

UNIVERSITÀ DEGLI STUDI DI NAPOLI FEDERICO II  
CORSO DI LAUREA IN INGEGNERIA INFORMATICA  
CORSO DI INGEGNERIA DEL SOFTWARE  
PROF. A.R. FASOLINO - A.A. 2024 - 25



## ***“SISTEMA SOFTWARE PER LA GESTIONE DI UNA BIBLIOTECA DIGITALE”***

**Studenti:**

Bottari Matteo

Di Giorgio Antonio

Ferrante Jacopo

# INDICE

<b>1. SPECIFICHE INFORMALI .....</b>	<b>3</b>
<b>2. ANALISI E SPECIFICA DEI REQUISITI .....</b>	<b>4</b>
2.1 ANALISI NOMI-VERBI .....	4
2.2 REVISIONE DEI REQUISITI .....	5
2.3 GLOSSARIO DEI TERMINI .....	6
2.4 CLASSIFICAZIONE DEI REQUISITI .....	6
2.4.1 <i>Requisiti funzionali</i> .....	6
2.4.2 <i>Requisiti sui dati</i> .....	8
2.4.3 <i>Vincoli / Altri requisiti</i> .....	8
2.5 MODELLAZIONE DEI CASI D'USO .....	9
2.5.1 <i>Attori e casi d'uso</i> .....	9
MATRICE DI TRACCIABILITÀ: .....	11
2.5.2 <i>Diagramma dei casi d'uso</i> .....	12
2.5.3 <i>Scenari</i> .....	13
2.6 DIAGRAMMA DELLE CLASSI .....	19
2.6.1 <i>Responsabilità</i> .....	20
2.7 DIAGRAMMI DI SEQUENZA .....	22
2.7.1 <i>PrenotaLibroDisponibile</i> .....	22
2.7.2 <i>AggiungiLibro</i> .....	22
2.7.3 <i>RegistraRitiroCopiaPrenotata</i> .....	23
2.7.4 <i>RegistrazioneUtente</i> .....	24
2.7.5 <i>AggiungiCopia</i> .....	25
2.8 DIAGRAMMA DELLE CLASSI RAFFINATO .....	26
<b>3. PIANO DI TEST FUNZIONALE .....</b>	<b>27</b>
3.1 REGISTRAZIONE UTENTE .....	27
3.2 AGGIUNTA LIBRI .....	29
3.3 PRENOTA LIBRO DISPONIBILE .....	33
<b>4. PROGETTAZIONE .....</b>	<b>34</b>
4.1 DIAGRAMMA DELLE CLASSI .....	34
4.1.1 <i>Pattern BCED</i> .....	35
4.1.1.1 <i>Package Boundary</i> .....	35
4.1.1.2 <i>Package Controller</i> .....	37
4.1.1.3 <i>Package Entity</i> .....	38
4.1.1.4 <i>Package Database</i> .....	39
Progettazione del database: modello E/R e traduzione delle associazioni .....	40
4.2 DIAGRAMMI DI SEQUENZA .....	41
4.2.1 <i>Registrazione Utente</i> .....	41
4.2.2 <i>Prenotazione Libro</i> .....	42
4.2.3 <i>Inserimento Libro</i> .....	43
<b>5. IMPLEMENTAZIONE .....</b>	<b>44</b>
5.1 PACKAGE DATABASE .....	44
5.2 PACKAGE ENTITY .....	45
5.3 PACKAGE CONTROLLER .....	46
5.4 PACKAGE BOUNDARY .....	46
5.5 PACKAGE DTO .....	47
5.6 INTERFACCIE GRAFICHE DA ECLIPSE (CON JAVASWING) .....	47
5.6.1 <i>Registrazione Utente</i> .....	47
5.6.2 <i>Prenotazione Libro</i> .....	48
5.6.3 <i>Inserimento Libro</i> .....	50
5.7 DIAGRAMMA DI DEPLOYMENT .....	51
<b>6. TEST FUNZIONALE .....</b>	<b>52</b>
6.1 REGISTRAZIONE UTENTE .....	52
6.2 PRENOTAZIONE LIBRO .....	55
6.3 INSERIMENTO LIBRO .....	56

# 1. Specifiche informali

Si desidera sviluppare un sistema software per la gestione di una biblioteca digitale che consenta agli utenti di cercare e prenotare libri online e ritirarli in sede.

*La biblioteca avrà un catalogo digitale che gli utenti potranno consultare, attraverso opportuna interfaccia grafica, per trovare libri di loro interesse, utilizzando filtri come autore, genere, o anno di pubblicazione. Ogni libro nel catalogo avrà una scheda dettagliata che include il titolo, l'autore, l'anno di pubblicazione, il genere, il codice ISBN, una breve descrizione e la disponibilità attuale di copie di quel libro. Ogni copia è identificata da un identificativo numerico univoco e può assumere lo stato di disponibile, prenotata, o in prestito.*

*Gli utenti potranno registrarsi al sito fornendo i propri dati personali, tra cui nome, cognome, indirizzo e-mail e una password. Una volta completata la registrazione, potranno accedere, tramite autenticazione, al proprio profilo personale. All'interno del profilo sarà possibile visualizzare e modificare i dati personali, caricare un'immagine del profilo, consultare l'elenco dei libri prenotati e la cronologia di quelli precedentemente consultati.*

*Dal profilo personale, inoltre, l'utente potrà accedere al catalogo dei libri disponibili nella biblioteca, al fine di poter prenotare eventualmente un libro di interesse, e gestire le prenotazioni attive, con la possibilità di annullarle in qualsiasi momento.*

*Quando un utente prenota un libro, il sistema ne verificherà la disponibilità di una copia e, in caso affermativo, la riserverà per l'utente. Quando un libro viene prenotato viene generato una ricevuta della prenotazione, che include, tra le altre informazioni, un ID della prenotazione, data di prevista consegna, e costo complessivo del prestito (calcolato in base al numero di giorni).*

*Gli addetti alla biblioteca invece hanno una interfaccia grafica riservata che gli permette di visualizzare l'elenco delle prenotazioni dei libri e di procedere a registrare il ritiro di una copia prenotata. All'atto del ritiro, l'utente fornirà l'ID della prenotazione ricevuto e l'addetto potrà cambiare lo stato della copia in "non disponibile".*

*Una ulteriore interfaccia invece consente agli addetti di generare un report di tutti i libri che sono in prestito, riportando per ciascun libro il numero di copie in prestito e quelle disponibili e di poter registrare la avvenuta restituzione di una copia di un libro alla biblioteca (attraverso l'ID della prenotazione), rendendola di nuovo disponibile a future prenotazioni.*

*La biblioteca digitale consentirà anche agli amministratori di gestire il catalogo aggiungendo nuovi libri, nuove copie di libri, modificando quelli esistenti o eliminando quelli obsoleti. Inoltre, gli amministratori potranno generare report mensili sull'uso del servizio (incasso), quali i libri più prenotati, gli utenti più attivi e le tendenze di lettura (genere più letto in un determinato intervallo di tempo).*

## 2. Analisi e specifica dei requisiti

### 2.1 Analisi nomi-verbi

LEGENDA:

Classe

Attributo

Funzionalità

Attore

Classe-Attore

La biblioteca avrà un catalogo digitale che gli utenti potranno consultare, attraverso opportuna interfaccia grafica, per trovare libri di loro interesse, utilizzando filtri come autore, genere, o anno di pubblicazione. Ogni libro nel catalogo avrà una scheda dettagliata che include il titolo, l'autore, l'anno di pubblicazione, il genere, il codice ISBN, una breve descrizione e la disponibilità attuale di copie di quel libro. Ogni copia è identificata da un identificativo numerico univoco e può assumere lo stato di disponibile, prenotata, o in prestito.

Gli utenti potranno registrarsi al sito fornendo i propri dati personali, tra cui nome, cognome, indirizzo e-mail e una password. Una volta completata la registrazione, potranno accedere, tramite autenticazione, al proprio profilo personale. All'interno del profilo sarà possibile visualizzare e modificare i dati personali, caricare un'immagine del profilo, consultare l'elenco dei libri prenotati e la cronologia di quelli precedentemente consultati.

Dal profilo personale, inoltre, l'utente potrà accedere al catalogo dei libri disponibili nella biblioteca, al fine di poter prenotare eventualmente un libro di interesse, e gestire le prenotazioni attive, con la possibilità di annullarle in qualsiasi momento.

Quando un utente prenota un libro, il sistema ne verificherà la disponibilità di una copia e, in caso affermativo, la riserverà per l'utente. Quando un libro viene prenotato viene generata una ricevuta della prenotazione, che include, tra le altre informazioni, un ID della prenotazione, data di prevista consegna, e costo complessivo del prestito (calcolato in base al numero di giorni).

Gli addetti alla biblioteca invece hanno una interfaccia grafica riservata che gli permette di visualizzare l'elenco delle prenotazioni dei libri e di procedere a registrare il ritiro di una copia prenotata. All'atto del ritiro, l'utente fornirà l'ID della prenotazione ricevuto e l'addetto potrà cambiare lo stato della copia in "non disponibile".

Una ulteriore interfaccia invece consente agli addetti di generare un report di tutti i libri che sono in prestito, riportando per ciascun libro il numero di copie in prestito e quelle disponibili e di poter registrare la avvenuta restituzione di una copia di un libro alla biblioteca (attraverso l'ID della prenotazione), rendendola di nuovo disponibile a future prenotazioni.

La biblioteca digitale consentirà anche agli amministratori di gestire il catalogo aggiungendo nuovi libri, nuove copie di libri, modificando quelli esistenti o eliminando quelli obsoleti. Inoltre, gli amministratori potranno generare report mensili sull'uso del servizio (incasso), quali i libri più prenotati, gli utenti più attivi e le tendenze di lettura (genere più letto in un determinato intervallo di tempo).

“Gli utenti potranno registrarsi al sito” / “Utente con profilo personale” = Utente registrato

## 2.2 Revisione dei requisiti

---

A seguito del colloquio con gli stakeholders abbiamo chiarito punti progettuali critici fondamentali per una corretta progettazione e analisi.

1. *Il sistema deve offrire un catalogo digitale consultabile dagli Utenti*
2. *Il sistema deve offrire la possibilità di trovare libri tramite filtri all'interno del catalogo (si filtra per autore, genere, anno di pubblicazione).*
3. *Di ogni libro si vuole memorizzare il titolo, l'autore, l'anno di pubblicazione, il genere, il codice ISBN, una breve descrizione e la disponibilità*
4. *Di ogni copia si vuole memorizzare un identificativo numerico univoco e lo stato di disponibilità o meno*
5. *Il sistema deve offrire la possibilità di registrarsi agli Utenti*
6. *L'utente per registrarsi deve fornire nome, cognome, e-mail e password, il Sistema memorizzerà tali dati*
7. *Il sistema deve dare la possibilità agli utenti registrati di accedere al proprio profilo tramite autenticazione*
8. *Il sistema deve dare la possibilità agli utenti registrati di poter modificare dati personali e caricare un'immagine del profilo.*
9. *Il sistema deve dare la possibilità all'utente registrato di consultare l'elenco dei libri prenotati e la cronologia di quelli precedentemente consultati*
10. *Dal profilo personale l'utente potrà accedere al catalogo dei libri disponibili nella biblioteca*
11. *Dal profilo personale l'utente può prenotare libri e gestire le prenotazioni attive, con la possibilità di annullarle.*
12. *Il sistema, a seguito di una prenotazione, verifica la disponibilità effettiva della copia, e nel caso la riserverà all'utente.*
13. *A seguito di una prenotazione, il Sistema genera una ricevuta della prenotazione*
14. *La ricevuta della prenotazione memorizza un ID univoco, data di prevista consegna e costo complessivo del prestito.*
15. *Gli addetti alla biblioteca possono visualizzare l'elenco delle prenotazioni dei libri tramite interfaccia grafica X*
16. *Gli addetti alla biblioteca possono registrare il ritiro di una copia prenotata tramite interfaccia grafica X*
17. *Gli addetti alla biblioteca possono generare un report di tutti i libri che sono in prestito, riportando per ciascun libro il numero di copie in prestito e quelle disponibili, e potranno registrare l'avvenuta restituzione di libri, il tutto tramite interfaccia grafica Y*
18. *Il sistema deve dare la possibilità agli amministratori di gestire il catalogo aggiungendo nuovi libri, nuove copie, modificare libri ed eliminare libri obsoleti*
19. *Il sistema deve dare la possibilità agli amministratori di generare report mensili sull'uso del servizio (incassi), quali sono i libri più prenotati e gli utenti più attivi, e visualizzare il genere letterario di tendenza (Per "utenti più attivi", si intende per numero di prenotazioni)*

## 2.3 Glossario dei termini

Glossario dei termini in una tabella termine-descrizione (significato), con eventuali sinonimi:

Termine	Descrizione	Sinonimi
<b>Utente</b>	Una generica persona che intende consultare il catalogo per trovare libri di proprio interesse	---
<b>Utente registrato</b>	Una generica persona che al seguito della registrazione potrà anche accedere al suo profilo e modificarne i dati	---
<b>Catalogo digitale</b>	Insieme di libri presenti in biblioteca che gli utenti possono visualizzare	Catalogo, Biblioteca digitale
<b>Libro</b>	Uno dei libri facenti parte del catalogo	---
<b>Copia</b>	Effettiva copia del libro presente nel catalogo digitale eventualmente prenotabile	---
<b>Addetto alla biblioteca</b>	Impiegato della biblioteca, con più permessi dei clienti ma meno permessi degli amministratori	Addetti
<b>Amministratore</b>	Personale della biblioteca che ricopre ruolo amministrativo e dirigenziale, con diretto accesso ai bilanci e dati sensibili	---

## 2.4 Classificazione dei requisiti

Classificazione dei requisiti, di seguito riportati i requisiti funzionali (RF), i requisiti sui dati (RD) e altri vincoli e requisiti di altri tipi (VR) (vincoli sull'interfaccia, sulla gestione dei dati, sulle regole di business e sull'usabilità). Ogni requisito ha una codifica univoca al fine di essere ripreso nella documentazione per motivare scelte progettuali e di implementazione:

### 2.4.1 Requisiti funzionali

ID	Requisito	Origine (n. frase dei requisiti revisionati)
<b>RF01</b>	Il sistema deve offrire un catalogo digitale consultabile dagli Utenti	1
<b>RF02</b>	Il sistema deve offrire la possibilità agli utenti di trovare libri filtrando tramite autore	1
<b>RF03</b>	Il sistema deve offrire la possibilità agli utenti di trovare libri filtrando tramite genere letterario	1
<b>RF04</b>	Il sistema deve offrire la possibilità agli utenti di trovare libri filtrando tramite anno di pubblicazione del libro	1
<b>RF05</b>	Il sistema deve offrire la possibilità agli Utenti di registrarsi fornendo i propri dati personali	4
<b>RF06</b>	Il sistema deve dare la possibilità agli utenti registrati di accedere al proprio profilo tramite autenticazione	5



<b>RF07</b>	Il sistema deve dare la possibilità agli utenti registrati di poter modificare dati personali	6
<b>RF08</b>	Il sistema deve dare la possibilità agli utenti registrati di poter caricare la propria immagine di profilo	6
<b>RF09</b>	Il sistema deve dare la possibilità all'utente registrato di consultare l'elenco dei libri prenotati	6
<b>RF10</b>	Il sistema deve dare la possibilità all'utente registrato di visualizzare la cronologia dei libri precedentemente consultati	6
<b>RF11</b>	Dal profilo personale l'utente registrato potrà accedere al catalogo dei libri disponibili nella biblioteca al fine di prenotare un libro di interesse	7
<b>RF12</b>	L'utente registrato, dal proprio profilo, potrà annullare le prenotazioni attive	7
<b>RF13</b>	Il sistema, a seguito di una prenotazione, verifica la disponibilità effettiva della copia	8
<b>RF14</b>	Il sistema, a seguito di una prenotazione, data la disponibilità della copia prenotata, riserverà tale copia all'utente	8
<b>RF15</b>	A seguito di una prenotazione, il Sistema genera una ricevuta della prenotazione	9
<b>RF16</b>	Gli addetti alla biblioteca possono visualizzare l'elenco delle prenotazioni dei libri, tramite interfaccia grafica X	10
<b>RF17</b>	Gli addetti alla biblioteca possono registrare il ritiro di una copia prenotata, tramite interfaccia grafica X	10
<b>RF18</b>	Gli addetti alla biblioteca, a seguito di una prenotazione, potranno cambiare lo stato di una copia in "non disponibile", tramite interfaccia grafica X	11
<b>RF19</b>	Gli addetti alla biblioteca possono generare un report di tutti i libri che sono in prestito, riportando per ciascun libro il numero di copie in prestito e quelle disponibili, tramite interfaccia grafica Y	12
<b>RF20</b>	Gli addetti alla biblioteca potranno registrare l'avvenuta restituzione di copie di libri della biblioteca, rendendola di nuovo disponibile a future prenotazioni, tramite interfaccia grafica Y	12
<b>RF21</b>	Il sistema deve dare la possibilità agli amministratori di gestire il catalogo aggiungendo nuovi libri tramite interfaccia	13
<b>RF22</b>	Il sistema deve dare la possibilità agli amministratori di gestire il catalogo aggiungendo nuove copie	13
<b>RF23</b>	Il sistema deve dare la possibilità agli amministratori di gestire il catalogo modificando i libri esistenti	13
<b>RF24</b>	Il sistema deve dare la possibilità agli amministratori di gestire il catalogo eliminando i libri obsoleti	13
<b>RF25</b>	Il sistema deve dare la possibilità agli amministratori di generare report mensili sull'uso del servizio (incassi), quali sono i libri più prenotati e gli utenti più attivi, e visualizzare il genere letterario di tendenza	14

### 2.4.2 Requisiti sui dati

ID	Requisito	Origine (n. frase dei requisiti revisionati)
<b>RD01</b>	Di ogni libro si vuole memorizzare il titolo, l'autore, l'anno di pubblicazione, il genere, il codice ISBN, una breve descrizione e la disponibilità (numero copie)	2
<b>RD02</b>	Di ogni copia si vuole memorizzare un identificativo numerico univoco e lo stato di disponibilità o meno	3
<b>RD03</b>	L'utente per registrarsi deve fornire nome, cognome, e-mail e password, il Sistema memorizzerà tali dati	4
<b>RD04</b>	La ricevuta della prenotazione memorizza un ID univoco, data di prevista consegna e costo complessivo del prestito	9

### 2.4.3 Vincoli / Altri requisiti

ID	Requisito	Origine (n. frase dei requisiti revisionati)
<b>V01</b>	Il sistema deve avere un'interfaccia grafica per permettere agli utenti di consultare il catalogo	1
<b>V02</b>	Ogni copia è identificata da un identificativo numerico univoco	3
<b>V03</b>	Le copie possono assumere solo tre stati specifici: disponibile, prenotata o in prestito.	3
<b>V04</b>	Un libro è prenotabile solamente se disponibile almeno una copia nel catalogo	7
<b>V05</b>	Gli addetti interagiscono con il sistema tramite doppia interfaccia grafica, separate da quella degli utenti, con funzionalità specifiche	10
<b>V06</b>	Restituzione di una copia da parte degli addetti avviene "attraverso l'ID della prenotazione", vincolo sul processo di restituzione	11
<b>V07</b>	Per certificare la validità delle credenziali inserite dall'Utente in fase di autenticazione, deve essere disponibile il Servizio di posta elettronica	5
<b>V08</b>	Una prenotazione può riferirsi solamente ad un unico libro	7
<b>V09</b>	Una prenotazione può essere effettuata solo da un utente registrato	7



## 2.5 Modellazione dei casi d'uso

---

### 2.5.1 Attori e casi d'uso

#### **Attori Primari:**

- Utente
- Utente Registrato
- Addetto alla biblioteca
- Amministratore

#### **Casi d'uso:**

- **UC1**: MostraCatalogo
- **UC2**: FiltraCatalogo
- **UC3**: RegistrazioneUtente
- **UC4**: AccessoProfiloUtente
- **UC5**: ModificaProfilo
- **UC6**: ElencaPrenotazioniUtente
- **UC7**: VisualizzaCronologiaLibriConsultati
- **UC8**: PrenotaLibroDisponibile
- **UC9**: AnnullaPrenotazione
- **UC10**: VisualizzaPrenotazioniComplessive
- **UC11**: RegistraRitiroCopiaPrenotata
- **UC12**: RegistraRestituzioneCopia
- **UC13**: GeneraReportPrestiti
- **UC14**: GestisciCatalogo
- **UC15**: AggiungiLibro
- **UC16**: AggiungiCopia
- **UC17**: ModificaLibro
- **UC18**: EliminaLibro
- **UC19**: GeneraReportMensile

