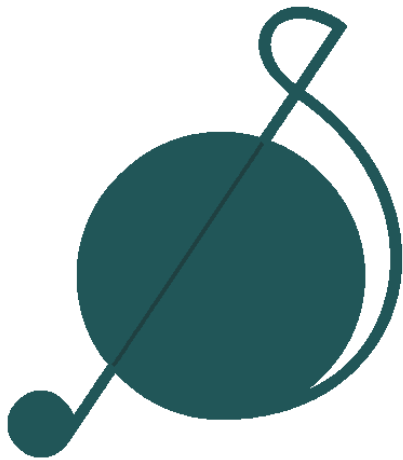




Laurea Triennale in informatica-Università di Salerno
Corso di
Ingegneria del software- Prof. Carmine Gravino



Spotibase

Requirement Analysis Document

Spotibase

Docente

Gravino Carmine

Studenti

Silvio Venturino	0512108837
Umberto Della Monica	0512107742
Gagliarde Nicolapio	0512106980
Alessandro Aquino	0512106527



Revision History

Data	Versione	Descrizione	Autori
15/10/21	0.1	Creazione documento, copertina e sommario	NG
22/10/21	0.2	Prima stesura degli scenari	Tutto il team
23/10/21	0.3	Prima stesura requisiti funzionali, non funzionali e pseudo-funzionali	Tutto il team
24/10/21	0.4	Prima stesura dell'introduzione	AA
25/10/21	0.5	Aggiunta dei riferimenti, panoramica del documento	AA, UM
26/10/21	0.6	Prima stesura degli use case	Tutto il team
27/10/21	0.7	Seconda stesura degli use case e controllo scenari	Tutto il team
28/10/21	0.8	Controllo e correzione use case	Tutto il team
29/10/21	0.9	Aggiunta use case diagram, prima individuazione degli oggetti entity, control e boundary	Tutto il team
02/11/21	0.10	Individuazione degli oggetti entity, control e boundary	Tutto il team
04/11/21	0.11	Correzione errori di formattazione	Tutto il team
05/11/21	0.12	Aggiunta del sequence diagram: SD_UC_9	Tutto il team



11/11/21	0.13	Aggiunta dei sequence diagram: SD_UC_2, SD_UC_3, SD_UC_8	Tutto il team
15/11/21	0.14	Aggiunta statechart	AA, NG
16/11/21	0.15	Aggiunta class diagram, mockup e navigational path	Tutto il team
19/11/21	0.16	Aggiunta glossario	NG
22/11/21	1.0	Revisione del documento, correzione errori e modifiche varie	Tutto il team
26/11/21	1.1	Revisione della sezione 3.1	UM, SV
06/12/21	1.2	Correzione utente guest e utente loggato in utente.	Tutto il team

Team Members

Nome	Ruolo nel progetto	Email	Acronimo
Alessandro Aquino	Team member	a.aquino33@studenti.unisa.it	AA
Nicolapio Gagliarde	Team member	n.gagliarde@studenti.unisa.it	NG
Umberto Della Monica	Team member	u.dellamonica2@studenti.unisa.it	UM
Silvio Venturino	Team member	s.venturino@studenti.unisa.it	SV



Sommario

1. Introduzione	5
1.1. Scopo del sistema	5
1.2. Ambito del sistema	5
1.3. Obiettivi e criteri di successo del sistema	6
1.4. Definizioni, acronimi e abbreviazioni	6
1.5. Riferimenti	7
1.6. Panoramica	7
2. Sistema corrente	9
3. Sistema proposto	10
3.1 Panoramica	10
3.2 Requisiti funzionali	11
3.3 Requisiti non funzionali	14
3.3.1 Usabilità	14
3.3.2 Affidabilità	14
3.3.3 Prestazioni	14
3.3.4 Sostenibilità	15
3.3.5 Pseudo-funzionali	15
3.4 Modelli di sistema	17
3.4.1 Scenari	17
3.4.2 Use Case model	25
3.4.3 Object model	39
3.4.4 Dynamic model	50
3.4.5 User interface - navigational paths e mock-ups	56
4. Glossario	666



1.Introduzione

1.1 Scopo del sistema

Spotibase intende fornire agli utenti una piattaforma musicale che offre lo streaming on demand di canzoni.

Inoltre offre i servizi fondamentali di un e-commerce, in quanto è possibile acquistare in formato digitale (mp3...) canzoni e album.

Lo scopo è quello di soddisfare le preferenze di ogni utente fornendo un' ampia scelta di canzoni musicali di tutti i generi, con la possibilità di gestire delle playlist, sempre più vicino alle esigenze degli utenti.

1.2 Ambito del sistema

L'ambito del sistema è quello di una piattaforma musicale integrata con i servizi base di un e-commerce assicurando che tutti gli stakeholder coinvolti possano interagire in modo agevole ed efficiente.

Le operazioni e funzionalità principali offerte dal sistema sono le seguenti:

1. Riproduzione di canzoni.
2. Gestione playlist: creazione/eliminazione playlist, aggiunta/rimozione canzoni.
3. La possibilità di acquistare canzoni ed album.

Il sistema non supporta:

1. La possibilità degli artisti di aggiungere canzoni/album sulla piattaforma.
2. Il riconoscimento delle canzoni tramite microfono del dispositivo che si sta utilizzando.
3. I videoclip delle canzoni presenti sulla piattaforma ma unicamente formati audio.



1.3 Obiettivi e criteri di successo del sistema

Gli obiettivi principali del progetto “Spotibase” sono quelli di:

- Riproduzione di canzoni in modo semplice ed intuitivo
- Acquisto di canzoni/album in pochi passaggi per velocizzare le procedure
- Download di canzoni per usufruirne anche in assenza di rete.
- Ottimizzazione di scelta con la creazione di playlist per riprodurre le canzoni che interessano di più al singolo utente.

Criteri di successo del sistema:

- Branch coverage dei casi di test: almeno 75%.
- Buona manutenibilità e di integrità.

1.4 Definizioni, acronimi e abbreviazioni

Definizioni

Canzone: breve composizione musicale.

Album: insieme di brani/canzoni che appartengono allo stesso artista.

Utente: utilizzatore della piattaforma.

Playlist: insieme di canzoni che non appartengono allo stesso artista ma scelte dall'utente.

Artista: creatore di canzoni e/o album.

Preferenza: scelta di un utente secondo al gusto soggettivo di valutazione

Admin: gestore della piattaforma.

Carrello: insieme di canzoni destinate all'acquisto



Acronimi

RAD = Requirement Analysis Document (Documento di Raccolta dei Requisiti)

RF = Requisito Funzionale

RNF = Requisito Non Funzionale

UC = Caso d'Uso

SC= Scenario

U= Utente

AD = Activity Diagram

OD = Object Diagram

SCD = Statechart Diagram

NA = Nessuna

1.5 Riferimenti

Libro:

- Object-Oriented Software Engineering (Using UML, Patterns, and Java) Third Edition

Autori:

- Bernd Bruegge & Allen H. Dutoit

1.6 Panoramica del documento

Il seguente documento “RAD” è diviso in sezioni ed ha la seguente composizione:

Sezione INTRODUZIONE:

Vi sarà presentata una breve descrizione del progetto e dall'esigenza da cui è nato.

Viene fornito e descritto il contesto di utilizzo del sistema per poi passare nel fornire gli obiettivi del sistema ed i criteri di successo dell'intero progetto. Successivamente vengono citate definizioni, acronimi e abbreviazioni. Utilizzate per facilitare il lettore a ricordare le parole chiave (con acronimi o abbreviazione) e capire concetti tecnici tramite le definizioni.



Sezione **SISTEMA CORRENTE:**

Questa sezione è dedicata alla spiegazione di come è la realtà attuale, prima dello sviluppo del sistema.

Sezione **SISTEMA PROPOSTO:**

Questa sezione del documento parte con una sottosezione di introduzione nella quale viene fornita una descrizione dell'idea di base di come il sistema dovrebbe essere.

Si procede con la sottosezione dei Requisiti Funzionali del sistema dove vengono identificate le funzionalità che il sistema deve offrire. I requisiti funzionali seguono questo tipo di convenzione: RF[numero] - nomeDelRequisitoFunzionale - descrizione - priorità).

La sezione del SISTEMA PROPOSTO continua con i Requisiti Non Funzionali, che seguono questo tipo di convenzione: NFR[numero] - descrizione.

I Requisiti Non Funzionali seguono il modello **FURPS+**, essi sono: Usabilità, Affidabilità, Prestazioni, Sostenibilità, Implementazione, Interfaccia, Packaging, Operazioni e Legali.

La composizione del documento segue con i **MODELLI DINAMICI**: Qui vi sono i diagrammi di sequenza o diagrammi delle attività.

La sezione SISTEMA PROPOSTO si conclude con l'**INTERFACCIA UTENTE: MOCK-UP E NAVIGATIONAL PATH.**

Viene definito il design dell'interfaccia utente e le varie pagine dell'applicazione usando i relativi mockups.

Il documento "RAD" si conclude con il **GLOSSARIO**, dove sono specificati i termini utilizzati nel documento per evitare ambiguità.



2.Sistema corrente

Attualmente esistono sistemi simili che permettono la riproduzione streaming on demand di canzoni come Spotify, Deezer, Amazon Music.... Questi sistemi anche se gestiscono la riproduzione e ascolto offline non hanno una vera e propria sezione e-commerce, quindi con la possibilità di aggiungere al carrello e acquistare i file mp3.

Dall'analisi dei competitor otteniamo le seguenti informazioni:

- la creazione delle playlist avviene secondo i seguenti passi:
 - L'utente clicca "Crea playlist"
 - Poi clicca "Inserisci titolo"
 - Poi clicca "Inserisci Immagine"
 - Poi clicca "Inserisci descrizione"
 - Poi clicca "Crea playlist"
- la riproduzione dei brani avviene tramite il click dell'utente sull'apposita icona, presente sulla copertina della canzone;
- la preferenza la si può esprimere tramite il click sull'apposita icona, presente dopo il numero di riproduzioni totali;
- per aggiungere canzoni alla playlist, l'utente deve andare sulla pagina della playlist, cercare le canzoni e usare il tasto "aggiungi"



3. Sistema Proposto

3.1 Panoramica

Il sistema da noi proposto è un'applicazione web con lo scopo di unire le funzionalità di un e-commerce con quelle di una piattaforma musicale. Gli utenti avranno la possibilità di registrarsi al sito tramite la compilazione dell'apposito modulo e di utilizzare ulteriori funzionalità.

Le varie funzionalità che il sito offre in base al tipo di utente sono le seguenti:

- **Funzionalità admin**

- La modifica di: canzoni, album e artisti presenti sul sito;
- Visualizzazione di dati statistici (es. guadagni totali, brani più venduti);

- **Funzionalità utente**

- Aggiungere e rimuovere dal carrello elementi;
- Riproduzione di brani;
- Visualizzazione pagina album, artisti e abbonamenti;
- Ricerca di brani, artisti e album.
- Login e registrazione.
- Possibilità di acquistare abbonamenti per incrementare il numero di playlist creabili;
- Gestione playlist: creazione/eliminazione playlist, aggiunta/rimozione brani;
- Esprimere preferenze per album, artisti e canzoni;
- Visualizzazione pagina libreria;
- Gestione del carrello e possibilità di completare l'acquisto;
- Scaricare mp3 acquistati;
- Logout



3.2 Requisiti funzionali

Lato amministratore

ID	NOME	DESCRIZIONE	PRIORITA'
RF1	Modifica	Il sistema deve dare la possibilità all'amministratore di modificare: canzoni, album e artisti presenti sul sito.	Alta
RF2	Visualizzazione Dati Statistici	Il sistema deve dare la possibilità all'amministratore di visualizzazione di dati statistici (es. guadagni totali, canzoni più vendute)	Alta
RF3	Visualizzazione Utenti	Il sistema dovrebbe dare la possibilità all'amministratore di visualizzare gli utenti registrati nel sistema	Media
RF4	Modifica Abbonamenti	Il sistema dovrebbe dare la possibilità all'amministratore di modificare le caratteristiche degli abbonamenti	Media
RF5	Rimozione Abbonamenti	Il sistema può dare la possibilità all'amministratore di rimuovere un abbonamento dal sistema	Bassa
RF6	Aggiunta Abbonamenti	Il sistema può dare la possibilità all'amministratore di inserire un abbonamento nel sistema	Bassa
RF7	Rimozione utente	Il sistema può dare la possibilità all'amministratore di rimuovere un utente dal sistema	Bassa



Lato utente

ID	NOME	DESCRIZIONE	PRIORITA'
RF8	Aggiunta carrello	Il sistema deve dare la possibilità all'utente di aggiungere al carrello elementi.	Alta
RF9	Rimozione carrello	Il sistema deve dare la possibilità all'utente di rimuovere dal carrello elementi.	Alta
RF10	Riproduzione Canzoni	Il sistema deve dare la possibilità all'utente di riprodurre le canzoni.	Alta
RF11	Visualizzazione pagine	Il sistema deve dare la possibilità all'utente di visualizzare la pagina album, artisti e abbonamenti.	Alta
RF12	Ricerca	Il sistema deve dare la possibilità all'utente di ricercare canzoni, artisti e album.	Alta
RF13	Login	Il sistema deve dare la possibilità all'utente di accedere alla piattaforma.	Alta
RF14	Registrazione	Il sistema deve dare la possibilità all'utente di creare un account	Alta
RF15	Aggiunta canzoni in coda	Il sistema dovrebbe dare la possibilità all'utente di aggiungere canzoni nella coda di riproduzione	Media
RF16	Condivisione canzoni	Il sistema può dare la possibilità all'utente di condividere i link delle canzoni	Bassa
RF17	Visualizzazione Tour Artista	Il sistema può dare la possibilità all'utente di visualizzare le date del tour di un artista	Bassa
RF18	Acquisto Abbonamenti	Il sistema deve dare la possibilità all'utente di acquistare abbonamenti per incrementare il numero di playlist creabili	Alta
RF19	Creazione Playlist	Il sistema deve dare la possibilità all'utente di creare la playlist.	Alta



Laurea Triennale in informatica-Università di Salerno
Corso di
Ingegneria del software- Prof. Carmine Gravino

RF20	Eliminazione Playlist	Il sistema deve dare la possibilità all'utente di eliminare la playlist	Alta
RF21	Aggiunta Canzoni in Playlist	Il sistema deve dare la possibilità all'utente di aggiungere le canzoni in playlist	Alta
RF22	Rimozione Canzoni in Playlist	Il sistema deve dare la possibilità all'utente di rimuovere le canzoni dalla playlist	Alta
RF23	Preferenze	Il sistema deve dare la possibilità all'utente di esprimere preferenze per album, artisti e canzoni	Alta
RF24	Visualizzazione Libreria	Il sistema deve dare la possibilità all'utente di visualizzare pagina libreria	Alta
RF25	Acquisto Carrello	Il sistema deve dare la possibilità all'utente di completare l'acquisto	Alta
RF26	Download Canzoni	Il sistema deve dare la possibilità all'utente di scaricare mp3 acquistati	Alta
RF27	Logout	Il sistema deve dare la possibilità all'utente di effettuare il logout	Alta
RF28	Aggiornamento Password	Il sistema dovrebbe dare la possibilità all'utente di aggiornare la propria password utente.	Media
RF29	Aggiornamento Immagine Profilo	Il sistema può dare la possibilità all'utente di aggiornare un'immagine profilo.	Bassa
RF30	Visualizzazione Testo	Il sistema può dare la possibilità di visualizzare il testo di una canzone.	Bassa



3.3 Requisiti non funzionali

3.3.1 Usabilità

RNF1	<p>Il sistema deve essere semplice ed intuitivo. L'utente si deve sentire a proprio agio utilizzando i servizi a sua disposizione e non deve avere la sensazione di non trovare le funzionalità desiderate. L'utente deve capire cosa sta accadendo dopo aver effettuato un'azione e percepire che la risposta del sistema corrisponda a quello che si aspetta. Il sistema deve dare all'utente un riscontro delle proprie azioni sulla piattaforma. Il sistema dovrà fornire all'utente la possibilità di cambiare i colori dell'interfaccia grafica in base alle proprie preferenze (dark mode).</p> <p>In particolare l'utente dovrà saper utilizzare tutte le funzionalità del sistema dopo 10 minuti di utilizzo.</p>
-------------	--

3.3.2 Affidabilità

RNF2	<p>Il sistema deve mantenere un livello di sicurezza necessario affinché eventuali accessi minacciosi non comportino una perdita di dati sensibili (e-mail o password) o una modifica di essi.</p>
-------------	--

3.3.3 Prestazioni

RNF3	<p>Il sistema deve riuscire a gestire anche picchi di carico dovuti ad un numero di accessi al sistema simultaneo, in modo da garantire una navigazione fluida e veloce (tempi di risposta minori a 1 secondo).</p>
-------------	---



3.3.4 Sostenibilità

RNF4	Il sistema deve essere scalabile, e facilmente adattabile a nuove funzionalità del dominio applicativo. Deve essere di facile integrazione a nuovi sistemi e adattabile a nuove tecnologie. Deve essere inoltre fortemente modulare in modo da riuscire a modificare la logica di business o quella di sistema senza particolari difficoltà.
-------------	--

3.3.5 Pseudo-funzionali:

Implementazione

RNF5	Il software sarà realizzato tramite le tecnologie di sviluppo web (HTML5, CSS, AJAX, JavaScript) e tramite il linguaggio di programmazione Java.
-------------	--

Interfaccia

RNF6	Le informazioni e i dati viaggeranno attraverso la rete. Il sistema utilizzerà JSON che è un formato adatto all'interscambio di dati fra applicazioni client/server
-------------	---

Packaging

RNF7	Il sistema verrà installato su un web server. Gli utenti potranno usufruire del software attraverso i loro browser web preferiti.
-------------	---



Operazioni

Laurea Triennale in informatica-Università di Salerno
Corso di
Ingegneria del software- Prof. Carmine Gravino

RNF9

La gestione del sistema è limitata ad un admin

Legali

RNF10

Il sistema avrà una licenza “Open Source”.



