

UNIVERSITÀ POLITECNICA DELLE MARCHE

INGEGNERIA DEL SOFTWARE

---

# Software per la gestione della Biblioteca Universitaria

---

*Autori:*

Francesco FINIZII  
Kristian LIKAJ



27 febbraio 2023

# Indice

<b>1 DESCRIZIONE</b>	<b>4</b>
1.1 Accesso alla piattaforma . . . . .	4
1.2 Catalogo Biblioteca . . . . .	4
1.3 Prestito librario . . . . .	5
1.4 Prenotazione posto aula studio . . . . .	5
1.5 Glossario di progetto . . . . .	7
<b>2 ANALISI DEI REQUISITI</b>	<b>8</b>
2.1 Requisiti funzionali . . . . .	9
2.1.1 Gestione Catalogo . . . . .	9
2.1.2 Gestione Prestiti . . . . .	9
2.1.3 Gestione Prenotazioni . . . . .	10
2.1.4 Gestione Profilo . . . . .	10
2.1.5 Gestione Utente . . . . .	10
2.1.6 Gestione Sistema . . . . .	10
2.2 Requisiti non funzionali . . . . .	11
2.2.1 Generali . . . . .	11
2.2.2 Implementazione . . . . .	12
<b>3 CASI D'USO</b>	<b>13</b>
3.1 Gestione Catalogo . . . . .	15
3.1.1 AggiungiDocumento . . . . .	17
3.1.2 VisualizzaDocumento . . . . .	18
3.1.3 ModificaDocumento . . . . .	19
3.1.4 RimuoviDocumento . . . . .	20
3.1.5 ConsultazioneCatalogo . . . . .	21
3.1.6 RicercaDocumento . . . . .	22
3.2 Gestione Prenotazioni . . . . .	23

3.2.1	AggiungiPrenotazione . . . . .	25
3.2.2	VisualizzaPrenotazione . . . . .	26
3.2.3	RimuoviPrenotazione . . . . .	27
3.2.4	VisualizzaListaPrenotazioni . . . . .	28
3.2.5	RicercaTurniDisponibili . . . . .	29
3.2.6	RicercaPrenotazione . . . . .	30
3.3	Gestione Sistema . . . . .	31
3.3.1	Autenticazione . . . . .	33
3.3.2	Backup . . . . .	34
3.3.3	SospensioneAccount . . . . .	35
3.3.4	TerminaPrenotazioni . . . . .	36
3.4	Gestione Prestiti . . . . .	37
3.4.1	EffettuaPrestito . . . . .	39
3.4.2	VisualizzaPrestito . . . . .	40
3.4.3	ProrogaPrestito . . . . .	41
3.4.4	TerminaPrestito . . . . .	42
3.4.5	AttivazioneAccount . . . . .	43
3.4.6	RicercaPrestito . . . . .	44
3.4.7	VisualizzaListaPrestiti . . . . .	45
3.5	Gestione Utente . . . . .	46
3.5.1	VisualizzaUtente . . . . .	47
3.5.2	VisualizzaListaUtenti . . . . .	48
3.5.3	RicercaUtente . . . . .	49
3.6	Gestione Profilo . . . . .	50
3.6.1	VisualizzaAccount . . . . .	52
3.6.2	ModificaAccount . . . . .	53
3.6.3	EliminaAccount . . . . .	54
3.6.4	RegistrazioneAccount . . . . .	55
<b>4</b>	<b>MATRICE DI MAPPING</b>	<b>56</b>
<b>5</b>	<b>CLASSI DI ANALISI</b>	<b>58</b>
<b>6</b>	<b>CLASSI DI PROGETTAZIONE</b>	<b>60</b>
6.1	Model . . . . .	61
6.2	Controller . . . . .	62
6.3	View . . . . .	63
6.3.1	Generale . . . . .	64
6.3.2	Studente . . . . .	69

6.3.3	Amministratore . . . . .	79
<b>7</b>	<b>DIAGRAMMI DI SEQUENZA</b>	<b>98</b>
7.1	AggiungiDocumento . . . . .	99
7.2	ModificaDocumento . . . . .	100
7.3	RimuoviDocumento . . . . .	101
7.4	Autenticazione . . . . .	102
7.5	EffettuaPrestito . . . . .	103
7.6	VisualizzaListaPrenotazioni . . . . .	104
7.7	VisualizzaPrestito . . . . .	105
<b>8</b>	<b>DIAGRAMMI DI ATTIVITÀ</b>	<b>106</b>
8.1	Autenticazione . . . . .	107
8.2	VisualizzaListaPrenotazioni . . . . .	108
8.3	act-RicercaPrenotazione . . . . .	109
8.4	TerminaPrestito . . . . .	110
<b>9</b>	<b>DIAGRAMMI DELLE MACCHINE A STATI</b>	<b>111</b>
9.1	Prenotazione . . . . .	112
9.2	Prestito . . . . .	113
9.3	Studente . . . . .	114
<b>10</b>	<b>DIAGRAMMA DEI COMPONENTI</b>	<b>115</b>
<b>11</b>	<b>DIAGRAMMA DI DEPLOYMENT</b>	<b>116</b>
<b>12</b>	<b>UNIT TEST</b>	<b>117</b>
12.1	test_studente . . . . .	118
12.2	test_amministratore . . . . .	118
12.3	test_libro . . . . .	119
12.4	test_rivista . . . . .	120
12.5	test_prestito . . . . .	121
12.6	test_prenotazione . . . . .	122
<b>13</b>	<b>NOTE</b>	<b>123</b>

# **1. DESCRIZIONE**

Il progetto proposto consiste nella realizzazione di un software per la gestione del Sistema Bibliotecario dell'Università Politecnica delle Marche. Il software in questione si occupa di due principali servizi:

1. Prestito librario.
2. Prenotazione posto aula studio.

## **1.1 Accesso alla piattaforma**

Per poter usufruire dei servizi offerti dal software, gli studenti e il personale della biblioteca si possono registrare gratuitamente utilizzando la matricola fornita dall'Ateneo. Come da regolamento, per completare la registrazione, il personale della biblioteca deve fornire il codice di sicurezza fornito dal Centro di Documentazione di Ateneo (CAD). Una volta avuto accesso alla piattaforma è possibile visualizzare, modificare o eliminare il proprio profilo utente. La rimozione del profilo è concessa solamente agli studenti che non hanno prestiti in corso. Il personale addetto alla gestione della piattaforma può ricercare un utente presente nel sistema e visualizzarne i dettagli e l'attuale stato.

## **1.2 Catalogo Biblioteca**

Il Sistema Bibliotecario di Ateneo propone a tutti gli studenti un ricco patrimonio bibliografico. La piattaforma consente di consultare solamente l'archivio della Biblioteca Tecnico-Scientifica-Biomedica. Nel catalogo sono inclusi solamente libri e riviste in quanto unici documenti che possono essere concessi in prestito. Solo il personale addetto alla gestione della piattaforma può aggiungere, modificare ed eliminare un documento dal catalogo. A ogni utente è concesso verificare se un'opera è presente

nelle raccolte del Sistema bibliotecario e vederne la disponibilità e i dettagli. Inoltre la piattaforma permette di filtrare i risultati e visualizzarli nell'ordine desiderato.

### **1.3 Prestito librario**

Il prestito librario è concesso agli studenti dell'Ateneo per non più di 2 volumi contemporaneamente. Il prestito è personale e non può essere trasferito ad altri. La durata del prestito è di 30 giorni. È possibile prorogare tale scadenza per altri 7 giorni. Gli studenti iscritti possono fare richieste di prestito tramite la piattaforma. La richiesta viene presa in carico dal sistema e dal successivo giorno feriale è possibile recarsi in sede per ritirare il documento fornendo il codice identificativo del prestito al personale della biblioteca. Una volta restituito il documento il personale della biblioteca provvede a terminare il prestito dello studente tramite piattaforma. I prestiti devono essere restituiti entro i tempi previsti. Lo studente che non risulta in regola con le restituzioni viene disabilitato automaticamente dal servizio e non può effettuare nuovi prestiti, prenotazioni o proroghe. La riabilitazione al servizio di prestito avviene automaticamente con la restituzione del documento o, in caso di smarrimento e deterioramento, con l'acquisto di titoli sostitutivi che verranno indicati dal personale.

### **1.4 Prenotazione posto aula studio**

Le biblioteche del CAD sono aperte per i servizi in presenza dal lunedì al venerdì dalle 8:30 alle 19:00. Tramite piattaforma gli studenti iscritti possono prenotare il proprio posto in una delle aule dell'Ateneo. Le aule studio sono a disposizione nelle seguenti sedi:

- Biblioteca Tecnico-Scientifica-Biomedica (Montedago)
- Biblioteca Economico-Giuridica-Sociologica (Villarey)
- Biblioteca Medica (Torrette)

Ogni studente può effettuare massimo 1 prenotazione al giorno distante massimo 3 giorni feriali dal giorno corrente. Ogni prenotazione può durare massimo 2 ore e non è possibile selezionare diverse fasce orarie nell'arco della stessa giornata. Il posto non può essere scelto, ma viene assegnato automaticamente dal software il quale lo associa alla matricola dello studente in questione. Per consentire la massima disponibilità

delle postazioni, gli studenti sono tenuti a cancellare in anticipo le prenotazioni nel caso in cui non dovessero essere presenti. La cancellazione avviene esclusivamente tramite piattaforma.

## 1.5 Glossario di progetto

PAROLA	DEFINIZIONE	SINONIMI	OMONIMI
Autenticazione	Riconoscimento dell'Utente tramite matricola e Password.	Accesso	Nessuno
Registrazione	Creazione account da parte dell'utente per accedere al software.	Nessuno	Nessuno
Utente	Chi utilizza il software.	Nessuno	Nessuno
Studente	Persona iscritta all'Ateneo che utilizza i servizi offerti dal software.	Nessuno	Nessuno
Amministratore	Impiegato addetto all'amministrazione della biblioteca.	Nessuno	Nessuno
Bibliotecario/a	Addetto/a al servizio della biblioteca.	Nessuno	Nessuno
Assistente	Addetto che esegue i compiti affidati dal bibliotecario	Nessuno	Nessuno
Archivista	Addetto specializzato nella gestione dell'archivio bibliografico	Nessuno	Nessuno
Matricola	Codice identificativo fornito dall'Ateneo a studenti e personale universitario.	Nessuno	Nessuno
Password	Serie di caratteri alfanumerici che costituisce la parola d'ordine per accedere al sistema.	Nessuno	Nessuno
Ruolo	Compito svolto dal personale della biblioteca.	Nessuno	Nessuno
Codice sicurezza	Codice fornito dal CAD al personale addetto alla gestione del software.	Nessuno	Nessuno
Prestito	Cessione temporanea di un documento a uno studente da parte della biblioteca.	Nessuno	Nessuno
Prenotazione	Impegno a occupare e diritto ad aver riservato un posto studio in apposite strutture offerte dall'Ateneo.	Nessuno	Nessuno
Catalogo	Archivio ordinato e sistematico dei documenti prestabili dalla biblioteca.	Nessuno	Nessuno
Sospensione	Interruzione temporanea dell'accesso al software per lo studente.	Nessuno	Nessuno
Attivazione	Riabilitazione all'accesso al software per lo studente.	Nessuno	Nessuno
Turno	Periodo di tempo assegnato alla prenotazione effettuata tramite software.	Fascia oraria	Nessuno
Backup	Copia di sicurezza di tutti i dati archiviati nel sistema.	Nessuno	Nessuno

## 2. ANALISI DEI REQUISITI

I requisiti del sistema sono suddivisi in *requisiti funzionali* e *requisiti non funzionali* come illustrato in figura 2.1:

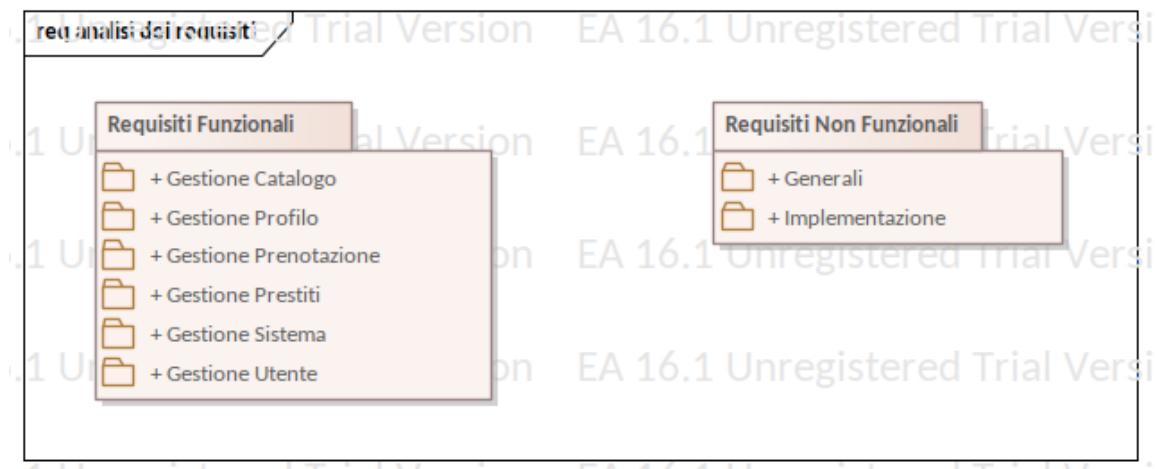


Figura 2.1: Analisi dei requisiti

## 2.1 Requisiti funzionali

I requisiti funzionali descrivono il comportamento del software. Sono stati suddivisi in 6 macro-categorie ognuna rappresentata da un package contenente i corrispondenti requisiti come mostrato in figura 2.3

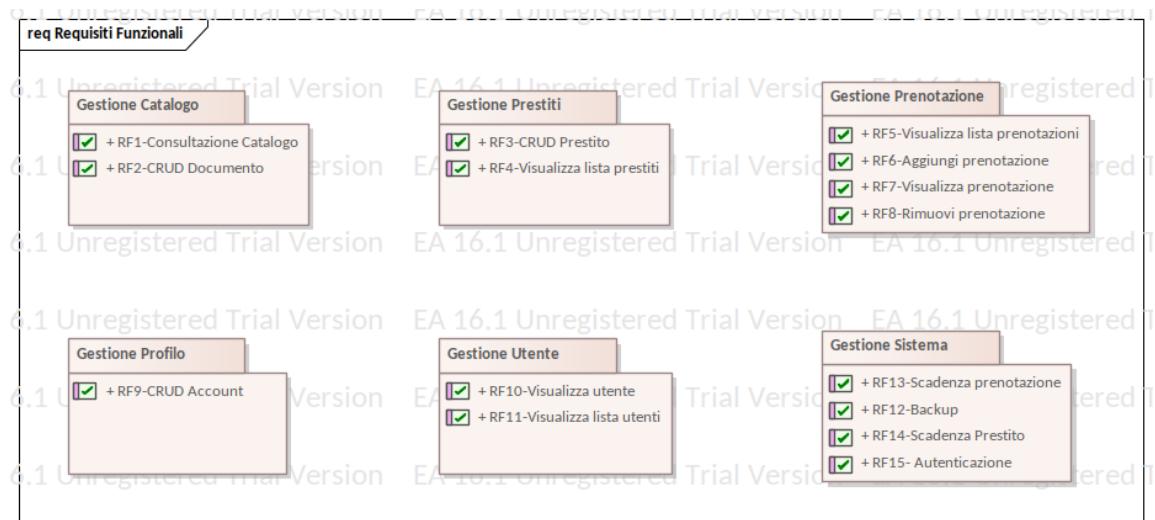


Figura 2.2: Requisiti Funzionali

### 2.1.1 Gestione Catalogo

- **RF1-Consultazione Catalogo:** Il sistema dovrà garantire la consultazione dei titoli disponibili al prestito librario e dei titoli presenti nell'archivio della biblioteca.
- **RF2-CRUD Documento:** Il sistema dovrà gestire le attività CRUD sui documenti.

### 2.1.2 Gestione Prestiti

- **RF3-CRUD Prestito:** Il sistema dovrà gestire le attività CRUD sui prestiti.
- **RF4-Visualizza lista prestiti:** Il sistema dovrà gestire la visualizzazione dei prestiti attualmente in corso.

### 2.1.3 Gestione Prenotazioni

- **RF5-Visualizza lista prenotazioni:** Il sistema dovrà gestire la visualizzazione delle prenotazioni presenti nelle relative sedi dell'Ateneo.
- **RF6-Aggiungi prenotazione:** Il sistema dovrà permettere agli studenti di prenotare un postazione finalizzata allo studio nelle apposite sedi dell'Ateneo.
- **RF7-Visualizza prenotazione:** Il sistema dovrà permettere agli studenti di visualizzare la propria prenotazione.
- **RF8-Rimuovi prenotazione:** Il sistema dovrà permettere agli studenti di cancellare la propria prenotazione.

### 2.1.4 Gestione Profilo

- **RF9-CRUD Account:** Il sistema dovrà gestire le attività CRUD sui profili utenti.

### 2.1.5 Gestione Utente

- **RF10-Visualizza utente:** Il sistema dovrà permettere di visualizzare i dettagli relativi a un utente.
- **RF11-Visualizza lista utenti:** Il sistema dovrà permettere al personale della biblioteca di visualizzare gli utenti registrati.

### 2.1.6 Gestione Sistema

- **RF12-Backup :** Il sistema dovrà effettuare un backup periodico dei dati.
- **RF13-Scadenza prenotazione:** Il sistema dovrà gestire il termine delle prenotazioni.
- **RF14-Scadenza Prestito:** Il sistema dovrà gestire la scadenza dei prestiti.
- **RF15- Autenticazione:** Il sistema dovrà consentire solamente agli utenti registrati di accedere alla piattaforma.

## 2.2 Requisiti non funzionali

I requisiti non funzionali rappresentano i vincoli imposti al sistema. Sono stati suddivisi in 2 macro-categorie ognuna rappresentata da un package contenente i corrispondenti requisiti come mostrato in figura 2.3

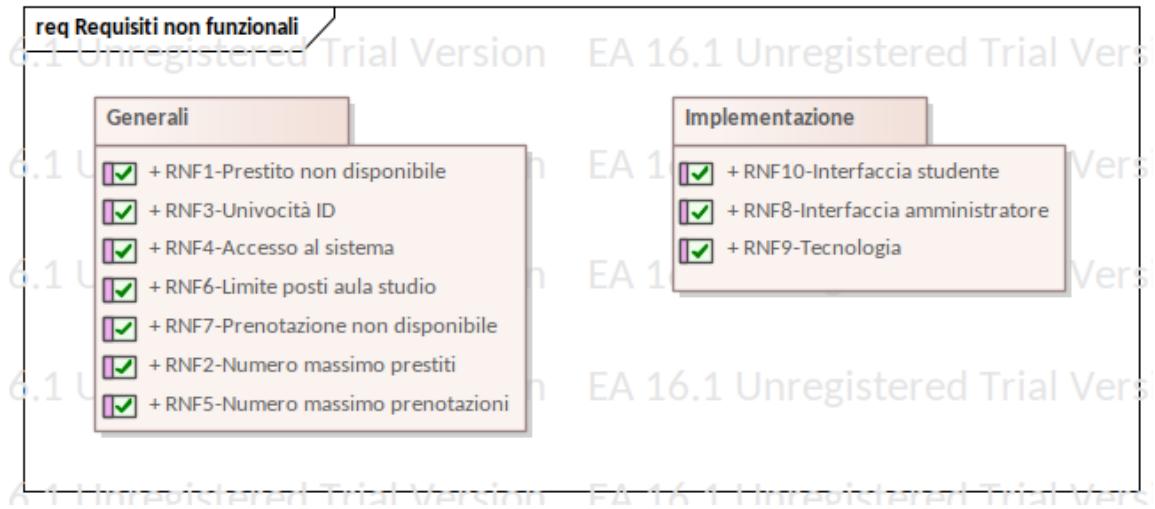


Figura 2.3: Requisiti Funzionali

### 2.2.1 Generali

- **RNF1-Prestito non disponibile:** Il sistema non dovrà permettere di effettuare il prestito di un documento non disponibile.
- **RNF2-Numero massimo prestiti:** Il sistema non dovrà permettere che uno studente effettui più di due prestiti alla volta.
- **RNF3-Univocità ID:** Il sistema dovrà utilizzare dei codici identificativi univoci: Gli studenti, gli amministratori e le prenotazioni saranno identificati dalla matricola fornita dall’Ateneo. I libri saranno identificati dal proprio ISBN, mentre le riviste saranno identificate dal proprio ISSN.
- **RNF4-Accesso al sistema:** Il sistema dovrà permettere all’utente l’accesso alla piattaforma tramite l’utilizzo di un codice identificativo (matricola) e una password alfanumerica compresa tra gli 8 e i 32 caratteri.

- **RNF5-Numero massimo prenotazioni:** Il sistema non dovrà permettere che uno studente effettui più di una prenotazione alla volta.
- **RNF6-Limite posti aula studio:** Il sistema prenotazioni dovrà tener conto della capienza massima delle aule adibite allo studio.
- **RNF7-Prenotazione non disponibile:** Il sistema non dovrà permettere di effettuare la prenotazione di un turno non disponibile.

### 2.2.2 Implementazione

- **RNF8-Interfaccia amministratore:** : Il sistema dovrà essere dotato di un'interfaccia grafica comune a tutto il personale della biblioteca.
- **RNF9-Tecnologia:** Il sistema dovrà essere implementato utilizzando Python 3.10
- **RNF10-Interfaccia studente::** Il sistema dovrà essere dotato di un'interfaccia grafica comune a tutti gli studenti.

