**OBJECT – ORIENTED SOFTWARE DESIGN PROJECT**

Digital Library

2017-2018

Florio Stefano **246817 stefano.florio@student.univaq.it**

Gasbarro Giuseppe **246886 giuseppe.gasbarro@student.univaq.it**

Monte Enrico **246980 enrico.monte@student.univaq.it**

**SPECIFICA**

Una biblioteca digitale è uno spazio in cui mettere insieme collezioni, servizi e persone a supporto dell'intero ciclo di vita di creazione, uso, preservazione di dati, informazione e conoscenza. Lo scopo di questo progetto è di consentire la consultazione dei manoscritti che devono essere digitalizzati e che costituiscono un patrimonio bibliografico antico per un totale di 60.000 carte (ms. sec. XV-XIX) contenenti memorie storiche della città dell’Aquila.

Il sistema si divide nei seguenti **sottosistemi**:

1. Viewer

2. Uploader  
3. Transcriber  
4. Manager  
5. Administrator

I sottosistemi sono indipendenti tra loro ma comunicano al fine di realizzare i loro scopi. In particolare, vengono descritti come segue.

***Viewer***

Tale parte del sistema consente la consultazione delle opere digitali a utenti registrati. Consente la ricerca nel catalogo per *metadati* (descritti in seguito) oppure all’interno del testo della trascrizione. Le opere possono essere suddivisi in categorie. Appena si sceglie un’opera, verrà visualizzata con una schermata che avrà sulla destra il testo della trascrizione (se disponibile) e sulla sinistra l’immagine della pagina dell’opera che si sta visualizzando, sarà possibile sfogliare le pagine tramite un paginatore. Alcuni utenti con particolari privilegi possono effettuare il download dell’opera. L’utente può fare richiesta tramite un modulo per essere collaboratore del sistema (trascrittore). Gli utenti possono accedere al loro profilo personale dove saranno visualizzati i dati inseriti nella registrazione, tra cui: titolo di studio, professione, indirizzo, email, etc.

***Uploader***

Ogni opera è formata da più immagini (scansioni), ognuna delle quali rappresenta una pagina del manoscritto. Per ogni opera vengono caricati dei *metadati* (titolo, anno, ...). Al fine di rendere agevole il caricamento delle immagini e il successivo controllo, per ogni opera si vuole fornire la visualizzare sia tutte le pagine in miniatura e di una pagina per volta da scorrere con un paginatore. La digitalizzazione viene controllata da supervisori all’acquisizione per assicurarne la correttezza (ad esempio, in accordo con standard richiesti) e la qualità.

***Transcriber***

Ogni opera acquisita deve essere trasformata in un testo digitale, ciò avviene attraverso operazioni di trascrizioni in formato TEI (Text Encoding Initiative). Le trascrizioni sono digitate manualmente attraverso un text editor TEI integrato. Quindi ogni immagine (pagina) avrà il corrispondente testo associato. Più trascrittori possono lavorare sulla stessa pagina, è necessario sincronizzare le modifiche. Le trascrizioni sono oggetto di revisione da parte di revisori alle trascrizioni.

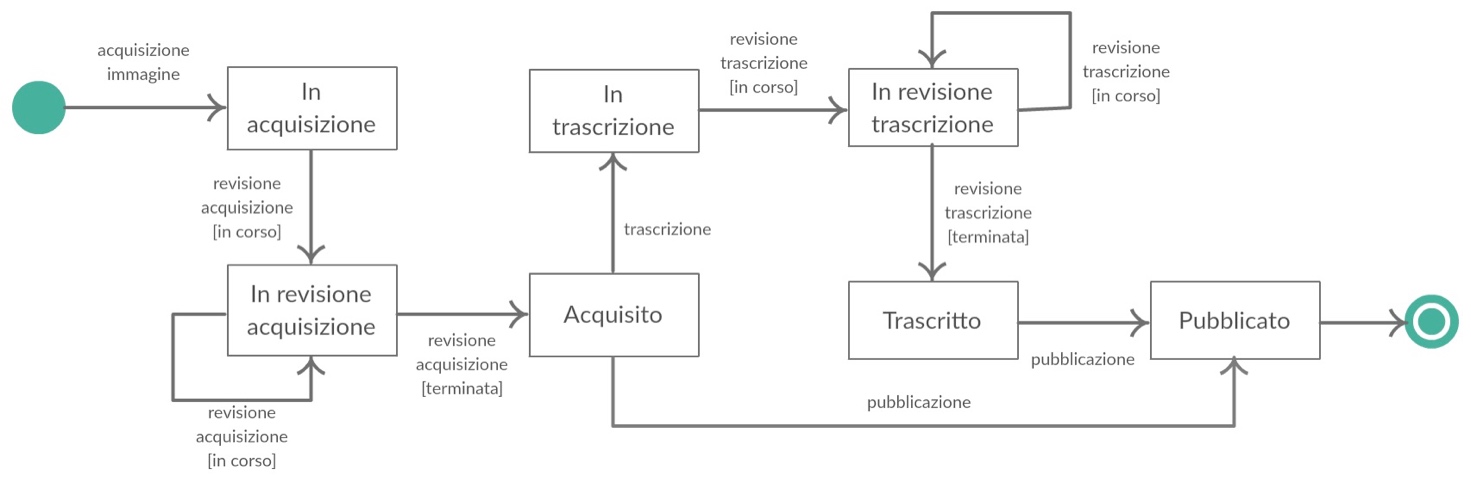
***Manager***

Questo sottosistema gestisce le assegnazioni, ovvero consente di assegnare parte di un’opera (1 o più immagini) a 1 o più trascrittori. Inoltre, consente di revisionare le trascrizioni concluse, di effettuare correzioni e validazione. E’ possibile anche riassegnare delle pagine ai trascrittori. Consente la pubblicazione delle trascrizioni e delle opere (solo immagini). Gestisce i livelli dei trascrittori (ogni trascrittore ha un livello 1-5 in base alla sua esperienza). Consente la supervisione dell’acquisizione immagini.

***Administrator***

Gestione in back-end di tutto il sistema: anagrafica utenti, opere, etc.

Processo di pubblicazione:



1. REQUIREMENTS
   1. **Documento dei requisiti**

* **Attori del sistema**

I primi attori identificati sono: l’**Utente Base** che può consultare l’elenco delle opere pubblicate, e l’**Utente Privilegiato** con il privilegio di poter consultare un’opera anche offline scaricandola in locale dal nostro sistema.

Successivamente troviamo l’**Acquisitore** che si occupa di acquisire nuove immagini ed associarle a pagine di un’opera,e il **Trascrittore** con il compito di trascrivere le immagini inserite dal primo.

Infine troviamo l’**Amministratore**, con le funzioni di gestione generale del sistema, quali gestione utente e gestione generale del sistema. Inoltre, l’amministratore può avere i privilegi di tutti gli attori sopra citati.