3.4 Modelli di Sistema

3.4.1 Scenari

NOME SCENARIO	SC1_PreferenzaAlbum	
ATTORI	Ugo: utente	
FLUSSO DEGLI EVENTI	ATTORE	SISTEMA
	Ugo ama l'album dei Queen "A Night At The Opera" e decide di aggiungerlo alla sua lista di album preferiti. Ugo clicca sull'album.	
		Spotibase mostra la pagina dell'album con tutte le canzoni che ne fanno parte. Accanto ad ogni canzone è presente la possibilità di aggiungerla tra i preferiti e in fondo alla pagina si può esprimere la preferenza anche per l'album.
	Ugo ascolta ancora una volta l'album e aggiunge la preferenza cliccando sul bottone apposito.	
		Spotibase aggiunge l'album nella libreria di Ugo.



NOME SCENARIO	SC2_AggiuntaCanzonePlaylist	
ATTORI	Jake: utente	
FLUSSO DEGLI EVENTI	ATTORE	SISTEMA
	Jake essendo un amante delle canzoni pop, ha una playlist intitolata “I Love Pop”. Navigando nel sito trova la nuova canzone di Dua Lipa “We’re Good”. Ascoltando la canzone, decide di aggiungerla in una playlist e clicca il pulsante apposito.	
		Spotibase mostra in un alert le playlist di Jake.
	Jake seleziona la sua playlist “I Love Pop”.	
		Spotibase fa comparire a video un alert che garantisce il corretto inserimento della canzone nella playlist.



NOME SCENARIO	SC3_RicercaCanzoni	
ATTORI	Nik: utente	
FLUSSO DEGLI EVENTI	ATTORE	SISTEMA
	Nik si trova nella home del sito e vuole ascoltare la canzone "Bohemian Rhapsody", quindi digita la parola "Bohemian" nella barra di ricerca e batte invio.	
		Spotibase mostra le canzoni, gli artisti e gli album che nel titolo hanno la parola "Bohemian".
	Nik scorre i risultati di ricerca e nota la canzone che stava cercando.	



NOME SCENARIO	SC4_RegistrazioneUtente	
ATTORI	Bob:utente	
FLUSSO DEGLI EVENTI	ATTORE	SISTEMA
	Bob essendo appassionato di musica naviga su internet per poter trovare una piattaforma musicale in grado di poterlo soddisfare, e tra le varie inserzioni e annunci trova Spotibase. Iniziando a navigare su Spotibase, incomincia a riprodurre le canzoni che più gli piacciono e quindi decide di esprimere la preferenza.	
		Spotibase mostra un messaggio di alert che indica a Bob di registrarsi.
	Bob allora clicca sull'opzione di Registrazione.	
		Spotibase mostra un form per permettere a Bob di registrarsi inserendo Username, Email, Password e la riconferma della Password.
	Bob inserisce i dati richiesti .	
		Spotibase controlla i dati di Bob e lo rimanda sulla pagina home del sito.
	Bob continua a navigare sul sito da utente.	



NOME SCENARIO	SC5_AcquistoAbbonamenti	
ATTORI	Luca:utente	
FLUSSO DEGLI EVENTI	ATTORE	SISTEMA
	Luca si trova nella pagina Libreria e decide di creare una nuova playlist.	
		Spotibase con un alert avvisa Luca che ha terminato il numero di playlist creabili.
	Luca per poter creare un numero maggiore di playlist, va nella pagina Abbonamenti per acquistarne uno tra quelli disponibili	
		Spotibase mostra la pagina con i relativi Abbonamenti.
	Luca decide quindi di acquistare un abbonamento tra quelli disponibili.	
		Spotibase avvisa Luca con un alert che l'acquisto è andato a buon fine.



NOME SCENARIO	SC6_CreazionePlaylist	
ATTORI	Alice:utente	
FLUSSO DEGLI EVENTI	ATTORE	SISTEMA
	Alice vuole creare una playlist per salvare le canzoni che ama ascoltare. Quindi dopo aver effettuato il login, si sposta nella pagina “Libreria”.	
		Spotibase mostra la pagina “Libreria”, con le preferenze, gli acquisti, gli abbonamenti acquistati e le playlist.
	Alice scorre la pagina “Libreria” fino ad arrivare alla sezione “playlist” e clicca sul pulsante per creare una playlist.	
		Spotibase mostra un form con titolo e note della playlist da creare.
	Alice inserisce il nome: ”Canzoni che amo” con le note e clicca il pulsante “Invia”.	
		Spotibase ricarica la pagina “Libreria” con la nuova playlist.



NOME SCENARIO	SC7_DownloadCanzone	
ATTORI	Michael: utente	
FLUSSO DEGLI EVENTI	ATTORE	SISTEMA
	Michael è uno sportivo che ama ascoltare la musica durante i suoi allenamenti. La palestra che frequenta non dispone di sistemi per collegarsi alla rete. Michael, essendo una persona intelligente, il giorno prima decide di scaricare le canzoni che vorrà ascoltare in palestra. Michael si sposta nella libreria per scaricare le canzoni che ha acquistato.	
		Spotibase mostra la pagina “Libreria”, con le preferenze, gli acquisti, gli abbonamenti acquistati e le playlist.
	Michael, scorre la libreria e clicca sull'icona di download della relativa canzone.	
		Spotibase inizia il download restituendo il relativo file in formato mp3.



NOME SCENARIO	SC8_AcquistoCarrello	
ATTORI	Anthony: utente	
FLUSSO DEGLI EVENTI	ATTORE	SISTEMA
	Anthony, essendo un appassionato di musica, decide di acquistare l'intero album del suo artista preferito. Anthony accede alla home del sito e clicca sull'album da acquistare.	
		Spotibase mostra la pagina dell'album.
	Anthony clicca sul bottone aggiungi al carrello.	
		Spotibase aggiorna il badge dell'utente di un'unità, notificando l'aggiunta al carrello.
	Anthony si reca nella pagina del carrello per visualizzare l'album scelto.	
		Spotibase mostra la pagina del carrello, con l'album scelto da Anthony e il relativo prezzo.
	Anthony cliccando sul pulsante di acquisto realizza l'ordine.	
		Spotibase, reindirizza Anthony alla propria libreria.



3.4.2 Use Case model

Identificativo	UC_1	Data	25/10/2021
		Versione	0.1
Nome	Scaricare Canzone	Autore	Gagliarde Nicolapio
Descrizione	Lo UC fornisce la funzionalità di scaricare il file mp3 di una canzone		
Attore Principale	Utente E' interessato a scaricare il file mp3 di una canzone precedentemente acquistata.		
Attori secondari	NA		
Entry Condition	Questo UC inizia quando l'utente ha effettuato il login ed ha già acquistato la canzone		
Exit condition On success	Questo UC termina quando la canzone in formato mp3 viene scaricato sul dispositivo dell'utente.		
Exit condition On failure	Questo UC termina quando non viene scaricata nessuna canzone in formato mp3.		
Rilevanza/User Priority	Alta		
Frequenza stimata	NA		
Extension point	NA		
Generalization of	NA		
FLUSSO DI EVENTI PRINCIPALE/MAIN SCENARIO			
1	Utente:	Clicca sull'apposito pulsante di download.	
2	Sistema:	Avvia il download della canzone in formato mp3.	
3	Sistema:	Mostra un alert che notifica l'utente che è stata scaricata la canzone.	
Scenario/Flusso di eventi di ERRORE: sistema non riesce ad avviare il download del file mp3			
2.1	Sistema:	Visualizza un alert di errore all'utente. Il messaggio segnala che non è stato possibile scaricare il file mp3.	



Laurea Triennale in informatica-Università di Salerno
Corso di
Ingegneria del software- Prof. Carmine Gravino

Identificativo	UC_2	Data	25/10/2021
		Versione	0.1
Nome	Aggiungere la Canzone al Carrello	Autore	Umberto Della Monica
Descrizione	Lo UC fornisce la funzionalità di aggiungere una canzone al carrello		
Attore Principale	Utente E' interessato ad aggiungere una canzone al carrello		
Attori secondari	NA		
Entry Condition	NA		
Exit condition On success	Questo UC termina quando la canzone è stata aggiunta con successo al carrello		
Exit condition On failure	La canzone NON è stata aggiunta correttamente al carrello		
Rilevanza/User Priority	Alta		
Frequenza stimata	NA		
Extension point	NA		
Generalization of	NA		
FLUSSO DI EVENTI PRINCIPALE/MAIN SCENARIO			
1	Utente:	Preme il pulsante per aggiungere la canzone al carrello.	
2	Sistema:	Verifica che la canzone non sia già inserita nel carrello.	
3	Sistema:	Salva la canzone all'interno del carrello.	
4	Sistema:	Mostra un alert che notifica il corretto inserimento della canzone nel carrello.	
Scenario/Flusso di eventi Alternativo: Canzone già inserita			
2.1	Sistema:	Visualizza un alert che segnala all'utente che la canzone già è stata inserita	
Scenario/Flusso di eventi di ERRORE: Sistema non riesce ad aggiungere la canzone al carrello			
3.1	Sistema:	Visualizza un alert che segnala all'utente che la canzone non è inserita nel carrello	



Laurea Triennale in informatica-Università di Salerno
Corso di
Ingegneria del software- Prof. Carmine Gravino

Identificativo	UC_3	Data	25/10/2021
		Versione	0.1
Nome	Acquistare carrello	Autore	Umberto Della Monica
Descrizione	Lo UC fornisce la funzionalità di finalizzare l'acquisto del carrello		
Attore Principale	Utente E' interessato ad acquistare il contenuto del carrello.		
Attori secondari	NA		
Entry Condition	Questo UC inizia quando l'utente ha effettuato il login, il carrello contiene almeno una canzone e l'utente si trova nella pagina carrello.		
Exit condition On success	Questo UC termina quando l'utente ha acquistato con successo le canzoni.		
Exit condition On failure	Questo UC termina quando l'utente non ha potuto acquistare nessuna canzone.		
Rilevanza/User Priority	Alta		
Frequenza stimata	NA		
Extension point	NA		
Generalization of	NA		
FLUSSO DI EVENTI PRINCIPALE/MAIN SCENARIO			
1	Utente:	Clicca il pulsante per l'acquisto.	
2	Sistema:	Aggiorna gli acquisti dell'utente	
3	Sistema:	Svuota il carrello	
4	Sistema:	Mostra un alert che notifica l'acquisto del carrello.	
Scenario/Flusso di eventi di ERRORE: Sistema non riesce a registrare l'acquisto			
2.1	Sistema:	mostra un alert che notifica l'errore	



Laurea Triennale in informatica-Università di Salerno
Corso di
Ingegneria del software- Prof. Carmine Gravino

Identificativo	UC_4	Data	26/10/2021
		Versione	0.1
Nome	Esprimere preferenza Canzone	Autore	Gagliarde Nicolapio
Descrizione	Lo UC fornisce la funzionalità di esprimere la preferenza per una canzone		
Attore Principale	Utente E' interessato a esprimere una preferenza per una canzone		
Attori secondari	NA		
Entry Condition	Questo UC inizia quando l'utente ha effettuato il login e vuole esprimere una preferenza		
Exit condition On success	Questo UC termina quando il sistema registra la preferenza per la canzone		
Exit condition On failure	Questo UC termina quando il sistema non salva nessuna nuova preferenza		
Rilevanza/User Priority	Alta		
Frequenza stimata	NA		
Extension point	NA		
Generalization of	NA		
FLUSSO DI EVENTI PRINCIPALE/MAIN SCENARIO			
1	Utente:	Clicca l'apposito pulsante per esprimere la preferenza	
2	Sistema:	Salva la preferenza	
3	Sistema:	Notifica il corretto inserimento della preferenza per la canzone.	
Scenario/Flusso di eventi di ERRORE: sistema non riesce a salvare la preferenza			
2.1	Sistema:	Non aggiorna nessuna preferenza	
2.2	Sistema:	Mostra un alert che notifica l'errore	



Laurea Triennale in informatica-Università di Salerno
Corso di
Ingegneria del software- Prof. Carmine Gravino

Identificativo	UC_5	Data	26/10/2021
		Versione	0.1
Nome	Registrare account	Autore	Venturino Silvio
Descrizione	Lo UC fornisce la funzionalità di registrazione di un nuovo utente		
Attore Principale	Utente E' interessato a registrarsi al sistema per poter utilizzare le funzionalità riservate agli utenti registrati.		
Attori secondari	NA		
Entry Condition	NA		
Exit condition On success	Questo UC termina quando l'account è registrato ed attivo. L'utente verrà reindirizzato alla home come utente		
Exit condition On failure	Questo UC termina quando nessun nuovo account è registrato.		
Rilevanza/User Priority	Alta		
Frequenza stimata	NA		
Extension point	NA		
Generalization of	NA		
FLUSSO DI EVENTI PRINCIPALE/MAIN SCENARIO			
1	Utente:	Richiede di potersi registrare presso il sito cliccando il pulsante di registrazione.	
2	Sistema:	Visualizza una form che richiede l'inserimento di: <ul style="list-style-type: none">• Username: Stringa di caratteri alfanumerici, senza spazi, con lunghezza tra 1 e 40• Email: Stringa formato email• Password: Stringa di almeno 8 caratteri, di cui almeno una cifra e almeno una lettera maiuscola e una lettera minuscola. Con lunghezza massima pari a 20.• Conferma password: Stringa formato analogo alla password. Tutti i suddetti campi sono obbligatori.	
3	Utente:	Riempie tutti i campi obbligatori e sottomette il form compilato.	
4	Sistema:	Verifica che: <ul style="list-style-type: none">• tutti i campi obbligatori siano stati compilati.• L'username è senza spazi e con lunghezza tra 1 e 40.• l'username non sia già in uso• l'email non sia già presente nel sistema.• l' email sia in formato corretto• le password inserite siano uguali.• la password sia lunga almeno 8 caratteri e che contenga almeno una cifra ed una lettera.	
5	Sistema:	Salva i dati dell'utente.	
6	Sistema:	Reindirizza alla schermata home mostrando un alert che notifica il successo della creazione dell' account.	



Laurea Triennale in informatica-Università di Salerno
Corso di
Ingegneria del software- Prof. Carmine Gravino

Scenario/Flusso di eventi Alternativo: qualche campo obbligatorio non è stato compilato		
4.a1	Sistema:	Visualizza un alert di errore che segnala all'utente che non ha inserito tutti i dati obbligatori.
4.a2	Sistema	Resta in attesa di una nuova sottomissione della form.
Scenario/Flusso di eventi Alternativo: "password" e "conferma password" hanno contenuti diversi		
4b.1	Sistema:	Visualizza un alert di errore che segnala all'utente che i campi password e conferma password non corrispondono.
4b.2	Sistema	Resta in attesa di una nuova sottomissione della form.
Scenario/Flusso di eventi Alternativo: la username inserita dall'utente è già in uso		
4c.1	Sistema:	Visualizza un alert di errore all'utente che gli segnala che l'username scelto già è presente all'interno del sistema e pertanto deve cambiarlo.
4c.2	Sistema	Resta in attesa di una nuova sottomissione della form.
Scenario/Flusso di eventi Alternativo: la email inserita dall'utente è già in uso		
4d.1	Sistema:	Visualizza un messaggio di errore all'utente che gli segnala che l'email scelta è già presente all'interno del sistema e pertanto deve cambiarla.
4d.2	Sistema	Resta in attesa di una nuova sottomissione della form.
Scenario/Flusso di eventi Alternativo: l'email inserita non è corretta		
4e.1	Sistema:	Visualizza un alert di errore all'utente che gli segnala che l'email scelta non è corretta e pertanto deve cambiarla.
4e.2	Sistema	Resta in attesa di una nuova sottomissione della form.
Scenario/Flusso di eventi di ERRORE: sistema non riesce ad effettuare il salvataggio dei dati		
5.1	Sistema:	non salva nessun nuovo utente
5.2	Sistema	mostra un alert di errore invitando l'utente a riprovare più tardi
Scenario/Flusso di eventi Alternativo: il formato della password non è corretto		
4f.1	Sistema:	Visualizza un alert di errore che segnala all'utente che il formato password non è corretto
4f.2	Sistema	Resta in attesa di una nuova sottomissione della form.



Laurea Triennale in informatica-Università di Salerno
Corso di
Ingegneria del software- Prof. Carmine Gravino

Identificativo		UC_6	Data	26/10/2021
			Versione	0.1
Nome		Accedere alla Piattaforma	Autore	Venturino Silvio
Descrizione		Lo UC fornisce la funzionalità di login di un utente		
Attore Principale		Utente L'utente è interessato ad accedere al sito		
Attori secondari		NA		
Entry Condition		Questo UC inizia quando l'utente è registrato sulla piattaforma e vuole effettuare il login		
Exit condition On success		Questo UC termina quando l'utente ha effettuato il login		
Exit condition On failure		Questo UC termina quando l'utente non riesce a completare il login		
Rilevanza/User Priority		Alta		
Frequenza stimata		NA		
Extension point		NA		
Generalization of		NA		
FLUSSO DI EVENTI PRINCIPALE/MAIN SCENARIO				
1	Utente:	Clicca il pulsante per accedere alla piattaforma.		
2	Sistema:	Visualizza un form che richiede l'inserimento di: <ul style="list-style-type: none">• Indirizzo e-mail: Stringa formato email• Password: Stringa Tutti i suddetti campi sono obbligatori.		
3	Utente:	Riempie tutti i campi obbligatori e sottomette il form compilato.		
4	Sistema:	Verifica che: <ul style="list-style-type: none">• tutti i campi obbligatori siano stati compilati.• l'utente sia presente nel sistema.		
5	Sistema:	Riconosce correttamente l'utente.		
6	Sistema:	Notifica con un alert il corretto accesso alla piattaforma.		
Scenario/Flusso di eventi Alternativo: qualche campo obbligatorio non è stato compilato				
4.a1	Sistema:	Visualizza un alert di errore che segnala all'utente che non ha inserito tutti i dati obbligatori.		
4.a2	Sistema	Resta in attesa di una nuova sottomissione della form.		
Scenario/Flusso di eventi Alternativo: "password" e/o "email" inesistenti				
4b.1	Sistema:	Visualizza un alert di errore che segnala all'utente che i campi password e/o email sono errati		
4b.2	Sistema	Resta in attesa di una nuova sottomissione della form.		