### 3. CASI D'USO

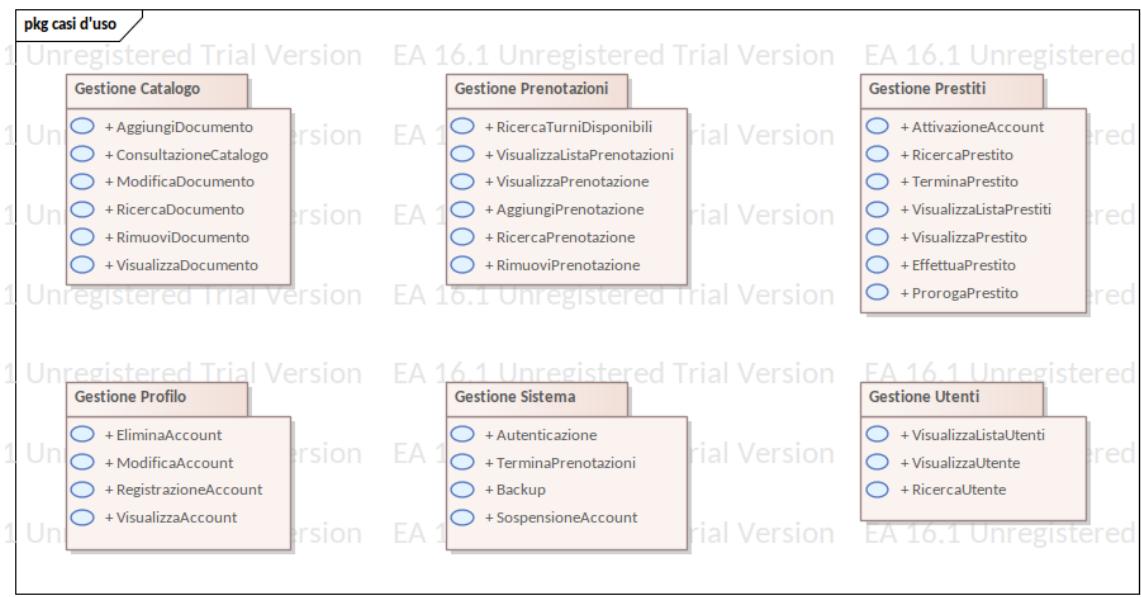


Figura 3.1: Casi d'uso

I casi d'uso del sistema sono stati suddivisi in 6 macro-categorie:

- Gestione Catalogo
- Gestione Prenotazioni
- Gestione Prestiti
- Gestione Profilo
- Gestione Sistema

- Gestione Utenti

Tale suddivisione è identica a quella adottata nel paragrafo 2.1, ognuna concretizzata in un diagramma dei casi d'uso, come rappresentato in figura 3.1

Gli attori che interagiscono con il sistema sono rappresentati in figura 3.2

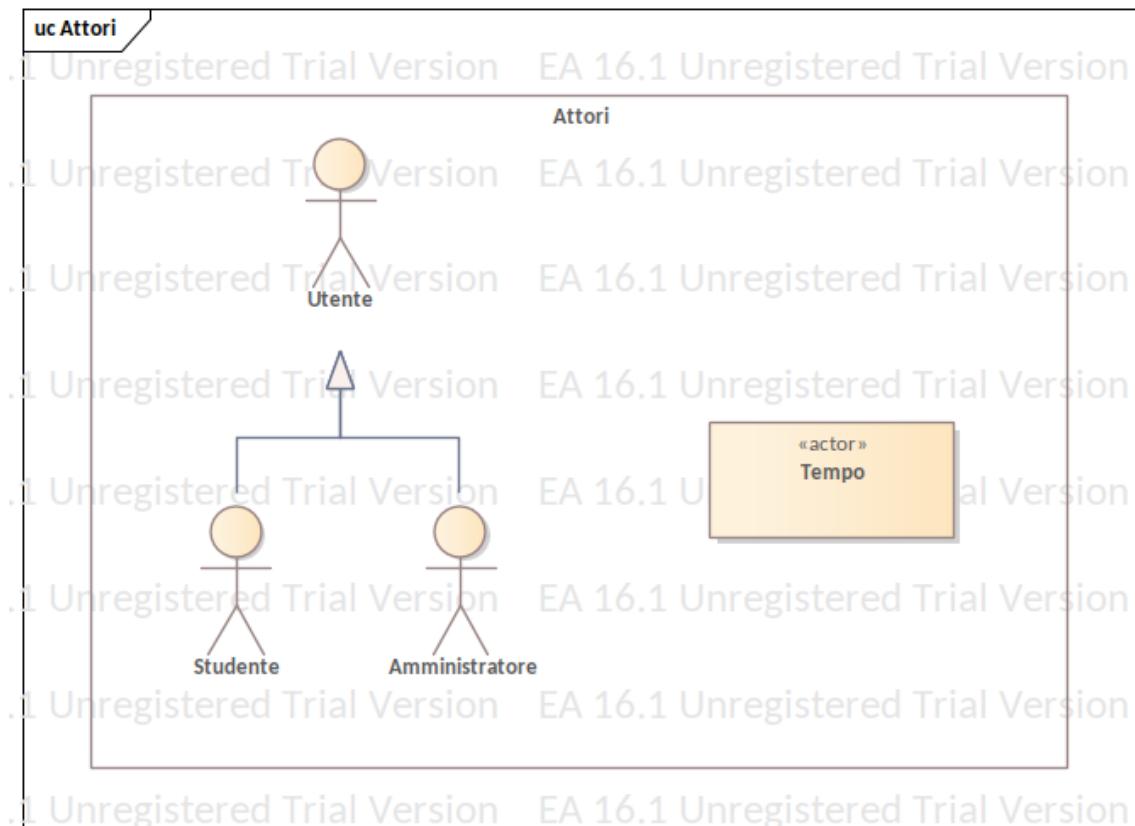


Figura 3.2: Casi d'uso

### 3.1 Gestione Catalogo

Il diagramma dei casi d'uso mostrato in figura 3.3 rappresenta la gestione dei documenti all'interno del sistema.

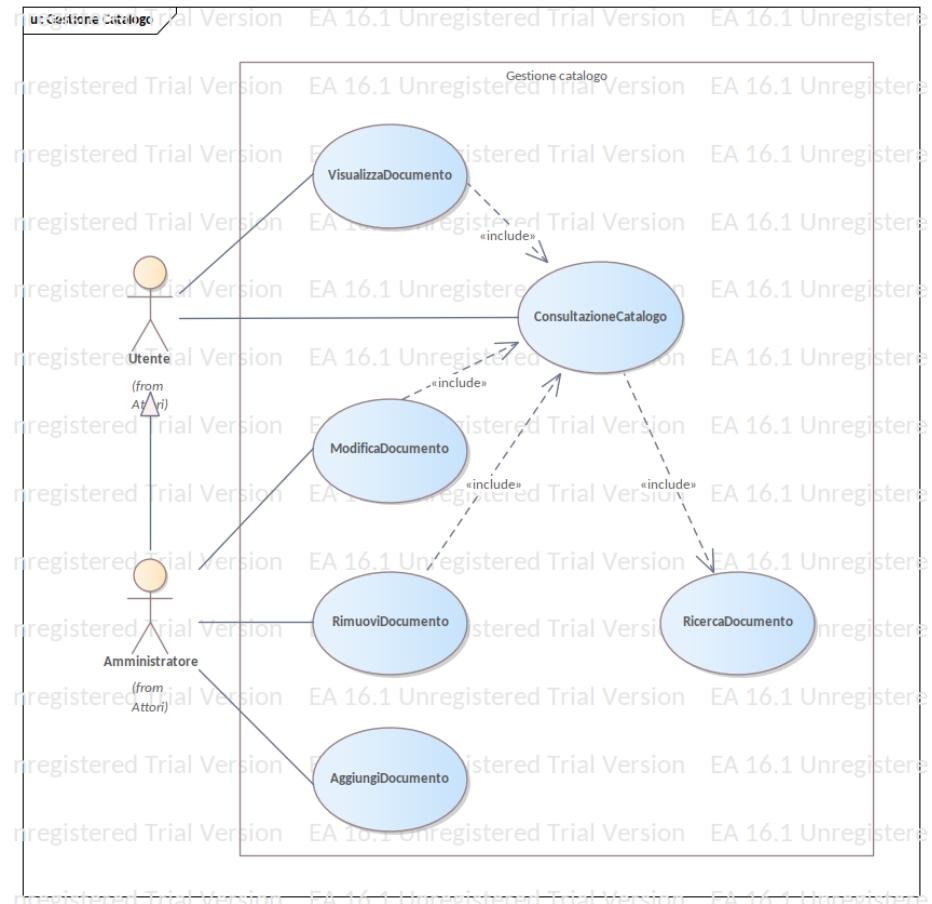


Figura 3.3: Gestione Catalogo

I casi d'uso presenti in questo package sono:

- AggiungiDocumento
- VisualizzaDocumento
- ModificaDocumento
- RimuoviDocumento

- RicercaDocumento
- ConsultazioneCatalogo

Gli attori che interagiscono con questi casi d'uso sono l'*Utente* l'*Amministratore*.

### **3.1.1 AggiungiDocumento**

Questo caso d'uso consente di aggiungere un nuovo documento al catalogo della biblioteca.

**Attori primari:** Amministratore.

**Attori secondari:** Nessuno.

**Precondizioni:** L'attore primario è stato autenticato dal sistema.

**Postcondizioni:** Il documento è stato aggiunto al database della biblioteca.

**Sequenza degli eventi principale:**

1. Il caso d'uso ha inizio quando l'attore primario vuole aggiungere un nuovo documento all'interno della biblioteca;
2. Il sistema preleva i parametri relativi al documento.
3. *if* i dati inseriti sono validi:
  - 3.1 *if* il codice identificativo del documento da aggiungere non è già associato a un altro documento:
    - 3.1.1 il sistema memorizza il nuovo documento.
  - 3.2 *else:*
    - 3.2.1 il sistema mostra un opportuno messaggio di errore;
4. *else:*
  - 4.1 Il sistema mostra un opportuno messaggio di errore;

**Sequenza degli eventi alternativa:** Nessuna.

### **3.1.2 VisualizzaDocumento**

Questo caso d'uso consente di visualizzare le informazioni relative ad un documento.

**Attori primari:** Utente.

**Attori secondari:** Nessuno.

**Precondizioni:** L'attore primario è stato autenticato dal sistema.

**Postcondizioni:** Il sistema mostra i dettagli relativi al documento.

**Sequenza degli eventi principale:**

1. Il caso d'uso ha inizio quando l'attore primario vuole visualizzare i dati relativi ad un documento;
2. *include (ConsultazioneCatalogo)*
3. L'attore primario seleziona e conferma il documento da visualizzare dal catalogo;
4. Il sistema preleva la chiave di ricerca del documento;
5. Il sistema recupera il documento dal database della biblioteca;
6. il sistema visualizza i dati relativi al documento.

**Sequenza degli eventi alternativa:** Nessuna.

### **3.1.3 ModificaDocumento**

Questo caso d'uso consente di modificare le informazioni relative ad un documento

**Attori primari:** Amministratore.

**Attori secondari:** Nessuno.

**Precondizioni:** L'attore primario è stato autenticato dal sistema.

**Postcondizioni:** Il sistema salva le modifiche apportate al documento.

**Sequenza degli eventi principale:**

1. Il caso d'uso ha inizio quando l'attore primario vuole modificare i dati relativi ad un documento.
2. *include (ConsultazioneCatalogo)*
3. L'attore primario seleziona e conferma il documento da modificare dal catalogo;
4. Il sistema preleva la chiave di ricerca del documento;
5. Il sistema recupera il documento dal database della biblioteca e lo restituisce;
6. L'attore primario modifica i dati relativi al documento.
7. *if* i dati inseriti sono validi:
  - 7.1 il sistema memorizza le modifiche effettuate del documento.
8. *else:*
  - 8.1 Il sistema mostra un opportuno messaggio di errore;

**Sequenza degli eventi alternativa:** Nessuna.

### **3.1.4 RimuoviDocumento**

Questo caso d'uso consente di rimuovere un documento all'interno del sistema.

**Attori primari:** Amministratore.

**Attori secondari:** Nessuno.

**Precondizioni:** L'attore primario è stato autenticato dal sistema.

**Postcondizioni:** Il documento è stato rimosso dal database della biblioteca.

**Sequenza degli eventi principale:**

1. Il caso d'uso ha inizio quando l'attore primario vuole rimuovere un documento all'interno del sistema;
2. *include (ConsultazioneCatalogo)*
3. L'attore primario seleziona e conferma il documento da rimuovere dal catalogo;
4. Il sistema chiede all'attore primario se desidera rimuovere il documento;
5. *if* l'attore primario conferma l'operazione:
  - 5.1 Il sistema preleva la chiave di ricerca del documento;
  - 5.2 Il sistema provvede alla rimozione del documento dal database della biblioteca.

**Sequenza degli eventi alternativa:** Nessuna.

### **3.1.5 ConsultazioneCatalogo**

Questo caso d'uso consente di visualizzare i documenti presenti all'interno della biblioteca che soddisfano i parametri di ricerca.

**Attori primari:** Utente.

**Attori secondari:** Nessuno.

**Precondizioni:** L'attore primario è stato autenticato dal sistema.

**Postcondizioni:** Il sistema mostra la lista dei documenti che soddisfano i parametri di ricerca.

**Sequenza degli eventi principale:**

1. Il caso d'uso ha inizio quando l'attore primario vuole consultare il catalogo documenti della biblioteca;
2. *include (RicercaDocumento)*
3. il sistema mostra i risultati di ricerca.

**Sequenza degli eventi alternativa:** Nessuna.

### **3.1.6 RicercaDocumento**

Questo caso d'uso consente di ricercare i documenti presenti nel database della biblioteca.

**Attori primari:** Utente

**Attori secondari:** Nessuno.

**Precondizioni:** L'attore primario è stato autenticato dal sistema.

**Postcondizioni:** Il sistema restituisce i documenti che soddisfano i parametri di ricerca.

**Sequenza degli eventi principale:**

1. Il caso d'uso ha inizio quando l'attore primario vuole ricercare un documento nel catalogo della biblioteca;
2. L'attore primario specifica i criteri di ricerca;
3. Il sistema ricerca i documenti che soddisfano i parametri specificati;
4. *for each* documento trovato:
  - 4.1 Il sistema aggiunge il documento ai risultati di ricerca;
5. Il sistema restituisce i risultati di ricerca.

**Sequenza degli eventi alternativa:** Nessuna.

## 3.2 Gestione Prenotazioni

Il diagramma dei casi d'uso mostrato in figura 3.4 rappresenta la gestione delle prenotazioni all'interno del sistema.

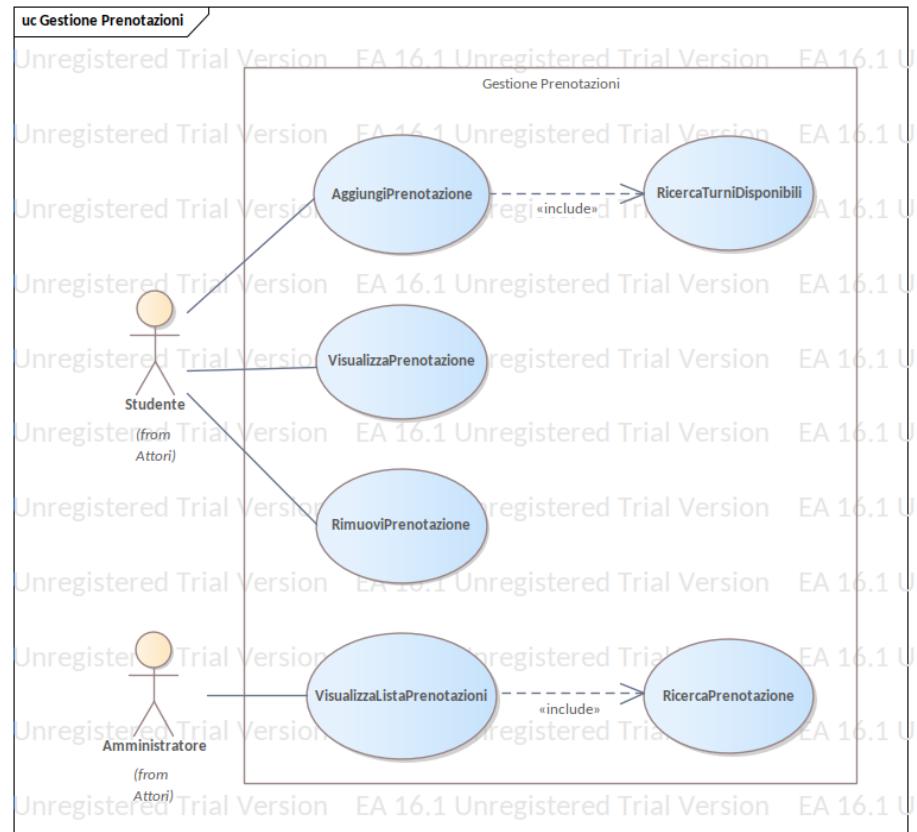


Figura 3.4: Gestione prenotazioni

I casi d'uso presenti in questo package sono:

- AggiungiPrenotazione
- VisualizzaPrenotazione
- RimuoviPrenotazione
- VisualizzaListaPrenotazioni
- RicercaPrenotazione

- RicercaTurniDisponibili

Gli attori che interagiscono con questi casi d'uso sono lo *Studente* e l'*Amministratore*.

### 3.2.1 AggiungiPrenotazione

Questo caso d'uso consente di aggiungere una nuova prenotazione.

**Attori primari:** Studente.

**Attori secondari:** Nessuno.

**Precondizioni:**

1. L'attore primario è stato autenticato dal sistema.
2. Nel momento della prenotazione almeno un posto aula studio è disponibile.

**Postcondizioni:** La prenotazione è stata aggiunta.

**Sequenza degli eventi principale:**

1. Il caso d'uso inizia quando l'attore primario vuole aggiungere una nuova prenotazione;
2. *include (RicercaTurniDisponibili)*
3. L'attore primario seleziona e conferma la fascia oraria in cui intende prenotarsi;
4. *if* la fascia oraria contiene massimo 4 turni *and* comprende solo turni consecutivi:
  - 4.1 *if* l'attore primario possiede già una prenotazione:
    - 4.1.1 Il sistema mostra un messaggio in cui chiede all'attore primario se desidera sostituire la prenotazione;
  - 4.2 *else:*
    - 4.2.1 Il sistema aggiunge la prenotazione e mostra un messaggio di conferma.
5. *else:*
  - 5.1 Il sistema visualizza un opportuno messaggio di errore;

**Sequenza degli eventi alternativa:** Nessuna.

### **3.2.2 VisualizzaPrenotazione**

Questo caso d'uso consente di visualizzare le informazioni relative ad una prenotazione.

**Attori primari:** Studente.

**Attori secondari:** Nessuno.

**Postcondizioni:** Il sistema mostra i dati relativi alla prenotazione.

**Precondizioni:**

1. L'attore primario è stato autenticato dal sistema;
2. L'attore primario possiede una prenotazione.

**Sequenza degli eventi principale:**

1. Il caso d'uso ha inizio quando l'attore primario vuole visualizzare i dati relativi ad una prenotazione;
2. Il sistema preleva la chiave di ricerca della prenotazione;
3. Il sistema recupera la prenotazione dal database della biblioteca;
4. Il sistema visualizza i dati relativi alla prenotazione.

**Sequenza degli eventi alternativa:** Nessuna.

### **3.2.3 RimuoviPrenotazione**

Questo caso d'uso consente di rimuovere una prenotazione.

**Attori primari:** Studente.

**Attori secondari:** Nessuno.

**Precondizioni:**

1. L'attore primario è stato autenticato dal sistema.
2. L'attore primario possiede una prenotazione.

**Postcondizioni:** La prenotazione è stata rimossa.

**Sequenza degli eventi principale:**

1. Il caso d'uso ha inizio quando lo studente vuole rimuovere una prenotazione;
2. Il sistema mostra un messaggio in cui chiede di confermare la rimozione della prenotazione;
3. *if* l'attore primario conferma l'operazione:
  - 3.1 Il sistema preleva la chiave di ricerca della prenotazione
  - 3.2 Il sistema provvede alla rimozione della prenotazione dal database della biblioteca.

**Sequenza degli eventi alternativa:** Nessuna.

### **3.2.4 VisualizzaListaPrenotazioni**

Questo caso d'uso consente di visualizzare le prenotazioni presenti all'interno della biblioteca che soddisfano i parametri di ricerca.

**Attori primari:** Amministratore.

**Attori secondari:** Nessuno.

**Precondizioni:** L'attore primario è stato autenticato dal sistema.

**Postcondizioni:** Il sistema visualizza la lista delle prenotazioni che soddisfano i parametri di ricerca.

**Sequenza degli eventi principale:**

1. Il caso d'uso ha inizio quando l'attore primario vuole visualizzare le prenotazioni presenti nel sistema della biblioteca;
2. *include (RicercaPrenotazione)*
3. Il sistema mostra i risultati di ricerca.

**Sequenza degli eventi alternativa:** Nessuna.

### **3.2.5 RicercaTurniDisponibili**

Questo caso d'uso restituisce tutti le fasce orarie in cui è possibile prenotarsi in base alla sede e data specificata.

**Attori primari:** Studente

**Attori secondari:** Nessuno.

**Precondizioni:** Nessuna.

**Postcondizioni:** Il sistema restituisce i turni disponibili che soddisfano i parametri di ricerca.

**Sequenza degli eventi principale:**

1. Il caso d'uso ha inizio quando uno studente vuole effettuare una prenotazione;
2. L'attore primario specifica i parametri di ricerca;
3. il sistema ricerca i turni disponibili che soddisfano i criteri specificati;
4. *for each* turno trovato:
  - 4.1 Il sistema aggiunge il turno ai risultati di ricerca;
5. Il sistema restituisce e mostra i risultati di ricerca;

**Sequenza degli eventi alternativa:** Nessuna.

### **3.2.6 RicercaPrenotazione**

Questo caso d'uso consente di ricercare le prenotazioni presenti nel database della biblioteca.

**Attori primari:** Amministratore.

**Attori secondari:** Nessuno.

**Precondizioni:** L'attore primario è stato autenticato dal sistema.

**Postcondizioni:** Il sistema restituisce le prenotazioni che soddisfano i parametri di ricerca.

**Sequenza degli eventi principale:**

1. Il caso d'uso ha inizio quando l'attore primario vuole ricercare una prenotazione nel database della biblioteca;
2. L'attore primario specifica i criteri di ricerca;
3. il sistema ricerca le prenotazioni che soddisfano i parametri specificati;
4. *for each* prenotazione trovata:
  - 4.1 Il sistema aggiunge la prenotazione ai risultati di ricerca;
5. Il sistema restituisce i risultati di ricerca;

**Sequenza degli eventi alternativa:** Nessuna.

### 3.3 Gestione Sistema

Il diagramma dei casi d'uso mostrato in figura 3.5 rappresenta la gestione generale del sistema bibliotecario.

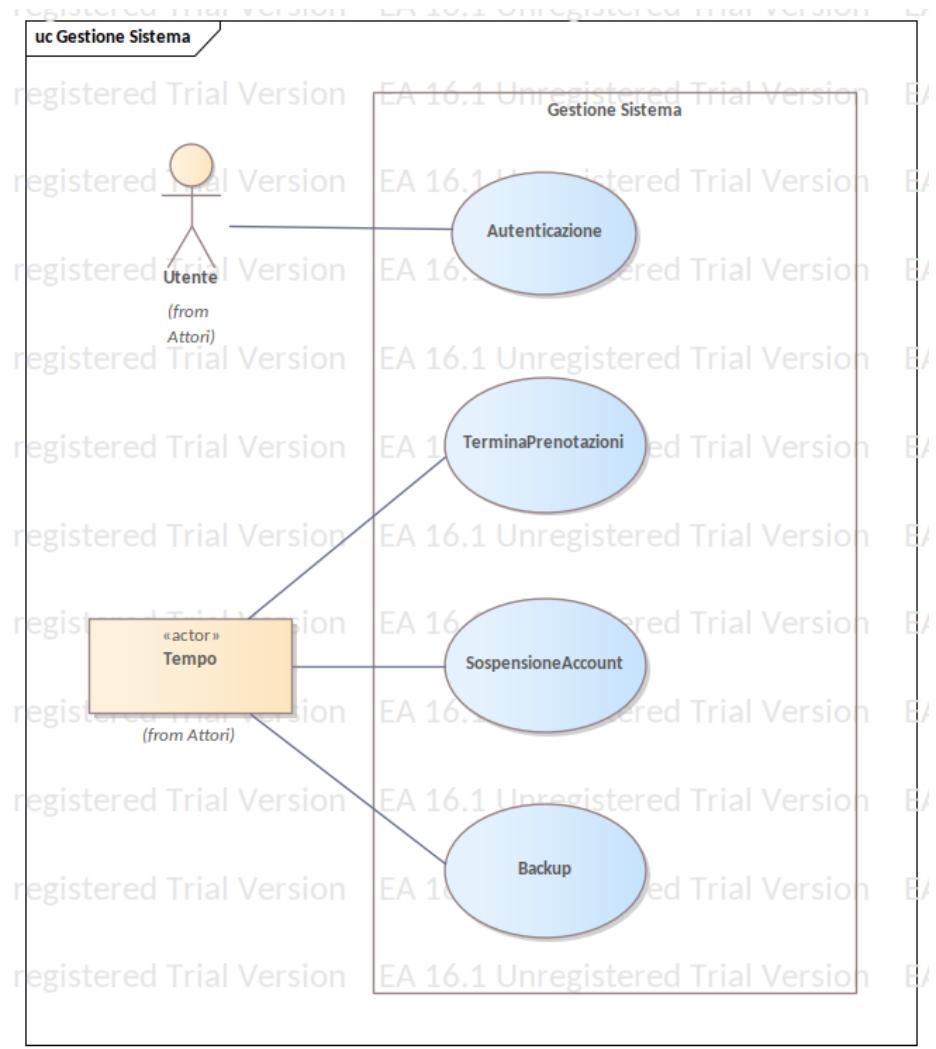


Figura 3.5: Gestione Sistema

I casi d'uso presenti in questo package sono:

- Autenticazione
- Backup

- SospensioneAccount
- TerminaPrenotazioni

Gli attori che interagiscono con questi casi d'uso sono l'*Utente* e il *Tempo*.

### 3.3.1 Autenticazione

Questo caso d'uso consente l'accesso al sistema da parte dell'Utente.

**Attori primari:** Utente.

**Attori secondari:** Nessuno.

**Precondizioni:** Nessuna.

**Postcondizioni:** L'attore primario è stato autenticato dal sistema.

**Sequenza degli eventi principale:**

1. Il caso d'uso inizia quando l'utente vuole accedere al sistema;
2. Il sistema preleva i dati inseriti dall'utente;
3. *if* le credenziali inserite sono valide:
  - 3.1 *if* le credenziali inserite sono associate ad un account *and* l'account non è sospeso *and* la chiave d'accesso è corretta:
    - 3.1.1 il sistema permette all'utente di accedere al sistema.
  - 3.2 *else:*
    - 3.2.1 il sistema visualizza un opportuno messaggio di errore;
4. *else:*
  - 4.1 il sistema visualizza un opportuno messaggio di errore;

**Sequenza degli eventi alternativa:** Nessuna.

### **3.3.2 Backup**

Questo caso d'uso gestisce il backup dei dati relativi al sistema informativo di riferimento.