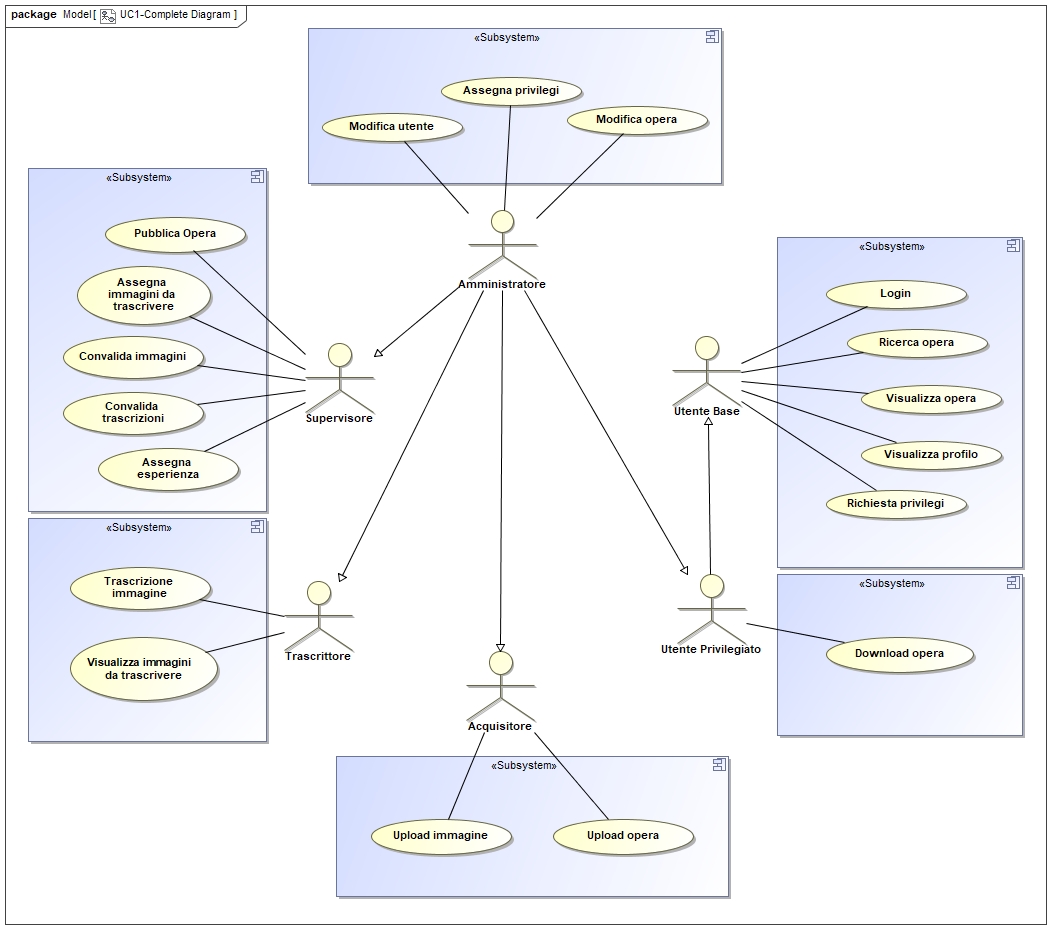
* **Requisiti Funzionali**

**<<Tra parentesi () la prioritizzazione del requisito in una scala da 1 a 5>>**

|  |  |
| --- | --- |
| ATTORE | DESCRIZIONE |
| ANONIMO | Il *visitatore anonimo* può loggarsi o registrarsi nel sistema. Se un utente non è loggato non può consultare le opere. (3) |
| UTENTE BASE | L’*utente* può effettuare una ricerca delle opere digitalizzate presenti nel sistema. (5) |
| L’*utente* può aprire e visualizzare le opere digitalizzate con la relativa trascrizione (se disponibile). (5) |
| L’*utente* può compilare un modulo per richiedere di diventare trascrittore/acquisitore/utente privilegiato. (3) |
| L’*utente* può visualizzare/modificare il proprio profilo. (1) |
| UTENTE PRIVILEGIATO | L’utente privilegiato può eseguire il download delle opere. (2) |
| ACQUISITORE | L’*acquisitore* può inserire nel sistema nuove opere e, per ognuna, associare dei metadati. (3) |
| L’*acquisitore* può fare l’upload delle immagini delle pagine mancanti di un’opera. (4) |
| TRASCRITTORE | Il *trascrittore* può trascrivere le immagini delle pagine caricate di un’opera attraverso un text editor TEI. (4) |
| SUPERVISORE | Il *supervisore* può associare una o più immagini di un’opera ad uno o più trascrittori. (4) |
| Il *supervisore* può assegnare esperienza ai trascrittori. (2) |
| Il *supervisore* può revisionare e convalidare immagini e trascrizioni. (4) |
| Il *supervisore* può pubblicare un’opera (trascritta o solo immagini). (3) |
| GUI | L’*interfaccia* deve fornire un User Experience diversa in base al tipo di utente che ha eseguito l’accesso nel sistema. (3) |
| SISTEMA | Il *sistema* deve essere provvisto di un text edito TEI integrato per la trascrizione manuale delle opere. (3) |
| AMMINISTRATORE | L’*amministratore* è in grado di gestire gli utenti e i relativi privilegi. (4) |

* **Requisiti Non Funzionali**

|  |  |
| --- | --- |
| TIPO | DESCRIZIONE |
| PERFORMANCE | Il sistema deve permettere le modifiche sincronizzate per le trascrizioni delle opere. Deve far si che più *trascrittori* possano lavorare sulla stessa pagina. |
| Il *sistema* deve essere in grado di contenere e gestire fino a 60.000 opere. |
| L’*interfaccia* deve permettere di avere una situazione sempre aggiornata delle opere digitalizzate. |
| Il *sistema* deve fornire una risposta all’utente in modo reattivo. |
| Scalability: Il *sistema* deve essere in grado di supportare eventuali modifiche ed espansioni funzionali future. |
| USABILITY | L’*interfaccia* (*user experience*), diversa in base al tipo di utente in attività, deve essere semplice ed intuitiva, per permettere a tutte le varietà di persone di utilizzare il sistema. (Anche persone senza competenze digitali). |
| AVAILABILITY | Il *sistema* dovrà essere sempre disponibile 24/24H e 7/7GG per permettere l’accesso e la consultazione delle opere a chiunque in qualunque momento. |
| REILABILITY | Il *sistema* deve essere affidabile, ovvero deve far sì che le immagini siano di qualità, e le trascrizioni debbano essere sempre coerenti all’opera e non contenere informazioni errate. |
| Il *sistema* deve avere un alto grado di robustezza in modo da resistere ad eventuali guasti ed evitare la perdita di dati ed opere. Quindi viene adottata una soluzione di fault-tolerance relativa alla duplicazione del database. |
| SECURITY | Il *sistema* deve essere protetto tramite un login dall’accesso riguardante fonti sconosciute e non autorizzate, per evitare di modificare un’opera inserendo trascrizioni o immagini inopportune. Inoltre deve prevedere un metodo che blocca l’accesso se sono state inserite credenziali non valide per più di 3 volte (in modo da evitare attacchi bruteforce). |

* 1. **Modello Use Case**
* Lo Use Case Diagram descrive le funzionalità che i vari tipi di utente possono effettuare. E’ importante specificare come, nel nostro sistema, gli utenti possono richiedere i permessi di trascrittore, acquisitore e privilegiato; a questo punto l’amministratore potrà accettare o rifiutare le richieste di permesso pervenute. Un utente in possesso di un permesso potrà usufruire di tutte le funzionalità offerte dal sottosistema collegato a quel permesso.
  + - * **Descrizione Use Case**

|  |  |
| --- | --- |
| Nome | **Login/Registrazione** |
| Attore | *Anonimo* |
| Descrizione | L’Utente anonimo può accedere o registrarsi nella biblioteca digitale tramite una form, inserendo i dati anagrafici.  Per usufruire dei servizi della libreria digitale è richiesto un account. |
| Condizione Di Entrata | L’utente visita la digital library |
| Risultato | Login o Registrazione riuscita/non riuscita |

|  |  |
| --- | --- |
| Nome | **Ricerca Opera** |
| Attore | *Utente Base* |
| Descrizione | L’Utente Base ricerca un’opera, se esistente, tramite l’inserimento di una o più parole chiave nell’apposita barra di ricerca.  L’Utente base vedrà l’elenco delle opere cercate, selezionandone una, chiama l’UC “Visualizza Opera”. |
| Condizione Di Entrata | L’Utente base ha cliccato sulla barra di ricerca. |
| Risultato | Visualizzazione elenco delle opere cercate che fanno match con le parole chiave inserite. |

