

TRACK TUNE

SW Engineering Documentation

Autori:

Mattia REBONATO
Davide FOZZATO

Professore:

Prof Carlo COMBI

A.A. 2024-2025

Indice

1	Introduzione	2
1.1	Prefazione	2
1.2	Panoramica	2
2	Analisi dei requisiti	3
2.1	Descrizione del sistema	3
2.2	Requisiti funzionali	4
2.2.1	Gestione degli utenti	4
2.2.2	Gestione dei contenuti	4
2.2.3	Gestione dei commenti	4
2.3	Casi d'uso	4
2.3.1	Utente non loggato	6
2.3.2	Utente loggato	7
2.3.3	Amministratore	14
2.4	Glossario dei termini	16
2.5	Activity Diagrams	16
3	Modello logico dei dati	22
3.1	Diagramma E/R	22
3.2	Descrizione delle entità	22
4	Architettura logica	23
4.1	Diagramma UML delle classi - Model	23
4.2	Diagramma UML delle classi - Control	24
4.3	Diagramma UML delle classi - Utility ed Exception	25
4.4	Pattern architetturale	25
5	Developing	26
5.1	Gestione dell'accesso ai dati (DAO e DAOProvider)	26
5.2	Design pattern utilizzati	26
5.3	Diagramma UML delle classi - Utility ed Exception	27
5.4	Struttura del progetto	27
6	Fase di Test	28
6.1	JUnit Test	28
6.2	Test degli sviluppatori	28
6.3	Test soggetti esterni	28

1 Introduzione

1.1 Prefazione

Il presente documento definisce in modo esaustivo i requisiti, l'architettura e le specifiche di progettazione del software *TrackTune*, una piattaforma applicativa dedicata alla gestione, condivisione e analisi collaborativa di contenuti musicali.

La documentazione è rivolta a sviluppatori, tester, amministratori di sistema e a tutti gli stakeholder coinvolti nella realizzazione, manutenzione e gestione del progetto.

1.2 Panoramica

La piattaforma si rivolge a musicisti, compositori, insegnanti, studenti e appassionati di musica, offrendo un ambiente digitale comodo per l'archiviazione, la collaborazione e l'accesso a materiale musicale di vario genere, tra cui spartiti, testi, accordi, file audio/video e link esterni (ad esempio YouTube).

Essa si propone non solo come archivio digitale, ma anche come spazio per l'approfondimento culturale, lo scambio di interpretazioni e feedback tra utenti, favorendo la condivisione di esperienze musicali, commenti critici e annotazioni esecutive su brani e contenuti attinenti alla musica.

2 Analisi dei requisiti

2.1 Descrizione del sistema

Si vuole progettare un sistema software per gestire la collezione e la condivisione di spartiti, testi, accordi, MIDI, MP3, video, link e molto altro, relativamente a brani musicali di diversa tipologia/-genere.

Gli utenti, previa autorizzazione dell'amministratore, devono poter caricare, scaricare, commentare e interagire con i vari contenuti. Per ogni risorsa il sistema deve consentire di specificare:

- autori
- genere/generi
- strumenti musicali utilizzati

Inoltre, per le risorse multimediali (audio, video, ecc.) il sistema deve consentire di specificare il luogo e la data di registrazione.

Ogni utente può aggiungere note di testo libero su segmenti specifici di esecuzione (MP3, MP4 o video YouTube). Un segmento è definito dal momento di inizio e fine, e per ciascuno è possibile includere dettagli come assoli, esecutori, strumenti, ritmi e altre caratteristiche. I commenti possono essere arricchiti con ulteriori risposte, consentendo una profondità illimitata di discussioni relativamente a quel contenuto.

Un brano può essere inserito anche da un autore o un interprete. Deve essere possibile quindi distinguere il proprio ruolo all'interno del brano e, nel caso di interpreti, gli strumenti utilizzati. I commenti sulle modalità esecutive devono apparire in maniera rilevante rispetto a quelli di altri utenti.

I video YouTube possono essere visualizzati direttamente nel software o nel browser, ma tutti i commenti relativi sono gestiti dal sistema.

Gli utenti possono aggiungere commenti su spartiti, testi e accordi, relativi alle modalità esecutive (come strumenti, ritmo, intensità, ecc.), sia su specifiche parti del brano che sull'intero brano.

Un utente può richiedere la registrazione alla piattaforma tramite un form, che l'amministratore valuterà e risponderà a seconda dell'esito. L'amministratore gestisce gli utenti, ha il compito di rimuovere coloro che pubblicano contenuti non pertinenti e di moderare i commenti. Inoltre, valuta le richieste di registrazione, il cui esito verrà comunicato all'utente interessato.

Il sistema deve permettere la ricerca delle risorse tramite i seguenti filtri:

- genere
- titolo del brano
- autore/i

Ogni risorsa per cui un utente ha lasciato un commento deve essere direttamente accessibile.

Gli strumenti musicali, i generi, i titoli dei brani e i nomi degli autori vanno gestiti attraverso dizionari aggiornabili, da usare anche per le opportune ricerche.

2.2 Requisiti funzionali

2.2.1 Gestione degli utenti

- Registrazione tramite form dedicato;
- Valutazione da parte dell'amministratore delle richieste di registrazione e relativa comunicazione dell'esito;
- Profilazione (distinzione tra utente/amministratore);
- Autorizzazione al caricamento dei file per utente;

2.2.2 Gestione dei contenuti

- Aggiunta, modifica e rimozione di contenuti multimediali
 - La rimozione può essere eseguita solo dall'amministratore o dall'utente che ha caricato il contenuto
- Visualizzazione delle risorse tramite software di terze parti o tramite l'applicativo stesso.
- Moderazione degli utenti
 - L'amministratore può rimuovere utenti;
 - L'amministratore può rimuovere commenti/risorse inappropriate.
- Gestione dei dizionari aggiornabili (solo amministratore) relativi a:
 - Autori;
 - Generi;
 - Strumenti musicali;

2.2.3 Gestione dei commenti

- Possibilità di commentare brani;
- Possibilità di rispondere ai commenti;
- Possibilità di commentare su specifici segmenti di esecuzione;
- Visualizzazione differenziata dei commenti per autori/executori del brano.

2.3 Casi d'uso

L'applicazione sarà disponibile esclusivamente agli utenti abilitati dall'amministratore, il quale è anch'esso un utente ma con privilegi di amministrazione sui comportamenti del sistema. L'unica operazione che può effettuare un utente non registrato è la richiesta di creazione di un account.

