

Giovanni Maria Giglio (Gruppo 15)

Ingegneria Gestionale Informatica (curriculum informatico)

0646456

giovanni.giglio02@community.unipa.it

SpotiClient

24 maggio 2018

Sommario

1. Introduzione
 - a. Panoramica
 - b. Obiettivo del documento
 - c. Glossario
2. Sistema proposto
 - a. Requisiti funzionali
 - b. Requisiti non funzionali
 - c. Implementazione
3. Casi d'uso e diagrammi di sequenza
4. Struttura del database
5. Diagramma degli oggetti

1. Introduzione

PANORAMICA

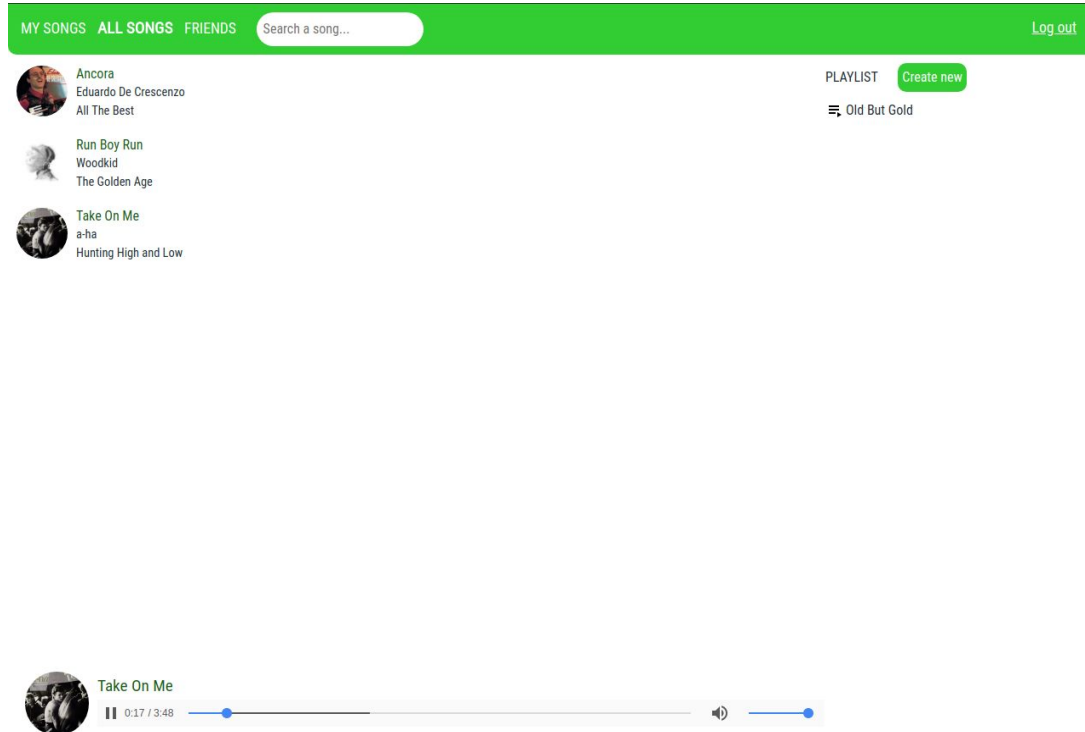
SpotiClient è una web app per lo streaming musicale da un server multimediale ad un dispositivo sia desktop che mobile.

OBIETTIVO DEL DOCUMENTO

Il presente documento si prefigge l'obiettivo di descrivere il sistema in termini di requisiti funzionali e non funzionali e fornisce la base contrattuale tra cliente e team di sviluppo.

GLOSSARIO

MainScreen:

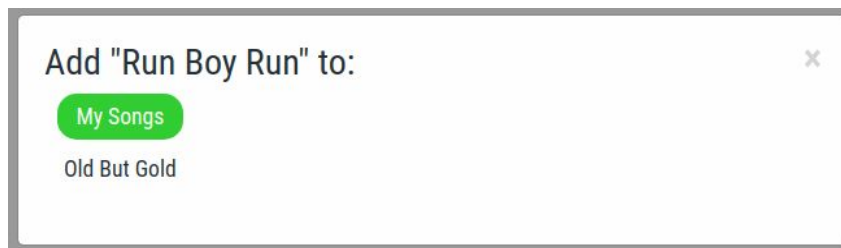


Schermata principale dalla quale è possibile eseguire tutte le operazioni.

Modal:

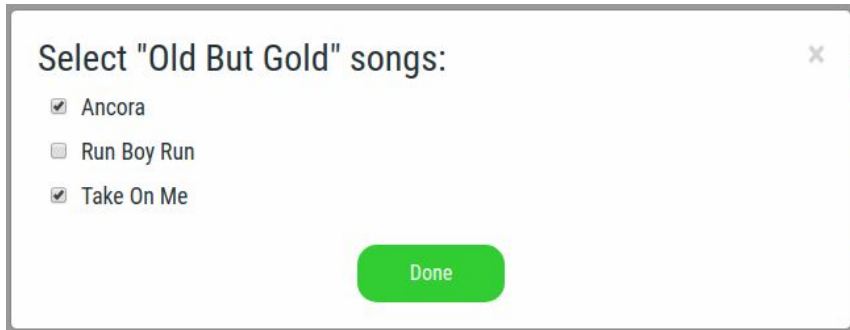
Un Modal è un popup che occupa una porzione dello schermo ed è usato per mettere in primo piano una particolare schermata. Quando un modal è attivo, la schermata in secondo piano non può essere utilizzata.

AddToPlaylist Modal:



Modal che consente l'inserimento del brano selezionato all'interno della raccolta personale dell'utente, o in una playlist.

EditPlaylist Modal:

A modal window titled "Select 'Old But Gold' songs:" with a close button (X) in the top right corner. It contains three items, each with a checkbox: "Ancora" (checked), "Run Boy Run" (unchecked), and "Take On Me" (checked). A green "Done" button is centered at the bottom.

Select "Old But Gold" songs:

- ☒ Ancora
- ☐ Run Boy Run
- ☒ Take On Me

Done

Modal che consente la modifica dei brani di una playlist. I brani possono essere inclusi o rimossi rispettivamente selezionando e deselectando le checkbox che li precedono. Il pulsante Done serve a confermare le modifiche e registrarle sul server.

CreatePlaylist Modal:

A modal window titled "Create new playlist" with a close button (X) in the top right corner. It features a label "Playlist name:" followed by a green input field containing the placeholder text "Your playlist name...". Below the input field is a green "Create" button.

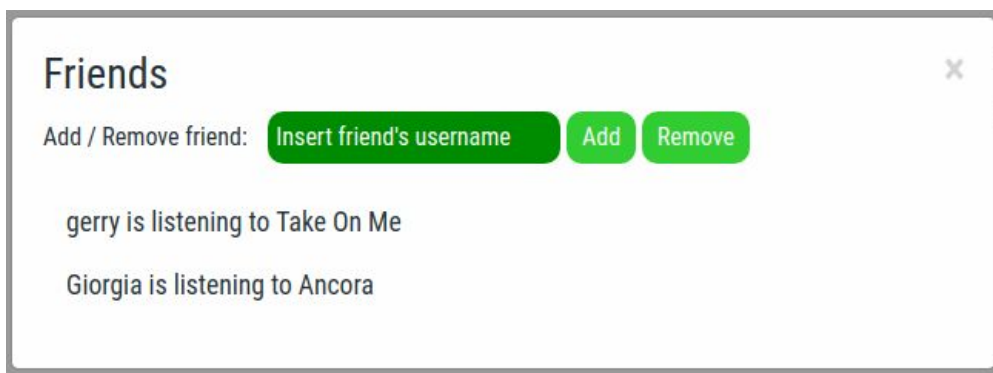
Create new playlist

Playlist name:

Create

Modal che consente la creazione di una playlist. L'input field conterrà il nome della playlist, ed il pulsante Create registrerà le modifiche sul server.

Friends Modal:

A modal window titled "Friends" with a close button (X) in the top right corner. It has a label "Add / Remove friend:" followed by a green input field with the placeholder "Insert friend's username". To the right of the input field are two green buttons: "Add" and "Remove". Below these elements, there are two lines of text: "gerry is listening to Take On Me" and "Giorgia is listening to Ancora".

Friends

Add / Remove friend:

gerry is listening to Take On Me

Giorgia is listening to Ancora

Modal che consente di aggiungere e rimuovere amici, conoscendone gli username, e di vedere cosa stanno ascoltando in questo momento i tuoi amici.