|  |  |
| --- | --- |
| Nome | **Visualizza profilo** |
| Attore | *Utente Base* |
| Descrizione | L’Utente registrato accede al proprio pannello utente tramite l’apposito bottone, e visualizza il proprio profilo con la relativa anagrafica.  Eventualmente può modificare i propri dati. |
| Condizione Di Entrata | L’Utente base clicca sull’apposito bottone relativo al proprio profilo. |
| Risultato | Visualizzazione/modifica dati anagrafici. |

|  |  |
| --- | --- |
| Nome | **Visualizzazione Opera** |
| Attore | *Utente Base* |
| Descrizione | Cliccando su di un’opera verranno visualizzate tutte le sue pagine in miniatura, alla selezione di una, verrà riprodotta l’immagine a sx e, se presente, la rispettiva trascrizione a dx.  L’Utente base può eventualmente scorrere le pagine tramite un paginatore. |
| Condizione Di Entrata | L’Utente base clicca su un’opera. |
| Risultato | Visualizzazione delle opere digitalizzate. |

|  |  |
| --- | --- |
| Nome | **Download Opera** |
| Attore | *Utente Privilegiato* |
| Descrizione | Un’utente privilegiato può scaricare un’opera in locale tramite l’apposito bottone. |
| Condizione Di Entrata | L’Utente privilegiato sta visualizzando un’opera (UC “Visualizza Opera”). |
| Risultato | Opera scaricata in locale sulla macchina dell’utente privilegiato. |

|  |  |
| --- | --- |
| Nome | **Upload Opera** |
| Attore | *Acquisitore* |
| Descrizione | L’acquisitore clicca sull’apposito bottone per inserire una nuova opera, inserendone codice, titolo, descrizione, anno, categoria, lingua e autore. |
| Condizione Di Entrata | L’Acquisitore è loggato nel sistema. |
| Risultato | E’ stata inserita una nuova opera nel database. |

|  |  |
| --- | --- |
| Nome | **Richiesta Privilegi** |
| Attore | *Utente Base* |
| Descrizione | L’Utente base clicca sull’apposito bottone, e gli viene fornito un form da compilare per diventare trascrittore/acquisitore/downloader selezionando il privilegio che si vuole richiedere. |
| Condizione Di Entrata | L’utente base è loggato nel sistema. |
| Risultato | La richiesta arriva all’amministratore. |

|  |  |
| --- | --- |
| Nome | **Convalida Trascrizione – Convalida Digitalizzazione** |
| Attore | *Supervisore* |
| Descrizione | Il supervisore clicca sul bottone per visualizzare le trascrizioni/immagini acquisite non ancora revisionate; in seguito ne può selezionare una, controllando che l’immagine/trascrizione in questione rispetti gli standard (qualità, assenza di errori, contenuti corretti);  Il supervisore preme il bottone per convalidare o meno la digitalizzazione/trascrizione. |
| Condizione Di Entrata | Il supervisore è loggato nel sistema. |
| Risultato | Trascrizione/immagine validata o respinta. |

|  |  |
| --- | --- |
| Nome | **Assegna Esperienza** |
| Attore | *Supervisore* |
| Descrizione | Il supervisore assegna punti esperienza ad un trascrittore tramite l’apposito bottone. L’esperienza varia in una scala da 1 (minima) a 5 (massima). |
| Condizione Di Entrata | Il supervisore apre la sezione relativa ai trascrittori e clicca sul trascrittore relativo. |
| Risultato | L’esperienza dei trascrittori viene modificata. |

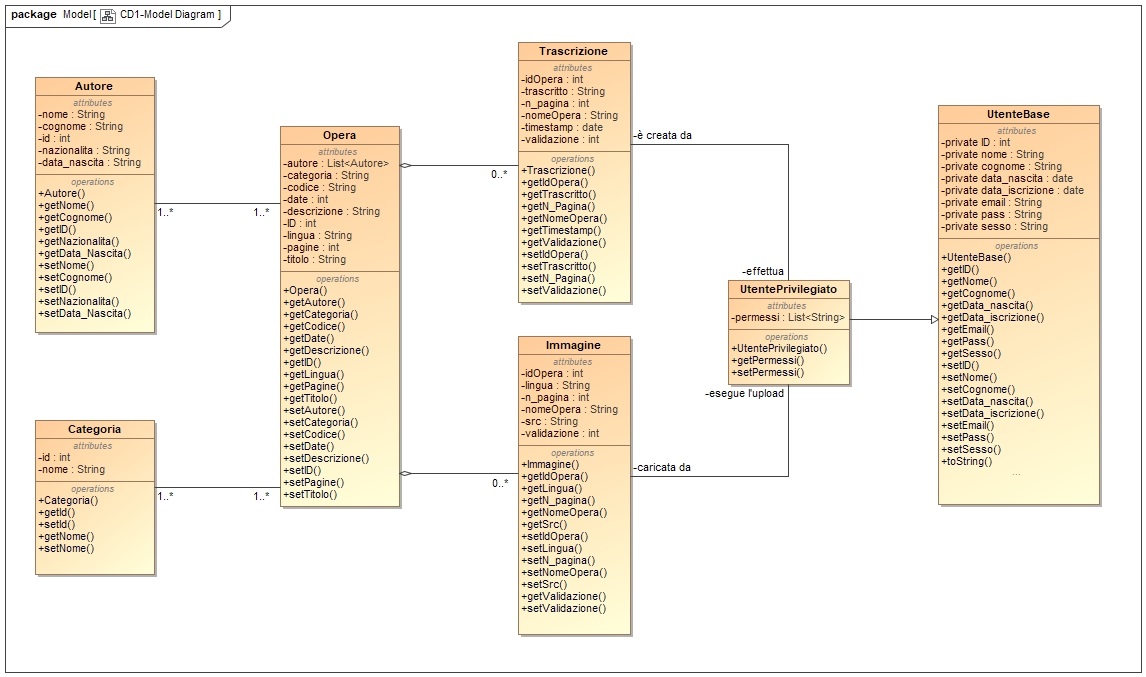
|  |  |
| --- | --- |
| Nome | **Assegna Immagini da Trascrivere** |
| Attore | *Supervisore* |
| Descrizione | Il supervisore visualizza pagine con immagini non ancora trascritte e le assegna ad uno o più trascrittori. |
| Condizione Di Entrata | Il supervisore è loggato nel sistema. |
| Risultato | Il supervisore ha assegnato una o più immagini a uno o più trascrittori. |

|  |  |
| --- | --- |
| Nome | **Visualizza Immagini da Trascrivere/Trascrizione Immagine** |
| Attore | *Trascrittore* |
| Descrizione | Il trascrittore preme sull’apposito bottone per visualizzare le immagini da trascrivere a lui associate; selezionandone una, si apre text editor TEI integrato con il quale può trascrivere testualmente l’immagine; alla fine preme il bottone per mandare la trascrizione in attesa di essere verificata. |
| Condizione Di Entrata | Il trascrittore sta visualizzando la pagina dell’opera a lui assegnata. |
| Risultato | L’immagine è stata trascritta; la trascrizione viene inserita nelle trascrizioni da revisionare. |

|  |  |
| --- | --- |
| Nome | **Pubblicazione Opera** |
| Attore | *Supervisore* |
| Descrizione | Il supervisore visualizza le opere che hanno almeno tutte le pagine con l’immagine acquisita e validata; può selezionarne una e pubblicarla. |
| Condizione Di Entrata | Il supervisore si accerta che l’opera è stata completamente digitalizzata e revisionata. |
| Risultato | Opera pubblicata e resa visibile agli utenti. |

