



**UNIVERSITÀ DEGLI STUDI DI SALERNO**

**Ingegneria del Software**

**SYSTEM DESIGN DOCUMENT**

**IL MERCANTE**



**DI LIBRI**

Anno Accademico 2019/2020

|                                   |                |
|-----------------------------------|----------------|
| PROGETTO: “Il Mercante di Libri”  | Versione 2.0   |
| DOCUMENTO: System Design Document | Data: 10/10/19 |

## Partecipanti

| Nome           | Matricola  |
|----------------|------------|
| Abbate Giorgia | 0512104544 |
| Stanescu Tania | 0512104493 |

## Revisione History

| Data     | Versione | Descrizione                    | Autore          |
|----------|----------|--------------------------------|-----------------|
| 10/10/19 | 1.0      | Prima revisione del documento  | Abbate/Stanescu |
| 14/02/20 | 2.0      | Ultima revisione del documento | Abbate/Stanescu |

# **INDICE**

## **1. *Introduzione***

### **1.2 Scopo del sistema**

### **1.2 Obiettivi di Design**

- Criteri di performance
- Criteri di affidabilità
- Criteri di costo
- Criteri di manutenzione
- Criteri per l'utente finale

## **2. *Architettura Software Proposta***

### **2.1 Panoramica**

### **2.2 Decomposizione del sistema**

### **2.3 Mapping Hardware/Software**

### **2.4 Diagrammi a Design-Time**

### **2.5 Controllo degli accessi e della sicurezza**

### **2.6 Controllo globale del software**

### **2.7 Condizioni di boundary**

## **3. *Servizi dei sottosistemi***

### **3.1 Gestione Autenticazione**

### **3.2 Gestione Registrazione**

### **3.3 Gestione Libri**

### **3.4 Gestione Utenti**

### **3.5 Gestione Acquisti**

## **4. *Glossario***

# 1. Introduzione

## 1.1 Scopo del Sistema

Sono molte le persone interessate al mondo dei libri. Chi per studio, per ricerca, chi per passione, chi per semplice curiosità. Si sa che però spesso i costi possono essere ingenti. Quale miglior modo di arginare il danno ai portafogli di studenti, ricercatori o semplici lettori che quello di offrire un servizio che permette la ricerca di libri usati, a prezzo ridotto? *Il Mercante di Libri* offre ai suoi utenti la possibilità di ricercare libri usati per qualsiasi esigenza a prezzi concorrenti. Offre una piattaforma facile e semplificata che oltre all'acquisto dei libri ne permette anche la vendita. Gli utenti possono quindi essere sia acquirenti che venditori, con l'obiettivo di fornire un servizio completo e facilmente accessibile che contribuisce al risparmio, al riciclo e al mantenimento di una mentalità improntata alla non dissipazione di risorse.

## 1.2 Obiettivi di Design

Il sito *Il Mercante di Libri* deve poter essere il più efficiente ed intuitivo possibile. Tale efficienza sarà costruita attraverso rapidi tempi di risposta ad ogni genere di input, ma anche differenti politiche di tolleranza all'errore. In più si punterà ad avere una buona manutenibilità attraverso il facile inserimento di nuove funzionalità. Sarà intuitivo attraverso agevoli interfacce. Per ottenere gli obiettivi finali vanno seguiti dei criteri di progettazione tenendo presente: Performance, Affidabilità, Costi, Manutenzione e Utente Finale.

### 1.2.1 Criteri di Performance

|                          |   |
|--------------------------|---|
| <b>Tempo di risposta</b> | Il sistema deve essere molto reattivo in quanto non si prevedono carichi di grosse moli di dati. Tuttavia, essendo un sistema web, molto dipenderà dalla qualità della connessione e della congestione della rete sul sistema online.               |
| <b>Throughput</b>        | I picchi di carico devono essere gestiti dal Sistema senza rallentamenti, garantendo la massima fluidità e un latenza bassa. Nei momenti di maggiore carico il Sistema deve garantire consistenza in tutte le chiamate al database                  |
| <b>Memoria</b>           | Il sistema utilizza un database relazionale per memorizzare tutti i dati. La mole dei dati non rappresenterà un problema di performance del sistema. Vengono adottate strategie in modo da favorire le prestazioni e ridurre il carico sul sistema. |

### 1.2.2 Criteri di Affidabilità

|                               |  |
|-------------------------------|--|
| <b>Robustezza</b>             | I componenti devono essere affidabili , in caso di guasti o problemi dovuti all'hardware e devono essere in grado di mantenere i dati.   |
| <b>Disponibilità</b>          | Il Mercante di Libri deve essere disponibile all'uso degli utenti 24 ore su 24, grazie all'impiego di un server sempre attivo.   |
| <b>Tolleranza dell'errore</b> | Il Sistema deve funzionare anche in condizioni di errore. Ciò è reso possibile grazie al fatto che il Sistema viene modulato e diviso in modo tale che se si verifica un errore in una parte questo non incide sulle altre funzionalità.   |
| <b>Sicurezza</b>              | Le tecniche per garantire sono basate prevalentemente su un login, la quale permette il riconoscimento dell'utente. Tutti gli accessi avvengono tramite un'operazione di controllo, in particolare l'amministratore avrà i massimi permessi poiché ha la facoltà di utilizzare qualsiasi tipo di funzione disponibile. Infine un utente ha la facoltà di utilizzare qualsiasi tipo di funzione che riguarda la propria gestione e organizzazione dei prodotti. |

