



System Design Document



Riferimento	
Versione	1.0
Data	10/12/2021
Destinatario	Prof.ssa F Ferrucci, Prof. F.Palomba
Presentato da	ALESSIA AMATO, ALFONSO MADDALONI, ANTONIO GIORDANO, ENRIQUE CAMACHO GARCÍA, MARIANNA VUJKO, LUCA MORELLI, GIUSEPPE RAGOSTA, MARCO PALMISCIANO
Approvato da	ALESSIA AMATO, ALFONSO MADDALONI, ANTONIO GIORDANO, ENRIQUE CAMACHO GARCÍA, MARIANNA VUJKO, LUCA MORELLI, GIUSEPPE RAGOSTA, MARCO PALMISCIANO



Revision History

Data	Versione	Descrizione	Autori
01/12/2021	v0.1	Stesura di "1: introduzione", "2: Sistema corrente"	tutti
4/12/2021	v0.2	Stesura di "3: Architettura del sistema proposto"	tutti
09/12/2021	v0.3	Stesura condizioni limite	Alessia Amato, Antonio Giordano
9/12/2021	v0.4	Stesura di "4: Servizi dei sottosistemi"	Marianna Vujko, Luca Morelli
10/12/2021	v1.0	Completamento e revisione finale	Alessia Amato, Alfonso Maddaloni, Luca Morelli, Marianna Vujko



Sommario

Revision History	3
Sommario	4
1.Introduzione	7
1.1 Scopo del sistema	7
1.2 Design Goals	8
Criteri di performance	8
Tempo di risposta	8
Memoria	8
Criteri di affidabilità	8
Robustezza	8
Affidabilità	8
Disponibilità	8
Tolleranza ai guasti	8
Sicurezza	9
Criteri di costi	9
Costi di sviluppo	9
Criteri di manutenzione	9
Estensibilità	9
Adattabilità	9
Portabilità	9
Tracciabilità dei requisiti	9
Criteri di usabilità	9
Usabilità	9
Utilità	9
1.2 Design Trade-off	10
1.3 Definizioni, acronimi e abbreviazioni	12



Definizioni	12
Acronimi	13
1.4 Riferimenti	13
1.5 Panoramica del documento	14
1.INTRODUZIONE	14
2.ARCHITETTURA DEL SISTEMA CORRENTE	14
3.ARCHITETTURA DEL SISTEMA PROPOSTO	14
4.SERVIZI DEI SOTTOSISTEMI	14
5.GLOSSARIO	14
2. Architettura del sistema corrente	15
3. Architettura del Sistema Proposto	15
3.1 Panoramica	15
3.2 Decomposizione in sottosistemi	16
Component Diagram	17
Deployment Diagram	18
3.3 Mapping Hardware/Software	18
3.4 Gestione dati persistenti	19
USER	20
BOOK	21
BOOK CLUB	21
BOOKCLUB_MEMBERSHIP	21
BOOKCLUB_USER_INVITE	21
BOOKCLUB_READ_GOAL	22
BOOK_READ_SESSION_ACTIVITY	22
BOOK_READ_SESSION	22
3.5 Controllo degli accessi e sicurezza	22
3.6 Controllo del flusso globale	23
3.7 Condizioni limite	24



Start-up	24
Shut-down	24
Fallimento	24
4. Servizi dei sottosistemi	25
5. Glossario	28



1.Introduzione

1.1 Scopo del sistema

Il sistema Book Club si prefigge lo scopo di avvicinare le persone all'hobby della lettura offrendo una piattaforma utile a confronto e discussione letterari. L'idea nasce a causa della sempre minore propensione dei giovani ad avvicinarsi alla lettura nel tempo libero. Grazie a Book Club, la lettura viene resa più coinvolgente, dinamica e immersa nel sociale con il coinvolgimento di altre persone all'interno dei gruppi di lettura, facilitando ulteriormente gli incontri di chi già prima di usare questo sistema organizzava attività di questo genere.

Gli obiettivi principali del progetto Book Club sono:

- Avvicinare i giovani alla lettura
- Facilitare l'incontro tra persone con gli stessi interessi
- Facilitare scambi di opinioni su libri letti
- Tenersi aggiornati sul progresso della lettura del singolo o del gruppo

I criteri di successo del sistema:

- Branch coverage dei casi di test: almeno 75%
- Buona manutenibilità ed integrità del sistema
- Numero di warning dovuti al check-style inferiore ad una certa soglia



1.2 Design Goals

I Design Goals sono organizzati in cinque categorie: Performance, Dependability, Cost, Maintenance, End User. 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 e tempi di risposta minimi nella esecuzione delle funzionalità offerte.

Memoria

La memoria complessiva del sistema (riguardo lo storage) dipenderà dalla memoria disponibile per la memorizzazione dei dati persistenti.

Criteri di affidabilità

Robustezza

Il sistema informerà l'utente di eventuali input errati attraverso pop-ups di errore e segnalazioni di vario tipo. Gli input saranno validati in diversi strati dell'architettura del software.

Affidabilità

Il sistema dovrà garantire l'affidabilità dei servizi proposti. I risultati visualizzati saranno attendibili rispetto lo stato dei dati persistenti nel momento della richiesta. I libri presenti nel catalogo che l'applicazione mette a disposizione rappresenteranno i medesimi libri nella realtà, con stesso numero di pagine, autore e contenuto. Il processo di login da parte di tutti gli utenti sarà gestito in modo sicuro, assicurando il corretto funzionamento del sistema.

Disponibilità

Una volta online, i servizi e le funzionalità offerte dal sistema saranno disponibili a tutti gli utenti previsti.

Tolleranza ai guasti

Il sistema potrebbe essere soggetto a fallimenti, ma avrà la capacità di rimanere disponibile se questi fallimenti non inficeranno il normale flusso di controllo dell'esecuzione del software.