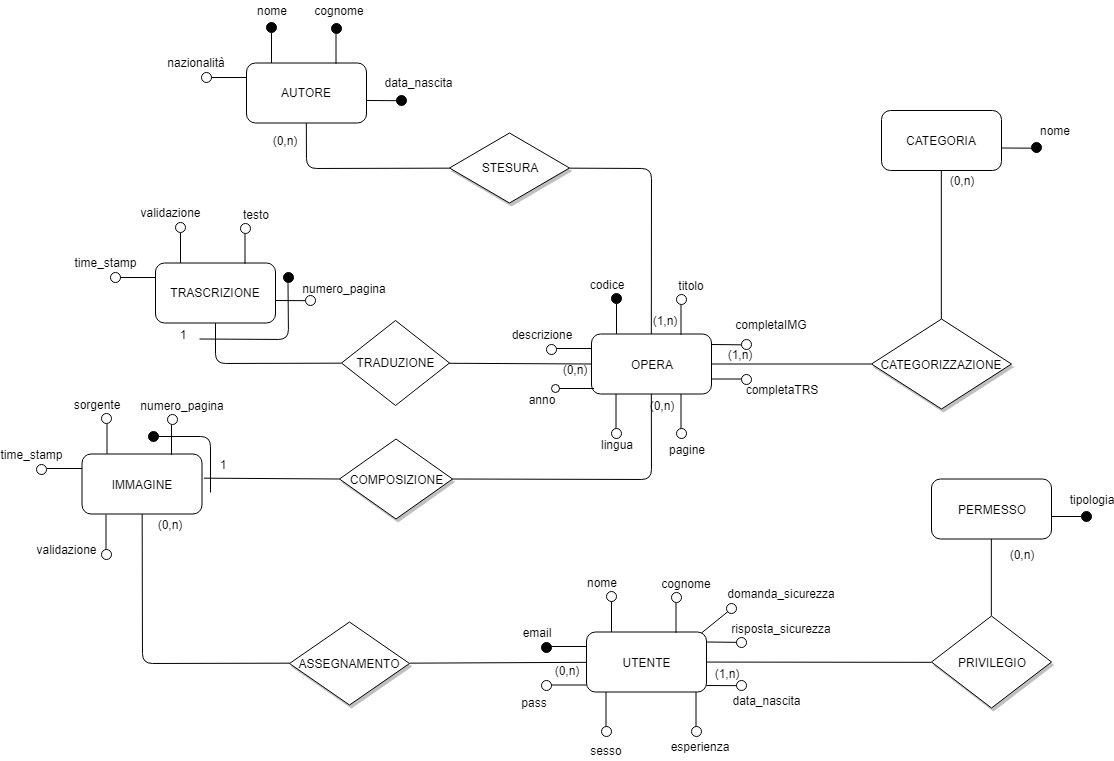
|  |  |
| --- | --- |
| Nome | **Assegna Privilegi** |
| Attore | *Amministratore* |
| Descrizione | L’Amministratore visualizza le richieste di privilegi effettuate dagli utenti, e tramite un bottone decide se accettarle o meno. |
| Condizione Di Entrata | Il supervisore è loggato nel sistema. |
| Risultato | L’Amministratore ha valutato e gestito le richieste di privilegi. |

* 1. **Modello Di Dominio**

****

* La classe *Opera* è collegata da associazioni di **aggregazione** a *Trascrizione* e Immagine con una cardinalità 0..\*; ciò significa che ogni opera sarà composta da un numero di immagini e trascrizioni che all’inizio è 0, e poi man mano con l’attività degli utenti **le pagine verranno inserite e validate** fino a che l’opera sarà **completa**. Inoltre la classe *Opera*, per completare la sua implementazione, ha bisogno delle classi *Autore* e *Categoria*.
* I **permessi** sono modellati come una lista di stringhe per la classe *UtentePrivilegiato*, la quale estende la classe più generica di *UtenteBase*.

**1.3.1 Diagramma ER**

****

* 1. **Analisi Di Classi Entity, Boundary e Controller**

**Classi Entity**: Rappresentano le entità “persistenti” nel sistema.

-Utente:

* Base
* Privilegiato
* Acquisitore
* Trascrittore
* Supervisore
* Amministratore

-Opera

-Immagine

-Trascrizione

**Classi Boundary**: Rappresentano le classi di interazione tra l'utente e il sistema.

-ProfileButton: bottone per la visualizzazione del profilo utente;

-GetOperaButton: bottone per la visualizzazione dell’opera selezionata;

-UploadButton: bottone per l’upload delle immagini;

-SearchButton: bottone per la ricerca delle opere;

-DownloadOpera: bottone per il download dell’opera;

-RichiestaButton: bottone per la richiesta di privilegi;

-RequestsButton: bottone per la visualizzazione delle richieste di privilegi;

-TranscriberButton: bottone per iniziare la trascrizione di una pagina;

-TEI: editor di testo per la trascrizione di una pagina;

-AssegnazioneButton: bottone per l’assegnazione di una trascrizione;

-CheckButton: bottone per la richiesta delle opere da convalidare;

-ConvalidateButton: bottone per la convalida si un’opera;

**Classi Controller**: Rappresentano classi che coordinano classi entity e classi boundary. Ad ogni classe boundary è associata una classe Controller:

-ProfileController: controller responsabile della visualizzazione dei dati del profilo utente;

-GetOperaController: controller responsabile della visualizzazione dell’opera selezionata;

-UploadController: controller responsabile dell’Upload delle immagini;

-SearchController: controller responsabile della visualizzazione di una lista di opere data una stringa di ricerca;

-DownloadController: controller responsabile del download di un’ opera;