TracksList:



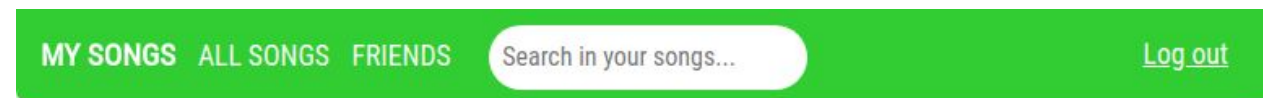
Sezione della Main Screen che contiene la lista delle tracce con le relative informazioni, quali titolo, artista, album. nonché l'immagine di copertina dell'album.

PlaylistsList:



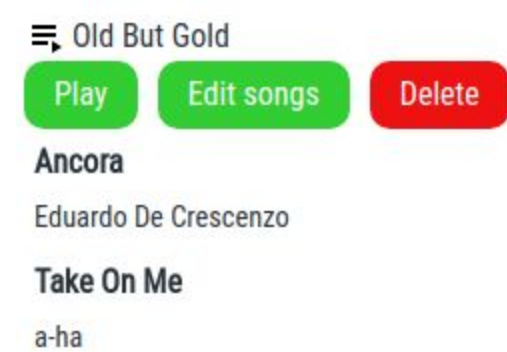
Sezione della main screen che contiene la lista delle playlist associate ad un utente, nonché un tasto per crearne di nuove.

Navbar:



Barra di navigazione che consente all'utente di interagire con il sistema.

PlaylistInfoSection:



Sezione della PlaylistsList che contiene la lista dei brani contenuti nella playlist, ed anche 3 pulsanti di controllo che servono rispettivamente a: riprodurre la sequenza di brani della playlist. visualizzare l'EditPlaylistModal e di eliminare la playlist.

2. Sistema proposto

Viene proposta una web app con una grafica accattivante che risulta sicura, facile nell'utilizzo e veloce. Di seguito ne vengono illustrati i requisiti e le funzionalità.

2a. Requisiti funzionali

Riproduzione in streaming di brani musicali presenti nel server:

I brani devono essere differenziati in due categorie: quelli globali e quelli appartenenti alla raccolta personale di un utente. La riproduzione dei brani deve avvenire in streaming per ridurre al massimo i tempi di attesa.

Creazione e modifica di playlist:

Il sistema deve consentire la creazione e la modifica di playlist, ovvero collezioni di brani. Non devono esserci limitazioni sul numero di playlist che un utente può creare, né tantomeno sulla loro capacità.

Aggiunta e rimozione di brani dalla raccolta personale:

Il sistema deve consentire la creazione di una raccolta personale, all'interno della quale salvare un sottoinsieme dei brani presenti sul server, nonché la sua modifica. Questa raccolta deve essere indipendente ed unica per ogni utente.

Ricerca di brani per titolo, artista ed album:

Il sistema deve consentire la ricerca dei brani differenziata per brani globali e brani presenti nella raccolta dell'utente. La ricerca deve potere essere eseguita a partire dal titolo, dall'artista o dall'album del brano ricercato.

Seguire un amico:

Il sistema deve consentire agli utenti di seguire altri utenti di cui conoscono lo username. L'aggiunta di un amico, non deve implicare che l'utente aggiunto sia amico dell'utente che lo ha aggiunto. Non devono esserci limitazioni sul numero di utenti che si possono seguire.

Panoramica sui brani che gli utenti amici stanno riproducendo.

Il sistema deve consentire la visualizzazione per ogni utente amico, del brano che sta riproducendo, se c'è.

2b. Requisiti non funzionali

● Affidabilità

- L'accesso alla web app sarà protetto da un sistema di autenticazione sicuro ed efficace.
- La web app farà uso di cookie per mantenere informazioni relative alla sessione degli utenti.
- Il database non memorizzerà dati sensibili (es. password) in chiaro.

● Prestazioni

- La web app presenterà tempi di risposta accettabili, anche in presenza di configurazioni hardware limitanti.

- **Adattabilità**

- La web app avrà un layout adattivo, per essere utilizzata al meglio sia da dispositivi mobili, quali smartphone e tablet, che da dispositivi desktop.

2c. Implementazione

Client:

Il Client fa uso dei seguenti framework:

- Bootstrap 4.1.1
- JQuery 3.3.1

Il Client dialoga con il server attraverso l'uso di `XMLHttpRequest`, incapsulando eventuali dati da trasmettere nel corpo del messaggio sotto forma di stringa JSON.

Tutte le richieste avranno l'attributo `withCredentials` impostato a `true` per aggiungere automaticamente l'header relativo al cookie ad ogni richiesta, che consente al server di identificare l'utente.

Tutte le richieste effettuate client-side sono gestite dal modulo `requests.js`

Applicazione Mobile:

La controparte mobile della web app è stata creata con Cordova.

Server:

Il server è realizzato interamente in javascript e fa uso dei seguenti moduli:

- Express
- body-parser
- mysql

Database:

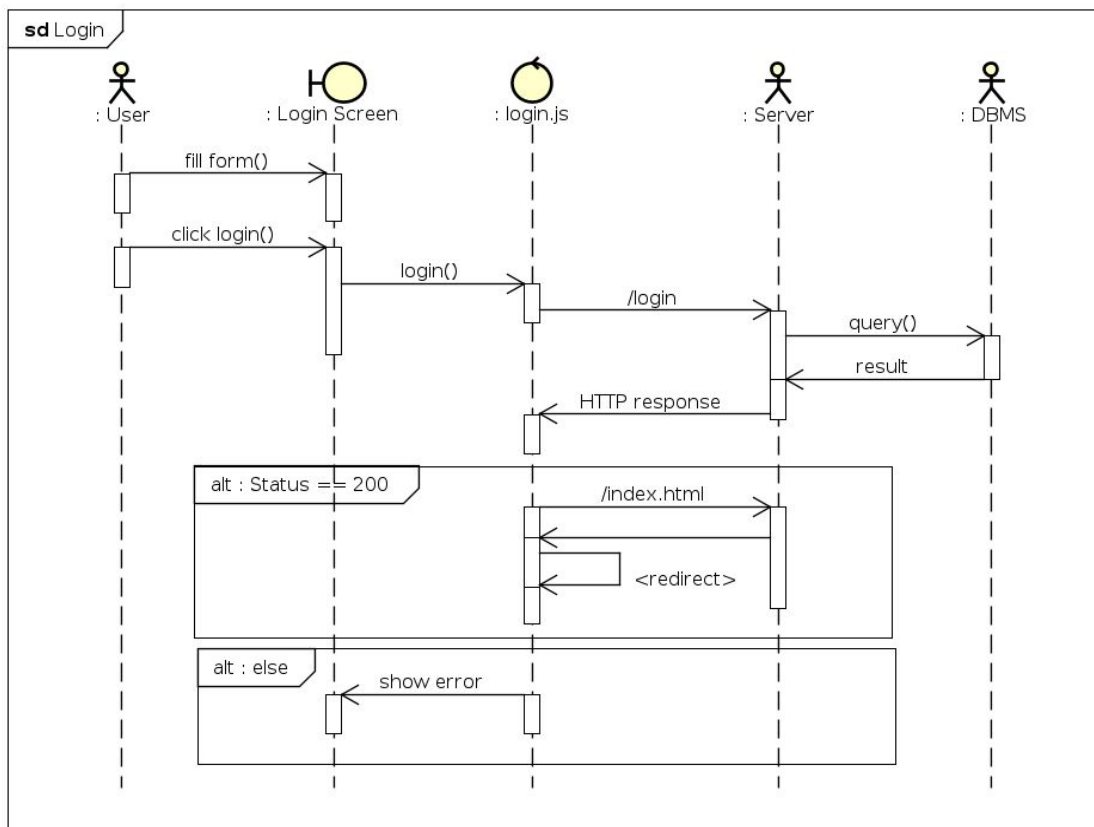
Il database di riferimento è Gruppo15 ed è gestito da mysql.

Per una maggiore sicurezza, il database non memorizza in chiaro le password associate agli utenti, bensì i loro hash, calcolati direttamente client-side per aggiungere un ulteriore strato di sicurezza anche allo scambio di messaggi HTTP col server.