#### **Casi d'uso di inclusione:**

- **UC20**: MostraLibriDisponibili

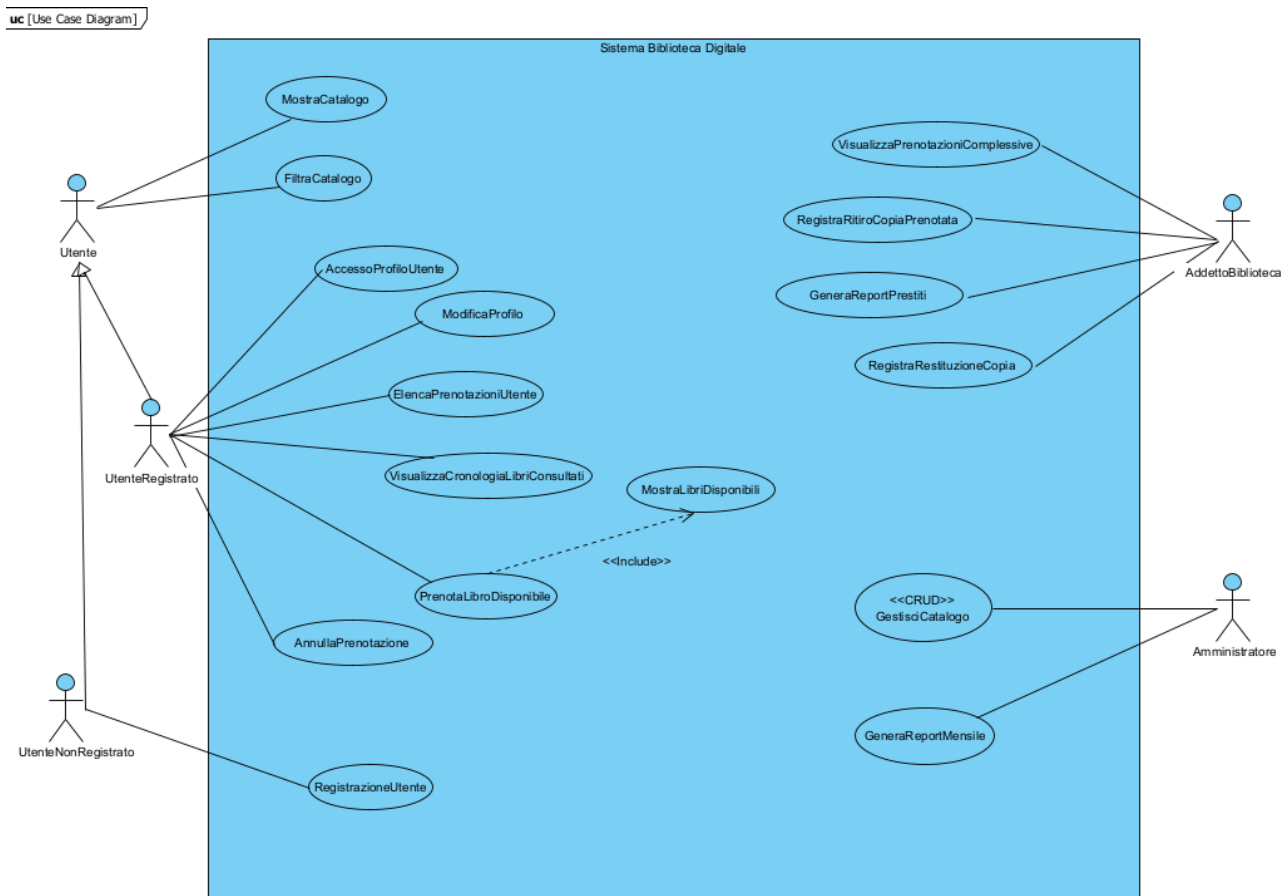
Caso d'uso	Attori Primari	Attori Secondari	Incl. / Ext.	Requisiti corrispondenti
<u>UC1:</u> MostraCatalogo	Utente	-	-	RF01
<u>UC2:</u> FiltraCatalogo	Utente	-	-	RF02, RF03, RF04
<u>UC3:</u> RegistrazioneUtente	UtenteNonRegistrato	-	-	RF05
<u>UC4:</u> AccessoProfiloUtente	UtenteRegistrato	-	-	RF06
<u>UC5:</u> ModificaProfilo	UtenteRegistrato	-	-	RF07, RF08
<u>UC6:</u> ElencaPrenotazioniUtente	UtenteRegistrato	-	-	RF09
<u>UC7:</u> VisualizzaCronologiaLibri Consultati	UtenteRegistrato	-	-	RF10
<u>UC8:</u> PrenotaLibroDisponibile	UtenteRegistrato	-	-	RF11, RF14, RF15, RF18
<u>UC9:</u> AnnullaPrenotazione	UtenteRegistrato	-	-	RF12
<u>UC10:</u> VisualizzaPrenotazioniComplessive	AddettoBiblioteca	-	-	RF16
<u>UC11:</u> RegistraRitiroCopia-Prenotata	AddettoBiblioteca	-	-	RF17
<u>UC12:</u> RegistraRestituzioneCopia	AddettoBiblioteca	-	-	RF20
<u>UC13:</u> GeneraReportPrestiti	AddettoBiblioteca	-	-	RF19
<u>UC14:</u> GestisciCatalogo	Amministratore	-	-	RF21, RF22, RF23, RF24
<u>UC15:</u> AggiungiLibro	Amministratore	-	-	RF21
<u>UC16:</u> AggiungiCopia	Amministratore	-	-	RF22
<u>UC17:</u> ModificaLibro	Amministratore	-	-	RF23
<u>UC18:</u> EliminaLibro	Amministratore	-	-	RF24
<u>UC19:</u> GeneraReportMensile	Amministratore	-	-	RF25
<u>UC20:</u> MostraLibriDisponibili	-	-	Incluso in PrenotaLibroDisponibile	RF13

## Matrice di tracciabilità:

La copertura tra casi d'uso e requisiti del sistema è soddisfatta, in quanto tutte le righe e tutte le colonne hanno almeno una corrispondenza.

		Caso D'uso																			
		UC1	UC2	UC3	UC4	UC5	UC6	UC7	UC8	UC9	UC10	UC11	UC12	UC13	UC14	UC15	UC16	UC17	UC18	UC19	UC20
Requisito	R1	X																			
	R2		X																		
	R3		X																		
	R4		X																		
	R5			X																	
	R6				X																
	R7					X															
	R8					X															
	R9						X														
	R10							X													
	R11								X												
	R12									X											
	R13																				X
	R14								X												
	R15								X												
	R16									X											
	R17										X										
	R18								X												
	R19													X							
	R20												X								
	R21														X	X					
	R22														X		X				
	R23														X			X			
	R24														X				X		
	R25																			X	

## 2.5.2 Diagramma dei casi d'uso



Nota: nel caso d'uso dell'Amministratore " <<CRUD>> GestisciCatalogo " sono racchiusi i casi d'uso per la creazione, l'aggiornamento e l'eliminazione dei Libri (AggiungiLibro, AggiungiCopia ed EliminaLibro) e la creazione di una Copia (AggiungiCopia).

## 2.5.3 Scenari

Caso d'uso:	RegistrazioneUtente
Attore primario	UtenteNonRegistrato
Attore secondario	-
Descrizione	Un utente non registrato effettua una richiesta di registrazione di un nuovo account, il sistema verifica che le credenziali siano valide e in seguito registra il nuovo account
Pre-Condizioni	-
Sequenza di eventi principale	<ol style="list-style-type: none"> <li>1. Il caso d'uso inizia quando l'utente non registrato richiede la registrazione</li> <li>2. L'utente non registrato inserisce nome, cognome, e-mail e password</li> <li>3. Il sistema verifica che il formato dei dati sia valido</li> <li>4. <b>if</b> il formato dei dati è valido <ol style="list-style-type: none"> <li>4.1. Il sistema crea l'account con i corrispettivi dati</li> <li>4.2. Il sistema comunica all'utente che l'account è stato creato con successo</li> </ol> </li> </ol>
Post-Condizioni	<p>L'utente è registrato al sistema</p> <p>È stato creato un nuovo account associato all'indirizzo e-mail fornito</p>
Casi d'uso correlati	-
Sequenza di eventi alternativi	Al punto 4, se la password, il nome, il cognome o l'e-mail inserita non è valida, il sistema restituisce un messaggio d'errore e viene chiesto all'utente di riprovare, ritornando al punto 2

Caso d'uso:	AccessoProfiloUtente
Attore primario	UtenteRegistrato
Attore secondario	-
Descrizione	Un utente registrato tenta di fare l'accesso al proprio account, il sistema verifica che le credenziali siano valide, se lo sono l'accesso è riuscito, altrimenti verrà chiesto di riprovare
Pre-Condizioni	L'utente è registrato al sistema
Sequenza di eventi principale	<ol style="list-style-type: none"> <li>1. Il caso d'uso inizia quando l'utente registrato vuole accedere all'account</li> <li>2. L'utente registrato inserisce e-mail e password</li> <li>3. Il sistema verifica che i dati siano validi</li> <li>4. <b>if</b> il sistema conferma la validità dei dati <ol style="list-style-type: none"> <li>4.1. Il login viene effettuato con successo</li> <li>4.2. L'utente è reindirizzato alla propria area personale del profilo</li> </ol> </li> </ol>
Post-Condizioni	L'utente è loggato nel sistema della biblioteca digitale
Casi d'uso correlati	-
Sequenza di eventi alternativi	Al punto 4, se la password o l'e-mail inserita non sono valide, il sistema restituisce un messaggio d'errore e viene chiesto all'utente di riprovare, ritornando al punto 2

<b>Caso d'uso:</b>	<b>ModificaProfilo</b>
<b>Attore primario</b>	UtenteRegistrato
<b>Attore secondario</b>	-
<b>Descrizione</b>	L'utente registrato, una volta eseguito l'accesso al profilo personale, potrà visualizzare e modificare i propri dati personali e/o caricare una nuova immagine del profilo
<b>Pre-Condizioni</b>	L'utente registrato ha effettuato correttamente il login al suo account
<b>Sequenza di eventi principale</b>	<ol style="list-style-type: none"> <li>1. L'utente accede all'area personale</li> <li>2. Modifica i suoi dati personali e/o la sua immagine di profilo</li> <li>3. Il sistema verifica che le modifiche apportate dall'utente siano valide</li> <li>4. <b>if</b> le modifiche sono valide <ol style="list-style-type: none"> <li>4.1. Il profilo dell'utente è aggiornato</li> <li>4.2. L'utente riceve un messaggio di conferma avvenuta modifica</li> </ol> </li> </ol>
<b>Post-Condizioni</b>	Il profilo dell'utente è aggiornato con i dati inseriti
<b>Casi d'uso correlati</b>	-
<b>Sequenza di eventi alternativi</b>	Al punto 3, se le informazioni personali dell'utente o l'immagine di profilo inserita non sono valide, il sistema mostra un messaggio di errore e verrà chiesto all'utente di riprovare, ritornando al punto 2

<b>Caso d'uso:</b>	<b>RegistraRitiroCopiaPrenotata</b>
<b>Attore primario</b>	AddettoBiblioteca
<b>Attore secondario</b>	-
<b>Descrizione</b>	Un addetto alla biblioteca registra il ritiro di una copia di un libro prenotata da un utente registrato
<b>Pre-Condizioni</b>	L'utente registrato ha effettuato una prenotazione. L'addetto alla biblioteca è autenticato nel sistema.
<b>Sequenza di eventi principale</b>	<ol style="list-style-type: none"> <li>1. L'utente fornisce all'addetto della biblioteca l'ID della prenotazione precedentemente effettuata</li> <li>2. L'addetto alla biblioteca accede all'interfaccia dedicata alle prenotazioni attive</li> <li>3. L'addetto alla biblioteca ricerca l'ID della prenotazione</li> <li>4. <b>if</b> l'ID è valido <ol style="list-style-type: none"> <li>4.1. Il sistema recupera e visualizza i dettagli della prenotazione corrispondente all'ID fornito, con la copia del libro prenotata e il suo stato attuale.</li> <li>4.2. <b>if</b> la copia risulta prenotata <ol style="list-style-type: none"> <li>4.2.1. L'addetto alla biblioteca conferma il ritiro della copia e aggiorna lo stato della copia da "prenotata" a "in prestito"</li> <li>4.2.2. Il sistema notifica la conferma del cambiamento dello stato della copia</li> </ol> </li> </ol> </li> </ol>
<b>Post-Condizioni</b>	L'utente registrato ha ritirato la copia del libro di interesse e lo stato di tale copia nel sistema è ora in prestito.
<b>Casi d'uso correlati</b>	-
<b>Sequenza di eventi alternativi</b>	<p>Al punto 4, se l'id non è valido:</p> <ol style="list-style-type: none"> <li>4.1. Il sistema restituisce un messaggio di errore indicando che l'ID della prenotazione non è valido o non è stato trovato.</li> <li>4.2. L'addetto alla biblioteca informa l'utente.</li> </ol> <p>Al punto 4.2, se la copia del libro non è "prenotata":</p> <ol style="list-style-type: none"> <li>4.2.1. il sistema restituisce un messaggio di errore all'addetto, indicando che la copia non può essere ritirata.</li> <li>4.2.2. L'addetto alla biblioteca avvisa l'utente.</li> </ol>



<b>Caso d'uso:</b>	<b>RegistraRestituzioneCopia</b>
<b>Attore primario</b>	AddettoBiblioteca
<b>Attore secondario</b>	-
<b>Descrizione</b>	Un addetto alla biblioteca registra l'avvenuta restituzione di una copia di un libro presso la biblioteca, rendendola di nuovo disponibile
<b>Pre-Condizioni</b>	L'utente ha una copia di un libro in prestito e vuole restituirla. L'addetto alla biblioteca è autenticato nel sistema.
<b>Sequenza di eventi principale</b>	<ol style="list-style-type: none"> <li>1. L'utente fornisce all'addetto della biblioteca l'ID della prenotazione.</li> <li>2. L'addetto alla biblioteca accede all'interfaccia dedicata alla visualizzazione dei libri in prestito</li> <li>3. L'addetto alla biblioteca ricerca l'ID della prenotazione</li> <li>4. <b>if</b> l'ID è valido <ol style="list-style-type: none"> <li>4.1. Il sistema recupera e mostra i dettagli della prenotazione corrispondente all'ID fornito, con la copia del libro prenotata e il suo stato attuale.</li> <li>4.2. <b>if</b> la copia risulta "in prestito" <ol style="list-style-type: none"> <li>4.2.1. L'addetto alla biblioteca conferma la restituzione della copia e aggiorna lo stato della copia del libro da "in prestito" a "disponibile"</li> <li>4.2.2. Il sistema notifica la conferma del cambiamento dello stato del libro</li> </ol> </li> </ol> </li> </ol>
<b>Post-Condizioni</b>	L'utente registrato ha restituito la copia del libro e lo stato di tale copia nel sistema è ora disponibile a nuove prenotazioni
<b>Casi d'uso correlati</b>	-
<b>Sequenza di eventi alternativi</b>	<p>Al punto 4, se l'ID non è valido:</p> <ol style="list-style-type: none"> <li>4.1. Il sistema restituisce un messaggio di errore indicando che l'ID della prenotazione non è valido o non è stato trovato.</li> <li>4.2. L'addetto alla biblioteca informa l'utente.</li> </ol> <p>Al punto 4.2, se la copia del libro non è "in prestito":</p> <ol style="list-style-type: none"> <li>4.2.1. Il sistema restituisce un messaggio di errore all'addetto, indicando che la copia non era stata ritirata in maniera corretta.</li> <li>4.2.2. L'addetto alla biblioteca informa l'utente.</li> </ol>

<b>Caso d'uso:</b>	<b>AggiungiLibro</b>
<b>Attore primario</b>	Amministratore
<b>Attore secondario</b>	-
<b>Descrizione</b>	L'amministratore aggiunge un nuovo libro al catalogo tramite apposita interfaccia
<b>Pre-Condizioni</b>	L'amministratore è autenticato all'interno del sistema. Un libro con gli stessi dati non è già registrato nel catalogo.
<b>Sequenza di eventi principale</b>	<ol style="list-style-type: none"> <li>1. Il caso d'uso inizia quando un amministratore richiede l'aggiunta di un nuovo libro al catalogo</li> <li>2. L'amministratore inserisce i dati del libro (titolo, autore, anno di pubblicazione, genere, codice ISBN, descrizione)</li> <li>3. Il sistema verifica che il formato dei dati del nuovo libro sia valido</li> <li>4. <b>if</b> il formato dei dati è valido <ol style="list-style-type: none"> <li>4.1. Il sistema aggiunge il nuovo libro al catalogo</li> <li>4.2. Il sistema comunica all'amministratore che il libro è stato aggiunto tramite messaggio di conferma</li> </ol> </li> </ol>
<b>Post-Condizioni</b>	Il nuovo libro è aggiunto correttamente nel catalogo digitale
<b>Casi d'uso correlati</b>	-
<b>Sequenza di eventi alternativi</b>	Al punto 4, se i dati del nuovo libro non sono validi, il sistema restituisce un messaggio d'errore e viene chiesto all'amministratore di riprovare, ritornando al punto 2

<b>Caso d'uso:</b>	<b>AggiungiCopia</b>
<b>Attore primario</b>	Amministratore
<b>Attore secondario</b>	-
<b>Descrizione</b>	L'amministratore registra una nuova copia fisica di un libro presente nel catalogo del sistema tramite apposita interfaccia. La copia aggiunta sarà così disponibile per la prenotazione e consultazione
<b>Pre-Condizioni</b>	L'amministratore è autenticato all'interno del sistema.
<b>Sequenza di eventi principale</b>	<ol style="list-style-type: none"> <li>1. Il caso d'uso inizia quando un amministratore richiede l'aggiunta di una copia di un libro</li> <li>2. L'amministratore inserisce il codice ISBN del libro corrispondente alla copia da aggiungere</li> <li>3. <b>if</b> il codice è associato ad un libro presente nel catalogo <ol style="list-style-type: none"> <li>3.1. Il sistema genera un nuovo identificativo univoco per la copia del libro</li> <li>3.2. Il sistema aggiunge la copia, rendendola disponibile alla prenotazione e incrementando il numero di copie disponibili del libro</li> <li>3.3. Il sistema comunica all'amministratore che la copia è stata aggiunta tramite un messaggio di conferma</li> </ol> </li> </ol>
<b>Post-Condizioni</b>	La nuova copia è stata aggiunta correttamente al catalogo ed è disponibile alla prenotazione
<b>Casi d'uso correlati</b>	-
<b>Sequenza di eventi alternativi</b>	Al punto 3, se i dati inseriti in ricerca non producono risultati il sistema restituisce un messaggio d'errore e viene chiesto all'amministratore di riprovare, ritornando al punto 2

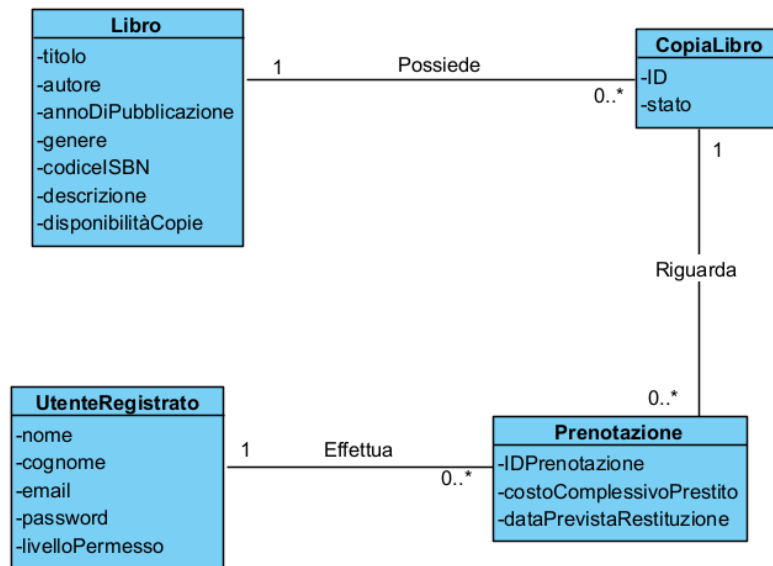
<b>Caso d'uso:</b>	<b>ModificaLibro</b>
<b>Attore primario</b>	Amministratore
<b>Attore secondario</b>	-
<b>Descrizione</b>	L'amministratore effettua modifiche, tramite apposita interfaccia, su un libro esistente nel catalogo, i dati del libro a seguito delle modifiche sono aggiornati
<b>Pre-Condizioni</b>	L'amministratore è autenticato all'interno del sistema.
<b>Sequenza di eventi principale</b>	<ol style="list-style-type: none"> <li>1. Il caso d'uso inizia quando un amministratore richiede la modifica dei dati di un libro nel catalogo</li> <li>2. L'amministratore inserisce il codice ISBN del libro presente nel catalogo</li> <li>3. <b>if</b> il codice è associato ad un libro presente nel catalogo <ol style="list-style-type: none"> <li>3.1. Il sistema richiede i nuovi dati del libro da modificare</li> <li>3.2. L'amministratore inserisce i nuovi dati</li> <li>3.3. <b>If</b> i nuovi dati inseriti sono validi <ol style="list-style-type: none"> <li>3.3.1. Il sistema comunica all'amministratore della riuscita modifica dei dati del libro</li> </ol> </li> </ol> </li> </ol>
<b>Post-Condizioni</b>	I dati del libro sono correttamente modificati
<b>Casi d'uso correlati</b>	-
<b>Sequenza di eventi alternativi</b>	<p>Al punto 3, se i dati inseriti in ricerca non forniscono un libro già presente nel catalogo il sistema restituisce un messaggio d'errore e viene chiesto all'amministratore di riprovare, ritornando al punto 2</p> <p>Al punto 3.3, se i nuovi dati inseriti non sono validi, il sistema mostra un messaggio d'errore e viene chiesto all'amministratore di riprovare, ritornando al punto 3.1</p>