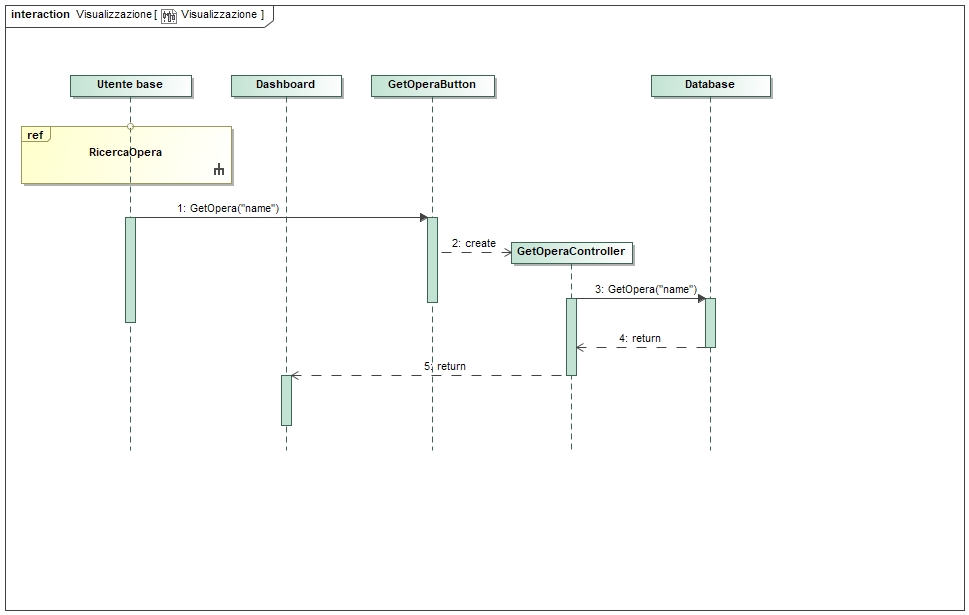
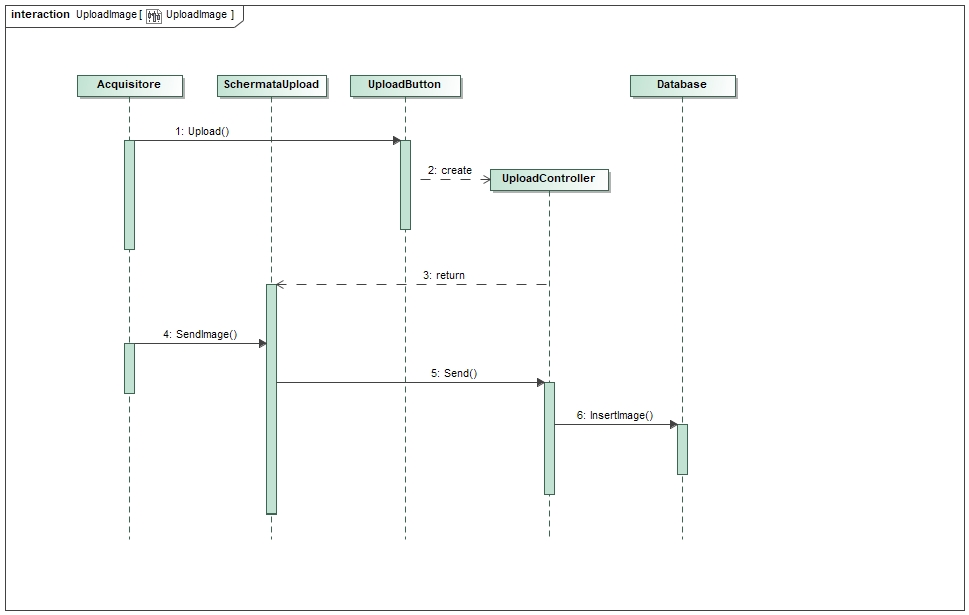
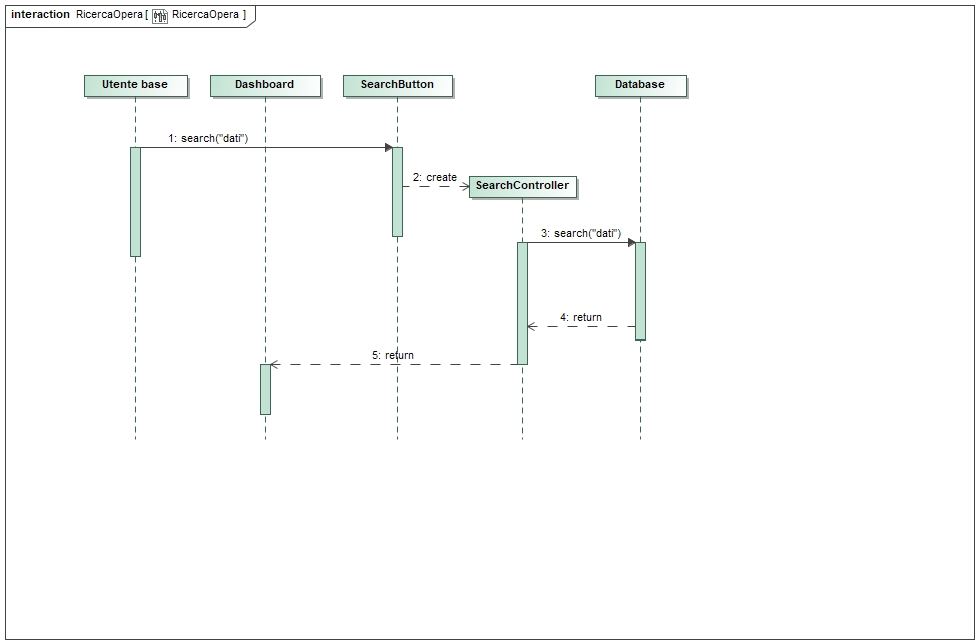
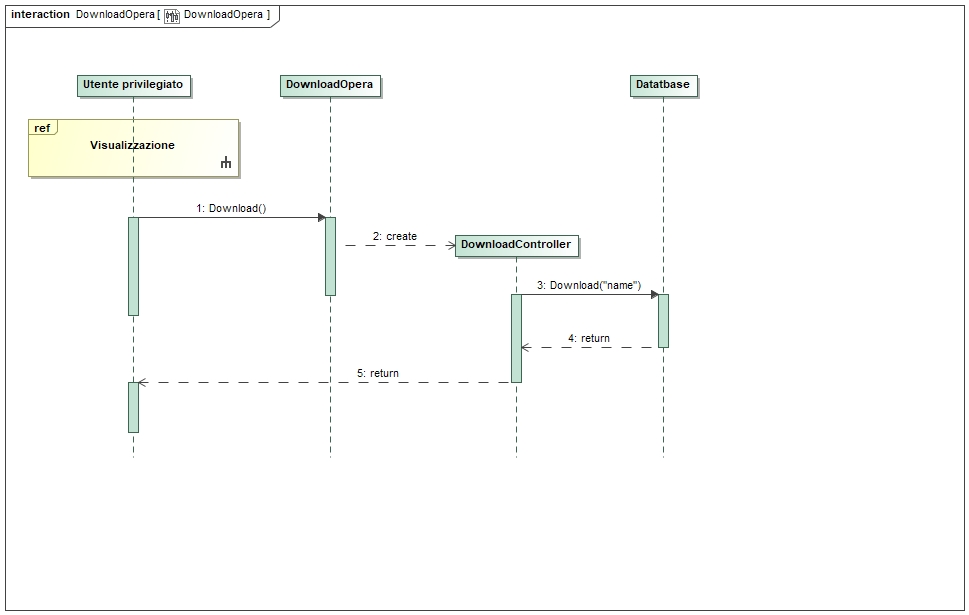
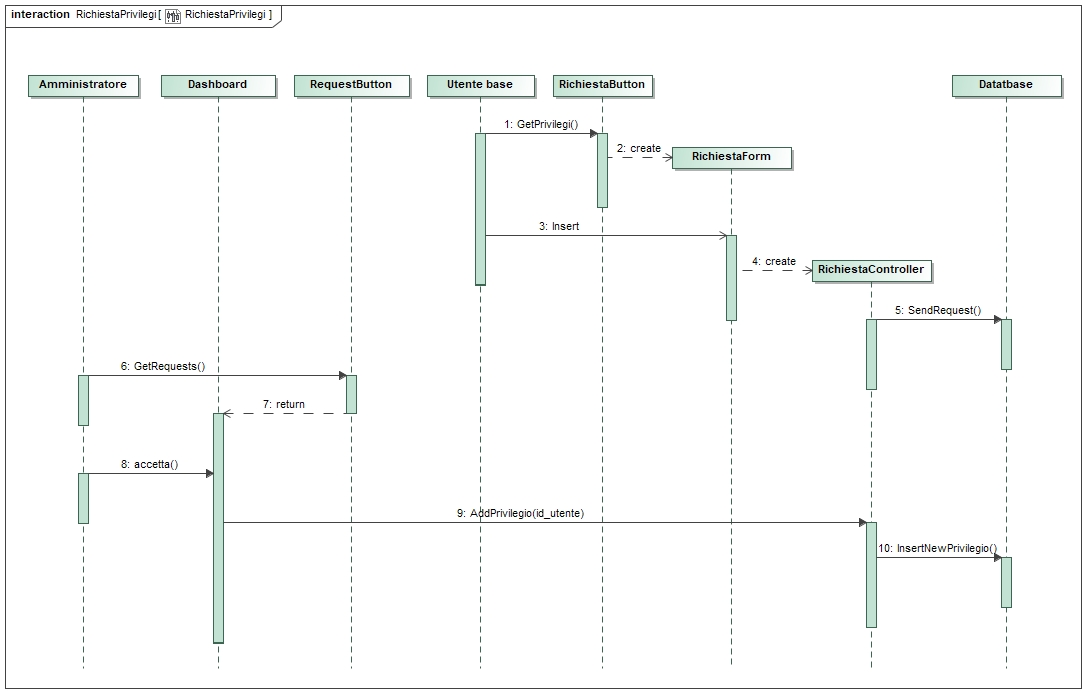
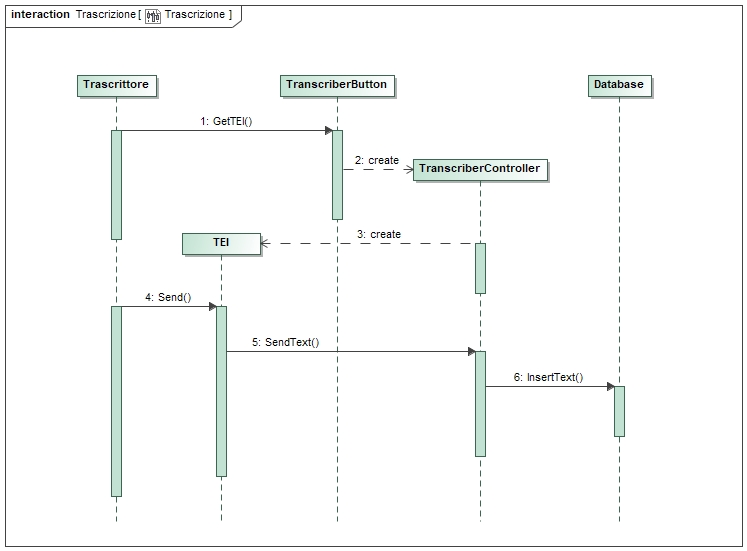
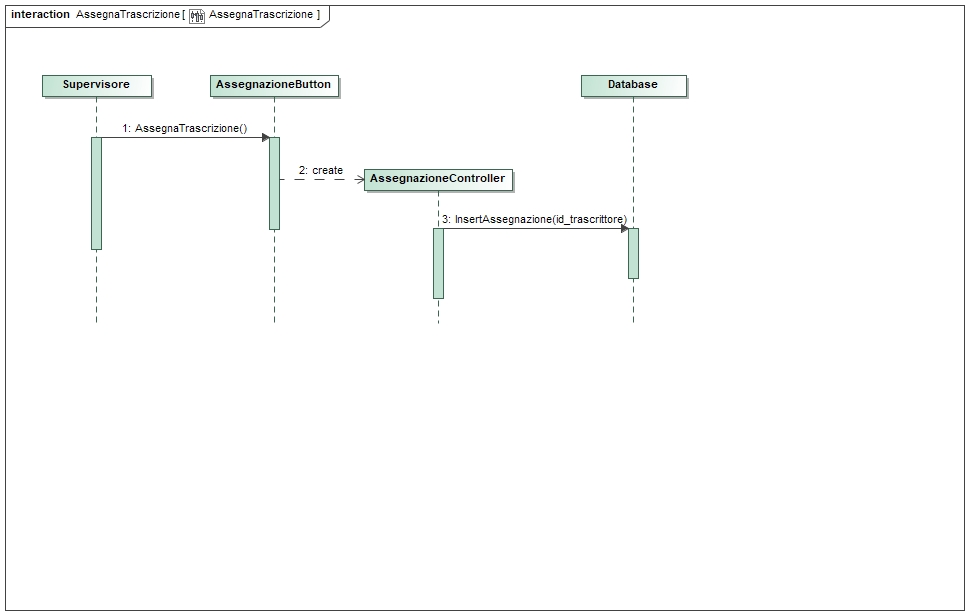
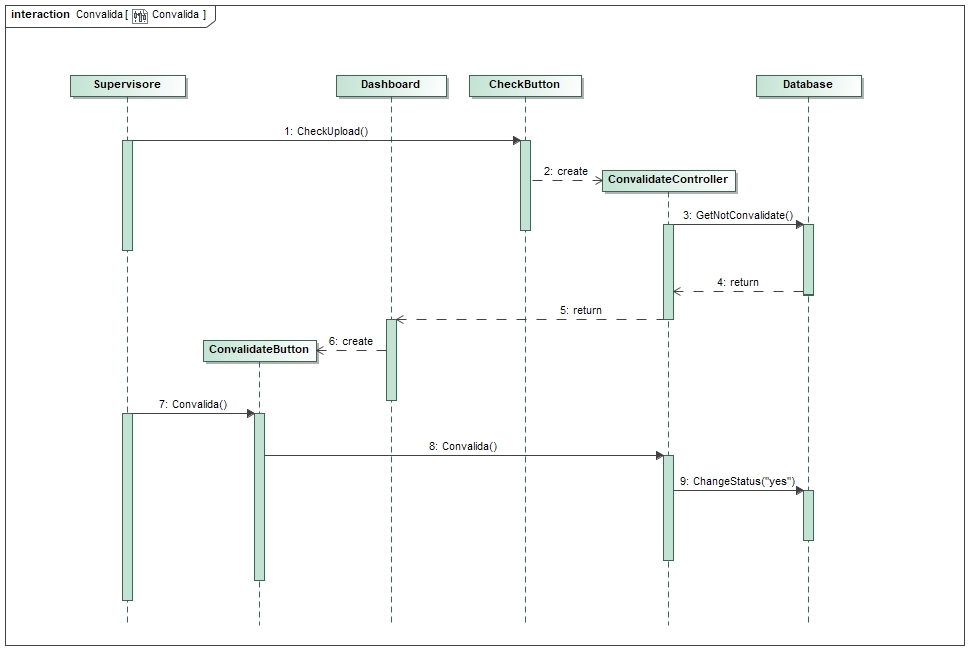
-RichiestaController: controller responsabile della richiesta di privilegi o della visualizzazione degli utenti che hanno richiesto un privilegio;