Sicurezza

L'accesso al sistema sarà protetto da credenziali quali indirizzo email e password. Le password saranno cifrate prima di essere memorizzate nel database, e tutte le comunicazioni remote saranno crittografate tramite TLS.

Criteri di costi

Costi di sviluppo

È stimato un costo complessivo di 400 ore per la progettazione e lo sviluppo del sistema (50 per ogni membro del progetto).

Criteri di manutenzione

Estensibilità

Sarà possibile sia aggiungere nuove funzionalità al software, sia effettuare manutenzione correttiva ed evolutiva, in funzione delle nuove esigenze del cliente o dell'avvento di tecnologie più moderne ed evolute.

Adattabilità

Il sistema sarà messo a disposizione in tutta Europa, ma sarà possibile utilizzarlo anche in paesi non Europei previa inclusione dei libri necessari.

Portabilità

L'interazione dell'utente con il sistema avverrà tramite l'uso di una applicazione mobile. Grazie alla scelta delle tecnologie utilizzate per implementare l'applicazione, sarà possibile installarla e usarla su dispositivi di diversa famiglia.

Tracciabilità dei requisiti

La tracciabilità dei requisiti è chiaramente illustrata in una *matrice di tracciabilità* attraverso la quale è possibile raggiungere un requisito a partire da qualunque altro manufatto del progetto. La tracciabilità è garantita a partire dalla fase di progettazione fino a quella del testing.

Criteri di usabilità

Usabilità

Il sistema sarà di facile comprensione e utilizzo. Permetterà di effettuare in modo semplice e immediato le varie operazioni grazie a un'interfaccia intuitiva, a misura di utente, e un sistema di interazione touch.

Utilità

L'utente sarà supportato nel processo di organizzazione e gestione di book club e in quello di aggiornamento dei propri progressi di lettura. Il sistema



sarà in grado di semplificare i processi suddetti e di permettere all'utente di raggiungere i propri obiettivi più facilmente.

1.2 Design Trade-off

Ciascun design goal è descritto da:

- **priorità:** specifica un valore compreso tra 1 e x (1 è il più alto e x è il più basso)
- **nome:** con un identificatore univoco
- **descrizione:** breve descrizione del design goal
- **criterio:** afferente ad un criterio sopra elencato
- **origine:** va ad identificare i requisiti non funzionali che lo hanno generato
- **trade off:** breve descrizione del trade off e della scelta fatta

Priorità	Nome	Descrizione	Criterio	Origine	Trade off
1	DG_1	Il sistema deve garantire una velocità di risposta non superiore ai 3 secondi.	Performance	RNF5	Performance vs Memoria: Il sistema deve garantire risposte rapide a discapito della memoria utilizzata.
2	DG_2	Il sistema dovrà essere in grado di garantire la corretta usabilità dell'applicazione fino a 10 secondi prima di un crash improvviso.	Affidabilità	RNF2	Affidabilità vs Tempo di risposta: Il sistema dovrà garantire affidabilità nei risultati a discapito del tempo di risposta.
3	DG_3	Il sistema, in caso di malfunzionamento di operazioni non centrali, dovrà essere comunque fruibile agli utenti della piattaforma	Tolleranza ai guasti	RNF2	Costi di sviluppo vs Tolleranza ai guasti: Il sistema dovrà essere fruibile anche in presenza di guasti non di primaria importanza
4	DG_4	Il sistema dovrà essere in grado di gestire picchi di carico dovuti ad almeno 1/3 dei	Prestazioni	RNF3	Affidabilità vs Tempo di risposta: Il sistema dovrà garantire



		partecipanti nella modifica del progresso di lettura.			affidabilità nei risultati a discapito del tempo di risposta.
5	DG_5	Il sistema dovrà essere semplice ed intuitivo, garantendo una facile navigazione anche agli utenti inesperti.	Usabilità	RNF1	
6	DG_6	Il sistema deve garantire una facile estensibilità nell'aggiunta di nuove funzionalità.	Estensibilità	RNF4	<u>Estensibilità vs Performance:</u> L'implementazione del sistema sarà svolta considerando che l'estensibilità sarà preferita al costo di sviluppo.
7	DG_7	Il sistema deve garantire la possibilità di adattarsi facilmente a nuove tecnologie.	Portabilità	RNF4	<u>Portabilità vs Performance:</u> L'implementazione del sistema sarà svolta considerando che la portabilità sarà preferita alla performance.



1.3 Definizioni, acronimi e abbreviazioni

Definizioni

- ❖ **Book club** = gruppo di lettura creato da un **fondatore**
- ❖ **Utente** = persona che usa l'applicazione. Può essere:
 - **ospite**, ossia un utente che non ha ancora effettuato l'accesso tramite le credenziali del proprio account
 - **autenticato**, colui che ha effettuato il login sull'applicazione
 - **fondatore**, ossia un utente non ospite che ha creato almeno un book club: spesso si usa in relazione a un book club, per esempio "il fondatore del book club" in base al contesto
 - **partecipante**, ossia un utente non ospite che partecipa ad almeno un book club: spesso si usa in relazione a un book club, per esempio "il/i partecipante/i del book club" in base al contesto
- ❖ L'account deve essere caratterizzato da:
 - Nome
 - Cognome
 - Email
 - Password
- ❖ **Libro** = prodotto selezionato dal catalogo dalla casa editrice Einaudi
- ❖ **Fondatore** = creatore del BC
- ❖ **Obiettivo di lettura** = numero di pagine da leggere per poter sostenere un incontro tra i partecipanti del Book Club. L'obiettivo di lettura è stabilito dal fondatore del Book Club in questione
- ❖ **Stato obiettivo di lettura** = stato che indica se **tutti** i partecipanti sono arrivati all'obiettivo di lettura indicato dal fondatore
- ❖ **Progresso di lettura** = numero di pagine lette da un partecipante di uno specifico libro
- ❖ **Stato progresso di lettura** = stato che indica se il **singolo** partecipante è arrivato all'obiettivo di lettura di un BC
- ❖ **Catalogo** = insieme di tutti i libri disponibili sull'applicazione
- ❖ **Bacheca** = insieme di tutti i book club a cui si partecipa e a cui ha partecipato

L'email sarà il campo tramite il quale l'utente sarà univocamente identificato.