<b>Caso d'uso:</b>	<b>EliminaLibro</b>
<b>Attore primario</b>	Amministratore
<b>Attore secondario</b>	-
<b>Descrizione</b>	L'amministratore elimina un libro presente nel catalogo tramite apposita interfaccia
<b>Pre-Condizioni</b>	L'amministratore è autenticato all'interno del sistema.
<b>Sequenza di eventi principale</b>	<ol style="list-style-type: none"> <li>1. Il caso d'uso inizia quando un amministratore richiede l'eliminazione di un libro dal catalogo</li> <li>2. L'amministratore inserisce il codice ISBN del libro che vuole eliminare</li> <li>3. <b>if</b> il codice è associato ad un libro presente nel catalogo <ol style="list-style-type: none"> <li>3.1. <b>if</b> il libro non ha nessuna copia in prestito o prenotata <ol style="list-style-type: none"> <li>3.1.1. Il sistema elimina il libro dal catalogo</li> <li>3.1.2. Il sistema comunica all'amministratore la riuscita eliminazione del libro dal catalogo</li> </ol> </li> </ol> </li> </ol>
<b>Post-Condizioni</b>	Il libro, e le corrispettive copie, sono correttamente eliminate dal catalogo
<b>Casi d'uso correlati</b>	-
<b>Sequenza di eventi alternativi</b>	<p>Al punto 3, se i dati inseriti in ricerca non forniscono un libro già presente nel catalogo il sistema restituisce un messaggio d'errore e viene chiesto all'amministratore di riprovare, ritornando al punto 2</p> <p>Al punto 3.1, se il libro specificato possiede una copia prenotata per un prestito futuro o è attualmente in prestito, il sistema notifica l'amministratore con l'impossibilità di eliminare il libro.</p>

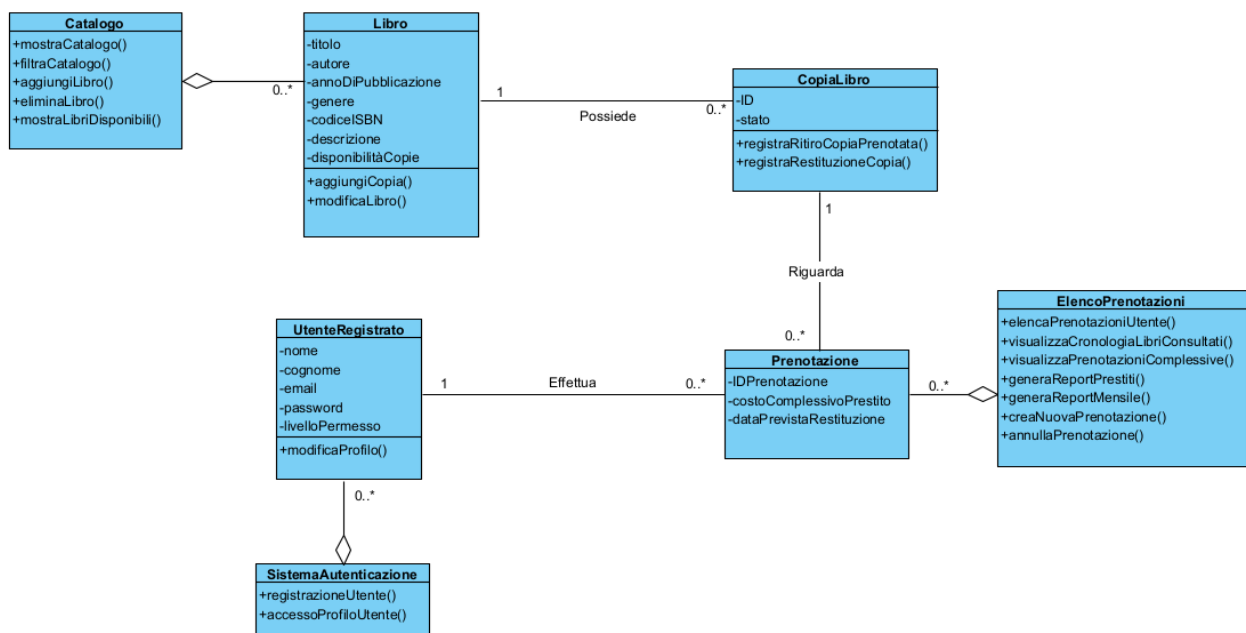
<b>Caso d'uso:</b>	<b>PrenotaLibroDisponibile</b>
<b>Attore primario</b>	UtenteRegistrato
<b>Attore secondario</b>	-
<b>Descrizione</b>	Un utente registrato, dalla propria area personale, effettua la prenotazione di un libro del catalogo. All'utente è mostrata la ricevuta di avvenuta prenotazione
<b>Pre-Condizioni</b>	L'utente ha effettuato l'accesso alla propria area personale.
<b>Sequenza di eventi principale</b>	<ol style="list-style-type: none"> <li>1. Il caso d'uso inizia quando l'utente richiede di effettuare una prenotazione</li> <li>2. &lt;&lt;include&gt;&gt; <i>MostraLibriDisponibili</i></li> <li>3. Il sistema mostra all'utente il catalogo di libri disponibili</li> <li>4. L'utente seleziona il libro di interesse</li> <li>5. L'utente inserisce una data di prevista restituzione</li> <li>6. <b>if</b> la data di prevista restituzione inserita è valida <ol style="list-style-type: none"> <li>6.1. Il sistema riserva una copia del corrispettivo libro all'utente cambiando lo stato della copia prenotata in "prenotata"</li> <li>6.2. Il sistema genera la ricevuta della prenotazione</li> <li>6.3. Il sistema mostra la ricevuta all'utente con un messaggio di avvenuta prenotazione</li> </ol> </li> </ol>
<b>Post-Condizioni</b>	L'utente è registrato al sistema È stato creato un nuovo account associato all'indirizzo e-mail fornito
<b>Casi d'uso correlati</b>	<i>MostraLibriDisponibili</i>
<b>Sequenza di eventi alternativi</b>	Al punto 6, se la data di prevista restituzione non è valida, il sistema mostra un messaggio di errore e richiede all'utente di inserirla, riportandolo al punto 5.

## 2.6 Diagramma delle classi

Diagramma delle classi di analisi corrispondenti al System Domain Model: rappresentazione statica del sistema focalizzata esclusivamente sulla struttura concettuale dei dati, indipendente da logiche di processo.



Versione raffinata del diagramma: sono state integrate le responsabilità alle classi e identificate alcune classi *Information Expert*, con l'obiettivo di aumentare la modularità e semplificare l'architettura complessiva del sistema.



## 2.6.1 Responsabilità

Segue una tabella riepilogativa delle principali responsabilità assegnate alle classi, a supporto di una progettazione orientata alla coesione e alla separazione dei compiti.

RESPONSABILITÀ	CLASSE
MostraCatalogo	<b>Catalogo</b>
FiltraCatalogo	<b>Catalogo</b>
RegistrazioneUtente	<b>SistemaAutenticazione</b>
AccessoProfiloUtente	<b>SistemaAutenticazione</b>
ModificaProfilo	<b>UtenteRegistrato</b>
ElencaPrenotazioniUtente	<b>ElencoPrenotazioni</b>
VisualizzaCronologiaLibriConsultati	<b>ElencoPrenotazioni</b>
creaNuovaPrenotazione	<b>ElencoPrenotazioni</b>
AnnullaPrenotazione	<b>ElencoPrenotazioni</b>
VisualizzaPrenotazioniComplessive	<b>ElencoPrenotazioni</b>
RegistraRitiroCopiaPrenotata	<b>CopiaLibro</b>
RegistraRestituzioneCopia	<b>CopiaLibro</b>
GeneraReportPrestiti	<b>ElencoPrenotazioni</b>
AggiungiLibro	<b>Catalogo</b>
AggiungiCopia	<b>Libro</b>
ModificaLibro	<b>Libro</b>
EliminaLibro	<b>Catalogo</b>
GeneraReportMensile	<b>ElencoPrenotazioni</b>
MostraLibriDisponibili	<b>Catalogo</b>

**mostraCatalogo** è responsabilità di *Catalogo*, in quanto è <<Information Expert>> dei libri che contiene. Ha accesso diretto alla lista dei libri e può quindi visualizzarli efficientemente. Questa assegnazione favorisce anche <<Low Coupling>>, evitando dipendenze superflue da altre classi.

**filtraCatalogo** è responsabilità di *Catalogo*, perché agisce sui propri dati interni. È una variazione di *mostraCatalogo* che richiede un filtraggio dei contenuti, dunque coerente con il pattern <<Information Expert>>.

**registrazioneUtente** e **accessoProfiloUtente** sono responsabilità di *SistemaAutenticazione*, in quanto è <<Information Expert>> sui meccanismi di autenticazione e validazione degli *UtentiRegistrati*.

**modificaProfilo** è responsabilità di *UtenteRegistrato*, perché è <<Information Expert>> dei propri attributi (nome, cognome, email, ecc.). Inoltre, questa scelta aderisce al principio di <<Low Coupling>>, evitando dipendenze esterne.

**elencaPrenotazioniUtente** è responsabilità di *ElencoPrenotazioni*, in quanto è <<Information Expert>> delle prenotazioni effettuate dagli utenti.



**visualizzaCronologiaLibriConsultati** è responsabilità di *ElencoPrenotazioni*, perché ha accesso a tutte le prenotazioni, incluse quelle concluse, e quindi è <<Information Expert>> della cronologia.

**creaNuovaPrenotazione** è responsabilità di *ElencoPrenotazioni*, poiché è <<Creator>> dell'oggetto *Prenotazione* (nonché information expert).

**annullaPrenotazione** è responsabilità di *ElencoPrenotazioni*, perché, seguendo il pattern <<Creator>>, è coinvolto nel ciclo di vita della prenotazione e quindi anche nella sua cancellazione.

**visualizzaPrenotazioniComplessive** è responsabilità di *ElencoPrenotazioni*, poiché è <<Information Expert>> delle prenotazioni esistenti, sia correnti che passate.

**registraRitiroCopiaPrenotata** è responsabilità di *CopiaLibro*, perché è <<Information Expert>> del proprio stato. La modifica dello stato di ritiro deve avvenire nella classe che lo conosce e lo gestisce direttamente.

**registraRestituzioneCopia** è responsabilità di *CopiaLibro*, sempre per il principio di <<Information Expert>>. La copia conosce il proprio stato e può aggiornarlo in seguito alla restituzione.

**generaReportPrestiti** è responsabilità di *ElencoPrenotazioni*, in quanto <<Information Expert>> di tutte le prenotazioni, da cui può estrarre i dati necessari ai report.

**aggiungiLibro** è responsabilità di *Catalogo*, poiché segue il pattern <<Creator>>: il catalogo gestisce l'esistenza e l'inserimento di nuovi libri.

**aggiungiCopia** è responsabilità di *Libro*, poiché è <<Information Expert>> del numero e dello stato delle copie disponibili.

**modificaLibro** è responsabilità di *Libro*, per il principio di <<Information Expert>>: i dati del libro sono noti e modificabili solo da esso stesso. Inoltre, questa scelta supporta <<Low Coupling>>.

**eliminaLibro** è responsabilità di *Catalogo*, poiché segue il pattern <<Creator>>: la classe che crea un oggetto è generalmente anche quella che lo elimina, mantenendo coerenza nella gestione del ciclo di vita.

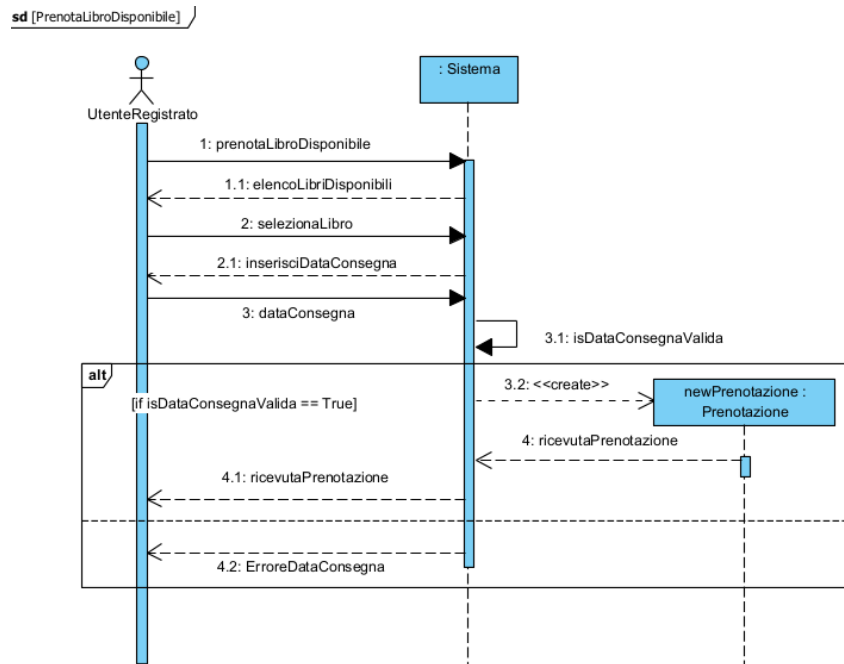
**generaReportMensile** è responsabilità di *ElencoPrenotazioni*, perché è <<Information Expert>> delle prenotazioni e può aggregarle secondo criteri temporali e statistici.

**MostraLibriDisponibili** è responsabilità di *Catalogo*, perché conosce i libri associate ad esso e può ottenere lo stato di ciascuna copia per verificarne la disponibilità

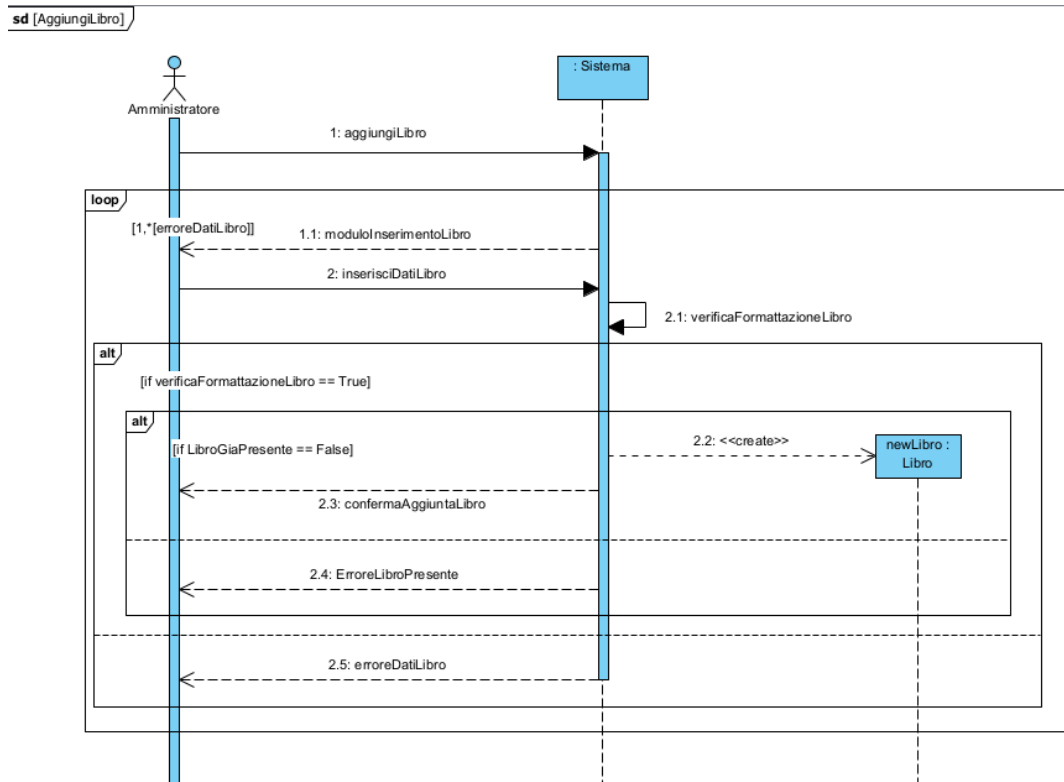
## 2.7 Diagrammi di sequenza

Si riportano alcuni esempi di diagrammi di sequenza di analisi per alcuni casi d'uso precedentemente individuati.