### 1.2.3 Criteri di Costo

|                   |   |
|-------------------|---|
| <b>Sviluppo</b>   | I costi di sviluppo prevedono l'affitto del server per la fase di sviluppo. Per quanto riguarda il dbms utilizzeremo mysql che è open source. |
| <b>Deployment</b> | Non sono previsti costi di deployment   |

#### 1.2.4 Criteri di Manutenzione

|                                    |   |
|------------------------------------|---|
| <b>Estensibilità</b>               | La progettazione sarà introdotta in modo tale da agevolare l'introduzione di nuove funzionalità, utilizzando il linguaggio HTML e I fogli CSS.  |
| <b>Modificabilità</b>              | Deve essere possibile intervenire sul codice esistente per correggere eventuali bug o implementare nuove funzionalità. Bisogna garantire che il codice sia leggibile per rendere agevole la modifica.     |
| <b>Leggibilità</b>                 | Il codice sarà ben strutturato per semplificare eventuali interventi su di esso.  |
| <b>Tracciabilità dei requisiti</b> | Grazie alla tracciabilità dei requisiti, sarà possibile effettuare le modifiche necessarie al corretto funzionamento del sistema, valutando correttamente i costi e i rischi che le modifiche porteranno. |

#### 1.2.5 Criteri per l'utente finale

|                  |   |
|------------------|---|
| <b>Usabilità</b> | Il Mercante di Libri deve essere facilmente apprendibile (l'utente deve essere in grado di interagire con il sistema e padroneggiare le funzionalità in modo rapido), deve essere flessibile e robusto (l'utente deve essere in grado di capire quando ha successo nel perseguire i suoi obiettivi nel sistema o quando sta sbagliando qualcosa per poter intervenire). |
|------------------|---|

## Panoramica

Il documento è composto da diverse parti, la prima parte riguarda gli obiettivi di design, mentre la seconda parte che verrà successivamente introdotta, riguarda brevemente l'architettura del sistema corrente prima e dopo il re-engineering.

Prima di iniziare ad introdurre la spiegazione dell'architettura è necessario fare un resoconto delle attività di system design che compongono l'architettura software del sistema.

- **Decomposizione del sistema:** Il sistema viene decomposto in sottosistema in modo tale da poter assegnare lo sviluppo di diverse parti di software facili da sviluppare ai vari componenti del team. Un sottosistema è un insieme di classi, associazioni, operazioni, eventi e vincoli che sono in relazione tra di loro, con lo scopo di fornire un servizio. Ogni sottosistema è caratterizzato da servizi che offre ad altri sottosistemi.
- **Mapping Hardware/Software:** descrive in che modo i sottosistemi vengono assegnati all'hardware e alle componenti "off-the-shelf". Elenca tutti i problemi relativi al riuso del software.
- **Gestione dati persistenti:** descrive i dati persistenti memorizzati dal sistema.
- **Politiche di accesso e sicurezza:** descrive il modello utente del sistema, stabilendo in modo preciso le informazioni e le operazioni effettuate da ogni singolo attore tramite le diverse politiche di accesso con le quali si autenticano al sistema.
- **Flusso di controllo globale:** descrive quali operazioni eseguire e in che ordine per poter stabilire e garantire il corretto flusso di controllo del sistema.
- **Condizioni Limite:** descrive lo start-up, lo shutdown e i comportamenti errati del sistema.

## 2. ARCHITETTURA DEL SOFTWARE PROPOSTO

### 2.1 Panoramica

Il sistema proposto è un web-based per la gestione di compravendita di libri online. Le tipologie di utenti si suddividono in amministratore e utente. L'amministratore è colui che usufruirà di tutte le funzionalità del sistema e gestirà eventuali problemi che vi si possono presentare ,rimozione di utenti e libri. L'utente potrà gestire solamente i propri prodotti e i propri acquisti dopo aver effettuato la registrazione alla piattaforma.

L'architettura del sistema si presenta come un'architettura Client/Server. Il server invia le richieste da parte del Client e cercherà di rispondere in un tempo ristretto.

I motivi di questa scelta sono:

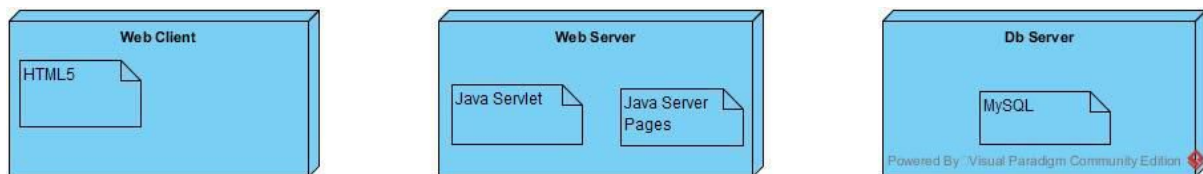
- **Portabilità:** il sistema potrà essere utilizzato su diverse macchine e diversi sistemi operativi.
- **Trasparenza:** il sistema è in grado di fornire i propri servizi al singolo utente senza interferire con gli altri utenti del sistema.

