Sara Di Berardino 247976 Davide Simboli Alessio Morgante 246991

[Selezionare la data]

***Biblioteca Digitale***

***Documento dei requisiti***

# Obiettivi

Sviluppare una biblioteca digitale per consentire la consultazione dei manoscritti che devono essere digitalizzati e che costituiscono un patrimonio bibliografico antico per un totale di 60000 carte, secolo XV-XIX, contenenti memorie storiche della città dell’Aquila.

# Requisiti

## Viewer:

1. Registrazione dell’utente;
2. Login;
3. Logout;
4. Consultazione delle opere digitali a utenti registrati;
5. Ricerca nel catalogo per metadati oppure all’interno del testo della trascrizione;
6. Visualizzazione dell’opera che verrà visualizzata con una schermata che avrà sulla destra il testo della trascrizione e sulla sinistra l’immagine della pagina dell’opera che si sta visualizzando;
7. Possibilità di sfogliare le pagine;
8. Possibilità per gli utenti privilegiati di effettuare il download dell’opera;
9. Possibilità per gli utenti di fare richiesta tramite un modulo per essere collaboratore di sistema ( trascrittore );
10. Accesso degli utenti al proprio profilo personale;
11. Possibilità da parte degli utenti di modificare dati del proprio profilo personale.

## Uploader:

1. Caricamento delle immagini dell’opera da parte dell’acquisitore;
2. Caricamento dei metadati dell’opera (titolo, anno, autore, n° pagine totali ) da parte dell’acquisitore;
3. La digitalizzazione viene controllata da supervisori all’acquisizione per assicurarne la correttezza e la qualità;
4. Pubblicazione dell’opera ( solo immagini ) da parte del supervisore.

## Transcriber:

1. Trascrizione digitale delle opere da parte di trascrittori;
2. Revisione delle trascrizioni da parte di revisori alle trascrizioni;
3. Pubblicazione delle trascrizioni da parte dei revisori alle trascrizioni;
4. Sincronizzazione delle modifiche.

## Manager:

1. Assegnazione di parte di un’opera a 1 o più trascrittori;
2. Consenso a revisionare le trascrizioni concluse, con correzione e validazione;
3. Riassegnazione delle pagine ai trascrittori;
4. Consenso alla pubblicazione delle trascrizioni e delle opere;
5. Gestione dei livelli dei trascrittori;
6. Consenso alla supervisione dell’acquisizione delle immagini;
7. Accetta/Rifiuta richieste per diventare trascrittore.