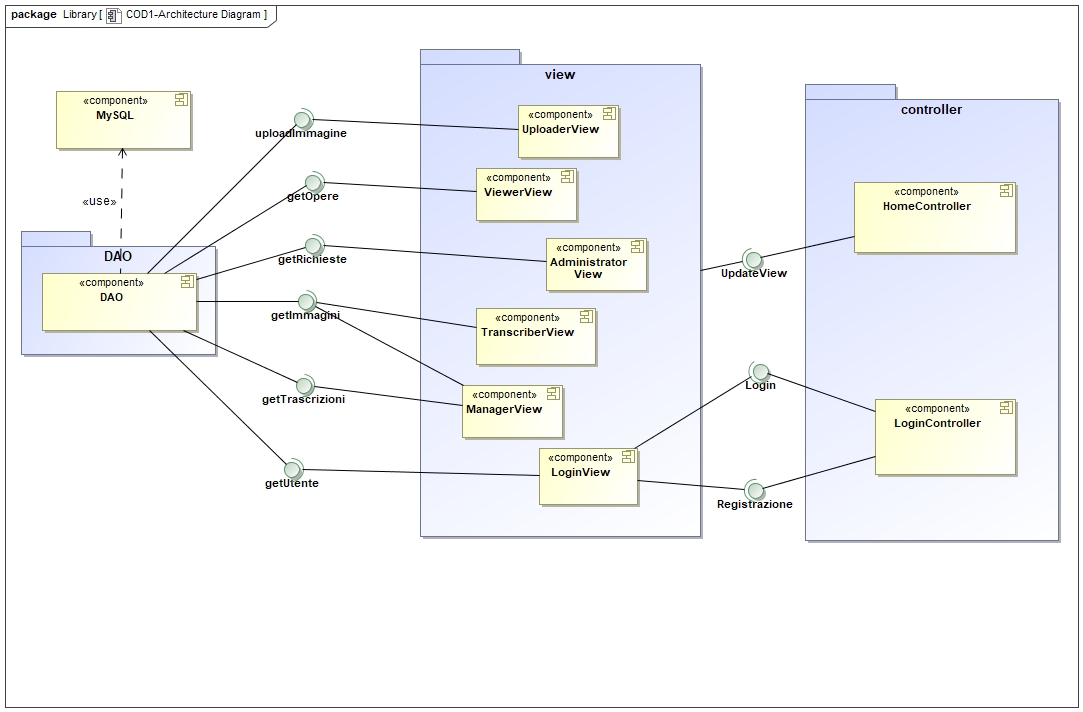
-TranscriberController: controller responsabile della creazione del TEI e della trascrizione di una pagina;

-AssegnazioneController: controller responsabile dell’ assegnazione di una trascrizione da parte di un supervisore;

-ConvalidateController: controller responsabile della visualizzazione delle opere da convalidare e della convalida di un’opera.