- **Performance:** il sistema garantirà un'elevata reattività per i task degli utenti collegati ma essendo web-based il tutto dipenderà anche dalla qualità della connessione.
- **Scalabilità:** il sistema sarà in grado di gestire un elevato numero di utenti connessi che effettuano tutte le operazioni contemporaneamente.
- **Flessibilità:** il sistema fornirà un'interfaccia grafica intuitiva e facile da utilizzare, con funzionalità specifiche per il ruolo dell'utente che ne effettua l'accesso.
- **Affidabilità:** i componenti client e server devono essere in grado di mantenere i dati anche in caso di guasto.

Nello sviluppo del software utilizzeremo un'architettura di tipo MVC, tale Pattern prevede che il software venga diviso in tre parti ognuna delle quali ha un compito diverso. La view si occuperà di curare l'interazione con l'utente. Il controller riceverà informazioni della View per mandarli al model o viceversa, mentre il model si occuperà della gestione dei dati e quindi dell'interazione con il database sottostante.

L'utilizzo di questo modello comporta numerosi vantaggi, primi fra tutti sono la facilità di manutenzione ed estensione. Infatti essendo uno standard di programmazione sarà di facile comprensione a programmatori futuri.

I model verranno realizzati utilizzando classi Java appropriate, la parte di view verrà implementata utilizzando html in pagine jsp e i control saranno realizzati tramite servlet.



## 2.2 Decomposizione del Sistema

Per realizzare il sistema Il Mercante di Libri è stato utilizzato lo stile architetturale three-tier in versione Client/server. L'architettura three-tier ("a tre strati") indica una particolare architettura software di tipo multi-tier per l'esecuzione di un'applicazione web che prevede la suddivisione dell'applicazione in tre strati dedicati rispettivamente alla interfaccia utente, alla logica funzionale e alla gestione dei dati persistenti. In particolare i tre strati si occupano di differenti funzionalità del sistema di seguito descritte.

- **Il Presentation Layer:** Include tutte le interfacce grafiche e in generale i boundary objects, come le form con cui interagisce l'utente. L'interfaccia verso l'utente è rappresentata da un Web server e da eventuali contenuti statici (es. pagine HTML).
- **L'Application Layer:** include tutti gli oggetti relativi al controllo e all'elaborazione dei dati. Questo avviene interrogando il database tramite lo storage layer per generare contenuti dinamici e accedere a dati persistenti.



- **Lo storage Layer:** effettua la memorizzazione, il recupero e l'interrogazione degli oggetti persistenti. I dati, i quali possono essere acceduti dall'Application Layer, sono depositati in maniera persistente su un database tramite DBMS.

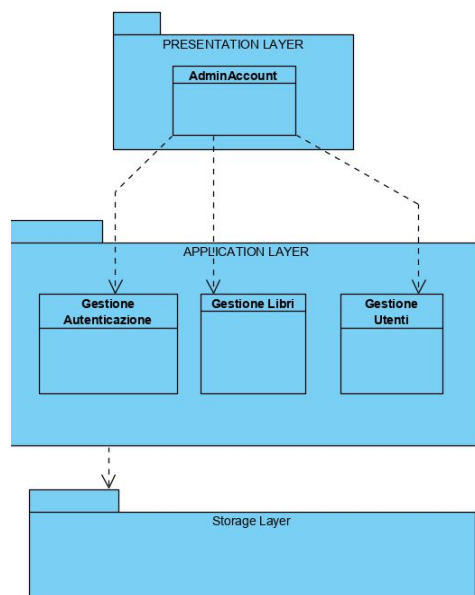
Descrizione delle gestioni del sistema:

Il sottosistema Application Layer è stato suddiviso in vari sottosistemi:

- **Gestione Autenticazione:** Questa funzionalità raccoglie tutte le informazioni per gestire l'autenticazione degli utenti su Il Mercante di Libri
- **Gestione Registrazione:** Questa funzionalità permette di registrarsi fornendo i propri dati.
- **Gestione Libri:** Questa funzionalità offre la possibilità di gestire i libri nel sistema
- **Gestione Utenti:** Questa funzionalità raccoglie tutte le operazioni necessarie per gestire gli utenti.
- **Gestione Acquisti:** Questa funzionalità consente all'utente di effettuare gli acquisti.

Il sottosistema presentation layer è stato suddiviso in due sottosistemi:

## Admin Account

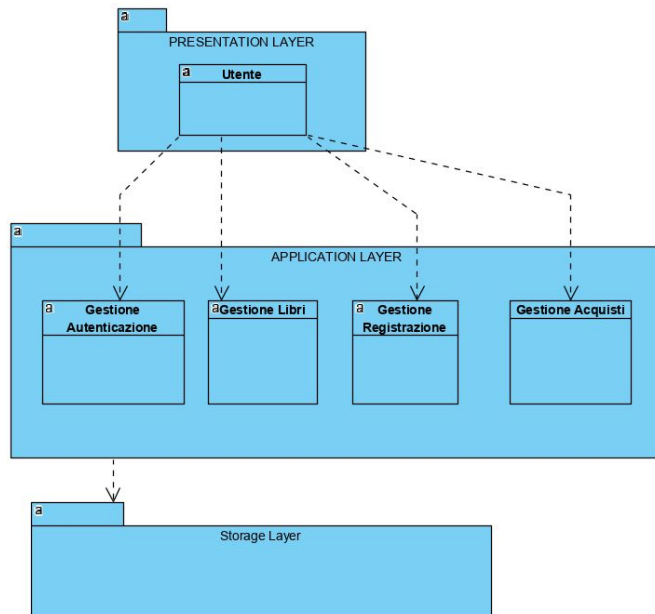


**Gestione Autenticazione:** L'admin può effettuare l'accesso al sistema utilizzando le proprie credenziali e di uscire dal sistema.