Laurea Triennale in informatica-Università di Salerno
Corso di
Ingegneria del software- Prof. Carmine Gravino

Identificativo	UC_7	Data	26/10/2021
		Versione	0.1
Nome	Riprodurre canzone	Autore	Venturino Silvio
Descrizione	Lo UC fornisce la funzionalità di riprodurre canzoni		
Attore Principale	Utente L'utente vuole riprodurre una canzone		
Attori secondari	NA		
Entry Condition	NA		
Exit condition On success	Questo UC termina quando il sistema riproduce la canzone scelta.		
Exit condition On failure	Questo UC termina quando il sistema non riproduce nessuna canzone.		
Rilevanza/User Priority	Alta		
Frequenza stimata	NA		
Extension point	NA		
Generalization of	NA		
FLUSSO DI EVENTI PRINCIPALE/MAIN SCENARIO			
1	Utente:	Clicca il pulsante di riproduzione per avviare la canzone.	
2	Sistema:	Avvia la riproduzione della canzone.	
3	Sistema:	Notifica con un alert di successo la riproduzione della canzone.	
Scenario/Flusso di eventi di ERRORE: sistema non riesce a riprodurre la canzone			
5.1	Sistema:	non riproduce nessuna canzone mostrando all'utente un alert di errore .	



Laurea Triennale in informatica-Università di Salerno
Corso di
Ingegneria del software- Prof. Carmine Gravino

Identificativo	UC_8	Data	26/10/2021
		Versione	0.1
Nome	Aggiungere canzone playlist	Autore	Alessandro Aquino
Descrizione	Lo UC fornisce la funzionalità di aggiungere una nuova canzone alla playlist		
Attore Principale	Utente E' interessato ad aggiungere una nuova canzone nella playlist.		
Attori secondari	NA		
Entry Condition	Questo UC inizia quando l'utente ha effettuato il login, la playlist è stata creata e la canzone è in riproduzione.		
Exit condition On success	Questo UC finisce quando l'utente ha aggiunto una nuova canzone alla playlist.		
Exit condition On failure	Questo UC termina quando il sistema non aggiunge nessuna canzone		
Rilevanza/User Priority	Alta		
Frequenza stimata	NA		
Extension point	NA		
Generalization of	NA		
FLUSSO DI EVENTI PRINCIPALE/MAIN SCENARIO			
1	Utente:	Clicca il pulsante di aggiunta canzone in playlist.	
2	Sistema:	Mostra a schermo un alert al cui interno ci sono le playlist dell'utente.	
3	Utente:	Clicca sul titolo della playlist nella quale vuole inserire la canzone.	
4	Sistema:	Verifica se la canzone è già inserita nella playlist.	
5	Sistema:	Salva la canzone nella playlist scelta dall'utente.	
6	Sistema:	Mostra un alert che notifica il successo dell'operazione.	
Scenario/Flusso di eventi Alternativo: canzone già aggiunta nella playlist			
4.a1	Sistema:	Mostra un alert notificando la presenza della canzone nella playlist.	
Scenario/Flusso di eventi di ERRORE: sistema non riesce ad effettuare il salvataggio della canzone nella playlist			
4b.1	Sistema:	Visualizza un alert di errore all'utente. Il messaggio segnala che non è stato possibile salvare i dati e invita a riprovare più tardi.	



Laurea Triennale in informatica-Università di Salerno
Corso di
Ingegneria del software- Prof. Carmine Gravino

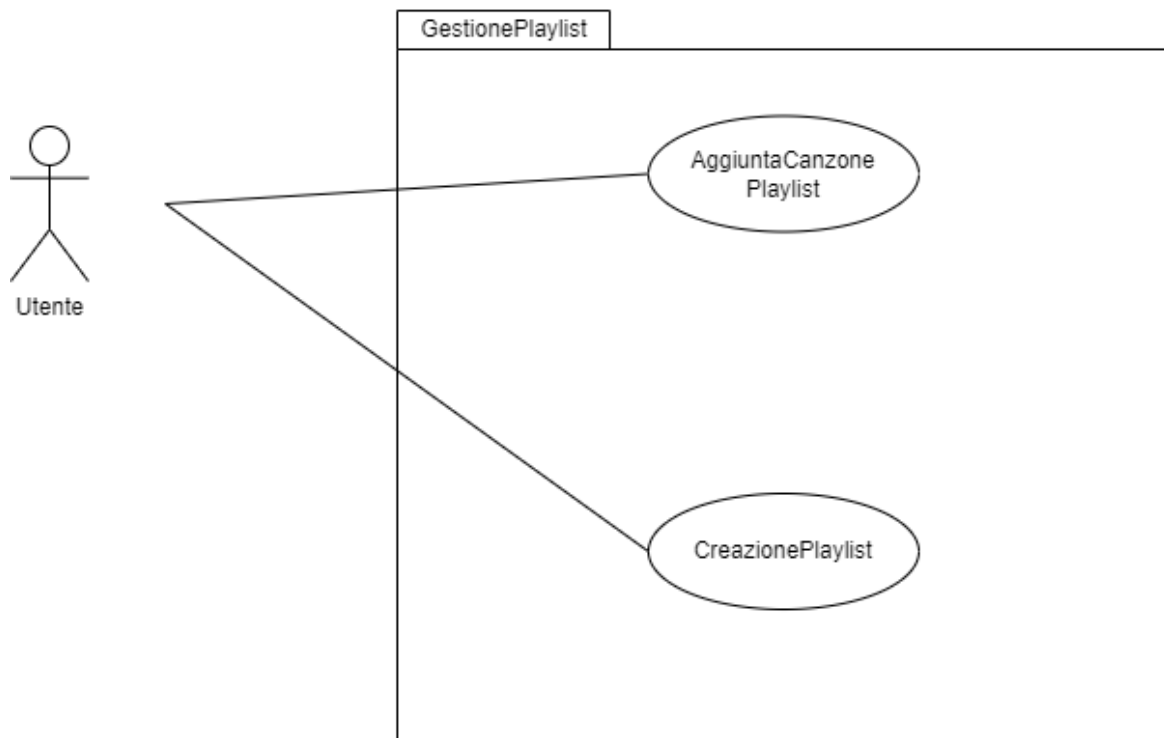
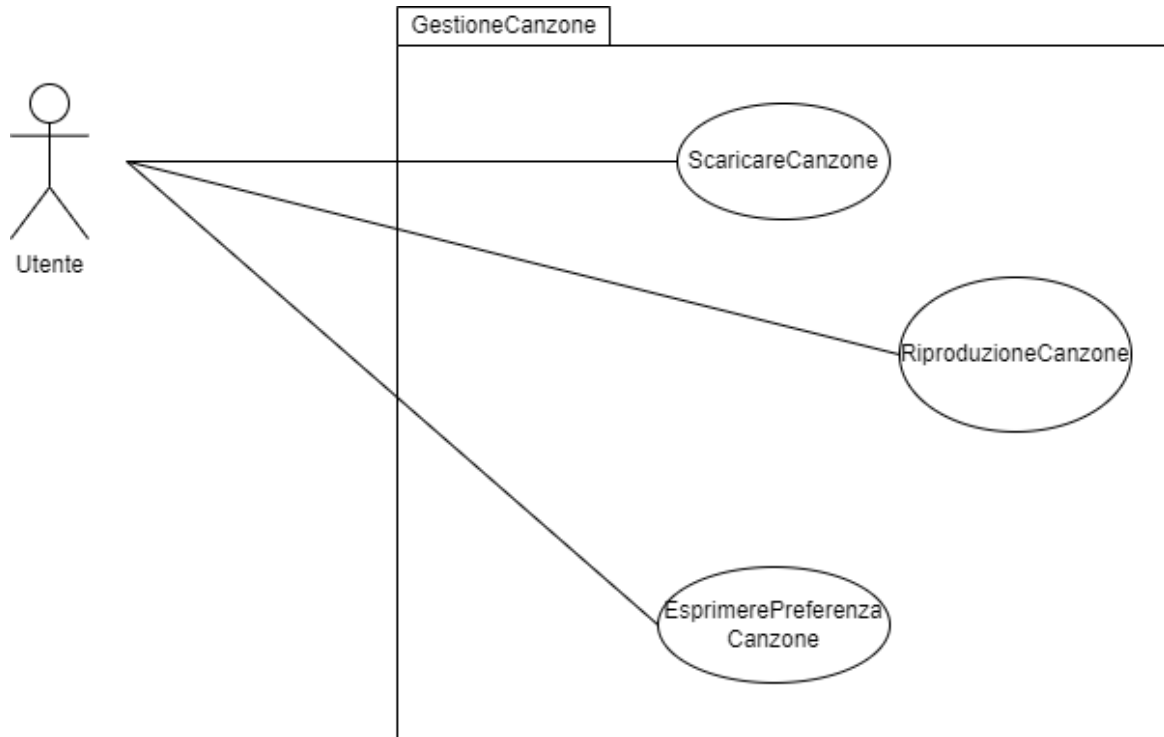
Identificativo	UC_9	Data	26/10/2021
		Versione	0.1
		Autore	Alessandro Aquino
Nome	Creare playlist		
Descrizione	Lo UC fornisce la funzionalità di creare una nuova playlist		
Attore Principale	Utente E' interessato a creare una nuova playlist.		
Attori secondari	NA		
Entry Condition	Questo UC inizia quando l'utente ha effettuato il login e si trova nella pagina libreria.		
Exit condition On success	Questo UC finisce quando il sistema ha creato una nuova playlist.		
Exit condition On failure	Questo UC termina quando il sistema non crea nessuna nuova playlist		
Rilevanza/User Priority	Alta		
Frequenza stimata	NA		
Extension point	NA		
Generalization of	NA		
FLUSSO DI EVENTI PRINCIPALE/MAIN SCENARIO			
1	Utente:	Clicca il pulsante per creare la playlist.	
2	Sistema:	mostra a schermo un form con due campi: <ul style="list-style-type: none">• Titolo: stringa di caratteri alfanumerici, con lunghezza compresa tra 1 e 100.• Note: stringa di caratteri alfanumerici, con lunghezza compresa tra 0 e 2000.	
3	Utente:	Compila i due campi e invia il form.	
4	Sistema:	Verifica che il titolo sia stato compilato.	
5	Sistema:	Verifica che il titolo appena inserito sia diverso dal titolo delle playlist già esistenti.	
6	Sistema:	Verifica se il limite di playlist creabili è stato raggiunto.	
7	Sistema:	Salva la playlist dell'utente.	
8	Sistema:	Ricarica la libreria con la nuova playlist creata notificando l'utente la sua corretta creazione.	
Scenario/Flusso di eventi Alternativo: il campo "Titolo" obbligatorio non è stato compilato			
4.a1	Sistema	Visualizza un alert di errore che segnala all'utente che non ha inserito il campo obbligatorio.	
4.a2	Sistema	Resta in attesa di una nuova sottomissione del form.	
Scenario/Flusso di eventi Alternativo: Raggiungimento del limite massimo di playlist creabili.			
6.a1	Sistema	Visualizza un alert che indica il raggiungimento del limite massimo delle playlist create.	

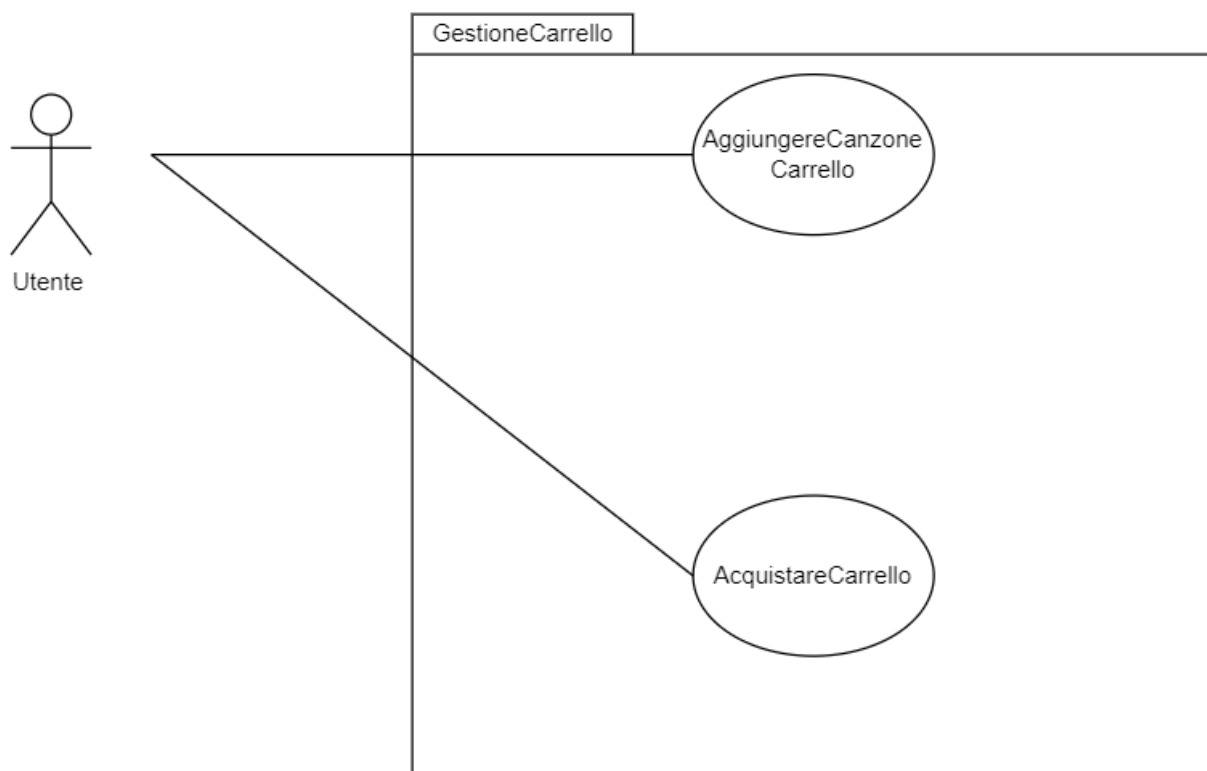
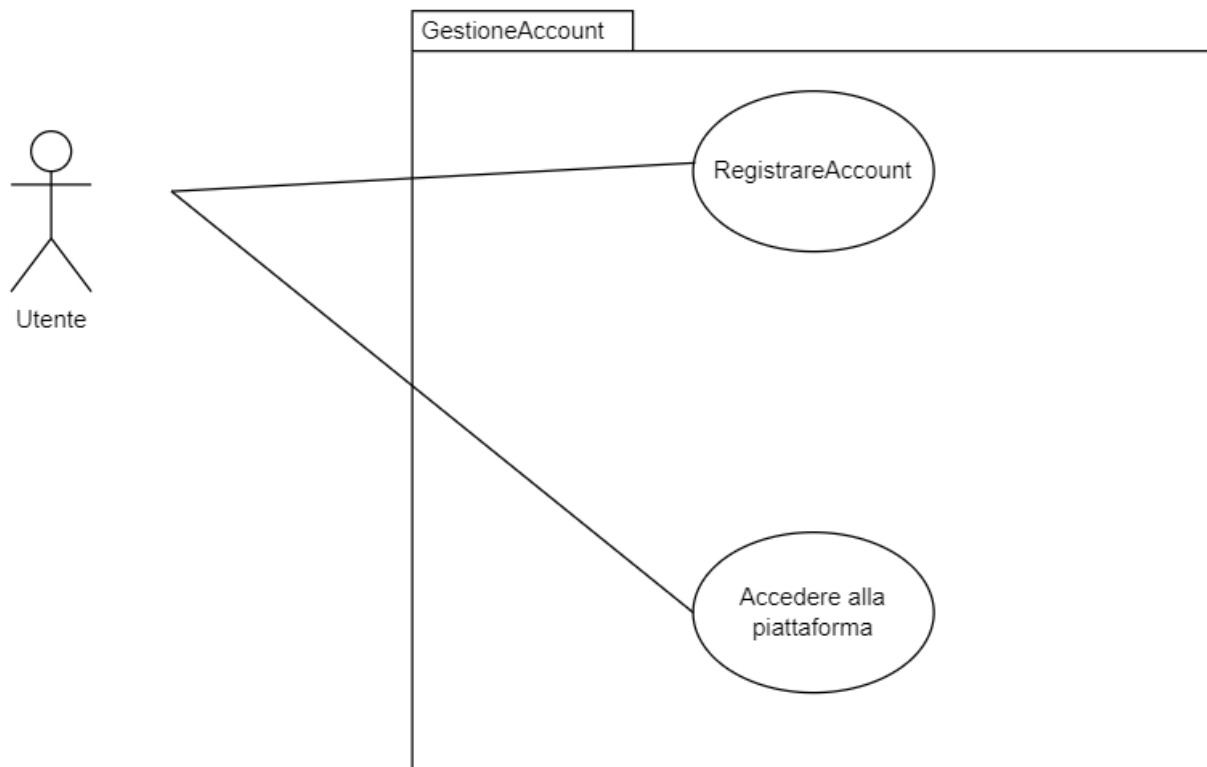


Laurea Triennale in informatica-Università di Salerno
Corso di
Ingegneria del software- Prof. Carmine Gravino

Scenario/Flusso di eventi Alternativo: Titolo della playlist già utilizzato.		
5.a1	Sistema	Visualizza un alert indicando che il titolo della playlist è già stato utilizzato.
5.a2	Sistema	Resta in attesa di una nuova sottomissione del form.
Scenario/Flusso di eventi di ERRORE: sistema non riesce ad effettuare il salvataggio della playlist		
7.1	Sistema	Visualizza un alert di errore all'utente. Il messaggio segnala che non è stato possibile salvare i dati e invita a riprovare più tardi.

