

# Università degli Studi di Salerno

Corso di Ingegneria del Software

## **SocialBook** **Documento System Design**



**Partecipanti:**

Nome	Matricola
Barbato Alessia	0512105858
Proietto Angelica	0512105762
Russo Luca	0512105840

## Revision History

Data	Versione	Descrizione	Autore
22/12/2020	1.0	Stesura introduzione (Paragrafi 1.1, 1.2, 1.3, 1.4)	Barbato Alessia
26/12/2020	2.0	Stesura architettura sistema corrente e proposto (Paragrafi 3.1, 3.2, 3.3, 3.4)	Barbato Alessia
29/12/2020	3.0	Stesura controllo degli accessi e sicurezza, controllo globale del software, condizioni limite	Proietto Angelica
30/12/2020	4.0	Stesura servizi dei sottosistemi, glossario	Russo Luca

# Indice

1. Introduzione .....	4
1.1 Obiettivi del sistema.....	4
1.2 Design goals .....	4
1.3 Acronimi e abbreviazioni.....	5
1.4 Riferimenti .....	6
1.5 Panoramica.....	6
2. Architettura del sistema corrente .....	6
3. Architettura del sistema proposto .....	6
3.1 Panoramica.....	6
3.2 Decomposizione in sottosistemi.....	7
3.3 Mapping hardware/software .....	9
3.3.1 Deployment .....	10
3.4 Gestione dati persistenti .....	10
3.4.1 Schema logico .....	11
3.4.2 Struttura delle tabelle .....	12
3.5 Controllo degli accessi e sicurezza .....	16
3.6 Controllo globale del software .....	17
3.7 Condizioni limite.....	17
4. Servizi dei sottosistemi .....	18
5. Glossario .....	21

# 1. Introduzione

## 1.1 Obiettivi del sistema

L'obiettivo del sistema è quello di offrire a tutti gli appassionati di lettura una piattaforma online sicura che permetta all'utente non solo di acquistare libri (in formato digitale), ma anche di organizzarli come meglio si preferisce, recensirli, e comunicare con gli altri utenti.

L'interfaccia del sistema dovrà essere estremamente intuitiva, in modo da invogliare gli utenti ad interagire tra di loro.

L'obiettivo della piattaforma è quella di fornire un servizio veloce e sempre aggiornato in modo da garantire l'esperienza migliore possibile agli utenti.

## 1.2 Design Goals

ID	DESCRIZIONE	CATEGORIA	ORIGINE
DG_1 Tempo di risposta	Il sistema garantisce il minor tempo possibile di risposta (massimo 1 sec).	Prestazioni	NFR_4
DG_2 Portabilità	Il sistema è responsive (visualizza correttamente le pagine adattandosi ai display di dispositivi diversi: pc, tablet e cellulari).	Usabilità Implementazione Mantenimento	NFR_1 NFR_6
DG_3 Usabilità	Il sistema è strutturato con una interfaccia intuitiva (organizzazione di pulsanti, form e menu chiara e ben precisa).	Usabilità	NFR_2
DG_4 Throughput	In caso di massimo carico del lavoro, il sistema non deve bloccarsi.	Prestazioni	NFR_4
DG_5 Sicurezza	Il sistema protegge i dati sensibili degli utenti attraverso la crittografia; l'utente può usufruire esclusivamente delle funzionalità associate alla propria categoria.	Affidabilità Security	NFR_7 NFR_3
DG_6 Memoria	Il sistema fa affidamento sul database relazionale MySQL per rendere persistenti e gestire le informazioni.	Implementazione	NFR_6
DG_7 Disponibilità	Fatta eccezione per il tempo di manutenzione, le funzionalità del	Affidabilità	NFR_3

	sistema sono accessibili in ogni momento.		
<b>DG_8 Robustezza</b>	L'inserimento di dati non conformi o il mancato riempimento dei campi viene segnalato tramite notifica di errore.	Affidabilità	NFR_3
<b>DG_9 Affidabilità</b>	Il sistema garantisce una corretta gestione delle funzionalità.	Affidabilità	NFR_3
<b>DG_10 Adattabilità</b>	Il sistema è flessibile (modulare ed estendibile) e si adatta alle nuove necessità dell'utente.	Implementazione Mantenimento	NFR_5 NFR_6

### 1.3 Acronimi e abbreviazioni

**RAD:** Requirements Analysis Document.

**FR:** Requisiti funzionali.

**NFR:** Requisiti non funzionali.

**SC:** Scenario.

**UC:** Use case.

**HTTP:** HyperText Transfer Protocol.

**TCP/IP:** Transmission Control Protocol/Internet Protocol.

**MVC:** Model-View-Controller.

**DG:** Design Goal.

**GUI:** Graphic User Interface.

**HTML:** Hyper Text Markup Language.

**CSS:** Cascading Style Sheets.

**JSTL:** JavaServer Pages Standard Tag Library.

**SQL:** Structured Query Language.

**JDBC:** Java DataBase Connectivity.

**DB:** DataBase.

**DBMS:** DataBase Management System.

**RDBMS:** Relational DataBase Management System.

**XML:** eXtensible Markup Language.

**PK:** Primary Key.

**FK:** Foreign Key.

**JSP:** Java Server Page.

**Utente:** Un utilizzatore della piattaforma che non si è ancora registrato.

**Utente registrato:** Un utilizzatore iscritto alla piattaforma.

**Customer manager:** Amministratore che si occupa della gestione degli utenti.

**System manager:** Amministratore che si occupa della gestione di bug, segnalati da utenti che utilizzano la piattaforma.