Acronimi

SDD = System Design Document (Documento di Progettazione del Sistema)

DBMS = DataBase Management System

ACC = Account

BC = book club, creato da un fondatore

CD = Class diagram

DI = Dipartimento di Informatica

NA = Nessuna

UI = User interface

GUI = Graphical User Interface (Interfaccia utente)

DBA = Database Administrator (Amministratore del DataBase)

1.4 Riferimenti

Il documento SDD farà riferimento a:

Requirements Analysis Document

Object-Oriented Software Engineering (Using UML, Patterns, and Java)
Third Edition i cui autori sono Bernd Bruegge & Allen H. Dutoit



1.5 Panoramica del documento

Il seguente documento SDD è composto da:

1.INTRODUZIONE

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

2.ARCHITETTURA DEL SISTEMA CORRENTE

Questa sezione contiene la descrizione del sistema corrente, o un sistema simile di riferimento.

3.ARCHITETTURA DEL SISTEMA PROPOSTO

Questa sezione 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 e le condizioni limite.

4.SERVIZI DEI SOTTOSISTEMI

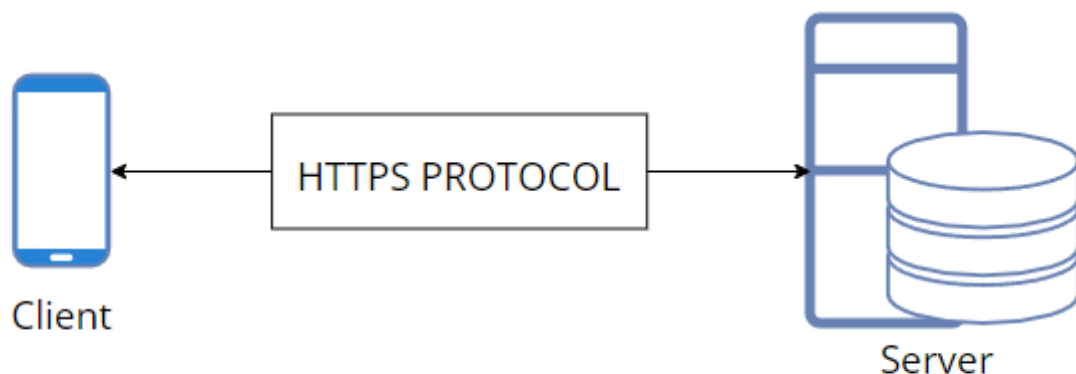
Questa sezione contiene la rappresentazione dei servizi dei sottosistemi.

5.GLOSSARIO

Il documento si conclude con la sezione "GLOSSARIO" nella quale verranno descritti alcuni dei termini utilizzati al fine di eliminare dubbi e ambiguità.

2. Architettura del sistema corrente

Attualmente esiste un sistema per la gestione di club di lettura: Reese's Book Club, un servizio che fornisce un'API Client/Server che usa il protocollo HTTPS per le comunicazioni. Essa si configura come unico competitor della nostra applicazione. Il diagramma riportato in seguito mostra con una certa approssimazione l'architettura del sistema citato.



3. Architettura del Sistema Proposto

3.1 Panoramica

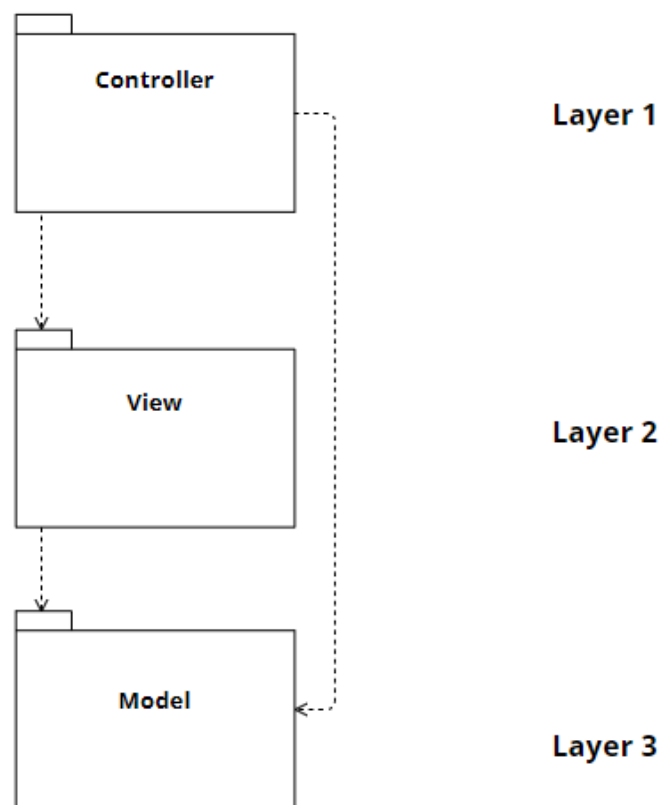
Il sistema Book Club sarà realizzato utilizzando, tra gli altri, principi di Greenfield Engineering. L'innovativo sistema proposto è una applicazione mobile che fornirà funzionalità a diversi tipi di utenti. Le funzioni basilari di registrazione, login, logout, visualizzazione del profilo, visualizzazione della bacheca e accettazione o rifiuto di eventuali inviti di partecipazione a book club sono fornite indistintamente a tutti gli utenti. Un partecipante invece può aggiornare il proprio progresso di lettura (di un libro che sta leggendo rispetto a un determinato book club), visualizzare le informazioni generali dei book club di cui fa parte e quelle relative a obiettivo di lettura e stati di lettura dei partecipanti. Un fondatore, specializzazione di un partecipante a un book club, potrà inviare inviti a book club e gestire gli obiettivi di lettura dei book club che ha creato.