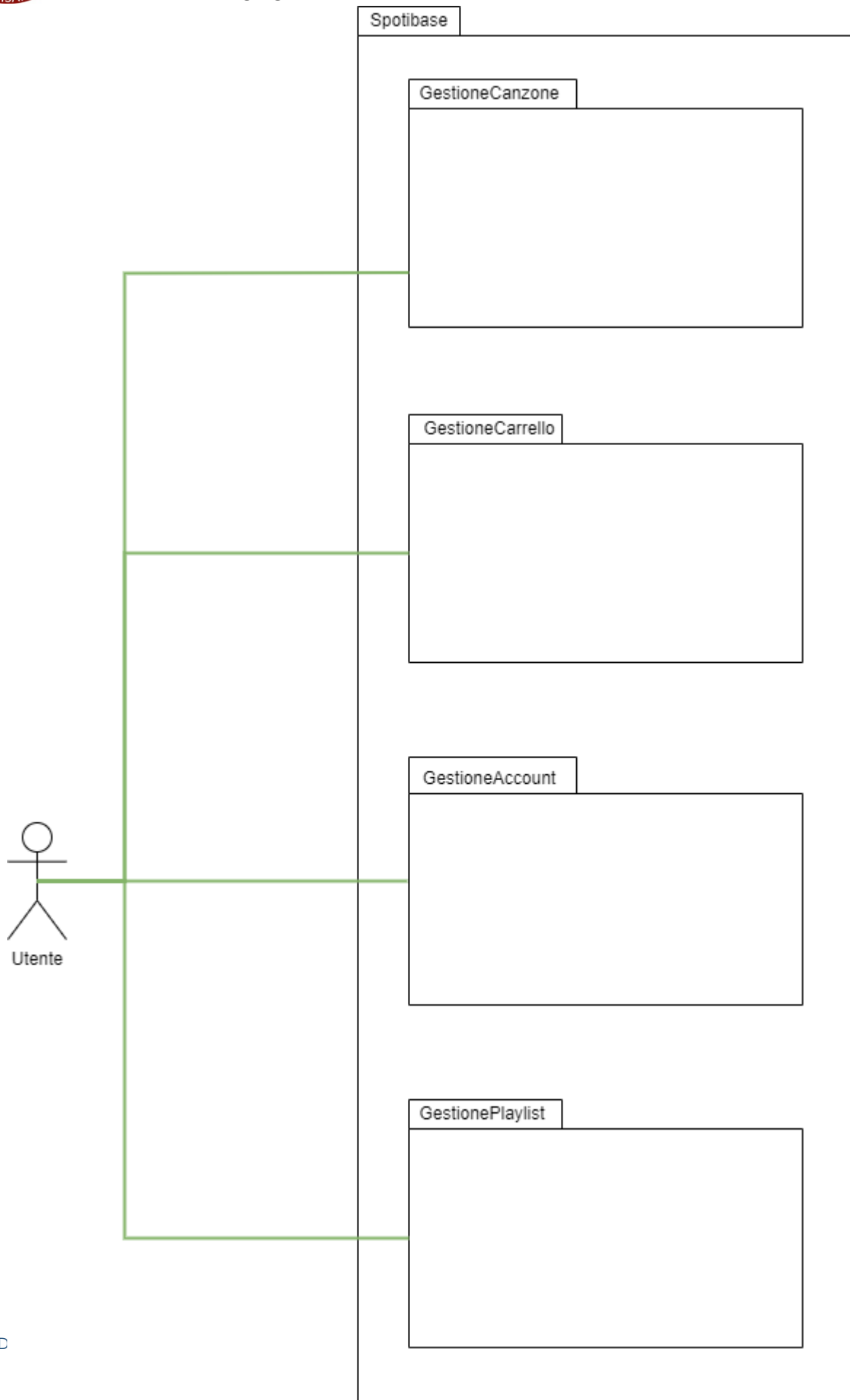
3.4.2.1 Use-case diagram







Laurea Triennale in informatica-Università di Salerno
Corso di
Ingegneria del software- Prof. Carmine Gravino





3.4.3 Object model

3.4.3.1 Tabella oggetti

NOME	TIPO	DESCRIZIONE
Canzone	Entity	Rappresenta un file audio formato mp3
Utente	Entity	L'utente che ha effettuato il login, e di conseguenza può esprimere preferenze, ascoltare canzoni, acquistare elementi per poi scaricarli. Inoltre può creare playlist, eliminarle e aggiungere canzoni.
Carrello	Entity	Un carrello per raccogliere canzoni che un utente vuole acquistare. Un carrello non può contenere più volte la stessa canzone. L'acquisto dell'intero carrello può essere fatto solo da utenti che hanno effettuato il login.
Acquisto	Entity	Rappresenta una canzone acquistata da un utente. L'utente non può acquistare più volte la stessa canzone.
Preferenza	Entity	L'utente può esprimere preferenze per delle canzoni.
Playlist	Entity	Raccoglie un insieme di canzoni scelte dall'utente. L'utente può inserire e rimuovere canzoni. L'utente può creare e cancellare playlist dal sistema. Nel momento della creazione l'utente deve indicare un nome e può scrivere delle note. L'utente non può creare due playlist con lo stesso nome.
DownloadButtonBoundary	Boundary	Bottone usato dall'utente per scaricare una canzone acquistata.



Laurea Triennale in informatica-Università di Salerno
Corso di
Ingegneria del software- Prof. Carmine Gravino

AlertDownloadAvvenutoConSuccessoBoundary	Boundary	Alert mostra all'utente il corretto download della canzone.
AlertErroreDownloadCanzoneBoundary	Boundary	Alert mostra all'utente l'errore di download di una canzone per un utente.
AddCanzoneCarrelloButtonBoundary	Boundary	Bottone usato dall'utente per aggiungere una canzone al carrello.
AlertCorrettoInserimentoCanzoneCarrelloBoundary	Boundary	Alert mostra all'utente il corretto inserimento della canzone nel carrello
AlertErroreCanzoneCarrelloBoundary	Boundary	Alert mostra all'utente che una canzone già è stata precedentemente inserita nel carrello. Quindi non inserisce la canzone nel carrello.
AlertErroreInserimentoCanzoneBoundary	Boundary	Alert mostra all'utente l'errore di inserimento della canzone nel carrello.
AcquistoCarrelloButtonBoundary	Boundary	Bottone usato dall'utente per finalizzare l'acquisto dell'intero carrello.
AlertSuccessoAcquistoBoundary	Boundary	Alert mostra all'utente il corretto acquisto del carrello.
AlertErroreAcquistoCarrelloBoundary	Boundary	Alert mostra all'utente che si è verificato un errore nella fase di acquisto.
AddPreferenzaButtonBoundary	Boundary	Bottone usato dall'utente per esprimere una preferenza ad una canzone
AlertPreferenzaInseritaCorrettamenteBoundary	Boundary	Alert mostra all'utente che la preferenza è stata inserita correttamente.
AlertErroreInserimentoPreferenzaBoundary	Boundary	Alert mostra all'utente che la preferenza non è stata inserita.
RegistrazioneButtonBoundary	Boundary	Bottone usato dall'utente per visualizzare la form di registrazione.
RegistrazioneFormBoundary	Boundary	Form mostrato all'utente per l'input dei propri dati. Questa form è mostrata quando l'utente usa il bottone



Laurea Triennale in informatica-Università di Salerno
Corso di
Ingegneria del software- Prof. Carmine Gravino

		RegistrazioneButtonBoundary. La form contiene i campi: email, username, password, conferma password e un bottone per inviare i dati.
AlertCorrettaCreazioneAccountBoundary	Boundary	Alert mostra all'utente che la registrazione è avvenuta correttamente.
AlertErroreCampoObbligatorioBoundary	Boundary	Alert mostra all'utente che qualche campo obbligatorio non è stato inserito.
AlertErroreInserimentoPasswordBoundary	Boundary	Alert mostra all'utente che la password non è stata inserita correttamente.
AlertErroreConfermaPasswordBoundary	Boundary	Alert mostra all'utente che il conferma password è differente dalla password
AlertErroreUsernameBoundary	Boundary	Alert mostra all'utente che l'username è già in uso.
AlertErroreEmailInUsoBoundary	Boundary	Alert mostra all'utente che l'email è già in uso.
AlertErroreEmailBoundary	Boundary	Alert mostra all'utente che l'email non è corretta.
AlertErroreSalvataggioBoundary	Boundary	Alert mostra all'utente che il sistema non riesce ad effettuare il salvataggio dei dati di registrazione.
LoginButtonBoundary	Boundary	Bottone usato dall'utente per visualizzare la form di login.
LoginFormBoundary	Boundary	Form mostrato all'utente per l'input dei propri dati. Questa form è mostrata quando l'utente usa il bottone LoginButtonBoundary. La form contiene: email, password e un bottone per inviare i dati.
AlertCorrettoLoginBoundary	Boundary	Alert mostra all'utente il successo di login.
AlertErroreLoginCampiBoundary	Boundary	Alert mostra all'utente che i campi



Laurea Triennale in informatica-Università di Salerno
Corso di
Ingegneria del software- Prof. Carmine Gravino

ndary		obbligatorie non sono stati inseriti.
AlertErroreLoginPassword UsernameBoundary	Boundary	Alert mostra all'utente che i campi login e password non sono corretti.
RiproduzioneCanzoneButto nBoundary	Boundary	Bottone usato dall'utente per avviare la riproduzione del file mp3 associato alla canzone.
AlertSuccessoRiproduzione CanzoneBoundary	Boundary	Alert mostra all'utente la corretta riproduzione della canzone.
AlertErroreRiproduzioneCa nzoneBoundary	Boundary	Alert mostra all'utente l'errore nella riproduzione della canzone.
InserisciCanzoneInPlaylistB uttonBoundary	Boundary	Bottone usato dall'utente per inserire una canzone in qualche playlist. Il sistema mostra un alert con la lista delle playlist dell'utente.
AlertVisualizzaPlaylistBound ary	Boundary	L'alert mostra all'utente le proprie playlist.
AlertCanzoneGiàPresenteIn PlaylistBoundary	Boundary	L'alter specifica all'utente che la canzone che si vuole inserire nella playlist è già presente.
AlertErroreSalvataggioCanz onePlaylistBoundary	Boundary	L'alert avvisa l'utente che il salvataggio della canzone nella playlist non è avvenuto.
AlertCanzoneSalvataggioRiu scitoPlaylistBoundary	Boundary	L'alert notifica il corretto inserimento della canzone nella playlist.
ButtonCreaPlaylistBoundary	Boundary	Bottone usato dall'utente per creare una nuova playlist. Il sistema mostrerà la form: "FormPlaylistBoundary".
FormPlaylistBoundary	Boundary	Form mostrato all'utente per creare una nuova playlist. Il form richiede il titolo (obbligatorio), e le note opzionali. L'utente compila il form e invia i dati con l'apposito bottone.



Laurea Triennale in informatica-Università di Salerno
Corso di
Ingegneria del software- Prof. Carmine Gravino

AlertCorrettaCreazionePlaylistBoundary	Boundary	Alert mostra all'utente la corretta creazione della playlist.
AlertCampoNonCompilatoPlaylistBoundary	Boundary	L>alert avvisa l'utente che il campo obbligatorio "titolo" non è stato compilato.
AlertNumeroMassimoPlaylistRaggiuntoBoundary	Boundary	L>alert avvisa l'utente che è stato raggiunto il numero massimo di playlist creabili.
AlertTitoloGiàInUsoPlaylistBoundary	Boundary	L>alert avvisa l'utente che è già presente una playlist con lo stesso titolo.
AlertSalvataggioCreazionePlaylistNonRiuscitoBoundary	Boundary	L>alert avvisa l'utente che il salvataggio della playlist nel sistema non è avvenuto.
RiproduzioneCanzoneControl	Control	Gestisce la funzione di riproduzione del file mp3. Questo oggetto è creato quando l'utente usa il bottone "RiproduzioneCanzoneButtonBoundary", avvia la riproduzione del file mp3.
DownloadCanzoneControl	Control	Gestisce la funzione di download mp3 associato alla canzone. Questo oggetto è creato quando l'utente usa il bottone "DownloadButtonBoundary", avvia il download del file mp3 e termina.
SalvataggioCanzoneCarrelloControl	Control	Gestisce la funzione di aggiunta canzone, questo oggetto viene creato quando l'utente usa il bottone "AddCanzoneCarrelloButtonBoundary". Questo oggetto salva l'informazione nell'entity "Carrello" e aggiorna il boundary "BadgeCarrelloBoundary" riportando il numero di elementi dell'entity "Carrello"
AcquistoCarrelloControl	Control	Gestisce la funzione di acquisto del carrello. Questo oggetto viene creato



Laurea Triennale in informatica-Università di Salerno
Corso di
Ingegneria del software- Prof. Carmine Gravino

		<p>quando l'utente usa il bottone "AcquistoCarrelloButtonBoundary".</p> <p>L'oggetto interagisce con le entity: "Acquisto", per salvare i dati di acquisto; "Carrello", per eliminare gli elementi acquistati.</p>
SalvaPreferenzaControl	Control	<p>Gestisce la funzione di preferenza per una canzone. Questo oggetto viene creato dall'utente quando usa il bottone "AddPreferenzaButtonBoundary".</p> <p>L'oggetto interagisce con la entity: "Preferenza" per salvare la preferenza per ogni utente.</p>
RegistrazioneAccountControl	Control	<p>Gestisce la funzione di registrazione di un nuovo account. Questo oggetto viene creato quando l'utente usa il bottone "RegistrazioneButtonBoundary", l'oggetto interagisce con il boundary "RegistrazioneFormBoundary". L'utente compila il form e lo sottomette, l'oggetto controlla i campi e interagisce con l'entity "Utente" per salvare il nuovo account.</p>
LoginAccountControl	Control	<p>Gestisce la funzione di login di un utente. Questo oggetto viene creato quando un utente usa il bottone "LoginButtonBoundary" l'oggetto interagisce con il boundary "LoginFormBoundary". L'utente compila il form. Il sistema verifica i campi interagendo con l'entity "Utente".</p>
AggiungereCanzonePlaylistControl	Control	<p>Gestisce l'aggiunta di una canzone in una playlist dell'utente. Questo oggetto viene creato quando l'utente usa il bottone "InserisciCanzoneInPlaylistButtonBoundary". Successivamente questo oggetto interagisce con "AlertVisualizzaPlaylistBoundary" per mostrare le playlist, l'utente ne sceglie</p>

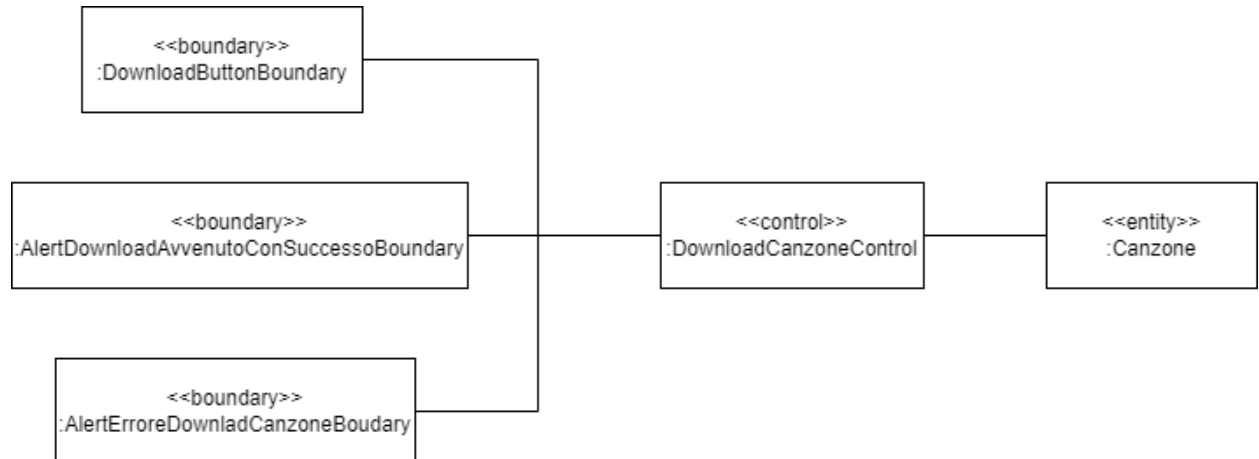


Laurea Triennale in informatica-Università di Salerno
Corso di
Ingegneria del software- Prof. Carmine Gravino

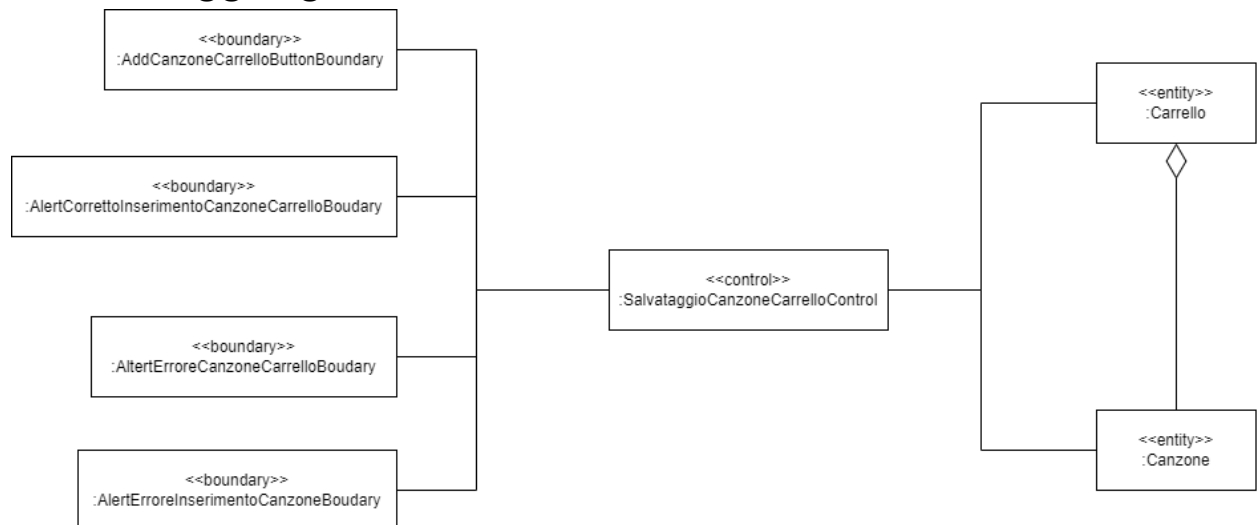
		una e l'oggetto interagisce con l'entity "Playlist" per salvare la nuova canzone aggiunta.
CreazionePlaylistControl	Control	Gestisce la creazione di una nuova playlist di un utente. Questo oggetto viene creato quando l'utente usa il bottone "ButtonCreaPlaylistBoundary". L'oggetto interagisce con il boundary "FormPlaylistBoundary". L'utente compila almeno i campi obbligatori e invia il form. Se è stato raggiunto il limite massimo di playlist creabili mostra l'oggetto boundary "AlertNumeroMassimoPlaylistRaggiunto boundary". In caso contrario viene aggiornato l'oggetto entity "Playlist".

