

**PHarmaLife**  
**System Design Document**  
**Version 1.4**



**PHARMA LIFE**

(Ultima modifica in data: 29/01/2022)

Progetto: PHarmaLife	Versione: 1.4
Documento: System Design Document	Data: 29/01/22

### Coordinatore del progetto:

Nome	Matricola
Carmine Gravino	

### Partecipanti:

Nome	Matricola
Alfredo Cannavaro [AC]	0512108651
Eduardo Scarpa [ES]	0512109503
Carmine Fierro [CF]	0512106203
Catello Staiano [CS]	0512106875

<b>Scritto da:</b>	Gruppo G21
--------------------	------------

## Revision History

Data	Versione	Descrizione	Autore
25/11/2021	1.0	Scopo del sistema	Alfredo Cannavaro [AC] e Eduardo Scarpa [ES]
25/11/2021	1.0	Design Goals	Carmine Fierro [CF] e Catello Staiano [CS]
25/11/2021	1.0	Trade-Off & Riferimenti	Carmine Fierro [CF] e Eduardo Scarpa [ES]
25/11/2021	1.0	Definizioni, acronimi e abbreviazioni	Carmine Fierro [CF]
26/11/2021	1.0	Panoramica del progetto	Carmine Fierro [CF] e Alfredo Cannavaro [AC]
26/11/2021	1.0	Architettura del sistema proposto: Panoramica e Decomposizioni in sottosistemi	Catello Staiano [CS] e Eduardo Scarpa [ES]
26/11/2021	1.0	Component Diagram e Mapping Hardware e Software	Catello Staiano [CS]
26/11/2021	1.0	Gestione dati persistenti e Controllo dati sicurezza	Carmine Fierro [CF]
26/11/2021	1.0	Controllo flusso globale del sistema e Condizione limite	Eduardo Scarpa [ES] e Alfredo Cannavaro [AC]

Progetto: PHarmaLife	Versione: 1.4
Documento: System Design Document	Data: 29/01/22

29/11/2021	1.0	Servizi di Sottosistemi	Eduardo Scarpa [ES] e Alfredo Cannavaro [AC]
22/12/2021	1.1	Modifiche al “Servizi di Sottosistemi”	Alfredo Cannavaro [AC] e Catello Staiano [CS]
29/12/2021	1.2	Modifica architettura sistema	Gruppo G21
03/01/2022	1.3	Modifica a ritroso dell’architettura sistema	Gruppo G21
04/01/2022	1.4	Correzioni riferiteci dal Prof. Gravino (dovuta alla consegna del 13DEC21)	Eduardo Scarpa [ES], Carmine Fierro [CF] e Alfredo Cannavaro [AC]
29/01/2022	1.4	Ultima revisione generale	Gruppo G21



## Sommario

1.	Introduzione.....	5
1.1.	Scopo del sistema.....	5
1.2.	Design Goals.....	6
1.2.1.	Trade-Off.....	8
1.3.	Definizioni, acronimi e abbreviazioni.....	9
1.4.	Riferimenti.....	9
1.5.	Panoramica del progetto.....	10
2.	Architetture di Sistemi Simili.....	10
3.	Architettura del Sistema Proposto.....	11
3.1.	Panoramica.....	11
3.2.	Decomposizione in sottosistemi.....	12
3.3.	Mapping Hardware/Software.....	14
3.4.	Gestione dati persistenti.....	15
3.5.	Controllo degli accessi e sicurezza.....	18
3.6.	Controllo flusso globale del sistema.....	20
3.7.	Condizione limite.....	20
4.	Servizi dei Sottosistemi.....	24



## 1. Introduzione

### 1.1. Scopo del sistema

Lo scopo del sistema è quello di realizzare un e-commerce dedicato alla vendita dei prodotti farmaceutici: “**PHarmaLife.it**”.

Sulla nostra piattaforma, ogni utente può registrarsi e creare un profilo personale caratterizzato dai propri dati anagrafici ed alcuni dati personali. Il profilo sarà sempre modificabile successivamente dall’utente. L’esigenza di PHarmaLife.it nasce dal bisogno di facilitare alle persone di acquistare farmaci direttamente da casa senza preoccuparsi di girare in numerose farmacie con l’intento di trovare il prodotto necessario.

Ogni utente avrà una sezione “home” in cui sarà possibile attraverso un menù a tendina visualizzare il catalogo dei prodotti divisi per categoria. Un utente avrà a disposizione una propria lista dei desideri dove potrà aggiungere o rimuovere prodotti ed un proprio carrello che permetterà all’utente di finalizzare l’acquisto dei prodotti che necessita. Inoltre in caso di problemi l’utente può rivolgersi all’Assistenza Clienti inviando un messaggio con all’interno la propria problematica.

La gestione del sito è affidata all’Amministratore che si occuperà di gestire il catalogo dei prodotti, visualizzare la lista utenti, visualizzare le statistiche e gestire l’Assistenza Clienti.