**Gestione Libri:** L'admin può visualizzare le informazioni di tutti i libri presenti sul sistema, può rimuovere un libro già esistente.

**Gestione Utenti:** L'admin può visualizzare le informazioni di tutti gli utenti presenti sul sistema e può rimuovere uno o più utenti.

## Utente



**Gestione Autenticazione:** L'utente può effettuare l'accesso al sistema utilizzando le proprie credenziali e di uscire dal sistema.

**Gestione Libri:** L'utente può cercare e visualizzare un determinato libro o una lista di libri.

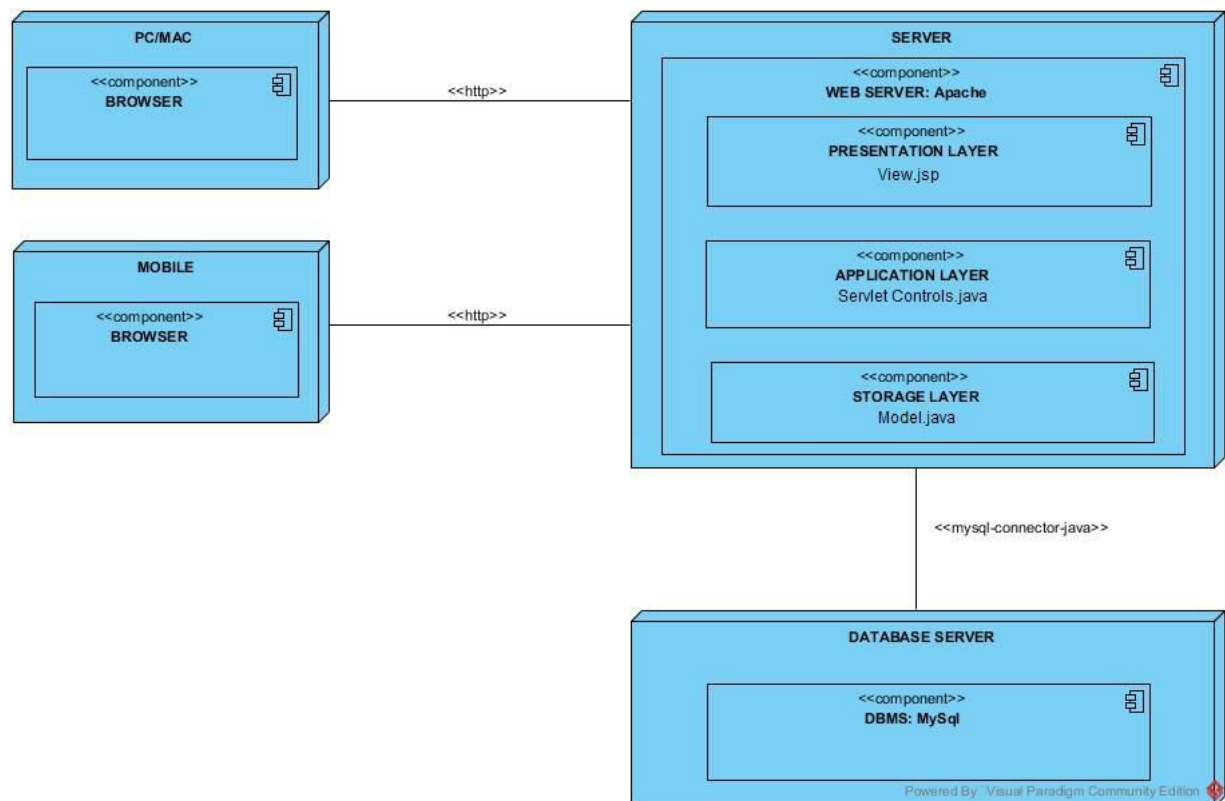
**Gestione Acquisti:** L'utente può inserire tutte le informazioni necessarie per effettuare gli acquisti.

**Gestione Registrazione:** L'utente si iscrive al sito in modo da permettere l'autenticazione in futuro.

## 2.3 Mapping Hardware/Software

La struttura hardware è costituita da un server e dai client cioè un qualsiasi utente può collegarsi al browser per accedere a Il Mercante di Libri. I client si collegano al server e al database, il tipo di utente viene determinato in fase di autenticazione controllando nel database il tipo di utente che corrisponde all'E-Mail inserita in quel client. I client dovranno effettuare delle richieste al server per eseguire le operazioni legate alle loro rispettive funzionalità. Il client ed il server saranno connesse tramite una rete che utilizzerà il protocollo TCP/IP.

Di seguito verrà mostrato il diagramma che mostra le funzionalità e i collegamenti hardware:



## Web Server

Il server utilizzato è Apache.

**Presentation layer:** l'utente utilizza il sistema mediante un browser installato all'interno del suo computer/mobile (es. Firefox, Chrome).