### 2.7.1 PrenotaLibroDisponibile

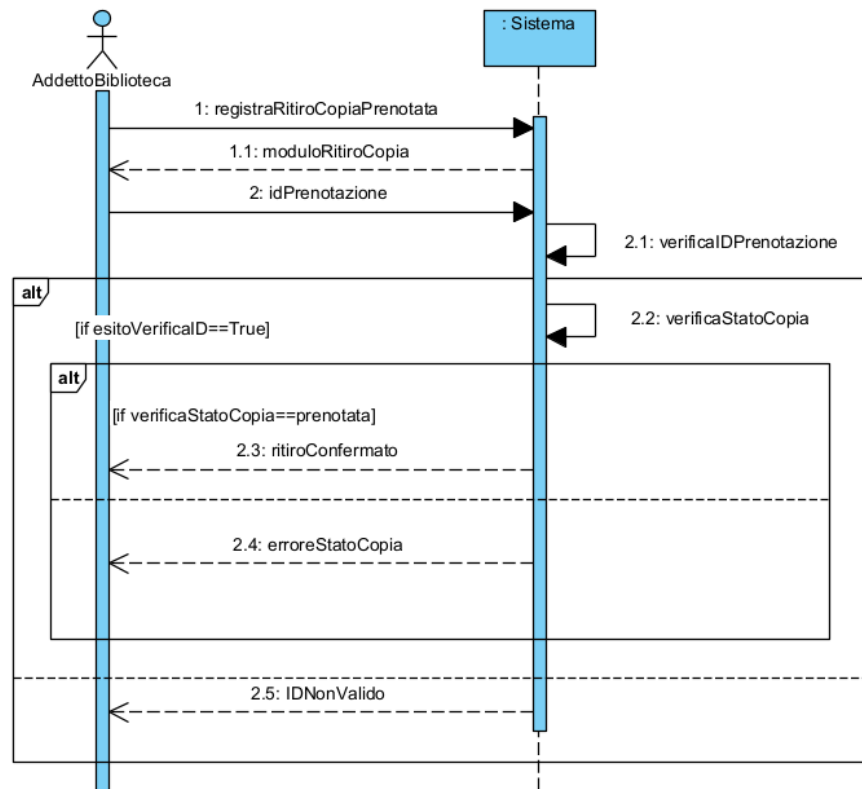


### 2.7.2 AggiungiLibro



## 2.7.3 RegistraRitiroCopiaPrenotata

sd [RegistraRitiroCopiaPrenotata]

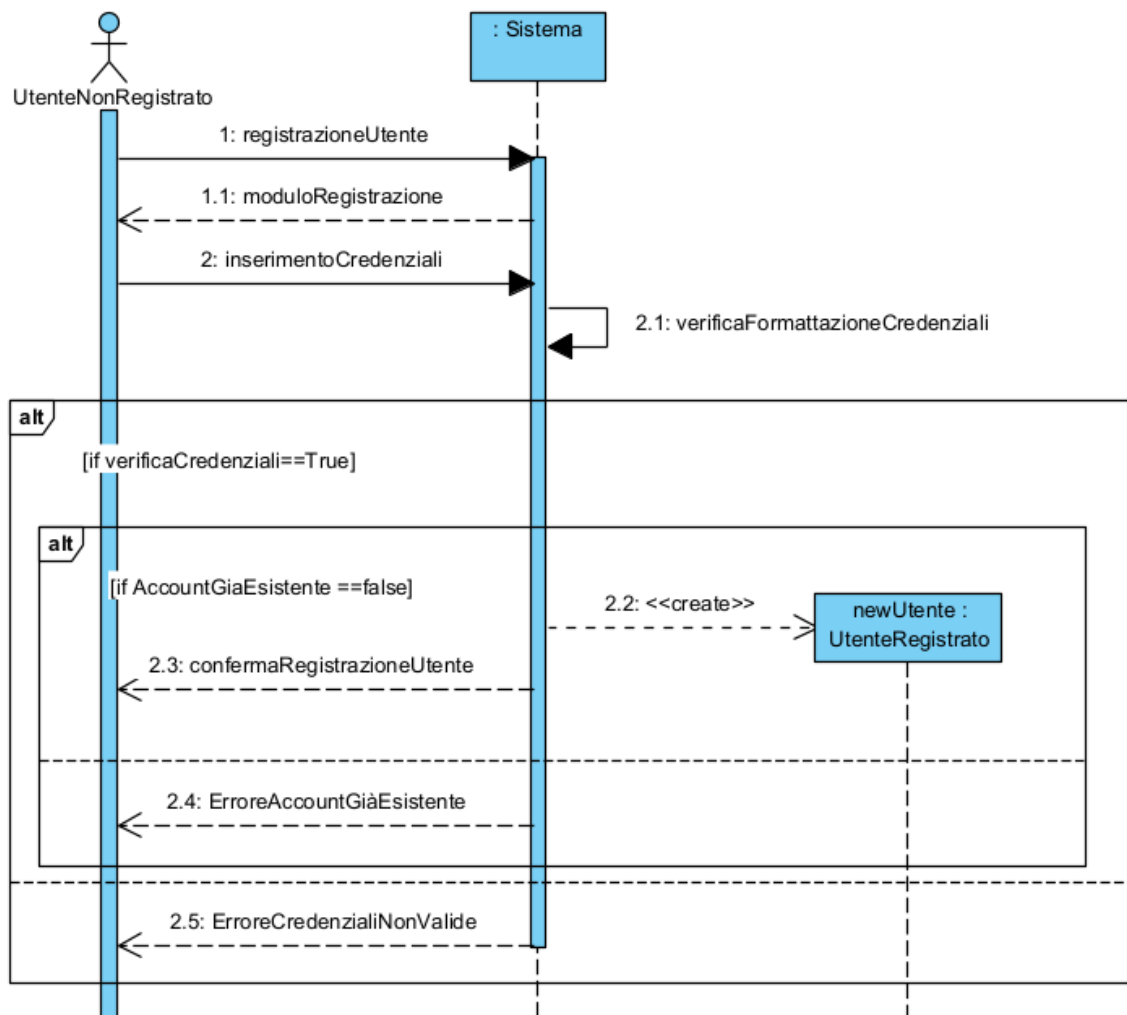


La creazione del sequence diagram, sviluppato a partire dalla descrizione dello scenario del caso d'uso *RegistraRitiroCopiaPrenotata*, ha fatto sorgere la necessità di definire un metodo specifico per la classe **ElencoPrenotazioni** chiamato **verificaIDPrenotazione** al fine di verificare l'esistenza della prenotazione.

Per la classe **CopiaLibro** invece è stato aggiunto un metodo specifico **verificaStatoCopia**, per verificare che lo stato sia corretto al momento del ritiro.

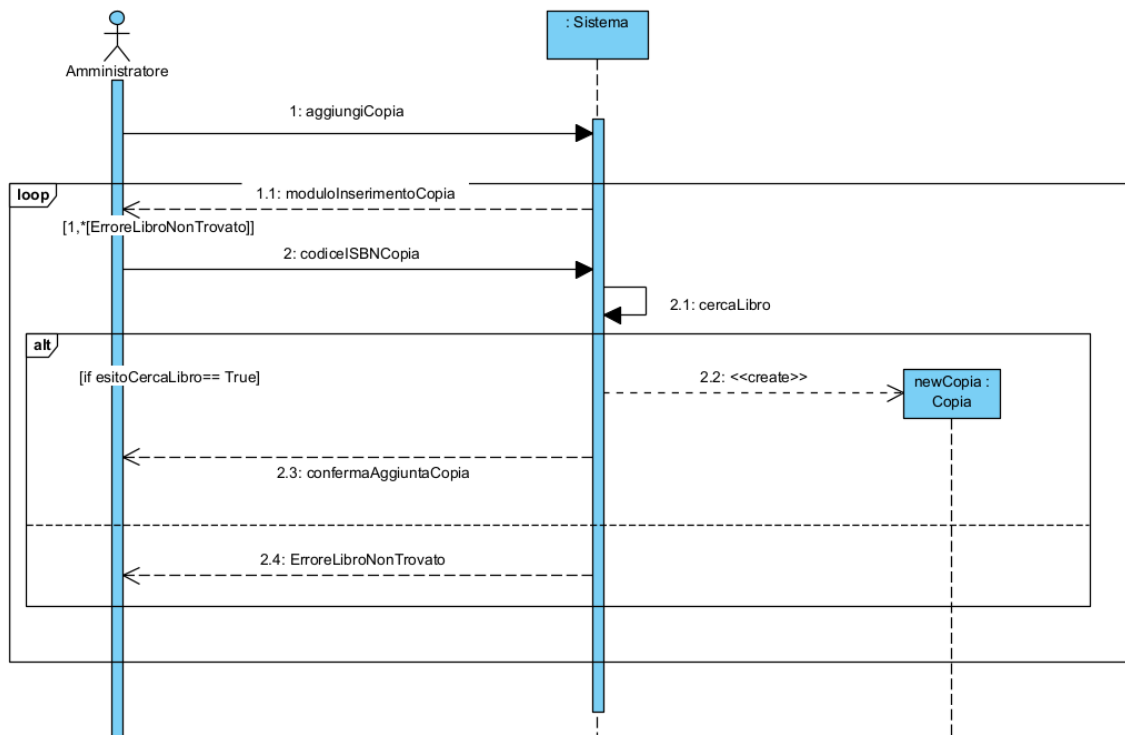
## 2.7.4 RegistrazioneUtente

**sd** [RegistrazioneUtente]



## 2.7.5 AggiungiCopia

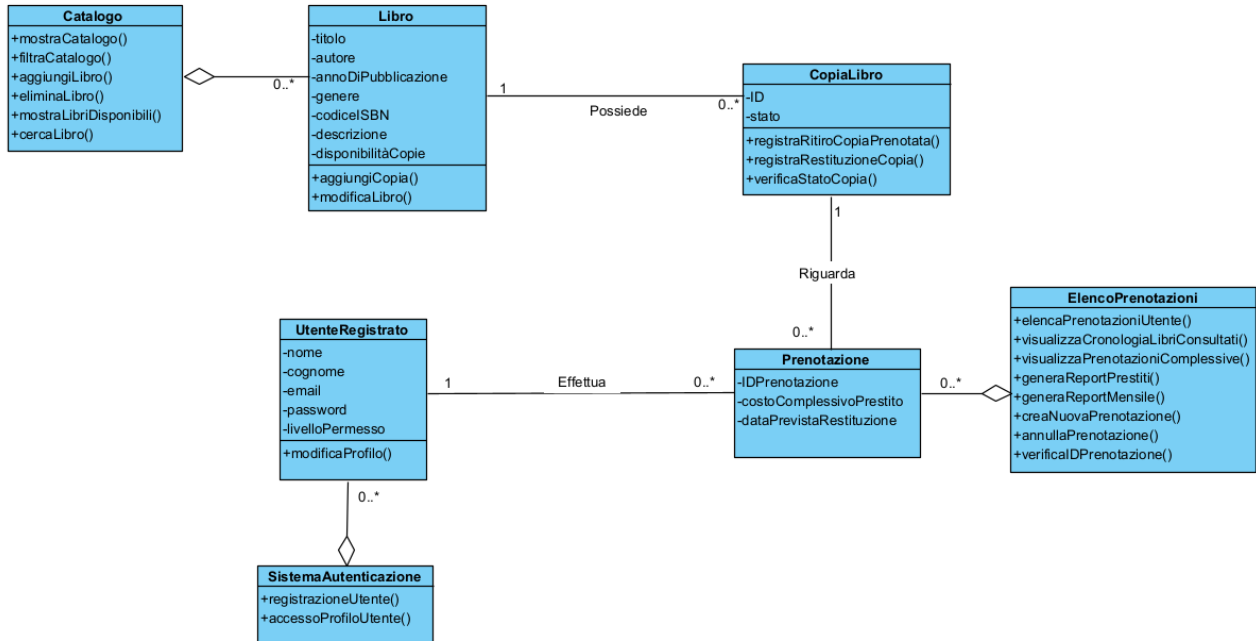
sd [AggiungiCopia]



La creazione del sequence diagram, sviluppato a partire dalla descrizione dello scenario del caso d'uso *AggiungiCopia*, ha fatto sorgere la necessità di definire un metodo, specifico per la classe **Catalogo**, **cercaLibro**, privato, per consentire al catalogo di verificare che un libro, dato il codice ISBN, sia presente nel catalogo.

## 2.8 Diagramma delle classi raffinato

Le aggiunte e le modifiche fatte nel corso della costruzione dei Sequence Diagrams hanno determinato lo sviluppo di un **Diagramma delle Classi** raffinato che riporta maggiori dettagli sugli attributi e le principali operazioni delle classi:





### 3. Piano di test funzionale

Si intende progettare i casi di test funzionale con la tecnica del Category Partition Testing.

### 3.1 Registrazione Utente

Registrazione Utente			
NOME	COGNOME	EMAIL	PASSWORD
<ul style="list-style-type: none"> <li>Stringa di caratteri di lunghezza &lt;= 50</li> <li>Dati inseriti validi</li> <li>Prima lettera non maiuscola <b>[ERROR]</b></li> <li>Stringa di caratteri di lunghezza &gt; 50 <b>[ERROR]</b></li> <li>Stringa che contiene simboli che non sono caratteri <b>[ERROR]</b></li> <li>Stringa di caratteri di lunghezza &lt; 1 <b>[ERROR]</b></li> </ul>	<ul style="list-style-type: none"> <li>Stringa di caratteri di lunghezza &lt;= 50</li> <li>Dati inseriti validi</li> <li>Stringa di caratteri di lunghezza &gt; 50 <b>[ERROR]</b></li> <li>Stringa che contiene simboli che non sono caratteri o apostrofo <b>[ERROR]</b></li> <li>Stringa di caratteri di lunghezza &lt; 1 <b>[ERROR]</b></li> </ul>	<ul style="list-style-type: none"> <li>Stringa di caratteri alfanumerici contenente carattere speciale '@'</li> <li>Stringa di caratteri alfanumerici non contenente carattere speciale '@' <b>[ERROR]</b></li> <li>Dati inseriti validi</li> <li>Email associata ad account già esistente <b>[ERROR]</b></li> <li>Stringa di caratteri di lunghezza &gt; 255 <b>[ERROR]</b></li> </ul>	<ul style="list-style-type: none"> <li>Stringa di caratteri di lunghezza &gt;= 8</li> <li>Dati inseriti validi</li> <li>Stringa di caratteri di lunghezza &lt;8 <b>[ERROR]</b></li> <li>Stringa non contenente maiuscole <b>[ERROR]</b></li> <li>Stringa non contenente minuscole <b>[ERROR]</b></li> <li>Stringa non contenente caratteri speciali <b>[ERROR]</b></li> <li>Stringa non contenente numeri <b>[ERROR]</b></li> </ul>

Il numero di test da effettuarsi senza particolari vincoli è:  $6 * 5 * 5 * 7 = 1050$ .

Con i vincoli [ERROR], invece, il numero di test da eseguire per testare singolarmente i vincoli è 15 (4 per Nome, 3 per Cognome, 3 per email, 5 per password).

Il numero di test risultante è:  $(1*1*1*1) + 15 = 16$ .

TEST SUITE						
Test Case ID	Descrizione	Classi di equivalenza coperte	Pre-condizioni	Input	Output attesi	Post-condizioni attese
1	Tutti input validi	Nome valido Cognome valido Email valida Password valida	-	{Nome: "Antonio", Cognome: "Di Giorgio", Email: "antonio.email@gmail.com", Password: "Password123!"}	L'utente può accedere alla propria aera personale	L'utente è registrato
2	Stringa Nome > 50 caratteri	Nome > 50 caratteri [ERROR] Cognome valido Email valida Password valida	-	{Nome: "Jacopoooooooooooooooooooo", Cognome: "Ferrante", Email: "jacopo.email@gmail.com", Password: "Pass987@l!"}	Il nome inserito non è valido. Per favore, controlla e riprova.	L'utente è reindirizzato o alla schermata di registrazione
3	Stringa Nome contenente simboli che non sono caratteri	Nome con simboli [ERROR], Cognome valido Email valida Password valida	-	{Nome: "Anto_d", Cognome: "Di Giorgio", Email: "antonio.email@gmail.com", Password: "Password123!"}	Il nome inserito non è valido. Per favore, controlla e riprova.	L'utente è reindirizzato o alla schermata di registrazione
4	Stringa Nome < 1 carattere	Nome < 1 carattere [ERROR].	-	{Nome: "", Cognome: "D'Amico", Email:	Il nome inserito non è valido.	L'utente è reindirizzato



13	Stringa Password non contenente minuscole	Nome valido Cognome valido Email valida Password [ERROR]	-	{Nome: "Matteo", Cognome: "Bottari", Email: "ma.bo@studenti.unina.it", Password: "CIAO123_"}	La password non soddisfa i requisiti di sicurezza. Per favore, riprova.	L'utente è reindirizzato alla schermata di registrazione
14	Stringa Password non contenente caratteri speciali	Nome valido Cognome valido Email valida Password [ERROR]	-	{Nome: "Matteo", Cognome: "Bottari", Email: "ma.bo@studenti.unina.it", Password: "Ciao1234"}	La password non soddisfa i requisiti di sicurezza. Per favore, riprova.	L'utente è reindirizzato alla schermata di registrazione
15	Stringa Password non contenente numeri	Nome valido Cognome valido Email valida Password [ERROR]	-	{Nome: "Matteo", Cognome: "Bottari", Email: "ma.bo@studenti.unina.it", Password: "CiaoCiao_"}	La password non soddisfa i requisiti di sicurezza. Per favore, riprova.	L'utente è reindirizzato alla schermata di registrazione
16	Stringa email > 255	Nome valido Cognome valido Email [ERROR] Password valida	-	{Nome: "Matteo", Cognome: "Bottari", Email: "ma.boooooooooo(o*255)@studenti.unina.it", Password: "Matteo04_"}	L'email inserita ha un formato non corretto. Per favore, riprova.	L'utente è reindirizzato alla schermata di registrazione

## 3.2 Aggiunta Libri

Aggiunta Libri					
TITOLO	AUTORE	ANNO DI PUBBLICAZIONE	GENERE	CODICE ISBN	DESCRIZIONE
<ul style="list-style-type: none"> <li>Dati inseriti validi</li> <li>Stringa che contiene caratteri speciali [ERROR]</li> <li>Stringa di caratteri di lunghezza &lt;= 50</li> <li>Stringa di caratteri di lunghezza &gt; 50 [ERROR]</li> <li>Stringa di caratteri di lunghezza &lt; 1 [ERROR]</li> </ul>	<ul style="list-style-type: none"> <li>Stringa di caratteri di lunghezza &lt;= 50</li> <li>Dati inseriti validi</li> <li>Stringa di caratteri di lunghezza &gt; 50 [ERROR]</li> <li>Stringa che contiene simboli che non sono caratteri o apostrofo [ERROR]</li> <li>Stringa di caratteri di lunghezza &lt; 1 [ERROR]</li> </ul>	<ul style="list-style-type: none"> <li>Stringa numerica di valore &gt;= 0</li> <li>Dati inseriti validi</li> <li>Stringa numerica di valore &lt; 0 [ERROR]</li> <li>Stringa non contenente solo caratteri numerici [ERROR]</li> </ul>	<ul style="list-style-type: none"> <li>Dati inseriti validi</li> <li>Stringa che contiene caratteri speciali [ERROR]</li> <li>Stringa di caratteri di lunghezza &lt;= 50</li> <li>Stringa di caratteri di lunghezza &gt; 50 [ERROR]</li> </ul>	<ul style="list-style-type: none"> <li>Dati inseriti validi</li> <li>Stringa non contenente solo caratteri numerici [ERROR]</li> <li>Stringa numerica di lunghezza = 13</li> <li>Stringa numerica di lunghezza ≠ 13 [ERROR]</li> <li>Codice ISBN già presente nella biblioteca [ERROR]</li> </ul>	<ul style="list-style-type: none"> <li>Stringa di caratteri di lunghezza &lt;= 500</li> <li>Dati inseriti validi</li> <li>Stringa di caratteri di lunghezza &gt; 500 [ERROR]</li> </ul>

Il numero di test da effettuarsi senza particolari vincoli è:  $5 * 5 * 4 * 4 * 5 * 3 = 6000$ .

Con i vincoli [ERROR], invece, il numero di test da eseguire per testare singolarmente i vincoli è 14 (3 per Titolo, 3 per Autore, 2 per Anno di pubblicazione, 2 per Genere, 3 per codice ISBN, 1 per Descrizione).

Il numero di test risultante è:  $(1*1*1*1*1*1) + 14 = 15$ .

TEST SUITE						
Test Case ID	Descrizione	Classi di equivalenza coperte	Pre-condizioni	Input	Output attesi	Post-condizioni attese
1	Tutti input validi	Titolo valido Autore valido Anno di Pubblicazione valido Genere valido ISBN valido Descrizione valida	-	{Titolo: "Il Signore delle Mosche", Autore: "William Golding", Anno di Pubblicazione: "1954", Genere: "Romanzo", ISBN: "9780143124290", Descrizione: "Romanzo di William Golding che narra la storia di un gruppo di ragazzi."}	Il libro ora è presente e visualizzabile nel catalogo della biblioteca	Libro aggiunto al catalogo
2	Stringa Titolo contenente caratteri speciali	Titolo con simboli [ERROR] Autore valido Anno di Pubblicazione valido Genere valido ISBN valido Descrizione valida	-	{Titolo: "Il Signore delle Mosche!", Autore: "William Golding", Anno di Pubblicazione: "1954", Genere: "Romanzo", ISBN: "9780143124290", Descrizione: "Romanzo di William Golding che narra la storia di un gruppo di ragazzi."}	Il titolo contiene caratteri speciali non ammessi.	L'amministratore è reindirizzato all'interfaccia per aggiungere Libri.
3	Stringa Titolo di lunghezza > 50 caratteri	Titolo > 50 [ERROR] Autore valido Anno di Pubblicazione valido Genere valido ISBN valido Descrizione valida	-	{Titolo: "Il Signore delle Moscheeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee", Autore: "William Golding", Anno di Pubblicazione: "1954", Genere: "Romanzo", ISBN: "9780143124290", Descrizione: "Romanzo di William Golding che narra la storia di un gruppo di ragazzi."}	Il titolo non può superare i 50 caratteri.	L'amministratore è reindirizzato all'interfaccia per aggiungere Libri.
4	Stringa titolo di lunghezza < 1	Titolo [ERROR] Autore Valido Anno di pubblicazione Valido Genere Valido ISBN Valido Descrizione Valido	-	{Titolo: "", Autore: "William Golding", Anno di Pubblicazione: "1954", Genere: "Romanzo", ISBN: "9780143124290", Descrizione: "Romanzo di William Golding che narra la storia di un gruppo di ragazzi."}	Il campo titolo non può essere vuoto.	L'amministratore è reindirizzato all'interfaccia per aggiungere Libri.
5	Stringa Autore di lunghezza > 50 caratteri	Titolo valido Autore > 50 [ERROR] Anno di Pubblicazione valido Genere valido ISBN valido Descrizione valida	-	{Titolo: "Il Signore delle Mosche", Autore: "William Golding", Anno di Pubblicazione: "1954", Genere: "Romanzo", ISBN: "9780143124290", Descrizione: "Romanzo di William Golding che narra la storia di un gruppo di ragazzi."}	Il campo autore non può superare i 50 caratteri.	L'amministratore è reindirizzato all'interfaccia per aggiungere Libri.