Il sistema farà uso dello stile architetturale Repository per realizzare i vari sottosistemi contenenti logiche di gestione dati persistenti. Dall'impiego del pattern architetturale MVC si trarranno benefici in separazione di logica applicativa, logica di presentazione e logica di persistenza.

3.2 Decomposizione in sottosistemi

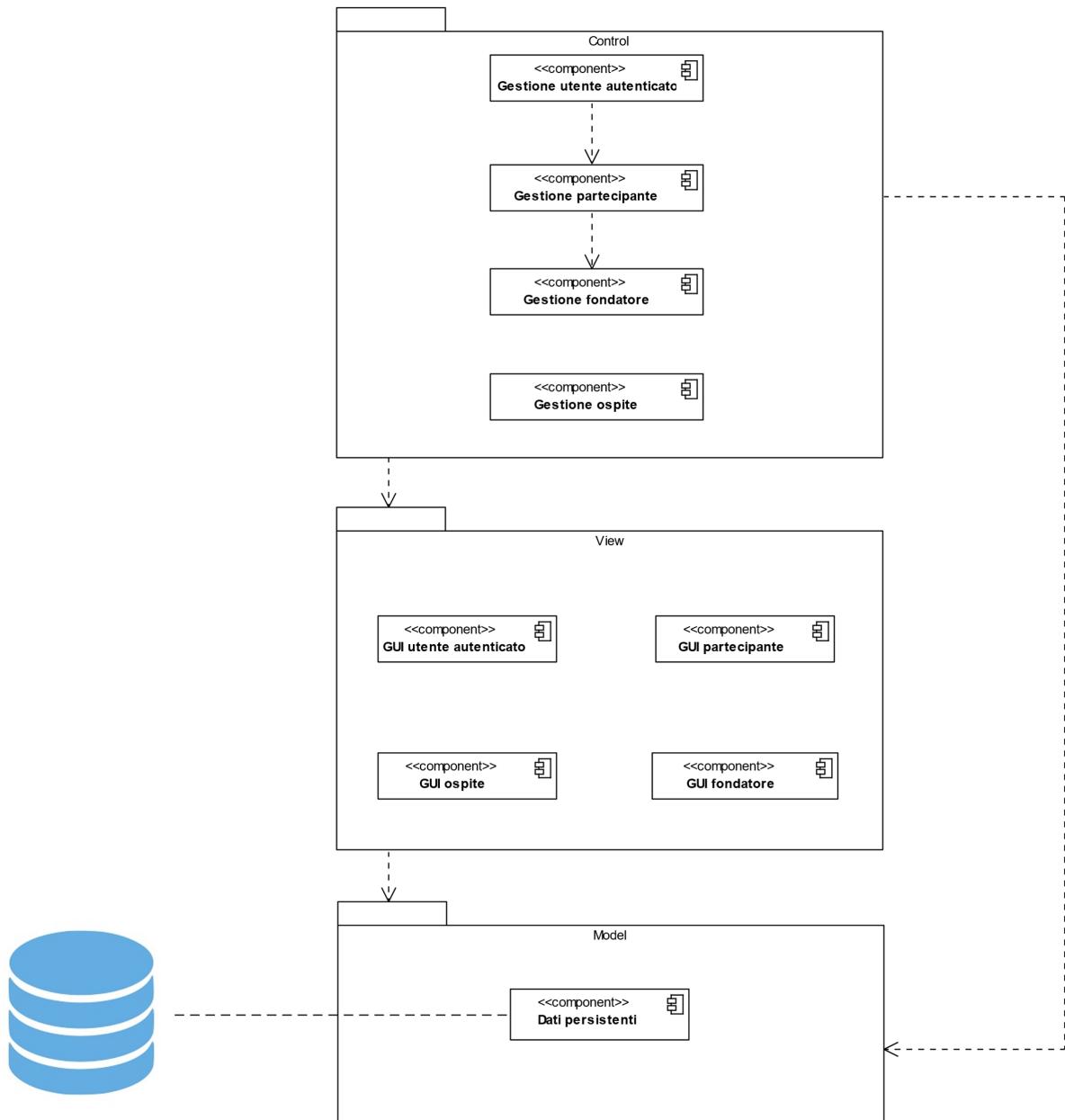
La decomposizione del sistema prevede i seguenti tre layer che si occupano di gestire aspetti e funzionalità differenti:

- **Model:** si occupa di strumenti di accesso e di modellazione delle entità del dominio applicativo
- **Controller:** si occupa dell'esecuzione di logica di gestione Models e di interazione con altri sottosistemi
- **View:** costituisce la logica di presentazione dell'applicazione, composta da elementi di interfaccia grafica e gestione degli eventi innescati su di essi



Seguendo le esigenze del sistema, si è notato che un'architettura aperta semplificherebbe lo sviluppo del sistema, riuscendo comunque a mantenere un basso accoppiamento tra le componenti. Ciascuna di essa inoltre è stata ideata per avere un'alta coesione, permettendo una facile manutenzione qualora ce ne sia bisogno.