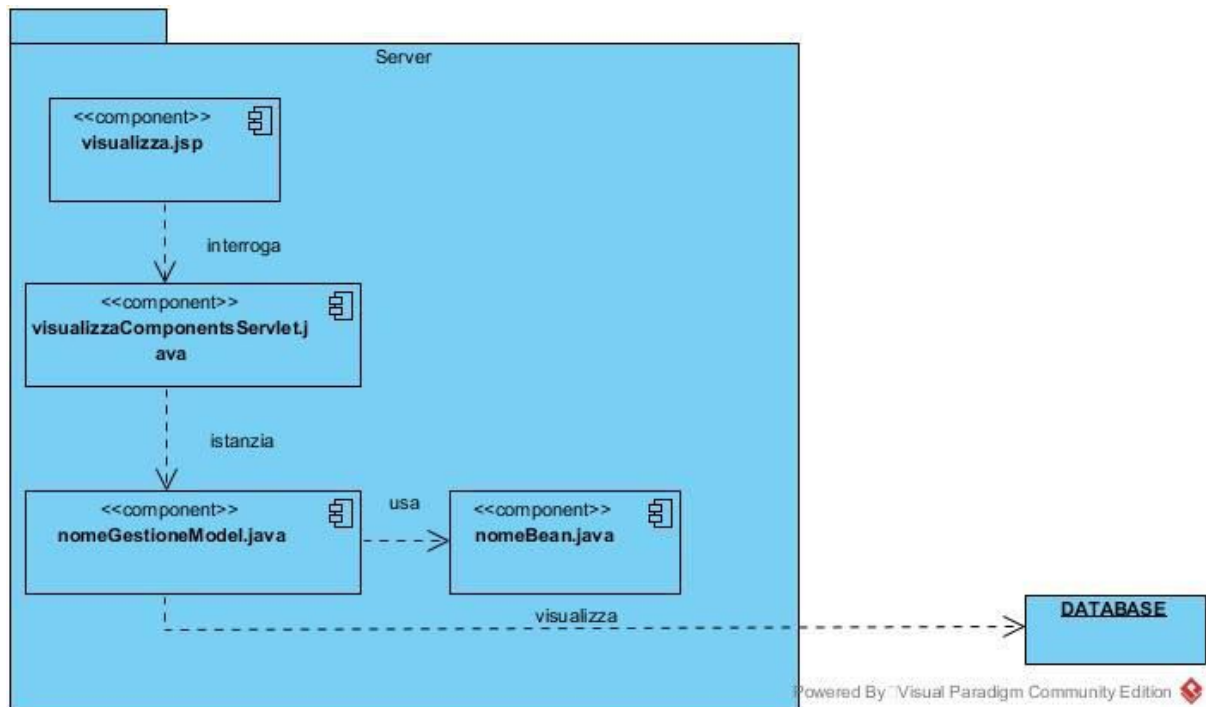
**Application layer:** il codice è scritto in linguaggio HTML5+java.

**Storage layer:** rappresenta il collegamento con il server da parte del sistema e si occupa delle richieste di accesso e di modifica sui dati stabili presenti nel database.

**Database Server:** Il DBMS usato è MySQL che permette l'interazione tra sistema e database.

## 2.4 Diagrammi a Design-Time:

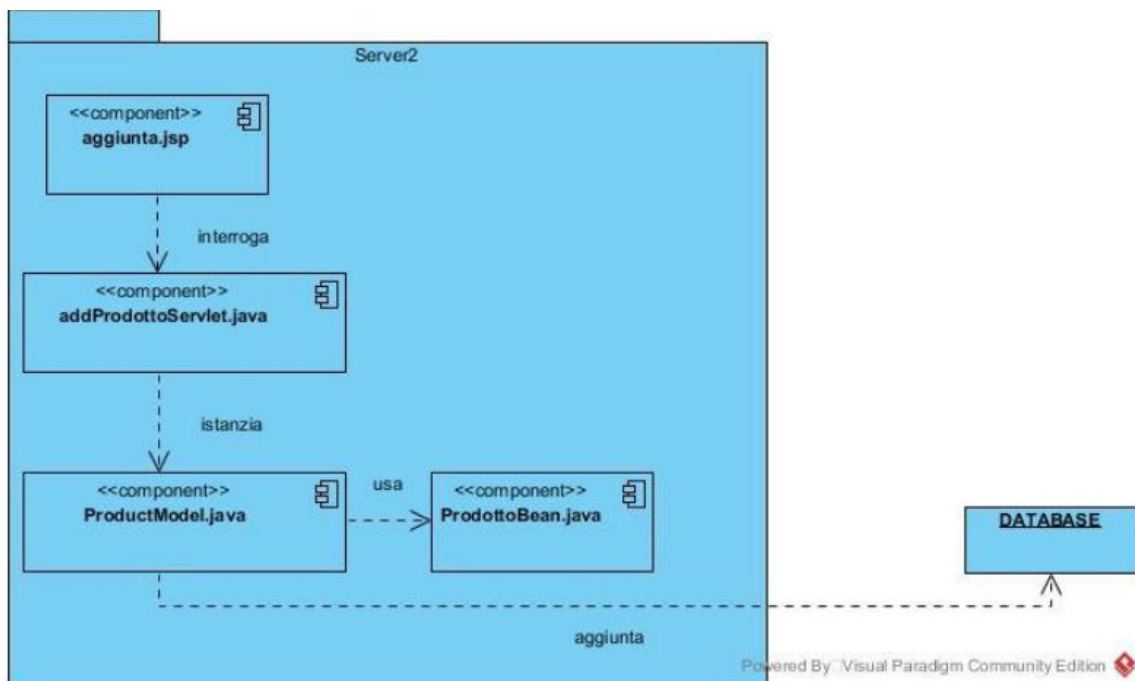
### Visualizza



Il precedente diagramma che descrive staticamente cosa succede a design time quando si vuole visualizzare una lista di entità preesistenti e si riferisce ai seguenti sequence diagram:

- SD\_3.1: visualizza utenti
- SD\_4.1: visualizza carrello
- SD\_2.1: visualizza catalogo

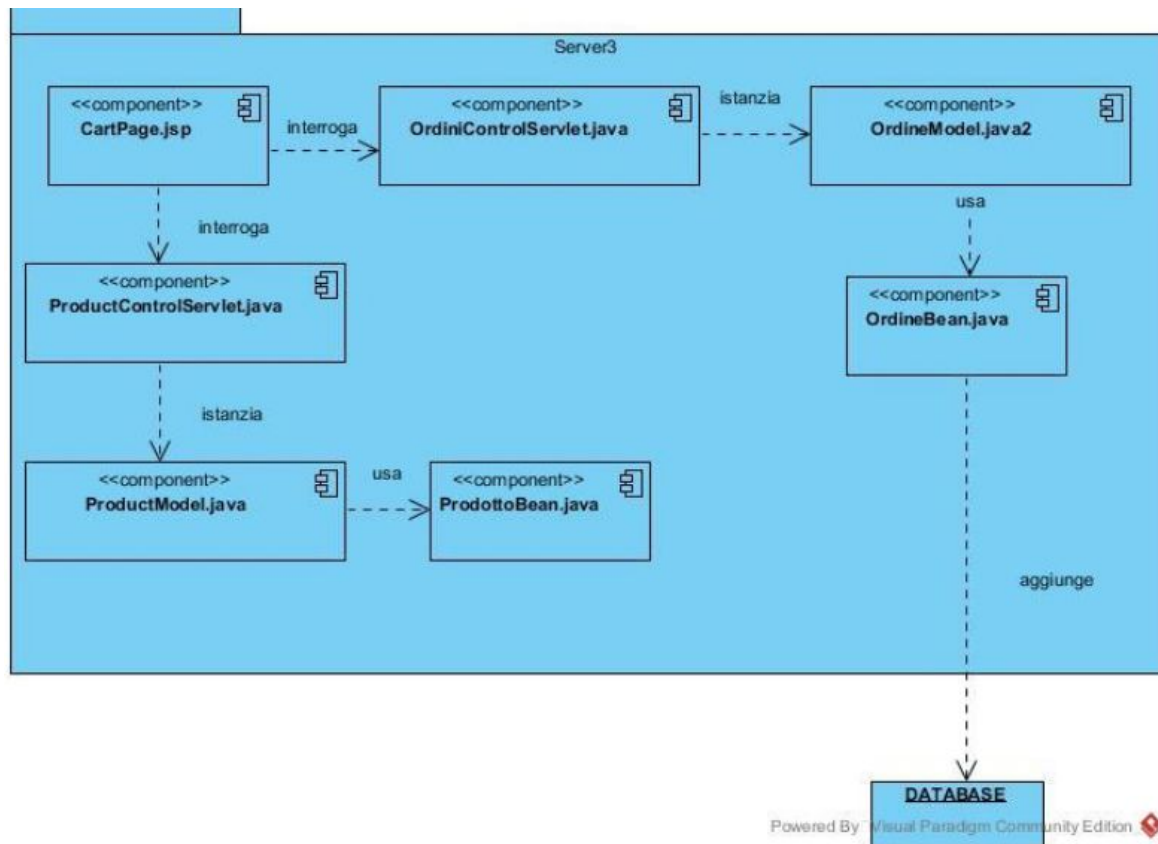
### Aggiunta



Il precedente diagramma che descrive staticamente cosa succede a design time quando si vuole effettuare un'aggiunta si riferisce ai seguenti sequence diagram:

-SD\_2.2

## Ordini



Il precedente diagramma che descrive staticamente cosa succede a design time quando si vuole effettuare un acquisto si riferisce ai seguenti sequence diagram:

-SD\_4.0

## 2.5 Controllo degli accessi e della sicurezza

Il Mercante di Libri è un sistema multi-utente, ci sono diversi attori che hanno il permesso di eseguire diverse operazioni su vari insiemi di oggetti. Per schematizzare al meglio il controllo degli accessi abbiamo suddiviso per tipologia di utente le azioni consentite, al fine di ottenere una visione più compatta e dettagliata grazie ad una matrice degli accessi riportata di seguito:

| Attori                | Utente          | Admin           |
|-----------------------|-----------------|-----------------|
| <b>Oggetti</b>        |                 |                 |
| <b>Autenticazione</b> | Login<br>Logout | Login<br>Logout |

|                      |  |   |
|----------------------|--|---|
| <b>Registrazione</b> | Registra   |   |
| <b>Libri</b>         | Visualizza catalogo<br>Ricerca Libro<br>Aggiungi Libro | Visualizza Catalogo<br>Ricerca Libro<br>Rimuovi Libro |
| <b>Utenti</b>        |  | Visualizza Utente<br>Rimuovi Utente                   |
| <b>Ordini</b>        | Acquista Libro   |   |

## 2.6 Controllo Globale del Software

Il flusso di controllo globale è la sequenza di azioni nel sistema. Il sistema ha un flusso guidato di eventi; le funzionalità richiedono un'interazione continua da parte dell'utente.