**Attori primari:** Tempo.

**Attori secondari:** Nessuno.

**Precondizioni:** sono passate 24h dall'ultimo backup.

**Postcondizioni:** i dati sono stati copiati.

**Sequenza degli eventi principale:**

1. il caso d'uso inizia quando l'attore primario richiede il backup dei dati;
2. il sistema preleva tutti i dati presenti nel sistema;
3. il sistema copia sul disco i dati prelevati;

**Sequenza degli eventi alternativa:** Nessuna.

### **3.3.3 SospensioneAccount**

Questo caso d'uso gestisce la sospensione dell'account studente quando quest'ultimo non conclude un prestito entro i termini stabiliti.

**Attori primari:** Tempo.

**Attori secondari:** Nessuno.

**Precondizioni:** Nessuna.

**Postcondizioni:** Lo studente è stato sospeso dalla piattaforma.

**Sequenza degli eventi principale:**

1. il caso d'uso inizia ogni giorno alle ore 00.00;
2. il sistema preleva tutti i prestiti presenti nel database della biblioteca;
3. *for each* prestito:
  - 3.1 *if* la data di scadenza del prestito è passata:
    - 3.1.1 Il sistema sospende l'account studente a cui è associato il prestito.

**Sequenza degli eventi alternativa:** Nessuna.

### **3.3.4 TerminaPrenotazioni**

Questo caso d'uso consente di gestire la scadenza di una prenotazione.

**Attori primari:** Tempo.

**Attori secondari:** Nessuno.

**Precondizioni:** Nessuna.

**Postcondizioni:** Le prenotazioni scadute sono state rimosse dal database della biblioteca.

**Sequenza degli eventi principale:**

1. il caso d'uso inizia ogni giorno alle ore 19.30;
2. il sistema preleva tutte le prenotazioni presenti nel database della biblioteca;
3. *for each* prenotazione:
  - 3.1 *if* la data della prenotazione corrisponde al giorno corrente *and* l'ora di andata è passata:
    - 3.1.1 Il sistema rimuove la prenotazione dal database della biblioteca.

**Sequenza degli eventi alternativa:** Nessuna.

## 3.4 Gestione Prestiti

Il diagramma dei casi d'uso mostrato in figura 3.6 rappresenta la gestione dei prestiti all'interno del sistema bibliotecario.

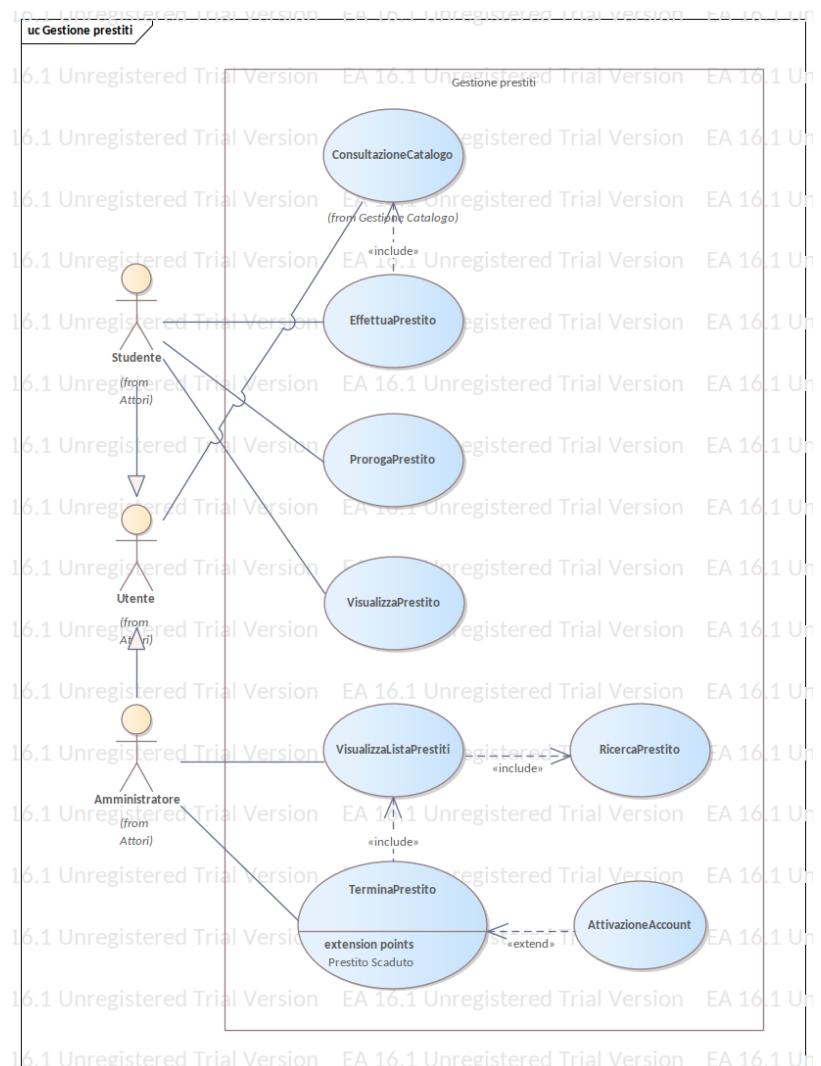


Figura 3.6: Gestione Prestiti

I casi d'uso presenti in questo package sono:

- EffettuaPrestito

- VisualizzaPrestito
- ProrogaPrestito
- TerminaPrestito
- AttivazioneAccount
- VisualizzaListaPrestiti
- RicercaPrestito

Gli attori che interagiscono con questi casi d'uso sono l'*Amministratore* e lo *Studente*.

### 3.4.1 EffettuaPrestito

Questo caso d'uso consente di effettuare il prestito di un documento presente all'interno della biblioteca.

**Attori primari:** Studente

**Attori secondari:** Nessuno

**Precondizioni:** L'attore primario è stato autenticato dal sistema.

**Postcondizioni:** Il prestito è stato effettuato.

**Sequenza degli eventi principale:**

1. Il caso d'uso ha inizio quando l'attore primario vuole effettuare il prestito di un documento presente all'interno del catalogo della biblioteca;
2. *include (ConsultazioneCatalogo)*
3. L'attore primario seleziona e conferma il documento del quale vuole effettuare il prestito;
4. Il sistema preleva il codice identificativo associato all'attore primario;
5. *if* il numero di prestiti associati all'attore primario è minore di 2 *and* il documento è disponibile:
  - 5.1 il sistema preleva il codice identificativo associato al documento selezionato;
  - 5.2 *if* il documento selezionato non è stato già preso in prestito:
    - 5.2.1 il sistema effettua il prestito del documento selezionato.
  - 5.3 *else:*
    - 5.3.1 il sistema visualizza un messaggio di errore;
6. *else:*
  - 6.1 il sistema mostra un opportuno messaggio di errore;

**Sequenza degli eventi alternativa:** Nessuna.

### **3.4.2 VisualizzaPrestito**

Questo caso d'uso consente di visualizzare le informazioni relative ad un prestito.

**Attori primari:** Studente

**Attori secondari:** Nessuno

**Precondizioni:**

1. L'attore primario è stato autenticato dal sistema.
2. L'attore primario possiede un prestito.

**Postcondizioni:** Il sistema visualizza il prestito.

**Sequenza degli eventi principale:**

1. Il caso d'uso ha inizio quando l'attore primario vuole visualizzare i dati relativi ad un prestito;
2. Il sistema preleva la chiave di ricerca del prestito;
3. Il sistema recupera il prestito dal database della biblioteca;
4. Il sistema visualizza i dati relativi al prestito.

**Sequenza degli eventi alternativa:** Nessuna.

### **3.4.3 ProrogaPrestito**

Questo caso d'uso consente di prolungare la scadenza di un prestito;

**Attori primari:** Studente.

**Attori secondari:** Nessuno.

**Precondizioni:**

1. L'attore primario è stato autenticato dal sistema.
2. L'attore primario possiede un prestito che non è stato precedentemente prorogato.

**Postcondizioni:** La proroga del prestito è stata effettuata.

**Sequenza degli eventi principale:**

1. Il caso d'uso ha inizio quando l'attore primario vuole prolungare la scadenza di un prestito;
2. Il sistema preleva la chiave di ricerca del prestito;
3. Il sistema recupera il prestito dal database della biblioteca e ne prolunga di una settimana la scadenza.

**Sequenza degli eventi alternativa:** Nessuna.

### **3.4.4 TerminaPrestito**

Questo caso d'uso consiste nella riconsegna del documento da parte dello studente al personale della biblioteca e nella conseguente conclusione del prestito.

**Attori primari:** Amministratore.

**Attori secondari:** Nessuno.

**Precondizioni:** L'attore primario è stato autenticato dal sistema.

**Postcondizioni:** Il prestito è terminato.

**Sequenza degli eventi principale:**

1. Il caso d'uso ha inizio quando uno studente riconsegna un documento precedentemente preso in prestito al personale della biblioteca;
2. Lo studente fornisce all'attore primario il codice identificativo del prestito;
3. *include (VisualizzaListaPrestiti)*
4. L'attore primario seleziona e conferma il prestito da concludere;  
*punto di estensione:)* prestito scaduto
5. il sistema provvede a incrementare la disponibilità del documento precedentemente preso in prestito;
6. il sistema rimuove il prestito associato allo studente dal database della biblioteca;

**Sequenza degli eventi alternativa:** Nessuna

### **3.4.5 AttivazioneAccount**

**segmento 1:** Il sistema provvede alla riattivazione dell'account associato allo studente precedentemente sospeso a causa della mancata restituzione del prestito entro i termini stabiliti.

**Attori primari:** Amministratore

**Attori secondari:** Nessuno

**Precondizioni del segmento 1:** Il prestito è scaduto.

**Postcondizioni del segmento 1:** L'account associato allo studente è stato riattivato.

**Sequenza degli eventi del segmento 1:**

1. il sistema preleva la chiave di ricerca dello studente;
2. Il sistema recupera il profilo studente dal database della biblioteca e lo riabilita a poter utilizzare la piattaforma.

**Sequenza degli eventi alternativa:** Nessuna

### **3.4.6 RicercaPrestito**

Questo caso d'uso consente di ricercare un prestito all'interno del database della biblioteca.

**Attori primari:** Amministratore.

**Attori secondari:** Nessuno.

**Precondizioni:** L'attore primario è stato autenticato dal sistema.

**Postcondizioni:** Il sistema restituisce i prestiti che soddisfano i parametri di ricerca.

**Sequenza degli eventi principale:**

1. Il caso d'uso ha inizio quando l'attore primario vuole ricercare un prestito all'interno del sistema della biblioteca;
2. L'attore primario specifica i criteri di ricerca;
3. il sistema ricerca i prestiti che soddisfano i parametri specificati;
4. *for each* prestito trovato:
  - 4.1 Il sistema aggiunge il prestito ai risultati di ricerca;
5. Il sistema restituisce i risultati di ricerca

**Sequenza degli eventi alternativa:** Nessuna.

### **3.4.7 VisualizzaListaPrestiti**

Questo caso d'uso consente di visualizzare i prestiti presenti all'interno della biblioteca che soddisfano i parametri specificati.

**Attori primari:** Amministratore.

**Attori secondari:** Nessuno.

**Precondizioni:** L'attore primario è stato autenticato dal sistema.

**Postcondizioni:** Il sistema mostra tutti i prestiti presenti all'interno del sistema che soddisfano i parametri richiesti.

**Sequenza degli eventi principale:**

1. Il caso d'uso ha inizio quando l'attore primario vuole visualizzare i prestiti presenti all'interno del sistema;
2. *include (RicercaPrestito)*
3. il sistema mostra i risultati di ricerca.

**Sequenza degli eventi alternativa:** Nessuna.

## 3.5 Gestione Utente

Il diagramma dei casi d'uso mostrato in figura 3.7 rappresenta la gestione degli utenti del sistema bibliotecario.

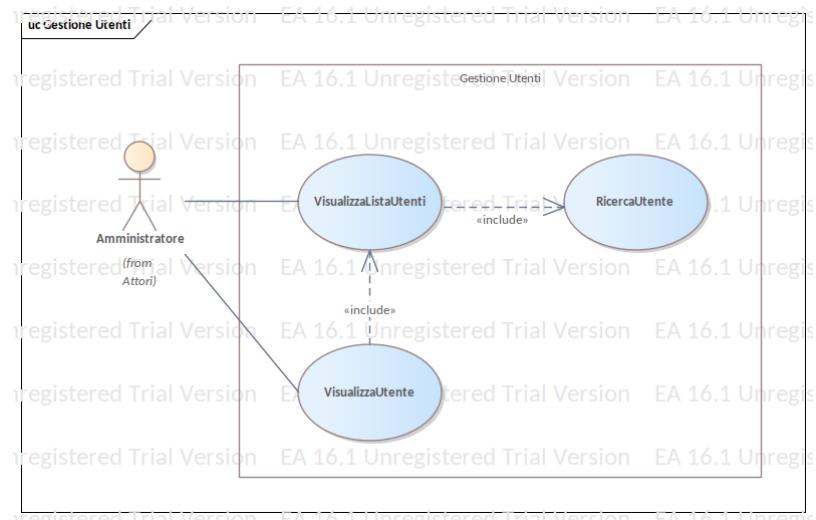


Figura 3.7: Gestione Utenti

I casi d'uso presenti in questo package sono:

- VisualizzaUtente
- VisualizzaListaUtenti
- RicercaUtente

L'attore che interagisce con questi casi d'uso è l'*Amministratore*.

### **3.5.1 VisualizzaUtente**

Questo caso d'uso consente di visualizzare le informazioni relative ad un utente.

**Attori primari:** Amministratore.

**Attori secondari:** Nessuno.

**Precondizioni:** L'attore primario è stato autenticato dal sistema.

**Postcondizioni:** Il sistema mostra i dati relativi all'utente.

**Sequenza degli eventi principale:**

1. Il caso d'uso ha inizio quando l'attore primario vuole visualizzare i dati relativi ad un utente;
2. *include (VisualizzaListaUtenti)*
3. L'attore primario seleziona l'utente desiderato dalla lista dei risultati di ricerca;
4. il sistema preleva dal database della biblioteca le informazioni associate all'utente selezionato.
5. il sistema mostra i dettagli relativi all'utente.

**Sequenza degli eventi alternativa:** Nessuna .

### **3.5.2 VisualizzaListaUtenti**

Questo caso d'uso consente di visualizzare gli utenti presenti nel database della biblioteca che soddisfano i parametri di ricerca.

**Attori primari:** Amministratore.

**Attori secondari:** Nessuno.

**Precondizioni:** L'attore primario è stato autenticato dal sistema.

**Postcondizioni:** Il sistema visualizza la lista di utenti che soddisfano i criteri di ricerca.

**Sequenza degli eventi principale:**

1. Il caso d'uso ha inizio quando l'attore primario vuole visualizzare gli utenti presenti nel sistema della biblioteca;
2. *include (RicercaUtente)*
3. Il sistema mostra i risultati di ricerca.

**Sequenza degli eventi alternativa:** Nessuna .

### **3.5.3 RicercaUtente**

Questo caso d'uso consente di ricercare un utente all'interno del database della biblioteca.

**Attori primari:** Amministratore.

**Attori secondari:** Nessuno.

**Precondizioni:** L'attore primario è stato autenticato dal sistema.

**Postcondizioni:** Il sistema restituisce gli utenti che soddisfano i parametri di ricerca.

#### **Sequenza degli eventi principale**

1. Il caso d'uso ha inizio quando l'attore primario vuole ricercare un utente all'interno del sistema della biblioteca;
2. L'attore primario specifica i parametri di ricerca;
3. Il sistema ricerca nel database della biblioteca gli utenti che soddisfano i criteri di ricerca specificati;
4. *for each* utente trovato:
  - 4.1 Il sistema aggiunge l'utente ai risultati di ricerca;
5. Il sistema restituisce i risultati di ricerca.

**Sequenza degli eventi alternativa:** Nessuna.

## 3.6 Gestione Profilo

Il diagramma dei casi d'uso mostrato in figura 3.8 rappresenta la gestione degli utenti del sistema bibliotecario.

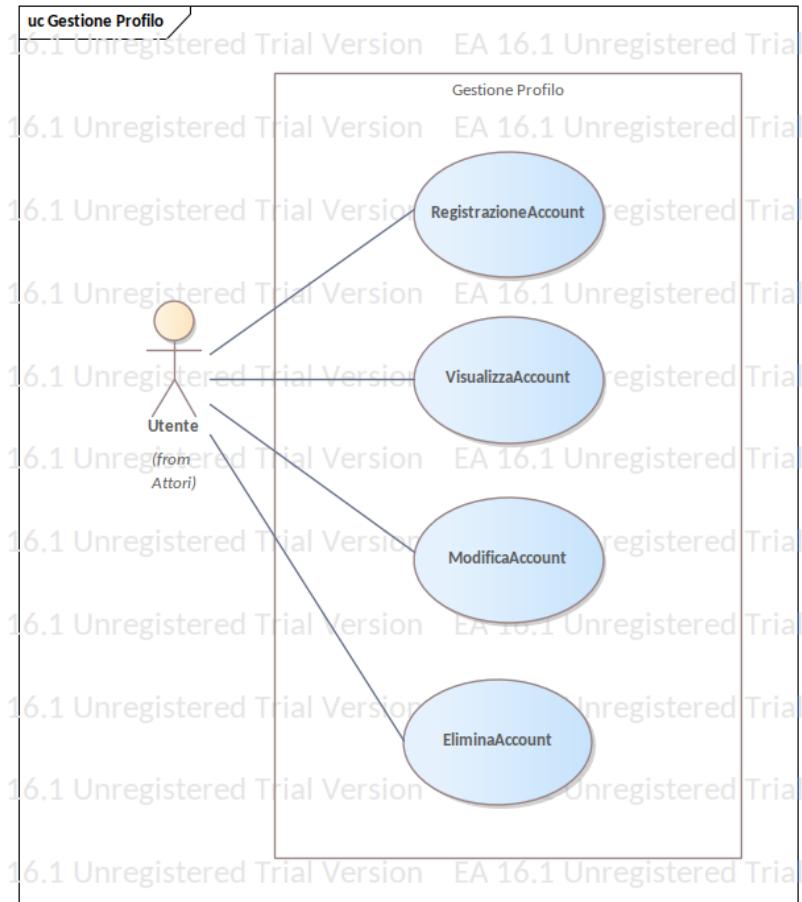


Figura 3.8: Gestione Profilo

I casi d'uso presenti in questo package sono:

- RegistrazioneAccount
- VisualizzaAccount
- ModificaAccount
- EliminaAccount

L'attore che interagisce con questo casi d'uso è l'*Utente*.

### **3.6.1 VisualizzaAccount**

Questo caso d'uso consente di visualizzare le informazioni relative al profilo utente della biblioteca.

**Attori primari:** Utente.

**Attori secondari:** Nessuno.

**Precondizioni:** L'attore primario è stato autenticato dal sistema.

**Postcondizioni:** Il sistema mostra i dettagli relativi al proprio account.

**Sequenza degli eventi principale:**

1. Il caso d'uso ha inizio quando l'attore primario vuole visualizzare i dettagli relativi al proprio account;
2. Il sistema recupera dal database della biblioteca le informazioni relative all'utente;
3. il sistema mostra i dettagli del profilo utente.

**Sequenza degli eventi alternativa:** Nessuna.

### **3.6.2 ModificaAccount**

Questo caso d'uso consente di modificare il profilo utente della biblioteca.

**Attori primari:** Utente.

**Attori secondari:** Nessuno.

**Precondizioni:** L'attore primario è stato autenticato dal sistema.

**Postcondizioni:** Il sistema salve le modifiche apportate all'account.

**Sequenza degli eventi principale:**

1. Il caso d'uso ha inizio quando l'attore primario vuole modificare i dettagli relativi al proprio account;
2. Il sistema recupera dal database della biblioteca le informazioni relative all'utente;
3. L'attore primario modifica e conferma i dati relativi al proprio account.
4. *if* i dati inseriti sono validi:
  - 4.1 il sistema memorizza le modifiche effettuate del proprio account.
5. *else:*
  - 5.1 Il sistema mostra un opportuno messaggio di errore;

**Sequenza degli eventi alternativa:** Nessuna.

### **3.6.3 EliminaAccount**

Questo caso d'uso consente di eliminare il profilo utente dal sistema della biblioteca.

**Attori primari:** Utente.

**Attori secondari:** Nessuno.

**Precondizioni:** L'attore primario è stato autenticato dal sistema.

**Postcondizioni:** Il profilo utente è stato rimosso dal database della biblioteca.

**Sequenza degli eventi principale:**

1. Il caso d'uso ha inizio quando l'attore primario vuole eliminare il proprio profilo utente;
2. Il sistema chiede all'attore primario di confermare l'eliminazione del proprio account;
3. *if* l'attore primario conferma l'operazione:
  - 3.1 *if* l'attore primario è uno studente *and* al suo profilo è associato almeno un prestito:
    - 3.1.1 Il sistema mostra un opportuno messaggio di errore.
  - 3.2 *else*:
    - 3.2.1 Il sistema provvede alla rimozione dell'utente dal database della biblioteca.

**Sequenza degli eventi alternativa:** Nessuna.

### **3.6.4 RegistrazioneAccount**

Questo caso d'uso consente all'Utente di registrarsi nella piattaforma.

**Attori primari:** Utente

**Attori secondari:** Nessuno

**Precondizioni:** Nessuna

**Postcondizioni:**

1. L'account utente è stato creato.
2. L'attore primario è stato autenticato dal sistema.

**Sequenza degli eventi principale:**

1. Il caso d'uso inizia quando l'utente vuole creare un nuovo account;
2. Il sistema preleva i dati inseriti dall'utente;
3. *if* i dati inseriti sono validi:
  - 3.1 *if* l'utente non dispone già di un account:
    - 3.1.1 Il sistema procede alla creazione del nuovo account;
    - 3.1.2 il sistema permette all'utente di accedere al sistema;
  - 3.2 *else:*
    - 3.2.1 il sistema visualizza un opportuno messaggio di errore;
4. *else:*
  - 4.1 Il sistema mostra un opportuno messaggio di errore;

**Sequenza degli eventi alternativa:** Nessuna.

## 4. MATRICE DI MAPPING

In figura 4.1 viene illustrata la *Matrice di Mapping* la quale garantisce che ogni requisito funzionali proposto dal software venga soddisfatto da almeno un caso d'uso.

Source	Target
:AggiungiDocumento	:RF1-Consultazione Catalogo
:AggiungiPrenotazione	:RF10-Visualizza utente
:AttivazioneAccount	:RF11-Visualizza lista utenti
:Autenticazione	:RF12-Backup
:Backup	:RF13-Scadenza prenotazione
:ConsultazioneCatalogo	:RF14-Scadenza Prestito
:EffettuaPrestito	:RF15-Autenticazione
:EliminaAccount	:RF2-CRUD Documento
:ModificaAccount	:RF3-CRUD Prestito
:ModificaDocumento	:RF4-Visualizza lista prestiti
:ProrogaPrestito	:RF5-Visualizza lista prenotazioni
:RegistrazioneAccount	:RF6-Agggiungi prenotazione
:RicercaDocumento	:RF7-Visualizza prenotazione
:RicercaPrenotazione	:RF8-Rimuovi prenotazione
:RicercaPrestito	:RF9-CRUD Account
:RicercaTurniDisponibili	
:RicercaUtente	
:RimuoviDocumento	
:RimuoviPrenotazione	
:SospensioneAccount	
:TerminaPrenotazioni	
:TerminaPrestito	
:VisualizzaAccount	
:VisualizzaDocumento	
:VisualizzaListaPrenotazioni	
:VisualizzaListaPrestiti	
:VisualizzaListaUtenti	
:VisualizzaPrenotazione	
:VisualizzaPrestito	
:VisualizzaUtente	

Figura 4.1: Matrice di mapping

## 5. CLASSI DI ANALISI

In figura 5.1 viene illustrato il *diagramma delle classi di analisi* che modella i concetti chiave del progetto. Si è sviluppato così un modello iniziale del software che è stato successivamente approfondito in fase di progettazione.