Il sistema progettato è un e-commerce alla quale avranno accesso l’amministratore di sistema e l’utente che si registrerà. Possiamo dividere il sistema due macro-aree:

- Area Amministratore
- Area Utente

Il sistema deve fornire un metodo di autenticazione sicuro in modo che un utente non possa aver effettuato più di un accesso contemporaneamente e che dopo la terminazione della sessione, bisogna reinserire i dati per evitare accessi fraudolenti. Il sistema inoltre dovrà essere facile da apprendere e intuitivo da utilizzare, consentendo una navigazione fluida e permettendo l’utilizzo del sistema anche senza il consulto della documentazione.



## 1.2 Design Goals

I Design Goals sono organizzati in cinque categorie: Performance, Dependability, Cost, Maintenance, End User and Criteria. I Design Goals identificati nel nostro sistema sono i seguenti:

### Criteri di Performance

- **Tempo di risposta:**

Il software deve consentire una navigazione rapida a tutti i tipi di utenti, quindi tempi di risposta minimi nello svolgimento delle funzionalità da esso offerte.

- **Memoria:**

La memoria complessiva del sistema dipende dalla memoria utilizzata per il mantenimento del database.

### Criteri di Affidabilità

- **Robustezza:**

Il sistema informerà l'utente di eventuali input errati attraverso messaggi di errore.

- **Affidabilità:**

Il sistema deve garantire l'affidabilità dei servizi proposti. I risultati visualizzati saranno attendibili. Per quanto riguarda i prodotti, verranno visualizzati esattamente quelli ricercati in base ai vari parametri di ricerca quali categoria, marchi, e ricerca personalizzata tramite un filtro specifico. Ad esempio, per ogni Categoria o marchio ci saranno esattamente quei prodotti con i rispettivi costi. Il processo di login da parte di tutti gli utenti sarà gestito in modo affidabile, assicurando il corretto funzionamento del sistema.

- **Disponibilità:**

Una volta online, il catalogo dei prodotti sarà disponibile per tutti gli utenti.

- **Tolleranza ai guasti:**

Il sistema potrebbe essere soggetto a fallimenti dovuti a varie cause tra cui un sovraccarico di dati nel database.

- **Sicurezza:**

L'accesso ai sistemi sarà garantito tramite l'inserimento di email e password.



## Criteri di Costi

- **Costo di sviluppo:**

E' stimato un costo complessivo di 200 ore per la progettazione e lo sviluppo del sistema (50 ore per ogni membro del progetto).

## Criteri di Manutenzione

- **Estensibilità:**

È possibile aggiungere nuove funzionalità al sistema, dettate dalle esigenze del cliente o dall'avvento di nuove tecnologie.

- **Portabilità:**

L'interazione con il sistema avviene tramite browser, quindi possiamo definirlo portabile. Il sistema è accessibile da qualunque dispositivo, che sia esso mobile o meno, purché abbia un browser installato. Questa caratteristica garantisce la portabilità dello stesso.

- **Tracciabilità dei requisiti:**

La tracciabilità dei requisiti sarà possibile grazie ad una matrice di tracciabilità, attraverso la quale sarà possibile retrocedere al requisito associato ad ogni parte del progetto. La tracciabilità sarà garantita dalla fase di progettazione fino al testing.

## Criteri di Usabilità

- **Usabilità:**

Il sistema sarà di facile comprensione e utilizzo, permettendo di effettuare in modo semplice e immediato le varie operazioni grazie a un'interfaccia intuitiva, di facile comprensione e utilizzo. L'intuitività è garantita in quanto il sistema avrà una buona prevedibilità, cioè la risposta del sistema ad un'azione utente sarà corrispondente alle aspettative.

- **Utilità:**

Il lavoro dell'utente verrà supportato nel miglior modo possibile dal sistema, infatti l'utente compirà le operazioni consentite al fine di ottimizzare il tempo per la ricerca di un prodotto da acquistare.



## 1.2.1 Trade-Off

### Tempo di rilascio vs Funzionalità

Nonostante i tempi di sviluppo ridotti, il team si impegna nel consegnare il sistema nella data di consegna prevista, prevedendo di implementare dapprima le funzionalità necessarie per il corretto acquisto e gestione dei prodotti, ponendo successivamente, tramite opportuni interventi di manutenzione ed aggiornamenti, l'aggiunta delle funzionalità secondarie.

### Prestazioni vs Costi

Per rientrare nel budget a disposizione, il team cercherà di ottenere le migliori prestazioni ma nelle ore-lavoro garantite dal budget.

### Criteri di manutenzione vs Criteri di performance

Il sistema sarà implementato preferendo le performance in modo da facilitare gli utenti nella fase di navigazione della piattaforma web a discapito della manutenibilità.

Di seguito è riportata una tabella che mostra i design goal preferiti nei trade off. Il **grassetto** indica la preferenza.

