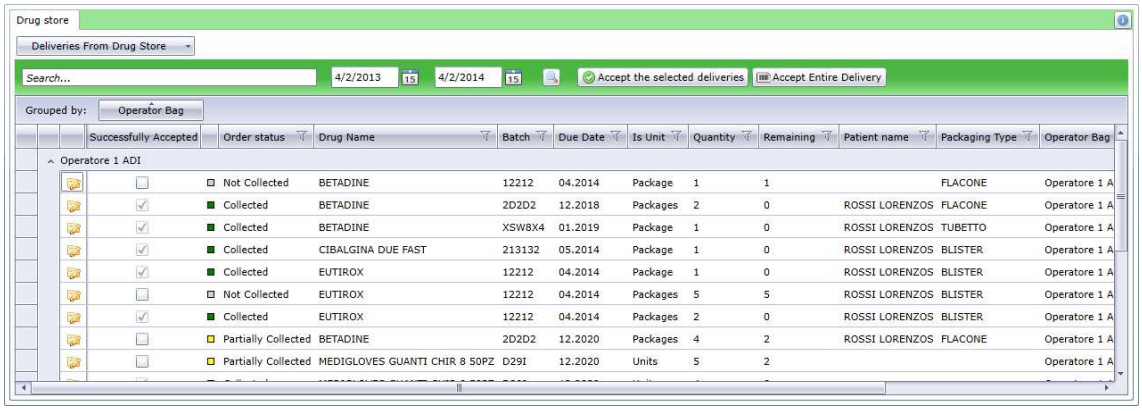


This page also allows users to request new items to the drug store. By clicking the button, a search form will open in which the requested product will be chosen. Then a form will open requesting details about the drug store that will receive the request, the operator sending it and the patient to whom the item is prescribed (this last option is only for the PharmaHospice scenario).



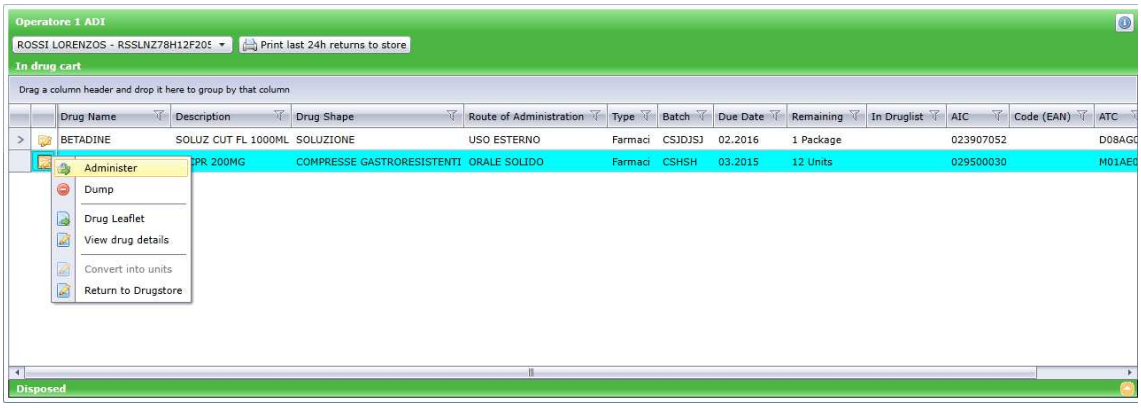
Confirming the request will activate the validation procedure in the drug store unit of wHospital.

The last page of this module allows accepting deliveries from the drug store. Items can be accepted singularly, by selecting some of them or by clicking the select by delivery list button.

