

Anno accademico 2022/2023  
CORSO DI PROGETTAZIONE E ALGORITMI

# **WePlay**

Documentazione di progetto

**Irene Bravo** matr. 1071820

**Samuel Locatelli** matr. 1054674

**Greta Marrone** matr. 1058513



# Capitolo 1

## Iterazione 0

### 1.1 Requisiti

Si vuole realizzare un sistema software che offra un servizio di sharing musicale, unito alle funzionalità tipiche di un social network. Alla registrazione, l'utente dovrà inserire una lista di generi musicali preferiti. L'algoritmo proporrà quindi all'utente i brani più popolari basati sulle sue preferenze e gli permetterà di esplorare le sue affinità musicali. Gli attori in gioco sono i seguenti:

- utenti che vogliono ascoltare musica, conoscere e condividere le proprie attività musicali con gli amici;
- artisti indipendenti e non per pubblicare e gestire la propria musica.

L'utente, per poter usufruire dei servizi forniti, deve registrarsi e creare un proprio profilo personale. Una volta iscritto, verrà indirizzato alla homepage che mostra i brani più popolari dei generi preferiti dall'utente, la possibilità di cercare un brano specifico e la sezione di impostazioni personali e social. Quando viene selezionato il brano da scaricare, vengono fornite tutte le informazioni relative (titolo, artista, album, anno di uscita, genere di appartenenza, numero di ascolti, durata del brano, casa discografica).

Il software offre anche la possibilità di mettere "like" ai brani e di creare playlist personalizzate. Nella propria libreria saranno quindi presenti le playlist dell'utente e la playlist dei brani preferiti. L'interfaccia social permette invece di aggiungere gli amici e di chattare con essi.

La funzionalità principale offerta dal servizio è l'algoritmo "Discover", che consente di trovare i brani più apprezzati dalle persone che ascoltano gli stessi generi musicali dell'utente così da permettergli di scoprire sempre nuove canzoni.

L'artista, a differenza dell'utente ascoltatore, può caricare brani singoli e album inserendo i dati relativi alla propria produzione artistica.

L'utente, per poter usufruire dei servizi forniti, deve registrarsi e creare un proprio profilo personale. Una volta iscritto, verrà indirizzato alla homepage che mostra i brani più popolari dei generi preferiti dall'utente, la possibilità di cercare un brano specifico e la sezione di impostazioni personali e social. Quando viene selezionato il brano da scaricare, vengono fornite tutte le informazioni relative (titolo, artista, album, anno di uscita, genere di appartenenza, numero di ascolti, durata del brano, casa discografica).

Il software offre anche la possibilità di mettere "like" ai brani e di creare playlist personalizzate. Nella propria libreria saranno quindi presenti le playlist dell'utente e la playlist dei brani preferiti. L'interfaccia social permette invece di aggiungere gli amici e di chattare con essi.

La funzionalità principale offerta dal servizio è l'algoritmo "Discover", che consente di trovare i brani più apprezzati dalle persone che ascoltano gli stessi generi musicali dell'utente così da permettergli di scoprire sempre nuove canzoni.

L'artista, a differenza dell'utente ascoltatore, può caricare brani singoli e album inserendo i dati relativi alla propria produzione artistica.

## 1.2 Studio di fattibilità

Procedendo con una valutazione dei costi e dei benefici della possibile realizzazione del presente sistema, è possibile concludere che i costi principali sono relativi alla funzionalità di Discover e brani consigliati.

Per realizzare questa funzionalità è necessaria l'implementazione di un algoritmo che vada a studiare i gusti e le preferenze di ogni singolo utente e, dopo la preliminare fase di raccolta dati, proporrà all'utente una serie di brani in linea con le sue preferenze.

Per quanto riguarda invece la fattibilità in termini di tecnologie e strumenti per la realizzazione del sistema il costo principale da considerare è quello relativo allo sviluppo del software e il servizio di hosting sia della piattaforma che del database.

## 1.3 Casi d'uso

Di seguito vengono riportati i casi d'uso raggruppati in base all'attore coinvolto. Come anticipato gli attori in gioco sono i seguenti:

- Utente ascoltatore;
- Artista;

### 1.3.1 Casi d'uso utente

- **UC1: Sign up**

L'utente può registrarsi inserendo le sue informazioni personali.

- **UC2: Log in**

L'utente può accedere al proprio profilo personale inserendo nome utente e password.