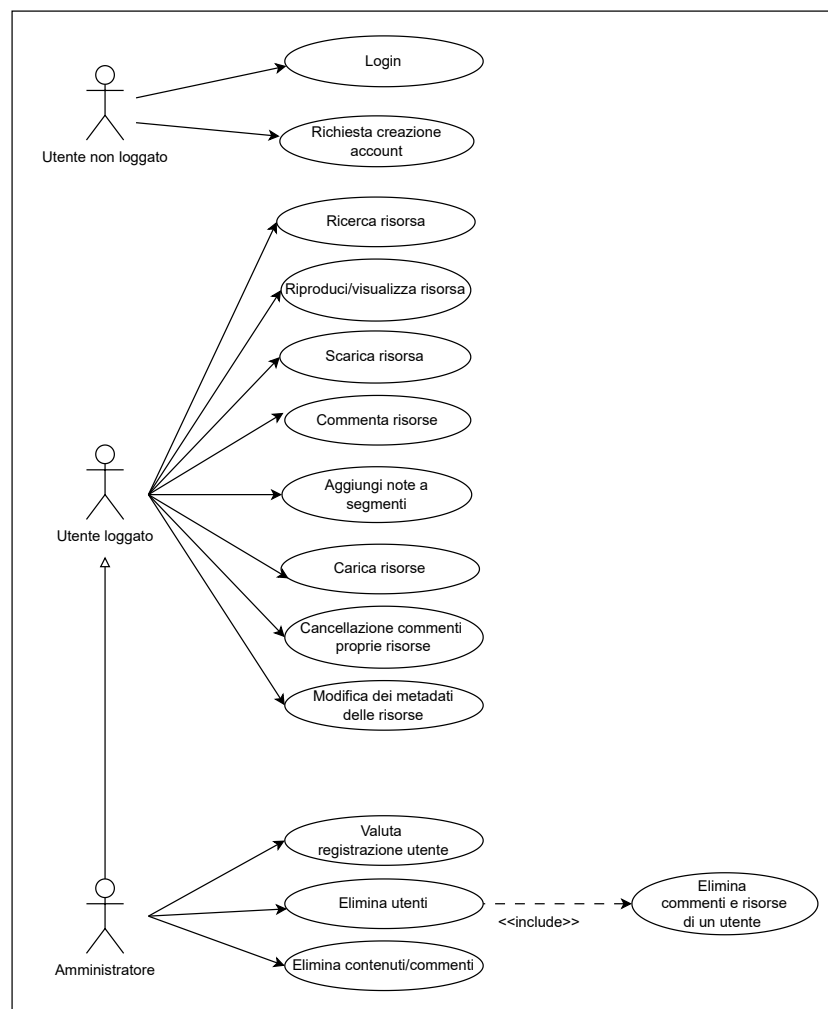


Figura 1: Use Cases

2.3.1 Utente non loggato

L'utente non autenticato può effettuare esclusivamente le operazioni:

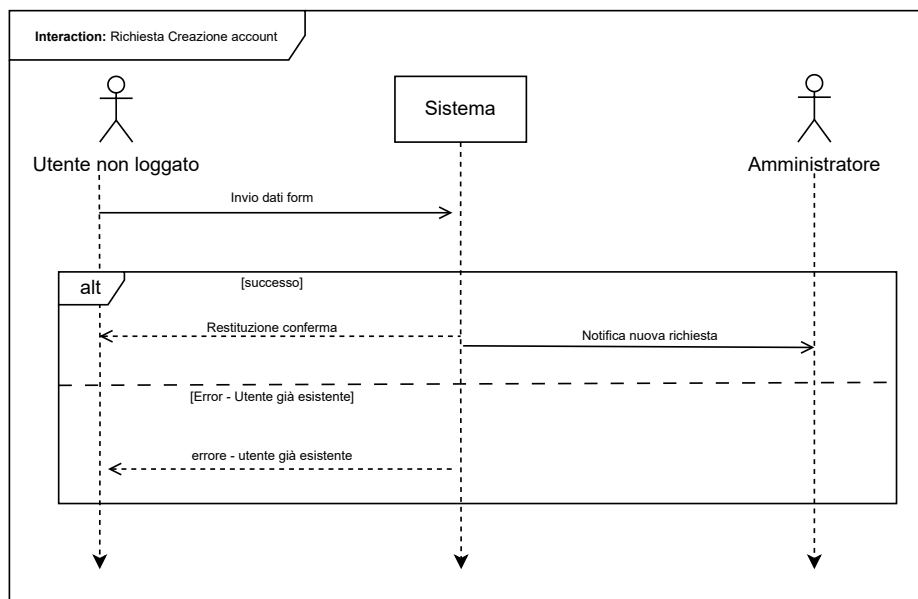
1. Richiedere un account ad un amministratore
2. Effettuare l'accesso tramite un form di autenticazione

Caso d'uso: Login	
Id	UC0
Attori	Utente non loggato
Precondizioni	L'utente non è autenticato sulla piattaforma.
Sequenza degli eventi	<ol style="list-style-type: none"> 1. L'utente compila i campi per effettuare l'autenticazione 2. Il sistema verifica i dati e gestisce le autorizzazioni dell'utente
Post condizioni	Utente autenticato e correttamente gestito

Tabella 1: Descrizione caso d'uso *Login*

Caso d'uso: Richiesta creazione account	
Id	UC1
Attori	Utente non loggato
Precondizioni	<p>L'utente non è autenticato sulla piattaforma.</p> <p>L'utente accede alla sezione di registrazione.</p>
Sequenza degli eventi	<ol style="list-style-type: none"> 1. Vengono compilati i dati relativi alla richiesta 2. La richiesta viene inviata 3. Il sistema registra la richiesta e la mette in stato di approvazione 4. L'account viene salvato come account provvisorio 5. Il sistema notifica l'amministratore della nuova richiesta
Sequenza alternativa	Utente già esistente che comporta un messaggio di errore
Post condizioni	La richiesta di creazione dell'account deve essere approvata da un amministratore

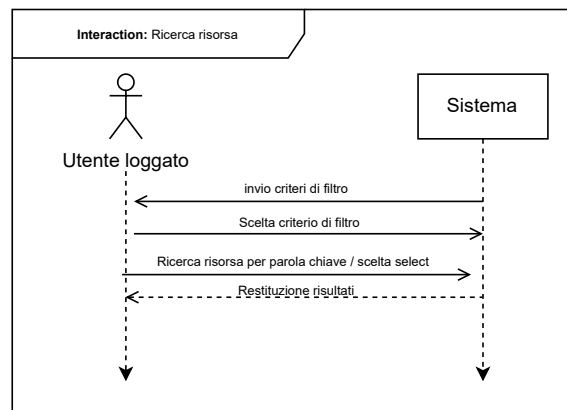
Tabella 2: Descrizione caso d'uso *Richiesta creazione account*

Figura 2: Sequence diagram del caso d'uso *Richiesta creazione account*

2.3.2 Utente loggato

Caso d'uso: Ricerca risorsa	
Id	UC2
Attori	Utente loggato
Precondizioni	Utente autenticato
Sequenza degli eventi	<ol style="list-style-type: none"> 1. Il sistema invia i criteri di filtro disponibili 2. L'utente decide il filtro di ricerca 3. L'utente inserisce una parola chiave per ricercare una risorsa 4. Gli vengono restituite le risorse data quella ricerca
Sequenza alternativa 1	Errore nella ricerca: messaggio di errore
Post condizioni	L'utente ha visualizzato la risorsa correttamente