3.4.3.2 Object diagram

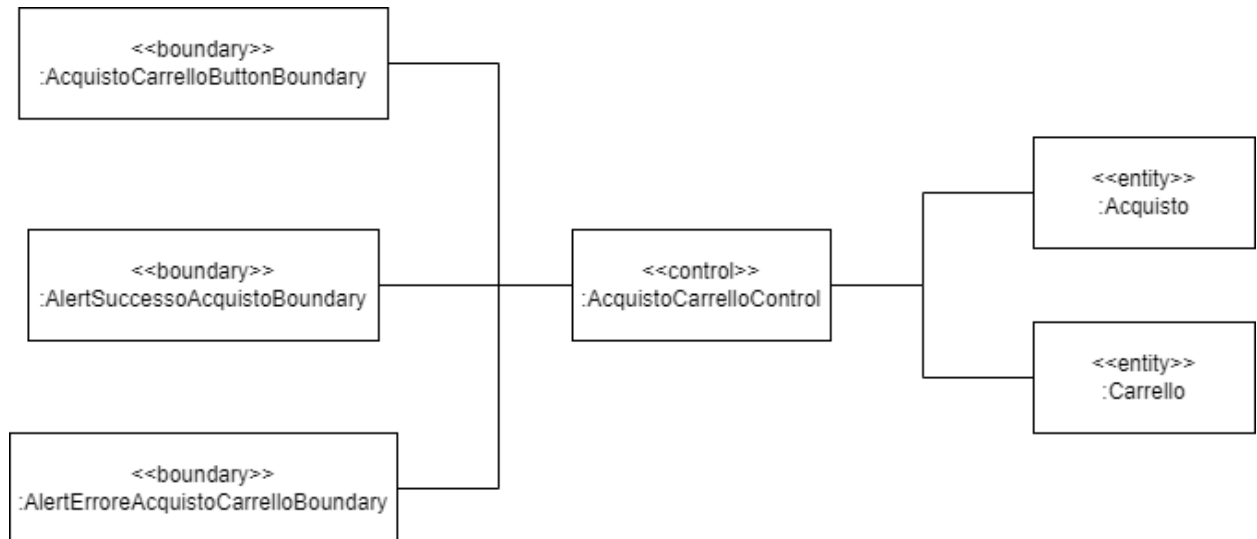
OD_UC1 Scaricare Canzone



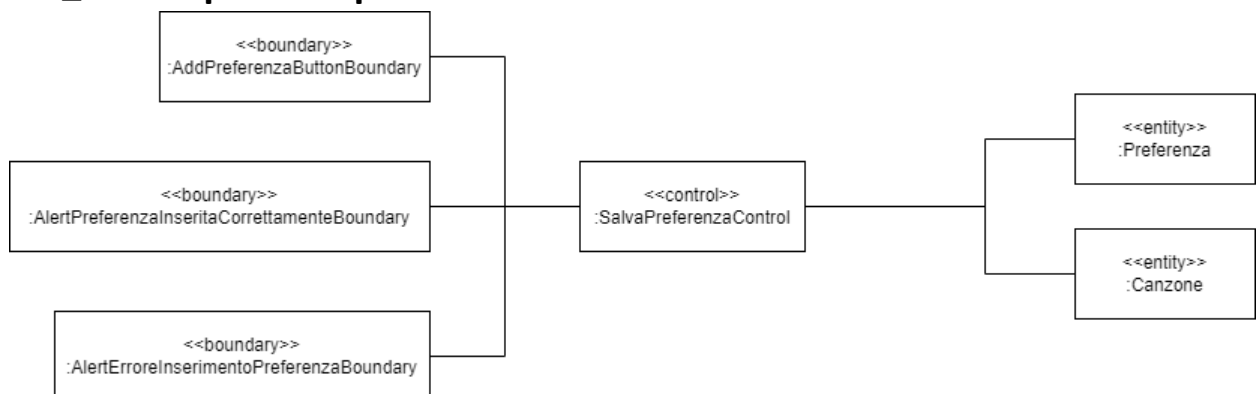
OD_UC2 Aggiungere Canzone al Carrello



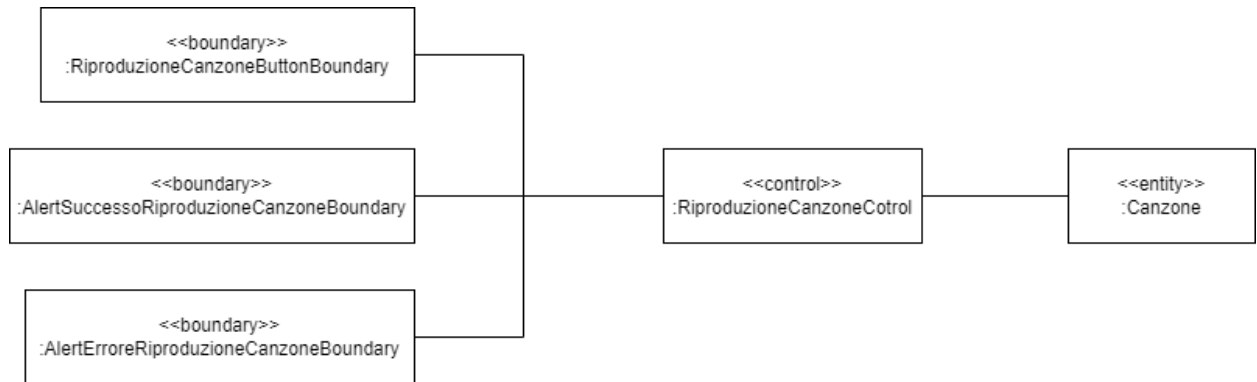
OD_UC3 Acquistare Carrello



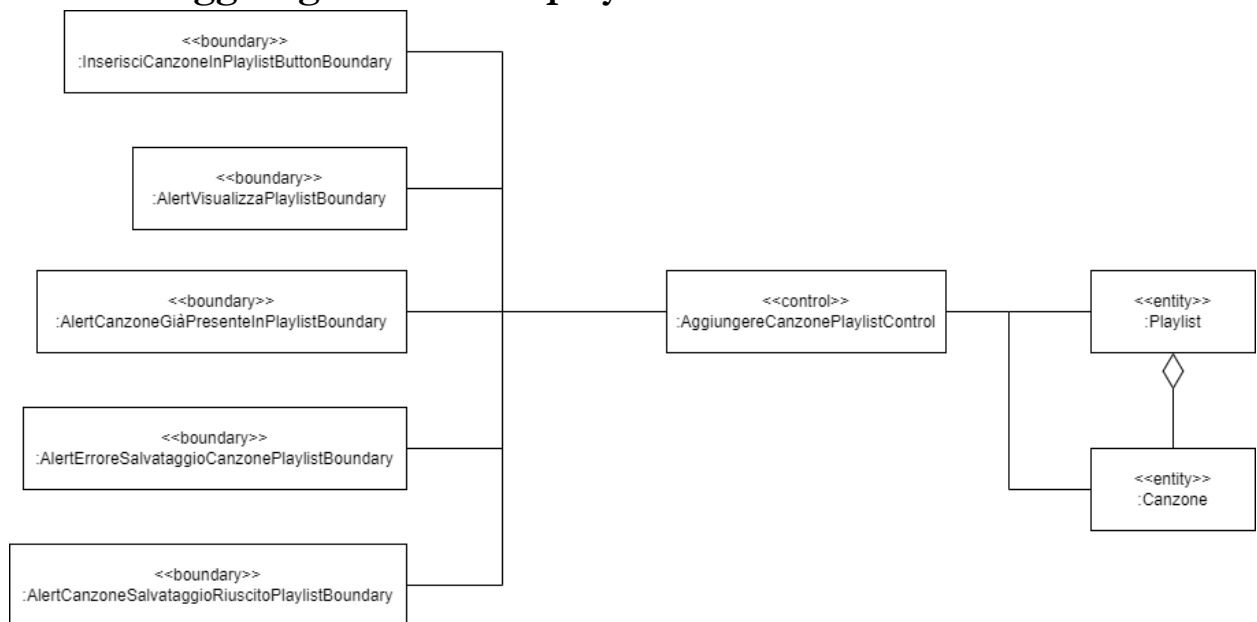
OD_UC4 Esprimere preferenza Canzone



OD_UC7 Riprodurre canzone

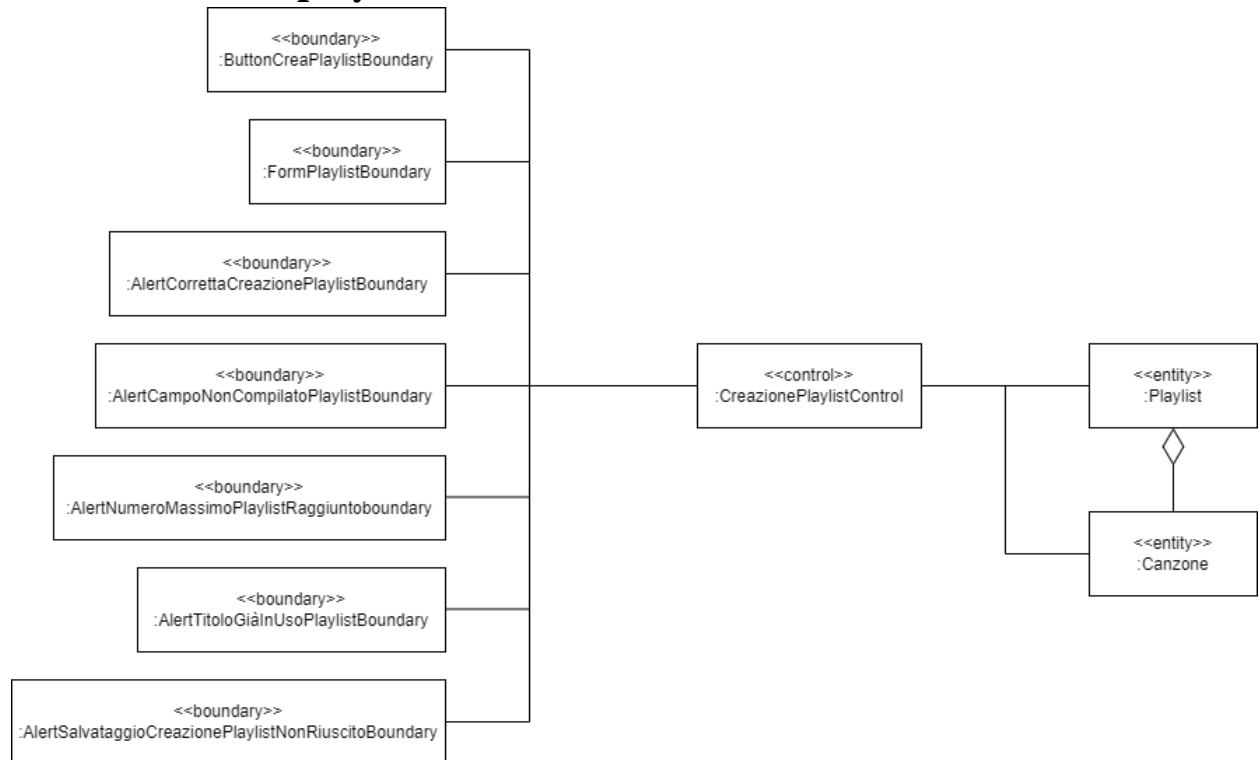


OD_UC8 Aggiungere canzone playlist





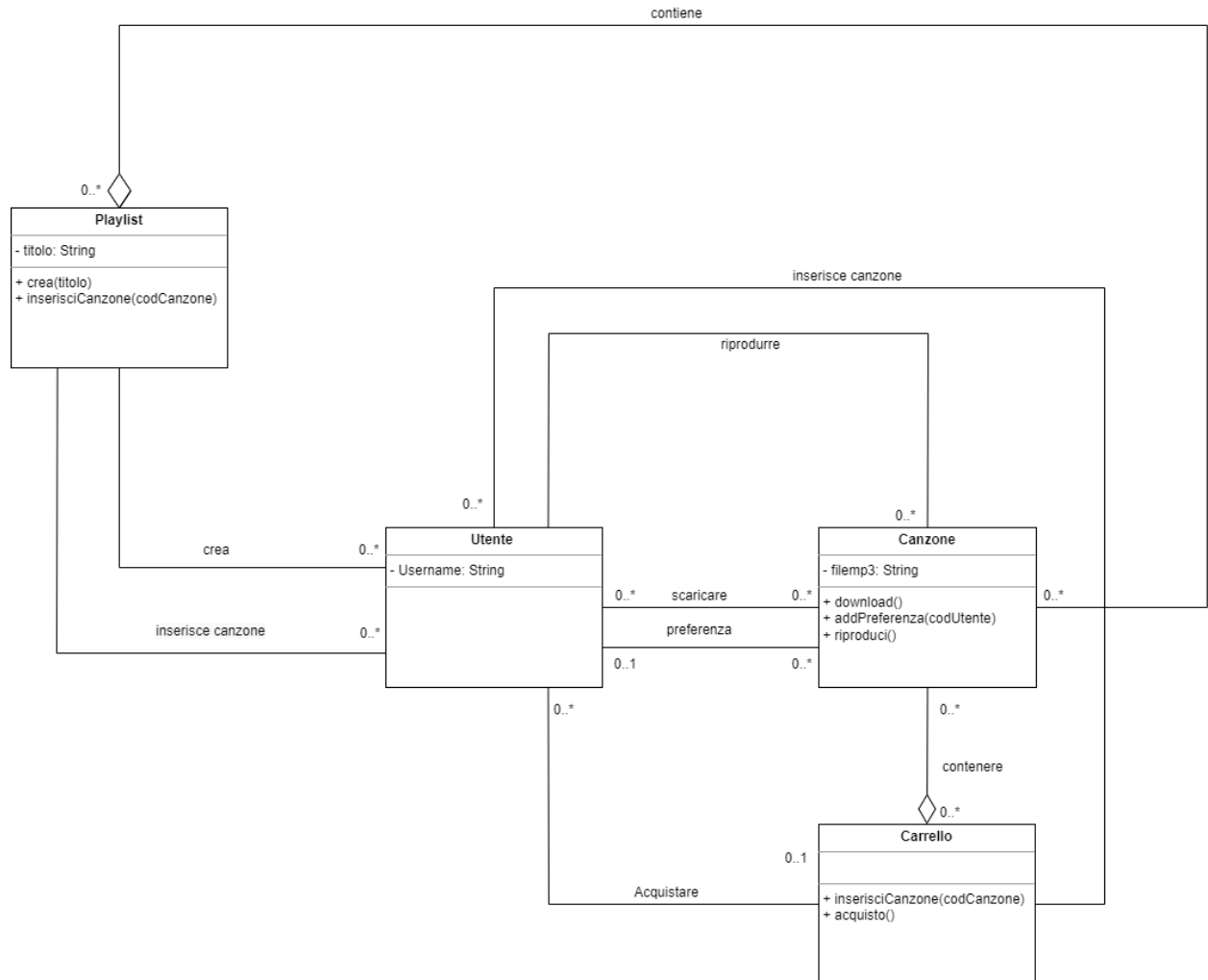
OD_UC9 Creare playlist





3.4.3.3 Class Diagram

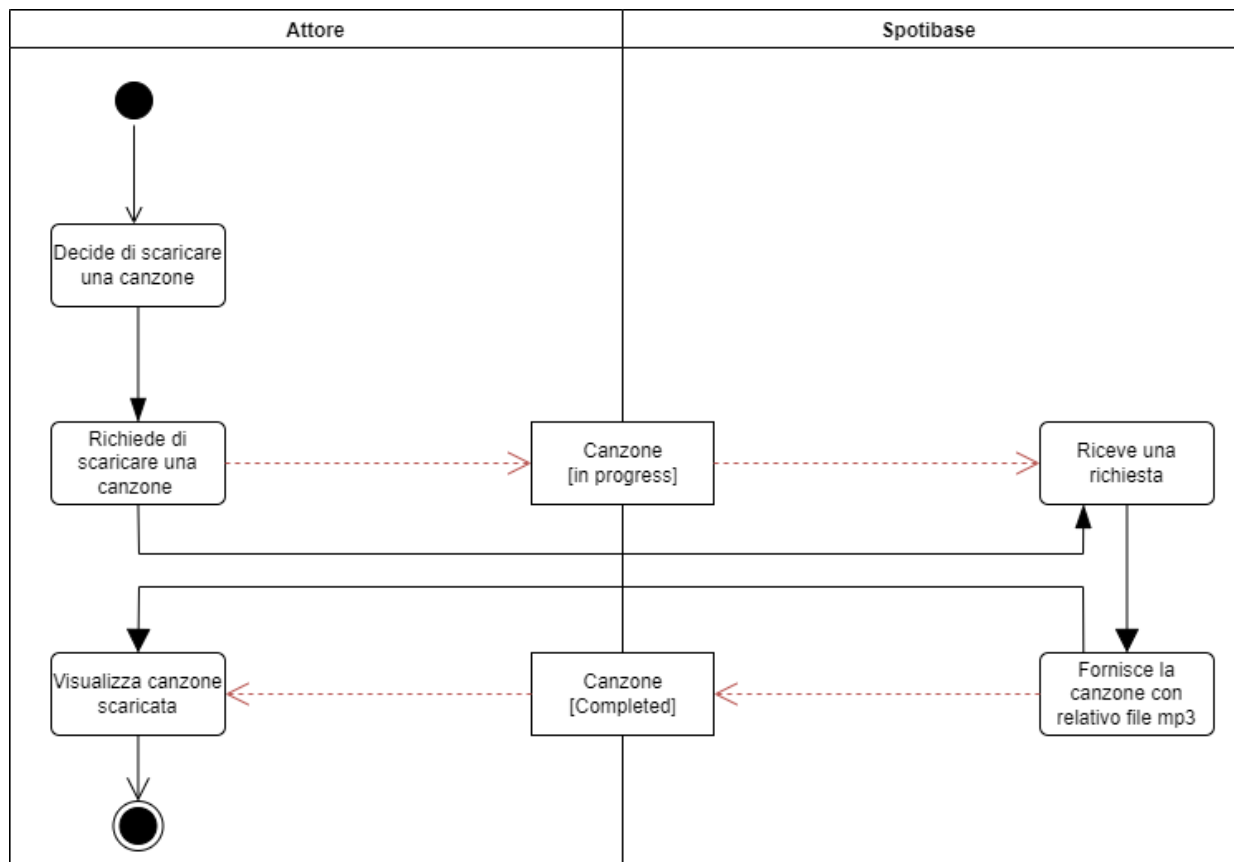
Questo class diagram mostra le classi individuate nelle fasi precedenti e come sono relazionate tra di loro. Considerando i vari oggetti entity individuati nell'object model.