- **UC3: Log out**

L'utente può disconnettere il proprio profilo personale dal sistema.

- **UC4: Cerca brano**

L'utente può digitare nella barra di ricerca per cercare un brano dal titolo.

- **UC5: Cerca album**

L'utente può digitare nella barra di ricerca per cercare un album dal titolo.

- **UC6: Cerca Artista**

L'utente può digitare nella barra di ricerca per cercare un artista dal nome.

- **UC7: Scarica brano**

L'utente può scaricare il brano.

- **UC8: Like al brano**

L'utente può esprimere il proprio interesse per un brano mettendo like.

- **UC9: Aggiungi brano a playlist**  
L'utente può aggiungere alla lista di brani della playlist un brano selezionato.
- **UC10: Cerca Utente**  
L'utente può cercare un utente per nome dalla barra di ricerca apposita (cerca album/artista).
- **UC11: Aggiungi Utente**  
L'utente può aggiungere alla propria lista amici un utente selezionato.
- **UC12: Visualizza informazioni profilo**  
L'utente può consultare le proprie informazioni personali inserite in fase di registrazione.
- **UC13: Modifica profilo**  
L'utente può modificare le proprie informazioni personali inserite in fase di registrazione.
- **UC14: Elimina profilo**  
L'utente può eliminare il proprio profilo dal sistema.
- **UC15: Visualizza playlist**  
L'utente può consultare tutte le proprie playlist.
- **UC16: Crea nuova playlist**  
L'utente può creare una nuova playlist nella quale aggiungere brani.
- **UC17: Elimina playlist**  
L'utente può eliminare una delle proprie playlist.
- **UC18: Modifica playlist**  
L'utente può modificare una delle proprie playlist, per esempio modificando il nome o rimuovendo brani.
- **UC19: "Discover" e classifiche**  
L'utente può consultare classifiche e brani scelti per lui dal sistema.

### 1.3.2 Casi d'uso artista

- **UC20: Crea pagina artista**

L'artista per poter pubblicare musica deve creare una pagina artista inserendo le proprie informazioni personali.

- **UC21: Visualizza pagina artista**

L'artista può consultare la propria pagina artista, dove vengono visualizzati i propri brani/album pubblicati.

- **UC22: Aggiungi brano**

L'artista può pubblicare un nuovo brano inserendo le informazioni relative.

- **UC23: Aggiungi album**

L'artista può pubblicare un nuovo album contenente i brani scelti.

- **UC24: Personalizza pagina artista**

L'artista può decidere quali brani e album mettere in evidenza nella propria pagina artista.

- **UC25: Consulta anagrafica**

L'artista può consultare l'anagrafica relativa agli ascolti e ai like dei brani e album pubblicati.

### 1.3.3 Dettaglio casi d'uso

Di seguito verranno analizzati più nel dettaglio i casi d'uso sopra citati, identificandone descrizione, attore e passi principali.

Gli utenti avranno la possibilità di registrarsi al sito per poter fruire dei servizi di streaming di musica. Al momento della registrazione sarà necessario inserire alcune informazioni personali di base come nome, cognome, email. Inoltre si dovrà inserire un nome utente identificativo e una password, necessari per il login. Nel caso in cui l'utente sia già registrato sarà necessario inserire solo questi campi per accedere al sistema.

Al termine dell'iscrizione è possibile navigare nella Home page, la quale si suddivide in varie sezioni; ad esempio, viene offerta una selezione dei brani più ascoltati e scaricati dagli utenti.

Una volta scelto il brano, per ognuno di essi vengono visualizzate alcune informazioni come il titolo del brano, il nome dell'artista e l'etichetta discografica di riferimento.

Alternativamente, è possibile cercare un brano dalla barra di ricerca (tramite il titolo, nome dell'artista o nome dell'album) e scaricarlo.

È offerta la possibilità di mettere like al brano e creare playlist personalizzate con i propri brani preferiti: visualizzando la propria libreria, infatti, è possibile visualizzare i Brani Preferiti e le proprie playlist.

Come anticipato il sistema offre anche funzionalità di un social network: infatti dalla barra di ricerca è possibile cercare per nome utente i propri amici e aggiungerli, in modo tale da poter condividere le proprie preferenze musicali.

Una delle funzionalità principali offerte dal sistema è chiamata Discover e consiste nel poter usufruire di playlist create ad hoc dal sistema per l'utente, tramite lo studio delle sue preferenze.