## Administrator:

1. Gestione in back-end del sistema.

# Vincoli

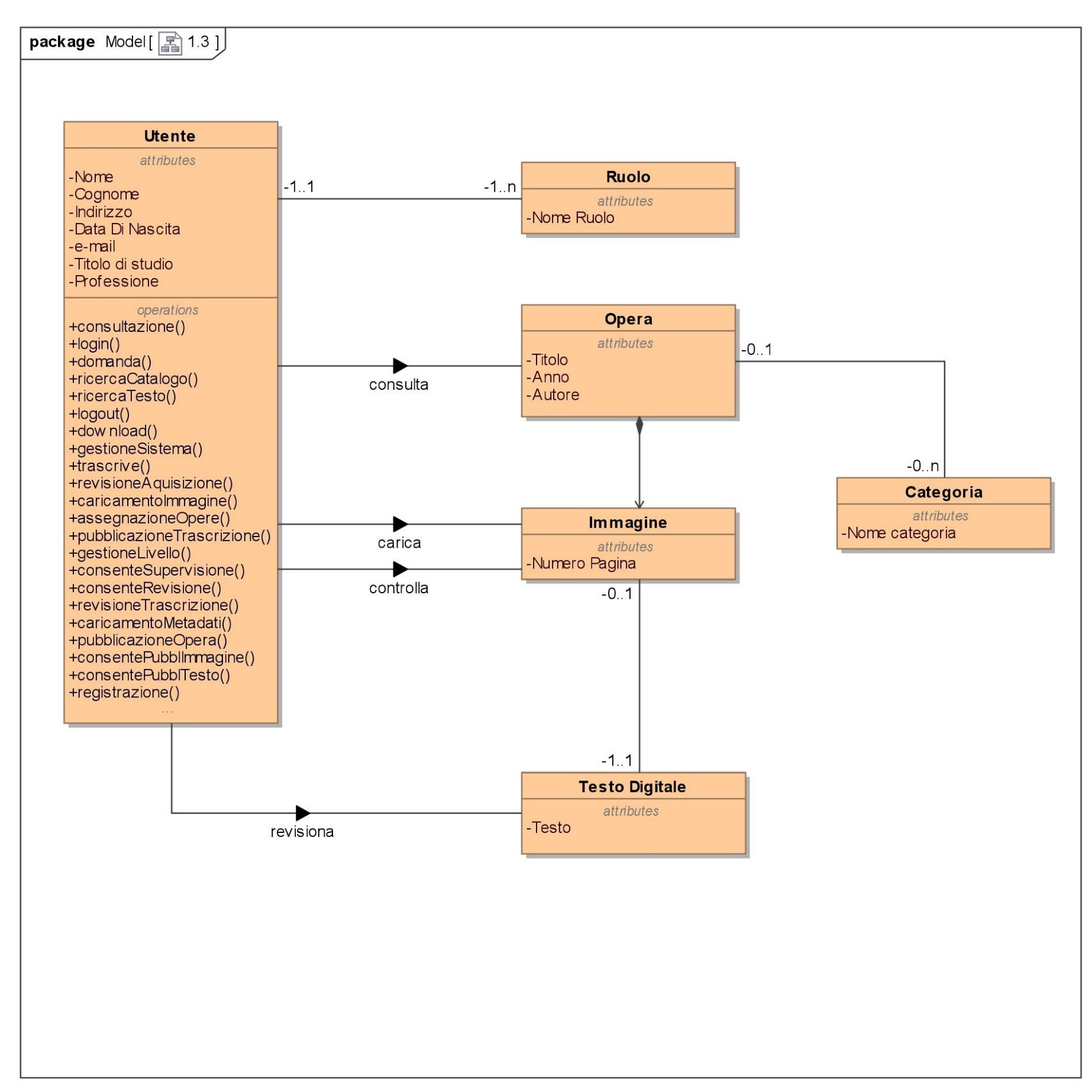
* Nella registrazione non è possibile usare il ruolo dell’amministratore. Esso è già nel sistema;
* Nel profilo personale gli utenti possono modificare solo nome, cognome, data di nascita indirizzo, titolo di studio, professione e password;
* Nel profilo personale vengono visualizzati i dati della registrazione: nome, cognome, data di nascita, email, titolo di studio, professione, indirizzo, password ( criptata ) e ruolo;
* Nel profilo personale gli utenti possono modificare solo nome, cognome, data di nascita, indirizzo, titolo di studio, professione, password;
* Solo l’utente base e l’utente privilegiato possono diventare trascrittori;
* Se si diventa trascrittori si perde il ruolo precedente;
* La richiesta per diventare trascrittori può essere mandata una sola volta fino a quando non si riceve una risposta. Se viene accettata si diventa trascrittori, se viene rifiutata può essere mandata di nuovo;
* Le opere possono essere divise per categorie;
* L’acquisitore può aggiungere le categorie se e solo se non sono già presenti nel database;
* Ogni opera è formata da più immagini, ognuna delle quali rappresenta una pagina del manoscritto;
* È possibile sfogliare le pagine dell’opera attraverso un paginatore;
* In fase di caricamento e di revisione delle immagini si da la possibilità di avere un riepilogo di esse attraverso una visualizzazione in miniatura e con un paginatore;
* Si possono acquisire solo 6 pagine alla volta;
* La pubblicazione delle opere può avvenire solo se tutte le immagini delle stesse sono state acquisite e revisionate;
* La trascrizione dell’opera in testo digitale avviene attraverso operazioni di trascrizione in ***formato TEI(togliere??)***. Le trascrizioni sono digitate manualmente attraverso un text editor integrato;
* Più trascrittori possono lavorare sulla stessa pagina;
* Ogni trascrittore ha un livello 1-5 in base alla propria esperienza;
* I revisori di trascrizioni possono inserire dei commenti nelle trascrizioni prima di rimandarle ai vari trascrittori;
* Se disponibile viene visualizzato sulla sinistra il testo della trascrizione e a destra l’immagine dell’opera che si sta visualizzando, altrimenti, se il testo della trascrizione non è disponibile viene visualizzata solo l’immagine dell’opera.
* Se il manager nega il consenso alla supervisione l’acquisitore dovrà acquisire di nuovo quella pagina. Altrimenti, se il consenso è ammesso l’immagine passa al supervisore.