TRADE-OFF	
<b>Tempo di rilascio</b>	Funzionalità
<b>Prestazioni</b>	Costi
<b>Criteri di performance</b>	Criteri di manutenzione





## 1.3 Definizioni, acronimi e abbreviazioni

### ***Definizioni***

**Amministratore** = Amministratore del sistema

**Piattaforma** = Applicazione Web

**Client** = Componente di una piattaforma web che si occupa di accedere ai servizi e risorse di un'altra componente detta Server

**Server** = Componente di una piattaforma web che si occupa di fornire dei servizi ad un client tramite delle apposite interfacce

### ***Acronimi***

DBMS = Database Management System

GUI = Graphical User Interface (Interfaccia utente)

DBA = Database Administrator (Amministratore del Database)

RUD = Create, Read, Update, Delete

## 1.4 Riferimenti

→ *Libro*: -- Object-Oriented Software Engineering (Using UML, Patterns, and Java) Third Edition

◆ *Autori*: -- Bernd Bruegge & Allen H. Dutoit

→ Documenti:

-- RAD\_PharmaLife – Requirements Analysis Document

+ Slide lezioni del Prof. Carmine Gravino

→ [www.faimed.it](http://www.faimed.it)



## 1.5 Panoramica del progetto

### **Capitolo 1**

Contiene l'introduzione agli obiettivi del sistema, i design goals, i trade-off e un elenco di definizioni, acronimi e abbreviazioni utili alla comprensione dell'intera documentazione.

### **Capitolo 2**

Contiene la descrizione del sistema Corrente, o un sistema simile di riferimento.

### **Capitolo 3**

Contiene la descrizione del sistema che verrà realizzato, degli obiettivi che andrà a realizzare, in cui sarà gestita la decomposizione in sottosistemi, il mapping Hardware/Software, la gestione dei dati persistenti, il controllo degli accessi e sicurezza, il controllo del flusso globale del sistema, le condizioni limite.

### **Capitolo 4**

Contiene la rappresentazione dei servizi dei sottosistemi.

## 2. Architetture di sistemi simili

La *piattaforma* da sviluppare presenta numerose analogie con le varie piattaforme di e-commerce presenti attualmente in rete. Le piattaforme per la gestione di servizi e commerce sono tipicamente basate su Architettura Client/Server, e le tecnologie impiegate per la realizzazione di tale servizio si dividono in due macro categorie:

1. Tecnologie Front-end: insieme di tecnologie che operano lato client al fine di rendere l'interazione con l'utente più fruibile tramite la GUI, le tecnologie messe in campo generalmente sono: html, css, javascript, bootstrap.
2. Tecnologie Back-end: insieme di tecnologie che operano lato Server al fine garantire il perfetto funzionamento della *piattaforma*, come la gestione dell'autenticazione, acquisto di un prodotto, operazioni CRUD per la gestione dei prodotti. Le tecnologie messe in campo possono essere le seguenti: php, python, java e un DBMS per la gestione dei dati persistenti.



## 3. Architettura del sistema proposto

### 3.1 Panoramica

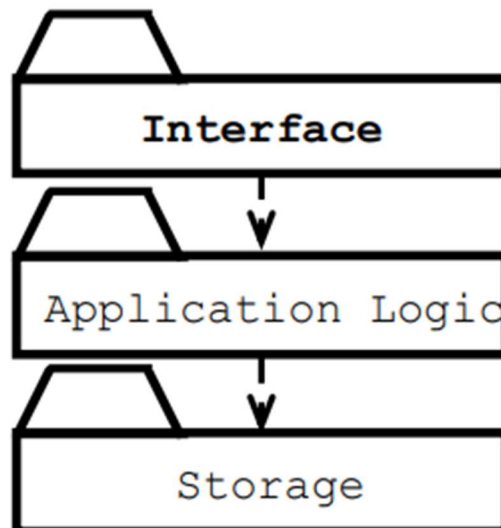
Il sistema che si vuole realizzare rientra nel campo della *Greenfield Engineering*. Il nuovo sistema **PHarmaLife** è un'applicazione web, in locale per motivi di sicurezza, che verrà sottoposto a reengineering con il fine di aggiungere nuove funzionalità e migliorare quelle già presenti. Tutti gli utenti potranno effettuare login e logout; gli utenti avranno la possibilità di registrarsi al sito mentre l'admin verrà "settato" dagli sviluppatori. Il software da noi proposto metterà a disposizione varie funzionalità, a seconda dell'utente che effettua l'accesso. L'utente non registrato potrà visualizzare l'elenco dei prodotti presenti nel catalogo, e per ogni prodotto potrà visualizzare la scheda del prodotto con tutte le relative informazioni, solo una volta effettuata la registrazione al sito sarà possibile concludere l'acquisto, ovviamente, solo dopo averlo aggiunto al carrello e inserire i prodotti nella propria lista dei desideri. L'amministratore del sistema oltre a poter svolgere tutte le attività di un utente qualsiasi è anche responsabile della gestione dei prodotti e dell'assistenza utenti, le principali operazioni che coinvolgono la figura dell'admin sono: inserimento di un prodotto nel catalogo, modifica delle informazioni e eliminazione di un prodotto dal catalogo.