Gli artisti potranno gestire e pubblicare la propria musica tramite varie funzionalità offerte dal sistema. Al momento della registrazione bisogna seguire la procedura classica di sign-in, in seguito in una sezione apposita è offerta la possibilità di creare la pagina artista, all'interno della quale verrà inserito il nome che apparirà pubblicamente, insieme ad altre informazioni personali.

Dopo aver creato il proprio profilo artista si potrà accedere alla propria pagina per pubblicare le proprie tracce, album, playlist, personalizzandola. Ogni volta che viene pubblicato un brano è possibile inserire eventuali nomi di collaboratori, artisti che hanno partecipato al brano e scrittori di testo e musica. Ogni artista può scegliere la data dalla quale rendere accessibile il brano o l'album.



### 1.3.4 Diagramma dei casi d'uso

In figura 1.1 è stato riportato il diagramma dei casi d'uso, dove sono stati inseriti quelli già precedentemente descritti mettendoli in correlazione con il corrispettivo attore.

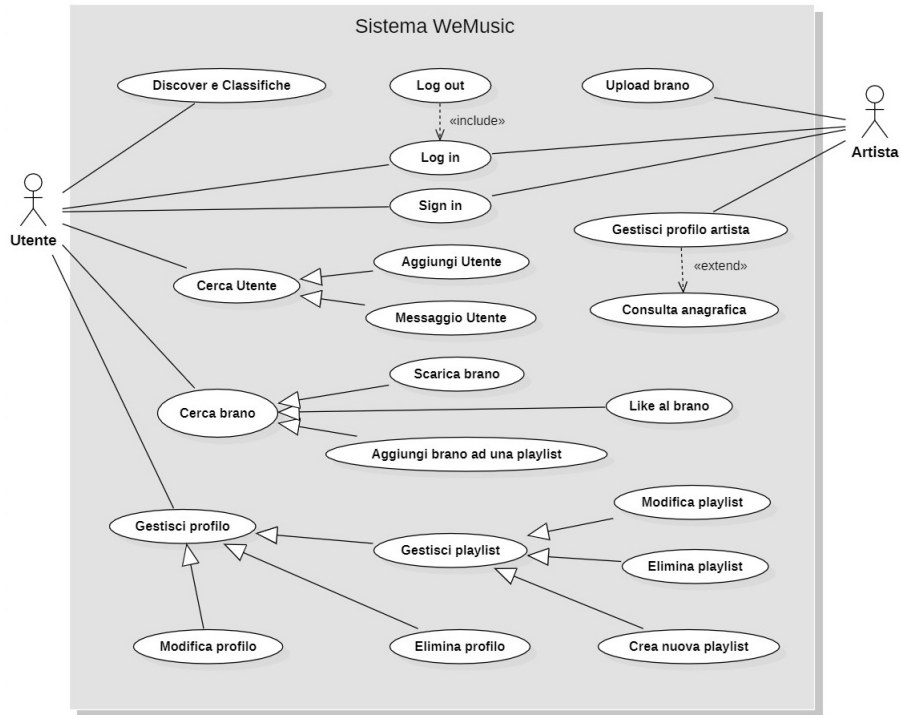


Figura 1.1: UML Use Cases Diagram

## 1.4 Architettura del sistema

L'architettura del sistema è stata formalizzata attraverso la rappresentazione tramite Deployment Diagram, diagramma di tipo statico sviluppato al fine di modellizzare e descrivere un sistema in termini di risorse hardware e di relazioni fra di esse. Vengono sviluppatidue Deployment Diagram differenti:

il primo, riportato in figura 1.2, rappresenta il sistema attraverso una notazione a stile libero, mentre il secondo, riportato in figura 1.3, rappresenta la topologia del sistema tramite lo stile UML.

### 1.4.1 Deployment diagram – Informal

In figura 1.2 viene evidenziata la componente principale del sistema, il sistema gestore (WeMusic) il Web Server (sviluppato tramite il framework Django) che, tramite una connessione ad internet, permette agli utenti di poter consultare il sito.