6	Stringa Autore contenente simboli che non sono caratteri o apostrofo	Titolo valido Autore con simboli [ERROR] Anno di Pubblicazione valido Genere valido ISBN valido Descrizione valida	-	{Titolo: "Il Signore delle Mosche", Autore: "William Golding03_", Anno di Pubblicazione: "1954", Genere: "Romanzo", ISBN: "9780143124290", Descrizione: "Romanzo di William Golding che narra la storia di un gruppo di ragazzi."}	Il campo autore può contenere solo lettere e apostrofi.	L'amministratore è reindirizzato all'interfaccia per aggiungere Libri.
7	Stringa Autore di lunghezza < 1	Titolo Valido Autore [ERROR] Anno di pubblicazione Valido Genere Valido ISBN Valido Descrizione Valido	-	{Titolo: "Il Signore delle Mosche", Autore: "", Anno di Pubblicazione: "1954", Genere: "Romanzo", ISBN: "9780143124290", Descrizione: "Romanzo di William Golding che narra la storia di un gruppo di ragazzi."}	Il campo autore non può essere vuoto.	L'amministratore è reindirizzato all'interfaccia per aggiungere Libri.
8	Stringa Anno di Pubblicazione di valore< 0	Titolo Valido Autore Valido Anno di pubblicazione [ERROR] Genere Valido ISBN Valido Descrizione Valido	-	{Titolo: "Il Signore delle Mosche", Autore: "William Golding", Anno di Pubblicazione: "-154", Genere: "Romanzo", ISBN: "9780143124290", Descrizione: "Romanzo di William Golding che narra la storia di un gruppo di ragazzi."}	L'anno di pubblicazione deve essere un numero intero positivo.	L'amministratore è reindirizzato all'interfaccia per aggiungere Libri.
9	Stringa Anno di Pubblicazione non contenente solo caratteri numerici	Titolo Valido Autore Valido Anno di pubblicazione [ERROR] Genere Valido ISBN Valido Descrizione Valido	-	{Titolo: "Il Signore delle Mosche", Autore: "William Golding", Anno di Pubblicazione: "1a54", Genere: "Romanzo", ISBN: "9780143124290", Descrizione: "Romanzo di William Golding che narra la storia di un gruppo di ragazzi."}	L'anno di pubblicazione deve essere un numero intero positivo.	L'amministratore è reindirizzato all'interfaccia per aggiungere Libri.
10	Stringa Genere contenente caratteri speciali	Titolo Valido Autore Valido Anno di pubblicazione Valido Genere [ERROR] ISBN Valido Descrizione Valido	-	{Titolo: "Il Signore delle Mosche", Autore: "William Golding", Anno di Pubblicazione: "1954", Genere: "Ro_manz", ISBN: "9780143124290", Descrizione: "Romanzo di William Golding che narra la storia di un gruppo di ragazzi."}	Il campo genere contiene caratteri speciali non ammessi.	L'amministratore è reindirizzato all'interfaccia per aggiungere Libri.
11	Stringa Genere di lunghezza > 50 caratteri	Titolo Valido Autore Valido Anno di pubblicazione Valido Genere [ERROR] ISBN Valido Descrizione Valido	-	{Titolo: "Il Signore delle Mosche", Autore: "William Golding", Anno di Pubblicazione: "1954", Genere: "Romanzo000", ISBN: "9780143124290", Descrizione: "Romanzo di William Golding che narra la storia di un gruppo di ragazzi."}	Il campo genere non può superare i 50 caratteri.	L'amministratore è reindirizzato all'interfaccia per aggiungere Libri.
12	Stringa Codice ISBN non contenente solo caratteri numerici	Titolo Valido Autore Valido Anno di pubblicazione Valido	-	{Titolo: "Il Signore delle Mosche", Autore: "William Golding", Anno di Pubblicazione: "1954", Genere: "Romanzo", ISBN: "abc0143124290", Descrizione: "Romanzo di	Il codice ISBN deve contenere solo cifre.	L'amministratore è reindirizzato all'interfaccia per





### 3.3 Prenota Libro Disponibile

Prenota Libro Disponibile	
SCELTA LIBRO	DATA PREVISTA RESTITUZIONE
<ul style="list-style-type: none"> <li>Scelta Libro valida</li> <li>Libro non selezionato <b>[ERROR]</b></li> </ul>	<ul style="list-style-type: none"> <li>Data prevista restituzione valida</li> <li>Data prevista restituzione (giorno/mese/anno) non valida <b>[ERROR]</b></li> <li>Data prevista restituzione precedente a quella odierna <b>[ERROR]</b></li> <li>Data prevista restituzione non inserita <b>[ERROR]</b></li> </ul>

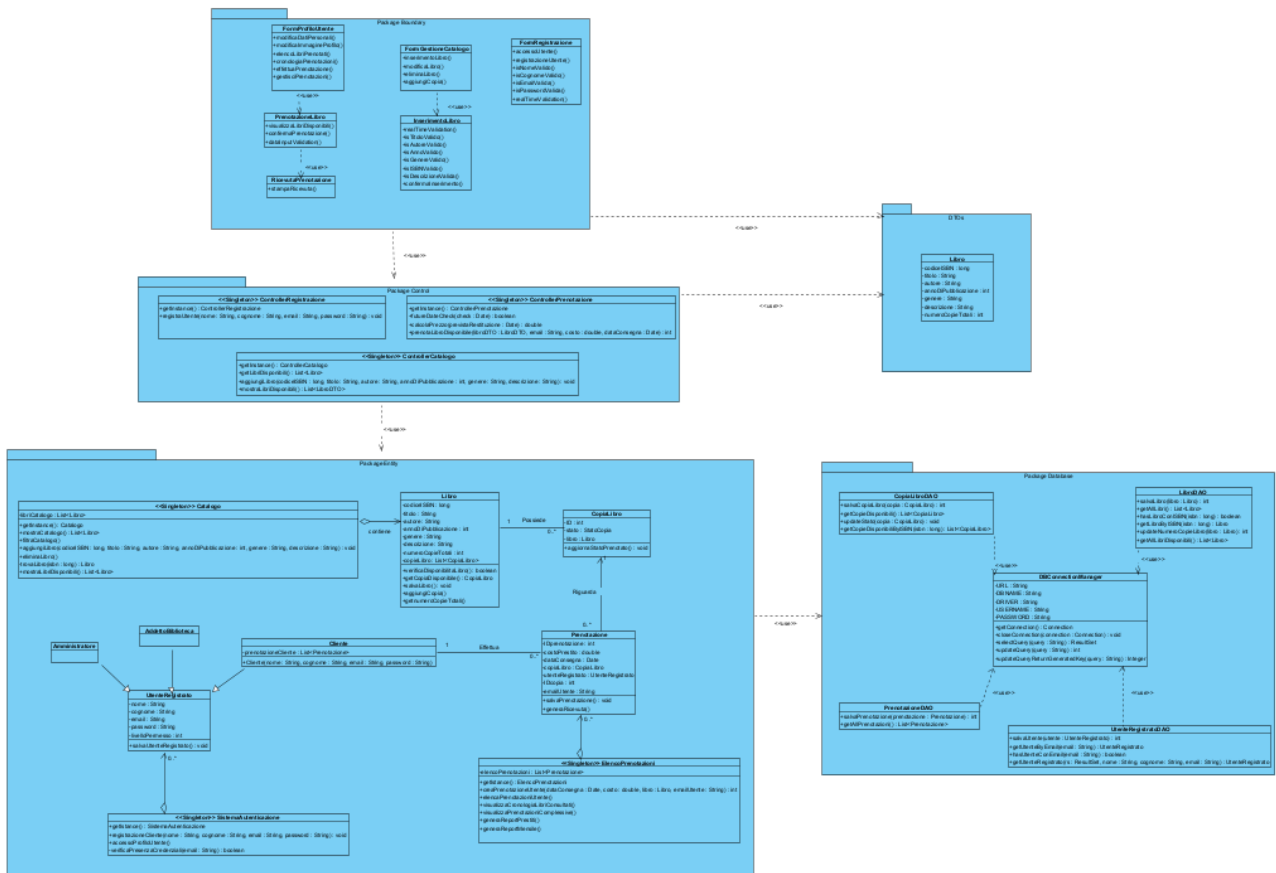
Il numero di test da effettuarsi senza particolari vincoli è:  $2 * 4 = 8$

Con i vincoli **[ERROR]**, invece, il numero di test da eseguire per testare singolarmente i vincoli è 4 (1 per Scelta Libro, 3 per Data).

Il numero di test risultante è:  $(1*1) + 4 = 5$ .

TEST SUITE						
Test Case ID	Descrizione	Classi di equivalenza coperte	Pre-condizioni	Input	Output attesi	Post-condizioni attese
1	Tutti input validi	Scelta Libro Valid Scelta Data Prevista restituzione	L'utente ha effettuato il login	{Scelta Libro: "Ingegneria del Software", Data Prevista restituzione: "10/04/2026"}	La ricevuta è mostrata a video con un invito a conservarla per ritirare il libro	L'utente ha prenotato un libro di interesse ed è stata restituita una ricevuta
2	Scelta Libro non selezionato	Scelta Libro <b>[ERROR]</b> Scelta Data Prevista restituzione	L'utente ha effettuato il login	{Scelta Libro: "", Data Prevista restituzione: "10/04/2026"}	Seleziona un libro e inserisci la data di restituzione	Il sistema reindirizza l'utente all'interfaccia dei libri disponibili
3	Data Consegna precedente a quella odierna	Scelta Libro Valid Scelta Data Prevista restituzione <b>[ERROR]</b>	L'utente ha effettuato il login	{Scelta Libro: "Ingegneria del Software", Data Prevista restituzione: "10/03/2000"}	Per favore inserire una data futura.	Il sistema reindirizza l'utente all'interfaccia dei libri disponibili
4	Data Consegna con formato non valido	Scelta Libro Valid Scelta Data Prevista restituzione <b>[ERROR]</b>	L'utente ha effettuato il login	{Scelta Libro: "Ingegneria del Software", Data Prevista restituzione: "40/20/198"}	Per favore inserire una data del tipo DD/MM/YY YY valida.	Il sistema reindirizza l'utente all'interfaccia dei libri disponibili
5	Data Prevista restituzione non selezionata	Scelta Libro Valid Scelta Data Prevista restituzione <b>[ERROR]</b>	L'utente ha effettuato il login	{Scelta Libro: "Ingegneria del Software", Data Prevista restituzione: ""}	Seleziona un libro e inserisci la data di restituzione	Il sistema reindirizza l'utente all'interfaccia dei libri disponibili

## 4.1 Diagramma delle classi



## 4.1.1 Pattern BCED

### 4.1.1.1 Package Boundary

#### Registrazione utente:

Per quanto riguarda il caso d'uso di *Registrazione Utente* è stata creata una singola classe all'interno del package **Boundary**.

Questa classe gestisce l'interfaccia grafica iniziale in cui vengono presentati all'utente due pulsanti: "**Accedi**" e "**Registrati**".

Se l'utente seleziona "Registrati", viene reindirizzato a una nuova schermata in cui può inserire i propri dati personali. Ogni campo testuale è accompagnato da un'icona informativa che, al passaggio del cursore, mostra i requisiti di validità (ad esempio: lunghezza minima, assenza di numeri, ecc.).

I campi sono dotati di *placeholder* che rendono ogni campo identificabile, e di un bordo dinamico che cambia colore in base alla validità dell'input: rosso per input non valido, verde per input corretto, con aggiornamento in tempo reale durante la digitazione.

Una volta premuto il tasto "**Conferma dati**", se tutti i dati sono corretti, l'utente sarà registrato al sistema, altrimenti verrà mostrato un messaggio di errore specifico.

È inoltre presente un pulsante "**Vai indietro**", che consente all'utente di tornare alla schermata principale.

#### Prenotazione Libro:

Per il caso d'uso *Prenotazione Libro* sono state realizzate tre classi all'interno del package *Boundary*, al fine di offrire un'interfaccia utente user friendly ed intuitiva, ma allo stesso tempo coerente con i requisiti funzionali indicati dagli stakeholder.

Il flusso inizia dalla schermata *Profilo Utente*, implementata dalla classe "**FormProfiloUtente**", che consente all'utente di visualizzare e modificare, tramite la propria area personale, le proprie informazioni (Nome, Cognome ed e-mail), credenziali e immagine di profilo, con la possibilità di mostrare o nascondere la password.

Nell'area personale sarà anche presente un bottone "Menu a tendina" attraverso il quale l'utente potrà accedere a diverse funzionalità: visualizzazione dei libri prenotati, cronologia delle consultazioni, gestione delle prenotazioni attive ed effettuazione di una nuova prenotazione.

In particolare, cliccando su "Effettua Prenotazione", si passerà alla seconda fase della prenotazione, che corrisponde alla classe "**PrenotazioneLibro**". In questa schermata il cliente avrà accesso al catalogo dei libri disponibili e potrà selezionare un libro e specificarne una data di prevista restituzione.

Una volta che il cliente clicca il bottone "Conferma prenotazione", in caso di inserimento corretto, la prenotazione viene registrata e viene istanziata la classe "**RicevutaPrenotazione**", che mostra una ricevuta riepilogativa contenente tutte le informazioni necessarie per il ritiro della copia.



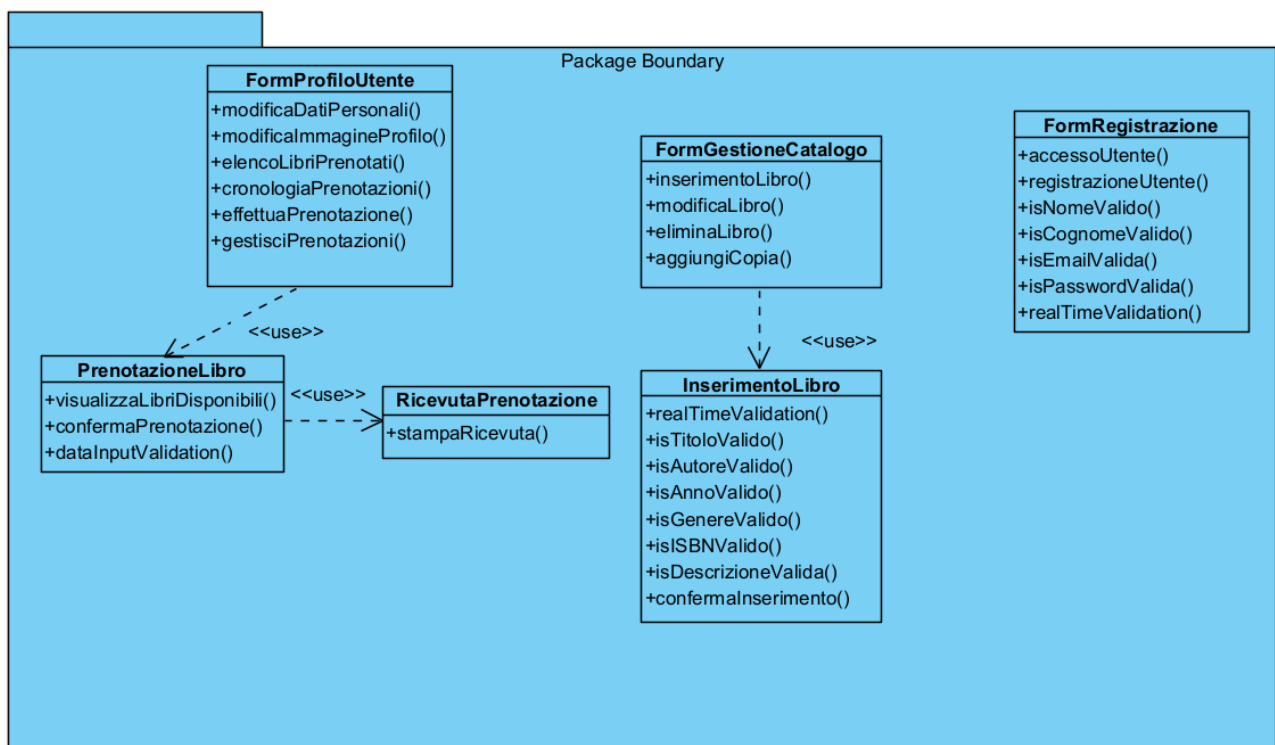
### Inserimento Libro:

Per il caso d'uso *Inserimento Libro* sono state realizzate due classi all'interno del package Boundary, con lo scopo di fornire un'interfaccia chiara e funzionale destinata all'amministratore.

La prima classe, "**FormGestioneCatalogo**", presenta una schermata iniziale contenente pulsanti per accedere rapidamente alle operazioni di modifica, eliminazione, aggiunta di copie e inserimento di nuovi libri.

Selezionando l'opzione "Aggiungi Libro", l'amministratore verrà reindirizzato ad una nuova schermata, implementata dalla classe "**InserimentoLibro**", nella quale potrà inserire le informazioni del libro che vuole aggiungere, in particolare: titolo, autore, anno di pubblicazione, genere, ISBN e descrizione.

Come per l'interfaccia grafica della registrazione avremo anche in questo caso delle icone informative, che al passaggio del cursore mostreranno i requisiti di validità, e la validazione in tempo reale rappresentata dal cambio dei colori dei bordi dei campi testuali.



#### 4.1.1.2 Package Controller

Questo package contiene gli oggetti che percepiscono gli eventi generati dalle interazioni con l'interfaccia utente e ne demandano la gestione all'unico componente del sistema software responsabile della gestione della Business Logic, il package Entity.

##### Registrazione Utente:

Per la realizzazione di questo caso d'uso è stata sviluppata la classe “**ControllerRegistrazione**” che attraverso il metodo *registraUtente* preleva dal boundary i dati inseriti dall'utente e li passa all'entità corrispondente, delegando la registrazione del nuovo cliente al sistema di autenticazione.

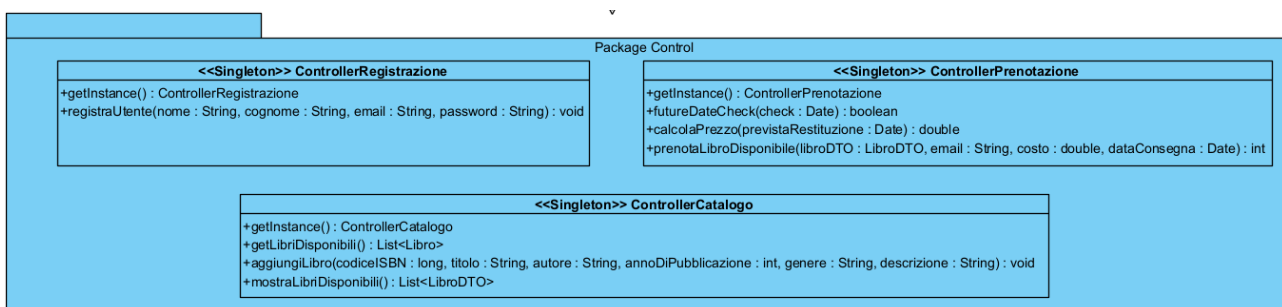
##### Prenotazione Libro:

Per la gestione del caso d'uso *Prenotazione Libro* è stata sviluppata la classe “**ControllerPrenotazione**”, incaricata di coordinare le operazioni relative alla prenotazione da parte dell'utente.

Questa classe include metodi per verificare che la data selezionata sia futura, per calcolare il costo complessivo della prenotazione e per confermare l'operazione tramite il metodo *prenotaLibroDisponibile*, che gestisce l'effettiva assegnazione di una copia disponibile. In supporto a tale processo è stato inoltre utilizzato il “**ControllerCatalogo**”, responsabile della visualizzazione del catalogo dei libri disponibili al cliente.

##### Inserimento Libro:

Per l'implementazione del caso d'uso Inserimento Libro è stata creata la classe “**ControllerCatalogo**”, che attraverso il metodo *aggiungiLibro()* mette in comunicazione il boundary, ovvero i dati del nuovo libro inseriti dall'amministratore, e l'entity, con il fine completare l'operazione di inserimento sul database.



### 4.1.1.3 Package Entity

Il Package Entity contiene tutti gli oggetti che rappresentano la semantica delle entità del dominio applicativo e corrispondono alle strutture dati presenti all'interno del database di persistenza.