## 3.2 Decomposizione in sottosistemi

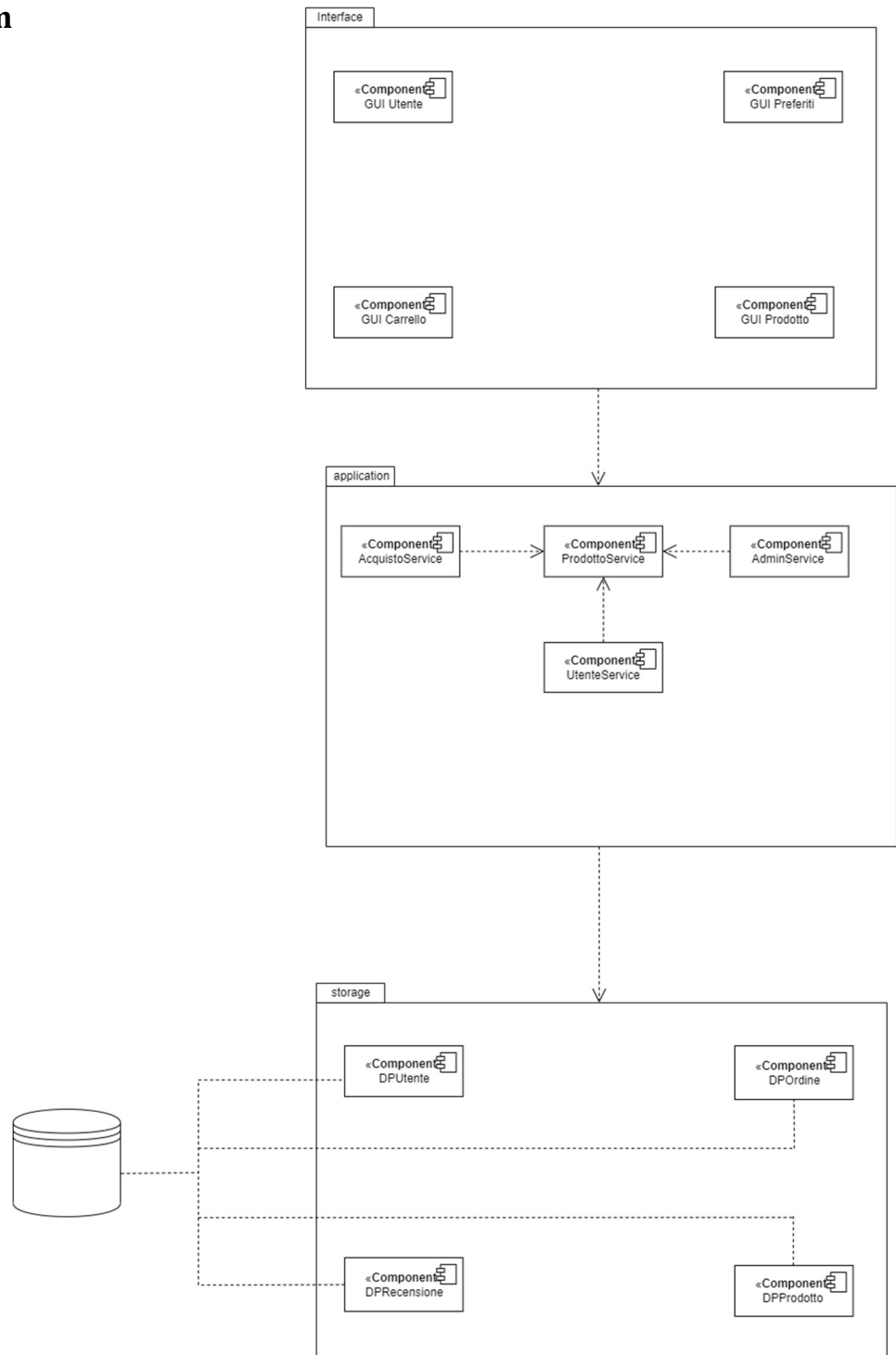
La decomposizione del sistema verrà realizzata seguendo l'architettura **THREE-TIER**.

- **INTERFACE**: sottosistema che si occupa dell'interazione con l'utente.
- **APPLICATION**: sottosistema che si occupa di gestire la logica dell'applicazione.
- **STORAGE**: sottosistema che si occupa della gestione dei dati.

La seguente architettura è stata designata adatta all'implementazione del sistema in quanto le interfacce utente sono soggette a maggiori cambiamenti di quanto possa rispetto agli oggetti del dominio applicativo (*storage*).



## Component Diagram



Lo strato **Application** gestisce 5 sottosistemi:

- Gestione Carrello
- Gestione Prodotto
- Gestione Preferenza
- Gestione Utenza

Lo strato **Interface** gestisce 4 sottosistemi:

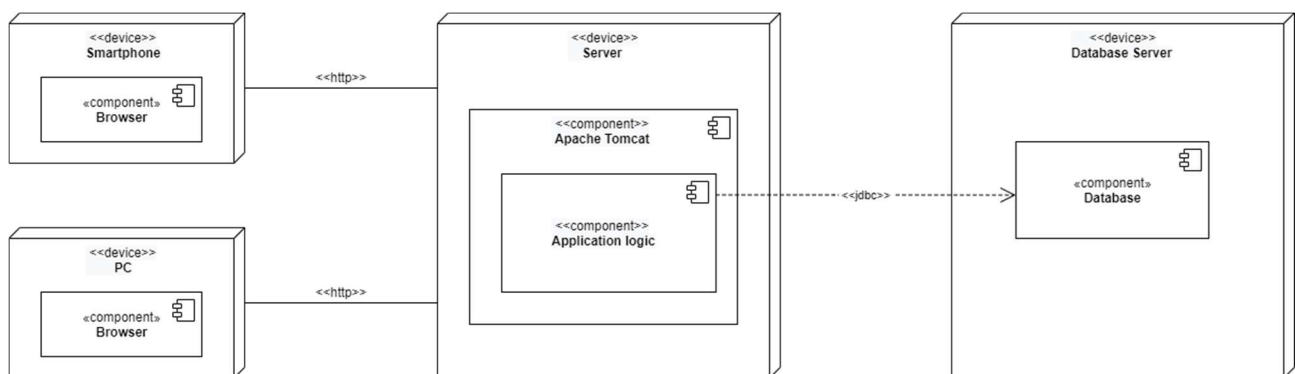
- GUI Prodotto
- GUI Utente
- GUI Catalogo
- GUI Carrello