Tabella 3: Descrizione caso d'uso *Ricerca risorsa*

Figura 3: Sequence diagram del caso d'uso *Ricerca risorsa*

Caso d'uso: Riproduci/Visualizza risorsa	
Id	UC3
Attori	User Logged
Precondizioni	Utente autenticato L'utente ha effettuato una ricerca o ne può selezionare una
Sequenza degli eventi	<ol style="list-style-type: none"> 1. L'utente seleziona una risorsa dal risultato delle ricerca 2. Il sistema riconosce il tipo di risorsa 3. Se multimediale, lo riproduce 4. Se testuale, lo visualizza
Sequenza alternativa 1	Errore nella riproduzione: messaggio di errore
Post condizioni	L'utente ha visualizzato la risorsa

Tabella 4: Descrizione caso d'uso *Riproduci/Visualizza risorsa*

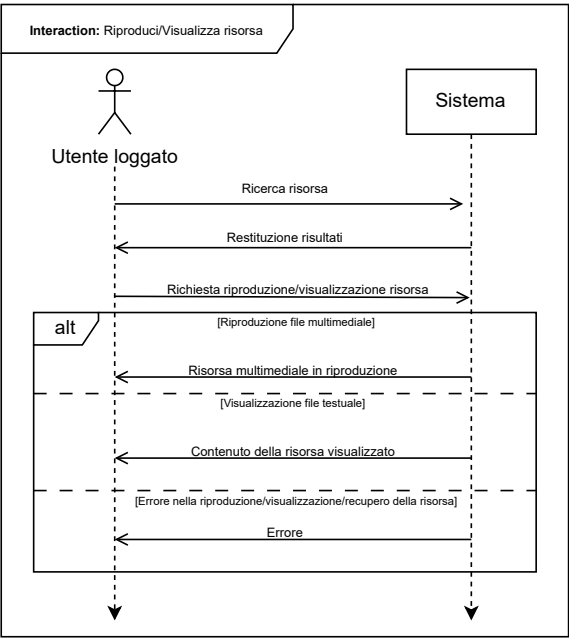
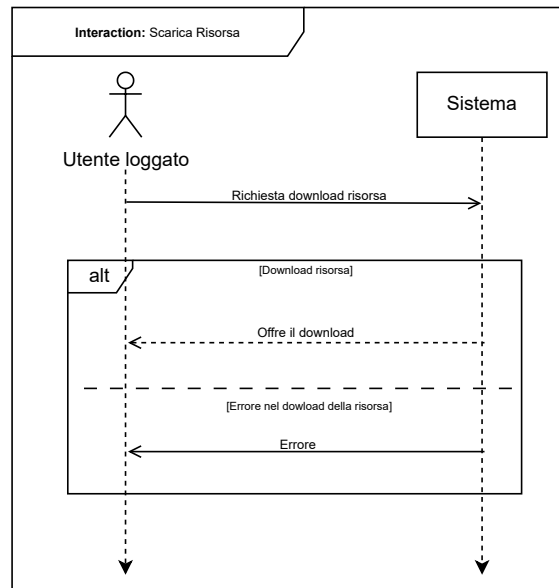


Figura 4: Sequence diagram del caso d'uso *Riproduci/Visualizza risorsa*

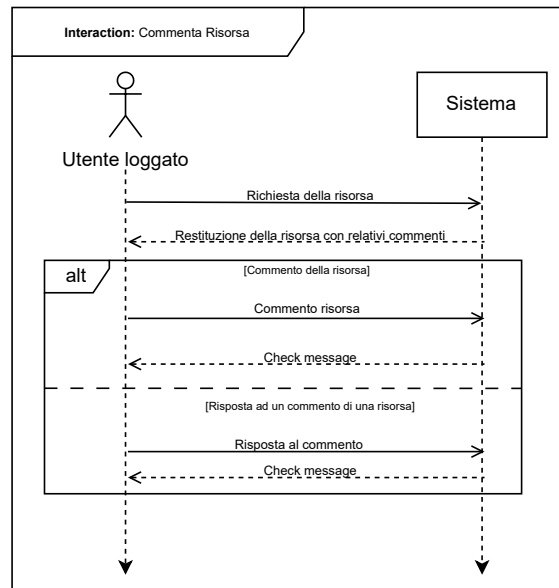
Caso d'uso: Scarica risorsa	
Id	UC4
Attori	User Logged
Precondizioni	Utente autenticato L'utente ha effettuato una ricerca o ne può selezionare una
Sequenza degli eventi	1. L'utente seleziona una risorsa dal risultato delle ricerca 2. L'utente richiede il download della risorsa
Sequenza alternativa 1	Errore nel download: messaggio di errore
Post condizioni	L'utente ha scaricato la risorsa correttamente

Tabella 5: Descrizione caso d'uso *Scarica risorsa*

Figura 5: Sequence diagram del caso d'uso *Scarica risorsa*

Caso d'uso: Commenta risorse	
Id	UC5
Attori	User Logged
Precondizioni	Utente autenticato L'utente può visualizzare una risorsa
Sequenza degli eventi	<ol style="list-style-type: none"> 1. L'utente seleziona una risorsa dal risultato delle ricerca 2. L'utente può scrivere un commento 3. Il commento viene salvato dal sistema
Sequenza alternativa 1	L'utente può rispondere ad un commento
Post condizioni	L'utente ha commentato la risorsa correttamente

Tabella 6: Descrizione caso d'uso *Commenta risorse*

Figura 6: Sequence diagram del caso d'uso *Commenta risorse*

Caso d'uso: Aggiungi note a segmenti	
Id	UC6
Attori	User Logged
Precondizioni	Utente autenticato L'utente ha ricercato/può visualizzare una risorsa
Sequenza degli eventi	<ol style="list-style-type: none"> 1. L'utente seleziona un intervallo di tempo per inserire una nota/commento 2. Scrive la /commento desiderata 3. Invia la nota/commento al sistema 4. Il sistema registra la nota/commento
Post condizioni	L'utente ha scritto una nota/commento su un segmento correttamente

Tabella 7: Descrizione caso d'uso *Aggiungi note a segmenti*

Caso d'uso: Carica Risorse	
Id	UC7
Attori	User Logged
Precondizioni	Utente autenticato
Sequenza degli eventi	<div>1. L'utente visualizza la schermata di upload dei file</div> <div>2. Seleziona la risorsa da caricare</div> <div>3. Inserisce i dati (file + metadati obbligatori)</div> <div>4. Invia i dati al sistema</div> <div>5. Il sistema li memorizza</div>
Sequenza alternativa 1	Errore di sistema: messaggio di errore
Sequenza alternativa 2	Errore nell'inserimento dei dati: messaggio di errore
Post condizioni	L'utente ha caricato la risorsa correttamente