#### Registrazione Utente:

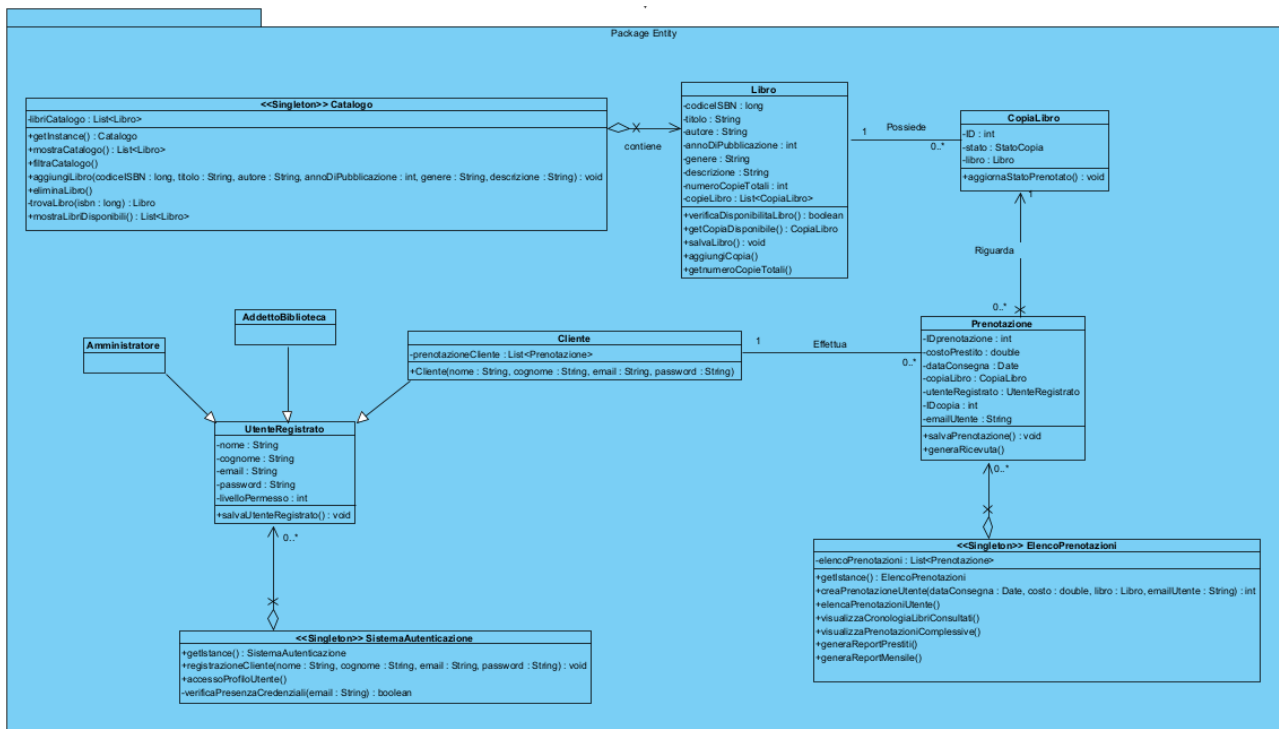
Per la realizzazione del caso d'uso Registrazione Utente è stato necessario realizzare una classe **"UtenteRegistrato"** e l'information expert associato, identificato nella classe **"SistemaAutenticazione"**.

#### Prenotazione Libro:

Per questo caso d'uso è stato necessario realizzare le classi **"Prenotazione"**, l'information expert associato, ovvero la classe **"ElencoPrenotazioni"**. Il caso d'uso utilizza anche le classi **"CopiaLibro"**, **"Libro"**, e l'information expert del libro implementato tramite la classe **"Catalogo"**.

#### Inserimento Libro:

Per la realizzazione del caso d'uso Inserimento Libro è stato necessario realizzare una classe **"Libro"** e l'information expert associato, identificato nella classe **"Catalogo"**.



#### 4.1.1.4 Package Database

Da questo punto di vista, il Package Database contiene tutte le classi responsabili dell'estrazione dei dati dal DB, esponendo una vera e propria interfaccia che di fatto rende indipendenti le classi della Business Logic (Entity) dalla tecnologia di persistenza utilizzata.

In particolare, tra le strategie di risoluzione del problema dell'**impedance mismatch**, che nasce dalla mancata corrispondenza tra il modello Object Oriented e quello relazionale, si è deciso di adottare quella delle classi **DAO** (Data Access Objects), che consiste nell'utilizzo di appositi oggetti per l'accesso ai dati.

Ognuna di queste classi conterrà i metodi CRUD per l'interrogazione e la manipolazione della corrispondente classe di dominio (*query*), implementati in funzione di un'ulteriore classe, **DBConnectionManager**, che costituisce di fatto l'unico punto di accesso vero e proprio al DB, sfruttando i metodi che questa mette a disposizione.

##### Registrazione Utente:

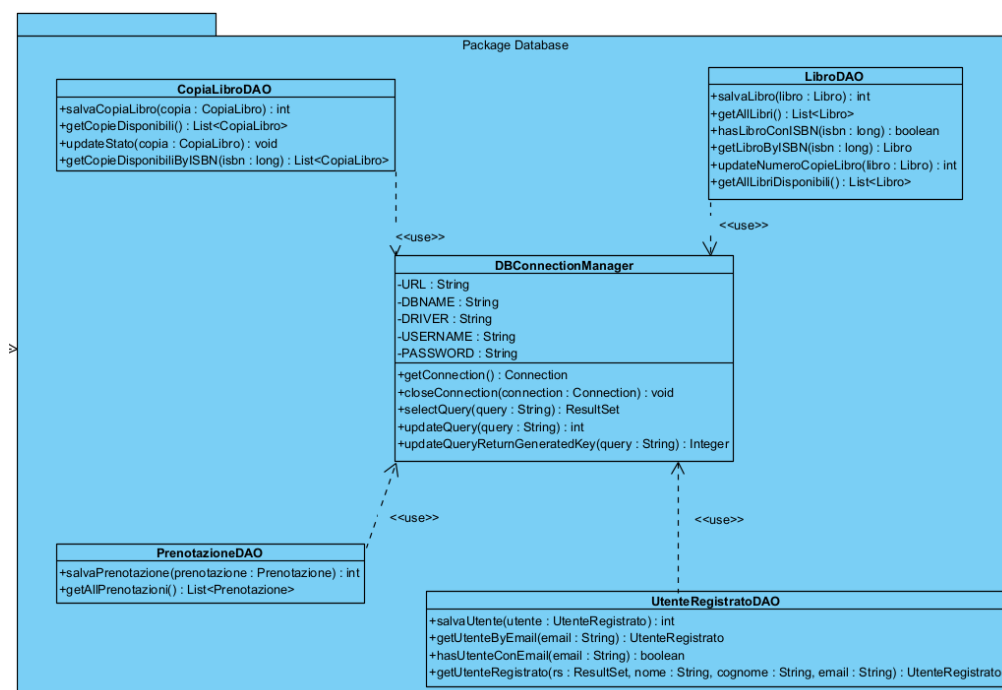
Per la realizzazione del caso d'uso "Registrazione Utente" è stato creato "**UtenteRegistratoDAO**", responsabile del salvataggio sul data base della tupla relativa al nuovo cliente.

##### Prenotazione Libro:

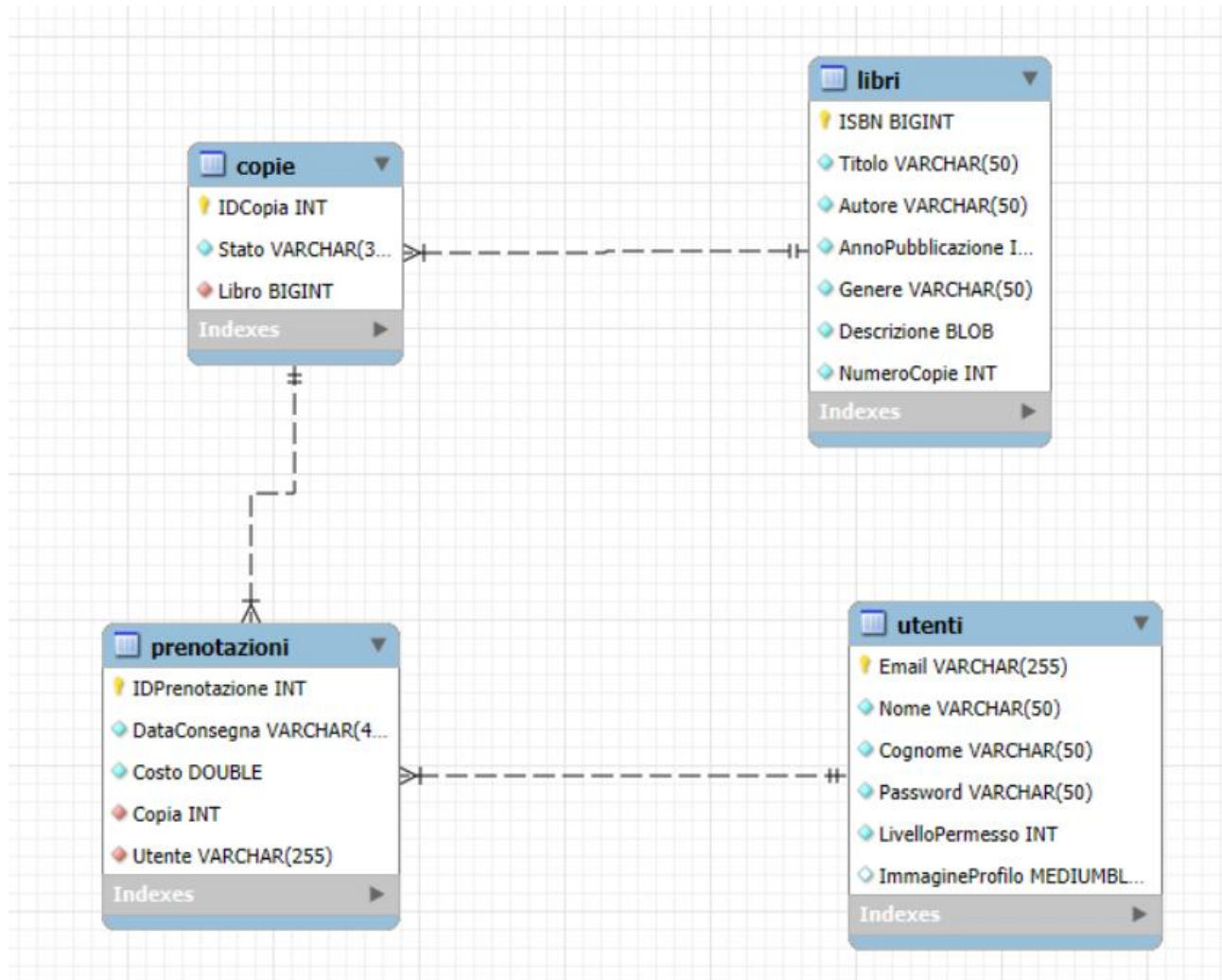
Per la realizzazione del caso d'uso "Prenotazione Libro" è stato creato "**PrenotazioneDAO**", responsabile del salvataggio sul data base della tupla relativa alla prenotazione effettuata dal cliente. Inoltre, utilizza "**LibroDAO**" e "**CopiaLibroDAO**" per permettere al cliente la corretta visualizzazione del catalogo dei libri disponibili e cambiare lo stato della copia prenotata.

##### Inserimento Libro:

Per la realizzazione del caso d'uso "Inserimento Libro" è stato creato "**LibroDAO**", responsabile del salvataggio sul data base della tupla relativa al nuovo libro inserito dall'amministratore.







### Chiavi primarie del data-base:

- La chiave primaria della tabella **copie** è stata assegnata all'attributo **IDCopia**, progettato per identificare univocamente ciascuna copia presente nella biblioteca.
- La chiave primaria della tabella **libri** è stata assegnata all'attributo **ISBN**, il quale rappresenta un identificativo univoco per ogni libro.
- La chiave primaria della tabella **prenotazioni** è stata assegnata all'attributo **IDPrenotazione**, utilizzato per identificare univocamente ogni prenotazione effettuata.
- La chiave primaria della tabella **utenti** è stata assegnata all'attributo **Email**, ritenuto sufficiente per l'identificazione univoca di ciascun utente.

### Associazioni del data-base:

- **Relazione uno-a-molti tra Libri e Copie**: un libro può essere associato a più copie fisiche presenti nella biblioteca.
- **Relazione uno-a-molti tra Copie e Prenotazioni**: una prenotazione è riferita a una singola copia, ma una stessa copia può essere oggetto di più prenotazioni nel tempo.
- **Relazione uno-a-molti tra Utenti e Prenotazioni**: una prenotazione è riferita a un solo utente, ma ciascun utente può effettuare più prenotazioni.





## Traduzione del modello E/R:

- Copie(IDCopia , Stato, libri:Libro)
- Libri(ISBN, Titolo, Autore, AnnoPubblicazione, Genere, Descrizione, NumeroCopie)
- Prenotazioni(IDPrenotazione, DataConsegna, Costo, copie:Copia, utenti:Utente)
- Utenti(Email, Nome, Cognome, Password, LivelloPermesso, ImmagineProfilo)

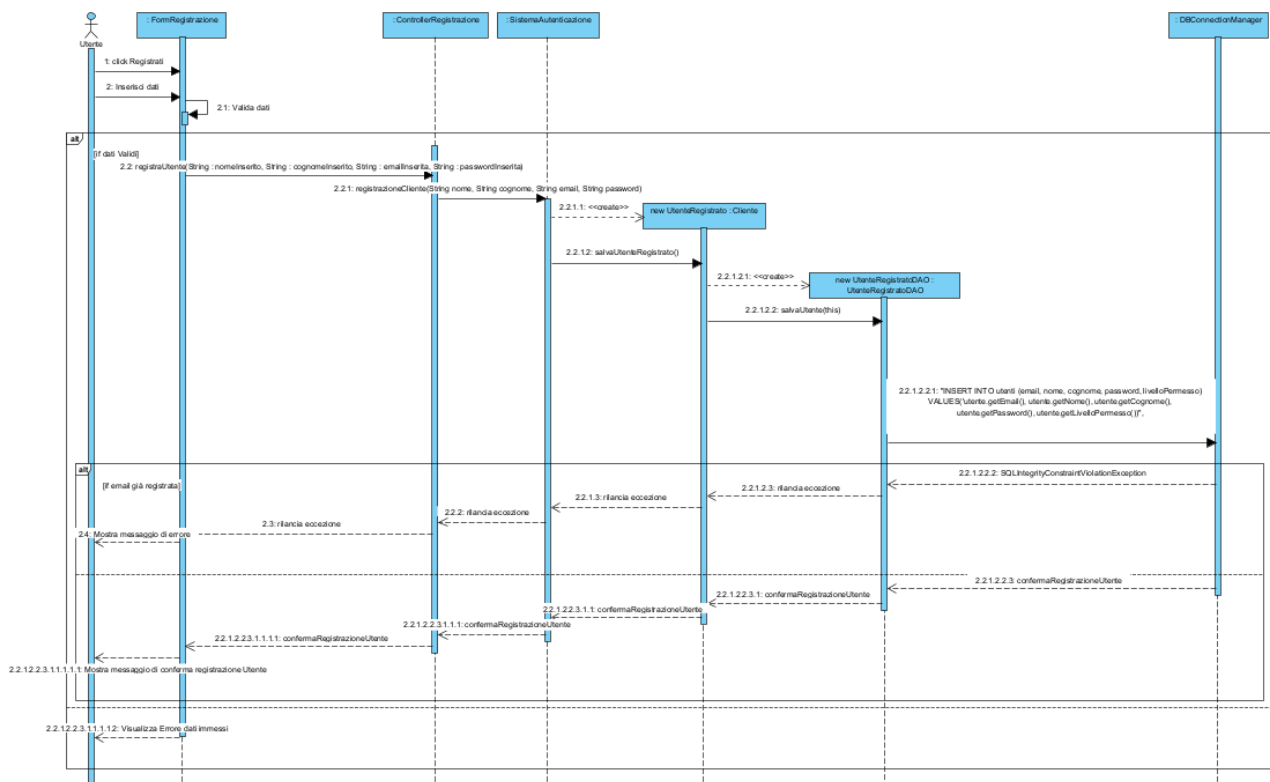
## Osservazioni importanti riguardanti il modello E/R:

Tutte le chiavi primarie sono state definite in accordo con i principi della progettazione relazionale, garantendo l'univocità e l'atomicità mediante l'uso di identificatori specifici. Inoltre, gli identificatori delle tabelle **Prenotazioni** e **Copie** (IDPrenotazione e IDCopia) sono stati configurati come **auto-incrementali** all'interno del database, al fine di garantire automaticamente la generazione di valori univoci e progressivi per ciascun nuovo record inserito.

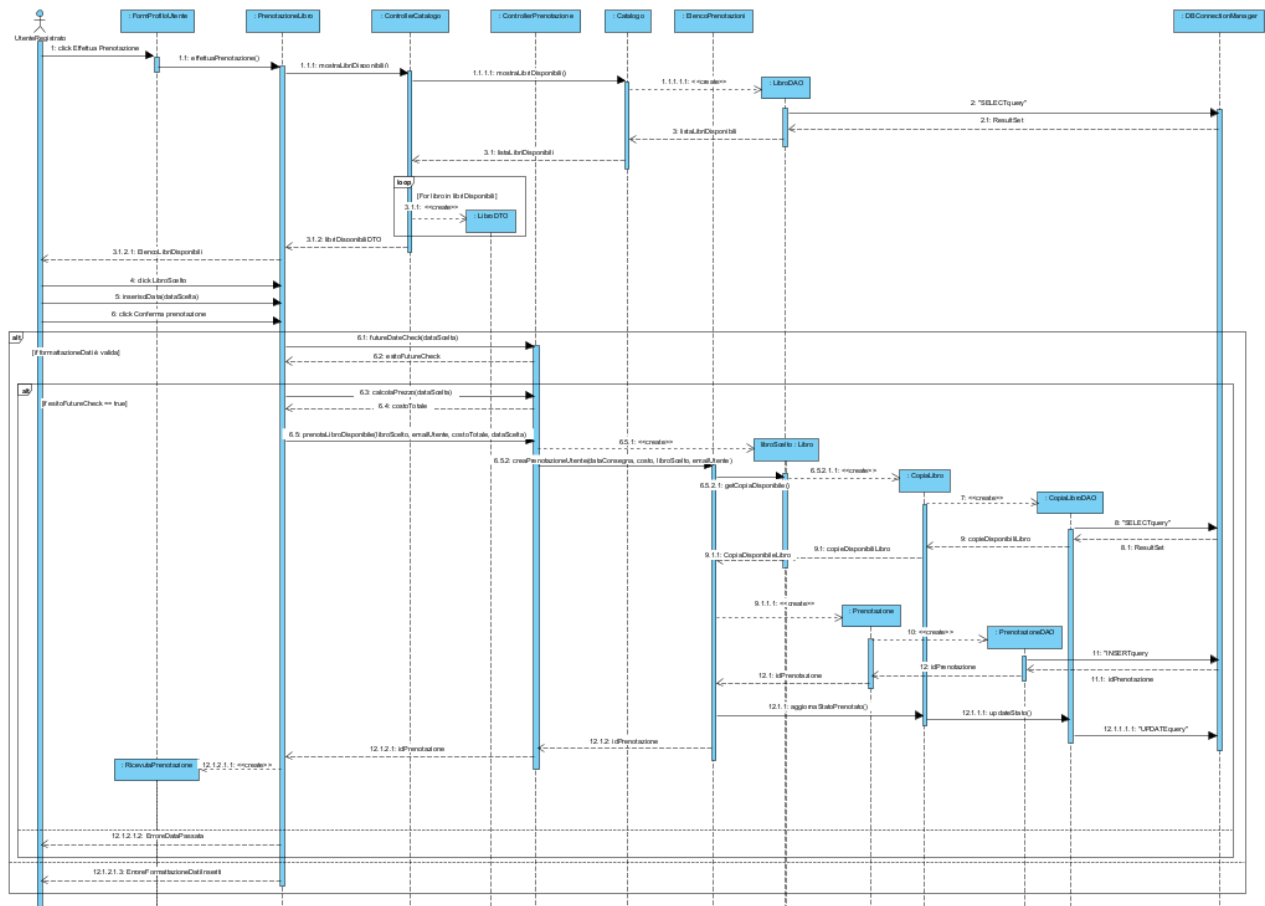
Nella tabella **Prenotazioni** sono presenti chiavi esterne (copie e utenti) per assicurare l'integrità referenziale, ovvero che ogni prenotazione sia associata a una copia e a un utente esistenti all'interno del database.