**Catalogue manager:** Amministratore che si occupa della gestione del catalogo libri.

**Admin:** Generalizzazione di customer manager, catalogue manager e system manager.

## **1.4 Riferimenti**

- Object-Oriented Software Engineering Using UML, Patterns, and Java, Bernd Bruegge & Allen H. Dutoit.
- SocialBook RAD.

## **1.5 Panoramica**

Il documento si compone di una prima parte in cui vengono introdotti gli obiettivi di design. Al secondo punto del documento viene presentata l'architettura del sistema corrente.

Al terzo punto viene presentata l'architettura del sistema proposto e nel dettaglio:

- La decomposizione del sistema in sottosistemi.
- Il mapping hardware/software.
- La gestione dei dati persistenti.
- Il controllo degli accessi e della sicurezza.
- Il controllo del flusso globale del sistema.
- Le condizioni limite.

Al quarto punto vengono presentati i servizi di ogni sottosistema.

Al quinto punto viene fornito il glossario dei termini utilizzati nel documento con le relative definizioni.

# **2. Architettura del sistema corrente**

Attualmente non esiste una piattaforma a tema e-book che offra sia funzionalità di marketing che quelle di un blog. Inoltre, i social attuali afferenti a tale campo sono popolati da cerchie di utenti della stessa zona (per fattori socio-lessicali); dato che limita pericolosamente la qualità della rete di users di cui si potrebbe disporre.

Infine, i siti di questo tipo vengono sponsorizzati da case editrici che forzano la presenza di libri da loro pubblicati affievolendo l'eterogeneità dei prodotti.

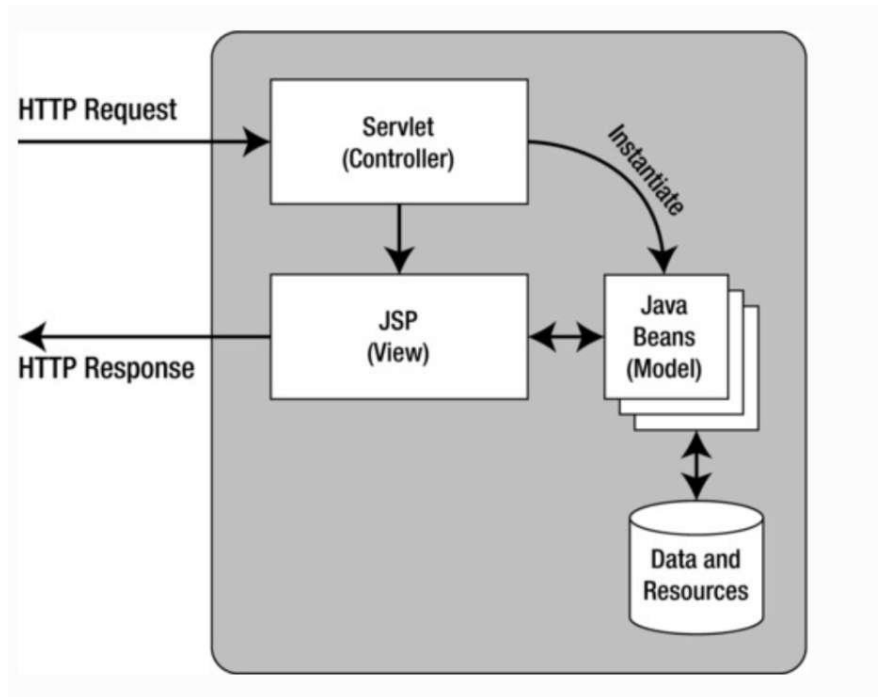
# **3. Architettura del sistema proposto**

## **3.1 Panoramica**

Il sistema proposto vuole rendere più completa la gamma di funzionalità di una piattaforma associata agli e-book. Per far ciò, offre delle funzionalità fondamentali:

- consente l'acquisto degli e-book e la possibilità di recensirli;
- consente di creare liste di e-book da comprare e leggere;
- consente di seguire altri utenti e/o liste di e-book;

Per la progettazione e per lo sviluppo di SocialBook si è scelta l'architettura MVC (Model-View-Controller) che separa la logica di business da quella di presentazione.



**Model:** Si occupa di gestire i dati persistenti, contiene metodi di lettura e scrittura sul Database.

**View:** Si occupa di far visualizzare i dati all'utente e dell'interazione tra quest'ultimo ed il sistema.

**Controller:** Si occupa della logica di controllo dell'applicazione interagendo con le altre due componenti.

### 3.2 Decomposizione in sottosistemi

Presentiamo una decomposizione del nostro sistema in 3 layer (o livelli):

- Presentation Layer
- Application Layer
- Data Access Layer

Questi livelli si occupano di gestire funzionalità ed aspetti differenti del sistema.

#### Presentation Layer (View)

Livello che gestisce la comunicazione con le entità esterne al sistema tramite delle componenti che si occupano di rappresentare l'informazione verso i client, consentendo essi di interagire con il sistema.

- *Utente non registrato:* GUI che gestisce l'interfaccia degli utenti che non hanno effettuato la registrazione e/o l'accesso alla piattaforma, i quali sono limitati alla visualizzazione del catalogo dei libri, la pagina personale di ognuno e dei profili degli altri utenti registrati.

- *Utente registrato:* GUI che gestisce l'interfaccia degli utenti che sono registrati e hanno effettuato l'accesso alla piattaforma e permette di visualizzare: il profilo personale (con ordini, preferiti, ticket e

booklist), il carrello, il catalogo dei libri, la pagina personale di ognuno, la lista di seguaci/seguiti e il profilo degli altri utenti.