* + 1. **Sequence Diagrams**

1. Visualizzazione Profilo
2. Visualizzazione Opere
3. Upload Immagini
4. Ricerca Opere
5. Download Opere
6. Richiesta Privilegi
7. Trascrizione Immagini
8. Assegna Trascrizione
9. Convalida Opere
10. SYSTEM DESIGN

**2.1 Modello Architettura software**

**2.2 Descrizione dell’Architettura**

Il nostro sistema è basato su **interfacce**, le quali racchiudono varie funzionalità diverse e sono accessibili **a seconda dei permessi che l’utente possiede**. Per questo motivo abbiamo utilizzato il pattern **MVC**, il quale ci consente di **separare** la logica dalle interfacce e dalle classi di dominio. Infatti ogni input che l’utente immette viene raccolto dal relativo controller collegato alla view, il quale la aggiornerà dopo aver raccolto i nuovi dati dal model.

Il model, però, verrà prima popolato con oggetti istanziati dal **DAO**. Il DAO, infatti, contiene una serie di metodi che **interrogano** il database e raccolgono dati con i quali **istanzierà gli oggetti** che il controller richiede per popolare la view.

Le **funzionalità** offerte dal sistema si trovano nei component *Uploader*, *Viewer*, *Administrator*, *Transcriber* e *Manager*, le quali sono usufruibili dall’utente solo se possiede il permesso associato.

Il sistema è diviso in diversi package:

- **application:** contiene il Main per l’avvio dell’applicativo;

- **model:** contiene le classi che costituiscono la parte statica del sistema; le più importanti sono UtentePrivilegiato che estende UtenteBase con una lista di permessi che ne implicano le funzionalità accessbili, le classi Immagine e ImmagineTrascrizione le quali rappresentano le immagini e le trascrizione delle pagine di un’opera;

- **view:** contiene tutte le interfacce, prive di logica;

**- controller:** la parte centrale del sistema, che si occupa di gestire gli input dell’utente sulla view, di aggiornarle e di caricarle; gestisce l’utente loggato e chiama i metodi del DAO per effettuare modifiche sul database e prelevare dati;

**- resources:** contiene le immagini delle pagine; ogni opera avrà una cartella con nome il codice dell’opera, e conterrà le pagine chiamate col numero pagina corretto;

**- DAO:** contiene le classi aventi una serie di metodi che eseguono l’accesso al database e raccolgono i dati richiesti dalle interfacce per essere popolate.

**2.3 Descrizione delle Scelte e delle Strategie Adottate**

* Abbiamo modellato gli utenti in tre classi: *UtenteBase*, *UtentePrivilegiato e Trascrittore*;

la prima rappresenta un utente registrato **privo di qualsiasi permesso**, e che quindi potrà solo **visualizzare il proprio profilo** e **consultare i manoscritti pubblicati**.

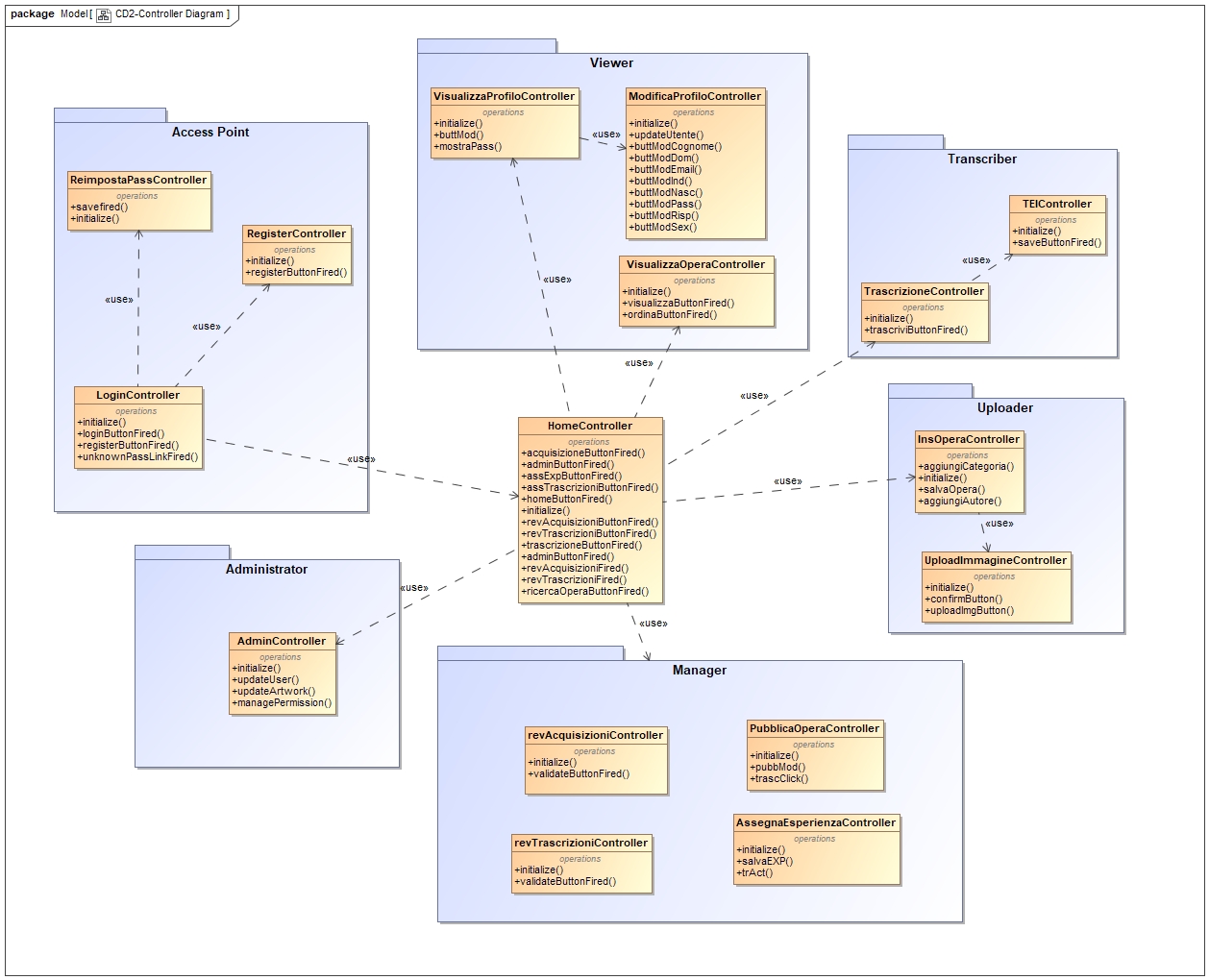
Il secondo avrà una **lista di permessi** attraverso la quale il sistema capirà quale funzionalità rendere accessibili. Il terzo sarà un’estensione del secondo avendo una lista di privilegi ed in più un indicatore di **esperienza acquisita**.

Tuttavia abbiamo pensato che i *supervisori* e gli *amministratori* siano utenti preimpostati e quindi i loro permessi non siano richiedibili. Infatti un utente potrà richiedere solo i permessi di:

* + **trascrittore**
  + **downloader**
  + **uploader**

Questi permessi sono richiedibili nella sezione *Modifica Profilo* presente in *Visualizza Profilo.*

* Per quanto riguarda le trascrizioni eseguibili in parallelo, abbiamo deciso di mostrare al trascrittore che apre l’immagine a lui assegnata **l’ultima trascrizione eseguita** su quella pagina, così eventualmente da **modificarla ed aggiornarla**. A quel punto un altro trascrittore che accederà a quella pagina vedrà le modifiche effettuate e così via.