Lo strato **Storage** gestisce 5 sottosistemi:

- DP Utente
- DP Prodotto
- DP Acquisto
- DP Preferenza

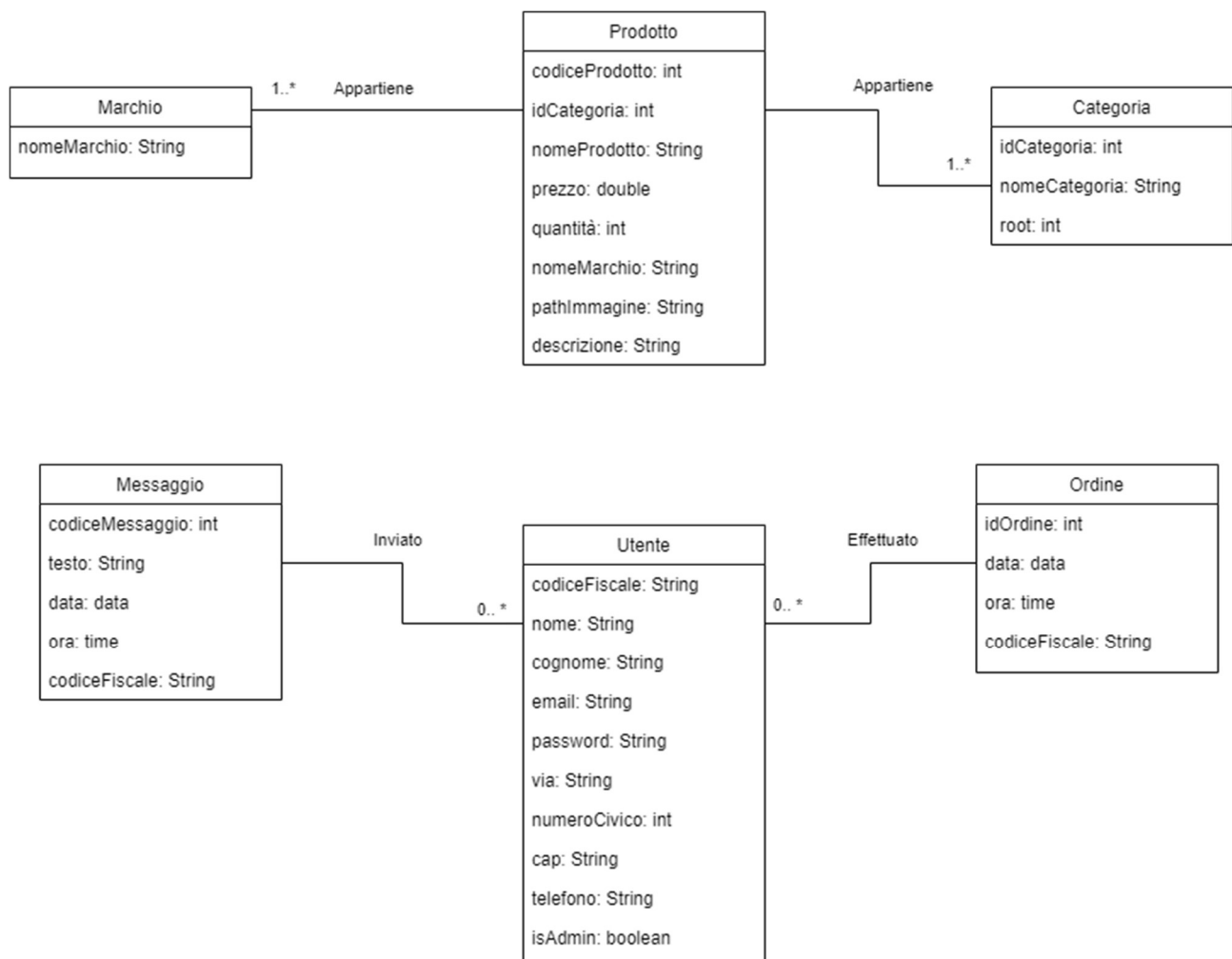
### 3.3 Mapping Hardware/Software

**PHarmaLife** utilizza un'architettura Client-Server. Il Web Server è realizzato da Apache Tomcat 9 ed è situato su una singola macchina, la logica è costituita da Java Servlet mentre l'interfaccia utente è realizzata utilizzando pagine Java Server Page (JSP). Il Client è rappresentato dal Web Browser utilizzato dall'utente che raggiunge il sito tramite un URL; il Web Server accetta le richieste dei client reindirizzandoli alla homepage del sito.