- *Customer Manager*: GUI che gestisce l'interfaccia dedicata al responsabile della gestione degli utenti e permette di visualizzare ed eliminare utenti oppure commenti degli utenti.
- *System Manager*: GUI che gestisce l'interfaccia dedicata al responsabile della gestione del sistema e permette la modifica del codice per la risoluzione di bug o implementazione di nuove features.
- *Catalogue Manager*: GUI che gestisce l'interfaccia dedicata al responsabile del catalogo e permette di aggiungere/eliminare libri al/dal catalogo, modificare il prezzo dei libri ed inserire un nuovo libro.

#### Application Layer (Controller)

Livello del sistema che si occupa del processamento dei dati, necessario per produrre i risultati da inoltrare al Presentation Layer.

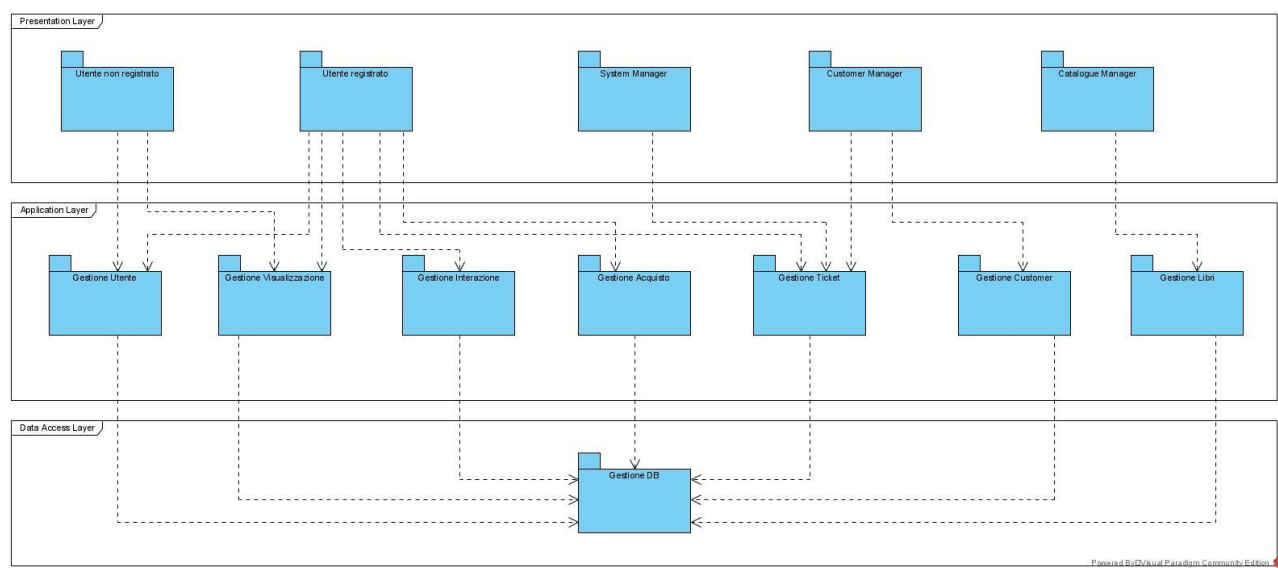
- *Gestione Utente*: Sottosistema che permette di registrarsi, effettuare il login e il logout, modificare alcune informazioni personali, accedere e visualizzare i servizi relativi all'area personale (booklist, preferiti) e recensire libri.
- *Gestione Visualizzazione*: Sottosistema che permette di visualizzare il catalogo, la pagina personale di ogni libro, i profili degli utenti registrati e di effettuare la ricerca (per titolo, genere e autore) e visualizzarne i risultati.
- *Gestione Interazione*: Sottosistema che permette di far seguire all'utente altri utenti e altre booklist (con la possibilità di smettere di seguire in un secondo momento) e visualizzare le liste (lista dei seguiti, dei seguaci, delle booklist seguite) proprie e degli altri.
- *Gestione Acquisto*: Sottosistema che permette all'utente registrato di aggiungere/rimuovere al carrello, acquistare libri e visualizzare gli ordini pregressi.
- *Gestione Ticket*: Sottosistema che gestisce la corrispondenza tra utente e admin e la visualizzazione dei ticket pregressi da parte dell'utente registrato e degli admin.
- *Gestione Libri*: Sottosistema che permette la gestione di tutte le funzionalità dedicate al Catalogue Manager, quali inserimento al catalogo o rimozione dal catalogo, modifica al prezzo dei libri, creazione di un nuovo libro.
- *Gestione Customer*: Sottosistema che permette la gestione delle funzionalità dedicate al Customer Manager, quali visualizzazione di tutti gli utenti registrati e la rimozione di commenti/utenti (registrati) dovute a segnalazioni (ticket) da parte di altri utenti.

#### Data Access Layer (Model)

Livello che gestisce i dati necessari al funzionamento dell'intero sistema, ovvero i dati persistenti.

- *Gestione DataBase (DB)*: Sottosistema che si occupa di immagazzinare e di prelevare i dati persistenti dal nostro DB.





### 3.3 Mapping hardware/software

Per il progetto software che si vuole sviluppare, verrà fatto uso di componenti “off-the-shelf”, componenti software disponibili sul mercato per facilitare la creazione del progetto.

Il framework che verrà adottato comprenderà Bootstrap, JavaScript, jQuery, JSTL.