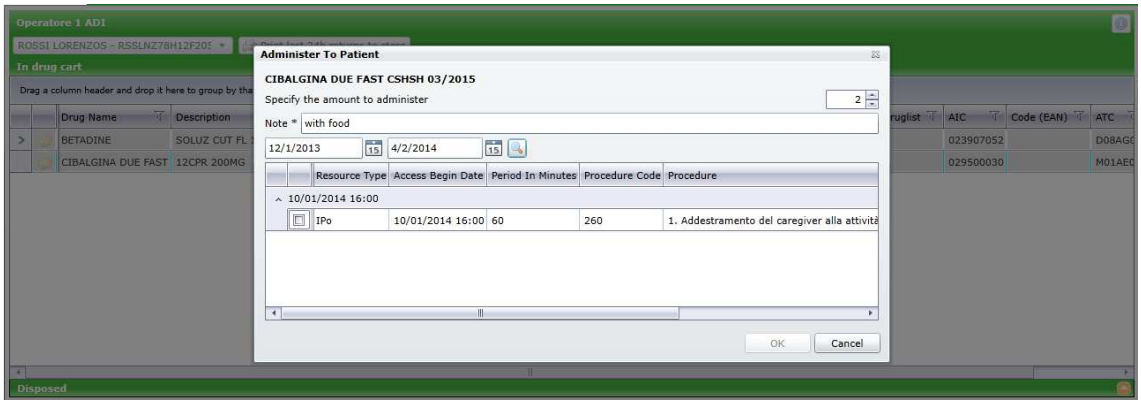


This last button opens a page where by reading the barcode of the printed delivery document it is possible to accept automatically all products of that list.

Operators in PharmaHospice mode can also access the Cart Manager module. This portion of PharmaHosp allows managing of items dispatched at a patient's house and administer them or throw them away accordingly to usage.



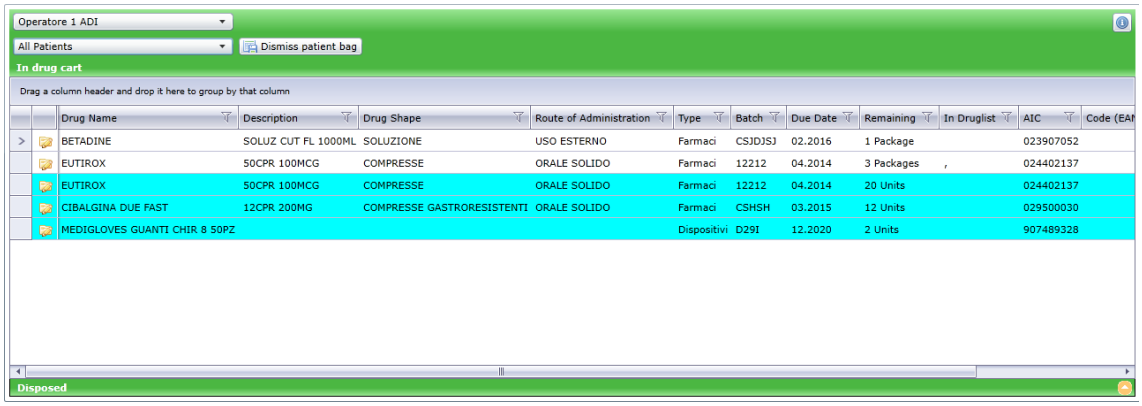
When a patient is selected it is possible to view the items currently at his place, convert boxes into units, view drug details and administer or throw away products.
The button next to the patient name prints the returns to the drug store sent in the last twenty four hours, this in order to allow operators to keep track of their returned items.
The orange icon in the bottom right corner will open the history of administered products.
By clicking the Administer button in the context menu that can be opened by selecting the folder next to the drug name the administration form will open:



This form will specify the number of items administered, eventual notes and will search for accesses to the patient's house scheduled for the current operator.

d) Coordinator

The last role of PharmaHosp, only in PharmaHospice mode, is the Coordinator. This user has usually the possibility to activate patient's bags for operators and to close them once the patient is dismissed. They can access the Cart manager module where they can select a specific operator and then all the product he dispatched to homes or the ones in a specific one.



It will see the module in view mode and therefore will not be able to interact a part for two operations:

- Manually correct a value, obviously it is suggested such operation only after a valuation of the reasons that lead to the error.
- Close a patient bag. This operation dismiss the patient and returns all the left items to the drug store.