### 3.4 Gestione dati persistenti

Per la gestione dei dati è stato scelto di utilizzare come MYSQL come DMBS al fine di consentire brevi tempi di risposta e ridurre i limiti di spazio di archiviazione. MYSQL oltre a presentare dati ben strutturati ci permette di effettuare operazioni con estrema semplicità e di aggiungere un ulteriore livello di sicurezza per l'accesso ai dati. È possibile ripristinare lo stato del database in caso di danni software o hardware attraverso una copia dei dati fatta periodicamente dal supervisore del gruppo.





Gli oggetti *Entity* che saranno resi persistenti sono:

- **Utente**

- codiceFiscale: varchar (PK)
- nome: varchar
- cognome: varchar
- email: varchar
- password: varchar
- via: varchar
- numeroCivico: varchar
- cap: varchar
- telefono: varchar
- isAdmin: boolean

La tabella *Utente* rappresenta l'account di un utente registrato a PHarmaLife e contiene informazioni utili per la spedizione. L'utente sarà individuato in modo univoco tramite la sua e-mail.

- **Prodotto**

- codiceProdotto: int (PK)
- nome: varchar
- prezzo: float
- marchio: Marchio (FK)
- idCategoria: Categoria (FK)
- quantita: int default 50
- pathImmagine: varchar
- descrizione: varchar

La tabella *Prodotto* contiene informazioni per la descrizione e caratterizzazione dei vari articoli presenti nel catalogo di PHarmaLife. Ogni prodotto è identificato da un codice univoco

- **Categoria**

- id: int (PK)
- nomeCategoria: varchar
- root: int

La tabella *Categoria* contiene tutte le categorie a cui un prodotto può appartenere, le categorie si dividono in macrocategorie, la root serve ad identificare la macrocategoria di appartenenza.

- **Marchio**

- nomeMarchio : varchar (PK)

La Tabella *Marchio* rappresenta i marchi dei vari prodotti.





- **Messaggio**

- codiceMessaggio: int (PK)
- testo: varchar
- dataMessaggio: date
- oraMessaggio: time
- codiceFiscale: Utente (FK)

La tabella *Messaggio* rappresenta un messaggio inviato da un utente registrato al sito al fine ricevere assistenza da un amministratore.

- **Ordine**

- idOrdine: int (PK)
- dataOrdine: date
- oraOrdine: time
- listaProdotti: longtext
- cfUtente: Utente (FK)

La tabella *Ordine* rappresenta l'insieme dei prodotti acquistati da un utente in una certa data. Un ordine è identificato da un ID numerico univoco.



### 3.5 Controllo degli accessi e sicurezza

Il sistema prevede un meccanismo di accessi alla piattaforma tramite le proprie credenziali, univoche per ogni utente, il salvataggio della password nel database prevede un meccanismo di cifratura per garantire la sicurezza di ogni singolo utente. Il sistema nella prima versione di rilascio non prevederà un meccanismo per il recupero della password. Il ciclo di vita dell'entità *Carrello* verrà gestita tramite sessione, al logout dell'utente verranno perse tutte le informazioni relative ai prodotti presenti nel carrello dell'utente.

Gli attori che interagiscono col sistema **PHarmaLife** possono essere classificati in 3 categorie:

- **Utente non loggato:** è un utente generico che accede al sito senza effettuare il login, le operazioni che può fare sono quelle che riguardano l'aggiunta di un prodotto al carrello, il prodotto resterà nel carrello per un certo periodo di tempo, allo scadere di quest'ultimo se non viene effettuata la registrazione al sito, l'utente perderà i prodotti nel carrello.
- **Utente loggato:** può inserire prodotti nel carrello e concludere i propri acquisti, creare una lista di desideri, inviare messaggi d'assistenza all'amministratore e scrivere recensioni.
- **Amministratore:** può svolgere tutte le operazioni di un utente loggato, in aggiunta si occupa anche di inserire, rimuovere e modificare i prodotti del catalogo, visualizzare statistiche sul sito relative al numero di prodotti presenti nel catalogo, agli utenti iscritti alla piattaforma, messaggi ricevuti e ordini effettuati dagli utenti.



Le operazioni che gli utenti dell'applicazione web possono effettuare sugli oggetti sono riportate nella tabella che segue:

- ❖ *In alto*, un'astrazione delle istanze delle classi del nostro sistema;
- ❖ *Sul lato sinistro*, gli attori;
- ❖ *La cella che incrocia attore e istanza*, rappresenta il permesso che quell'attore ha su quella istanza.