***Use Case Diagram***

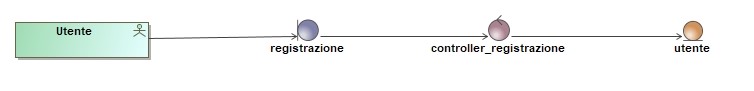
**Immagine che contiene testo, mappa

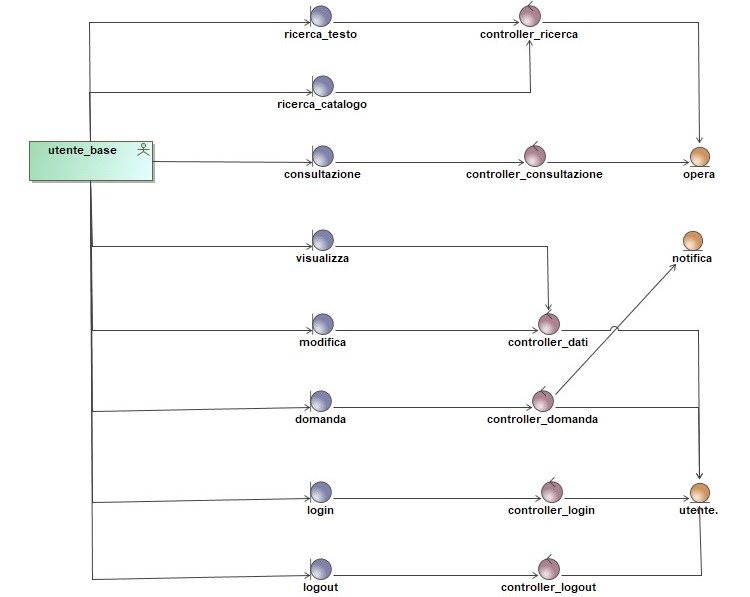
Descrizione generata automaticamente**

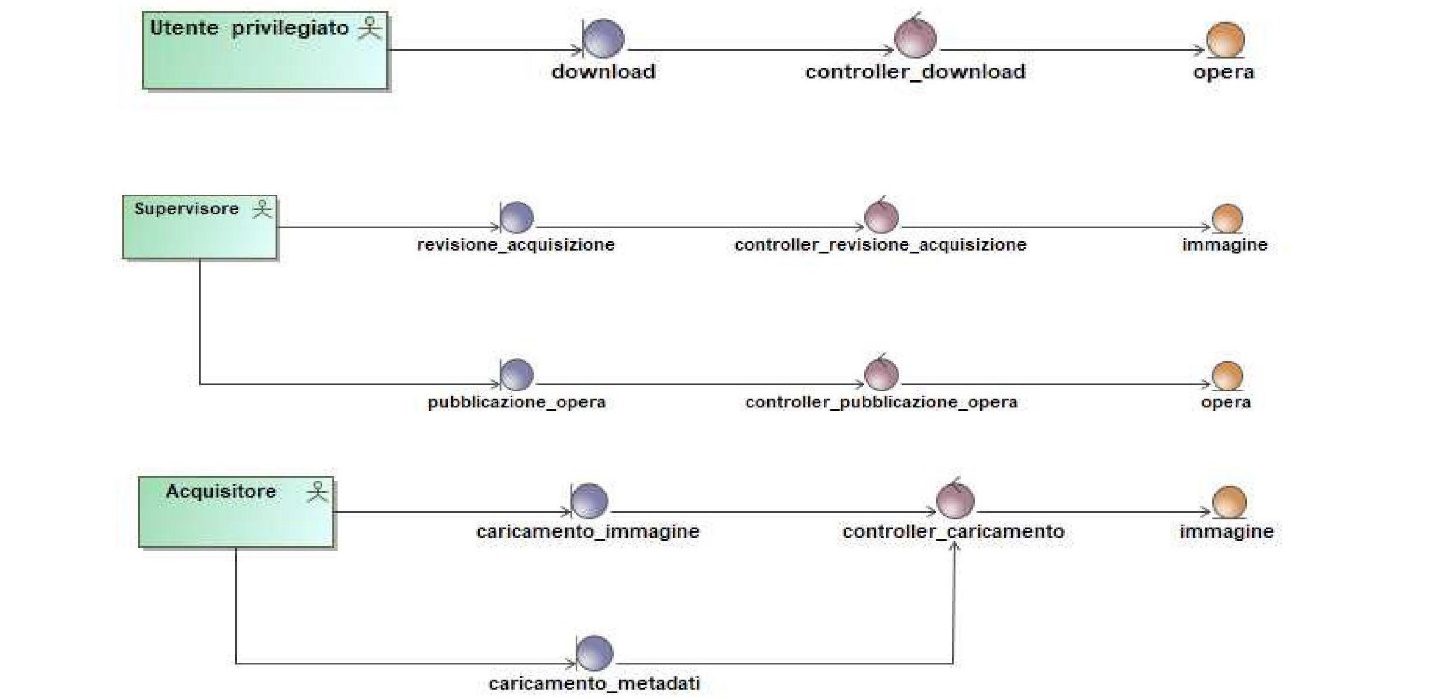
***Modello di dominio***

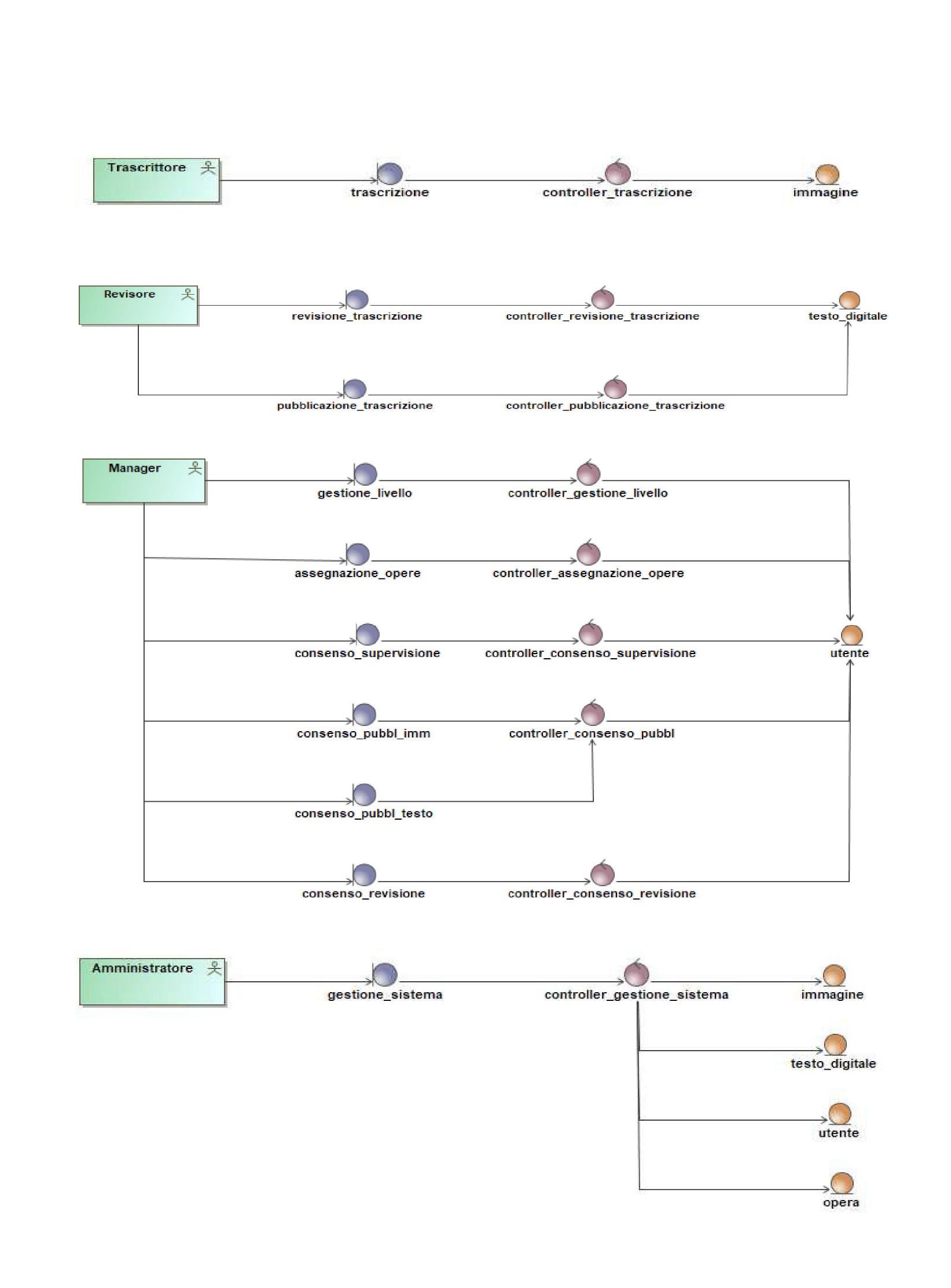