Tabella 8: Descrizione caso d'uso *Carica Risorse*

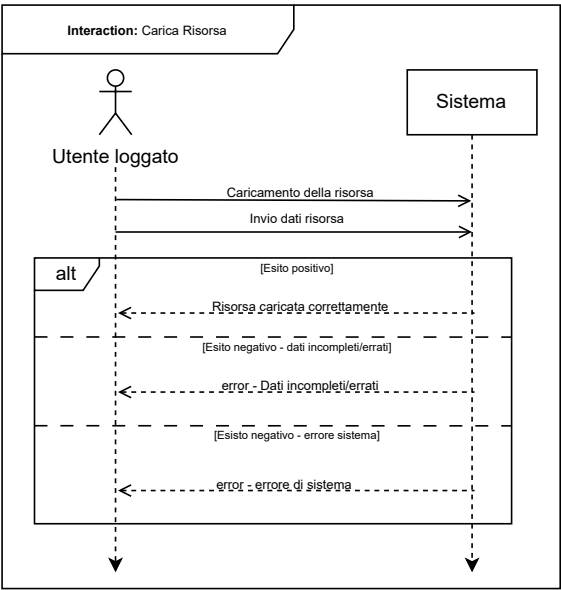


Figura 7: Sequence diagram del caso d'uso *Carica Risorse*

Caso d'uso: Cancellazione commenti delle proprie risorse	
Id	UC8
Attori	User Logged
Precondizioni	Utente autenticato e autorizzato L'utente sta visualizzando una propria risorsa caricata
Sequenza degli eventi	<ol style="list-style-type: none"> 1. L'utente desidera eliminare un qualsiasi commento della risorsa 2. Elimina il commento 3. Il sistema controlla che abbia i permessi necessari 4. In caso positivo effettua la rimozione 5. Vengono rimossi tutte le risposte a quel commento
Sequenza alternativa 1	Errore Autorizzazioni: messaggio di errore
Post condizioni	L'utente ha eliminato il commento correttamente

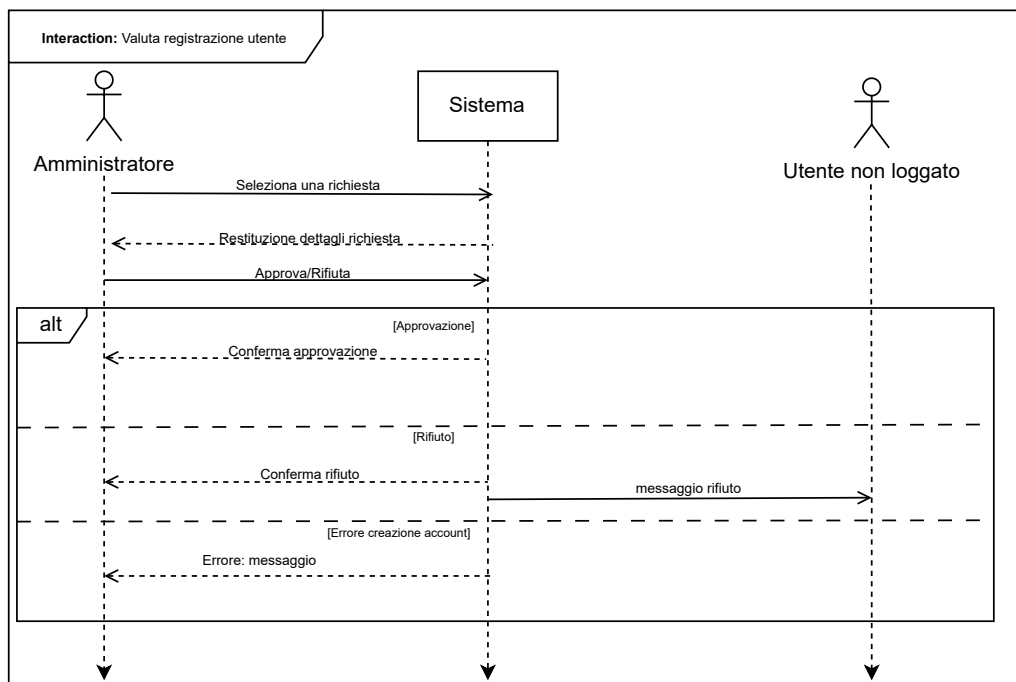
Tabella 9: Descrizione caso d'uso *Cancellazione commenti delle proprie risorse*

Caso d'uso: Modifica dei metadati delle risorse	
Id	UC9
Attori	User Logged Authorized
Precondizioni	Utente autenticato L'utente è il proprietario di quella risorsa L'utente sta visualizzando i dati di una risorsa
Sequenza degli eventi	<ol style="list-style-type: none"> 1. L'utente desidera modificare un qualsiasi metadato della risorsa 2. Modifica i metadati 3. Invia le modifiche al sistema 4. Il sistema effettua la modifica
Post condizioni	L'utente ha modificato correttamente i metadati di una risorsa

Tabella 10: Descrizione caso d'uso *Modifica dei metadati delle risorse*

2.3.3 Amministratore

Caso d'uso: Valuta registrazione utente	
Id	UC10
Attori	Administrator
Precondizioni	L'amministratore accede alla sezione di gestione delle registrazioni utente in attesa di valutazione
Sequenza degli eventi	<ol style="list-style-type: none"> 1. L'amministratore visualizza i dettagli 2. L'amministratore approva/rifiuta la richiesta 3. Il sistema elimina la richiesta e in caso di approvazione crea l'account 4. Il sistema notifica all'utente l'esito
Post condizioni	L'utente è registrato e abilitato all'accesso al sistema (se approvato) o la sua richiesta di registrazione viene rifiutata

Tabella 11: Descrizione caso d'uso *Valuta registrazione utente*Figura 8: Sequence diagram del caso d'uso *Valuta registrazione utente*

Caso d'uso: Eliminazione utente	
Id	UC11
Attori	Administrator
Precondizioni	L'amministratore accede alla sezione di gestione degli utenti
Sequenza degli eventi	<ol style="list-style-type: none"> 1. L'amministratore seleziona l'utente che desidera eliminare 2. L'amministratore elimina l'utente 3. Rimuove l'utente
Post condizioni	L'utente è stato cancellato correttamente

Tabella 12: Descrizione caso d'uso *Eliminazione utente*

Caso d'uso: Elimina commenti e risorse di un utente	
Id	UC12
Attori	Administrator
Precondizioni	Caso d'uso <i>Eliminazione Utente</i> (Tabella 12)
Sequenza degli eventi	<ol style="list-style-type: none"> 1. L'amministratore comunica al server un utente 2. Le risorse e commenti di questo utente vengono eliminate
Post condizioni	Le risorse/commenti sono stati cancellati correttamente