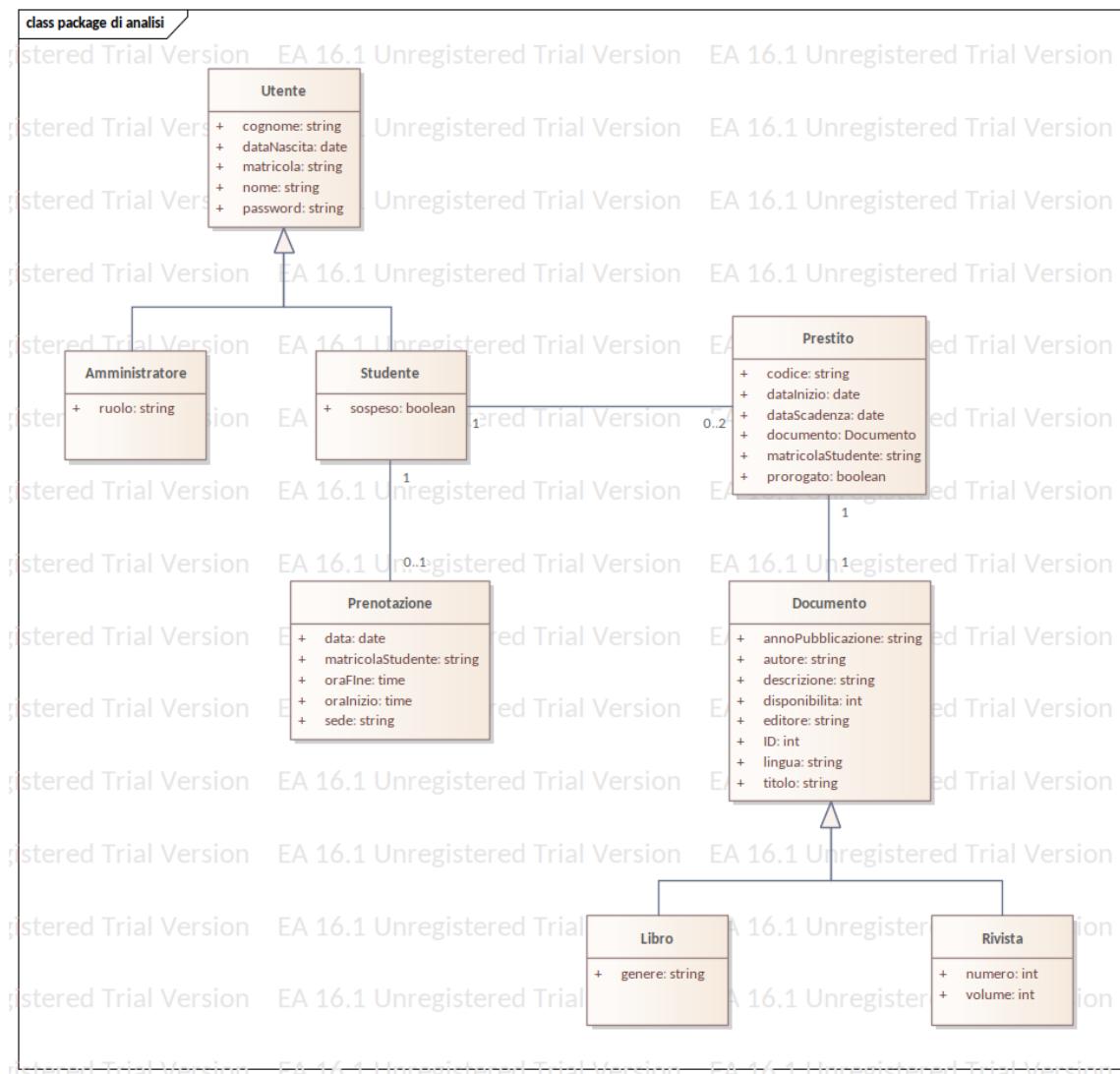


Figura 5.1: Classi di analisi

# 6. CLASSI DI PROGETTAZIONE

Partendo dal modello rappresentato dalle classi di analisi, nelle *classi di progettazione* si sono definite in modo completo le funzionalità implementate dal software. Si è utilizzato il pattern architetturale Model-View-Controller come illustrato in figura 6.1.

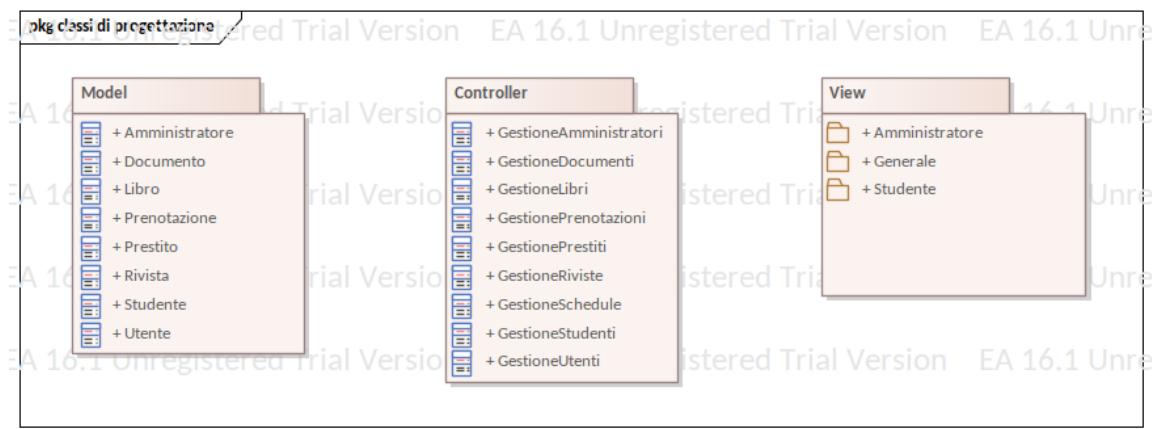


Figura 6.1: Package classi di progettazione

Le classi sono state suddivise nei seguenti package:

- Model
- Controller
- View:
  - Amministratore
  - Generale
  - Studente

## 6.1 Model

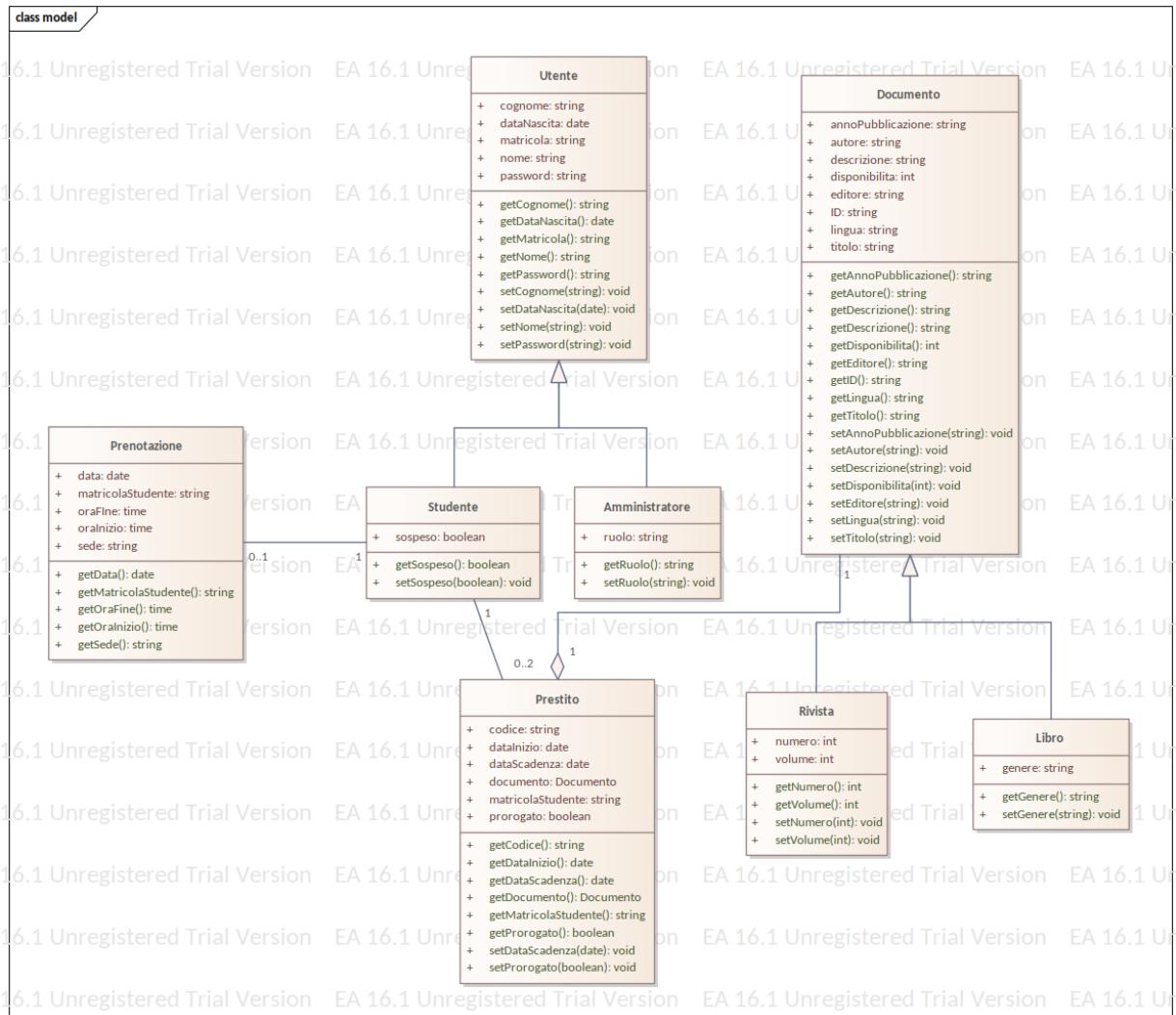


Figura 6.2: Model

## 6.2 Controller

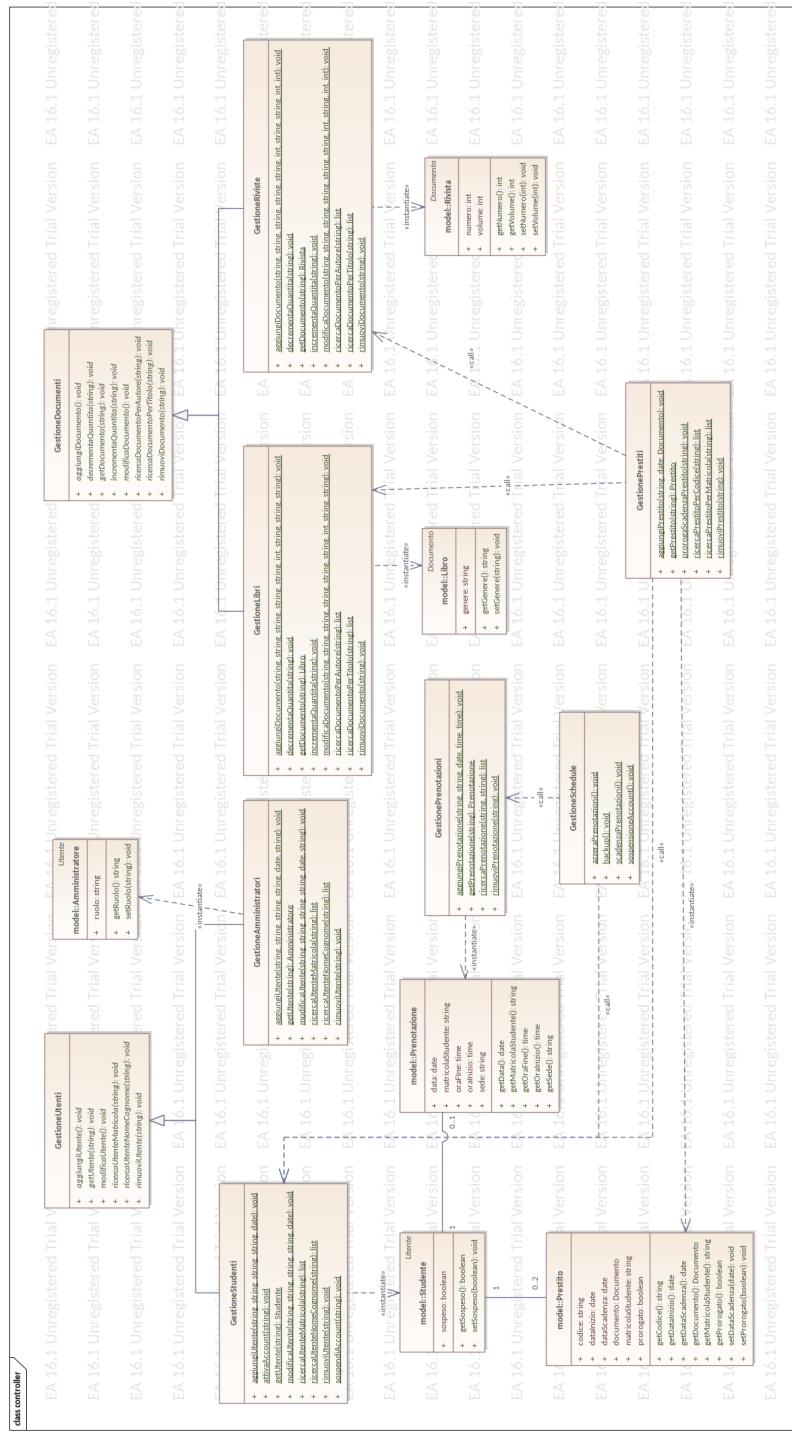


Figura 6.3: Controller

## 6.3 View

Le interfacce utente sono state realizzate utilizzando QtDesigner, tool offerto da PyQt5. Il package View è suddiviso in ulteriori packages come illustrato in 6.4:

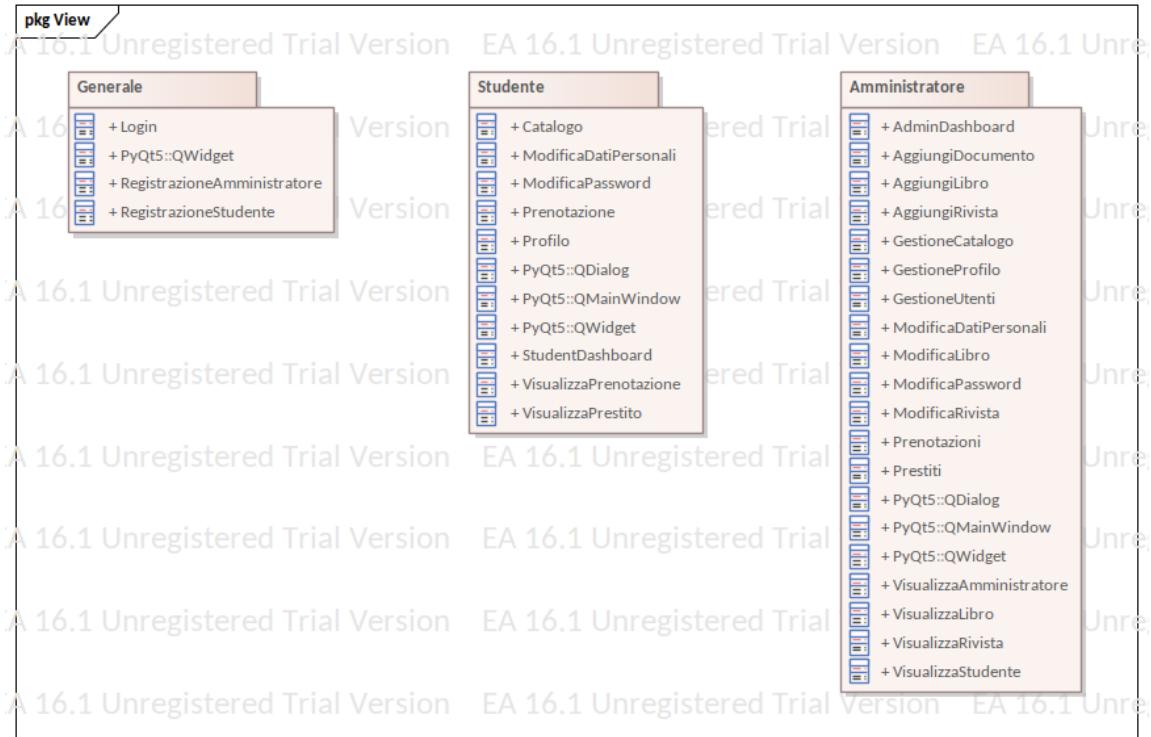


Figura 6.4: package View

Le classi sono state suddivise nei seguenti package:

- Amministratore
- Generale
- Studente

### 6.3.1 Generale

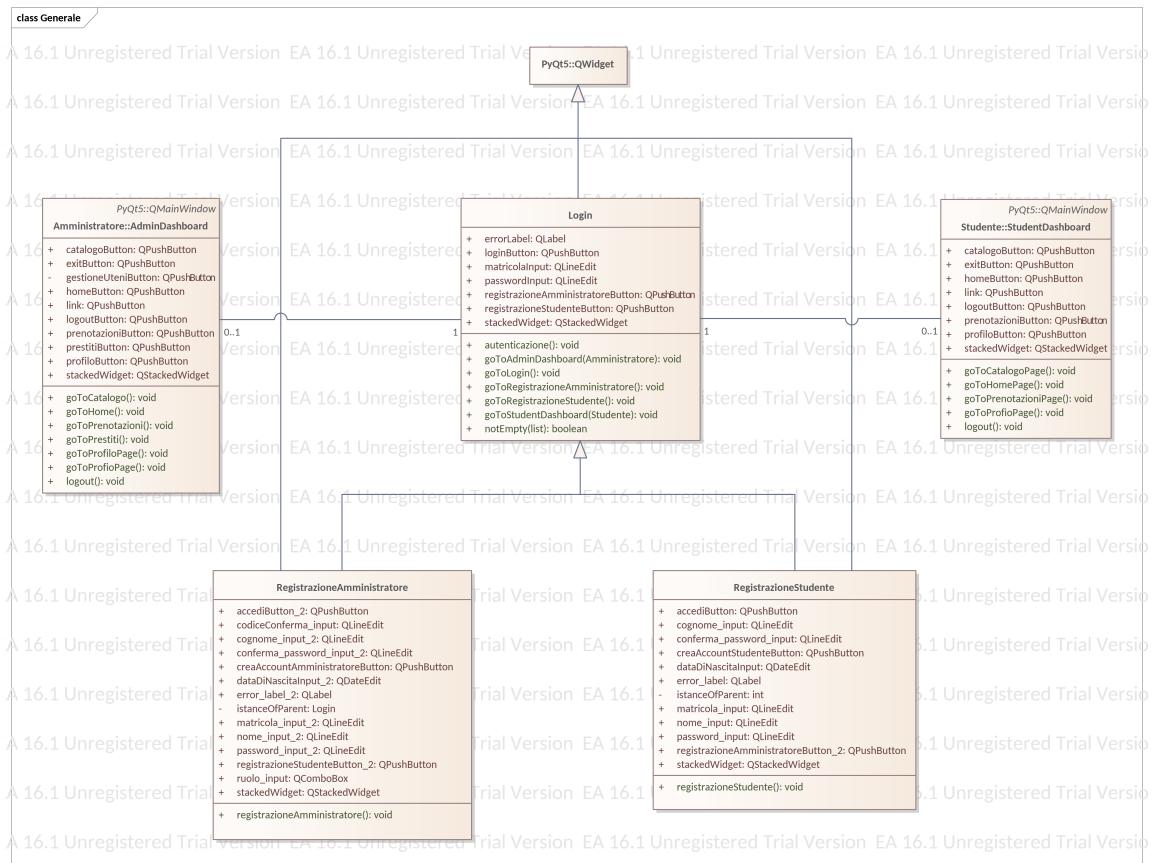


Figura 6.5: Generale

Come mostrato in 6.5 il package *Generale* include le seguenti classi ognuna delle quali importa un file .ui (User Interface):