**(da rivedere)**

***1.4 Analisi classi entity,boundary,controller***

****

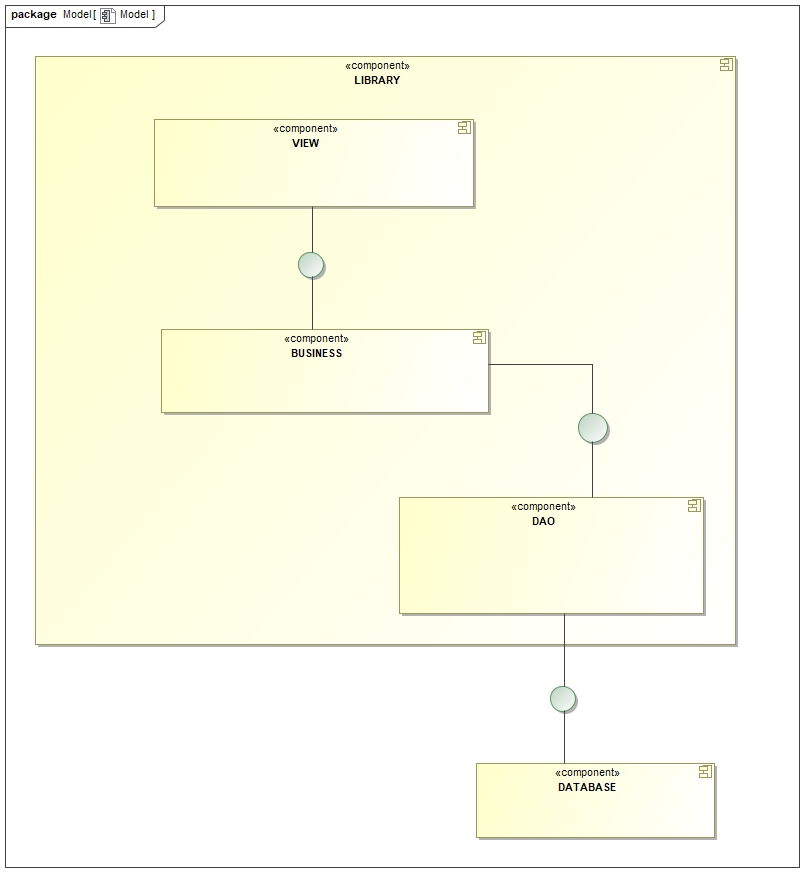
****

****

****

***Modello dell’architettura software***

***Component diagram***

****

***Descrizione dell’architettura***

La componente “Library” è suddivisa in diverse sottocomponenti:

* View
* Business
* Dao

La componente “View” contiene l’interfaccia grafica del sistema (realizzata con JavaFX). La componente ‘Business’ è composta da ‘Model’ (con i metodi get() e set()) e da ‘Controller’(che esegue le varie operazioni sui dati). Le varie funzioni per interfacciarsi con i dati del database vengono svolte all’interno della componente “Dao”.