Tabella 13: Descrizione caso d'uso *Elimina commenti e risorse di un utente*

Caso d'uso: Elimina contenuti/commenti	
Id	UC13
Attori	Administrator
Precondizioni	Visualizzazione di una risorsa/commento
Sequenza degli eventi	<ol style="list-style-type: none"> 1. L'amministratore visualizza una risorsa 2. L'amministratore decide di eliminare la risorsa/commento 3. Invia la richiesta al sistema 4. Il sistema elimina la risorsa/commento
Post condizioni	La risorsa/commento è stata eliminata correttamente

Tabella 14: Descrizione caso d'uso *Elimina contenuti/commenti*

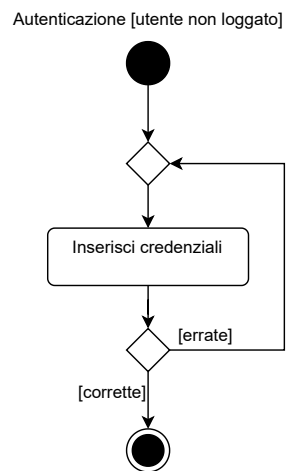
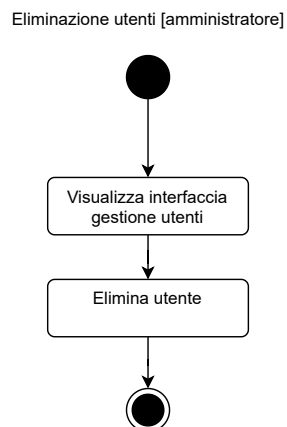
2.4 Glossario dei termini

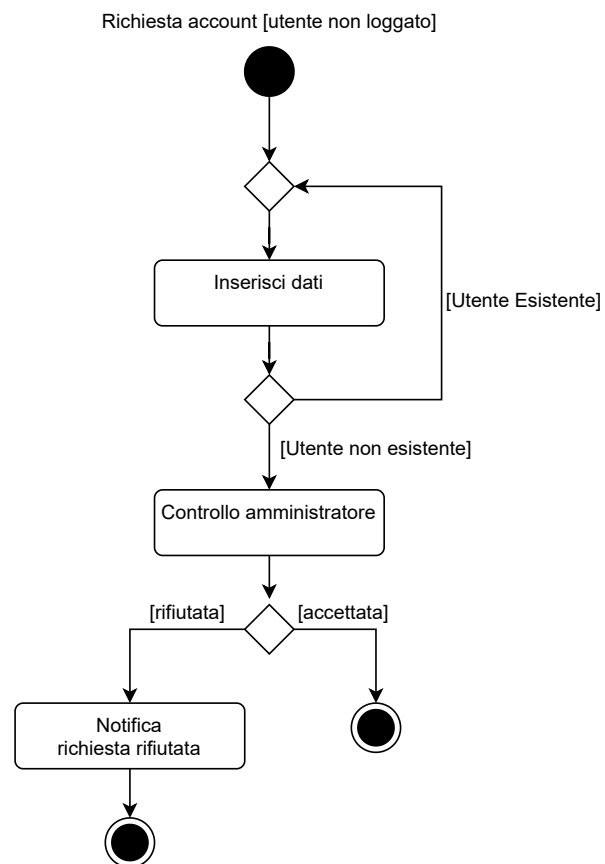
GLOSSARIO DEI TERMINI	
TERMINE	DEFINIZIONE
Contenuto musicale / Risorsa multimediale	Qualsiasi tipo di file audio, video o immagine caricato e reso disponibile sulla piattaforma. Può includere brani musicali, podcast, lezioni, video, ecc.
Autore / Esecutore / Interprete di un brano	L'individuo o il gruppo che ha contribuito in modo significativo alla creazione, composizione, esecuzione o interpretazione originale di un'opera musicale.
Segmento di un brano	Un sottoinsieme specifico di un contenuto multimediale (es. un brano musicale o un video), definito da un intervallo di tempo di inizio e fine, permettendo di focalizzare l'attenzione su una parte specifica.
Traccia	Una traccia può contenere più risorse relative a quel contesto, ad esempio il relativo file PDF dello spartito + un'eventuale risorsa multimediale (mp4/mp3)
Brano	Un'opera musicale originale completa e autonoma, che può essere parte di un album o una singola traccia, caratterizzata da una composizione unica.
Metadato	Un insieme di dati che descrivono e forniscono informazioni su altri dati (una risorsa). Per un brano, i metadati possono includere titolo, autore, genere, durata, data di pubblicazione, ecc.
Utente autenticato / Loggato	Un utente che ha completato con successo il processo di login fornendo credenziali valide ed è riconosciuto dal sistema, avendo accesso a funzionalità riservate.
Amministratore	Un utente con privilegi speciali all'interno della piattaforma, in grado di gestire utenti, contenuti, approvare richieste e monitorare le operazioni del sistema.
Richiesta di creazione account	La procedura iniziata da un utente non autenticato per ottenere un nuovo account sulla piattaforma, soggetta ad approvazione da parte di un amministratore.
Commento	Un testo o una nota aggiunta da un utente autenticato a una risorsa (o a un altro commento) per esprimere un'opinione, fornire feedback o interagire con altri utenti.
Nota a segmento	Un tipo specifico di commento o annotazione testuale associata a un intervallo di tempo preciso all'interno di una risorsa multimediale, permettendo di evidenziare o commentare una parte specifica del contenuto.