## 4.2 Diagrammi di sequenza

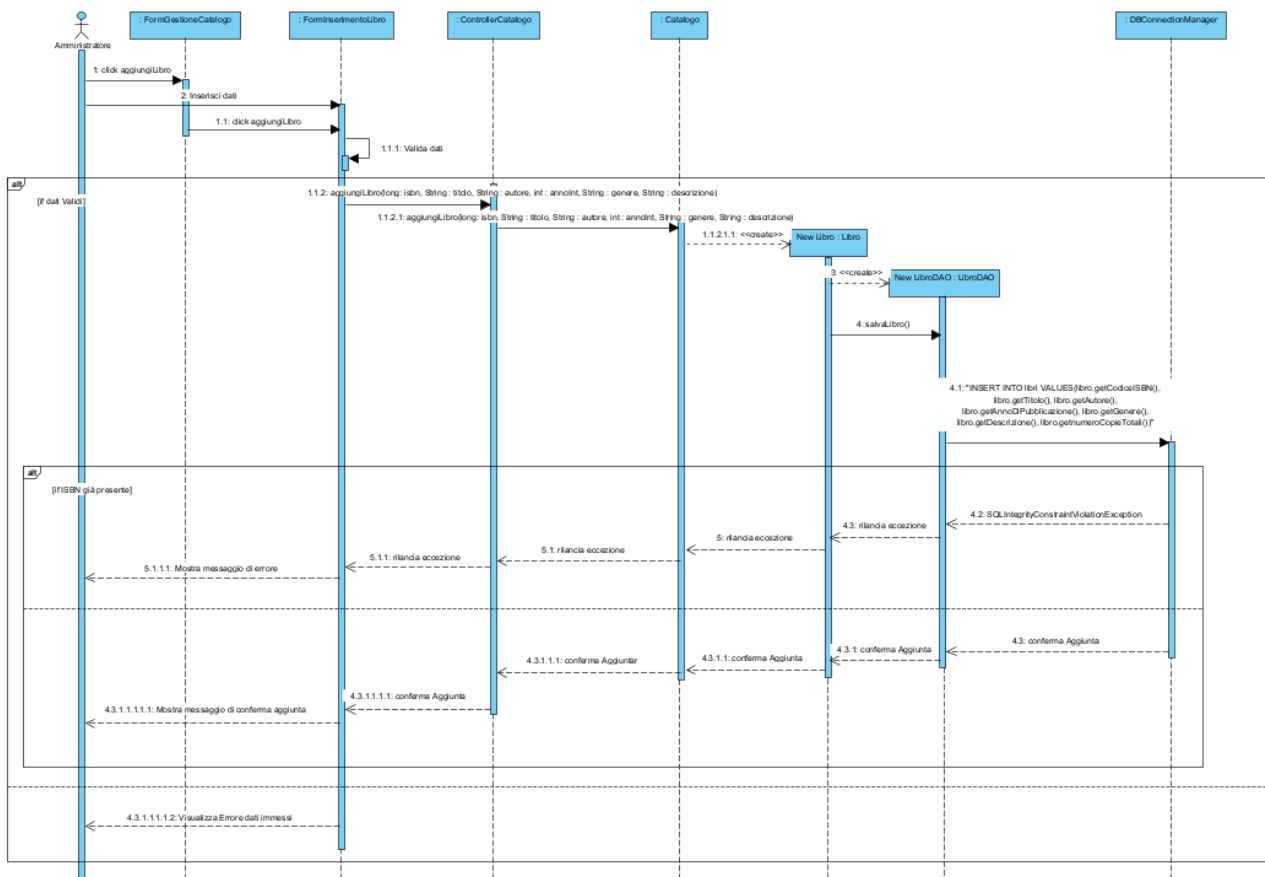
### 4.2.1 Registrazione Utente



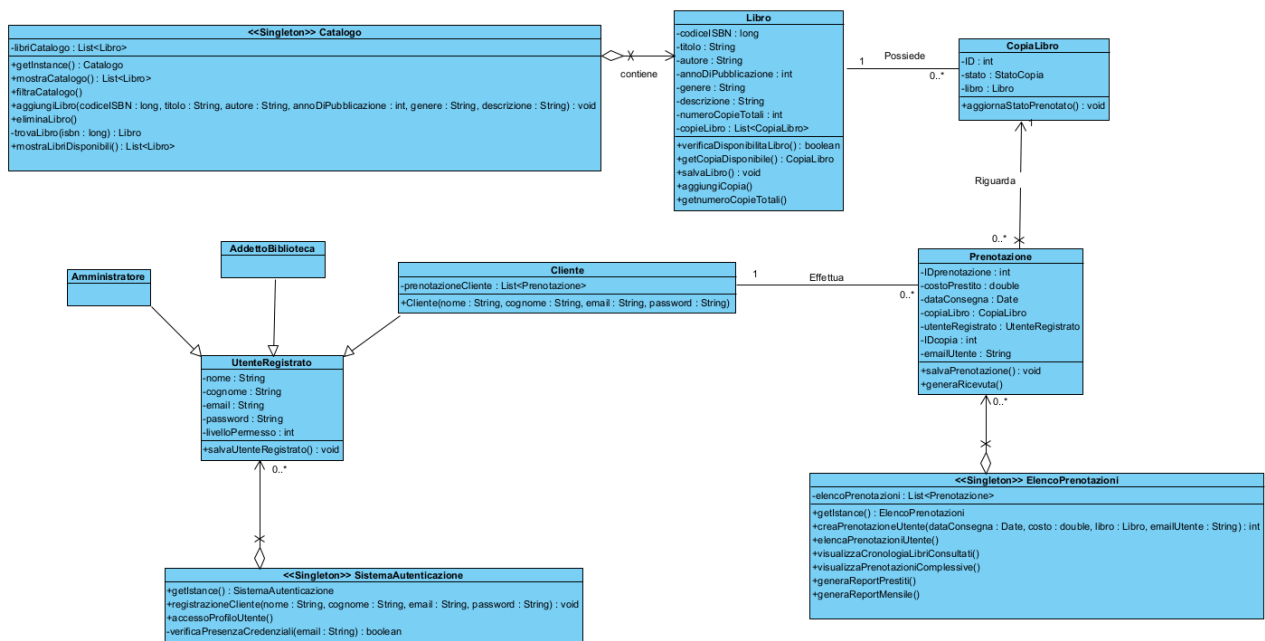
### 4.2.2 Prenotazione Libro



### 4.2.3 Inserimento Libro



I sequence progettati sono stati fondamentali per la corretta implementazione dell'applicazione software ed ha fatto nascere la necessità di definire ulteriori classi, metodi e funzioni, che hanno arricchito passo dopo passo il **Diagramma delle Classi di Progettazione**, fino ad ottenere la seguente versione finale:



## 5. Implementazione

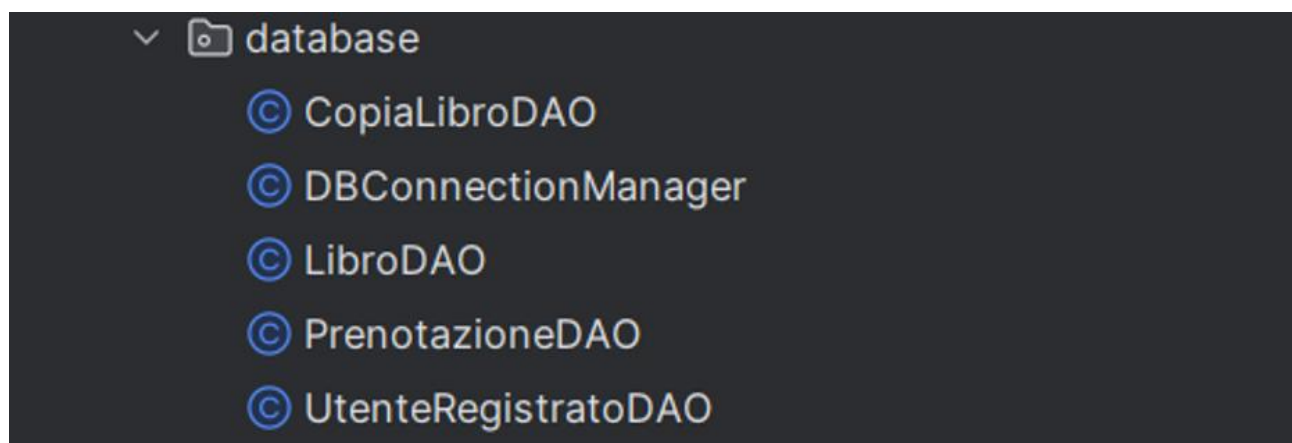
### 5.1 Package Database

All'interno del package database è stata creata la classe “**DBConnectionManager**”, in grado di gestire la connessione ad un database MySQL.

All'interno della classe sono definiti i parametri di connessione (URL, nome del database, utente, password) e fornisce metodi statici per aprire e chiudere la connessione. Oltre a questo, sono forniti i metodi per eseguire query e in generale le funzioni CRUD sul database (tramite i metodi *selectQuery()* e *updateQuery()*).

Inoltre, all'interno del package sono presenti varie classi DAO (Data Access Object) che si occupano di gestire l'accesso ai dati per specifiche entità del progetto, come libri, utenti, prenotazioni o copie.

Ogni DAO incapsula le operazioni CRUD relative alla propria entità, utilizzando i metodi di connessione forniti da DBConnectionManager.



Nella pratica l'intero package database nasce con l'obiettivo di creare un “ponte” tra l'applicazione effettiva e la base di dati, separando la logica di accesso al database alla logica applicativa.

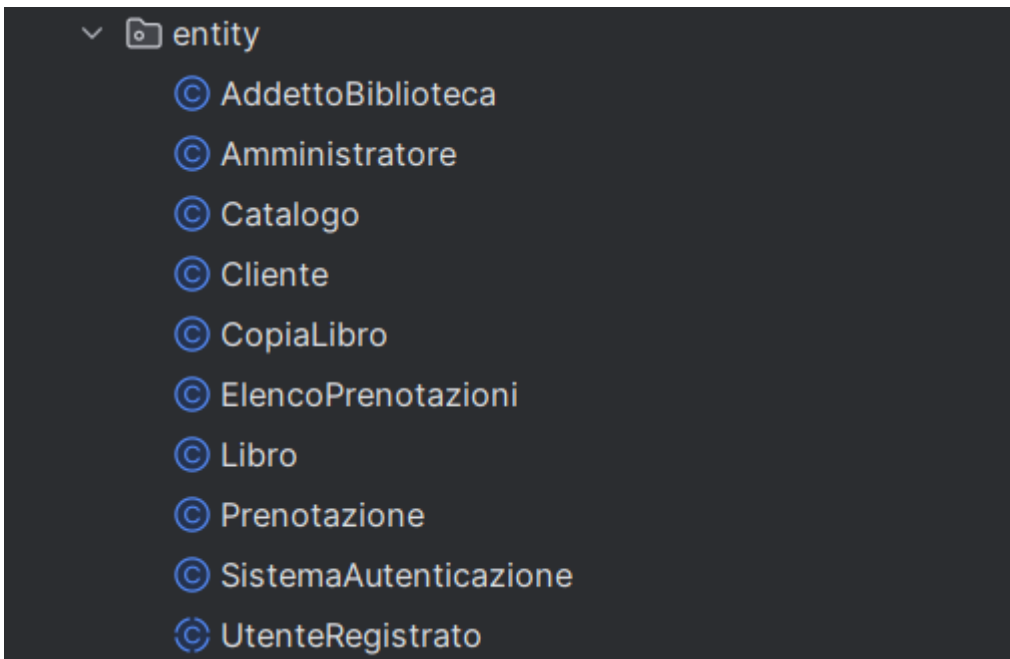


## 5.2 Package Entity

---

All'interno del package entity è stata creata una classe per ogni entità definita all'interno del class diagram. La fase di analisi è risultata molto utile al fine di avere in mente un flusso di lavoro coerente e funzionale.

Ogni classe rappresenta un'entità significativa del dominio applicativo e incapsula la relativa business logic, delegata dal controller, il quale funge da elemento di collegamento tra la logica applicativa e il livello di interazione con gli attori, rappresentato dai componenti boundary.



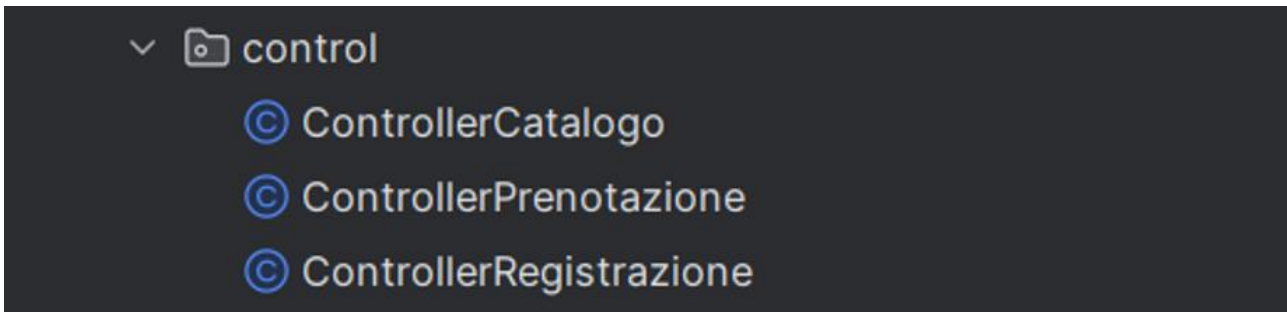
La modularità introdotta dal pattern architetturale BCED ha reso possibile una suddivisione chiara delle responsabilità, consentendo al team di lavorare in parallelo pur mantenendo un coordinamento efficace, grazie a una fase di analisi svolta in modo collaborativo. Tale approccio ha dimostrato la sua efficacia nel momento in cui si è reso necessario modificare l'implementazione interna delle entità: il fatto che l'interfaccia pubblica (API) sia rimasta invariata ha evitato ripercussioni sugli altri package, confermando la bontà della progettazione modulare adottata.



## 5.3 Package Controller

---

All'interno del package Controller abbiamo adottato una scelta duplice, non solo abbiamo scelto di creare i Controller in base al caso d'uso che vanno a realizzare, ma c'è stato un raggruppamento anche in base alla funzionalità effettiva della classe, creando classi coese in quanto unità logiche.

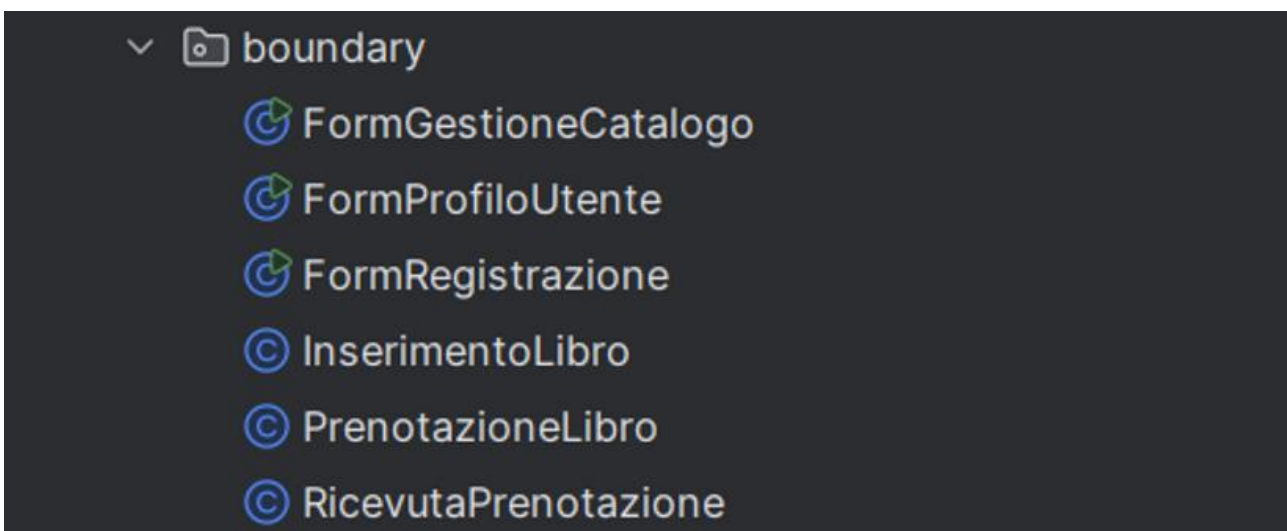


Ad esempio, all'interno del “**ControllerCatalogo**” sono gestite tutte le operazioni relative alla gestione dei libri e delle copie. Oltre ai metodi strettamente legati al caso d'uso associato, sono stati implementati anche altri metodi che, pur non essendo direttamente connessi a tale caso d'uso, rientrano comunque nell'ambito di responsabilità funzionale del controller. Questa scelta ha permesso di mantenere la coesione interna della classe, evitando una frammentazione eccessiva delle logiche applicative.

## 5.4 Package Boundary

---

All'interno del package boundary sono state realizzate le interfacce per permettere l'accesso alle funzionalità dell'applicazione da parte di un addetto, di un cliente o di un amministratore.



Anche in questo caso abbiamo scelto un approccio per favorire la modularità, creando delle classi eseguibili (che hanno un main) per dare inizio al caso d'uso effettivo, ma dividendo comunque il flusso di lavoro su più classi per separare l'interfaccia principale dall'effettiva implementazione applicativa.



## 5.5 Package DTO

L'introduzione di tale package, estraneo al pattern BCED, nasce dall'esigenza di mostrare sulla GUI collezioni di elementi.

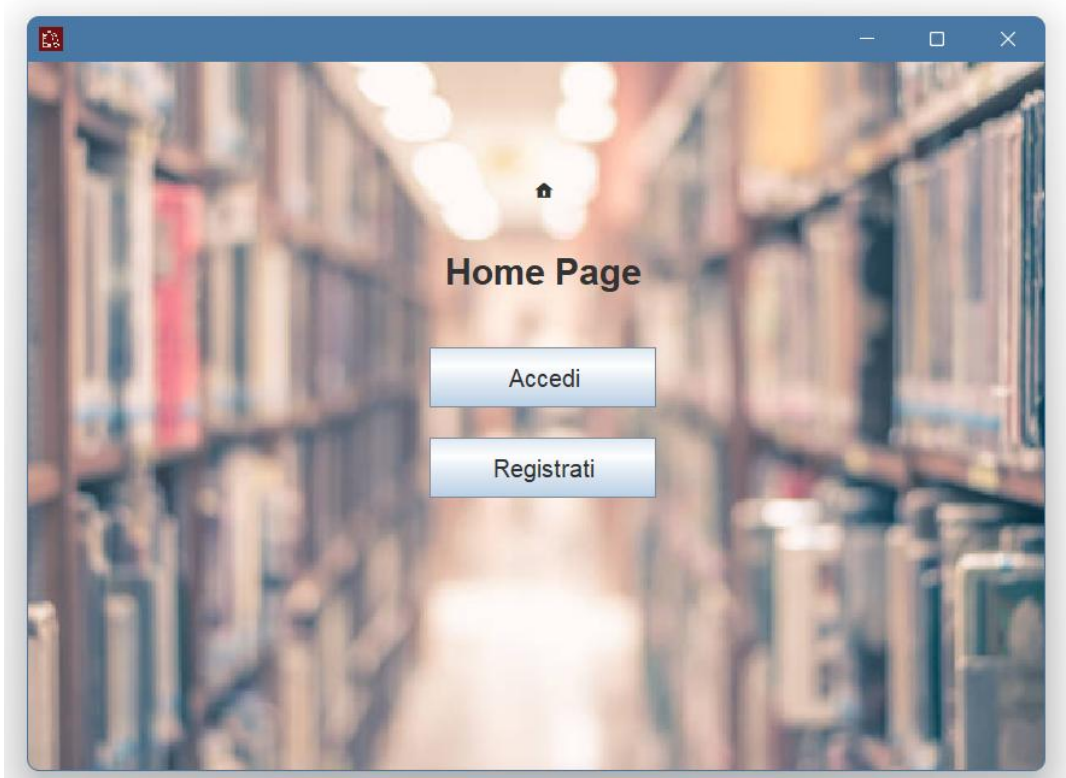
Da questo punto di vista, il problema principale è proprio quello che, qualora una determinata chiamata a funzione restituisse alla GUI un elenco di entity, questa, per poterlo visualizzare correttamente a video, dovrebbe conoscere di fatto la struttura interna di tale classe Entity, ma ciò porterebbe con sé un accoppiamento troppo elevato e quindi una chiara violazione dei vincoli del pattern a livelli adottato.

Si introduce allora il concetto di **Data Transfer Object (DTO)**, un oggetto in grado di trasportare dati tra processi (nel caso in oggetto tra livelli). Le classi DTO hanno tipicamente una struttura che rispecchia quella dell'entity di cui vanno a supporto, in particolare gli attributi coincidono con quelli dell'entity che si intendono visualizzare a schermo.