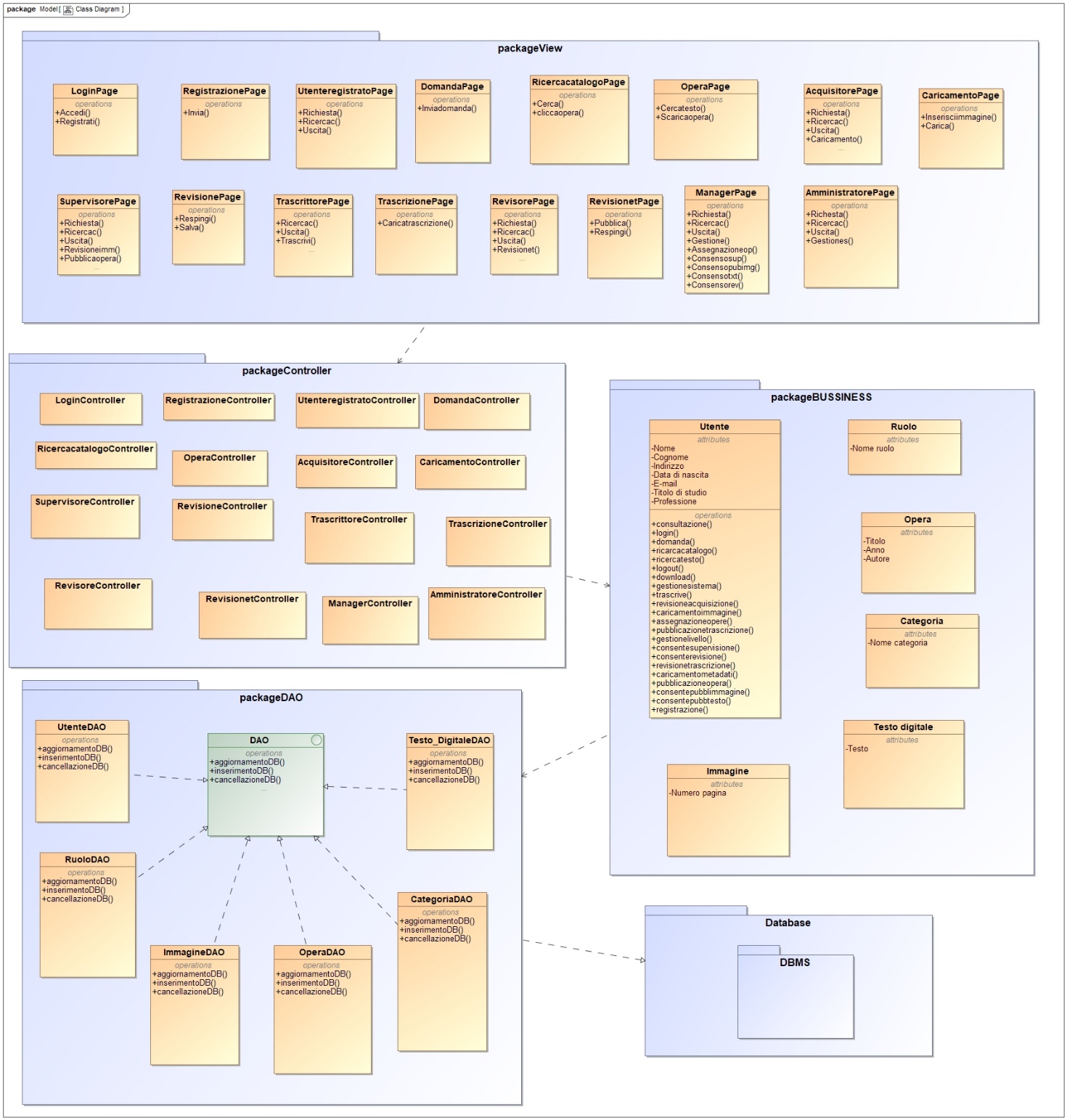
***Descrizione delle scelte e delle strategie adottate***