**3.1 Object Design Model**

- Nel diagramma viene descritto il comportamento dell’applicazione **FXML** e dei controller che la compongono; le funzionalità di ogni **sotto-sistema** sono offerte dai controller al suo interno, i quali aggiornano la view associata e al tempo stesso sfruttano il **DAO** per aggiornare o prelevare i dati.

**3.2 Design Decision**

* **Database Relazionale:** Il team dopo un’attenta valutazione ha deciso di adottare un DB relazionale, in particolare MYSQL 8.0, attualmente **MySQL 8.0**, con 50 utenti collegati, è in grado di sostenere un massimo di 600.000 richieste al secondo, valore ampiamente valido per i nostri scopi.
* **Java FX:** Per quanto riguarda lo sviluppo della grafica del sistema abbiamo scelto JAVA FX in quanto si sposa con il linguaggio di sviluppo java essendo basato su quest’ultimo, ed in combinazione con esso permette di creare applicazioni all’avanguardia. JavaFX oltre ad una grande libreria di funzionalità grafiche, utilizza un vero e proprio linguaggio chiamato “java FX script” orientato alla programmazione grafica e rende lo sviluppo di applicazioni grafiche particolarmente agevolato. Inoltre per la realizzazione delle interfacce abbiamo usato il tool “Scene Builder” che evita la creazione manuale di un file FXML e permette costruirle con semplici drag and drop. Inoltre supporta i css e integra un visualizzatore FXML il quale ci consente di visualizzare un’anteprima durante la creazione del progetto. Infine permette anche di aprire file FXML esistenti e modifcarli senza bisogno di scrivere codice.

**4 Appendix Prototype**

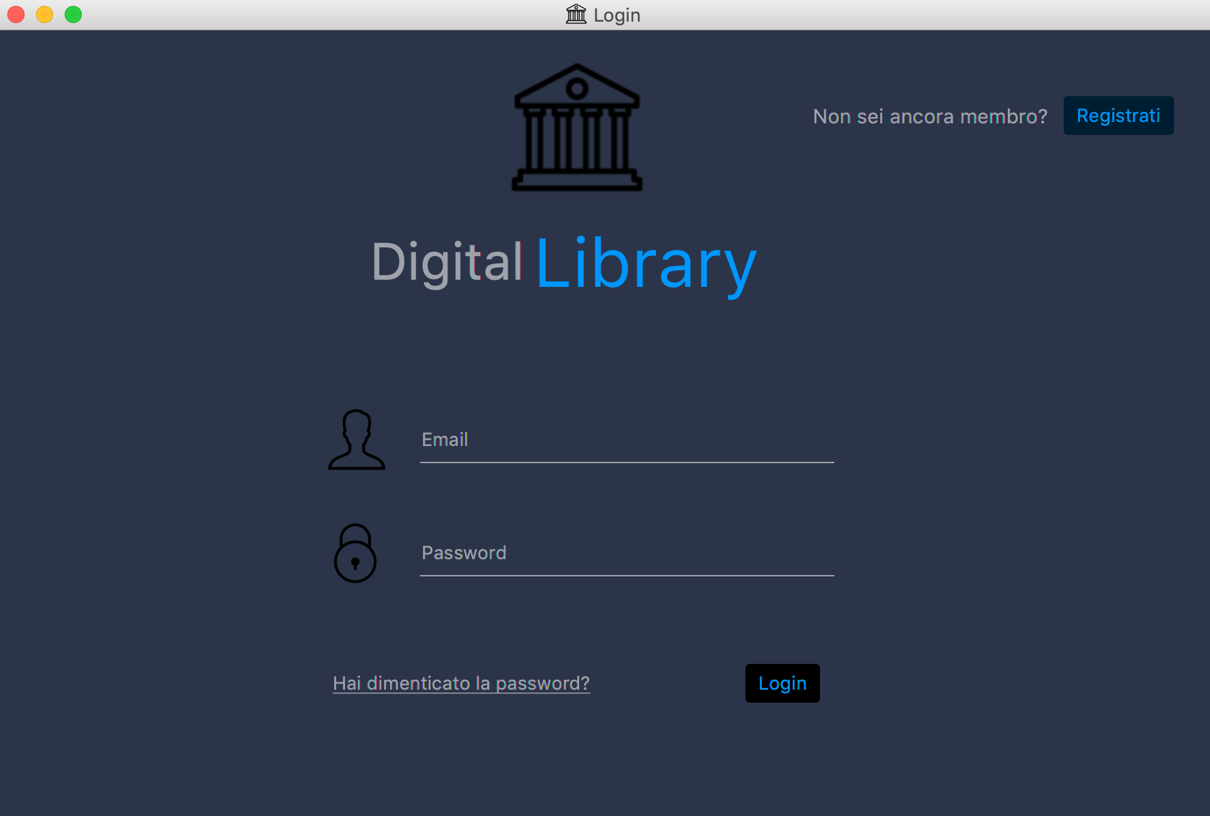
* **Interfaccia Login**

Fig. 1: all’apertura del prototipo si presenterà la schermata di login; qui è possibile accedere al sistema, registrarsi o, eventualmente, recuperare la password nel caso fosse stata dimenticata.

*  **Interfaccia Home**

Fig. 2: nella schermata principale(Home) l’utente può vedere le ultime opere aggiunte, e navigare tra le varie voci nel menù (In base al tipo di utente in carica). Inoltre, può ricercare un’opera in base alle proprie esigenze.

*  **Interfaccia Visualizzazione Opere**

Fig. 3: Quando un’opera viene aperta vediamo sulla sinistra le immagini estese delle pagine, sulla destra la trascrizione della pagina (se disponibile). In basso possiamo osservare tutte le pagine dell’opera in miniatura. Inoltre se l’utente corrente gode del privilegio può cliccare sul tasto di download per scaricare l’opera in pdf in locale.

* **Immagine che contiene screenshot, monitor

  Descrizione generata automaticamenteInterfaccia Ricerca Opere**

Fig. 4: la ricerca delle opere produce una view con l’elenco delle opere che fanno match con la ricerca, mostrando titolo ed autori. Cliccando su un’opera si aprirà l’interfaccia precedente.

* **Immagine che contiene screenshot, monitor

  Descrizione generata automaticamenteInterfaccia Acquisizioni**

Fig. 5: la schermata di acquisizione permette di scegliere l’opera e la relativa pagina che si vuole digitalizzare e tramite lo chooser prendiamo un’immagine dalla macchina dell’utente e la carichiamo nel sistema. Questa immagine dovrà essere poi convalidata dal supervisore.

* **Interfaccia Convalida Immagini/Trascrizioni**

**Immagine che contiene screenshot, monitor

Descrizione generata automaticamente**

Fig. 6: Qui possiamo scegliere se convalidare o meno un’immagine in attesa, cliccando su “trascrizioni” possiamo eseguire lo stesso compito sulle trascrizioni in attesa. Cliccando su una voce si apre l’immagine relativa così da poter visionarla e verificare la qualità e gli standard.

* **Immagine che contiene screenshot, monitor

  Descrizione generata automaticamenteInterfaccia Trascrizioni Assegnate**

Fig. 7: qui possiamo vedere le trascrizioni che sono state assegnate all’utente che ha fatto l’accesso.

* **Immagine che contiene screenshot

  Descrizione generata automaticamenteInterfaccia Trascrizione**

Fig. 8: la trascrizione avviene tramite un editor di testo in formato TEI. Cliccando su salva la trascrizione viene inserita nel DB e può essere poi ripresa ed eventualmente modificata da altri trascrittori che hanno avuto il compito di lavorare sulla stessa.

* **Immagine che contiene screenshot, monitor, schermo, interni

  Descrizione generata automaticamenteInterfaccia Inserimento Opere**

Fig. 9: per inserire un’opera è necessario aggiungere uno o più autori, una o più categorie e infine i metadati dell’opera e cliccare su salva.