Il controllo del flusso globale che utilizziamo è di tipo event-driven. Quindi non abbiamo una sequenza di operazioni prestabilite ma, è l'utente che sceglie l'operazione da eseguire.

## 2.7 Condizione Boundary

Le condizioni limite riguardano, dal lato Server, l'accensione e lo spegnimento del sistema. Riguardano invece, dal lato Client, gli errori di connessione al server.

### 2.7.1 Avvio del sistema

Il sistema dopo essersi avviato presenta un'interfaccia ai client. Una volta effettuata la fase di login ogni utente può accedere alle funzionalità disponibili.

### 2.7.2 Terminazione del sistema

Dopo aver usufruito di tutte le funzionalità del sistema, l'utente può far terminare la sua permanenza nel sistema eseguendo un'operazione di logout. Inoltre, il sistema può anche terminare in caso di manutenzione del sistema, esecuzioni di operazioni sospette da parte degli utenti ecc...

### 2.7.3 Fallimento del sistema

Nel caso si verifichi un errore dovuto all'hardware o al software si cercherà di ripristinare il sistema. I dati poiché sono gestiti dal DBMS non c'è alcuna possibilità che vengano persi. Tuttavia, non è da escludere la perdita dei dati, in quanto si può verificare un guasto al supporto di memorizzazione dei dati. Per minimizzare questo problema, il sistema sarà sottoposto periodicamente a controlli.

|  |   |
|--|---|
| <b>Nome Scenario</b>                       | SC_Startup Server   |
| <b>Istanze di attori partecipanti</b>      | Dina: Admin   |
| <b>Flusso di Eventi<br/>Amministratore</b> | <ol style="list-style-type: none"> <li>1. Dina, amministratrice del sito vuole avviare il sistema e quindi clicca sul pulsante “Avvia”.</li> <li>2. Il sistema attiva i server e i relativi servizi in remoto, con le opportune procedure di avvio.</li> <li>3. Il sistema notifica il successo dell’operazione.</li> </ol> |

|  |  |
|--|--|
| <b>Nome Scenario</b>                       | SC_Shutdown Server   |
| <b>Istanze di attori partecipanti</b>      | Dina: Admin  |
| <b>Flusso di Eventi<br/>Amministratore</b> | <ol style="list-style-type: none"> <li>1. Dina, amministratrice del sito vuole arrestare il sistema e quindi clicca sul pulsante “Arresta”.</li> <li>2. Il sistema controlla eventuali Client ancora connessi e ne termina la connessione.</li> <li>3. Tramite le opportune procedure di arresto il sistema disattiva i servizi in remoto e il server.</li> <li>4. Il sistema notifica il successo dell’operazione.</li> </ol> |

|   |   |               |                |   |  |
|---|---|---------------|----------------|---|--|
| <b>ID</b>   | UC_Startup  |               |                |   |  |
| <b>Nome Use Case</b>  | StartupServer   |               |                |   |  |
| <b>Partecipanti</b>   | Amministratore  |               |                |   |  |
| <b>Condizioni d’ingresso</b>                                      | L’amministratore accede al sistema  |               |                |   |  |
| <b>Flusso di Eventi</b>   | <table border="0"> <tr> <td><b>Utente</b></td> <td><b>Sistema</b></td> </tr> <tr> <td>L’amministratore accede al sistema e clicca sul pulsante “Avvia”.</td> <td>Il Mercante di Libri attiva i propri servizi rendendosi disponibile in tutte le sue funzionalità e notifica il successo dell’operazione.</td> </tr> </table> | <b>Utente</b> | <b>Sistema</b> | L’amministratore accede al sistema e clicca sul pulsante “Avvia”. | Il Mercante di Libri attiva i propri servizi rendendosi disponibile in tutte le sue funzionalità e notifica il successo dell’operazione. |
| <b>Utente</b>   | <b>Sistema</b>  |               |                |   |  |
| L’amministratore accede al sistema e clicca sul pulsante “Avvia”. | Il Mercante di Libri attiva i propri servizi rendendosi disponibile in tutte le sue funzionalità e notifica il successo dell’operazione.  |               |                |   |  |

|                             |   |
|-----------------------------|---|
| <b>Condizione di Uscita</b> | Il Sistema è attivo con tutte le sue funzionalità |
| <b>Eccezioni</b>            |   |
| <b>Requisiti di qualità</b> |   |