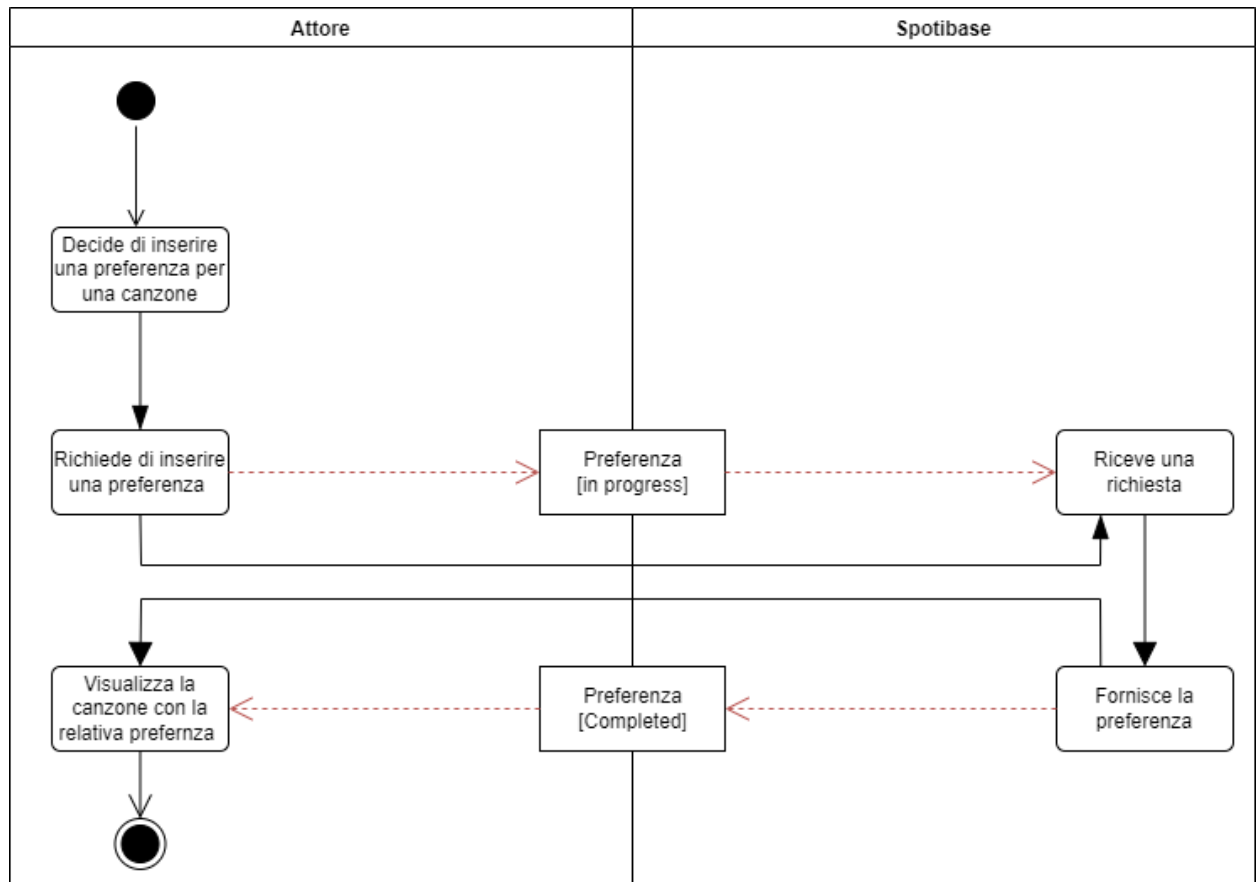
3.4.4 Dynamic model

3.4.4.1 Activity Diagram

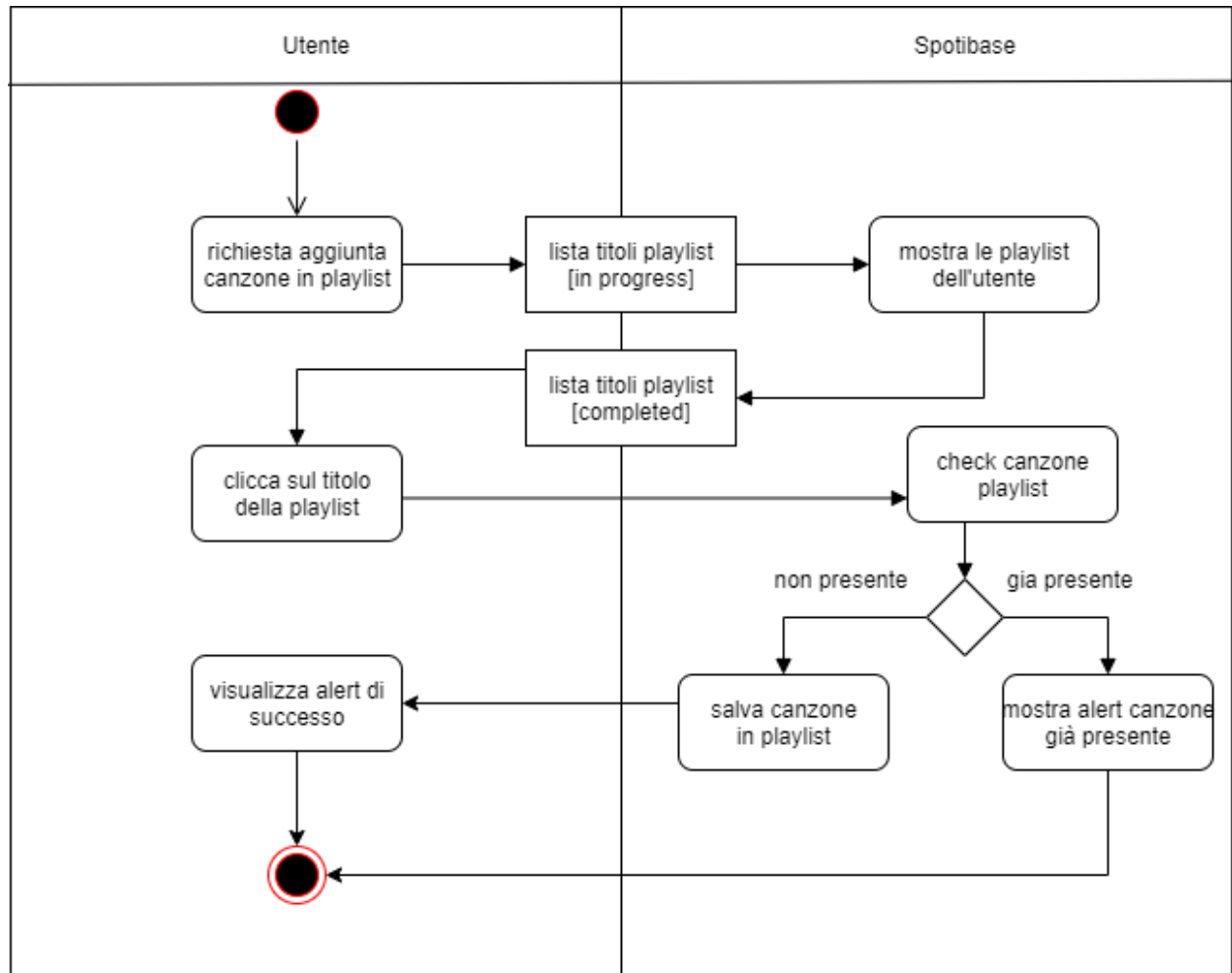
AD_UC1



AD_UC4



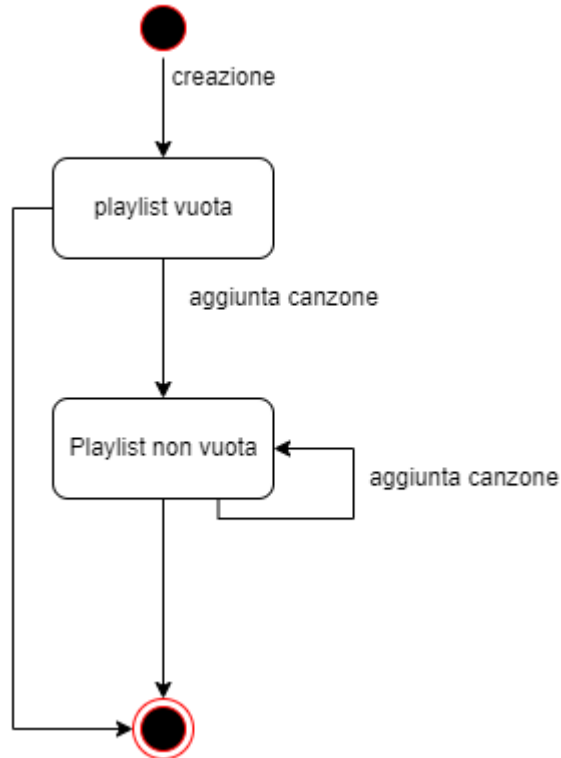
AD_UC8



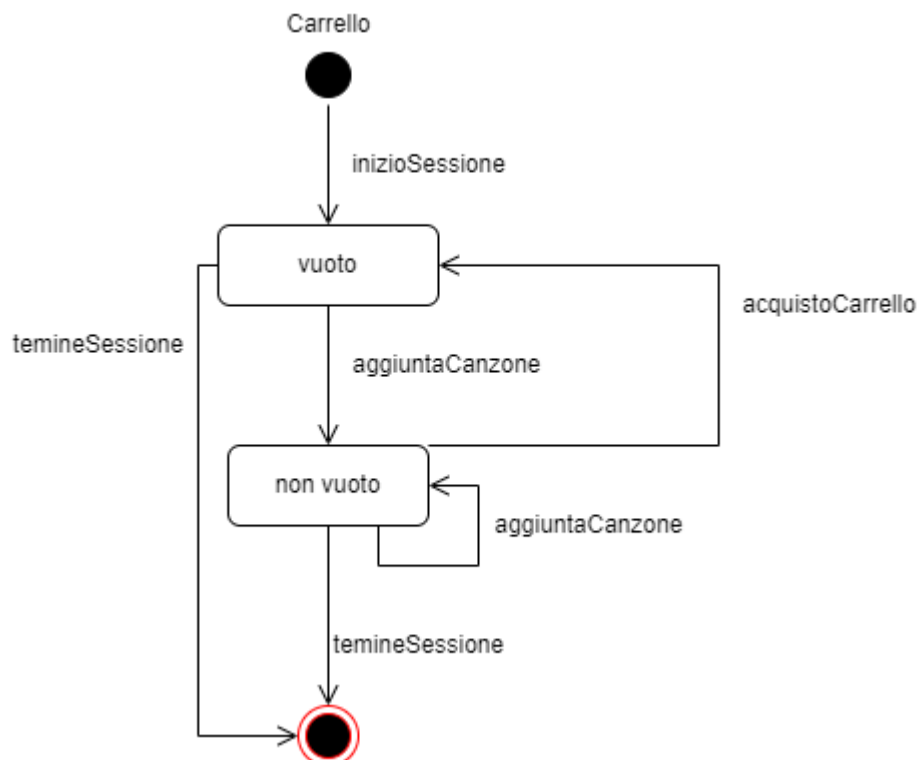


3.4.4.2 Statechart Diagram

SCD Playlist



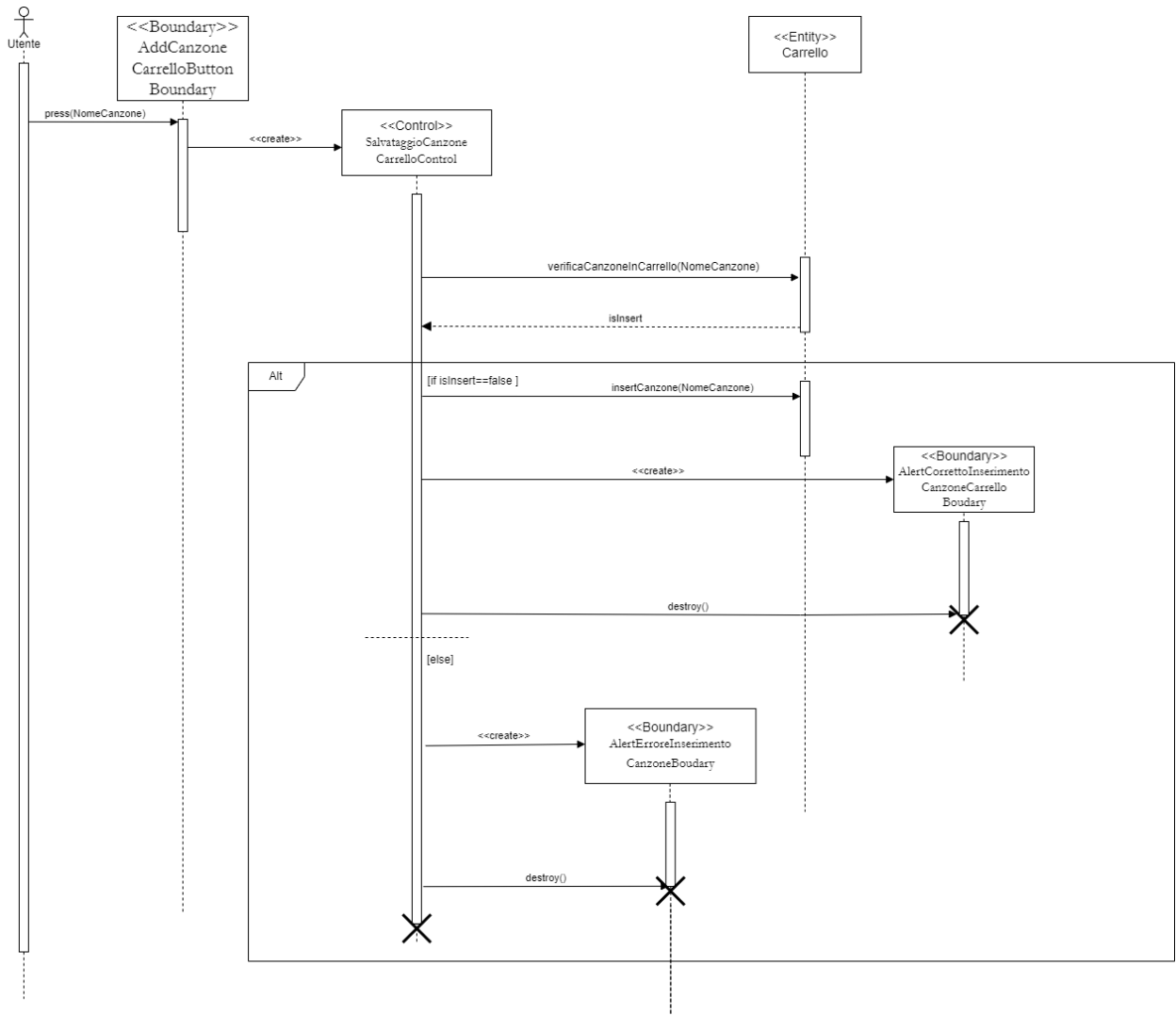
SCD Carrello



3.4.4.3 Sequence Diagram

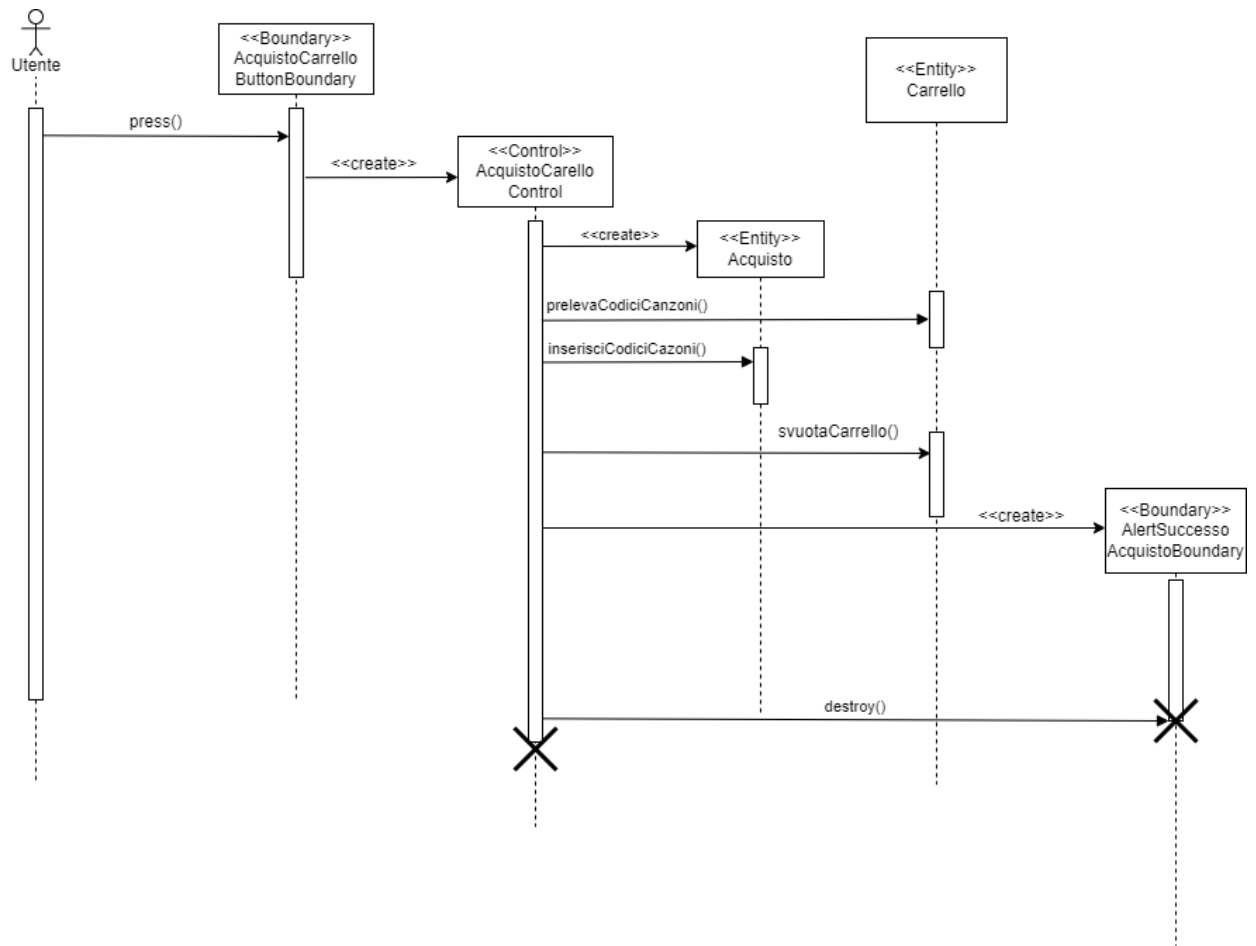
SD_UC_2

Aggiunta canzone nel carrello



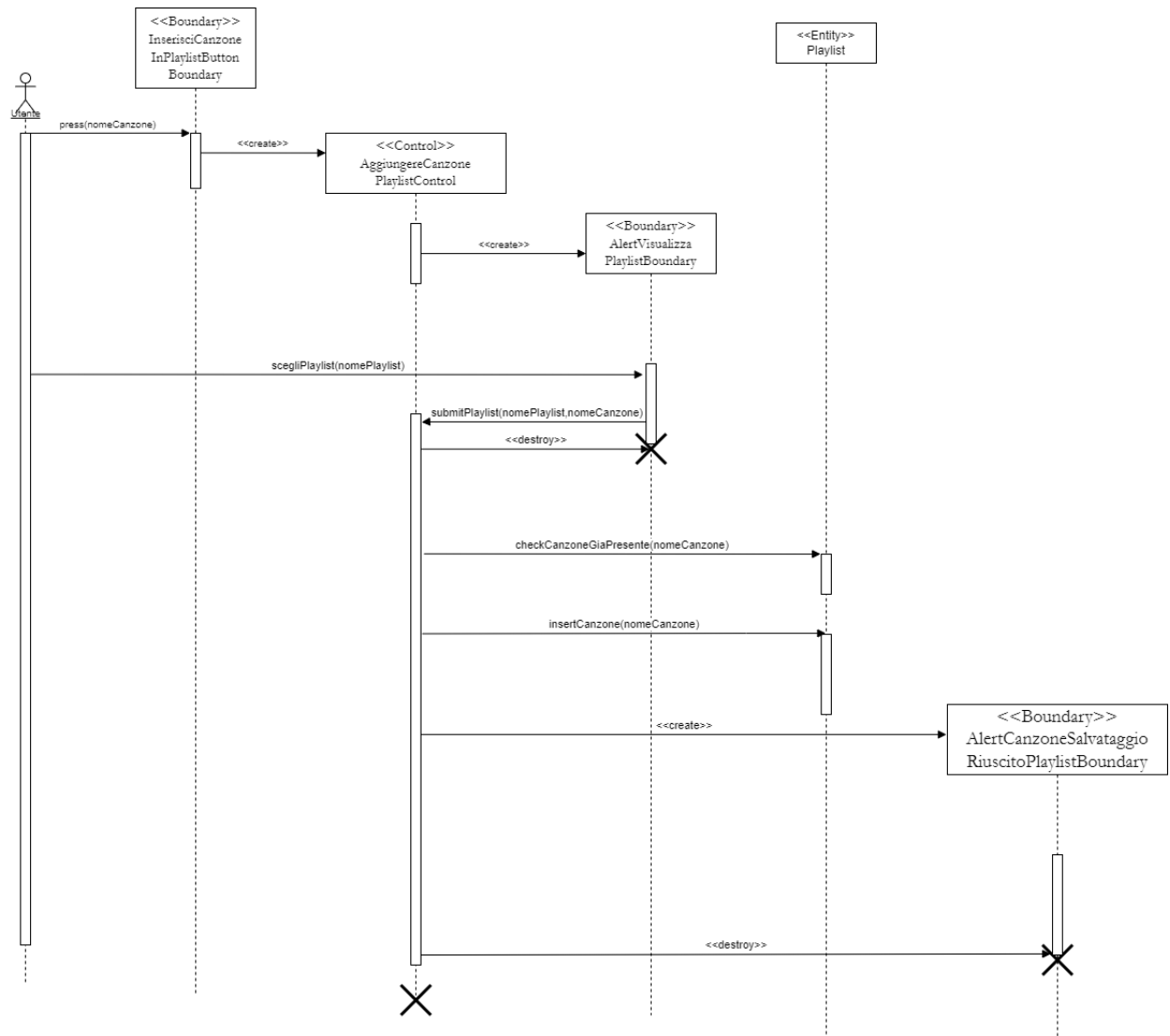
SD_UC_3

Acquisto del carrello