3. Casi d'uso e diagrammi di sequenza

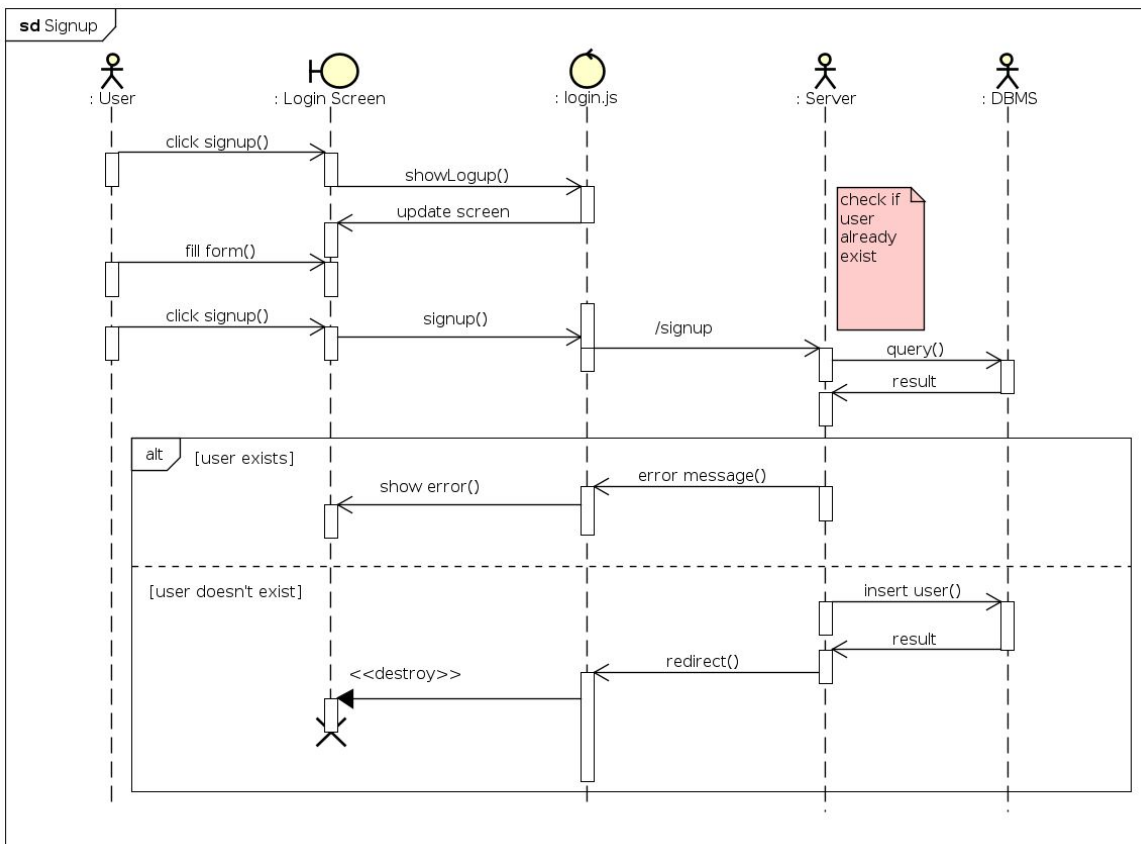
3.1. Login

Attori	Utente, Server
Precondizioni	Il sistema sta mostrando la pagina di accesso
Sequenza degli eventi	<ol style="list-style-type: none">1. Il sistema mostra una schermata con un form per l'inserimento di nome utente e password.2. L'utente compila i campi inserendo il proprio username e la propria password, quindi clicca 'Login'.3. Il sistema invia i dati al server per la validazione.4. Il server esegue la validazione dei dati interfacciandosi con il DBMS ed invia una risposta HTTP al client.5. Se il codice di stato della risposta è 200:<ol style="list-style-type: none">a. Il client effettua una redirectione alla pagina <i>index.html</i>.6. Altrimenti:<ol style="list-style-type: none">a. Il client mostra un messaggio di errore.
Postcondizioni	<ul style="list-style-type: none">- L'utente è loggato nella web app.- Il sistema sta mostrando la "Main Screen".



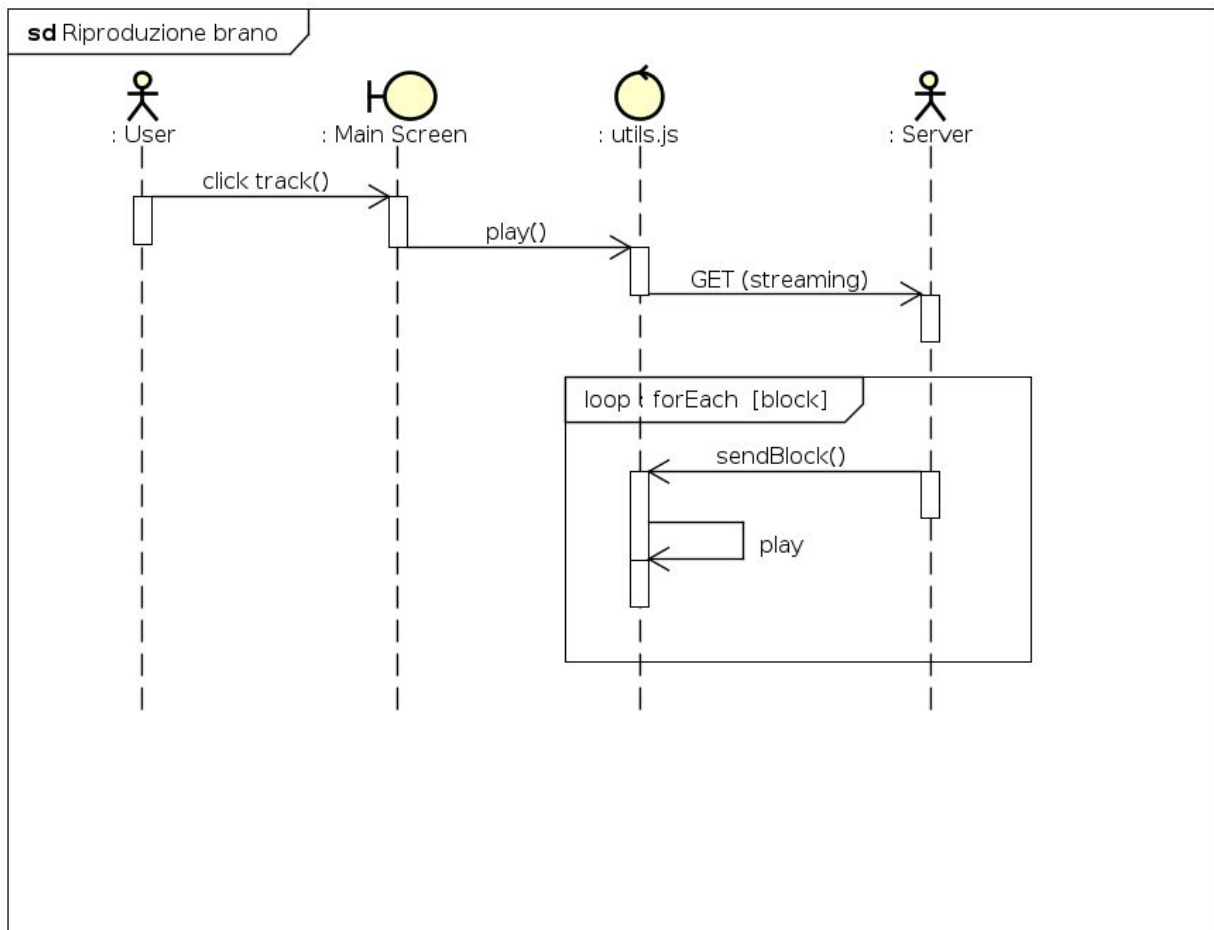
3.2. Registrazione

Attori	Utente, Server
Precondizioni	Il sistema sta mostrando la pagina di accesso
Sequenza degli eventi	<ol style="list-style-type: none"> 1. Il sistema mostra una schermata con un form per l'inserimento di nome utente e password ed un tasto per la registrazione di un nuovo utente. 2. L'utente clicca il pulsante per la registrazione di un nuovo utente. 3. Il sistema mostra un form per l'inserimento dei principali dati, quali username, email e password. 4. L'utente inserisce i dati richiesti 5. Il sistema invia i dati al server. 6. Il server si interfaccia con il DBMS per l'accesso al database. 7. Se esiste già un utente con lo stesso username. <ol style="list-style-type: none"> a. Il server invierà un messaggio di errore. 8. Altrimenti, il server memorizza le informazioni del nuovo utente all'interno del database, assegna un cookie all'utente e lo redireziona a <i>index.html</i>.
Postcondizioni	<ul style="list-style-type: none"> - L'utente è loggato nella web app. - Il sistema sta mostrando la "Main Screen".