Bootstrap è un framework open-source che contiene una raccolta di strumenti per facilitare lo sviluppo di interfacce web;

JavaScript è un linguaggio di scripting orientato agli oggetti e agli eventi, comunemente utilizzato nella programmazione Web lato client;

jQuery è una libreria JavaScript che viene utilizzata per accelerare lo sviluppo della logica di business delle interfacce web.

JSTL è una libreria di tag standard inclusa come componente della piattaforma software di sviluppo per applicazioni web Java EE. È un'estensione di JSP ed incorpora un insieme di tag HTML definiti tramite file XML e programmati in linguaggio Java. JSTL supporta attività comuni, strutturali come iterazione e condizionali, tag per la manipolazione di documenti XML, tag di internazionalizzazione e tag SQL.

Sono state inoltre scelti questi tipi di configurazione:

per la parte di Presentation (View), saranno utilizzate le JSP, i linguaggi HTML, CSS e JavaScript e la libreria JSTL;

per la parte di Application (Control), verrà utilizzato il linguaggio Java, il server Apache Tomcat e Maven, uno strumento di build automation utilizzato prevalentemente nella gestione di progetti Java;

per la parte di Data Access (Model), verranno utilizzati MySQL, relational database management system (RDBMS) composto da un client a riga di comando e un server e JDBC, connettore per database che consente l'accesso e la gestione della persistenza dei dati sulle basi di dati da qualsiasi programma scritto con il linguaggio Java, indipendentemente dal tipo di DBMS utilizzato.

#### 3.3.1 Deployment

Il sistema sviluppato si poggia su un'architettura Client/Server, cioè suddivisa in:

- una macchina server che risponde alle richieste effettuabili da un client;
- una qualsiasi macchina client dotata di connessione ad Internet e di un Browser Web, che può effettuare richieste.

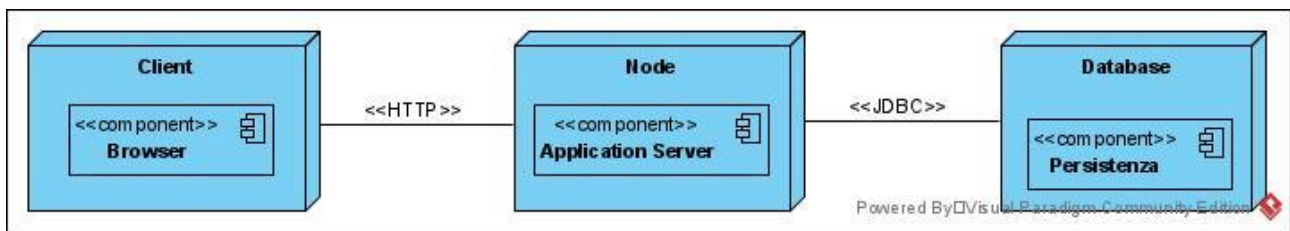
Tale modello presenta i seguenti vantaggi:

- mediante tale divisione l'elaborazione è soprattutto a carico del server, ciò consente ai client di fruire al meglio del sistema a prescindere dall'hardware di cui sono composti;
- la distanza fisica tra client e server risulta irrilevante grazie all'uso del protocollo HTTP;

Bisogna comunque tenere in considerazione alcuni aspetti negativi:

- il costo hardware e di manutenzione del server sono potenzialmente elevati dovuti all'alto carico di lavoro;
- l'utilizzo di un solo server centralizzato crea un singolo punto di rottura.

L'applicazione potrebbe risultare non fruibile in caso di errori (sia software che hardware).