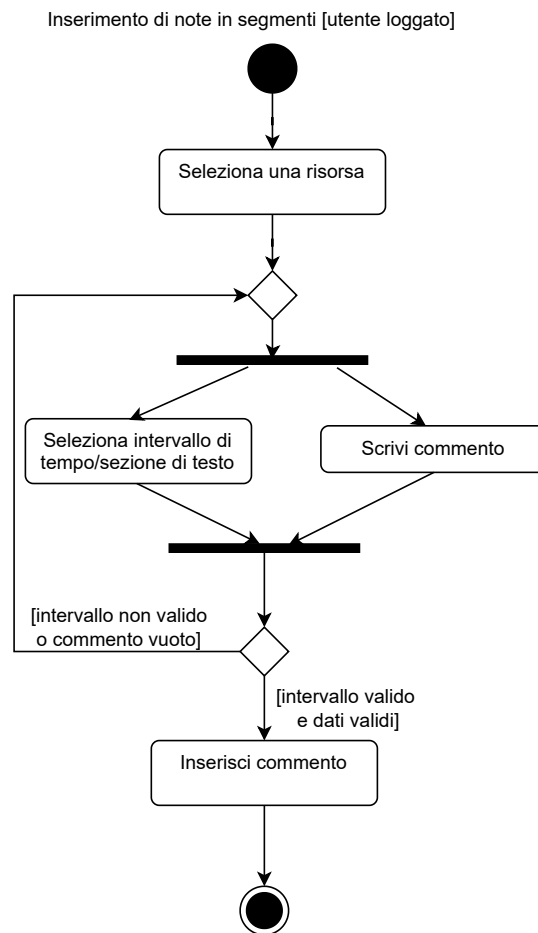
Tabella 15: Glossario dei Termini

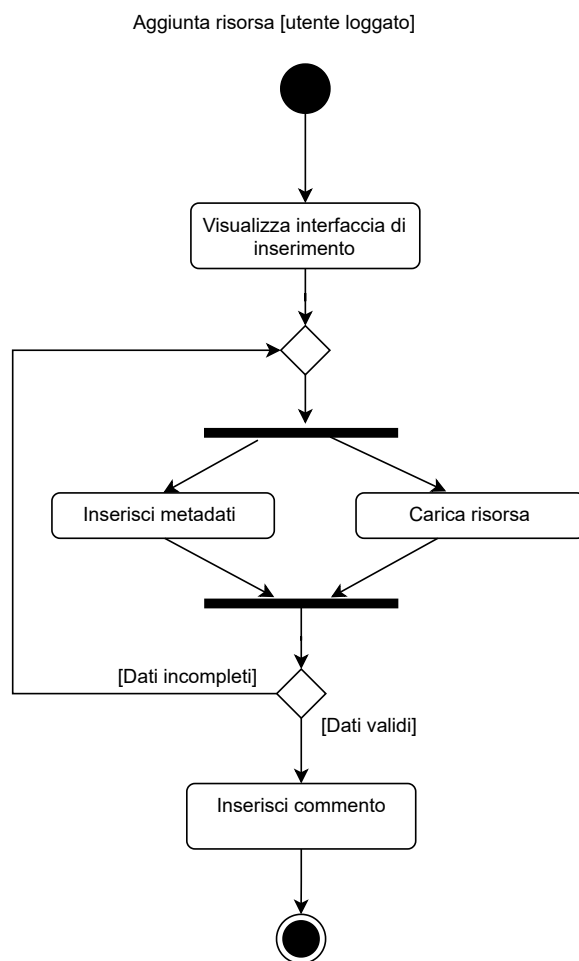
2.5 Activity Diagrams

Questa sezione illustra i principali flussi di lavoro del sistema attraverso gli activity diagrams.

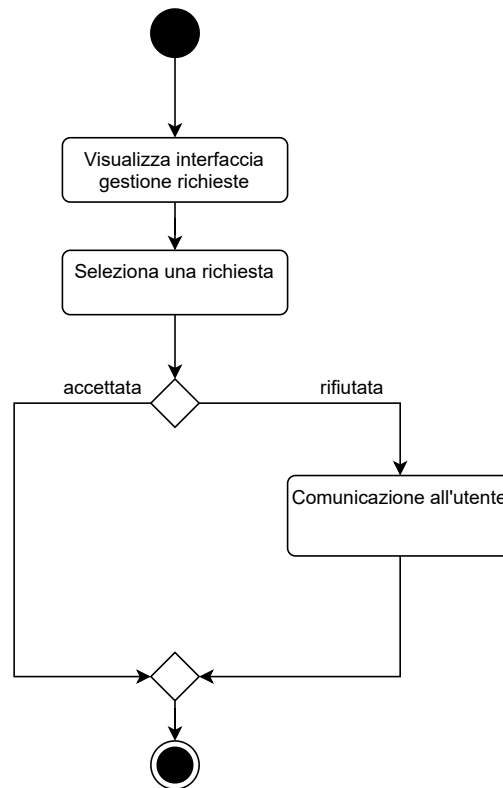
Figura 9: Activity Diagrams del caso d'uso *Login*Figura 10: Activity Diagrams del caso d'uso *Elimina utenti*

Figura 11: Activity Diagrams del caso d'uso *Richiesta creazione account*

Figura 12: Activity Diagrams del caso d'uso *Aggiungi note a segmenti*

Figura 13: Activity Diagrams del caso d'uso *Carica risorse*

Valutazione richiesta account [Amministratore]

Figura 14: Activity Diagrams del caso d'uso *Valuta registrazione utenti*

3 Modello logico dei dati

La presente sezione descrive il modello logico dei dati adottato nel sistema, che rappresenta l'organizzazione e la struttura concettuale delle informazioni da gestire.