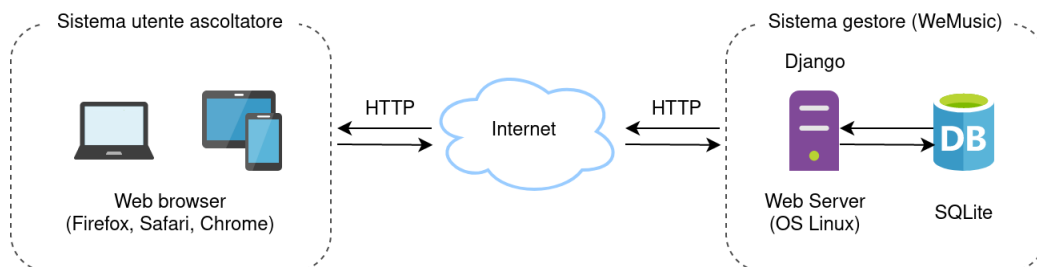


Figura 1.2: Architettura

### 1.4.2 Deployment diagram – UML

In figura 1.3 vengono evidenziati due nodi principali: un lato client che gestirà il front-end del sistema WeMusic, e un lato server relativo al back-end e alla gestione del DataBase, in modo da velocizzare e semplificare l'interazione tra logica e dati.

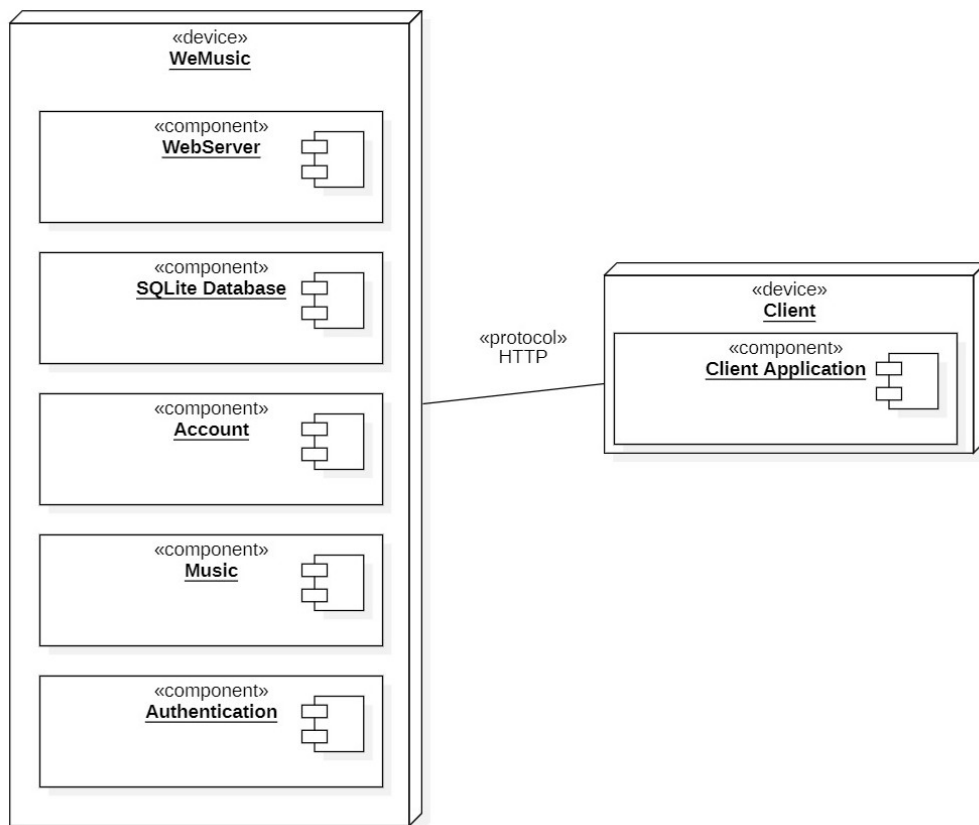


Figura 1.3: Deployment Diagram

## 1.5 Toolchain e tecnologie utilizzate

- **Modellazione:** Star UML (Use Case Diagram, Deployment Diagram); diagram.net (Component Diagram, Class Diagram).
- **Implementazione – lato client-server:** Django,
- **Implementazione – parte dati db:** SQLite.
- **Documentazione, versioning e organizzazione del team:** Latex, GitHub, Discord.

# Capitolo 2

## Iterazione 1

### 2.1 Introduzione

L'iterazione 1 ha come scopo quello di identificare le componenti dal modello dei casi d'uso, applicando le euristiche di early design: verrà perfezionata la specifica dei componenti progettati durante l'iterazione 0 al fine di definire meglio l'architettura software. È stato costruito lo scheletro dell'applicazione tramite la specifica delle differenti classi popolate nelle seguenti iterazioni.