- Login
- RegistrazioneStudente
- RegistrazioneAmministratore

## Login

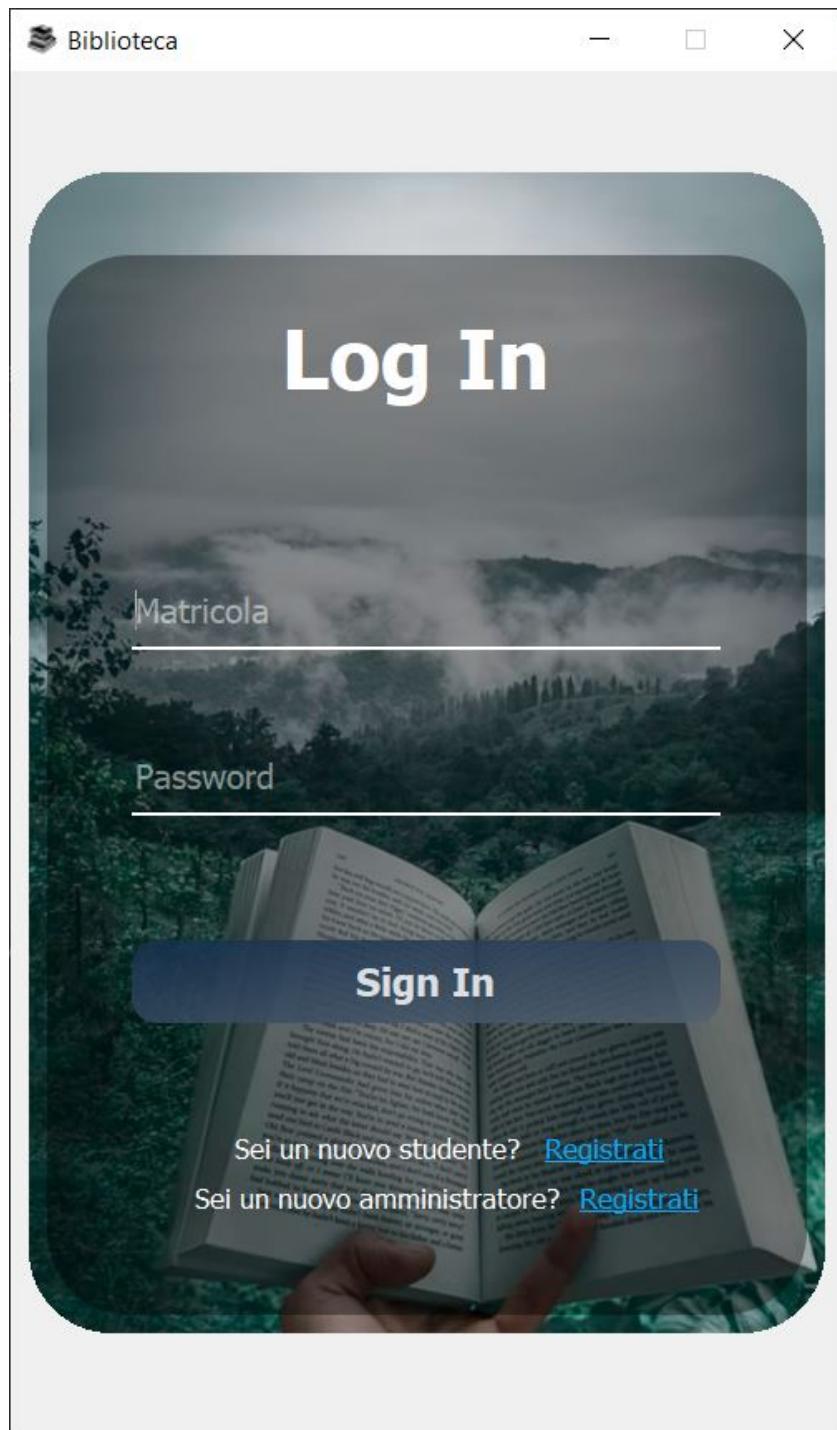


Figura 6.6: Login

## RegistrazioneStudente

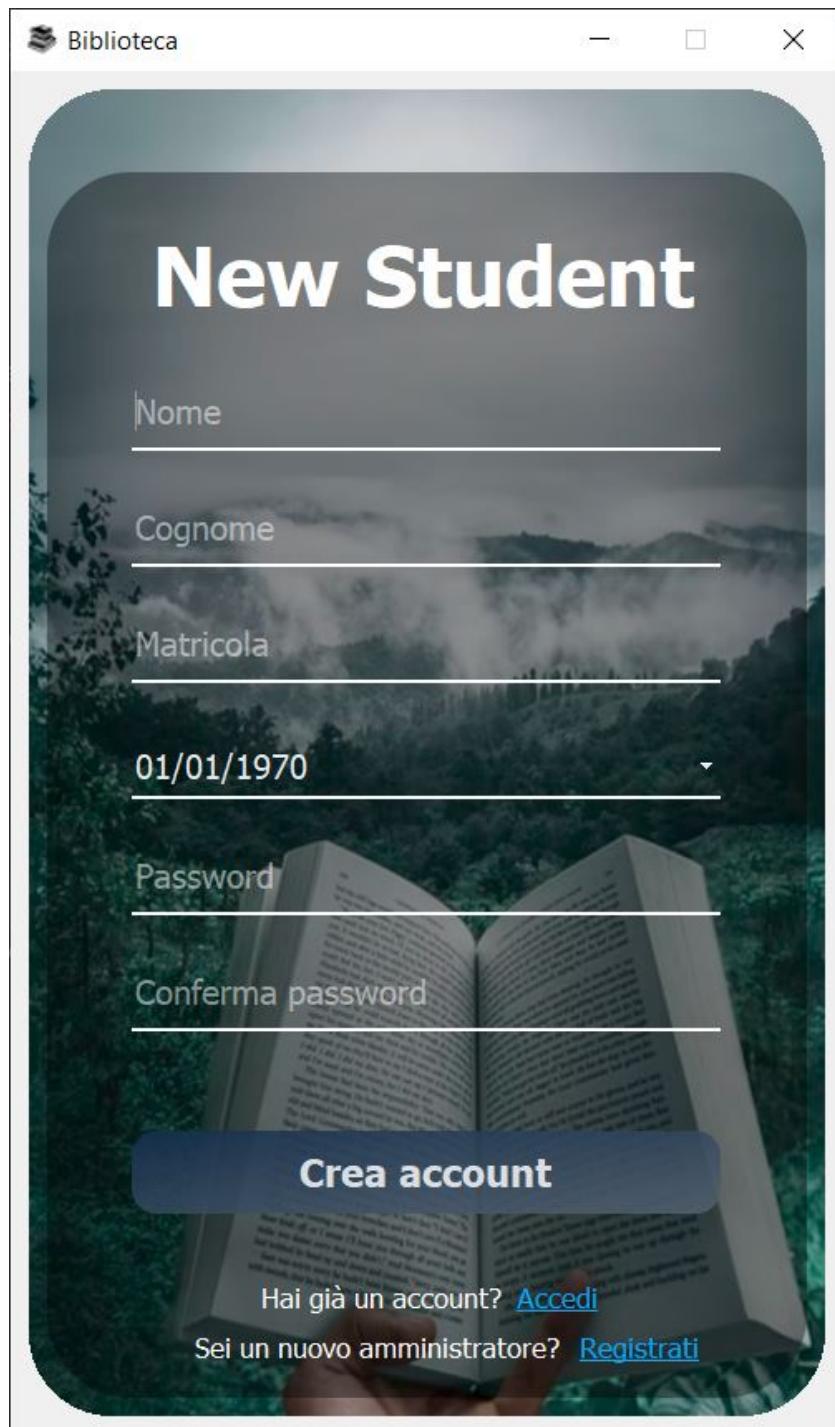


Figura 6.7: RegistrazioneStudente

## RegistrazioneAmministratore

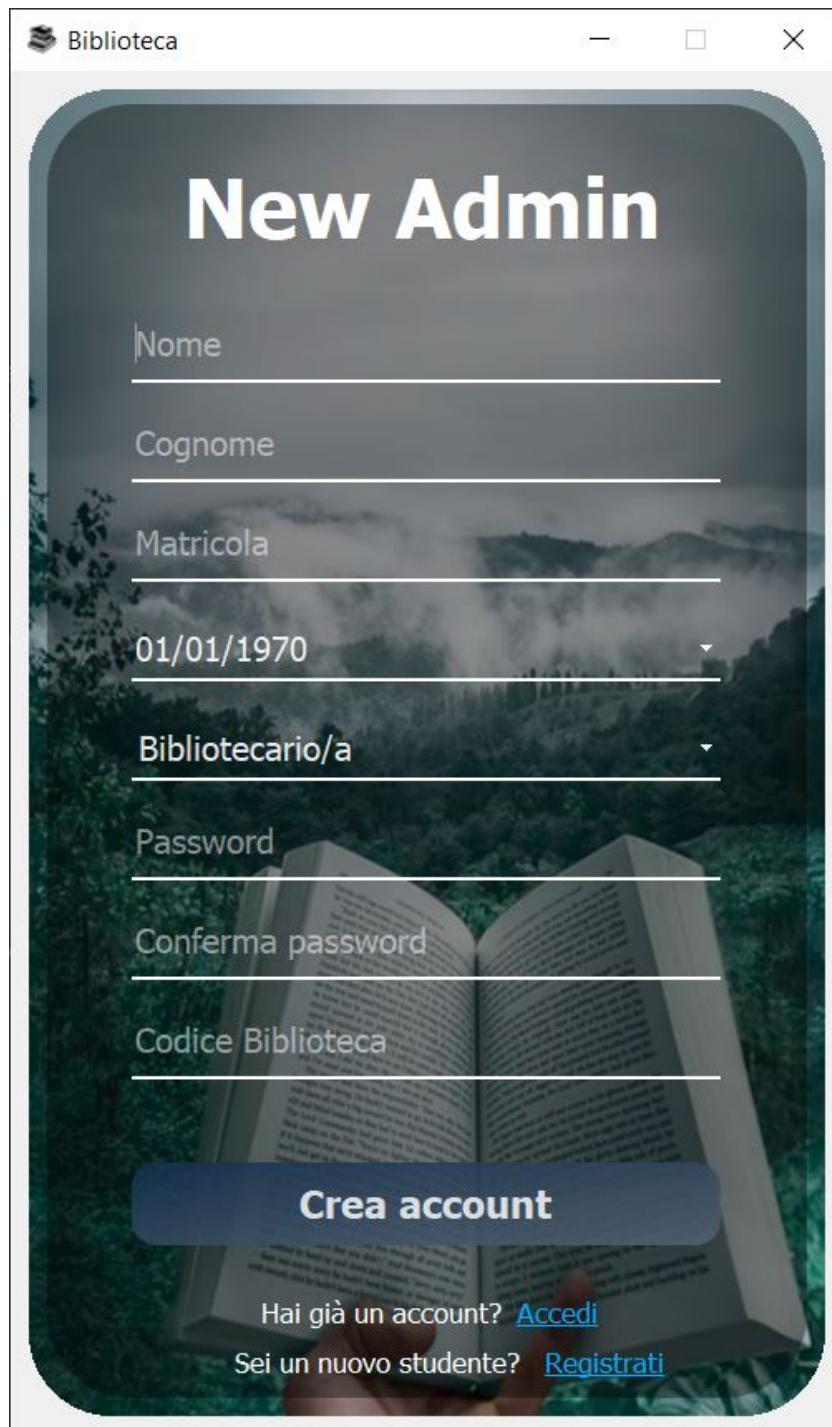


Figura 6.8: RegistrazioneAmministratore

## 6.3.2 Studente

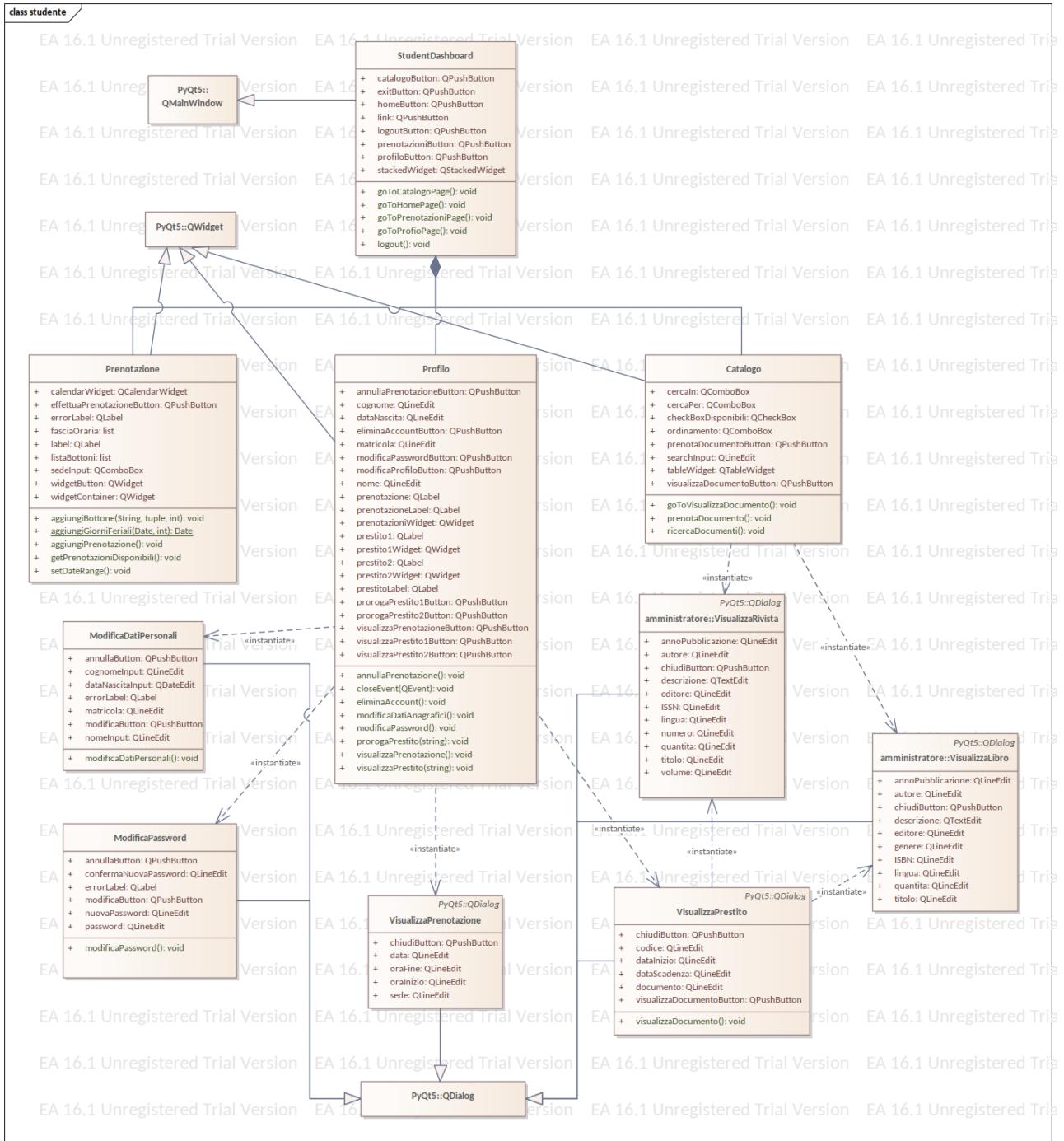


Figura 6.9: Studente

Come mostrato in 6.9 il package *Studente* include le seguenti classi ognuna delle quali importa un file .ui (User Interface):

- StudentDashboard
- Catalogo
- Prenotazione
- Profilo
- ModificaDatiPersonalni
- ModificaPassword
- VisualizzaPrenotazione
- VisualizzaPrestito

# Student Dashboard



Figura 6.10: StudentDashboard

## Catalogo

D Form - [Preview] - Qt Designer

# CATALOGO

search documents

Ordinamento: Titolo A-Z

Solo disponibili

Titolo	Autore	ID	Disponibilità	Tipologia
--------	--------	----	---------------	-----------

Prenota

Visualizza

The screenshot shows a window titled 'CATALOGO'. At the top left is a search bar with a magnifying glass icon and the placeholder text 'search documents'. To its right are two dropdown menus: 'Titolo' and 'Tutto il catalogo'. Below the search bar is a section for sorting, labeled 'Ordinamento:' followed by a dropdown menu set to 'Titolo A-Z'. Next to it is a checkbox labeled 'Solo disponibili'. The main area contains a table with columns: Titolo, Autore, ID, Disponibilità, and Tipologia. To the right of the table are two buttons: 'Prenota' and 'Visualizza', each with an associated icon.

Figura 6.11: Catalogo

## Prenotazione

— □ X

### Prenotazione sala studio

Benvenuto studente!  
In questa sezione è possibile effettuare una prenotazione per le aule studio presenti all'interno delle biblioteche di ateneo.  
E' consentita una sola prenotazione per volta.  
Una prenotazione può durare al massimo 2 ore, dunque è possibile selezionare al massimo 4 fasce orarie consecutive.

Sede Biblioteca Economico-Giuridica-Sociologica

lun	mar	mer	gio	ven	sab	dom
30	31	1	2	3	4	5
6	7	8	9	10	11	12
13	14	15	16	17	18	19
20	21	22	23	24	25	26
27	28	1	2	3	4	5
6	7	8	9	10	11	12

Selezione una fascia oraria:

*error label*

08:30-09:0009:00-09:3009:30-10:00  
10:00-10:3010:30-11:0011:00-11:30  
11:30-12:0012:00-12:3012:30-13:00  
13:00-13:3013:30-14:0014:00-14:30  
14:30-15:0015:00-15:3015:30-16:00  
16:00-16:3016:30-17:0017:00-17:30  
17:30-18:0018:00-18:3018:30-19:00

Effettua prenotazione

Figura 6.12: Prenotazione

## Profilo

Form - [Preview] - Qt Designer

### Informazioni personali

Matricola:

Nome:

Cognome:

Data di nascita:

Modifica dati personali Cambia password Elimina account

---

### Prestiti

*Non sono presenti prestiti*

Prestito 1 Visualizza Proroga

Prestito 2: Visualizza Proroga

---

### Prenotazioni

*Nessuna prenotazione presente*

Prenotazione Visualizza Annulla

Figura 6.13: Profilo

## ModificaDatiPersonali

The screenshot shows a modal dialog box titled "Modifica dati personali - [Preview] ...". At the top right are a question mark icon and a close button. Below the title, a red "error label" is displayed. The form contains four input fields: "Matricola:" (empty), "Nome:" (empty), "Cognome:" (empty), and "Data di nascita:" with the value "01-01-2000" in a dropdown menu. At the bottom are two buttons: "Modifica" with a pencil icon and "Annulla" with a red X icon.

error label

Matricola:

Nome:

Cognome:

Data di nascita:

Modifica Annulla

Figura 6.14: ModificaDatiPersonali

## ModificaPassword



Figura 6.15: ModificaPassword

## VisualizzaPrenotazione

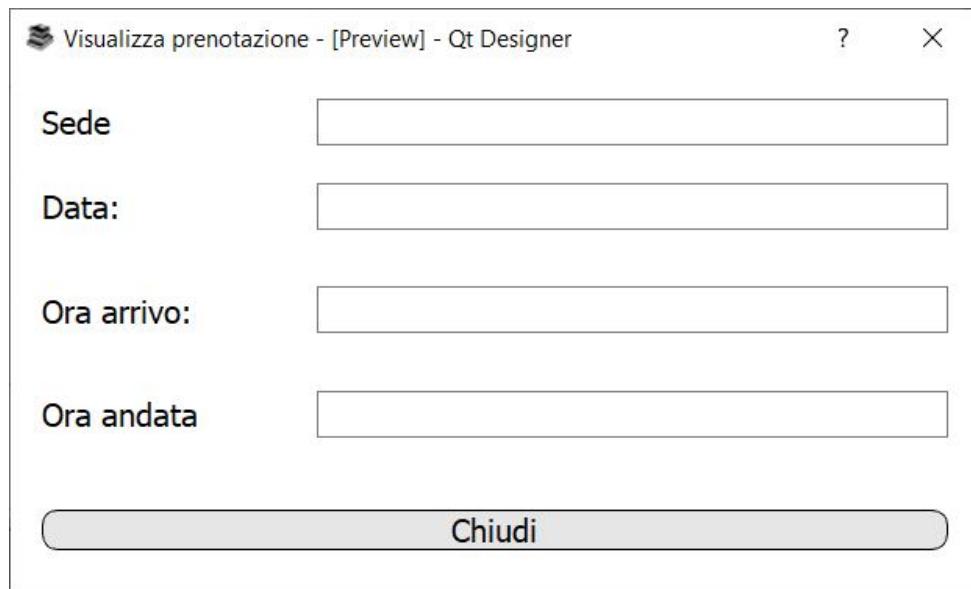


Figura 6.16: VisualizzaPrenotazione

## VisualizzaPrestito

Visualizza Prestito - [Preview] - Qt De... ? X

Prestito n°

Effettuato il:

Valido fino al:

ID Documento:  

**Chiudi**



Figura 6.17: VisualizzaPrestito

### 6.3.3 Amministratore

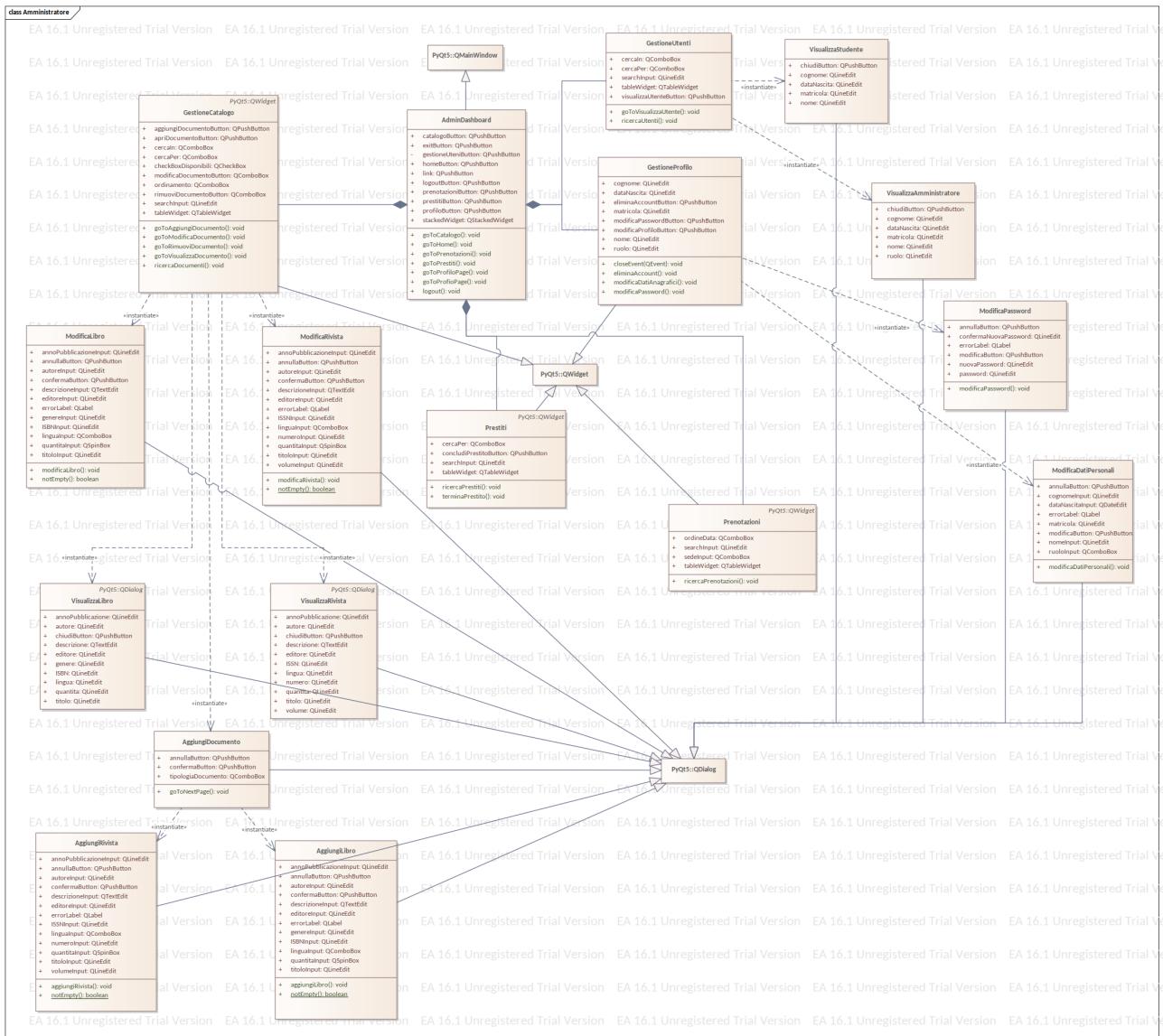


Figura 6.18: Amministratore

Come mostrato in 6.18 il package *Amministratore* include le seguenti classi ognuna delle quali importa un file .ui (User Interface):

- AdminDashboard
- GestioneCatalogo
- Prenotazioni
- Prestiti
- ModificaDatiPersonali
- ModificaPassword
- GestioneProfilo
- GestioneUtenti
- AggiungiDocumento
- AggiungiLibro
- AggiungiRivista
- VisualizzaLibro
- VisualizzaRivista
- ModificaLibro
- ModificaRivista
- VisualizzaAmministratore
- VisualizzaStudente

# AdminDashboard

The screenshot shows the AdminDashboard application window. On the left is a vertical sidebar with a yellow background containing the following menu items:

- HOME
- GESTIONE CATALOGO
- GESTIONE PRESTITI
- GESTIONE PRENOTAZIONI
- GESTIONE UTENTI
- PROFILO
- LOGOUT
- EXIT

The main content area has a white background. At the top right, there is a "Benvenuto!" (Welcome!) message. Below it, a text block describes the system's purpose and features:

Il Sistema Bibliotecario di Ateneo propone a tutti gli studenti un ricco patrimonio bibliografico. La seguente piattaforma consente agli studenti di:  
-Effettuare prestito librario;  
-Prenotare un posto aula studio;  
Inoltre la piattaforma aiuta il personale nella gestione della biblioteca.

**Il software**

A small screenshot of a terminal window is shown, displaying a snippet of Python code. The code appears to be part of a web application, possibly related to the project's backend logic.

La piattaforma è stata realizzata in python 3.10. La GUI è stata realizzata utilizzando PyQt5. Il codice sorgente e la descrizione dettagliata del progetto sono disponibili su GitHub al seguente link:  
[Biblioteca-universitaria](#)

**Chi siamo**

Il software è stato realizzato da FRANCESCO FINIZII e KRISTIAN LIKAJ, studenti di Ingegneria informatica e dell'automazione presso UNIVPM

The logo of the Università Politecnica delle Marche (UPM) is located in the bottom right corner. It features a circular emblem with two figures holding a torch, surrounded by the text "UNIVERSITÀ POLITECNICA DELLE MARCHE".

Figura 6.19: AdminDashboard

## GestioneCatalogo



Figura 6.20: GestioneCatalogo

## Prenotazioni

D Form - [Preview] - Qt Designer

# Gestione prenotazioni

ricerca matricola studente

Sede:

Data:

Matricola	Data	Ora arrivo	Ora andata	Sede	

Figura 6.21: Prenotazioni

## Prestiti

Form - [Preview] - Qt Designer

### Gestione prestiti

 ricerca prestiti

Codice	titolo studente	D documenti	Scadenza
--------	-----------------	-------------	----------

 Concludi

Figura 6.22: Prestiti

## ModificaDatiPersonali

Modifica dati personali - [Preview] ... ? X

error label

Matricola:

Nome:

Cognome:

Data di nascita:  ▾

Ruolo  ▾

 **Modifica**  **Annulla**

Figura 6.23: ModificaDatiPersonali

## ModificaPassword

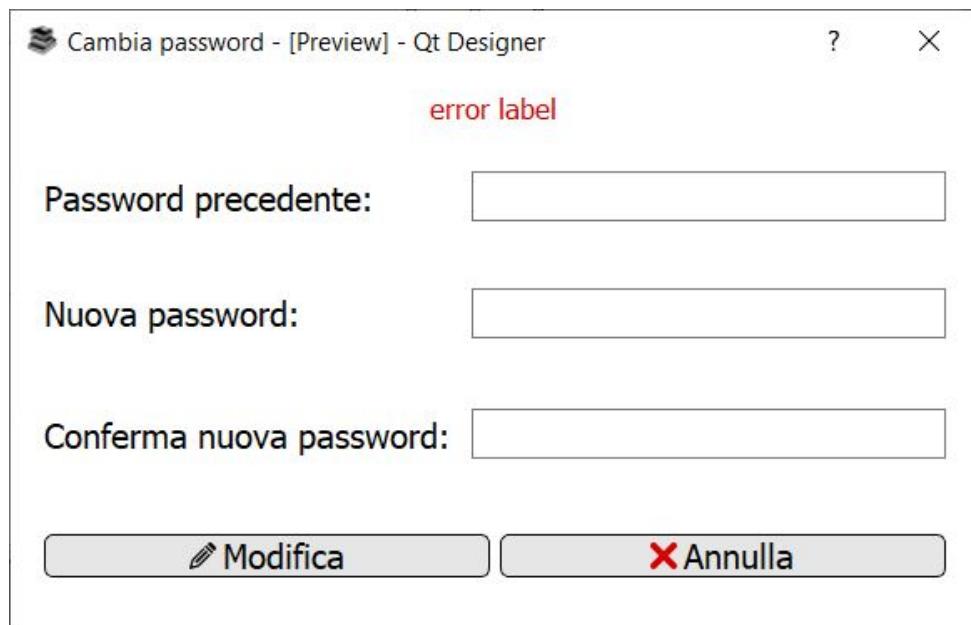


Figura 6.24: ModificaPassword

## GestioneProfilo

The screenshot shows a window titled "Form - [Preview] - Qt Designer". The main title of the form is "Informazioni personali". The form contains five input fields with labels: "Matricola:", "Nome:", "Cognome:", "Data di nascita:", and "Ruolo:". Below these fields is a row of three buttons: "Modifica dati personali" (with a pencil icon), "Cambia password" (with a pencil icon), and "Elimina account" (with a trash bin icon).