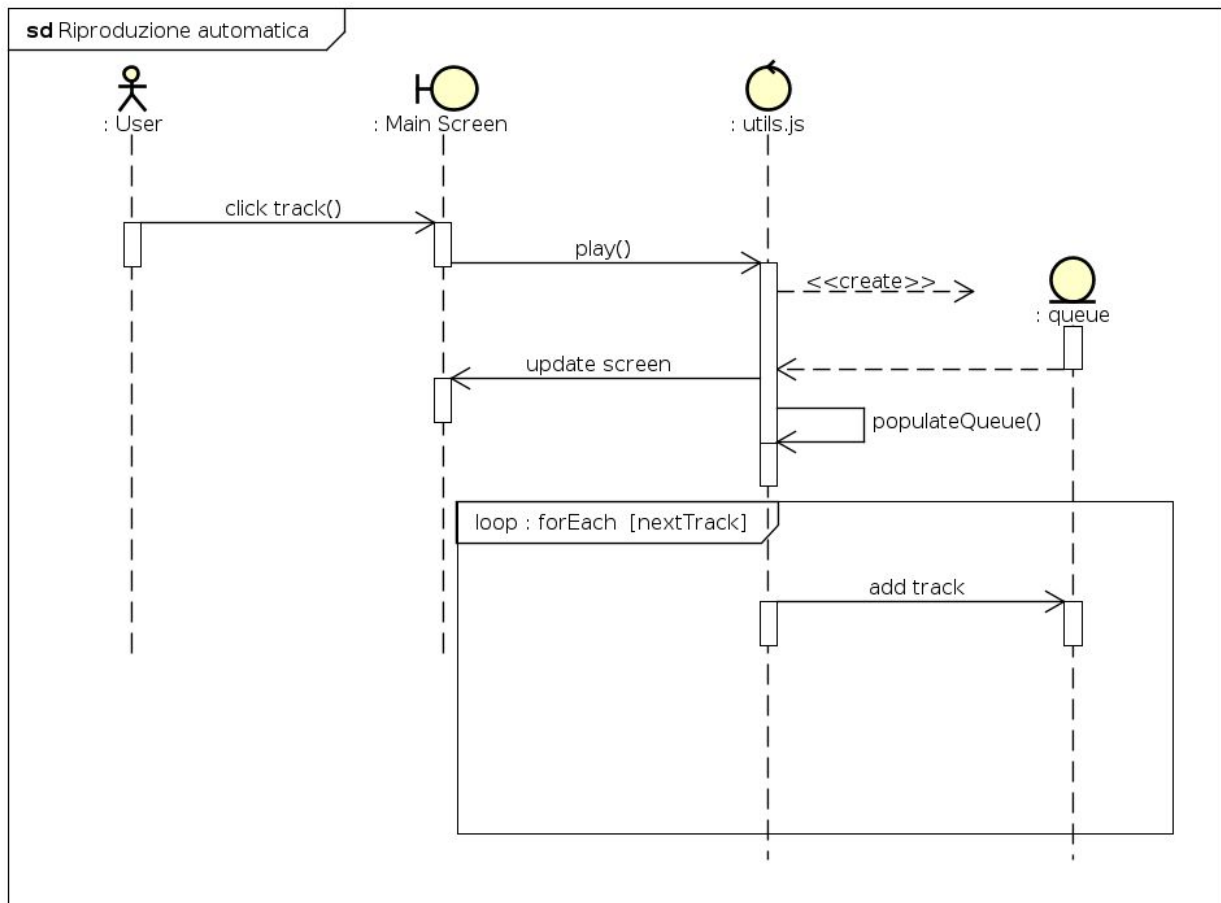
3.3 Riproduzione brano

Attori	Utente, Server
Precondizioni	<ul style="list-style-type: none"> - L'utente ha completato il caso d'uso "Login" - Il sistema sta mostrando la "Main Screen"
Sequenza degli eventi	<ol style="list-style-type: none"> 1. L'utente clicca sull'elemento della "Tracklist" che rappresenta il brano che intende riprodurre. 2. Il client richiede la traccia al server ed intanto inizia a riprodurla in streaming. 3. Il client invoca il caso d'uso Aggiorna Listening Song
Postcondizioni	<ul style="list-style-type: none"> - Il sistema sta riproducendo il brano richiesto. - Il sistema sta mostrando la "Main Screen".



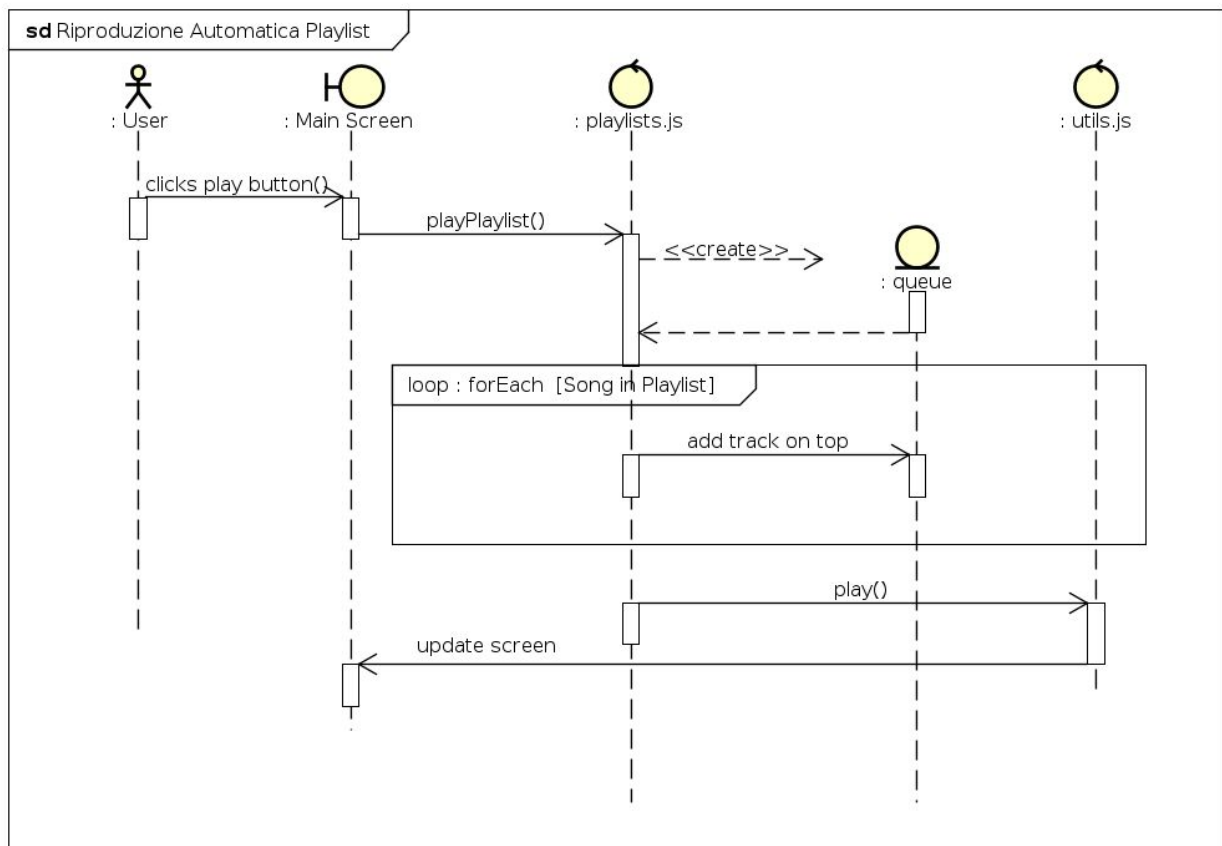
3.3 Riproduzione automatica

Attori	Utente, Server
Precondizioni	<ul style="list-style-type: none">- L'utente ha completato il caso d'uso "Login"- Il sistema sta riproducendo un brano.
Sequenza degli eventi	<ol style="list-style-type: none">1. L'utente clicca su un traccia audio.2. Il sistema inizia la sua riproduzione.3. Il sistema aggiunge alla coda di riproduzione le prossime tracce che l'utente sta visualizzando.4. Al termine della riproduzione del brano corrente il sistema, riproduce il prossimo brano e ricalcola la coda di riproduzione.
Postcondizioni	<ul style="list-style-type: none">- Il sistema sta riproducendo il brano richiesto.