Inizialmente, durante lo sviluppo dell'iterazione 0, il sistema nella sua totalità è stato visualizzato come una unica componente e sono stati introdotti gli attori, ognuno dei quali definito come un'entità esterna. In seguito sono stati sviluppati tutti i casi d'uso che riassumono il funzionamento del sistema, i quali verranno, nell'Iterazione 1, raccolti e raggruppati secondo un preciso criterio e affinità.

Vengono inoltre introdotti componenti e sottocomponenti di controllo di ciascun gruppo, o alternativamente componenti di dati; per ognuna di esse nascono parallelamente le classi candidate e le relazioni che le legano.

Per ogni caso d'uso che ha una interazione diretta con un attore esterno viene introdotta un'interfaccia per le operazioni visibili esternamente, ovvero le API, offerte o richieste dal componente o sottocomponente corrispondente, in base alla direzione dell'interazione.

Ogni variabile input da un attore, specificata nella descrizione di un caso d'uso, definisce un parametro di input dell'operazione dell'interfaccia corrispondente. Ogni variabile output per un attore specificata nella descrizione di un caso d'uso definisce un parametro di ritorno dell'operazione dell'interfaccia corrispondente (se modalità sincrona) o un input di un'operazione dell'interfaccia di callback (se modalità asincrona) dell'attore.

Viene scomposto il sistema in sottosistemi e componenti, applicando pattern e stili architetturali, per poi distribuire le componenti su nodi computazionali basati su uno sviluppo fisico del sistema.

## 2.2 Casi d'uso

I casi d'uso vengono raggruppati in tre macro categorie in base a un criterio di affinità: i primi sono relativi all'autenticazione dell'utente, i secondi sono relativi a tutto ciò che riguarda l'ambito musicale, ed infine quelli relativi all'account personale e lato social. Di seguito un elenco dettagliato dei casi d'uso suddivisi come anticipato.

### 2.2.1 Autenticazione

- **UC1:** Sign up
- **UC2:** Log in
- **UC3:** Log out

### 2.2.2 Musica

- **Brano:** in questa sezione rientrano tutti i casi d'uso relativi ai brani.
  - **UC4:** Cerca brano
  - **UC5:** Cerca album
  - **UC6:** Cerca Artista
  - **UC7:** Scarica brano
  - **UC8:** Like al brano
  - **UC19:** “Discover” e classifiche
- **Playlist:** in questa sezione rientrano tutti i casi d'uso relativi alla creazione e gestione delle playlist.
  - **UC9:** Aggiungi brano a playlist
  - **UC15:** Visualizza playlist
  - **UC16:** Crea nuova playlist
  - **UC17:** Elimina playlist
  - **UC18:** Modifica playlist
- **Artista:** in questa sezione rientrano tutti i casi d'uso relativi alla gestione del profilo artista.
  - **UC20:** Crea pagina artista

- **UC21:** Visualizza pagina artista
- **UC22:** Aggiungi brano
- **UC23:** Aggiungi album
- **UC24:** Personalizza pagina artista
- **UC25:** Consulta anagrafica

### 2.2.3 Account

- **Personale:** in questa sezione rientrano tutti i casi d'uso relativi alla gestione delle informazioni nel proprio profilo personale.
  - **UC12:** Visualizza informazioni profilo
  - **UC13:** Modifica profilo
  - **UC14:** Elimina profilo
- **Amici:** in questa sezione rientrano tutti i casi d'uso relativi alla parte social.
  - **UC10:** Cerca Utente
  - **UC11:** Aggiungi Utente

## 2.3 System Diagram

In figura sono rappresentate tutte le interfacce richieste ed esposte dal nostro sistema. Inoltre, i casi d'uso sono stati raggruppati indicando il nome dell'interfaccia che permetterà di implementare il particolare caso d'uso.



## 2.4 UML Component Diagram

Il Component Diagram in UML ha come obiettivo quello di mostrare la struttura del sistema software, descrivendo i singoli componenti, le relative interfacce e le dipendenze. La rappresentazione del sistema è stata gestita partendo da una una componente centrale, il server del sistema, alla quale si collegano il database e la parte front-end.