Matricola:

Nome:

Cognome:

Data di nascita:

Ruolo:

Modifica dati personali   Cambia password   Elimina account

Figura 6.25: GestioneProfilo

## GestioneUtenti

Form - [Preview] - Qt Designer

### Gestione Utenti

search users

Matricola	Nome	Cognome	Tipologia
-----------	------	---------	-----------

Visualizza

The screenshot shows a window titled "Gestione Utenti". At the top, there is a search bar with a magnifying glass icon and the placeholder text "search users". Below the search bar is a table with four columns labeled "Matricola", "Nome", "Cognome", and "Tipologia". To the right of the table is a "Visualizza" button. The window has standard operating system controls (minimize, maximize, close) at the top right.

Figura 6.26: GestioneUtenti

## AggiungiDocumento

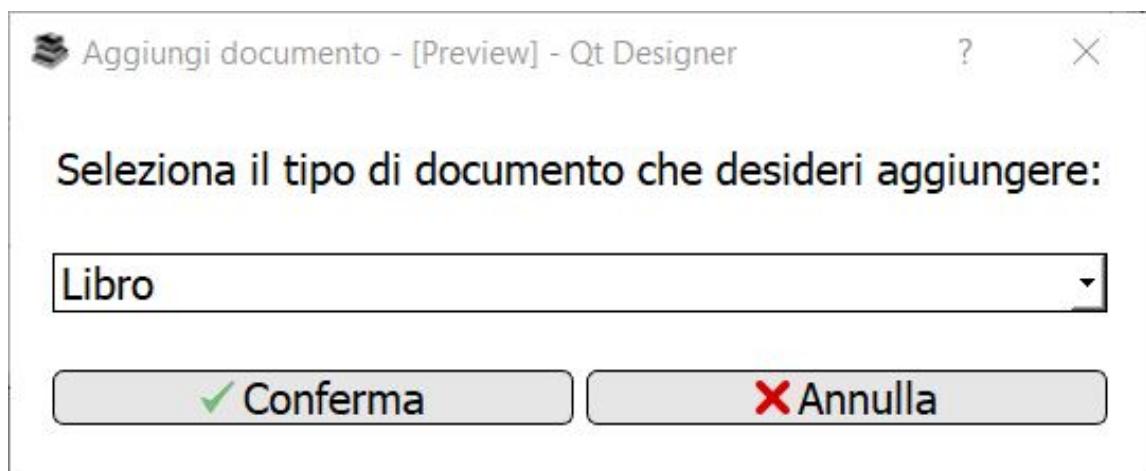


Figura 6.27: AggiungiDocumento

## AggiungiLibro

Aggiungi-modifica libro - [Preview] - Qt Designer ? X

error label

**Titolo:**

**Autore:**

**Editore:**

**lingua:**

**Anno pubblicazione:**

**Genere:**

**ISBN:**

**Disponibilità:**  ▲ ▼

**Descrizione:**

 Conferma  Annulla

Figura 6.28: AggiungiLibro

## AggiungiRivista

Aggiungi-modifica rivista - [Preview] - Qt Designer ? X

error label

**Titolo:**

**Autore:**

**Editore:**

**lingua:**

**Anno pubblicazione:**

**ISSN:**

**Volume:**

**Numero:**

**Disponibilità:**  ▲ ▼

**Descrizione:**

Figura 6.29: AggiungiRivista

## ModificaLibro

Aggiungi-modifica libro - [Preview] - Qt Designer ? X

error label

**Titolo:** titolo

**Autore:** autore

**Editore:** editore

**lingua:** italiano ▾

**Anno pubblicazione:** anno pubblicazione

**Genere:** genere

**ISBN:** ISBN

**Disponibilità:** 0 unità ▾

**Descrizione:** Descrizione

Conferma  Annulla

Figura 6.30: ModificaLibro

## ModificaRivista

Aggiungi-modifica rivista - [Preview] - Qt Designer ? X

error label

<b>Titolo:</b>	titolo
<b>Autore:</b>	autore
<b>Editore:</b>	editore
<b>lingua:</b>	italiano ▾
<b>Anno pubblicazione:</b>	anno pubblicazione
<b>ISSN:</b>	ISBN
<b>Volume:</b>	volume
<b>Numero:</b>	numero
<b>Disponibilità:</b>	0 unità ▾
<b>Descrizione:</b>	descrizione

Conferma       Annulla

Figura 6.31: ModificaRivista

## VisualizzaLibro



Figura 6.32: VisualizzaLibro

## VisualizzaRivista

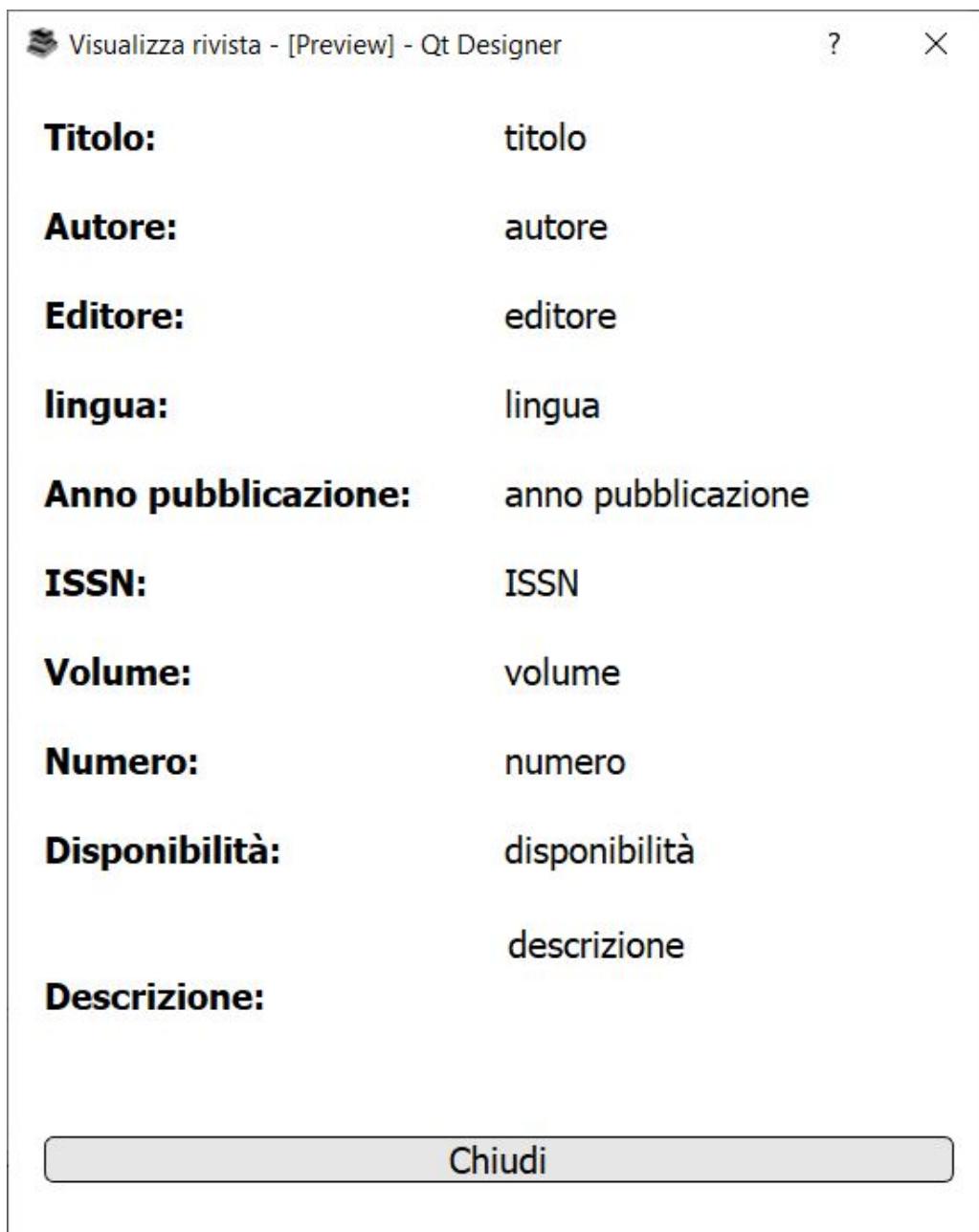


Figura 6.33: VisualizzaRivista

## VisualizzaAmministratore

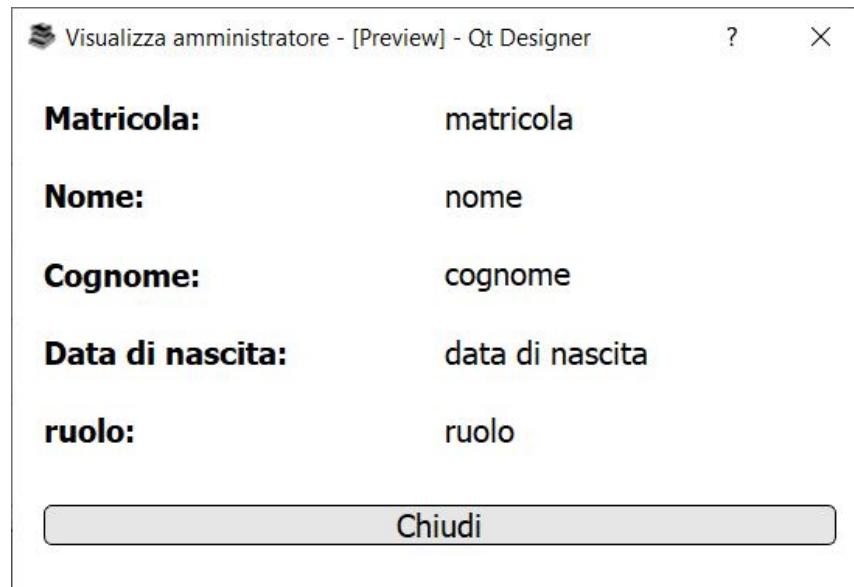


Figura 6.34: VisualizzaAmministratore

## VisualizzaStudente

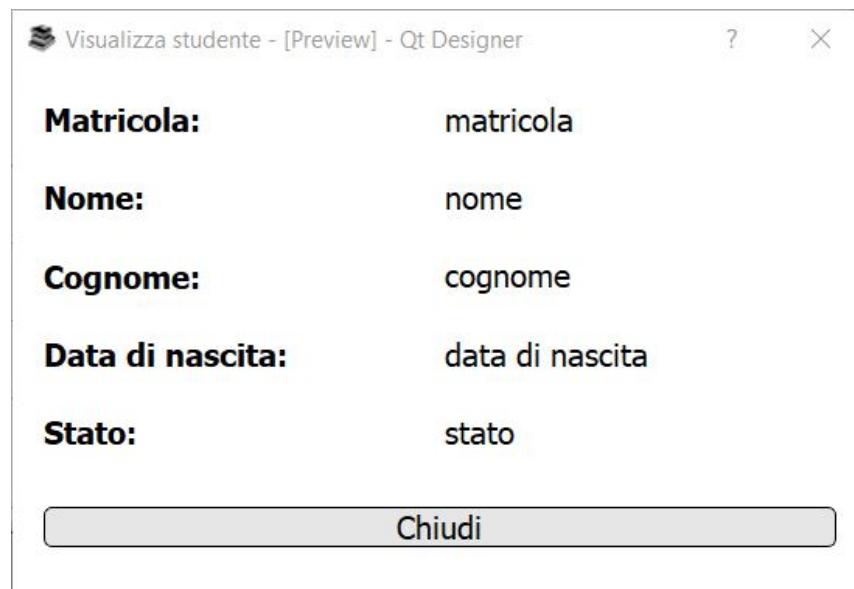


Figura 6.35: VisualizzaStudente

## 7. DIAGRAMMI DI SEQUENZA

Per avere una maggiore chiarezza del funzionamento del software sono stati realizzati dei diagrammi di sequenza. I casi d'uso analizzati sono i seguenti:

- AggiungiDocumento
- ModificaDocumento
- RimuoviDocumento
- Autenticazione
- EffettuaPrestito
- VisualizzaListaPrenotazioni
- VisualizzaPrestito