Component Diagram



Il **Control layer** contiene 4 sottosistemi:

- Gestione utente autenticato
- Gestione partecipante
- Gestione fondatore
- Gestione ospite

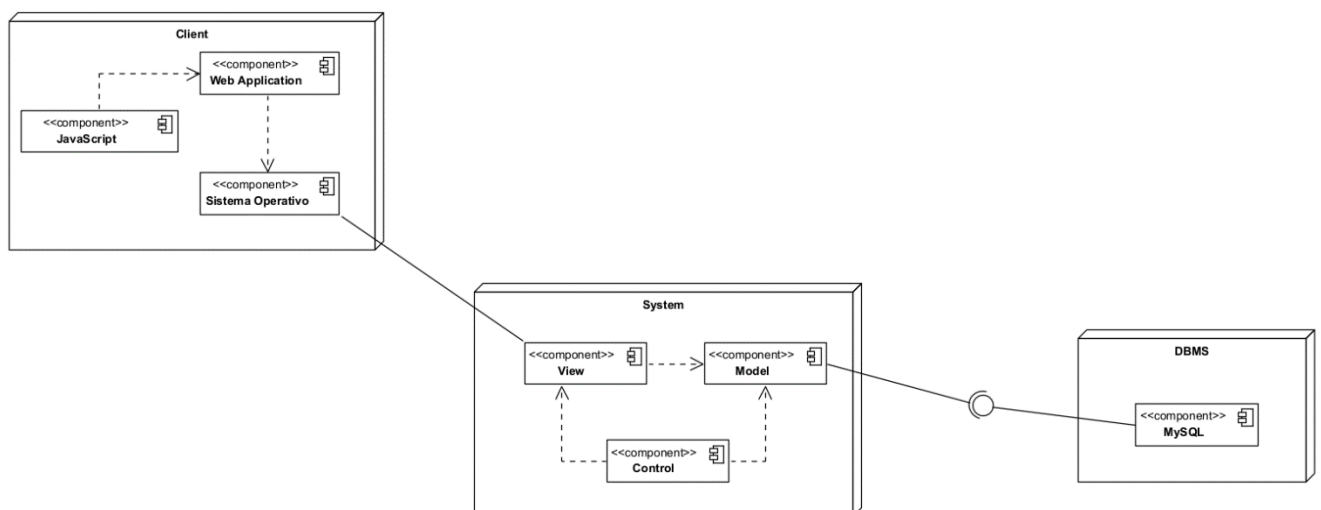
Il **View layer** contiene 4 sottosistemi:

- GUI utente autenticato
- GUI partecipante
- GUI fondatore
- GUI ospite

Il **Model layer** prevede modellazione e gestione dei dati persistenti.

Deployment Diagram

Il Client richiede le funzionalità tramite l'interfaccia che il sistema mette a disposizione. Il tier del Client connette lo strato di view del sistema sul quale vengono eseguite le funzioni apposite al completamento degli obiettivi del Client. La parte DBMS racchiude e gestisce la persistenza dei dati. Il Sistema viene eseguito su Next.js Server.