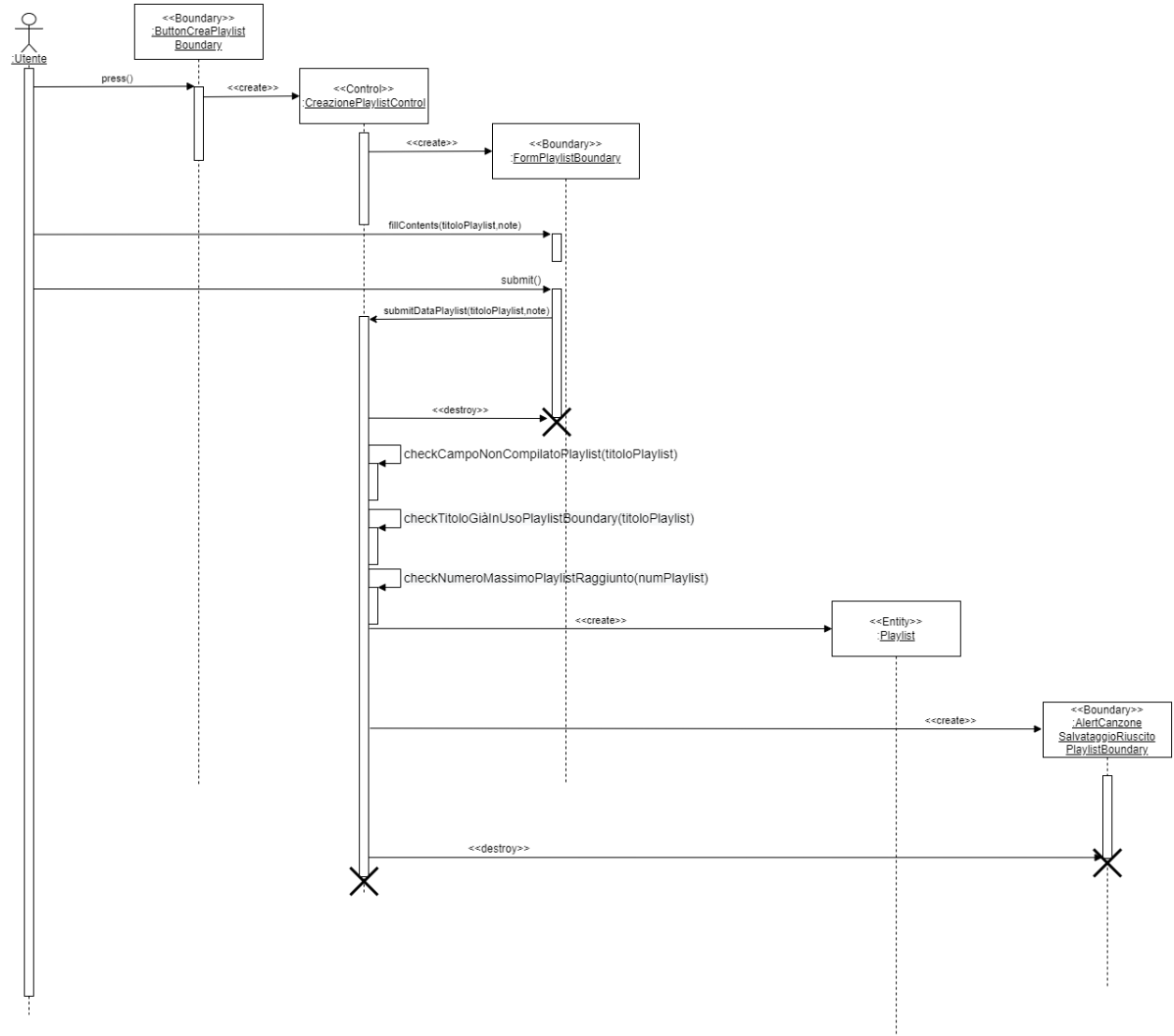
SD_UC_8

Inserimento canzone in playlist



SD_UC_9

Creazione playlist

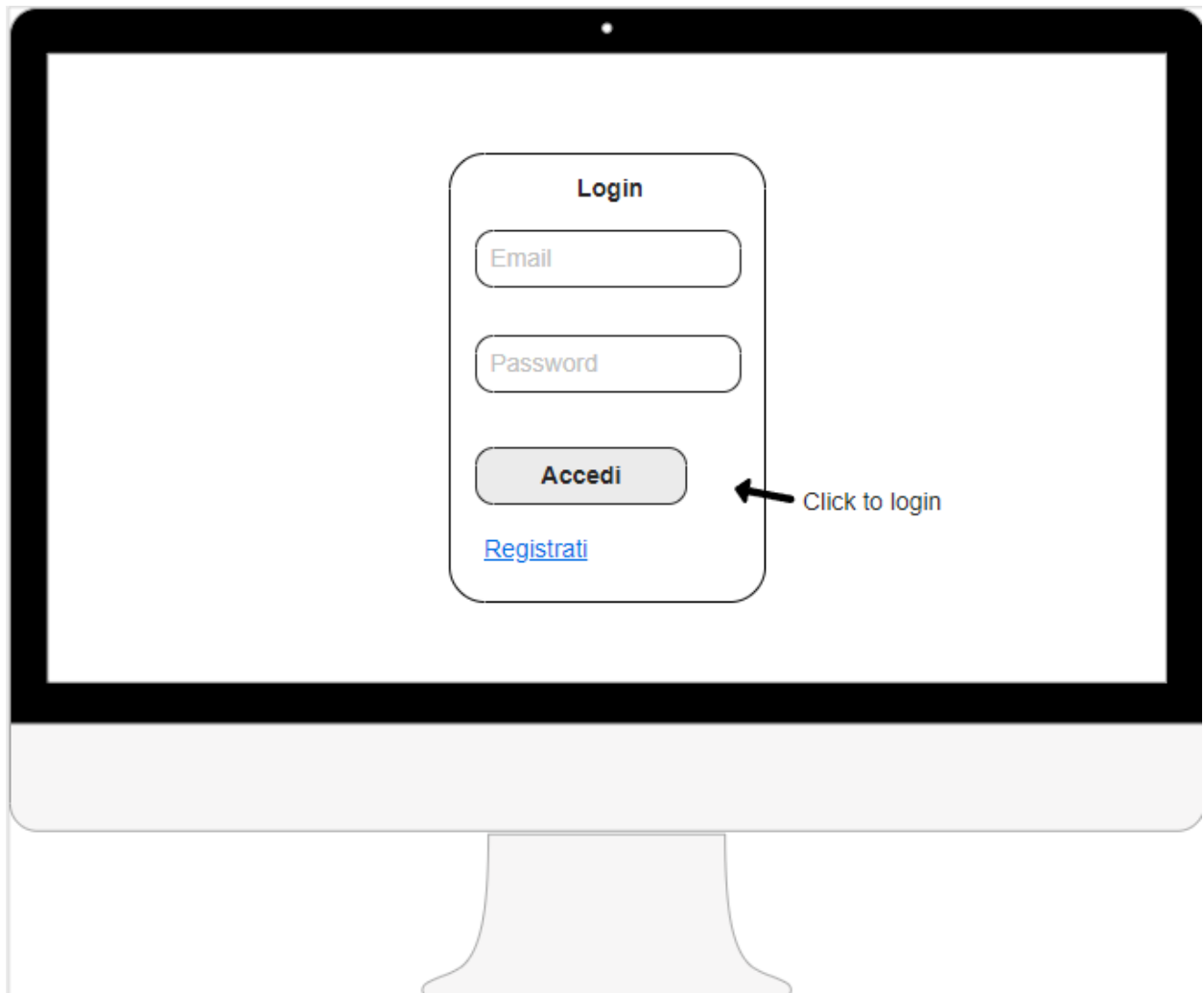




3.4.5 User interface - navigational paths e mock-ups

Area login/registrazione

MC1





Laurea Triennale in informatica-Università di Salerno
Corso di
Ingegneria del software- Prof. Carmine Gravino
MC2

Registrati

Username

Email

Password

Conferma Password

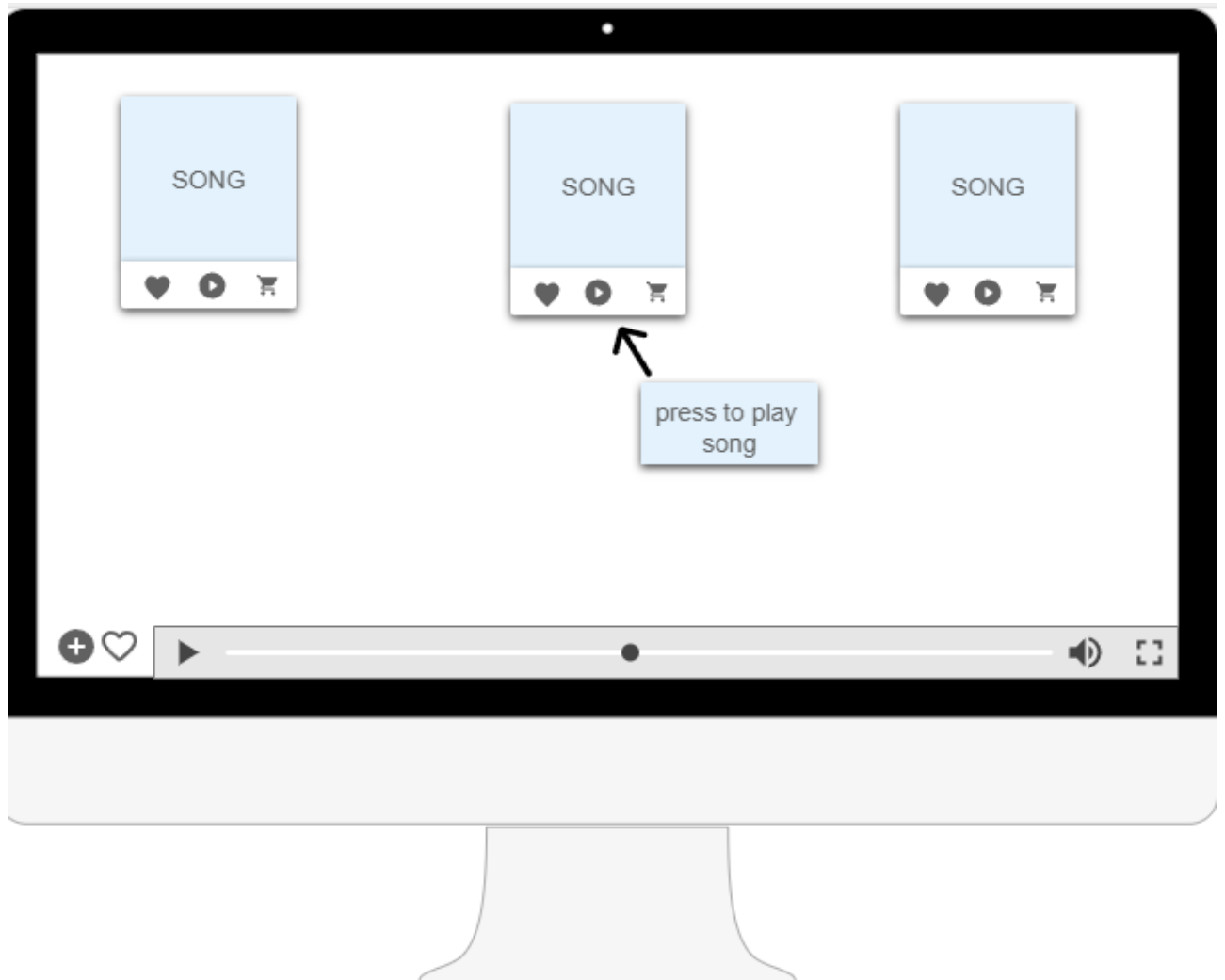
Registrati

[Hai già un account? Accedi](#)