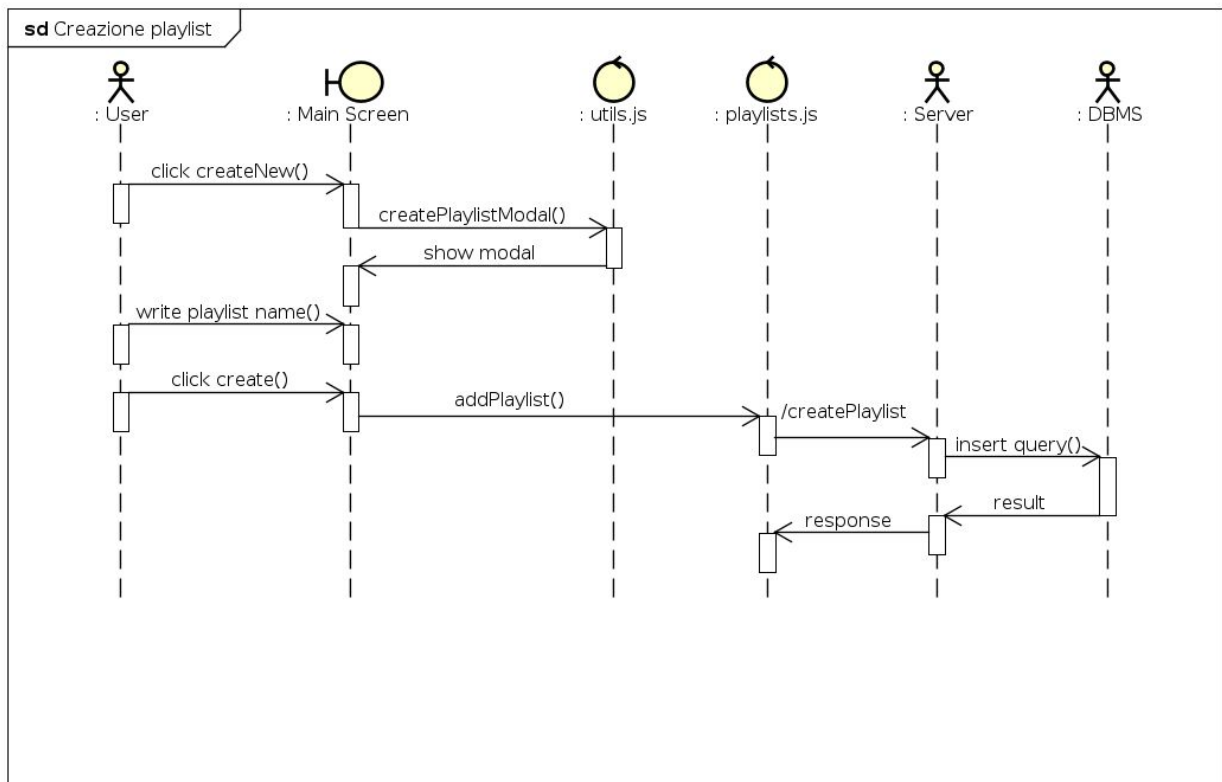
3.4 Riproduzione automatica Playlist

Attori	Utente, Server
Precondizioni	<ul style="list-style-type: none"> - L'utente ha completato il caso d'uso "Login" - L'utente ha cliccato il tasto <i>Play</i> della <i>Playlist Info Section</i>.
Sequenza degli eventi	<ol style="list-style-type: none"> 1. L'utente clicca Play dalla <i>Playlist Info Section</i>. 2. Il sistema aggiunge alla coda di riproduzione le tracce contenute nella playlist che l'utente intende riprodurre. 3. Il sistema avvia la riproduzione del primo brano della coda. 4. Al termine della riproduzione del brano corrente il sistema, riproduce il prossimo brano della playlist.
Postcondizioni	<ul style="list-style-type: none"> - Il sistema sta riproducendo il brano richiesto.



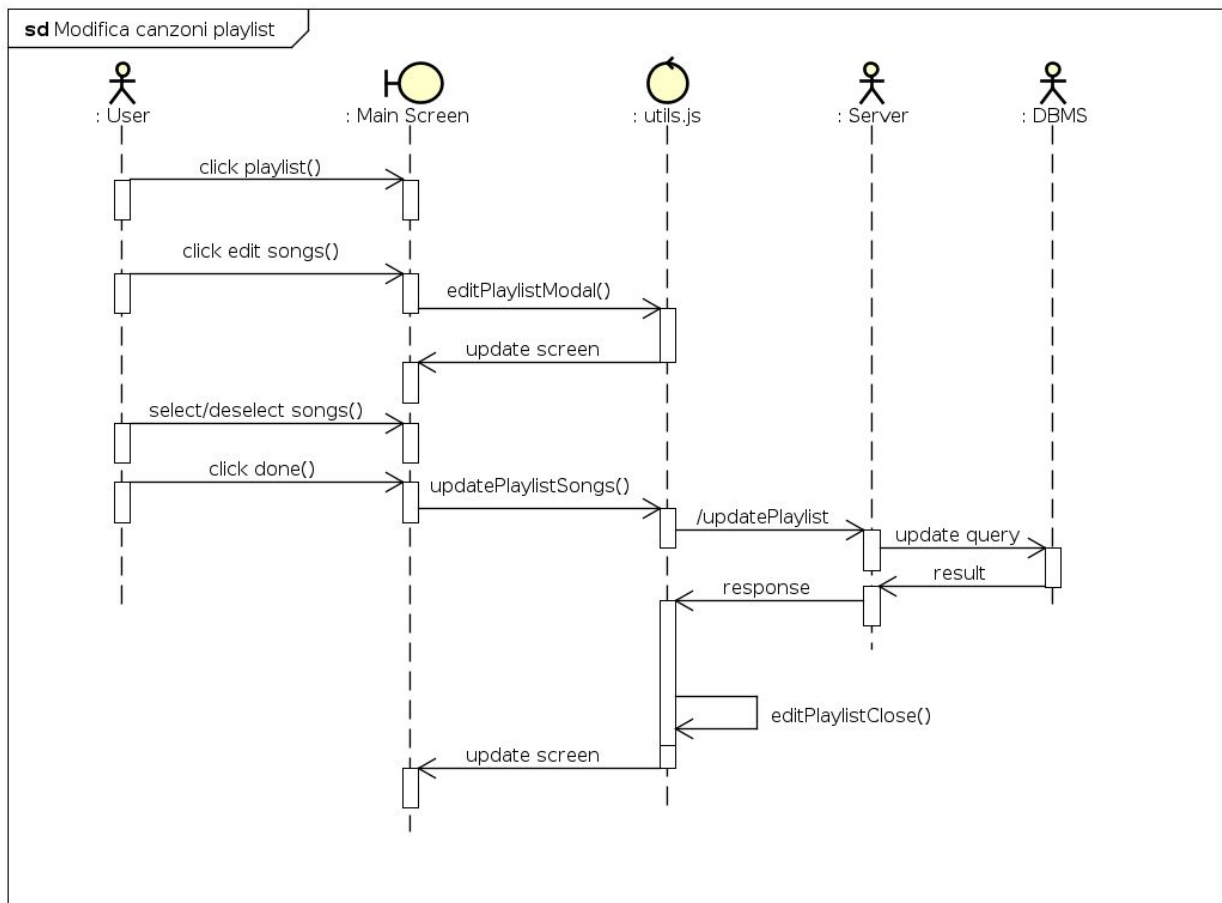
3.5 Creazione Playlist

Attori	Utente, Server, DBMS
Precondizioni	<ul style="list-style-type: none"> - L'utente ha completato il caso d'uso "Login" - Il sistema sta mostrando la "Main Screen"
Sequenza degli eventi	<ol style="list-style-type: none"> 1. L'utente clicca sul pulsante <i>Create new</i>, posto in cima alla <i>Playlists List</i>. 2. Il sistema mostra il <i>CreatePlaylist Modal</i>. 3. L'utente inserisce il nome della playlist, e clicca "Create". 4. Il sistema comunica al server i dettagli della nuova playlist, che registra nel database. 5. Il server ritorna un messaggio, il cui codice di stato descrive l'esito dell'operazione di inserimento nel database.
Postcondizioni	<ul style="list-style-type: none"> - Il server ha registrato la playlist. - Il sistema sta mostrando la "Main Screen".