	Gestione Utente	Prodotto	Carrello
Utente non Loggato	Registrazione	Visualizza	Aggiunta prodotto
Utente Loggato	Login Logout Modifica credenziali Modifica indirizzo	Visualizza	Aggiunta prodotto, Acquista prodotti
Admin	Login, Logout, Modifica credenziali	Aggiunta prodotto, Modifica prodotto, Elimina prodotto, Visualizza	Aggiunta prodotto, Acquista prodotti

	Ordine	Gestione Preferiti	Messaggio	Recensione
Utente non Loggato				Visualizza recensioni
Utente Loggato	Visualizza, Storico ordini	Aggiunta prodotto	Scrittura	Aggiunta recensione, Visualizza recensioni
Admin	Visualizza, Storico ordini	Aggiunta prodotto	Visualizza	Aggiunta recensione, Visualizza recensioni



## 3.6 Controllo flusso globale del sistema

Il flusso del sistema di “**PHarmaLife**” richiede una continua interazione dell’utente, per cui il controllo del flusso globale del sistema è di tipo event-driven, vale a dire che le azioni del sistema sono guidate dall’input dell’utente. Per quanto riguarda la concorrenza, le funzionalità offerte dal Web Server garantiscono un’interazione concorrente con tutti gli utenti connessi al sistema. Infatti, ogni utente connesso al sistema, tramite il suo browser web, avrà un thread dedicato tramite il quale il server interagirà con lui.

## 3.7 Condizione limite

### Start-up

Lo start-up del sistema prevede l’avvio del web server nel quale il sistema è installato e l’avvio del DBMS per accedere ai dati persistenti memorizzati nel database. Quando sia Web Server che DBMS sono in esecuzione, il sistema carica in memoria centrale le servlet principali attraverso le quali gli utenti possono effettuare le operazioni. Dopo l’avvio del sistema l’utente può interagire con esso.

### Shut-down

Quando il sistema deve essere arrestato, il gestore del sistema termina l’esecutivo del web server. Quando ciò avviene tutte le risorse che il sistema utilizza (connessione al database e connessione alla rete) vengono rilasciate, e nessun utente potrà più connettersi al sistema.

### Fallimento

1. Nel caso in cui si verificasse un’interruzione della fornitura elettrica al server, il sistema non prevede alcuna strategia che ne garantisca l’operabilità in questo tipo di condizione.
2. Nel caso in cui si presentasse una chiusura inaspettata del software, dovuta a errori avvenuti durante la fase di implementazione, il server risponderà con una pagina di errore.
3. Nel caso in cui si verificasse un sovraccarico di richieste al server, questo rimarrà congestionato.
4. Se un utente invia al server informazioni errate (volutamente o meno), o che non permettono la corretta esecuzione di un’operazione, il server risponderà con una pagina di errore.
5. Nel caso di errore critico dell’hardware, non è prevista una soluzione.



Identificativo	UCBC_1 - Start-up	Data	04/01/2022
		Vers.	0.00.001
		Autore	Alfredo Cannavaro [AC]
Descrizione	Lo UC permette l'avvio del web server nel quale il sistema è installato e l'avvio del DBMS per accedere ai dati persistenti memorizzati nel database		
Attore Principale	Amministratore		
Attori secondari	NA		
Entry Condition	L'Amministratore accede al Server		
Exit condition On success	Il sistema viene avviato correttamente		
Exit condition On failure	Il sistema non viene avviato		
FLUSSO DI EVENTI PRINCIPALE/MAIN SCENARIO			
1	Amministratore:	Esegue l'avvio del web server nel quale il sistema è installato e l'avvio del DBMS per accedere ai dati persistenti memorizzati nel database.	
2	Sistema:	Carica in memoria centrale le servlet principali attraverso le quali gli utenti possono effettuare le operazioni.	
3	Utente:	Dopo l'avvio del sistema l'utente può interagire con esso.	
Scenario/Flusso di eventi Alternativo: Web Server che DBMS non vengono eseguite correttamente			
2.a1	Sistema	Notifica l'Amministratore del problema.	
2.a2	Amministratore	Riesegue correttamente l'avvio del sistema	



Identificativo	UCBC_2 - Shut-down	Data	04/01/2022
		Vers.	0.00.001
		Autore	Alfredo Cannavaro [AC]
Descrizione	Lo UC permette lo spegnimento del web server		
Attore Principale	Amministratore		
Attori secondari	NA		
Entry Condition	L'Amministratore accede al Server AND Il Sistema è stato precedentemente avviato AND Il Sistema non è stato ancora spento		
Exit condition On success	Il sistema viene spento correttamente		
Exit condition On failure	Il sistema non viene spento		
FLUSSO DI EVENTI PRINCIPALE/MAIN SCENARIO			
1	Amministratore:	Il gestore del sistema termina l'esecutivo del web server.	
2	Sistema:	Tutte le risorse che il sistema utilizza (connessione al database e connessione alla rete) vengono rilasciate, e nessun utente potrà più connettersi al sistema.	
Scenario/Flusso di eventi Alternativo: Ci sono connessioni ancora aperte			
2.a1	Sistema	Notifica all'Amministratore che ci sono ancora connessioni aperte verso l'esterno.	
2.a2	Sistema	Attende una quantità di tempo per rispondere a eventuali richieste dall'esterno, non generando nuove connessioni se non allo scopo di rispondere a richieste già in corso.	
2.a3	Sistema	Controlla che non ci siano connessioni ancora aperte da o verso l'esterno e, se non ci sono, termina l'esecuzione del sistema.	
2.a4	Sistema	Notifica l'Amministratore dell'avvenuto spegnimento del sistema.	