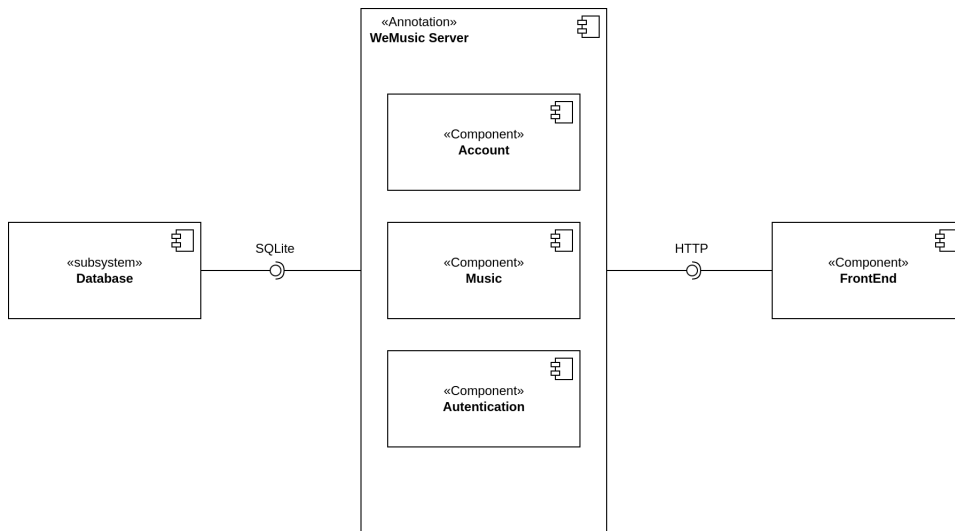


Figura 2.1: UML Component Diagram

## 2.5 UML Class Diagram

Il Class Diagram in UML ha come obiettivo quello di descrivere il sistema visualizzando i diversi tipi di oggetti all'interno di esso e le relazioni statiche che esistono fra loro: sono descritte in maniera più approfondita le tre componenti già precedentemente analizzate, ovvero Account, Musica e Autenticazione. In questo caso vengono rappresentati in unico diagramma il Data Class Diagram e l'interface/Package Diagram. Il primo descrive le classi che compongono la struttura dell'applicazione delle entità presenti al suo interno; il secondo rappresenta il diagramma dei package che riassumono la struttura logica dell'applicazione e delle interfacce che verranno implementate nelle successive iterazioni. Vengono anche illustrate le operazioni e gli attributi delle classi.