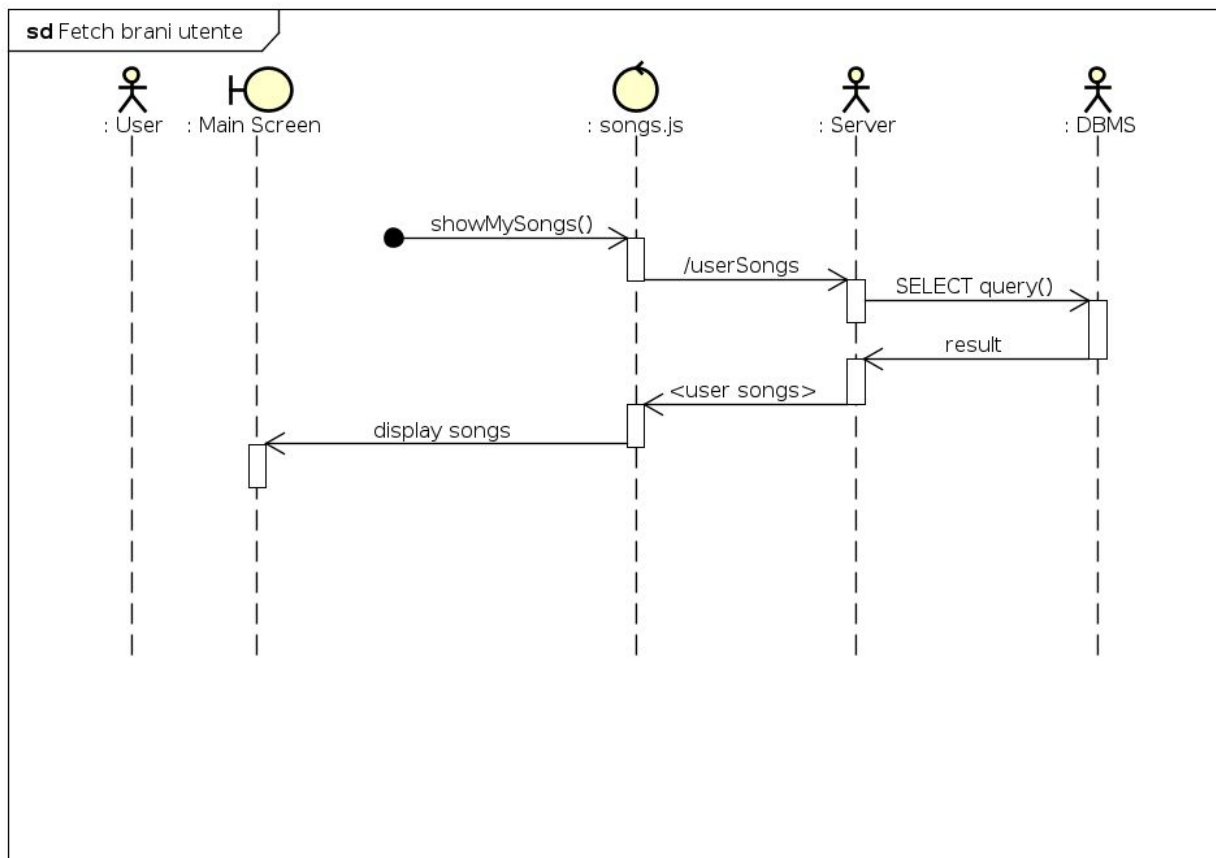
3.6 Modifica Canzoni Playlist

Attori	Utente, Server, DBMS
Precondizioni	<ul style="list-style-type: none"> - L'utente ha completato il caso d'uso "Login" - Il sistema sta mostrando la "Main Screen"
Sequenza degli eventi	<ol style="list-style-type: none"> 1. L'utente clicca sulla playlist che intende modificare, dalla <i>PlaylistsList</i>, quindi una volta espansa la <i>PlaylistInfoSection</i> clicca sul tasto <i>Edit Songs</i>. 2. Il sistema mostra a video l'<i>EditPlaylist Modal</i>. 3. L'utente aggiunge e rimuove le canzoni di una playlist, selezionando e deselezionando le relative checkbox, quindi clicca <i>Done</i>. 4. Il sistema comunica al server le modifiche da apportare alla playlist. 5. Il server si interfaccia con il DBMS per la registrazione delle modifiche della playlist. 6. Il server ritorna un messaggio il cui codice di stato descrive l'esito dell'operazione di modifica della playlist nel database.
Postcondizioni	<ul style="list-style-type: none"> - Il server ha registrato le modifiche apportate alla playlist. - Il sistema sta mostrando la "Main Screen".



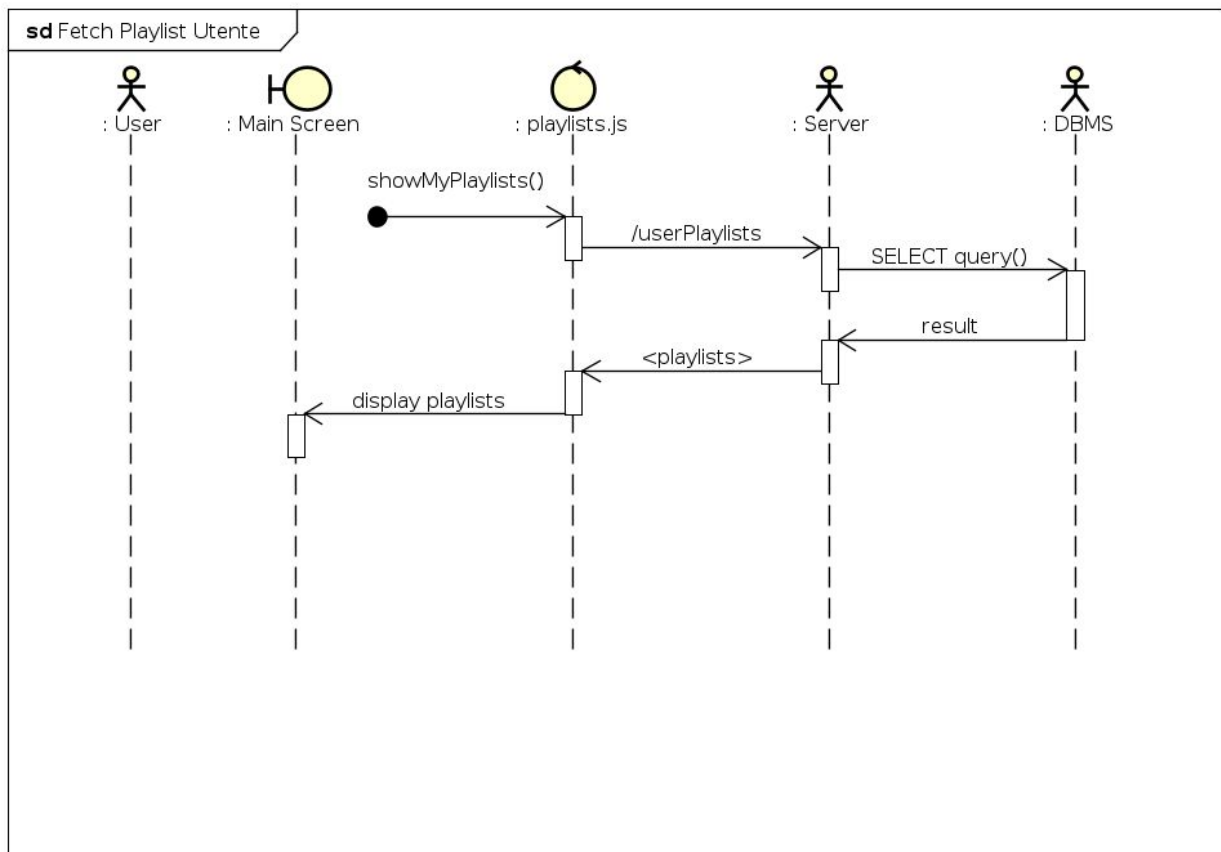
3.7 Fetch brani utente

Attori	Utente, Server, DBMS
Precondizioni	- L'utente ha completato il caso d'uso "Login"
Sequenza degli eventi	<ol style="list-style-type: none">1. Il client richiede al server la lista dei brani ad egli associati.2. Il server identifica l'utente in base al cookie, incluso nella richiesta. ed effettua quindi una query al DBMS per richiedere le informazioni sui brani associati a quell'utente.3. Il server invia sotto forma di JSON la lista delle informazioni dei brani associati all'utente.
Postcondizioni	<ul style="list-style-type: none">- Il sistema sta riproducendo il brano richiesto.- Il sistema sta mostrando la "Main Screen" ed i brani associati all'utente.