## 7.1 AggiungiDocumento

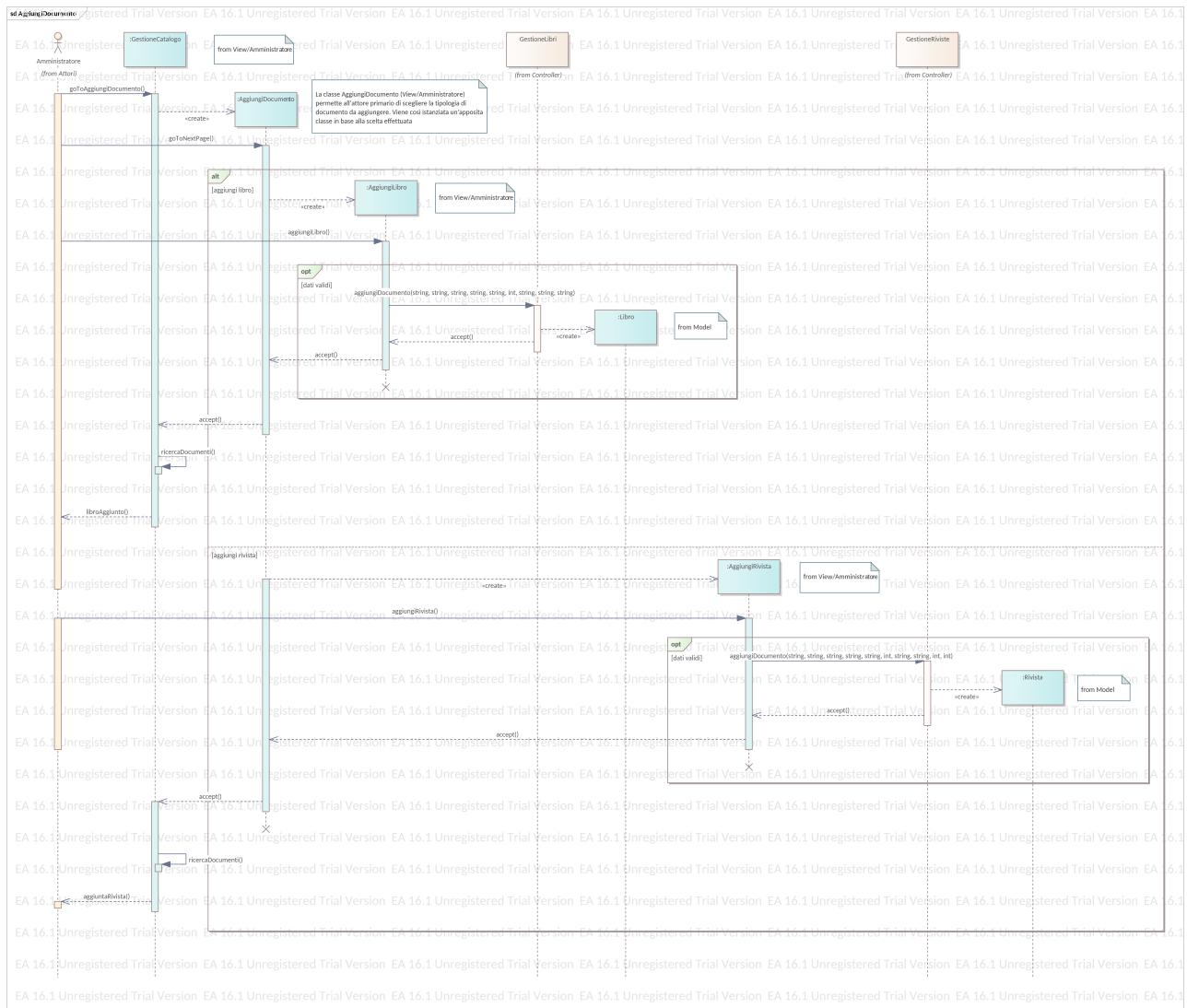


Figura 7.1: sd: AggiungiDocumento

## 7.2 ModificaDocumento

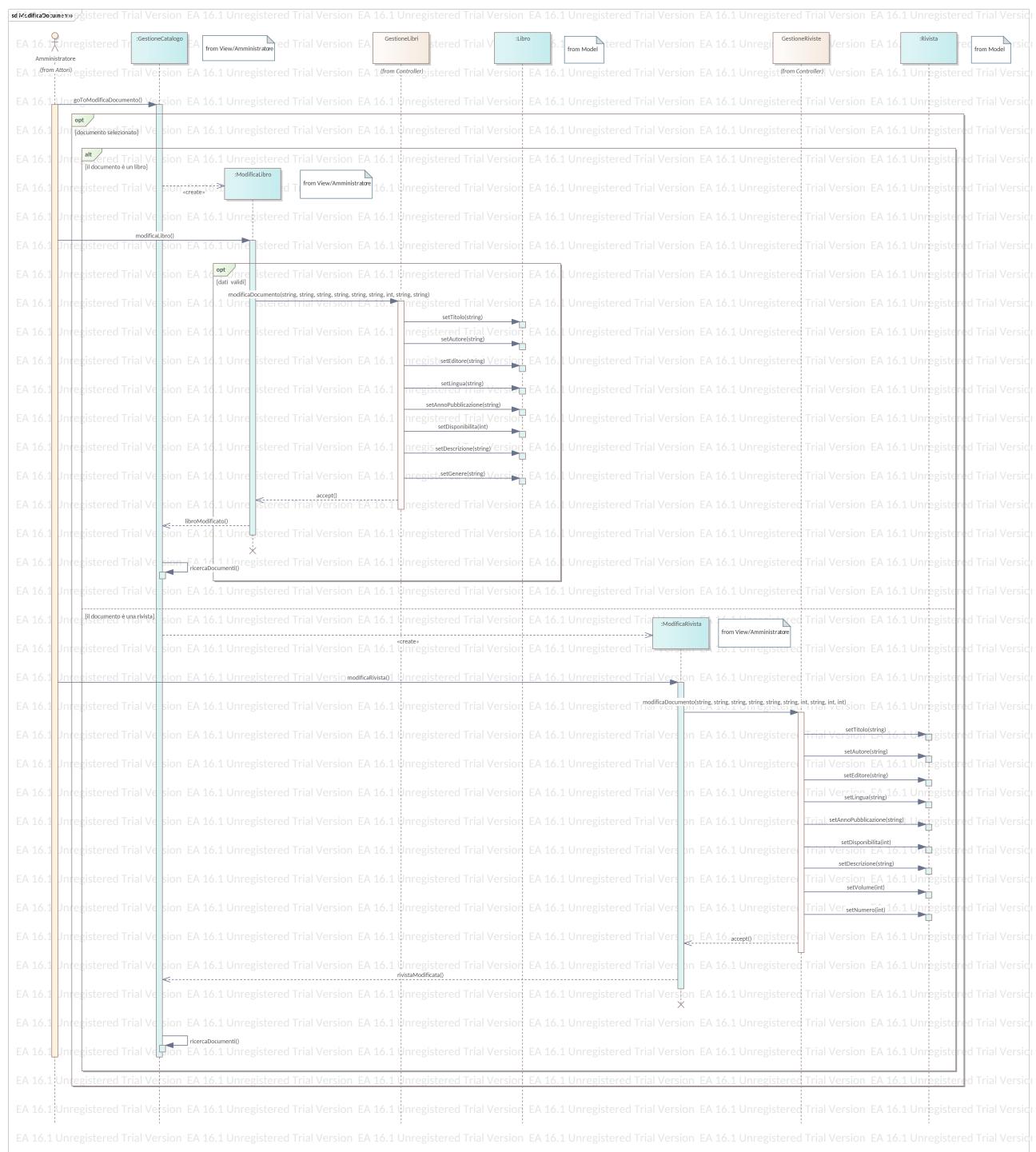


Figura 7.2: sd: ModificaDocumento

## 7.3 RimuoviDocumento

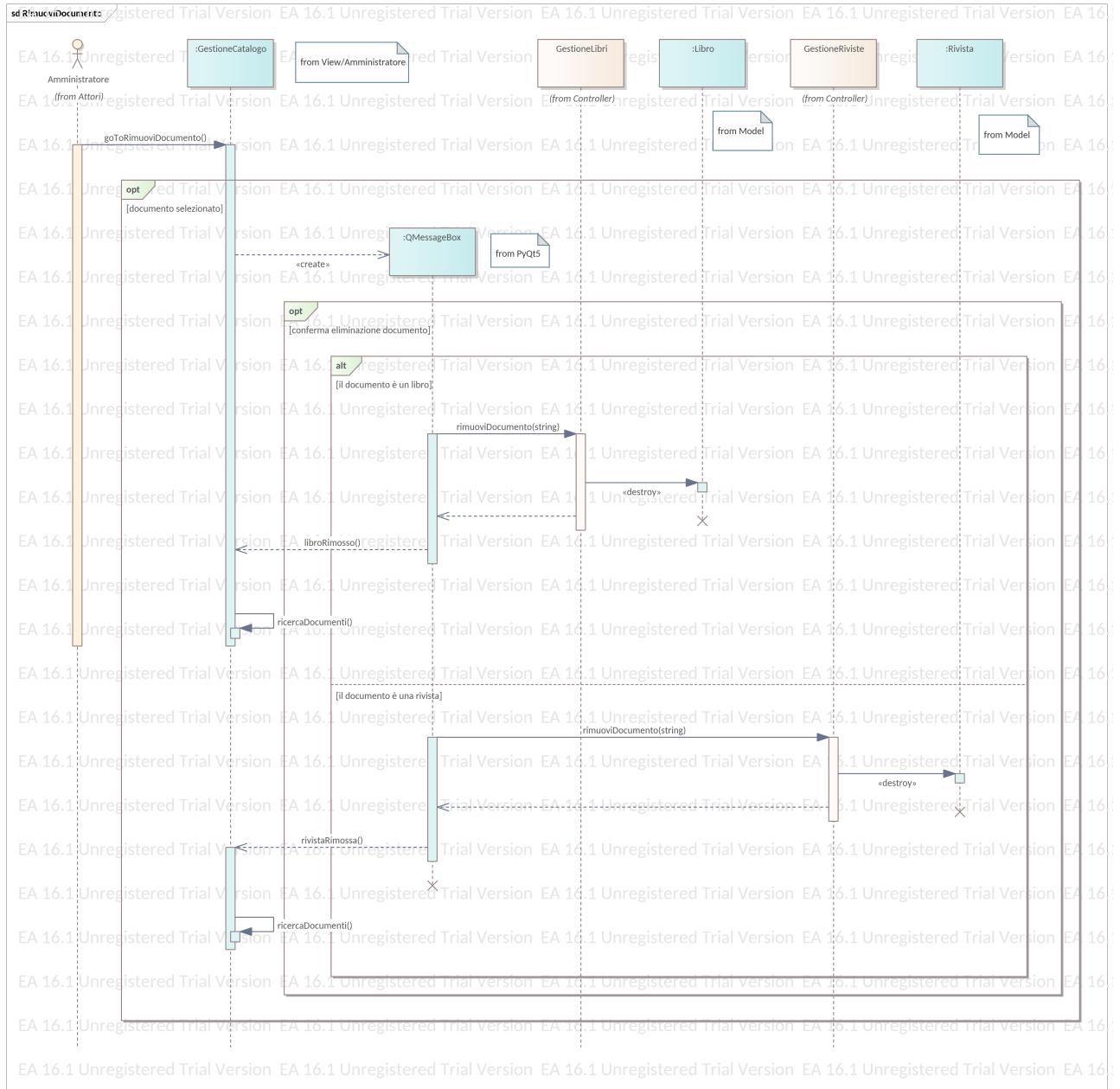


Figura 7.3: sd: RimuoviDocumento

## 7.4 Autenticazione

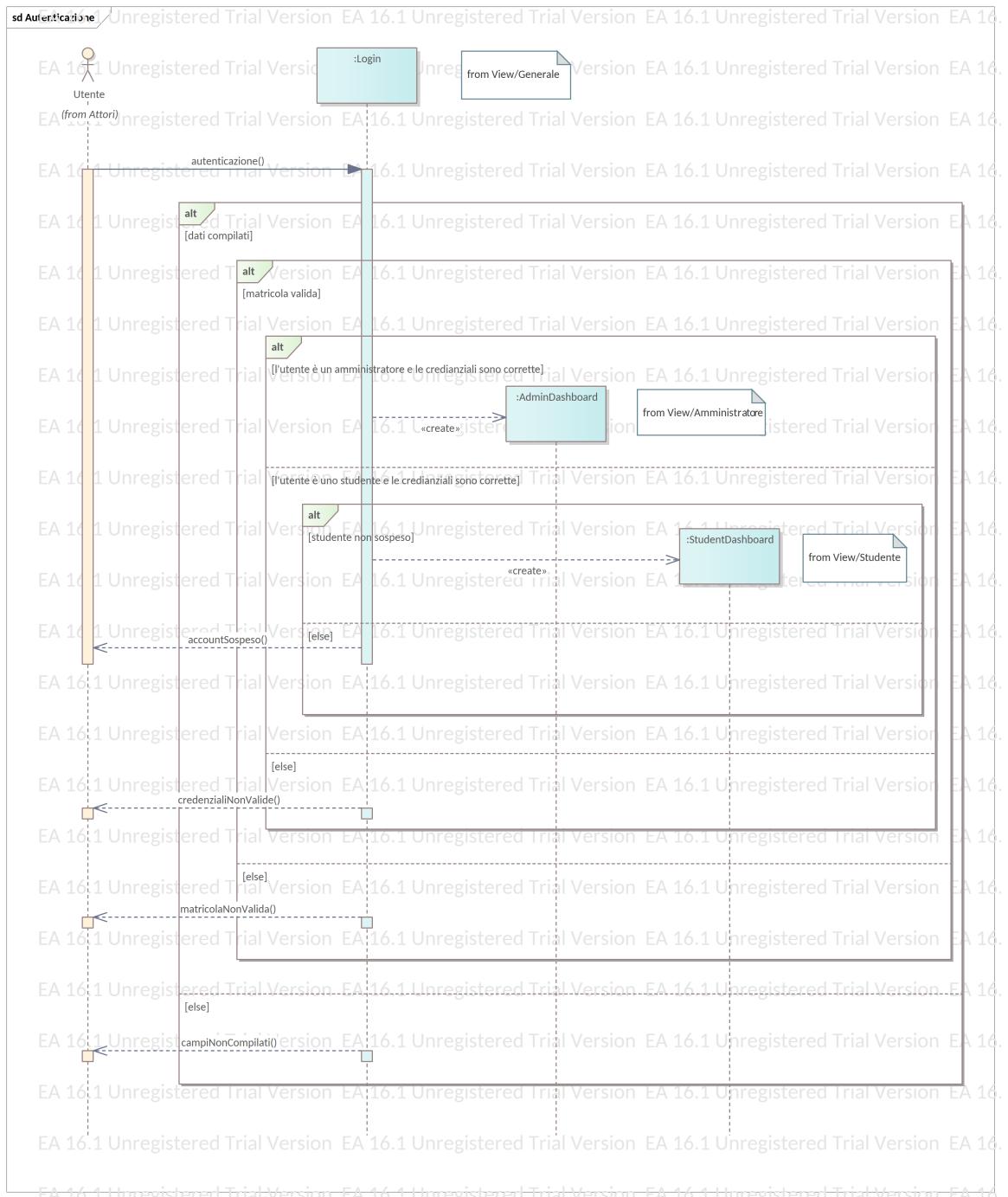


Figura 7.4: sd: Autenticazione

## 7.5 EffettuaPrestito

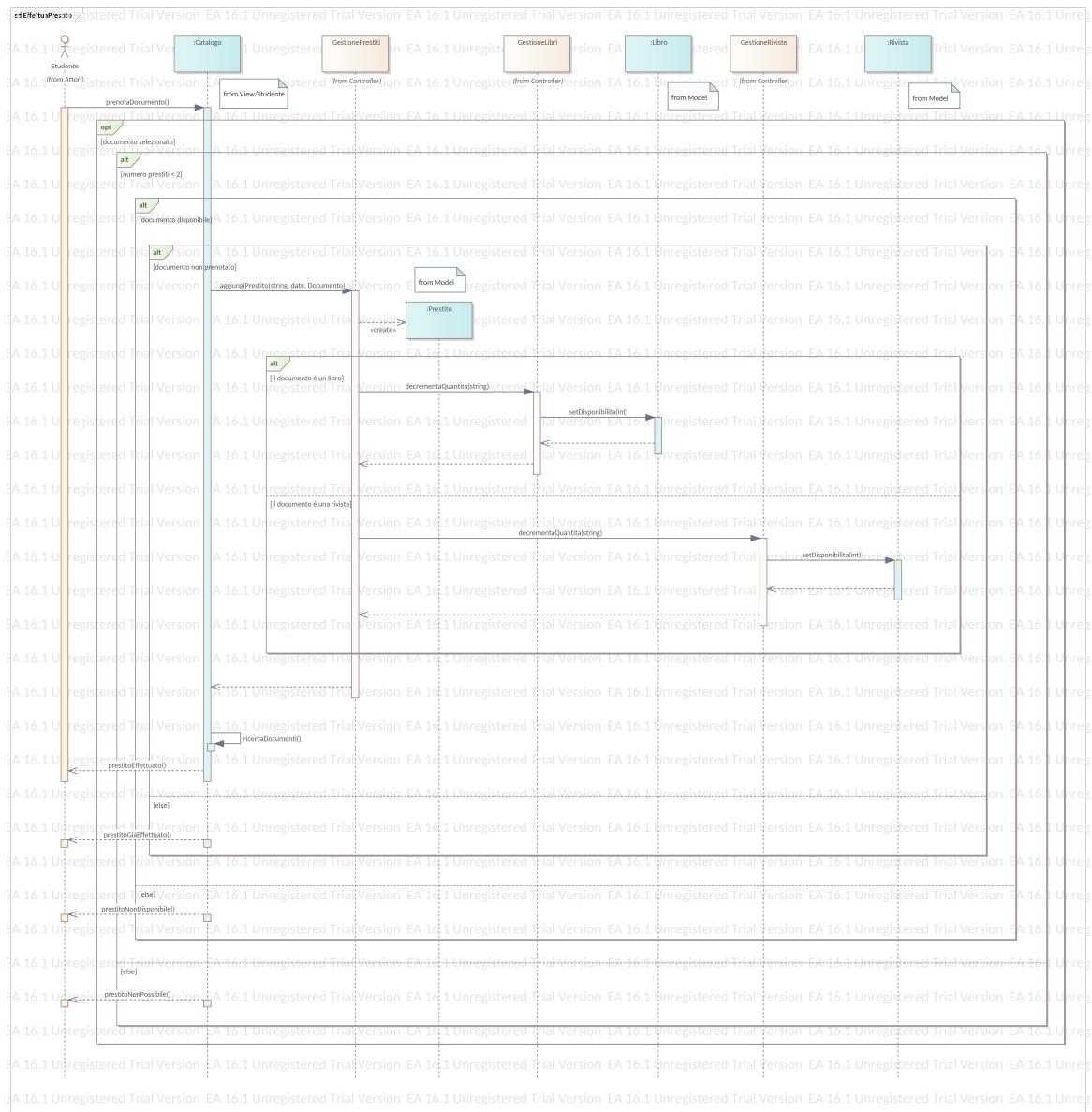


Figura 7.5: sd: EffettuaPrestito

## 7.6 VisualizzaListaPrenotazioni

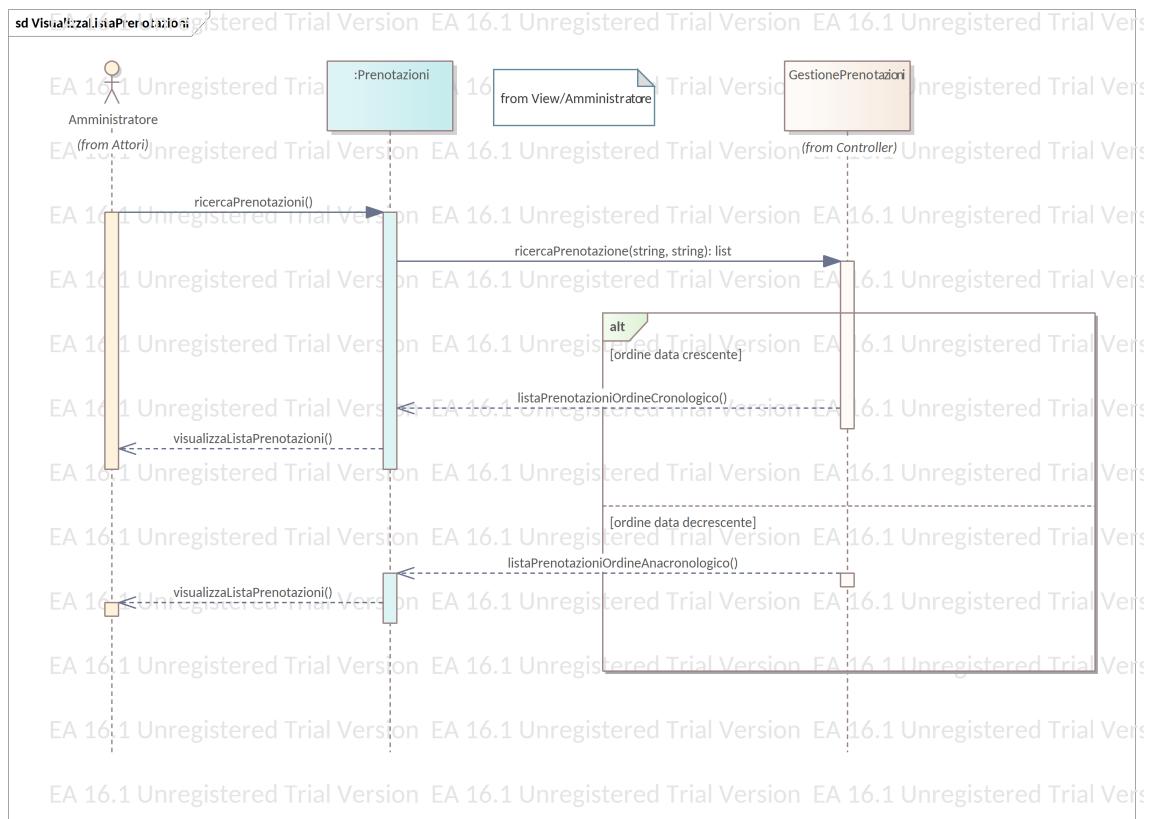


Figura 7.6: sd: VisualizzaListaPrenotazioni

## 7.7 VisualizzaPrestito

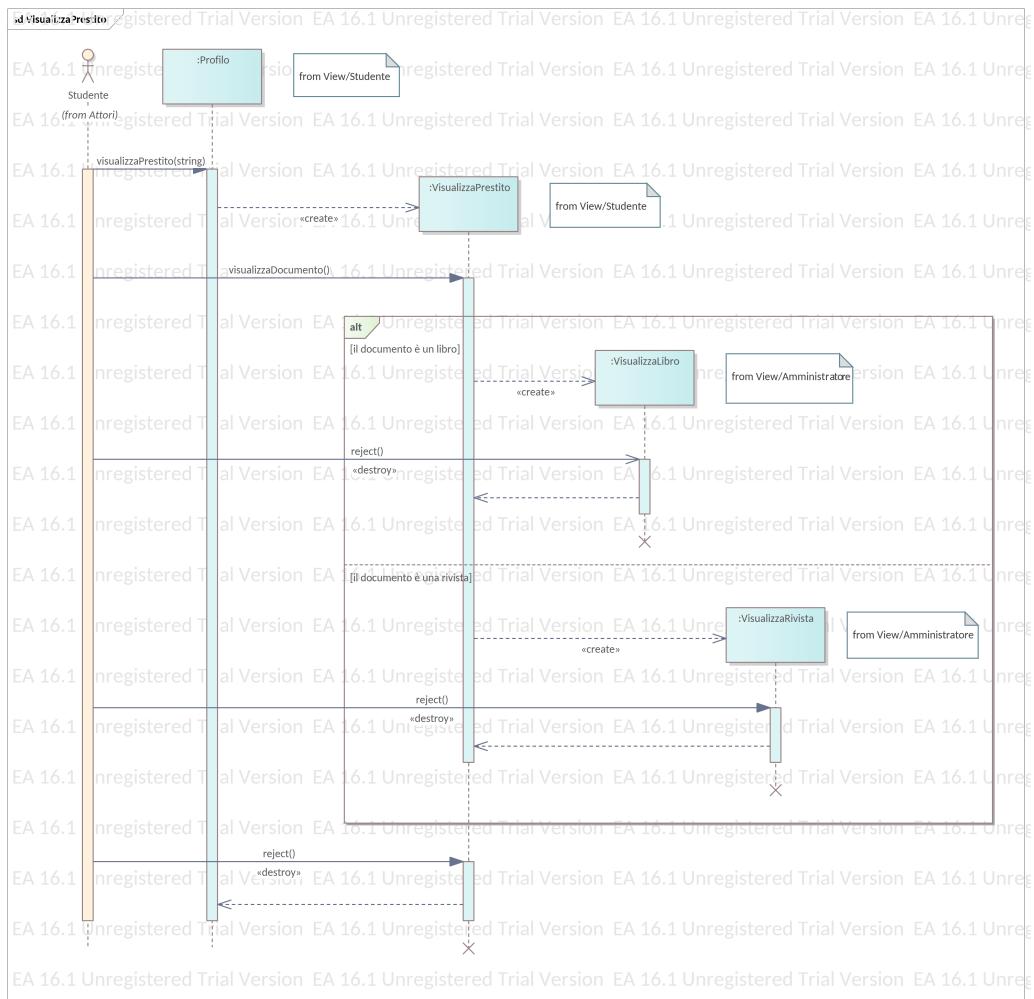


Figura 7.7: sd: VisualizzaPrestito

## **8. DIAGRAMMI DI ATTIVITÀ**

Per avere una maggiore chiarezza del funzionamento del software sono stati realizzati dei diagrammi di attività. I casi d'uso analizzati sono i seguenti:

- Autenticazione
- TerminaPrestito
- VisualizzaListaPrenotazioni
- RicercaPrenotazione

## 8.1 Autenticazione

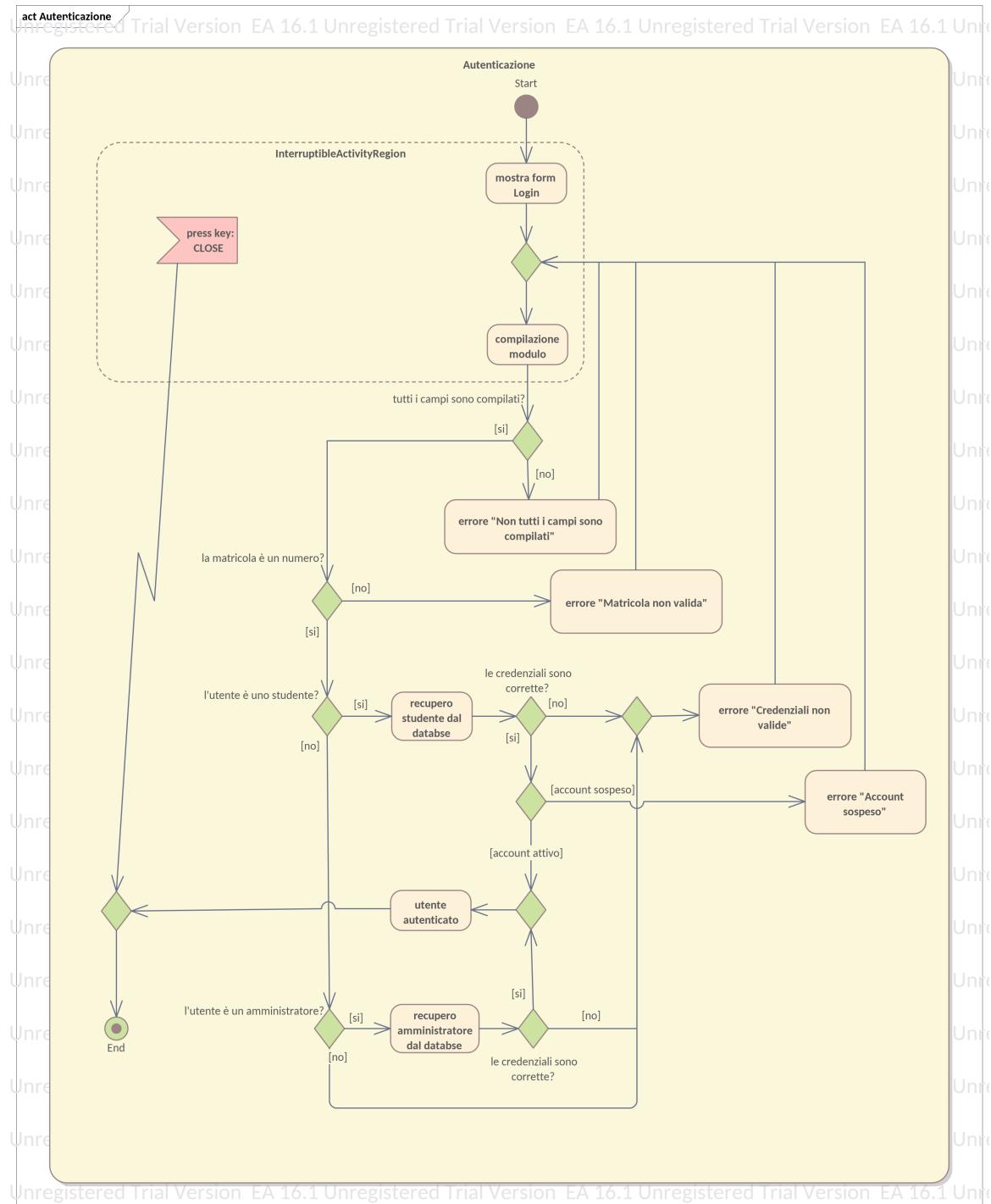


Figura 8.1: act: Autenticazione

## 8.2 VisualizzaListaPrenotazioni

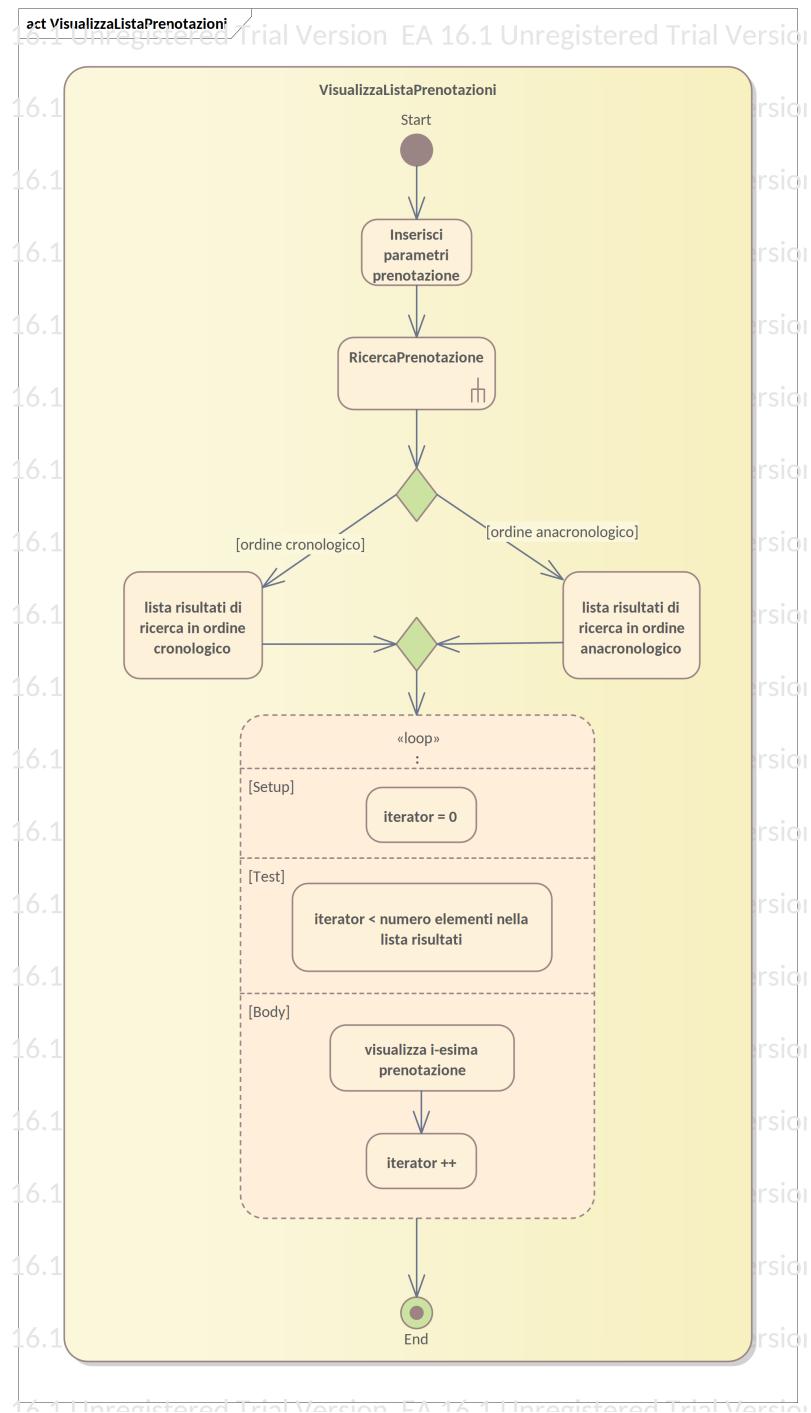


Figura 8.2: act: VisualizzaListaPrenotazioni

## 8.3 act-RicercaPrenotazione

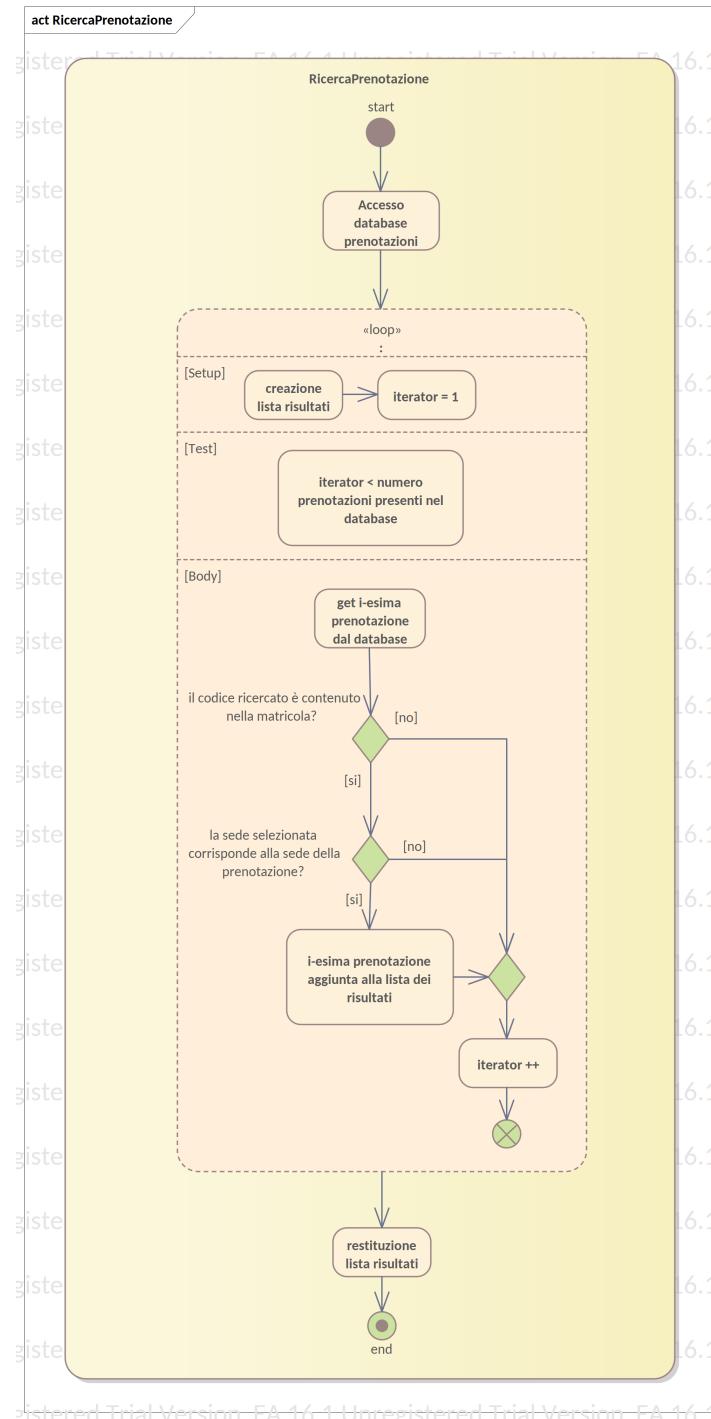


Figura 8.3: act: RicercaPrenotazione

## 8.4 TerminaPrestito

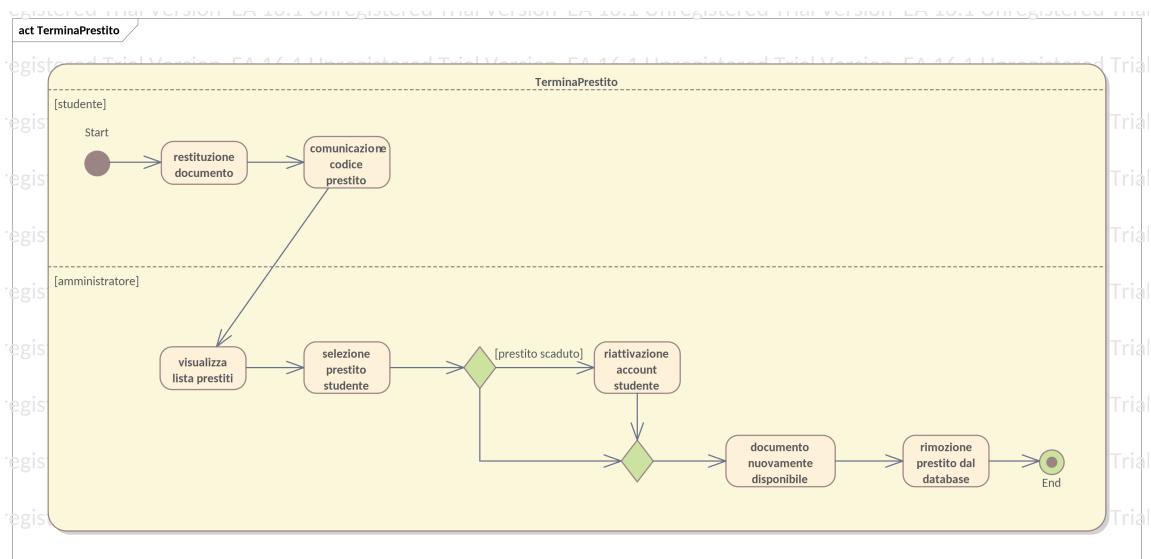


Figura 8.4: act: TerminaPrestito

## **9. DIAGRAMMI DELLE MACCHINE A STATI**

Il diagramma delle macchine a stati raffigura gli oggetti di una classe delineando i vari stati ad essi collegati all'interno del sistema: Le classi per cui si è realizzato tale diagramma sono:

- Prenotazione
- Prestito
- Studente

## 9.1 Prenotazione

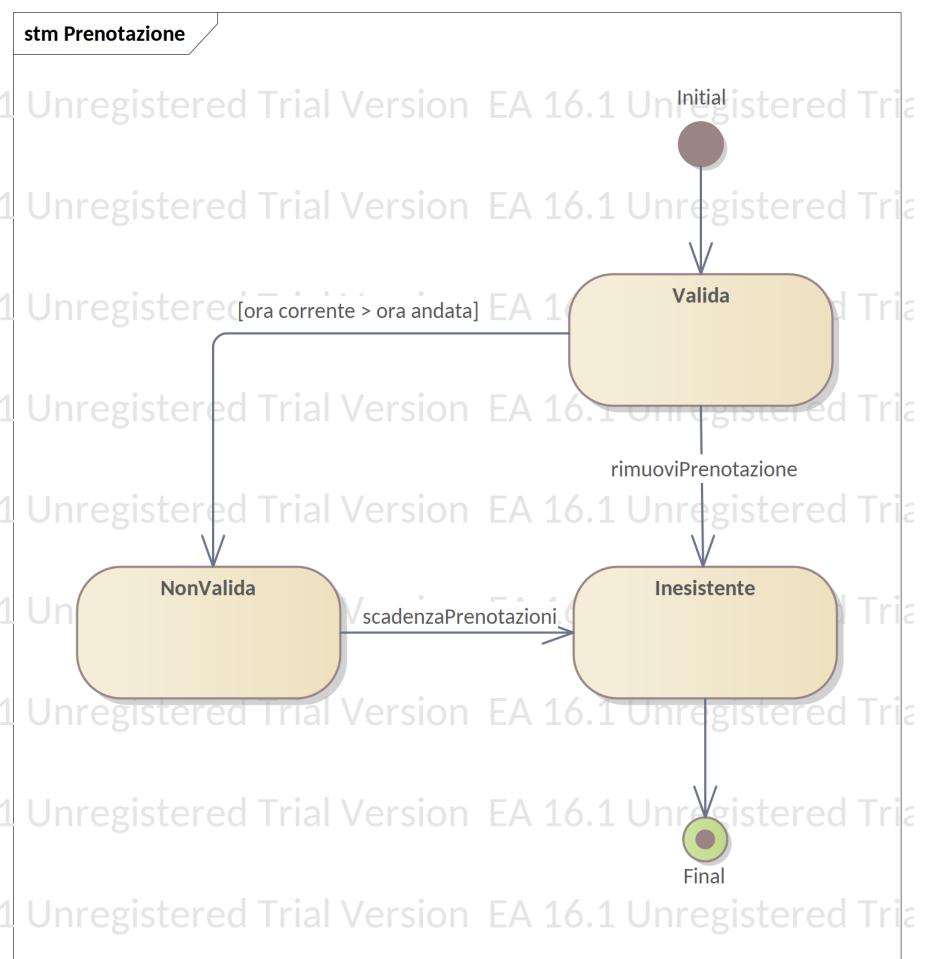


Figura 9.1: Diagramma delle macchine a stati: Prenotazione

## 9.2 Prestito

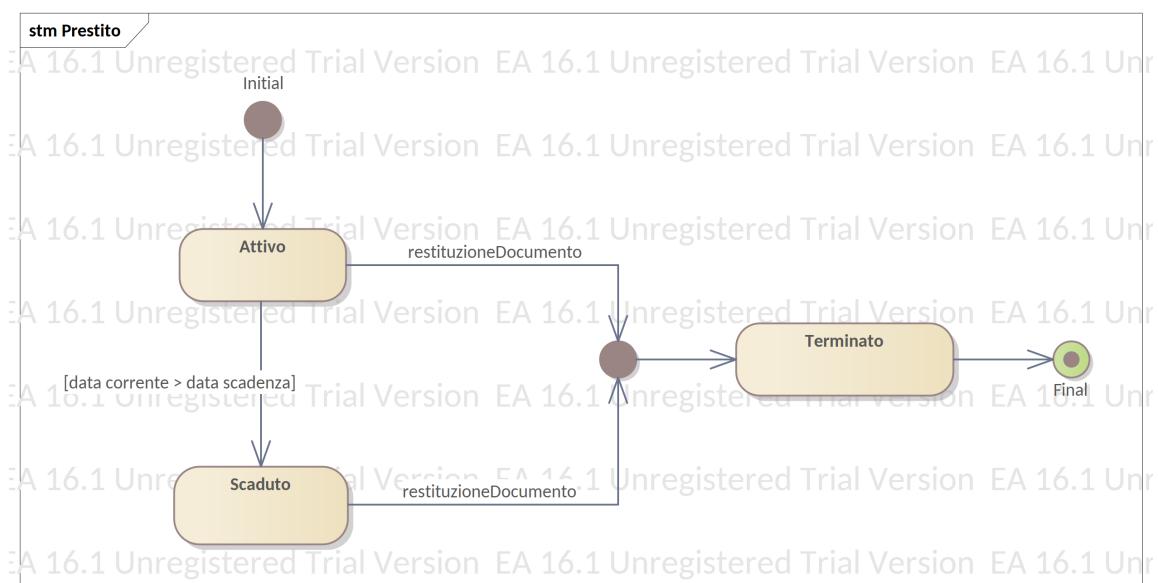


Figura 9.2: Diagramma delle macchine a stati: Prestito

### 9.3 Studente

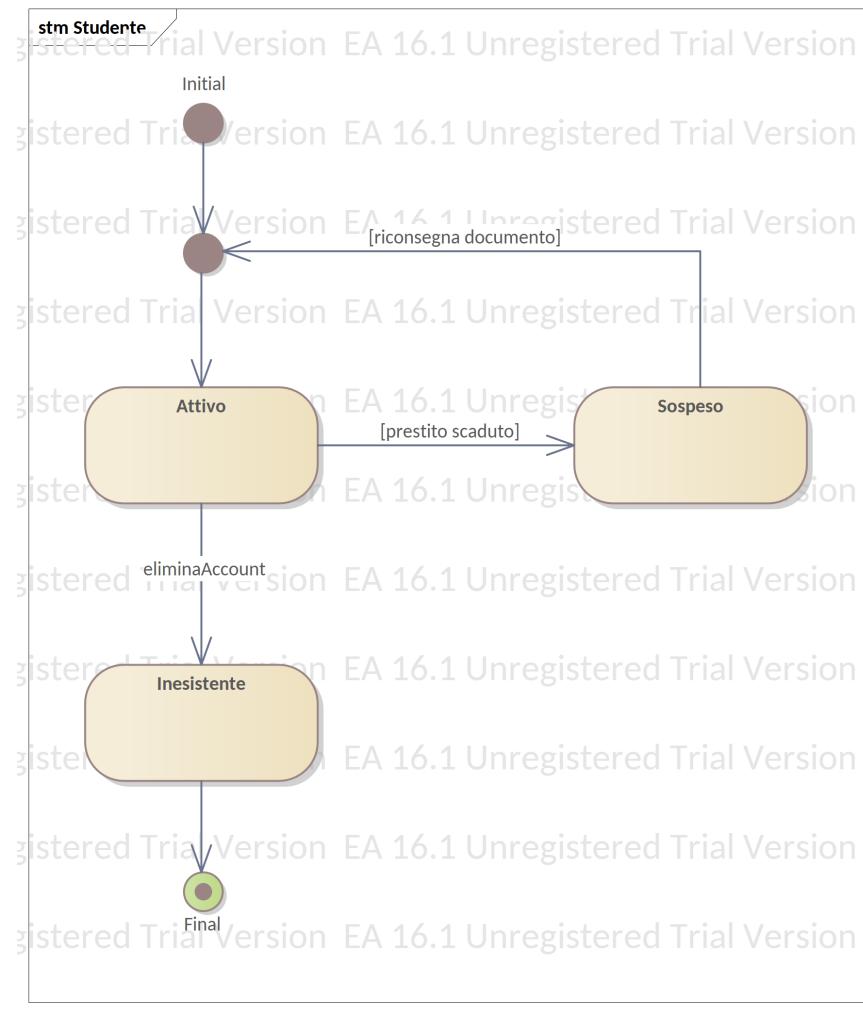


Figura 9.3: Diagramma delle macchine a stati: Studente

# 10. DIAGRAMMA DEI COMONENTI

Il diagramma dei componenti rappresentato in 10.1 mostra la struttura del sistema software dal punto di vista dello studente.

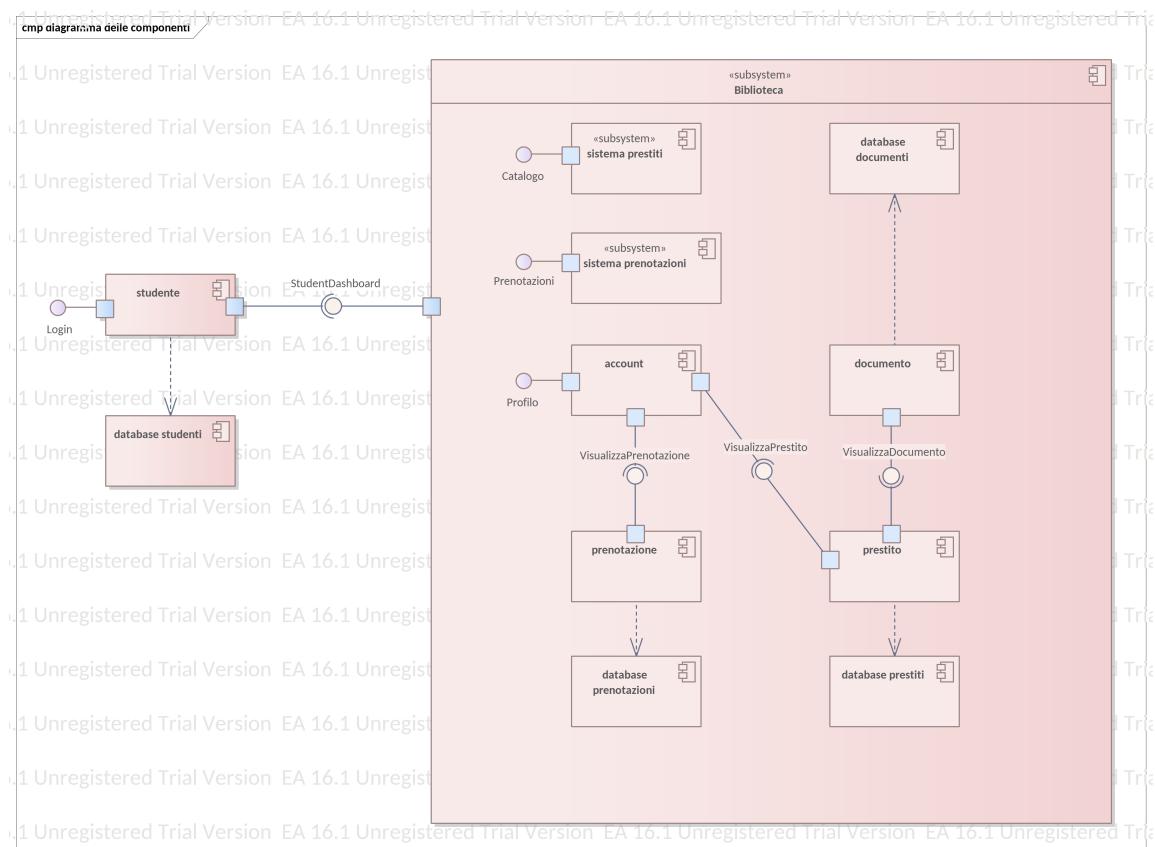


Figura 10.1: Diagramma dei componenti

# 11. DIAGRAMMA DI DEPLOYMENT

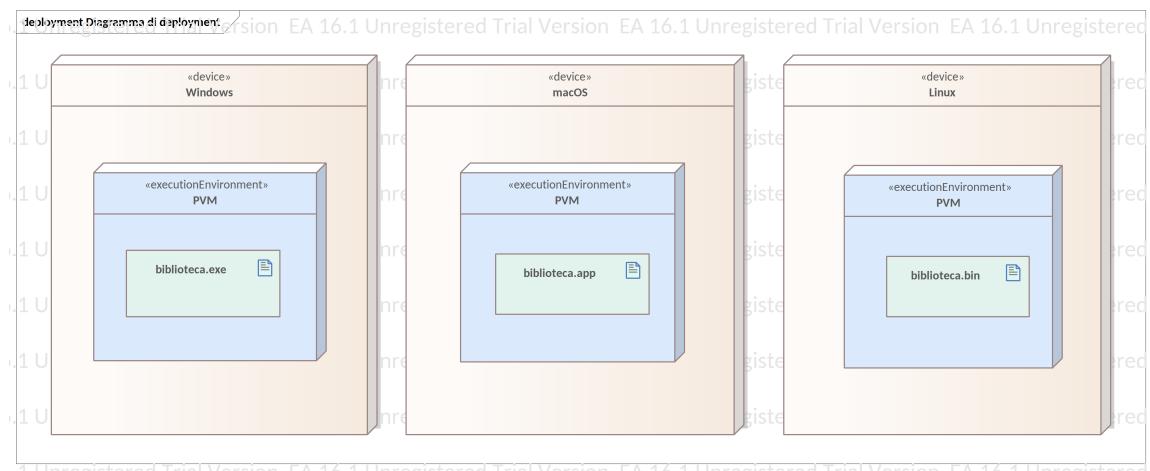


Figura 11.1: Diagramma di deployment

## 12. UNIT TEST

Durante la fase di implementazione, tramite il modulo **unittest** si sono utilizzati segmenti di codice per testare il corretto funzionamento del software. La classe "test" contenuta nella cartella "Tests" situata nella directory principale del progetto contiene le seguenti funzioni:

- test\_studente
- test\_amministratore
- test\_libro
- test\_rivista
- test\_prestito
- test\_prenotazione

## 12.1 test\_studente

```
def test_studente(self):
    #aggiunta studente
    GestioneStudenti.aggiungiUtente('Kristian', 'Likaj', '1097605', '12345678', datetime.date(2000, 10, 10))
    self.assertIsInstance(GestioneStudenti.getUtente('1097605'), Studente)

    #verifica modifiche studente
    GestioneStudenti.modificaUtente('Kristian', 'Likaj', '1097605', 'password', datetime.date(2000, 10, 10))
    self.assertEqual(GestioneStudenti.getUtente("1097605").getPassword(), "password")

    #rimozione studente
    GestioneStudenti.rimuoviUtente("1097605")
    self.assertIsNone(GestioneStudenti.getUtente("1097605"))
```

Figura 12.1: test Studente

## 12.2 test\_amministratore

```
def test_amministratore(self):
    # aggiunta amministratore
    GestioneAmministratori.aggiungiUtente('Mario', 'Rossi', '12345678', 'asdfghjkl', datetime.date(1970, 11, 11),
                                           'Assistente')
    self.assertIsInstance(GestioneAmministratori.getUtente('12345678'), Amministratore)

    # verifica modifiche amministratore
    GestioneAmministratori.modificaUtente('Mario', 'Rossi', '12345678', 'password', datetime.date(1970, 11, 11),
                                           'Assistente')
    self.assertEqual(GestioneAmministratori.getUtente("12345678").getPassword(), "password")

    # rimozione amministratore
    GestioneAmministratori.rimuoviUtente("12345678")
    self.assertIsNone(GestioneAmministratori.getUtente("12345678"))
```

Figura 12.2: test Amministratore

## 12.3 test\_libro

```
def test_libro(self):
    #aggiunta libro
    GestioneLibri.aggiungiDocumento('Follia', 'Patrick McGrath', 'Adelphi', 'italiano', '2012', 2, 'bellissimo',
                                    '1234567891234', 'Romanzo')
    self.assertIsInstance(GestioneLibri.getDocumento('1234567891234'), Libro)

    #verifica modifica libro
    GestioneLibri.modificaDocumento('1234567891234', 'Follia', 'Patrick McGrath', 'Adelphi', 'italiano', '2012',
                                    2, 'consigliato', 'Romanzo')
    self.assertEqual(GestioneLibri.getDocumento('1234567891234').getDescrizione(), "consigliato")

    #verifica incremento quantità
    quantitaIniziale = GestioneLibri.getDocumento("1234567891234").getDisponibilita()
    GestioneLibri.incrementaQuantita("1234567891234")
    self.assertEqual(GestioneLibri.getDocumento('1234567891234').getDisponibilita(), quantitaIniziale + 1)

    # verifica decremento quantità
    GestioneLibri.decrementaQuantita("1234567891234")
    self.assertEqual(GestioneLibri.getDocumento('1234567891234').getDisponibilita(), quantitaIniziale)

    #rimozione libro
    GestioneLibri.rimuoviDocumento("1234567891234")
    self.assertIsNone(GestioneLibri.getDocumento('1234567891234'))
```

Figura 12.3: test Libro

## 12.4 test\_rivista

```
def test_rivista(self):
    # aggiunta rivista
    GestioneRiviste.aggiungiDocumento('Follia', 'Patrick McGrath', 'Adelphi', 'italiano', '2012', 2, 'bellissimo',
                                      '12341234', 1, 3)
    self.assertIsInstance(GestioneRiviste.getDocumento('12341234'), Rivista)

    # verifica modifica rivista
    GestioneRiviste.modificaDocumento('12341234', 'Follia', 'Patrick McGrath', 'Adelphi', 'italiano', '2012', 2,
                                      'consigliato', 1, 3)
    self.assertEqual(GestioneRiviste.getDocumento('12341234').getDescrizione(), "consigliato")

    # verifica incremento quantità
    quantitaIniziale = GestioneRiviste.getDocumento("12341234").getDisponibilita()
    GestioneRiviste.incrementaQuantita("12341234")
    self.assertEqual(GestioneRiviste.getDocumento('12341234').getDisponibilita(), quantitaIniziale + 1)

    # verifica decremento quantità
    GestioneRiviste.decrementaQuantita("12341234")
    self.assertEqual(GestioneRiviste.getDocumento('12341234').getDisponibilita(), quantitaIniziale)

    # rimozione rivista
    GestioneRiviste.rimuoviDocumento("12341234")
    self.assertIsNone(GestioneRiviste.getDocumento('12341234'))
```

Figura 12.4: test Rivista

## 12.5 test\_prestito

```
def test_prestito(self):
    #aggiungi prestito
    GestioneLibri.aggiungiDocumento('Follia', 'Patrick McGrath', 'Adelphi', 'italiano', '2012', 2, 'bellissimo',
                                     '1234567891234', 'Romanzo')
    GestioneStudenti.aggiungiUtente('Kristian', 'Likaj', '1097605', '12345678', datetime.date(2000, 10, 10))
    libro = GestioneLibri.getDocumento('1234567891234')
    dataInizio = datetime.date.today()
    dataScadenza = dataInizio + datetime.timedelta(days=30)
    GestionePrestiti.aggiungiPrestito('1097605', dataInizio, libro)
    prestiti = GestionePrestiti.ricercaPrestitoPerMatricola('1097605')
    self.assertIsInstance(prestiti[0], Prestito)

    #modifica prestito
    GestionePrestiti.prorogaScadenzaPrestito(prestiti[0].getCodice())
    self.assertEqual(prestiti[0].getDataScadenza(), dataScadenza)

    #rimozione prestito
    GestionePrestiti.rimuoviPrestito(prestiti[0].getCodice())
    self.assertIsNone(GestionePrestiti.getPrestito(prestiti[0].getCodice()))
```

Figura 12.5: test Prestito

## 12.6 test\_prenotazione

```
def test_prenotazione(self):
    #aggiunta_prenotazione
    GestioneStudenti.aggiungiUtente('Kristian', 'Likaj', '1097605', '12345678', datetime.date(2000, 10, 10))
    locale.setlocale(locale.LC_ALL, "it")
    adesso = datetime.datetime.now()
    oraInizio = datetime.time(hour=adesso.hour, minute=adesso.minute, second=adesso.second)
    oraFine = adesso + datetime.timedelta(minutes=30)
    GestionePrenotazioni.aggiungiPrenotazione("1097605", "Biblioteca Tecnico-Scientifica-Biomedica",
                                                datetime.date.today(), oraInizio, oraFine)
    self.assertIsInstance(GestionePrenotazioni.getPrenotazione("1097605"), Prenotazione)

    #rimozione_prenotazione
    GestionePrenotazioni.rimuoviPrenotazione("1097605")
    self.assertIsNone(GestionePrenotazioni.getPrenotazione("1097605"))
    GestioneStudenti.rimuoviUtente("1097605")
```

Figura 12.6: test Prenotazione

## **13. NOTE**

Il codice sorgente del software è disponibili sulla piattaforma GitHub al seguente indirizzo <https://github.com/FrancescoFinizii/Biblioteca-universitaria.git>. Attualmente il software è stato testato solamente sul sistema operativo Windows. Prima di avviare il software si consiglia di visionare il file README.txt situato nella repository del progetto.