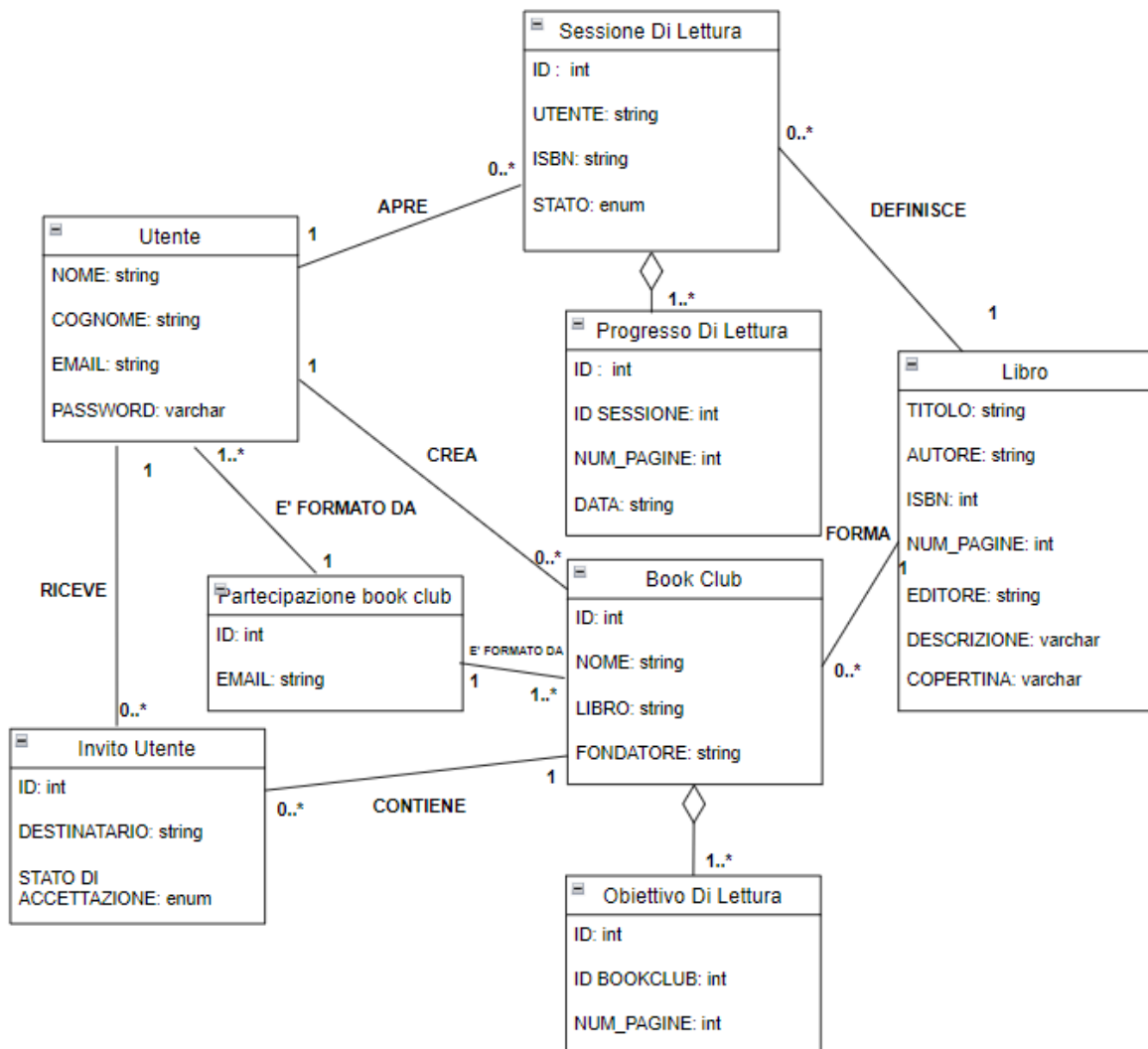


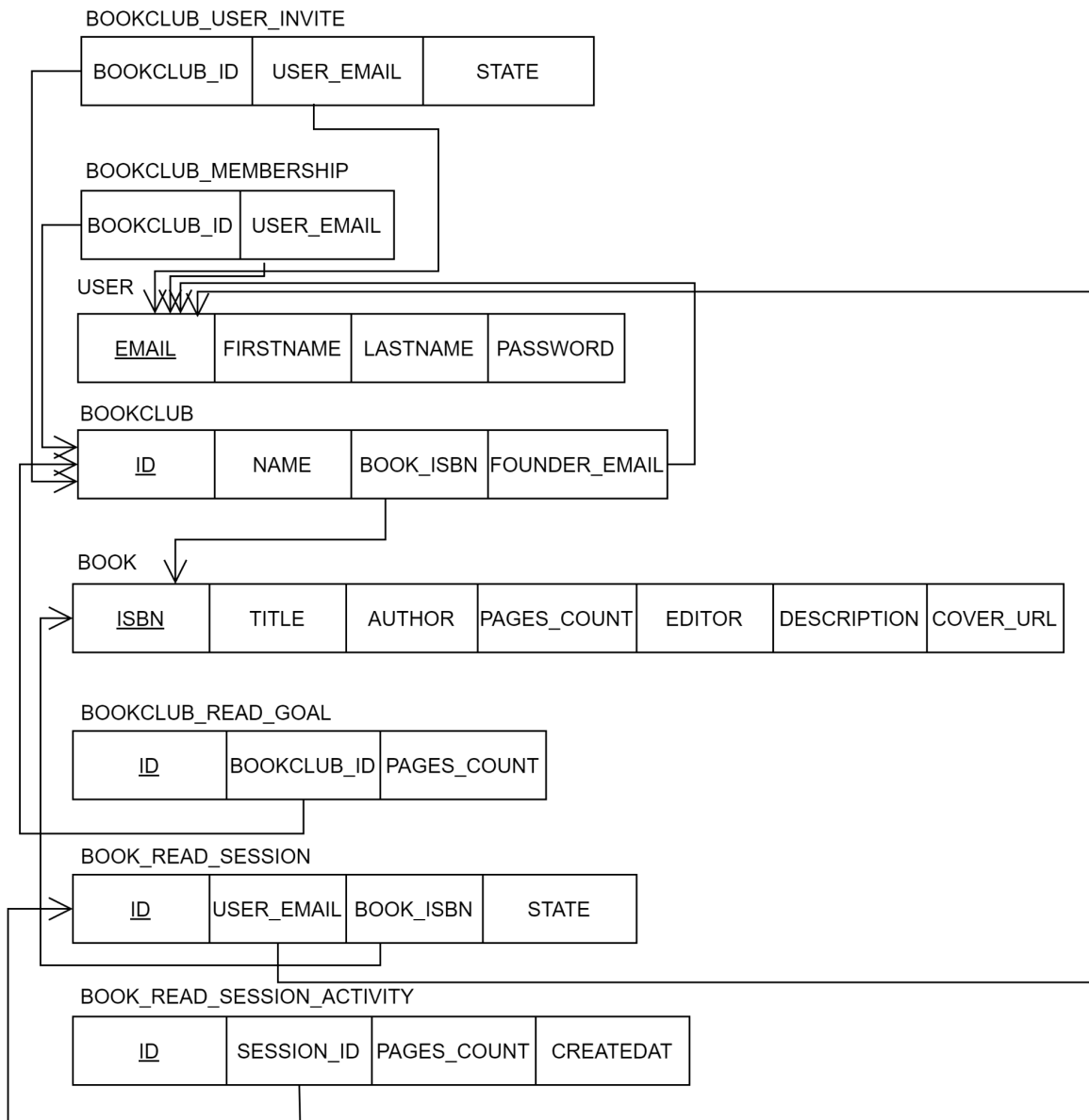
3.3 Mapping Hardware/Software

Il sistema che si vuole sviluppare potrà essere installato su dispositivi mobili. Esso userà un approccio Client/Server, in cui il Client richiede servizi al Server e questo glieli fornisce. Possiamo definire il Client come una macchina attraverso la quale un utente può accedere ad internet e usufruire della piattaforma e di tutte le sue funzionalità. Invece, possiamo definire il Server come il sistema che riceve richieste (tramite il Nest.js Server), gestisce logiche e interagisce con il DBMS. Client e Server comunicano tramite il protocollo HTTP. La gestione dei dati persistenti sarà svolta da un server MySQL, in comunicazione con il Server della applicazione.

3.4 Gestione dati persistenti

Per la gestione dei dati persistenti si è optato per un database relazionale gestito tramite il DBMS MySQL per consentire brevi tempi di risposta e beneficiare del modello relazionale





USER

NOME	TIPO	CONSTRAINTS	KEY
EMAIL	VARCHAR(60)	NOT NULL	PRIMARY KEY
FIRSTNAME	VARCHAR(20)	NOT NULL	
LASTNAME	VARCHAR(20)	NOT NULL	
PASSWORD	VARCHAR(44)	NOT NULL	



BOOK

NOME	TIPO	CONSTRAINTS	KEY
ISBN	VARCHAR(30)	NOT NULL	PRIMARY KEY
TITLE	VARCHAR(100)	NOT NULL	
AUTHOR	VARCHAR(100)	NOT NULL	
PAGES_COUNT	INT	NOT NULL	
EDITOR	VARCHAR(100)	NOT NULL	
DESCRIPTION	TEXT	NOT NULL	
COVER_URL	VARCHAR(512)	NOT NULL	