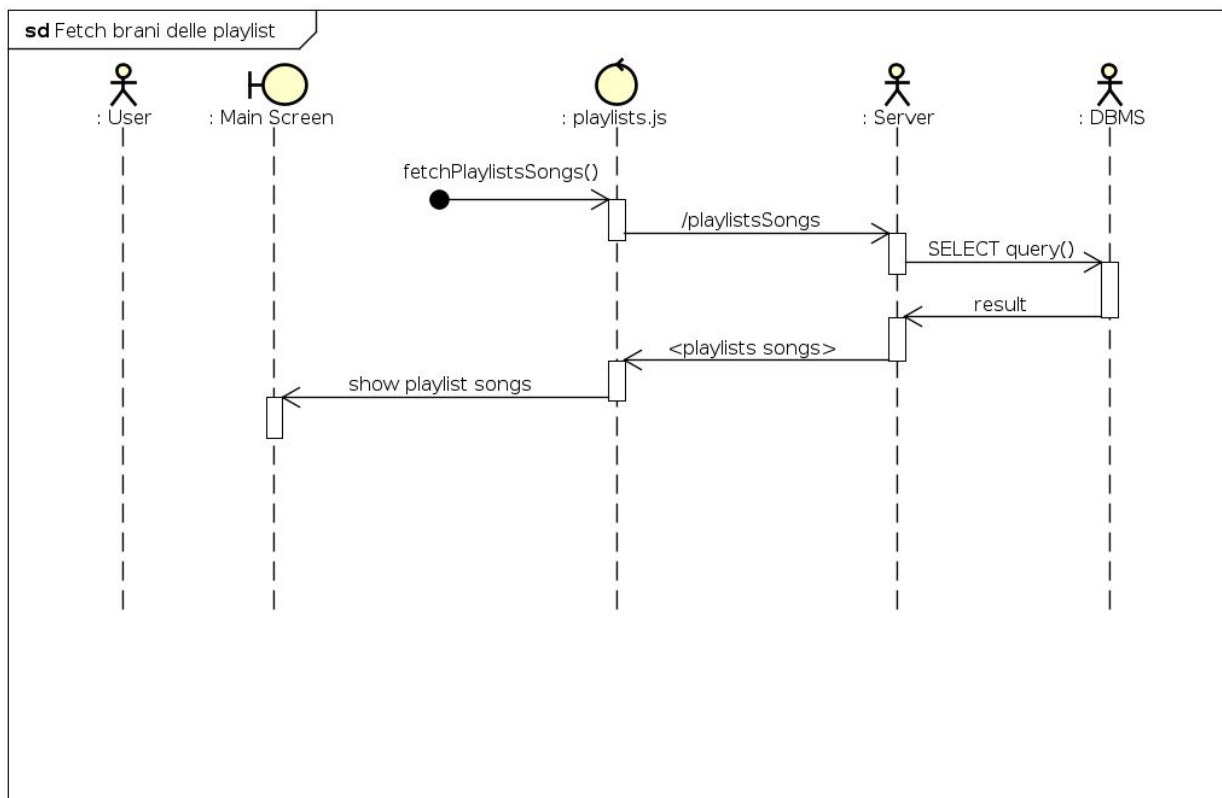
3.8 Fetch playlist utente

Attori	Utente, Server, DBMS
Precondizioni	- L'utente ha completato il caso d'uso "Login"
Sequenza degli eventi	<ol style="list-style-type: none"> 1. Il client richiede al server la lista delle playlist ad egli associate. 2. Il server identifica l'utente in base al cookie, incluso nella richiesta, ed effettua una query al DBMS per richiedere le informazioni sulle playlist associate a quell'utente. 3. Il server invia sotto forma di JSON la lista delle informazioni delle playlist associate all'utente.
Postcondizioni	<ul style="list-style-type: none"> - Il sistema sta riproducendo il brano richiesto. - Il sistema sta mostrando la "Main Screen" e la lista delle playlist associate all'utente.



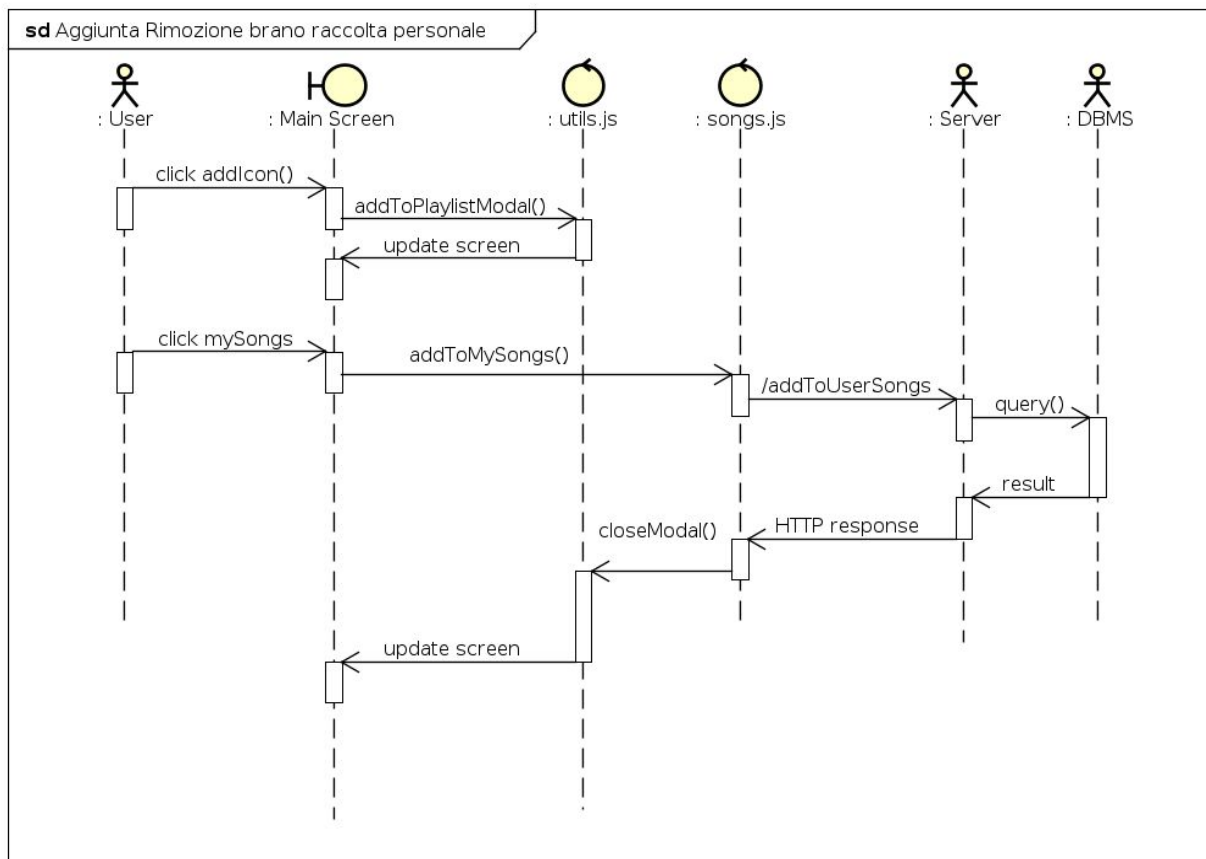
3.9 Fetch brani delle playlist

Attori	Utente, Server, DBMS
Precondizioni	- L'utente ha completato il caso d'uso "Login"
Sequenza degli eventi	<ol style="list-style-type: none"> 1. Il client effettua una richiesta al server includendo gli id delle playlist per le quali si intende richiedere i brani contenuti al loro interno. 2. Il server effettua una query al DBMS per richiedere le informazioni sui brani associati ad ognuna delle playlist identificate dagli id. 3. Il server invia sotto forma di JSON la lista delle informazioni dei brani associati ad ognuna delle playlist utente.
Postcondizioni	<ul style="list-style-type: none"> - Il sistema sta riproducendo il brano richiesto. - Il sistema sta mostrando la "Main Screen" e la lista delle playlist associate all'utente, con i relativi brani.