## 2.5.1 Data Class Diagram

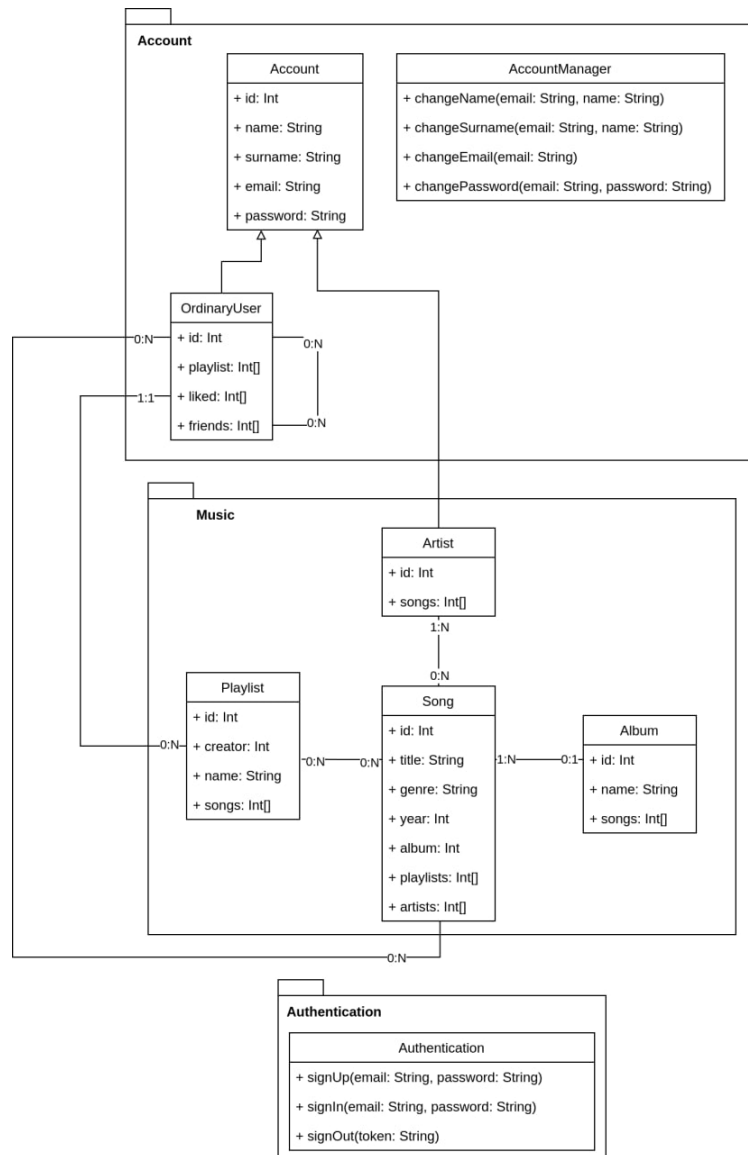


Figura 2.2: UML Data Class Diagram

## **2.5.2 Interface and Package Class Diagram**

Figura 2.3: UML Interface and Package Class Diagram

# Capitolo 3

## Iterazione 2

### 3.1 Introduzione

Nella seconda iterazione sono stati implementati la maggior parte dei casi d'uso: viene proposta un'analisi dettagliata di ognuno al fine di specificare gli attori coinvolti nel caso d'uso, l'esecuzione standard dello stesso, le eventuali precondizioni e le postcondizioni, le eccezioni. In seguito si procede al testing, quindi analisi statica e dinamica tramite(...)

## 3.2 Implementazione

Di seguito sono elencati i casi d'uso implementati nel sistema WeMusic. Si offre, per ognuno, una breve descrizione dello stesso al fine di riassumere la sua funzionalità; inoltre è evidenziato il flow di esecuzione del sito e le varie alternative che si propongono nel modello. Infine, vengono specificate eventuali informazioni relative a precondizioni, post condizioni e trigger dello stesso, e le possibili eccezioni nel caso avvenga una esecuzione scorretta. Seguendo la modellazione proposta in seguito, i casi d'uso relativi all'Autenticazione sono stati completamente implementati, come anche la maggior parte di quelli facenti parte del gruppo Musica e Amici.

### 3.2.1 UC1 – Sign up

L'operazione di Sign up consente all'utente di registrare al sito e di poter usufruire dei servizi offerti dallo stesso.

- **Actor:** Utente base, Artista.
- **Precondition:** Nessuna.
- **Postcondition:** L'utente è registrato al sito, le sue informazioni vengono salvate nel database.
- **Base Sequence:**
  - 1. L'utente inserisce nome utente e password.
  - 2. L'utente inserisce i generi preferiti.
  - 3. L'utente ha completato la registrazione.
- **Branch Sequence:** Nessuna.
- **Exception Sequence:** Il nome utente scelto non è valido o è già in uso.

### 3.2.2 UC2 – Log in

L'operazione di Log in permette ad un utente già registrato nel sistema di accedere e di poter usufruire dei servizi offerti dallo stesso.

- **Actor:** Utente base, Artista.
- **Precondition:** L'utente deve essere già registrato.
- **Postcondition:** L'utente ha effettuato il log in perciò può navigare la Home page del sito.
- **Base Sequence:**
  - 1. L'utente inserisce nome utente e password.
  - 2. L'utente clicca sul pulsante di log in.
  - 3. L'utente ha effettuato correttamente l'accesso.
- **Branch Sequence:** Nessuna.
- **Exception Sequence:** La password inserita è errata.

### 3.2.3 UC3 – Log out

L'operazione di Log out permette all'utente di disconnettere il proprio profilo personale dal sistema.

- **Actor:** Utente base, Artista.
- **Precondition:** L'utente deve essere registrato e deve aver effettuato il log in.
- **Postcondition:** L'utente ha effettuato il log out, perciò non ha più accesso al sistema fino al log in.
- **Base Sequence:**
  - 1. L'utente clicca il pulsante di Log out nell'interfaccia.
  - 2. L'utente è disconnesso.
- **Branch Sequence:** Nessuna.
- **Exception Sequence:** Nessuna.

### 3.2.4 UC4 – Cerca brano

Il presente caso d'uso permette di cercare un brano specifico dalla barra di ricerca tramite precise parole chiave, al fine di scaricarlo per poi riprodurlo in locale.