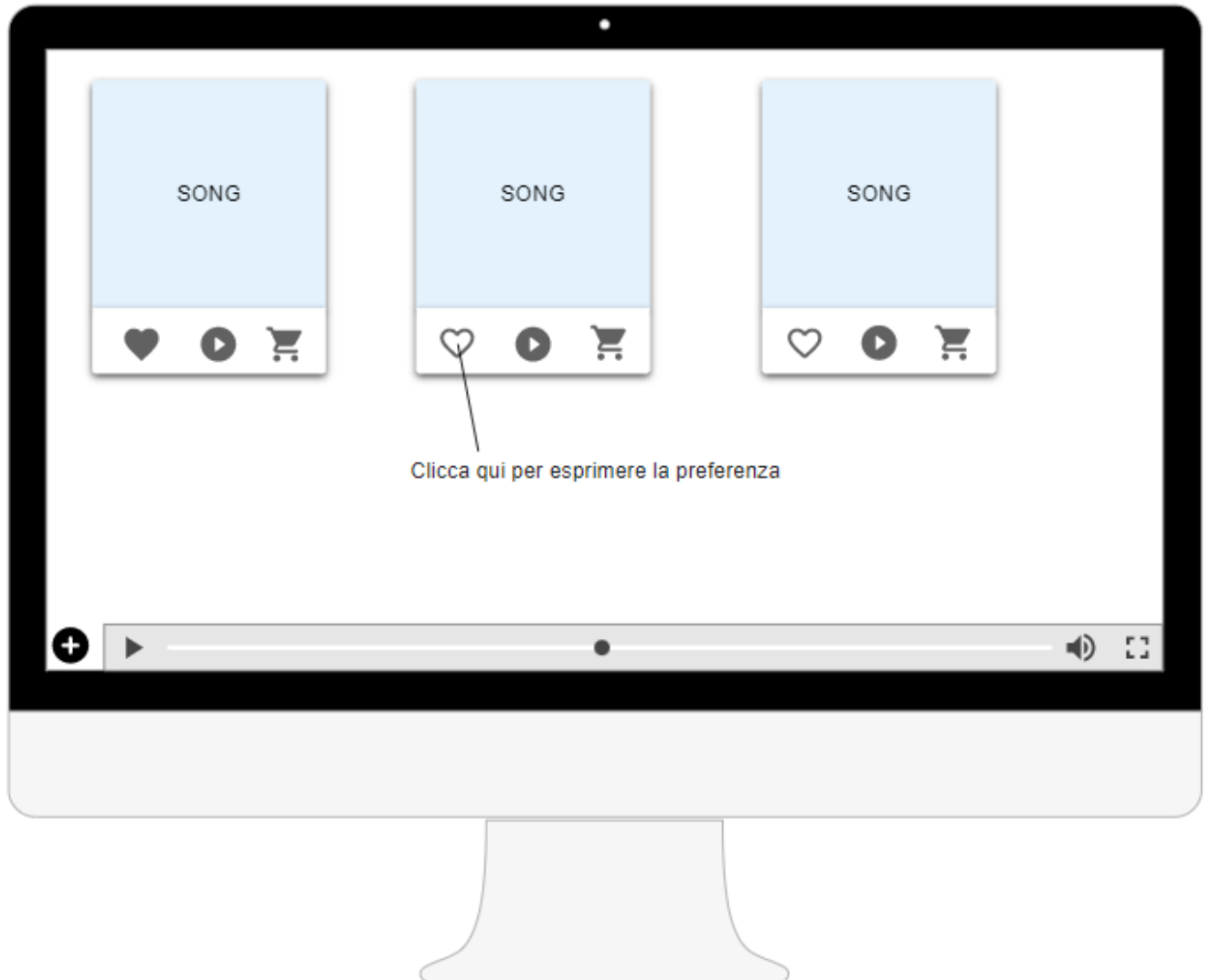
Click to signup



Laurea Triennale in informatica-Università di Salerno
Corso di
Ingegneria del software- Prof. Carmine Gravino
Area canzone
MC3



MC4



Area playlist M5

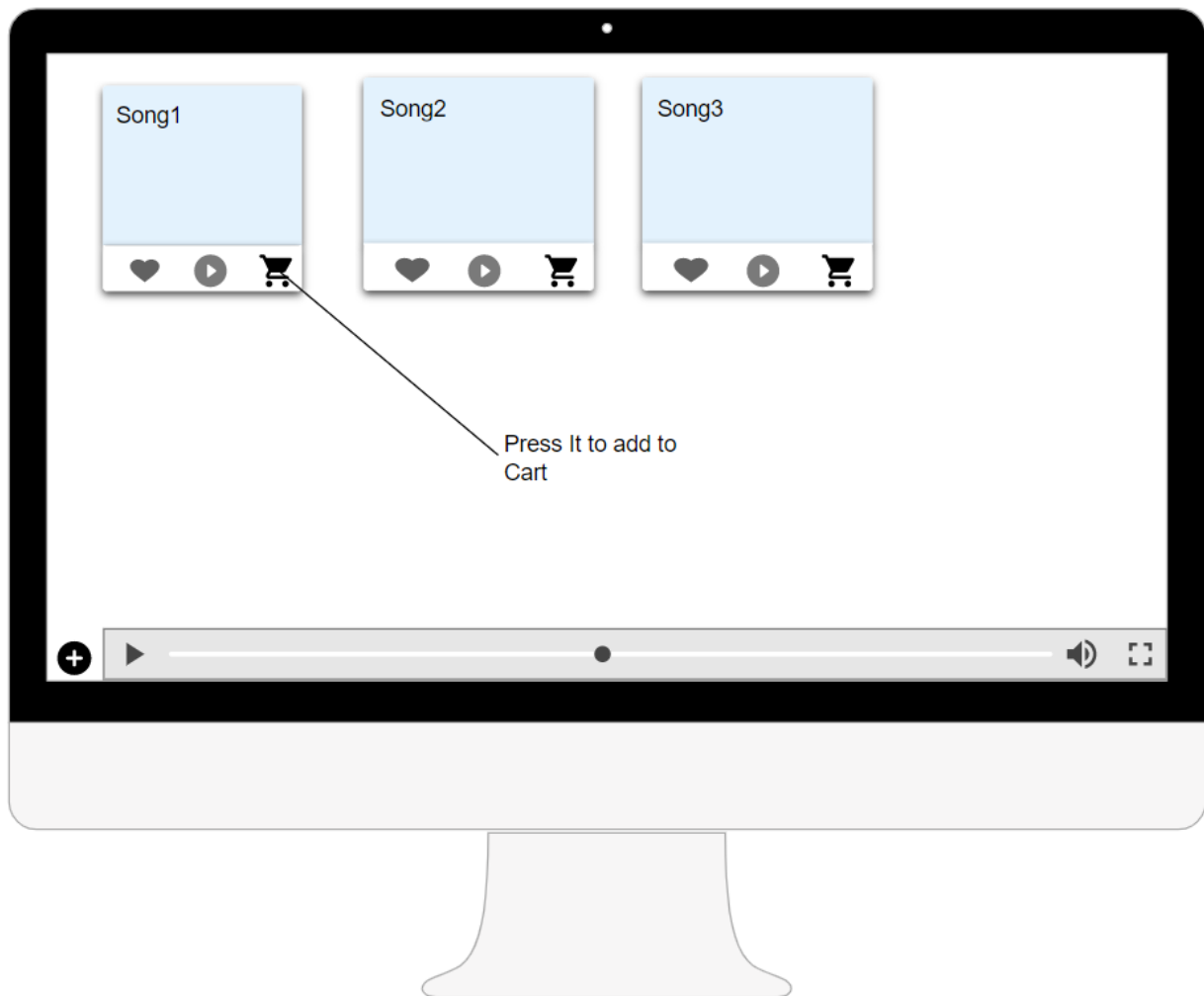


MC6



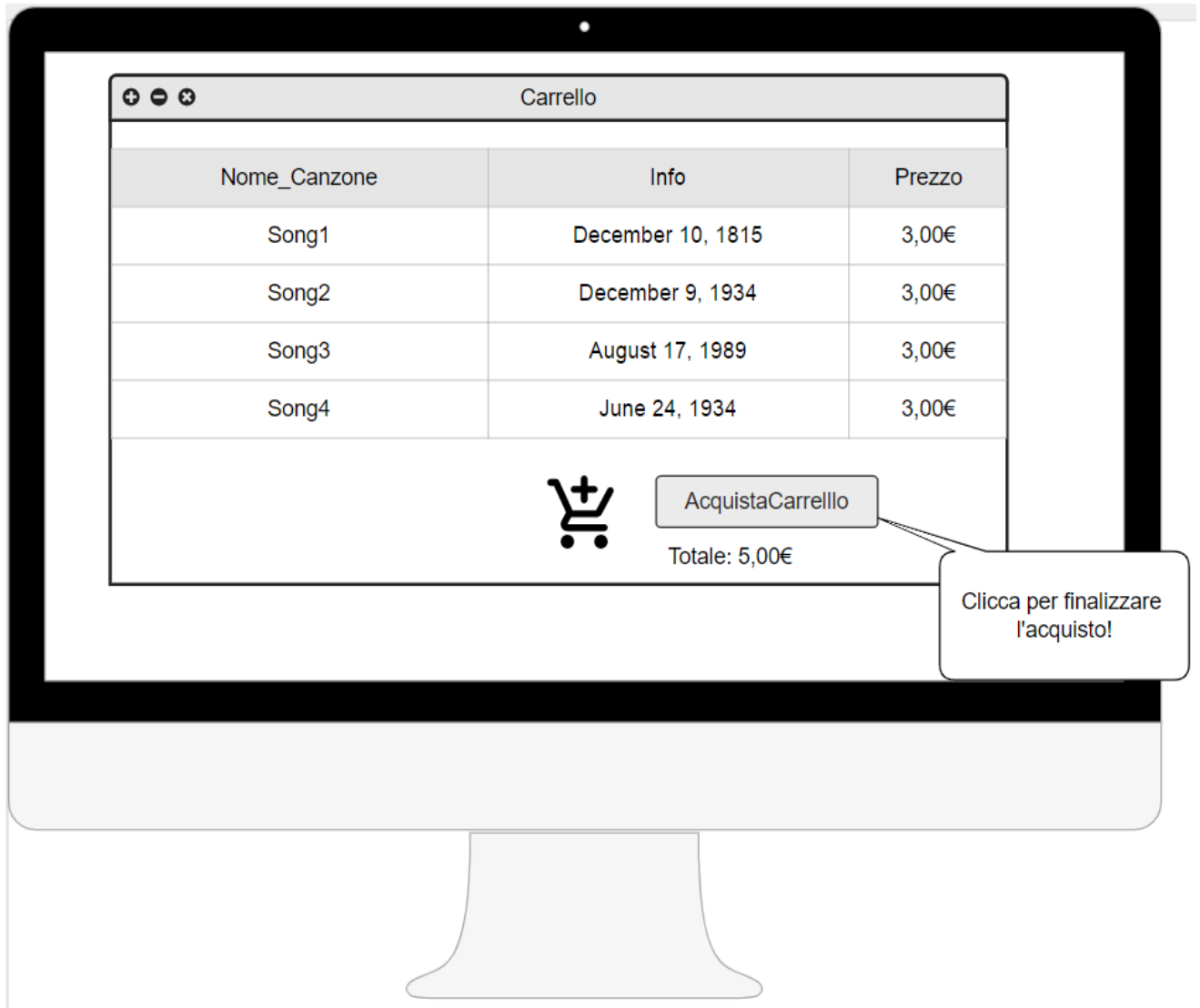


Area carrello MC7



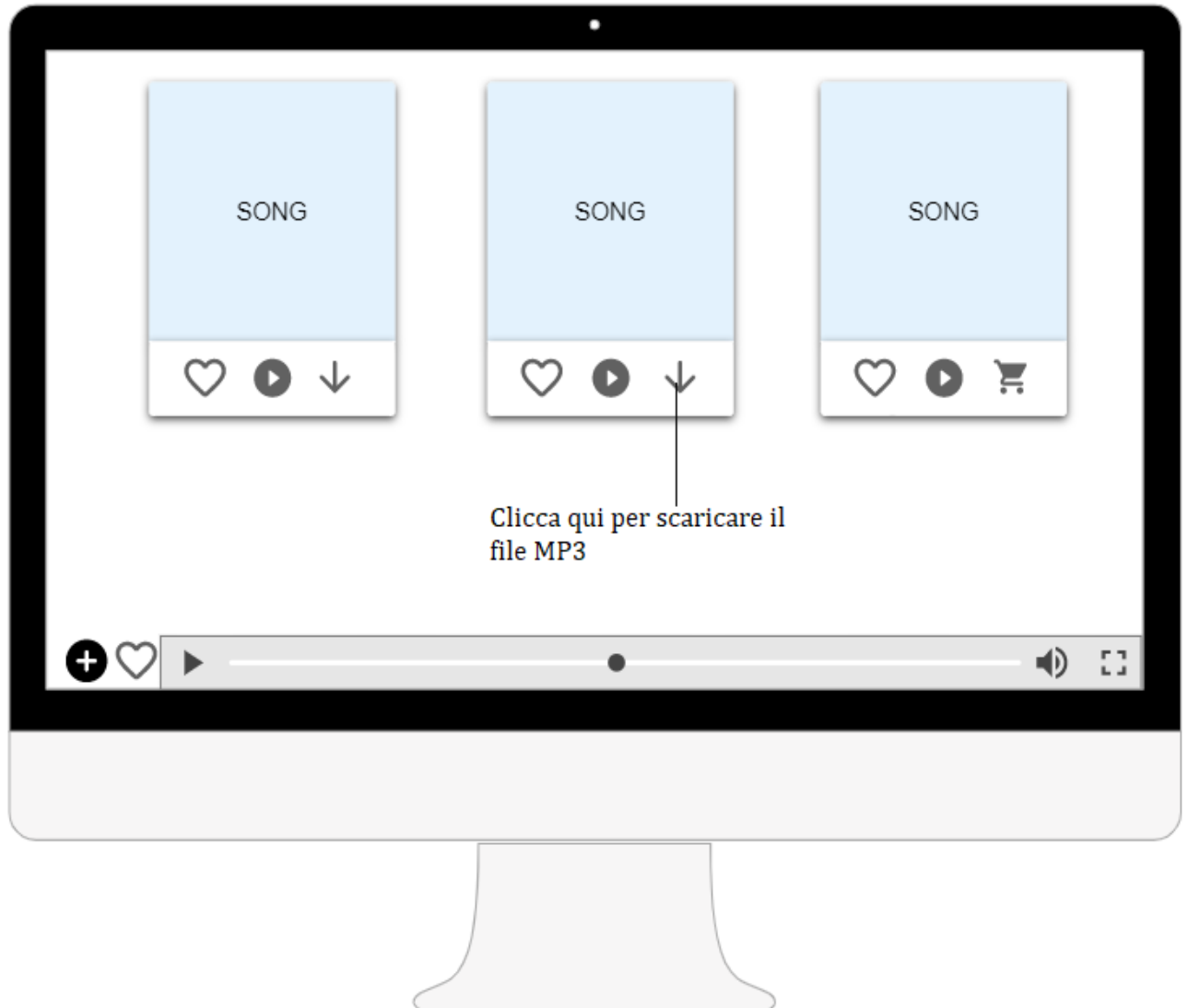


MC8



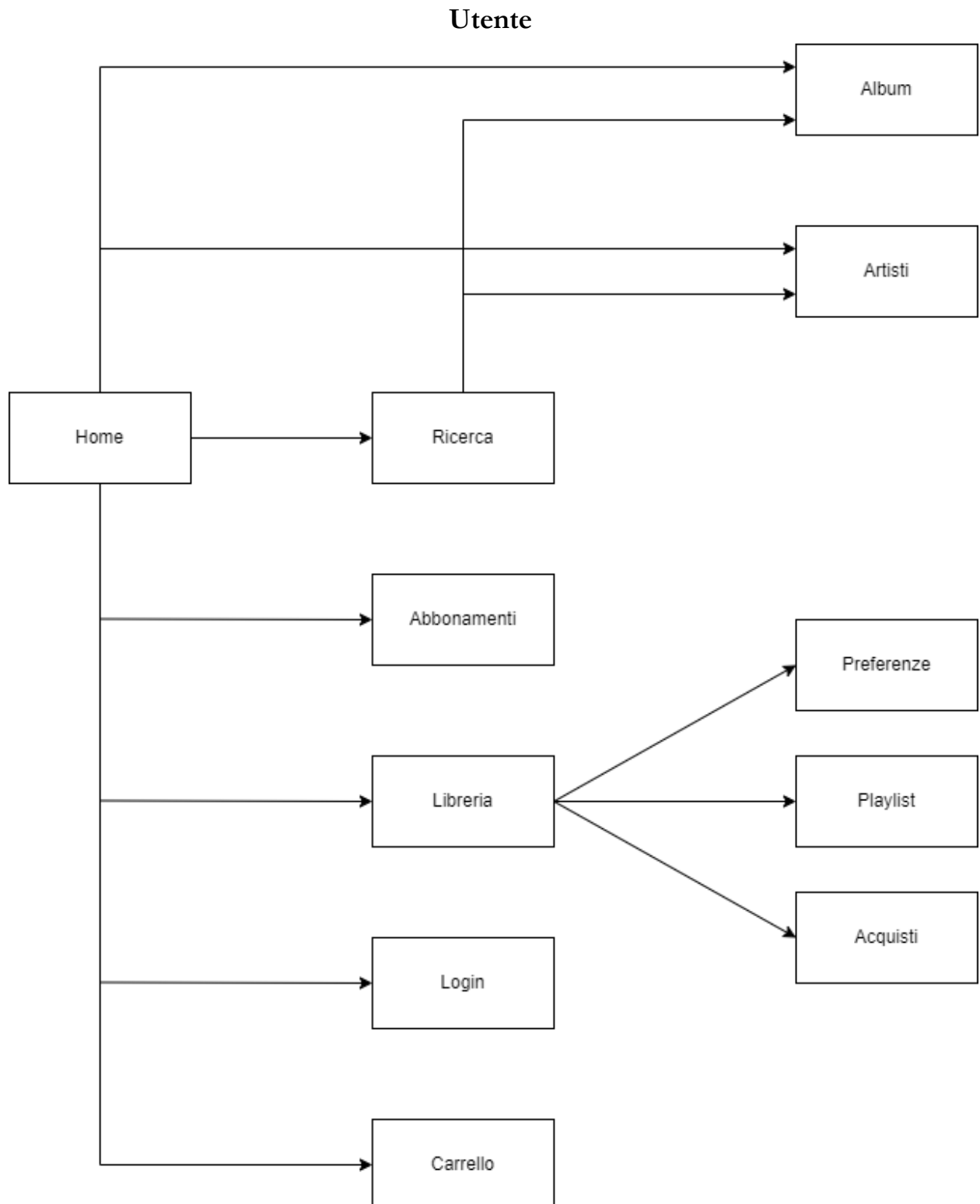


MC9





3.4.5.2 Navigational paths





4. Glossario

Playlist: un insieme di canzoni. L'utente può creare ed eliminare playlist, aggiungere e rimuovere canzoni.

Libreria: il sistema attraverso la libreria mostra le preferenze, gli acquisti, gli abbonamenti e le playlist dell'utente.

Alert: sinonimo di messaggio, notifica, avviso. Qualcosa che avverte l'utente riguardo l'avvenimento di un azione.

Form: insieme di campi da compilare.

Badge dell'utente: una notifica con all'interno un numero intero, notifica all'utente il numero di elementi nel carrello.