3.1 Diagramma E/R

IMMAGINE DIAGRAMMA ER PRIMA RISTRUTTURAZIONE

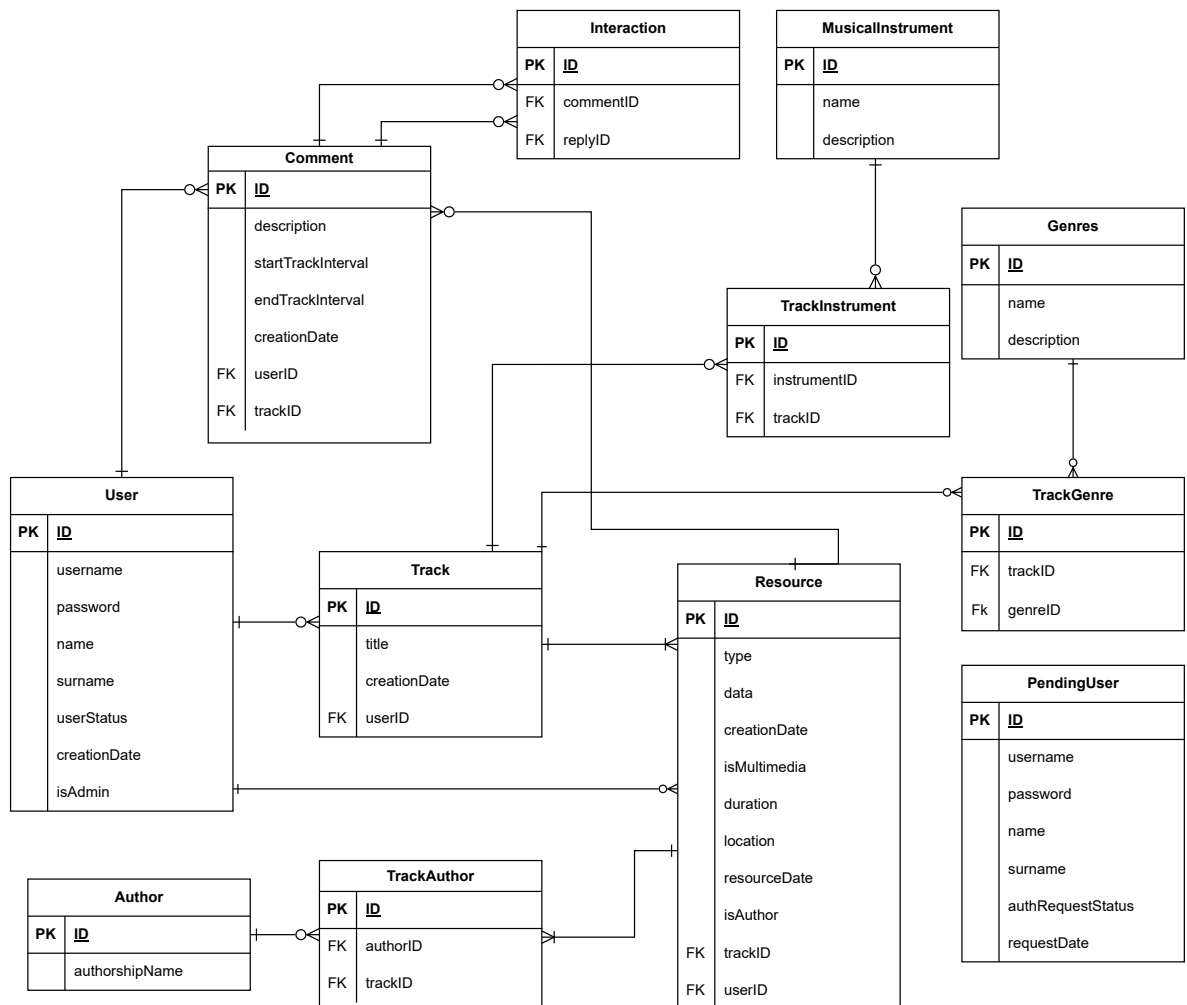


Figura 15: Diagramma E/R ristrutturato

3.2 Descrizione delle entità

Descrizione delle entità principali (possiamo rimuoverlo volendo)

4 Architettura logica

Questa sezione presenta l'architettura logica del sistema, ovvero come sono organizzati i principali componenti software e come interagiscono tra loro. L'obiettivo è offrire una panoramica chiara della struttura del progetto, utile per guidare lo sviluppo e facilitare la manutenzione nel tempo.

4.1 Diagramma UML delle classi - Model

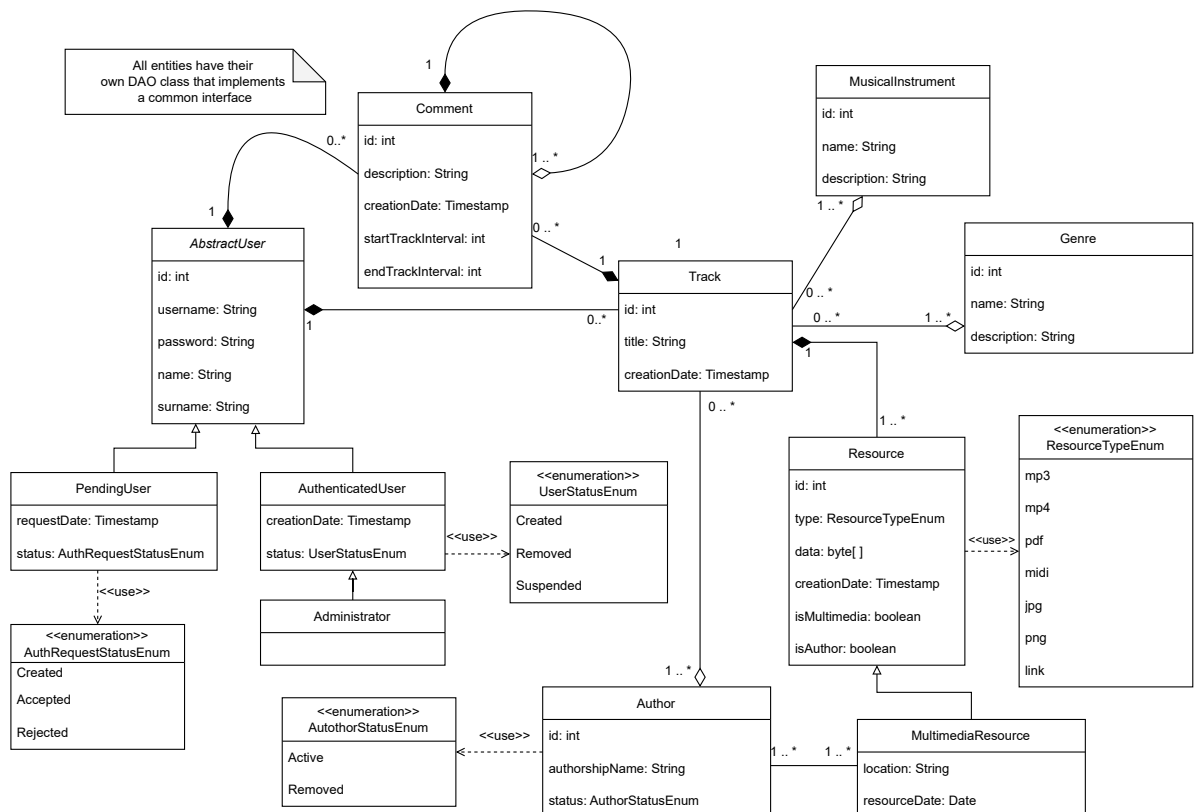
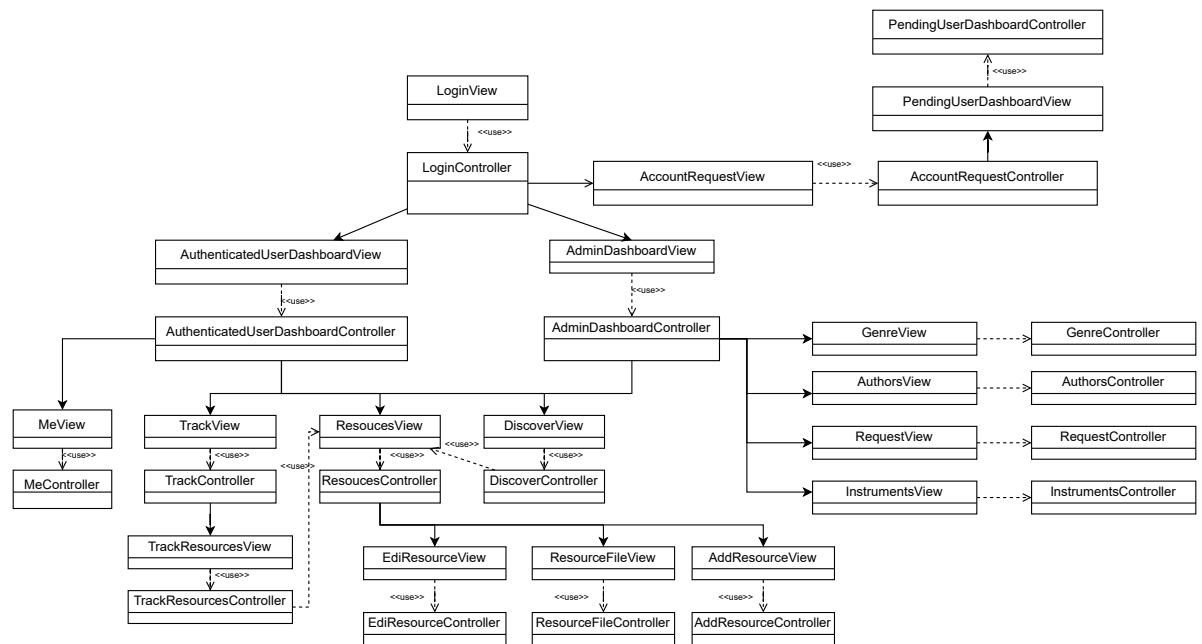