|  |   |               |                |  |  |
|--|---|---------------|----------------|--|--|
| <b>ID</b>  | UC_Shutdown   |               |                |  |  |
| <b>Nome Use Case</b>   | ShutdownServer  |               |                |  |  |
| <b>Partecipanti</b>  | Amministratore  |               |                |  |  |
| <b>Condizioni d'ingresso</b>                                       | L'amministratore accede al sistema  |               |                |  |  |
| <b>Flusso di Eventi</b>  | <table> <tr> <td><b>Utente</b></td><td><b>Sistema</b></td></tr> <tr> <td>L'amministratore accede al sistema e clicca sul pulsante "Spegni".</td><td>Il Mercante di Libri effettua una scansione per verificare client connessi e li disconnette; avvia la procedura di arresto. Il sistema notifica il successo dell'operazione.</td></tr> </table> | <b>Utente</b> | <b>Sistema</b> | L'amministratore accede al sistema e clicca sul pulsante "Spegni". | Il Mercante di Libri effettua una scansione per verificare client connessi e li disconnette; avvia la procedura di arresto. Il sistema notifica il successo dell'operazione. |
| <b>Utente</b>  | <b>Sistema</b>  |               |                |  |  |
| L'amministratore accede al sistema e clicca sul pulsante "Spegni". | Il Mercante di Libri effettua una scansione per verificare client connessi e li disconnette; avvia la procedura di arresto. Il sistema notifica il successo dell'operazione.  |               |                |  |  |
| <b>Condizione di Uscita</b>  | Il sistema viene arrestato.   |               |                |  |  |
| <b>Eccezioni</b>   | Errore di Shutdown  |               |                |  |  |
| <b>Requisiti di qualità</b>  |   |               |                |  |  |



### 3. SERVIZI DEI SOTTOSISTEMI

#### 3.1 Gestione Autenticazione

|                        |   |
|------------------------|---|
| <b>Sottosistema</b>    | Gestione Autenticazione   |
| <b>Descrizione</b>     | Gestisce le operazioni di autenticazione di tutti gli attori e le operazioni necessarie al controllo dei dati inseriti.   |
| <b>Servizi offerti</b> |   |
| Login                  | Questa funzionalità permette di effettuare l'accesso al Sistema, inserendo le proprie credenziali, in modo da poter sfruttare le varie funzionalità del Sistema |
| Logout                 | Questa funzionalità permette di uscire dal Sistema  |
| Dati non Trovati       | Questa funzionalità permette al Sistema di restituire un errore nel caso in cui non sono presenti dati inerenti durante la fase di login                        |
| Errore sui Dati        | Questa funzionalità permette al Sistema di restituire un errore nel caso in cui I dati immessi per il login presentano caratteri non consentiti                 |

#### 3.2 Gestione Registrazione

|                        |  |
|------------------------|--|
| <b>Sottosistema</b>    | Gestione Registrazione   |
| <b>Descrizione</b>     | Gestisce le operazioni di registrazione di tutti gli attori.   |
| <b>Servizi offerti</b> |  |
| Registra Utente        | Questa funzionalità permette ad un nuovo utente di registrarsi |

#### 3.3 Gestione Libri

|                        |  |
|------------------------|--|
| <b>Sottosistema</b>    | Gestione Libri   |
| <b>Descrizione</b>     | Gestisce tutti i libri presenti nel Sistema.                           |
| <b>Servizi offerti</b> |  |
| Visualizza Catalogo    | Permette di visualizzare i libri presenti nel sistema                  |
| Aggiunta Manuale       | Questa funzionalità permette di uscire dal Sistema                     |
| Rimozione Libro        | Permette la rimozione di più prodotti dal Sistema da parte dell'admin. |

### 3.4 Gestione Utenti

|                        |  |
|------------------------|--|
| <b>Sottosistema</b>    | Gestione Utenti  |
| <b>Descrizione</b>     | Permette la gestione degli utenti  |
| <b>Servizi offerti</b> |  |
| Rimozione Utenti       | Permette di rimuovere uno o più utenti registrati nel sistema.               |
| Visualizza Utenti      | Permette di visualizzare le informazioni sugli utenti registrati nel sistema |

### 3.5 Gestione Acquisti

|                        |  |
|------------------------|--|
| <b>Sottosistema</b>    | Gestione Acquisti  |
| <b>Descrizione</b>     | Gestisce tutti gli acquisti effettuati dagli utenti                        |
| <b>Servizi offerti</b> |  |
| Acquisto Libro         | Permette di acquistare gli elementi contenuti nel carrello                 |
| Visualizza Carrello    | Questa funzionalità permette di visualizzare i libri presenti nel carrello |
| Aggiungi al Carrello   | Permette all'utente di aggiungere un libro al carrello.                    |

## 4. Glossario

- **Il Mercante di Libri:** nome del sistema in sviluppo
- **Utente:** attore del sistema che effettua operazioni sul sistema
- **Admin:** attore del sistema che si occupa della gestione e della manutenzione del sistema.
- **Login:** attività di accesso all'account
- **Logout:** attività di uscita dell'account connesso
- **User-Friendly:** è un aggettivo che viene utilizzato per definire un software di facile utilizzo soprattutto per persone che non sono esperte nell'utilizzo dei computer
- **RAD:** Requirements Analysis Document
- **SDD:** System Design Document
- **DBMS:** Database Management System
- **URL:** Uniform Resource Locator, è una sequenza di caratteri che identifica in modo univoco un indirizzo che appartiene a una risorsa di internet
- **SQL:** Structed Query Language, è un linguaggio di database