BOOK CLUB

NOME	TIPO	CONSTRAINTS	KEY
ID	INT	AUTO INCREMENT	PRIMARY KEY
BOOKCLUB_NAME	VARCHAR(100)	NOT NULL	
BOOK_ISBN	VARCHAR(30)	NOT NULL	
FOUNDER_EMAIL	VARCHAR(60)	NOT NULL	FOREIGN KEY

BOOKCLUB_MEMBERSHIP

NOME	TIPO	CONSTRAINTS	KEY
BOOKCLUB_ID	INT	NOT NULL	FOREIGN KEY
USER_EMAIL	VARCHAR(60)	NOT NULL	FOREIGN KEY
MEMBERSHIP_ID	INT	NOT NULL	PRIMARY KEY
STATE	ENUM ('COMPLETED', 'NOT COMPLETED')	NOT NULL	
PAGE_REACHED	INT	NOT NULL	

BOOKCLUB_USER_INVITE

NOME	TIPO	CONSTRAINTS	KEY
INVITED_ID	INT	AUTO_INCREMENT	PRIMARY KEY
BOOKCLUB_ID	INT	NOT NULL	FOREIGN KEY
USER_EMAIL	VARCHAR(60)	NOT NULL	FOREIGN KEY
STATE	ENUM('PENDING', 'ACCEPTED', 'REFUSED')	NOT NULL	



BOOKCLUB_READ_GOAL

NOME	TIPO	CONSTRAINTS	KEY
ID	INT	AUTO_INCREMENT	PRIMARY KEY
BOOKCLUB_ID	INT	NOT NULL	FOREIGN KEY
PAGES_COUNT	INT	NOT NULL	

BOOK_READ_SESSION_ACTIVITY

NOME	TIPO	CONSTRAINTS	KEY
ID	INT	AUTO_INCREMENT	PRIMARY KEY
SESSION_ID	INT	NOT NULL	FOREIGN KEY
PAGES_COUNT	INT	NOT NULL	
CREATEDATE	DATETIME	NOT NULL	

BOOK_READ_SESSION

NOME	TIPO	CONSTRAINTS	KEY
SESSION_ID	INT	AUTO_INCREMENT	PRIMARY KEY
USER_EMAIL	VARCHAR(60)	NOT NULL	FOREIGN KEY
BOOK_ISBN	VARCHAR(30)	NOT NULL	FOREIGN KEY
STATE	ENUM ('ACTIVE', 'INACTIVE')	NOT NULL	

3.5 Controllo degli accessi e sicurezza

Diversi tipi di utenti possono interagire con il sistema Book Club, e in funzione di questi tipi il sistema permette o nega l'uso di determinate funzionalità. Utilizziamo, quindi, una matrice degli accessi per capire, tra tutte le funzionalità, a quali hanno accesso i diversi tipi di utenti. Quando un utente prova ad effettuare un'operazione a lui non permessa, viene mostrato un pop-up di errore. Nella seguente tabella verranno riportate:

- 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.



	Progresso di lettura	Obiettivo di lettura	Book Club	Invito	Ambito Account
Ospite					1)Registra 2)Accede 3)Logout
Utente autenticato			1)Crea	1)Visualizza 2)Accetta 3)Rifiuta	1)Modifica dati personali
Partecipante	1)Inserisce 2)Visualizza 3)Visualizza lo stato	1)Visualizza 2)Visualizza lo stato	1)Esce 2)Visualizza informazioni		
Fondatore		1)Aggiunge 2)Aggiorna	1)Termina	1)Invia 2)Elimina 3)Visualizza stato	

3.6 Controllo del flusso globale

Il controllo del flusso globale del sistema Book Club è di tipo event-driven: non esiste una sequenza di operazioni prestabilita ma è l'utente a scegliere di volta in volta l'operazione da eseguire. L'architettura software infatti prevede un layer di presentazione che permette all'utente di decidere di eseguire operazioni al limite delle funzionalità ad esso associate. Qualora si dovesse ricevere una richiesta di erogazione di un servizio, il layer di logica applicativa elaborerà la richiesta invocando opportuni metodi offerti dai sottosistemi coinvolti e mostrandone il risultato grazie al presentation layer, che provvederà a mostrarlo all'utente finale.



3.7 Condizioni limite

Start-up

Lo start up del sistema consiste nell'avvio dell'applicazione e l'avvio del DBMS per accedere ai dati persistenti memorizzati nel database. Quando sono in esecuzione sia il DBMS che l'applicazione, il sistema è pronto ad interagire con l'utente e soddisfare le sue richieste.