- **Actor:** Utente base.
- **Precondition:** L'utente deve aver effettuato il log in.
- **Postcondition:** L'utente ha cercato il brano, può dunque mettere like, aggiungerlo ad una playlist personale o scaricarlo.
- **Base Sequence:**
  - 1. L'utente clicca sulla barra di ricerca apposita nell'interfaccia.
  - 2. L'utente digita delle parole chiave per cercare il brano: titolo del brano, album, artista.
  - 3. L'utente naviga fra i risultati della ricerca e seleziona il brano desiderato.
- **Branch Sequence:** L'utente può scaricare il brano (UC7), mettere like al brano (UC8), aggiungerlo ad una playlist (UC9).
- **Exception Sequence:** Non esiste un brano che corrisponde ai criteri di ricerca.

### 3.2.5 UC5 – Cerca album

- **Actor:** Utente base
- **Precondition:** L'utente deve aver effettuato il log in.
- **Postcondition:** L'utente ha cercato l'album, può dunque navigare fra i brani che include lo stesso.
- **Base Sequence:**
  - 1. L'utente clicca sulla barra di ricerca apposita nell'interfaccia.
  - 2. L'utente digita delle parole chiave per cercare l'album: titolo dell'album, artista.



- **3.** L'utente naviga fra i risultati della ricerca e seleziona l'album desiderato.
- **Branch Sequence:** L'utente può navigare fra i brani e scaricarli (UC7), mettere like (UC8), aggiungerli ad una playlist (UC9).
- **Exception Sequence:** Non esiste un album che corrisponde ai criteri di ricerca.

### 3.2.6 UC6 – Cerca artista

- **Actor:**
- **Precondition:**
- **Postcondition:**
- **Base Sequence:**
- **Branch Sequence:**
- **Exception Sequence:**
- **Sub UseCase:**

### 3.2.7 UC7 – Scarica brano

- **Actor:**
- **Precondition:**
- **Postcondition:**
- **Base Sequence:**
- **Branch Sequence:**
- **Exception Sequence:**
- **Sub UseCase:**

### **3.2.8 UC8 – Like al brano**

- Actor:
- Precondition:
- Postcondition:
- Base Sequence:
- Branch Sequence:
- Exception Sequence:
- Sub UseCase:

### **3.2.9 UC9 – Aggiungi brano a playlist**

- Actor:
- Precondition:
- Postcondition:
- Base Sequence:
- Branch Sequence:
- Exception Sequence:
- Sub UseCase:

### **3.2.10 UC15 – Visualizza playlist**

- Actor:
- Precondition:
- Postcondition:
- Base Sequence:
- Branch Sequence:
- Exception Sequence:
- Sub UseCase:

### **3.2.11 UC16 – Crea nuova playlist**

- Actor:
- Precondition:
- Postcondition:
- Base Sequence:
- Branch Sequence:
- Exception Sequence:
- Sub UseCase:

### **3.2.12 UC17 – Elimina playlist**

- Actor:
- Precondition:
- Postcondition:
- Base Sequence:
- Branch Sequence:
- Exception Sequence:
- Sub UseCase:

### **3.2.13 UC18 – Modifica playlist**

- Actor:
- Precondition:
- Postcondition:
- Base Sequence:
- Branch Sequence:
- Exception Sequence:
- Sub UseCase:

### 3.2.14 UC10 – Cerca Utente

- Actor:
- Precondition:
- Postcondition:
- Base Sequence:
- Branch Sequence:
- Exception Sequence:
- Sub UseCase:

### 3.2.15 UC11 – Aggiungi Utente

- Actor:
- Precondition:
- Postcondition:
- Base Sequence:
- Branch Sequence:
- Exception Sequence:
- Sub UseCase:

### 3.2.16 UC12 – Visualizza informazioni profilo

- Actor:
- Precondition:
- Postcondition:
- Base Sequence:
- Branch Sequence:
- Exception Sequence:
- Sub UseCase:

### 3.2.17 UC13 – Modifica profilo

- Actor:
- Precondition:
- Postcondition:
- Base Sequence:
- Branch Sequence:
- Exception Sequence:
- Sub UseCase:

### 3.2.18 UC14 – Elimina profilo

- Actor:
- Precondition:
- Postcondition:
- Base Sequence:
- Branch Sequence:
- Exception Sequence:
- Sub UseCase:

### 3.2.19 UC19 – “Discover” e classifiche

- Actor:
- Precondition:
- Postcondition:
- Base Sequence:
- Branch Sequence:
- Exception Sequence:
- Sub UseCase:

# Capitolo 4

## Iterazione 3

### 4.1 Introduzione

ALGORITMO