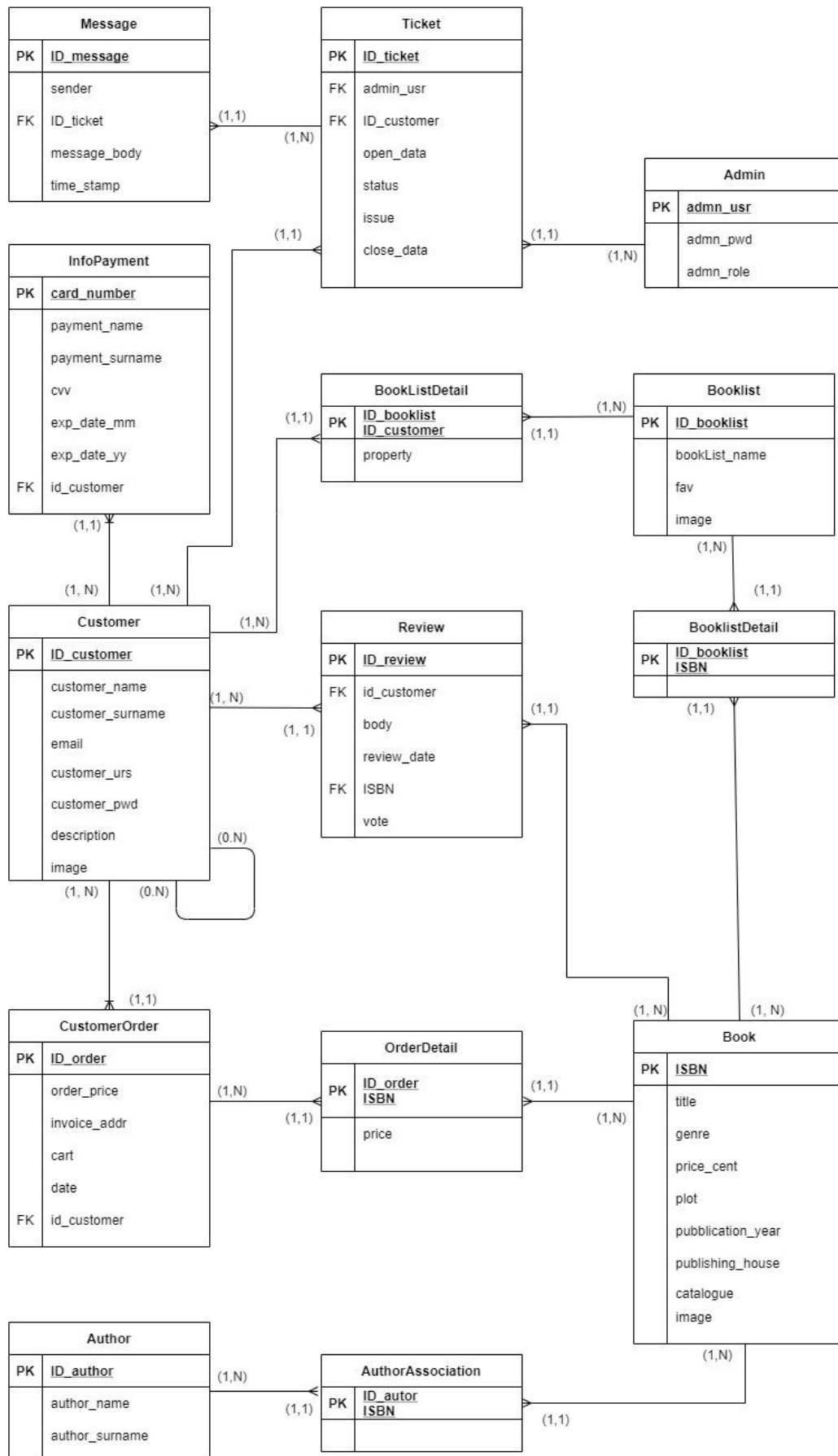


### **3.4 Gestione dati persistenti**

Il sistema sarà installato su un solo computer e utilizzerà un DBMS MySQL installato sullo stesso.

Il sistema sarà diviso in client e server che inizialmente saranno lo stesso PC, ma in caso di futura necessità potranno facilmente essere divisi in quanto i servizi saranno progettati separatamente.

### 3.4.1 Schema logico



### 3.4.2 Struttura delle tabelle

Tabella customer

Attributo	Tipo	Vincoli
id_customer	Integer	PRIMARY KEY
customer_name	String	NOT NULL
customer_surname	String	NOT NULL
e-mail	String	NOT NULL, UNIQUE
customer_pwd	String	NOT NULL
customer_usr	String	NOT NULL, UNIQUE
description	String	NOT NULL
image	String	

Tabella book

Attributo	Tipo	Vincoli
ISBN	String	PRIMARY KEY
title	String	NOT NULL
genre	String	NOT NULL
price_cent	Integer	NOT NULL
publication_year	Integer	NOT NULL
publishing_house	String	NOT NULL
plot	String	NOT NULL
catalogue*	Bool	NOT NULL
image	String	NOT NULL

\*1 = presente nel catalogo, 0 = non presente nel catalogo

Tabella author

Attributo	Tipo	Vincoli
id_author	Integer	PRIMARY KEY
author_surname	String	NOT NULL
author_name	String	NOT NULL

Tabella authorAssociation

Attributo	Tipo	Vincoli
id_author	Integer	PRIMARY KEY FOREIGN KEY AUTHOR(id_author)
ISBN	String	PRIMARY KEY FOREIGN KEY BOOK(ISBN)

Tabella customerOrder

Attributo	Tipo	Vincoli
id_order	Integer	PRIMARY KEY
order_price	Decimal	NOT NULL
invoice_addr	String	
cart*	Bool	NOT NULL
date	Date	
id_customer	Integer	NOT NULL FOREIGN KEY CUSTOMER(id_customer)

\* 0 = carrello, 1 = ordine

Tabella orderDetail

Attributo	Tipo	Vincoli
id_order	Integer	PRIMARY KEY FOREIGN KEY ORDER(id_order)
ISBN	String	PRIMARY KEY FOREIGN KEY BOOK(ISBN)
price	Decimal	NOT NULL

Tabella infoPayment

Attributo	Tipo	Vincoli
card_number	String	PRIMARY KEY
payment_name	String	NOT NULL
payment_surname	String	NOT NULL
exp_date_mm	String	NOT NULL
exp_date_yy	String	NOT NULL
cvv	String	NOT NULL

id_customer	Integer	NOT NULL FOREIGN KEY CUSTOMER(id_customer)
-------------	---------	--

Tabella booklist

Attributo	Tipo	Vincoli
id_booklist	Integer	PRIMARY KEY
booklist_name	String	NOT NULL
favorite*	Boolean	NOT NULL
image	String	

\* 0 = booklist, 1 = preferiti

Tabella booklistDetail

Attributo	Tipo	Vincoli
id_booklist	Integer	PRIMARY KEY FOREIGN KEY BOOKLIST(id_booklist)
id_customer	Integer	PRIMARY KEY FOREIGN KEY CUSTOMER(id_customer)
property*	Boolean	NOT NULL

\* 0 = proprietario della booklist , 1 = follower della booklist

Tabella admin

Attributo	Tipo	Vincoli
admn_usr	String	PRIMARY KEY
admn_pwd	String	NOT NULL
admn_role	String	NOT NULL