Figura 16: Class Diagrams UML relative al *Model*

4.2 Diagramma UML delle classi - Control

Figura 17: Class Diagrams UML relative al *Controller*

Il diagramma mostra la classe base **Controller** e tutte le sue sottoclassi organizzate nei seguenti pacchetti:

```

Controller
common
    AddResourceController
    DiscoverController
    EditResourceController
    ResourceFileController
    ResourcesController
    TrackResourcesController
    TracksController
admin
    AdminDashboardController
    AuthorsController
    GenresController
    InstrumentsController
    RequestsController
    UsersController
authenticatedUser
    AuthenticatedUserDashboardController
    MeController
authentication
    AccountRequestController
    LoginController
    SessionManager
pendingUser
    PendingUserDashboardController
  
```

Tutte le classi controller estendono la classe base **Controller** e implementano l'interfaccia **Initializable** di JavaFX.

4.3 Diagramma UML delle classi - Utility ed Exception

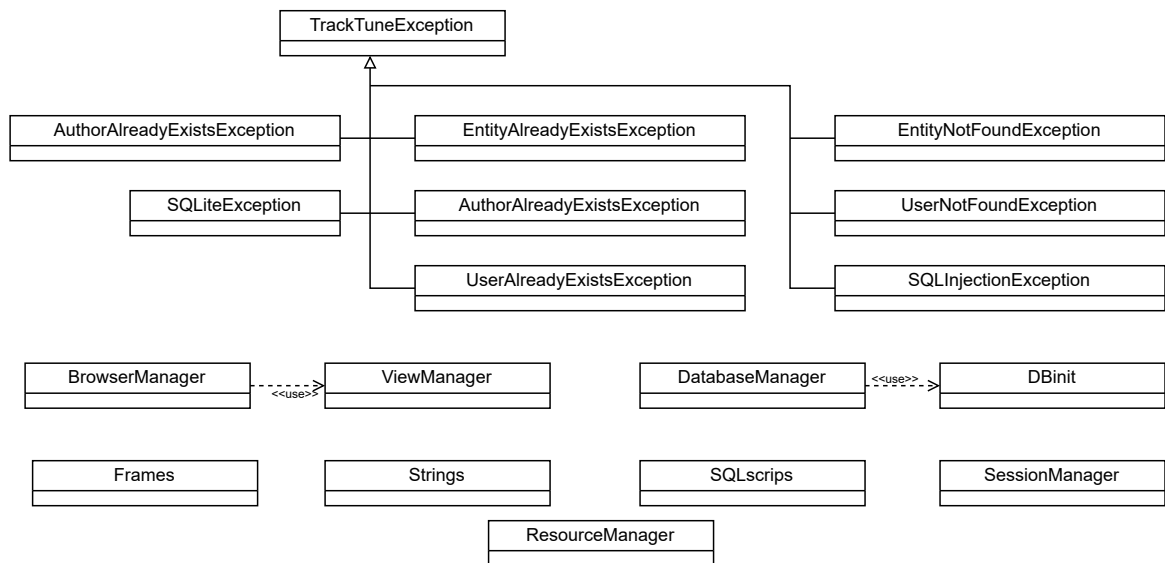


Figura 18: Class Diagrams UML relative a *Utility ed Exception*

4.4 Pattern architetturale

5 Developing

5.1 Gestione dell'accesso ai dati (DAO e DAOProvider)

Il sistema adotta il pattern *Data Access Object* (DAO) per gestire l'accesso ai dati in modo modulare e disaccoppiato dalla logica applicativa.

Per ciascuna entità principale del *Model*, è presente una corrispondente classe DAO che si occupa delle operazioni di persistenza (inserimento, aggiornamento, eliminazione e lettura) più eventuali operazioni specifiche. Ogni DAO implementa un'interfaccia dedicata, garantendo così l'astrazione e la possibilità di eventuali sostituzioni o test unitari.

L'accesso ai DAO all'interno dell'applicazione non avviene in modo diretto, ma è centralizzato attraverso una classe denominata *DAOProvider*. Questo componente funge da punto di accesso unico (simile a un *factory singleton*) per l'ottenimento delle istanze DAO, garantendo una gestione coerente e facilitando eventuali modifiche future nel meccanismo di creazione o nella logica di accesso ai dati.

5.2 Design pattern utilizzati

Il sistema è stato progettato seguendo il pattern architetturale *Model-View-Controller* (MVC), una struttura ampiamente adottata nello sviluppo di interfacce grafiche per separare le responsabilità delle diverse componenti software.

Il pattern MVC suddivide l'applicazione in tre componenti principali:

- **Model:** rappresenta la logica e lo stato dell'applicazione. Contiene i dati, le classi di dominio e la logica di accesso ai dati (DAO), indipendente dalla logica di presentazione.
- **View:** è la parte visiva dell'applicazione, ovvero ciò che l'utente vede e con cui interagisce. In JavaFX, questa è rappresentata dai file `.fxml`, dai fogli di stile CSS e dai componenti grafici associati.
- **Controller:** gestisce l'interazione tra Model e View. Riceve gli input dell'utente dalla View, elabora le richieste e aggiorna il Model o la View di conseguenza. I controller in JavaFX implementano l'interfaccia `Initializable` per gestire l'inizializzazione della scena.

L'utilizzo del pattern MVC risulta particolarmente adatto in combinazione con JavaFX, poiché questa libreria supporta nativamente la separazione tra interfaccia grafica (FXML), logica di controllo (Controller) e dati (Model). In questo modo, il codice risulta più modulare, testabile e manutenibile.

5.3 Diagramma UML delle classi - Utility ed Exception

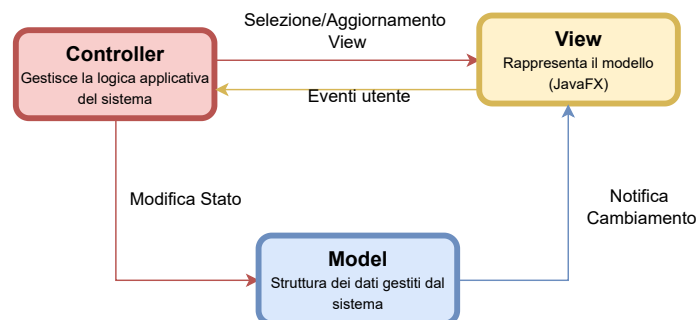


Figura 19: Pattern MVC

5.4 Struttura del progetto

[Albero dei file principali]

6 Fase di Test

6.1 JUnit Test

6.2 Test degli sviluppatori

6.3 Test soggetti esterni