Identificativo	UCBC_3 - Fallimento	Data	04/01/2022
		Vers.	0.00.001
		Autore	Alfredo Cannavaro [AC]
Descrizione	L'UC definisce il comportamento del Sistema in caso di fallimento		
Attore Principale	Amministratore		
Attori secondari	NA		
Entry Condition	Il Sistema viene terminato inaspettatamente		
Exit condition On success	Il Sistema viene riavviato correttamene		
Exit condition On failure	Il Sistema non viene riavviato		
FLUSSO DI EVENTI PRINCIPALE/MAIN SCENARIO			
1	Sistema:	Nel caso in cui si presentasse una chiusura inaspettata del software, dovuta a errori avvenuti durante la fase di implementazione.	
2	Sistema:	Il server risponderà con una pagina di errore.	
Scenario/Flusso di eventi Alternativo: C'è un'Interruzione della fornitura elettrica al server			
2.a1	Sistema	Il sistema non prevede alcuna strategia che ne garantisca l'operabilità in questo tipo di condizione.	
Scenario/Flusso di eventi Alternativo: C'è un sovraccarico di richieste al server			
3.a1	Sistema	Il server rimarrà congestionato.	
Scenario/Flusso di eventi Alternativo: C'è inserimento di informazioni errate			
4.a1	Utente	Invia al server informazioni errate (volutamente o meno), o che non permettono la corretta esecuzione di un'operazione.	
4.a2	Sistema	Il server risponderà con una pagina di errore.	
Scenario/Flusso di eventi Alternativo: C'è un errore critico dell'hardware			
5.a1	Sistema	Non prevede una soluzione	



## 4. Servizi di sottosistemi

Servizi	Descrizione	Interfaccia
login	Consente ad un utente/amministratore di accedere al sistema tramite una le proprie credenziali.	UtenteService
logout	Consente ad un utente/amministrare di scollegarsi dal sistema.	UtenteService
registrazioneUtente	Permette ad utente non registrato di creare un'account nel sistema.	UtenteService
visualizzazioneDatiPersonaliUtente	Consente ad un utente loggato di visualizzare le informazioni relative al proprio account.	UtenteService
modificaDatiUtente	Consente ad un utente loggato di modificare le informazioni relative al proprio account.	UtenteService
modificaIndirizzo	Consente ad un utente loggato di modificare il proprio indirizzo di spedizione.	UtenteService
visualizzazioneStoricoOrdini	Consente ad un utente loggato di visualizzare gli acquisti da lui effettuati.	UtenteService
VisualizzaListaDesideri	Consente ad un utente loggato utente di visualizzare la propria lista dei desideri.	UtenteService
aggiuntaProdotto	Consente ad un utente loggato di aggiungere un prodotto alla sua lista dei desideri.	UtenteService
rimuoviProdotto	Consente ad un utente loggato di rimuovere un prodotto dalla propria lista dei desideri.	UtenteService





Servizi	Descrizione	Interfaccia
visualizzaCarrello	Consente di visualizzare i prodotti aggiunti nel carrello e il prezzo totale.	CarrelloService
aggiungiProdottoAlCarrello	Consente di aggiungere un prodotto al carrello.	CarrelloService
rimuoviProdottoDalCarrello	Consente di rimuovere un prodotto dal carrello.	CarrelloService
checkoutCarrello	Permette ad un utente loggato di acquistare i prodotti nel carrello. Dopo quest'ultimo il carrello verrà svuotato. In caso di utente non registrato, verrà invitato ad effettuare a registrarsi.	CarrelloService



Servizi	Descrizione	Interfaccia
aggiuntaProdotto	Consente all'amministratore di aggiungere un prodotto al catalogo.	ProdottoService
rimozioneProdotto	Consente all'amministratore di rimuovere un prodotto dal catalogo.	ProdottoService
modificaProdotto	Consente all'amministratore di modificare un prodotto del catalogo.	ProdottoService
visualizzazioneListaProdotti	Consente a tutti gli utenti di visualizzare l'intera lista di prodotti.	ProdottoService
visualizzazioneListaProdottiByMarchio	Consente a tutti gli utenti di visualizzare l'intera lista di prodotti tramite il marchio.	ProdottoService
visualizzazioneListaProdottiByCategoria	Consente a tutti gli utenti di visualizzare l'intera lista di prodotti tramite categoria.	ProdottoService
visualizzazioneSchedaProdotto	Consente a tutti gli utenti di visualizzare una pagina che contiene più informazioni su quel singolo prodotto selezionato.	ProdottoService
filtraggioProdotti	Consente a tutti gli utenti di filtrare la lista dei prodotti presenti nel catalogo.	ProdottoService
ricercaProdotto	Consente a tutti gli utenti di ricercare un prodotto mediante la barra di ricerca.	ProdottoService
recensioneProdotto	Consente ad un prodotto di essere recensito.	ProdottoService
visualizzaRecensioni	Mostra le recensioni scritte da tutti gli utenti su quel prodotto.	ProdottoService



Servizi	Descrizione	Interfaccia
aggiuntaProdotto	Consente all'amministratore di aggiungere un prodotto al catalogo.	AdminService
rimozioneProdotto	Consente all'amministratore di rimuovere un prodotto dal catalogo.	AdminService
modificaProdotto	Consente all'amministratore di modificare un prodotto del catalogo.	AdminService