Tabella review

Attributo	Tipo	Vincoli
id_review	Integer	PRIMARY KEY
id_customer	Integer	NOT NULL FOREIGN KEY CUSTOMER(id_customer)
ISBN	String	NOT NULL

		FOREIGN KEY BOOK(ISBN)
review_date	Date	NOT NULL
body	String	
vote	Integer	

Tabella ticket

Attributo	Tipo	Vincoli
id_ticket	Integer	PRIMARY KEY
id_customer*	Integer	NOT NULL FOREIGN KEY CUSTOMER(id_customer)
admn_usr	String	NOT NULL FOREIGN KEY ADMIN(admn_urs)
open_date	Date	NOT NULL
issue	String	NOT NULL
close_date	Date	
status	String	NOT NULL

\* 4 = id\_customer fittizio, utilizzato per memorizzare i ticket degli utenti non registrati

Tabella message

Attributo	Tipo	Vincoli
id_message	Integer	PRIMARY KEY
sender*	Boolean	NOT NULL
id_ticket	Integer	NOT NULL FOREIGN KEY TICKET(id_ticket)
time_stamp	TimeStamp	NOT NULL
message_body	String	NOT NULL

\* 0= customer 1 = admin

Tabella follow

Attributo	Tipo	Vincoli
id_customer	Integer	PRIMARY KEY FOREIGN KEY CUSTOMER(id_customer)
id_follower	Integer	NOT NULL FOREIGN KEY CUSTOMER(id_customer)

### 3.5 Controllo degli accessi e sicurezza

Il sistema può essere utilizzato da qualsiasi utente con una connessione internet e la possibilità di utilizzare un web browser, per il compimento di un insieme ristretto di funzionalità che va ad allargarsi quando l'utente si registra alla piattaforma.

La piattaforma predispone dei controlli per l'iscrizione al sistema, dando ad ogni utente la possibilità di registrarsi utilizzando una coppia di credenziali (username e password, dove la password deve necessariamente seguire un formato specifico, cioè deve contenere una lettera maiuscola, una lettera minuscola, un numero e un simbolo speciale). L'accesso alla piattaforma per gli utenti è garantito dall'utilizzo di queste credenziali.

La sicurezza sui dati sensibili è garantita dall'accesso controllato, in quanto soltanto il customer manager può avere accesso ai dati relativi agli utenti (esclusa la password).

Tutte le funzionalità del sistema sono descritte nella matrice degli accessi sottostante.

Attori	Gestione Utente	Gestione Visualizzazione	Gestione Interazione	Gestione Acquisto	Gestione Ticket	Gestione Customer	Gestione Libri
Utente	<<Registrazione>>	<<Ricerca>> <<Visualizzazione catalogo>> <<Visualizzazione libro>> <<Visualizzazione profili utenti>>	<<Visualizzazione seguiti/seguaci>> <<Visualizzazione booklist>>		<<Contattare admin>>		
Utente registrato	<<Autenticazione>> <<Visualizzazione area personale>> <<Recensione libro>> <<Elimina recensione>> <<Visualizzazione preferiti>> <<Inserimento libro nei preferiti>> <<Rimozione libro dai preferiti>> <<Creazione booklist>> <<Modifica booklist>> <<Eliminazione booklist>> <<Visualizzazione booklist personali>>	<<Ricerca>> <<Visualizzazione catalogo>> <<Visualizzazione libro>> <<Visualizzazione profili utenti>>	<<Segui utente>> <<Smetti di seguire utente>> <<Segui booklist>> <<Smetti di seguire booklist>> <<Visualizzazione seguiti/seguaci>> <<Visualizzazione booklist>>	<<Visualizzazione carrello>> <<Inserimento libro nel carrello>> <<Rimozione libro dal carrello>> <<Acquisto>> <<Visualizzazione ordini>>	<<Contattare admin>>		



	<<Modifica informazioni personali>> <<Logout>>						
<i>Customer Manager</i>		<<Visualizzazione catalogo>>  <<Visualizzazione libro>>  <<Visualizzazione profili utenti>>			<<Visualizzazione ticket>>	<<Visualizzazione utenti>>  <<Rimozione utente>>  <<Rimozione commento>>	
<i>Catalogue Manager</i>		<<Visualizzazione catalogo>>  <<Visualizzazione libro>>  <<Visualizzazione profili utenti>>					<<Inserimento libro al catalogo>>  <<Rimozione libro dal catalogo>>  <<Modifica al prezzo libro>>  <<Creazione nuovo libro>>
<i>System Manager</i>		<<Visualizzazione catalogo>>  <<Visualizzazione libro>>  <<Visualizzazione profili utenti>>			<<Visualizzazione ticket>>		

### 3.6 Controllo globale del software

Il controllo del flusso del software è regolato da classi Java che fungono da ricevitori di eventi e che rispondono alle attivazioni di client.

Le richieste vengono generate da un client e la classe preposta a gestire quel determinato evento associato alla richiesta, prendendo talvolta degli input, si preoccupa di settare le richieste invocando classi specifiche che, eventualmente, si interfacciano con il database per lo svolgimento dell'operazione. Una volta ottenuto il risultato dell'operazione, la classe gestore si preoccupa di inoltrarlo al client che aveva generato la richiesta.

Il sistema software è gestito con l'uso di Servlet e JSP.

### 3.7 Condizioni limite

SERVER:

- Fase di Startup:

La fase di avvio del server Apache Tomcat dura circa trenta secondi, dopodiché viene avviato il server dove è contenuto il DBMS MySQL. La connessione al DBMS verrà effettuata tramite il driver JDBC.