Per interfacciarsi con i dati l’architettura del sistema segue il pattern architetturale DAO. Per quanto riguarda il database la scelta è ricaduta su un database relazionale. Per quanto riguarda la suddivisione della logica il sistema segue il pattern MVC:

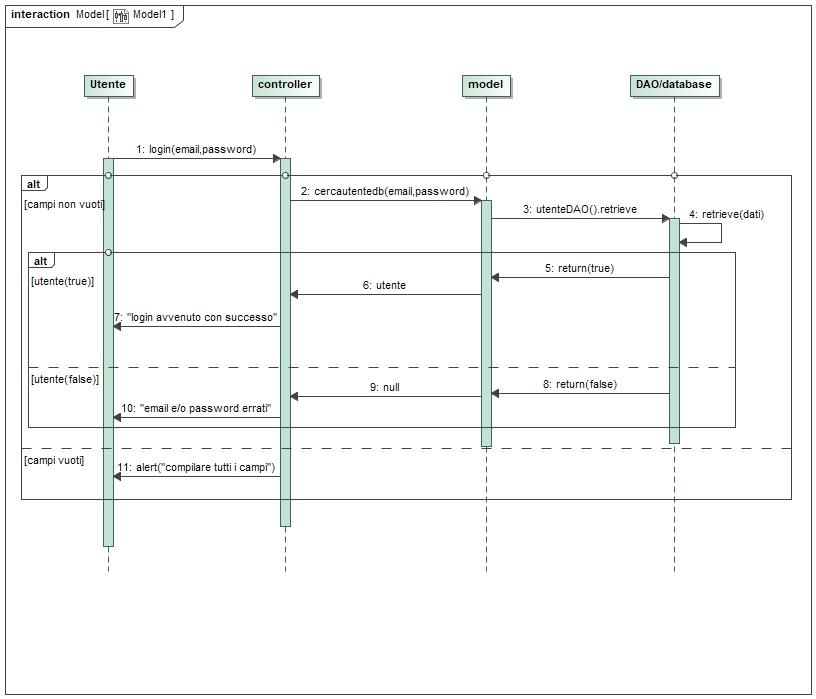
* Model: Business, ossia gestione dei dati
* View: Interfaccia, che permette la comunicazione tra utente e sistema
* Controller: che analizza i dati in input e li converte in comandi per Model e/o per View

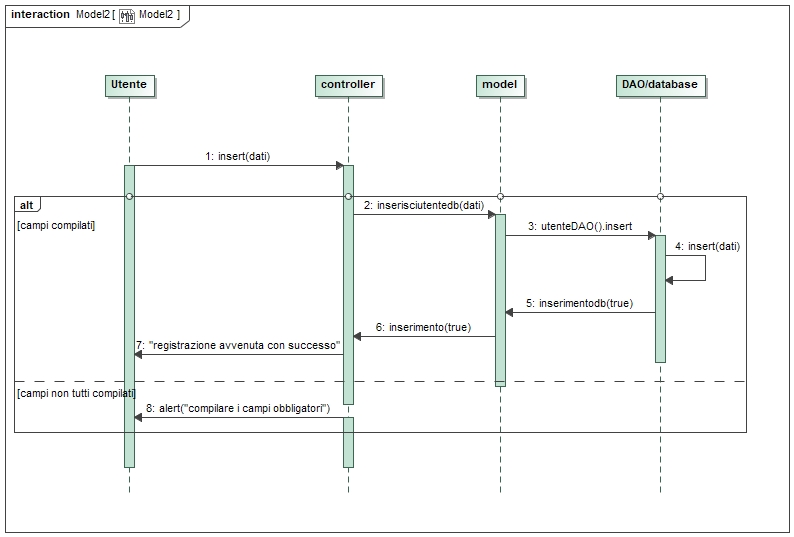
È stato inoltre utilizzato il design pattern relazionale singleton.

**Class diagram**

****

***Sequence diagram***

******

-login******

-register