APPENDIX B

STAGE DOCUMENTATION

POLITECNICO DI MILANO

SAT Ing. dell' Informazione / MI

Prot. 639/2013

Milano, 23/07/2013

ATTESTATO DI TIROCINIO

Si attesta che MARCO PASINI matr. 764225

studente al Politecnico di Milano Scuola di Ingegneria Industriale e dell'Informazione

Corso di Studi in Ingegneria Informatica

ha frequentato un periodo di tirocinio pratico

nell'ambito del Progetto Stage, concernente:

Sviluppo sistema di gestione base dati farmaci

presso la società Laserbiomed Srl

nel periodo: dal 07/11/2012 al 07/05/2013

avendo come Tutor del Politecnico Prof. SARA RENATA FRANCESCA MARCEGLIA

e come Tutor aziendale Lorenzo Rossi

Il Responsabile SAT
Prof.

DIPARTIMENTO DI ELETTRONICA, INFORMAZIONE E BIOINGEGNERIA

ACKNOWLEDGMENTS (in Italian)

All’interno del laboratorio di Ingegneria Biomedica ho trovato un fantastico ambiente che ha stimolato sia la mia curiosità accademica che intellettuale. Pertanto vorrei ringraziare per primi il professore Francesco Pinciroli ed i suoi colleghi Sara Marceglia e Stefano Bonacina tramite i quali ho imparato a conoscere il mondo della sanità digitale e che mi hanno sempre suggerito e riportato sulla giusta strada nei momenti di difficoltà durante la stesura. Forse mi sono calato nella realtà lavorativa con troppa fretta per dedicarmi al mio ultimo anno universitario con la necessaria dedizione, ma il loro contributo ha fatto sì che questo giorno arrivasse comunque.

Tengo inoltre a ringraziare i miei colleghi di Laserbiomed™ che si sono dimostrati disponibilissimi nel concedermi i tempi necessari per dedicarmi allo studio nonostante un rapporto di lavoro in essere. In particolare Lorenzo Margola, Lorenzo Rossi e Vania Manzelli che mi hanno accettato in quella che per loro forse è più una seconda famiglia che una azienda. A questi aggiungo Simone Viganò che, come mio tutor aziendale deve subire la mia stressante presenza ogni qualvolta la mia ignoranza o inesperienza porta a bloccare lo sviluppo e, come piacevolissima persona allieta tutti i momenti di relax.

La mia famiglia ha giocato un ruolo essenziale nella decisione di andare a Milano per concludere i miei studi universitari e non posso che ringraziarli con tutto il mio cuore per avermi sostenuto nei momenti di crisi e spronato nei momenti di rilassamento. In particolare mia mamma e mia nonna che, nonostante siano state travolte da mille difficoltà, non hanno mai smesso di credere in me, e mio fratello con la sua famiglia che rappresentano in ogni istante un bellissimo modello da seguire.

Una ringraziamento speciale va a mio padre. Nel mio animo spero che tu possegga ancora in cuor tuo, nonostante tutte le sfortune che ti hanno colpito, la possibilità di comprendere questo momento e tutta la mia gioia nel poterti abbracciare oggi. Alcune cose ormai perse mi mancano tantissimo, ma averti qui con noi è la cosa più bella.

Ci vorrebbe un capitolo a parte anche per la mia Lucia: resistiamo nonostante la distanza e tiriamo avanti assieme. Aver concluso gli studi è anche merito tuo che tiri sempre fuori il meglio di me.

Approfitto delle ultime righe per ringraziare gli amici più intimi: sebbene stia temendo quel che mi prepareranno nelle prossime ore o nei prossimi giorni, Paolo e Nicolò meritano una menzione a parte. A loro aggiungo Matteo, Giacomo, Frappy, Anna, Sofia, Braso, Giovanna, Chris, Martino, Dibla, Mary, Galva e Danilo.
Infine ringrazio Nicola e Alessandro che con la loro compagnia non mi fanno mai sentire solo qui a Milano, anche in quest'anno lontano dal mio sport. Questa però è un'altra sfida ancora da vincere, intanto fatemi esultare per questa.