Successivamente viene avviata la JVM sul server e vengono letti i dati di configurazione del database.

Infine, vengono caricati tutti i dati in memoria volatile e il sistema è pronto all'uso.

- Fase di Terminazione:

Il server può essere spento in caso di guasto o in caso di necessità di manutenzione.

Terminata l'attività di manutenzione o di riparazione del guasto il processo server viene avviato nuovamente.

- Fallimento:

Il server può fallire in caso di interruzione inaspettata dell'alimentazione e/o di guasti hardware.

Il server può andare in crash quando viene sollevata un'eccezione che non è stata gestita.

#### CLIENT:

- Scenario "Startup":

Salvatore è un tecnico che intende avviare la piattaforma SocialBook.

Una volta avviato il web server Apache Tomcat, configura il DBMS MySQL per essere raggiunto dal medesimo web server.

A questo punto il web server carica su Tomcat l'eseguibile della piattaforma SocialBook.

Dopo circa trenta secondi dall'avvio, la piattaforma è operativa e raggiungibile dagli utenti.

- Fase di Terminazione:

All'utente basterà chiudere la pagina del sistema per interrompere la comunicazione, interrompendo così la sessione.

- Scenario "Fallimento":

Il server mostra al client una schermata di errore. Per evitare la perdita di dati, viene comunicato al DBMS di effettuare una forzata scrittura su memoria fisica delle informazioni ancora presenti in memoria volatile. Il DBMS mantiene dei registri di log dove sono annotate tutte le operazioni sul DB.

## 4. Servizi dei sottosistemi

Sottosistema	Gestione Utente
Descrizione	Permette all'utente registrato di effettuare varie operazioni relative alla propria area personale.
Servizi offerti	
Servizio	Descrizione
Registrazione	Consente la registrazione di un qualsiasi utente che non sia già registrato.
Autenticazione	Consente l'autenticazione di un qualsiasi utente già registrato.
Logout	Consente il logout di un qualsiasi utente che abbia effettuato l'accesso.

Visualizzazione area personale	Consente a un utente registrato la visualizzazione della sua area personale (preferiti, booklist).
Modifica informazioni personali	Consente a un utente registrato di modificare la propria password, descrizione o immagine del profilo.
Recensione libro	Consente a un utente registrato di aggiungere una recensione a un qualsiasi libro.
Elimina Recensione	Consente a un utente registrato di eliminare una propria recensione inserita precedentemente.
Visualizzazione preferiti	Consente a un utente registrato di visualizzare la lista dei preferiti.
Inserimento libro nei preferiti	Consente a un utente registrato di inserire un qualsiasi libro ai preferiti.
Rimozione libro dai preferiti	Consente a un utente registrato di rimuovere un qualsiasi libro aggiunto precedentemente ai preferiti.
Creazione booklist	Consente a un utente registrato di creare una booklist.
Modifica booklist	Consente a un utente registrato di modificare una qualsiasi delle proprie booklist (modifica nome, aggiunta e rimozione libro).
Elimina booklist	Consente a un utente registrato di eliminare una qualsiasi delle proprie booklist.
Visualizzazione booklist personali	Consente a un utente registrato di visualizzare le proprie booklist.
<b>Sottosistema</b>	<b>Gestione Visualizzazione</b>
<i>Descrizione</i>	Permette all'utente la visualizzazione del catalogo, della pagina di un qualsiasi libro, dei profili degli utenti registrati e di effettuare la ricerca e visualizzarne i risultati.
<b>Servizi offerti</b>	
<i>Servizio</i>	<i>Descrizione</i>
Visualizzazione catalogo	Consente a un utente di visualizzare il catalogo dei libri.
Visualizzazione libro	Consente a un utente di visualizzare la pagina di un libro qualsiasi..
Visualizzazione profili utenti	Consente a un utente di visualizzare il profilo di un qualsiasi utente registrato.
Ricerca	Consente la ricerca di un libro per titolo/genere/autore e la visualizzazione dei risultati.
<b>Sottosistema</b>	<b>Gestione Interazione</b>
<i>Descrizione</i>	Permette all'utente registrato di interagire con gli altri utenti seguendoli oppure seguendo le loro booklist.
<b>Servizi offerti</b>	
<i>Servizio</i>	<i>Descrizione</i>
Segui utente	Consente a un utente registrato di seguire un qualsiasi altro utente registrato (e quindi tutte le sue booklist).