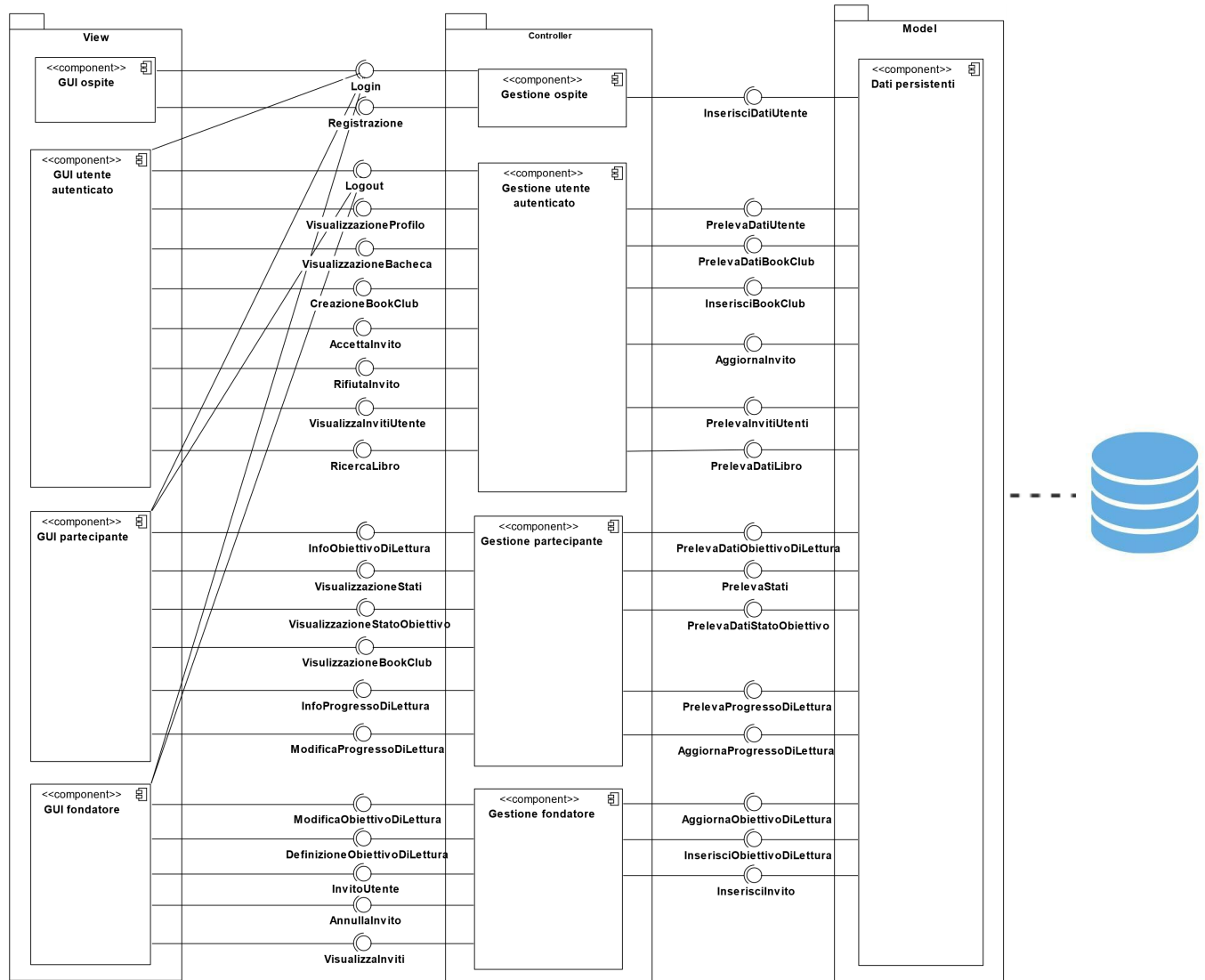
Shut-down

Quando il processo dell'applicazione viene terminato, 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 il dispositivo dovesse arrestarsi improvvisamente, non sarà possibile ripristinare lo stato precedente. Qualsiasi transazione in corso col database viene abortita evitando così di perdere coerenza coi dati;
2. In caso di failure dovute ad errori di implementazione, una pagina di errore sarà mostrata all'utente;
3. In caso di sovraccarico di richieste verso il server, il sistema subirà un rallentamento di esecuzione e potrebbe comportarsi in modo inaspettato;
4. In caso di immissione dati non validi, messaggi di errore saranno mostrati all'utente;
5. In caso di gravi problemi hardware l'applicazione non garantisce la corretta esecuzione delle funzionalità.

4. Servizi dei sottosistemi



Servizi offerti dal controller per il sottosistema view **GUI Ospite:**

- Registrazione
- Login

Servizi offerti dal controller per il sottosistema view **GUI utente autenticato:**

- Login
- Logout



- VisualizzazioneProfilo
- Visualizzazione Bakeca
- CreazioneBookClub
- AccettaInvito
- RifiutaInvito
- VisualizzaInvitiUtente
- RicercaLibro

Servizi offerti dal controller per il sottosistema view **GUI partecipante:**

- Login
- Logout
- InfoObiettivoDiLettura
- VisualizzazioneStati
- VisualizzazioneStatoObiettivo
- VisualizzazioneBookClub
- InfoProgressoDiLettura
- ModificaProgressoDiLettura

Servizi offerti dal controller per il sottosistema view **GUI fondatore:**

- Login
- Logout
- ModificaObiettivoDiLettura
- DefinizioneObiettivoDiLettura
- InvitoUtente
- AnnullaInvito
- VisualizzaInviti

Servizi offerti dal model per il sottosistema controller **Gestione ospite:**

- InserisciDatiUtente
- PrelevaDatiUtente



Servizi offerti dal model per il sottosistema controller **Gestione utente autenticato:**

- InserisciDatiUtente
- PrelevaDatiUtente
- PrelevaDatiBookClub
- InserisciBookClub
- AggiornaInvito
- PrelevaInvitiUtenti
- PrelevaDatiLibro

Servizi offerti dal model per il sottosistema controller **Gestione partecipante:**

- InserisciDatiUtente
- PrelevaDatiObiettivoDiLettura
- PrelevaSatati
- PrelevaDatiStatoObiettivo
- PrelevaProgressoDiLettura
- AggiornaProgressoDiLettura

Servizi offerti dal model per il sottosistema controller **Gestione fondatore:**

- InserisciDatiUtente
- AggiornaInvito
- PrelevaInvitiUtenti
- AggiornaObiettivoDiLettura
- InserisciObiettivoDiLettura
- InserisciInvito



5. Glossario

Di seguito riportiamo la spiegazione di alcuni termini utilizzati nel documento con lo scopo di renderne più agevole la comprensione. Per ulteriori definizioni fare riferimento al paragrafo 1.3.

Sigla/Termine	Definizione
Account	Rappresentazione dell'utente in formato digitale
Piattaforma	Base software o hardware su cui sono sviluppate o eseguite applicazioni
ISBN	Sequenza numerica che identifica un libro e la sua specifica edizione, accettata come standard internazionale
Casa editrice	Società la cui attività principale consiste nella produzione e distribuzione di libri
Client	Rappresentazione dell'utente nella relazione Client/Server