## 5.6 Interfacce Grafiche da Eclipse (con JavaSwing)

### 5.6.1 Registrazione Utente





Inserisci i dati nei campi di testo indicati:

Antonio1 ⓘ

Di Giorgio ⓘ

test@ ⓘ

..... ⓘ

Conferma dati

Vai indietro

## 5.6.2 Prenotazione Libro

Nome  
Matteo

Cognome  
Bottari

Email  
matteo@ingSW.it

Password  
.....

Modifica i dati personali

Modifica immagine profilo

Elenco Libri Prenotati

Cronologia Prenotazioni

Effettua Prenotazione

Gestisci Prenotazioni

Menu a tendina





Catalogo Libri Disponibili

Seleziona un libro disponibile:

- ☐ Fisica delle meraviglie | Giulio Rossi | 9781122334455 | 2021
- ☐ Il giardino segreto | Sara Neri | 9785566778899 | 2015
- ☐ Manuale di cucina italiana | Carlo Bianchi | 9786677889900 | 2019
- ☐ Il mistero del tempo | Luca Bianchi | 9781234567890 | 2020

Data prevista restituzione:

Conferma prenotazione

**\* Grazie per la Prenotazione! \***

**!!! ATTENZIONE !!!**

**Non perdere la ricevuta!**

Sarà necessaria per il ritiro del libro

--- Conserva la ricevuta stampata o digitale ---

>> Torna al Menu Principale

Ricevuta Prenotazione

Identificativo prenotazione: 30

Titolo: Fisica delle meraviglie

Codice ISBN: 9781122334455

Email utente: matteo@ingSW.it

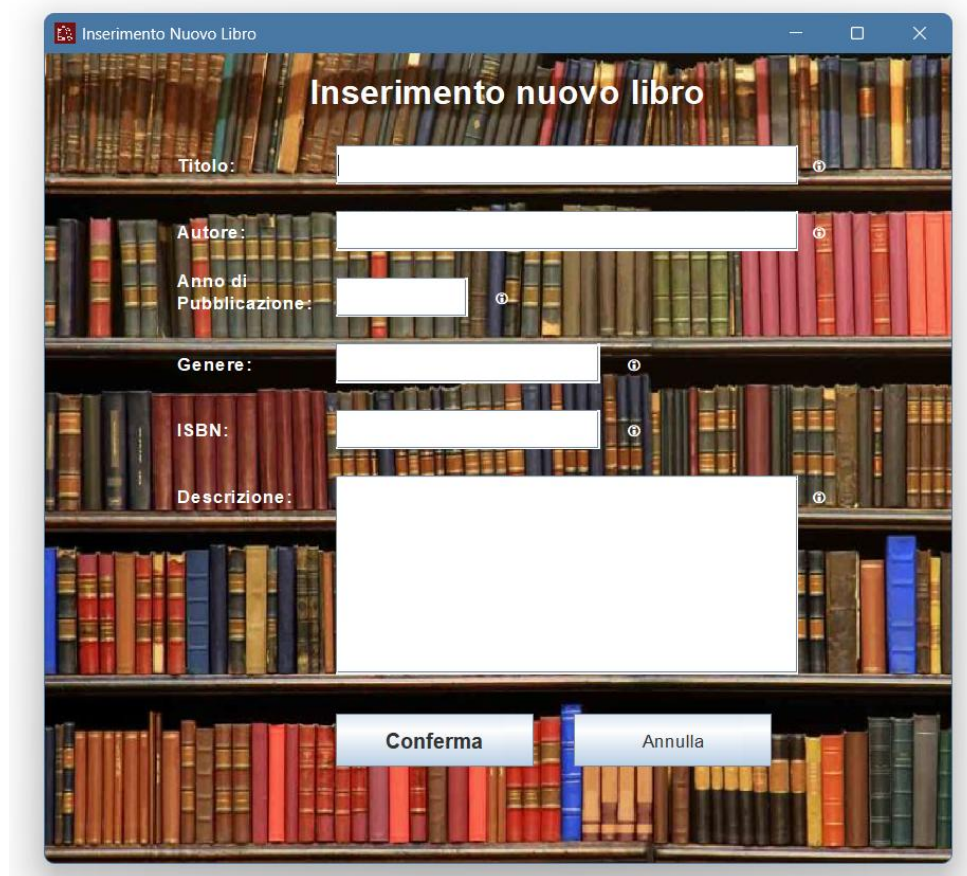
Data restituzione: 27/07/2025

**Prezzo: € 2,75**

Stampa ricevuta

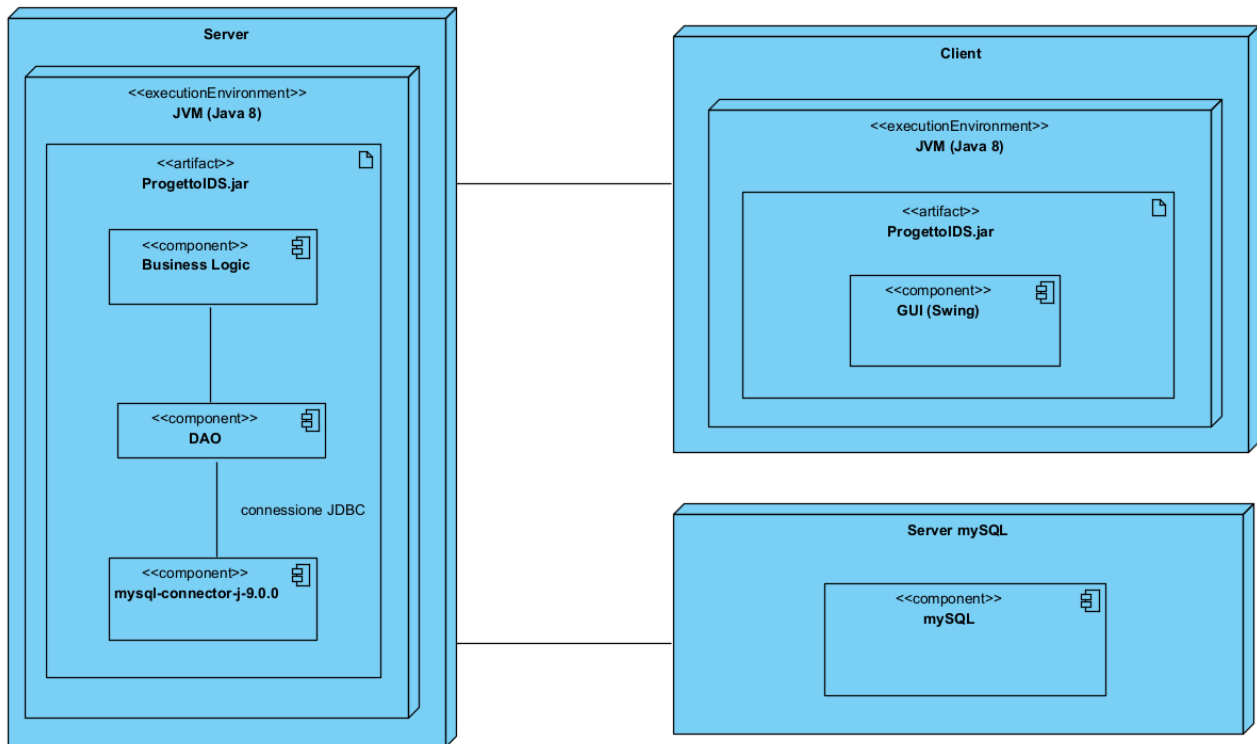


### 5.6.3 Inserimento Libro



## 5.7 Diagramma di Deployment

I diagrammi di deployment sono utilizzati per mostrare l'architettura fisica del sistema software realizzato; sono particolarmente utili per valutare, durante lo sviluppo, come un'applicazione si distribuisce tra le varie macchine.



## 6. Test Funzionale

## 6.1 Registrazione Utente

TEST SUITE							
Test Case ID	Descrizione	Classi di equivalenza coperte	Pre-condizioni	Input	Output attesi	Post-condizioni attese	Esito (FAIL, PASS)
1	Tutti input validi	Nome valido Cognome valido Email valida Password valida	-	{Nome: "Antonio", Cognome: "Di Giorgio", Email: "antonio.email@gmail.com", Password: "Password123!"}	L'utente può accede alla propria aera personale	L'utente è registrato	PASS
2	Stringa Nome > 50 caratteri	Nome > 50 caratteri [ERROR] Cognome valido Email valida Password valida	-	{Nome: "Japocoooooooooooooooooooooooooooooo ooooo", Cognome: "Ferrante", Email: "jacopecomail@gmail.com", Password: "Pass987@l!"}	Il nome inserito non è valido. Per favore, controlla e riprova.	L'utente è reindirizzato alla schermata di registrazione	PASS
3	Stringa Nome contenente simboli che non sono caratteri	Nome con simboli [ERROR], Cognome valido Email valida Password valida	-	{Nome: "Anto_d", Cognome: "Di Giorgio", Email: "antonio.email@gmail.com", Password: "Password123!"}	Il nome inserito non è valido. Per favore, controlla e riprova.	L'utente è reindirizzato alla schermata di registrazione	PASS
4	Stringa Nome < 1 carattere	Nome < 1 carattere [ERROR], Cognome valido Email valida Password valida	-	{Nome: "", Cognome: "D'Amico", Email: "jacopocomail@gmail.com", Password: "Pass987@l!"}	Il nome inserito non è valido. Per favore, controlla e riprova	L'utente è reindirizzato alla schermata di registrazione	PASS
5	Stringa Nome con prima lettera non maiuscola	Nome senza maiuscola [ERROR],	-	{Nome: "jacopo", Cognome: "Ferrante", Email: "jacopocomail@gmail.com", Password: "Pass987@l!"}	Il nome inserito non è valido. Per favore, controlla e riprova	L'utente è reindirizzato alla schermata di registrazione	PASS



		Cognome valido Email valida Password valida					
6	Stringa Cognome non contenente solo caratteri o apostrofo	Nome valido Cognome stringa con simboli [ERROR] Email valida Password valida	-	{Nome: "Antonio", Cognome: "Di_Giorgio!", Email: "antonio.email@gmail.com", Password: "Password123!"}	Il cognome inserito non è valido. Per favore, controlla e riprova	L'utente è reindirizzato alla schermata di registrazione	PASS
7	Stringa Cognome > 50 caratteri	Nome valido Cognome [ERROR] Email valida Password valida	-	{Nome: "Matteo", Cognome: "Bottariiiiiiiiiiiiiiiiiiiiiiiiiiiiiiiiiiiii", Email: "utente.gmail.com", Password: "Ciao123_"}	Il cognome inserito non è valido. Per favore, controlla e riprova	L'utente è reindirizzato alla schermata di registrazione	PASS
8	Stringa Cognome < 1 carattere	Nome valido Cognome [ERROR] Email valida Password valida	-	{Nome: "Matteo", Cognome: "", Email: "utente.gmail.com", Password: "Ciao123_"}	Il cognome inserito non è valido. Per favore, controlla e riprova	L'utente è reindirizzato alla schermata di registrazione	PASS
9	Stringa email non contenente '@'	Nome valido Cognome valido Email [ERROR] Password valida	-	{Nome: "Matteo", Cognome: "Bottari", Email: "utente.gmail.com", Password: "Ciao123_"}	L'email inserita ha un formato non corretto. Per favore, riprova.	L'utente è reindirizzato alla schermata di registrazione	PASS
10	Stringa email associata ad un account già esistente	Nome valido Cognome valido Email [ERROR] Password valida	L'email inserita è già associata ad un account registrato	{Nome: "Matteo", Cognome: "Bottari", Email: "utente.registrato@gmail.com", Password: "Ciao123_"}	L'email inserita è già stata utilizzata!	L'utente è reindirizzato alla schermata di registrazione	PASS
11	Stringa Password < 8 caratteri	Nome valido Cognome valido Email valida Password [ERROR]	-	{Nome: "Matteo", Cognome: "Bottari", Email: "ma.bo@studenti.unina.it", Password: "Ciao12_"}	La password non soddisfa i requisiti di sicurezza. Per favore, riprova.	L'utente è reindirizzato alla schermata di registrazione	PASS
12	Stringa password non contenente maiuscole	Nome valido Cognome valido Email valida Password [ERROR]	-	{Nome: "Matteo", Cognome: "Bottari", Email: "ma.bo@studenti.unina.it", Password: "ciao123_"}	La password non soddisfa i requisiti di sicurezza. Per favore, riprova.	L'utente è reindirizzato alla schermata di registrazione	PASS



13	Stringa Password non contenente minuscole	Nome valido Cognome valido Email valida Password [ERROR]	-	{Nome: "Matteo", Cognome: "Bottari", Email: "ma.bo@studenti.unina.it", Password: "CIAO123_"}	La password non soddisfa i requisiti di sicurezza. Per favore, riprova.	L'utente è reindirizzato alla schermata di registrazione	PASS
14	Stringa Password non contenente caratteri speciali	Nome valido Cognome valido Email valida Password [ERROR]	-	{Nome: "Matteo", Cognome: "Bottari", Email: "ma.bo@studenti.unina.it", Password: "Ciao1234"}	La password non soddisfa i requisiti di sicurezza. Per favore, riprova.	L'utente è reindirizzato alla schermata di registrazione	PASS
15	Stringa Password non contenente numeri	Nome valido Cognome valido Email valida Password [ERROR]	-	{Nome: "Matteo", Cognome: "Bottari", Email: "ma.bo@studenti.unina.it", Password: "CiaoCiao_"}	La password non soddisfa i requisiti di sicurezza. Per favore, riprova.	L'utente è reindirizzato alla schermata di registrazione	PASS
16	Stringa email > 255	Nome valido Cognome valido Email [ERROR] Password valida	-	{Nome: "Matteo", Cognome: "Bottari", Email: "ma.booooooooo(o*255)@studenti.unina.it", Password: "Matteo04_"}	L'email inserita ha un formato non corretto. Per favore, riprova.	L'utente è reindirizzato alla schermata di registrazione	PASS





## 6.2 Prenotazione Libro

TEST SUITE							
Test Case ID	Descrizione	Classi di equivalenza coperte	Pre-condizioni	Input	Output attesi	Post-condizioni attese	Esito (FAIL, PASS)
1	Tutti input validi	Scelta Libro Valida Scelta Data Prevista restituzione	L'utente ha effettuato il login	{Scelta Libro: "Ingegneria del Software", Data Prevista restituzione: "10/04/2026"}	La ricevuta è mostrata a video con un invito a conservarla per ritirare il libro	L'utente ha prenotato un libro di interesse ed è stata restituita una ricevuta	PASS
2	Scelta Libro non selezionato	Scelta Libro [ERROR] Scelta Data Prevista restituzione	L'utente ha effettuato il login	{Scelta Libro: "", Data Prevista restituzione: "10/04/2026"}	Seleziona un libro e inserisci la data di restituzione	Il sistema reindirizza l'utente all'interfaccia dei libri disponibili	PASS
3	Data Consegna precedente a quella odierna	Scelta Libro Valida Scelta Data Prevista restituzione [ERROR]	L'utente ha effettuato il login	{Scelta Libro: "Ingegneria del Software", Data Prevista restituzione: "10/03/2000"}	Per favore inserire una data futura.	Il sistema reindirizza l'utente all'interfaccia dei libri disponibili	PASS
4	Data Consegna con formato non valido	Scelta Libro Valida Scelta Data Prevista restituzione [ERROR]	L'utente ha effettuato il login	{Scelta Libro: "Ingegneria del Software", Data Prevista restituzione: "40/20/198"}	Per favore inserire una data del tipo DD/MM/YYYY valida.	Il sistema reindirizza l'utente all'interfaccia dei libri disponibili	PASS
5	Data Prevista restituzione non selezionata	Scelta Libro Valida Scelta Data Prevista restituzione [ERROR]	L'utente ha effettuato il login	{Scelta Libro: "Ingegneria del Software", Data Prevista restituzione: ""}	Seleziona un libro e inserisci la data di restituzione	Il sistema reindirizza l'utente all'interfaccia dei libri disponibili	PASS



## 6.3 Inserimento Libro

TEST SUITE							
Test Case ID	Descrizione	Classi di equivalenza coperte	Pre-condizioni	Input	Output attesi	Post-condizioni attese	Esito (FAIL, PASS)
1	Tutti input validi	Titolo valido Autore valido Anno di Pubblicazione valido Genere valido ISBN valido Descrizione valida	-	{Titolo: "Il Signore delle Mosche", Autore: "William Golding", Anno di Pubblicazione: "1954", Genere: "Romanzo", ISBN: "9780143124290", Descrizione: "Romanzo di William Golding che narra la storia di un gruppo di ragazzi."}	Il libro ora è presente e visualizzabile nel catalogo della biblioteca	Libro aggiunto al catalogo	PASS
2	Stringa Titolo contenente caratteri speciali	Titolo con simboli [ERROR] Autore valido Anno di Pubblicazione valido Genere valido ISBN valido Descrizione valida	-	{Titolo: "Il Signore delle Mosche_!", Autore: "William Golding", Anno di Pubblicazione: "1954", Genere: "Romanzo", ISBN: "9780143124290", Descrizione: "Romanzo di William Golding che narra la storia di un gruppo di ragazzi."}	Il titolo contiene caratteri speciali non ammessi.	L'amministratore è reindirizzato all'interfaccia per aggiungere Libri.	PASS
3	Stringa Titolo di lunghezza > 50 caratteri	Titolo > 50 [ERROR] Autore valido Anno di Pubblicazione valido Genere valido ISBN valido Descrizione valida	-	{Titolo: "Il Signore delle Moscheeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee", Autore: "William Golding", Anno di Pubblicazione: "1954", Genere: "Romanzo", ISBN: "9780143124290", Descrizione: "Romanzo di William Golding che narra la storia di un gruppo di ragazzi."}	Il titolo non può superare i 50 caratteri.	L'amministratore è reindirizzato all'interfaccia per aggiungere Libri.	PASS





4	Stringa titolo di lunghezza < 1	Titolo [ERROR] Autore Valido Anno di pubblicazione Valido Genere Valido ISBN Valido Descrizione Valido	-	{Titolo: "", Autore: "William Golding", Anno di Pubblicazione: "1954", Genere: "Romanzo", ISBN: "9780143124290", Descrizione: "Romanzo di William Golding che narra la storia di un gruppo di ragazzi."}	Il campo titolo non può essere vuoto.	L'amministratore è reindirizzato all'interfaccia per aggiungere Libri.	PASS
5	Stringa Autore di lunghezza > 50 caratteri	Titolo valido Autore > 50 [ERROR] Anno di Pubblicazione valido Genere valido ISBN valido Descrizione valida	-	{Titolo: "Il Signore delle Mosche", Autore: "William Golding", Anno di Pubblicazione: "1954", Genere: "Romanzo", ISBN: "9780143124290", Descrizione: "Romanzo di William Golding che narra la storia di un gruppo di ragazzi."}	Il campo autore non può superare i 50 caratteri.	L'amministratore è reindirizzato all'interfaccia per aggiungere Libri.	PASS
6	Stringa Autore contenente simboli che non sono caratteri o apostrofo	Titolo valido Autore con simboli [ERROR] Anno di Pubblicazione valido Genere valido ISBN valido Descrizione valida	-	{Titolo: "Il Signore delle Mosche", Autore: "William Golding03_", Anno di Pubblicazione: "1954", Genere: "Romanzo", ISBN: "9780143124290", Descrizione: "Romanzo di William Golding che narra la storia di un gruppo di ragazzi."}	Il campo autore può contenere solo lettere e apostrofi.	L'amministratore è reindirizzato all'interfaccia per aggiungere Libri.	PASS
7	Stringa Autore di lunghezza < 1	Titolo Valido Autore [ERROR] Anno di pubblicazione Valido Genere Valido ISBN Valido Descrizione Valido	-	{Titolo: "Il Signore delle Mosche", Autore: "", Anno di Pubblicazione: "1954", Genere: "Romanzo", ISBN: "9780143124290", Descrizione: "Romanzo di William Golding che narra la storia di un gruppo di ragazzi."}	Il campo autore non può essere vuoto.	L'amministratore è reindirizzato all'interfaccia per aggiungere Libri.	PASS
8	Stringa Anno di	Titolo Valido Autore Valido Anno di	-	{Titolo: "Il Signore delle Mosche", Autore: "William Golding", Anno di Pubblicazione: "-154", Genere:	L'anno di pubblicazione deve essere un numero intero positivo.	L'amministratore è reindirizzato	PASS