Smetti di seguire utente	Consente a un utente registrato di smettere di seguire un utente che aveva seguito precedentemente.
Segui booklist	Consente a un utente registrato di seguire una qualsiasi booklist.
Smetti di seguire booklist	Consente al Customer Manager di rimuovere un commento che ha ricevuto tante segnalazioni. Consente a un utente registrato di smettere di seguire una booklist che aveva seguito precedentemente.
Visualizzazione seguiti/seguaci	Consente a un utente di visualizzare le liste dei seguiti/seguaci di un qualsiasi utente registrato (e le proprie nel caso in cui anch'esso sia registrato).
Visualizzazione booklist	Consente a un utente di visualizzare le booklist degli altri utenti (e le proprie nel caso in cui anch'esso sia registrato).
<b>Sottosistema</b>	<b>Gestione Acquisto</b>
<i>Descrizione</i>	Permette all'utente registrato di gestire il proprio carrello e effettuare acquisti.
<b>Servizi offerti</b>	
<i>Servizio</i>	<i>Descrizione</i>
Visualizzazione carrello	Consente a un utente registrato di visualizzare il proprio carrello.
Inserimento libro nel carrello	Consente a un utente registrato di aggiungere qualsiasi libro al carrello.
Rimozione libro dal carrello	Consente a un utente registrato di rimuovere un qualsiasi libro aggiunto precedentemente al carrello.
Acquisto	Consente a un utente registrato di acquistare i libri che ha aggiunto al carrello.
Visualizzazione ordini	Consente a un utente registrato di visualizzare tutti i suoi ordini pregressi.
<b>Sottosistema</b>	<b>Gestione Ticket</b>
<i>Descrizione</i>	Permette la corrispondenza tra utente e admin e la visualizzazione dei ticket pregressi (da parte degli admin e l'utente solo se registrato).
<b>Servizi offerti</b>	
<i>Servizio</i>	<i>Descrizione</i>
Interazione ticket	Consente ad un utente di inviare un ticket ad un admin (di categoria Customer Manager oppure System Manager), inoltre permette lo scambio di messaggi tra utenti registrati e admin attraverso i ticket stessi.
Visualizzazione ticket	Consente a un utente registrato/admin di visualizzare i ticket pregressi.
<b>Sottosistema</b>	<b>Gestione Customer</b>
<i>Descrizione</i>	Permette al Customer Manager di visualizzare tutti gli utenti registrati e i vari ticket e di rimuovere un utente/commento segnalato.
<b>Servizi offerti</b>	
<i>Servizio</i>	<i>Descrizione</i>

Visualizzazione utenti	Consente al Customer Manager di visualizzare la lista degli utenti registrati e i loro dati (esclusa la password).
Rimozione utente	Consente al Customer Manager di rimuovere un utente registrato che ha ricevuto tante segnalazioni.
Rimozione recensione	Consente al Customer Manager di rimuovere un commento che ha ricevuto tante segnalazioni.
<b>Sottosistema</b>	<b>Gestione Libri</b>
<i>Descrizione</i>	Permette al Catalogue Manager di gestire il catalogo dei libri.
<b>Servizi offerti</b>	
<i>Servizio</i>	<i>Descrizione</i>
Inserimento libro al catalogo	Consente al Catalogue Manager di inserire un nuovo libro nel catalogo.
Modifica al prezzo libro	Consente al Catalogue Manager di modificare il prezzo di un libro presente nel catalogo.
Rimozione libro dal catalogo	Consente al Catalogue Manager di rimuovere un libro dal catalogo.
Creazione nuovo libro	Consente al Catalogue Manager di creare e salvare un nuovo libro.

## 5. Glossario

**SDD (System Design Document):** Documento formalizzato alla definizione di obiettivi di progettazione del sistema, decomposizione del sistema in sottosistemi più piccoli e scelta di architettura software più adatta al sistema.

**RAD (Requirement Analysis Document):** documento contenente informazioni inerenti al sistema da realizzare raccolte durante la fase di Requirement Analysis e Requirement Elicitation.

**Server:** componente che gestisce traffico di informazioni e fornisce servizi e risorse attraverso la rete.

**Client:** componente che accede a servizi e risorse del server.

**Web Browser:** applicazione software installata sul client che permette di visualizzare e navigare le risorse del web.

**DBMS:** sistema software per la creazione, manipolazione e interrogazione efficiente di database.

**SQL:** linguaggio standardizzato per database basati sul modello relazionale (RDBMS).

**PK:** Primary Key di una tabella, ovvero attributo (colonna) che non può contenere né valori ripetuti né valori nulli.

**FK:** Foreign Key, ovvero vincolo di integrità referenziale tra due o più tabelle. Identifica una o più colonne di una tabella, detta tabella referenziante, che riferenzia una o più colonne di una seconda tabella, detta referenziata.

**HTTP:** protocollo di trasferimento di ipertesti che consente a due macchine, client e server, di interagire attraverso un meccanismo di richiesta/risposta. Il client inoltra una richiesta al server, che verrà soddisfatta con la risposta di quest'ultimo.

**TCP:** protocollo di rete a pacchetto di livello di trasporto che si occupa di controllo della trasmissione, ovvero di rendere affidabile la comunicazione dati in rete tra mittente e destinatario.

**Servlet:** oggetti Java all'interno del server web che permettono di creare web applications in combinazione con JSP.

**JSP:** tecnologia di programmazione web utilizzata per fornire contenuti dinamici.

**HTML:** linguaggio di markup nato per la formattazione e impaginazione di documenti ipertestuali utilizzato principalmente per il disaccoppiamento della struttura logica di una pagina web (definita appunto dal markup) e la sua rappresentazione.

**CSS:** usato separare i contenuti di documenti HTML dalla loro formattazione, permettendo una programmazione più chiara e facile da utilizzare e garantendo il riutilizzo di codice e una più facile manutenzione.

**XML:** un linguaggio marcatore basato su un meccanismo sintattico che consente di definire e controllare il significato degli elementi contenuti in un documento o in un testo.

**Piattaforma:** Definisce l'insieme delle funzionalità fornite dal sistema attraverso l'applicazione web.

**Utente:** Un utilizzatore della piattaforma che non si è ancora registrato.

**Utente registrato:** Un utilizzatore iscritto alla piattaforma.

**Customer manager:** Amministratore che si occupa della gestione degli utenti registrati.

**System manager:** Amministratore che si occupa della gestione di bug, segnalati da utenti che utilizzano la piattaforma.

**Catalogue manager:** Amministratore che si occupa della gestione del catalogo libri.

**Admin:** Generalizzazione di customer manager, catalogue manager e system manager.