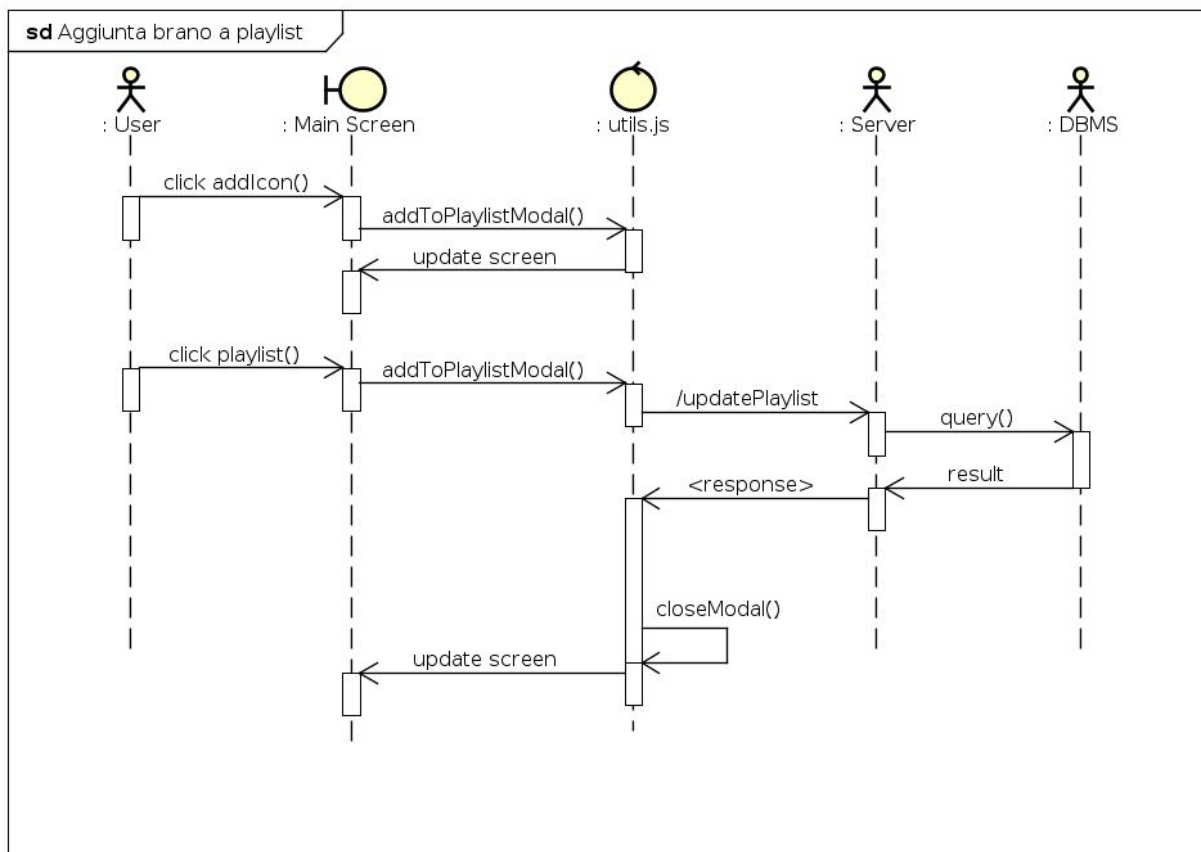
3.10 Aggiunta/Rimozione brano in raccolta personale

Attori	Utente, Server, DBMS
Precondizioni	<ul style="list-style-type: none"> - L'utente ha completato il caso d'uso "Login" - Il sistema sta mostrando la "Main Screen"
Sequenza degli eventi	<ol style="list-style-type: none"> 1. L'utente si posiziona sull'immagine di copertina del brano che intende aggiungere o rimuovere dalla raccolta personale. 2. Il sistema sostituisce l'immagine con un'icona a forma di "più". 3. L'utente clicca sull'icona a forma di "più". 4. Il sistema mostra l'<i>AddToPlaylist Modal</i>. 5. L'utente clicca il primo elemento della lista. 6. Il sistema comunica al server l'inclusione o la rimozione del brano nella raccolta personale. 7. Il server si interfaccia con il DBMS per la registrazione delle modifiche alla raccolta personale. 8. Il server ritorna un messaggio il cui codice di stato descrive l'esito dell'operazione di modifica della raccolta personale nel database.
Postcondizioni	<ul style="list-style-type: none"> - Il sistema ha registrato la modifica della raccolta personale. - Il sistema sta mostrando la "Main Screen".



3.11 Aggiunta brano a playlist

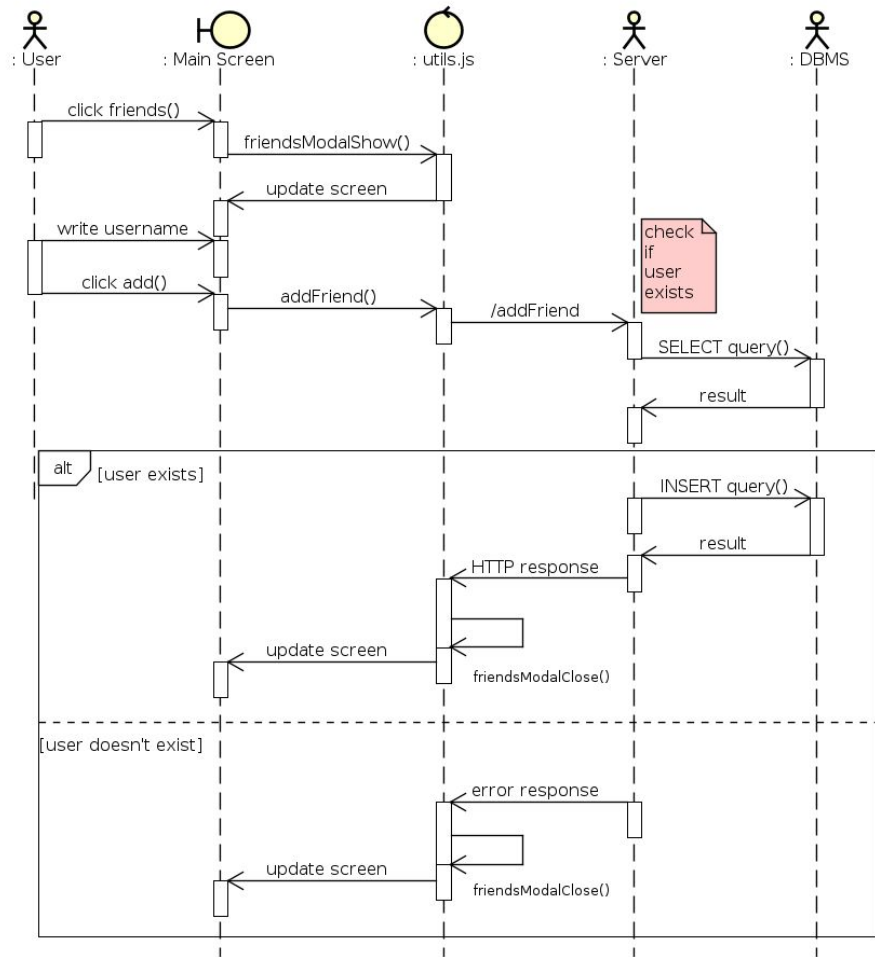
Attori	Utente, Server, DBMS
Precondizioni	<ul style="list-style-type: none">- L'utente ha completato il caso d'uso "Login"- Il sistema sta mostrando la "Main Screen"
Sequenza degli eventi	<ol style="list-style-type: none">1. L'utente si posiziona sull'immagine di copertina del brano che intende aggiungere ad una playlist.2. Il sistema sostituisce l'immagine con un'icona a forma di "più".3. L'utente clicca sull'icona a forma di "più".4. Il sistema mostra l'<i>AddToPlaylist Modal</i>.5. L'utente clicca sulla playlist a cui vuole aggiungere il brano.6. Il sistema verifica se la playlist non include già il brano.7. Se la playlist include già il brano:<ol style="list-style-type: none">a. Il sistema avvisa l'utente di ciò tramite un alert.b. Il sistema ritorna alla "Main Screen"8. Altrimenti:<ol style="list-style-type: none">a. Il sistema comunica al server la modifica della playlistb. Il server si interfaccia con il DBMS per registrare le modifiche sul database.c. Il server ritorna un messaggio il cui codice di stato descrive l'esito dell'operazione di modifica della playlist nel database.
Postcondizioni	<ul style="list-style-type: none">- Il sistema ha registrato la modifica della raccolta personale.- Il sistema sta mostrando la "Main Screen".



3.12 Aggiungi Amico

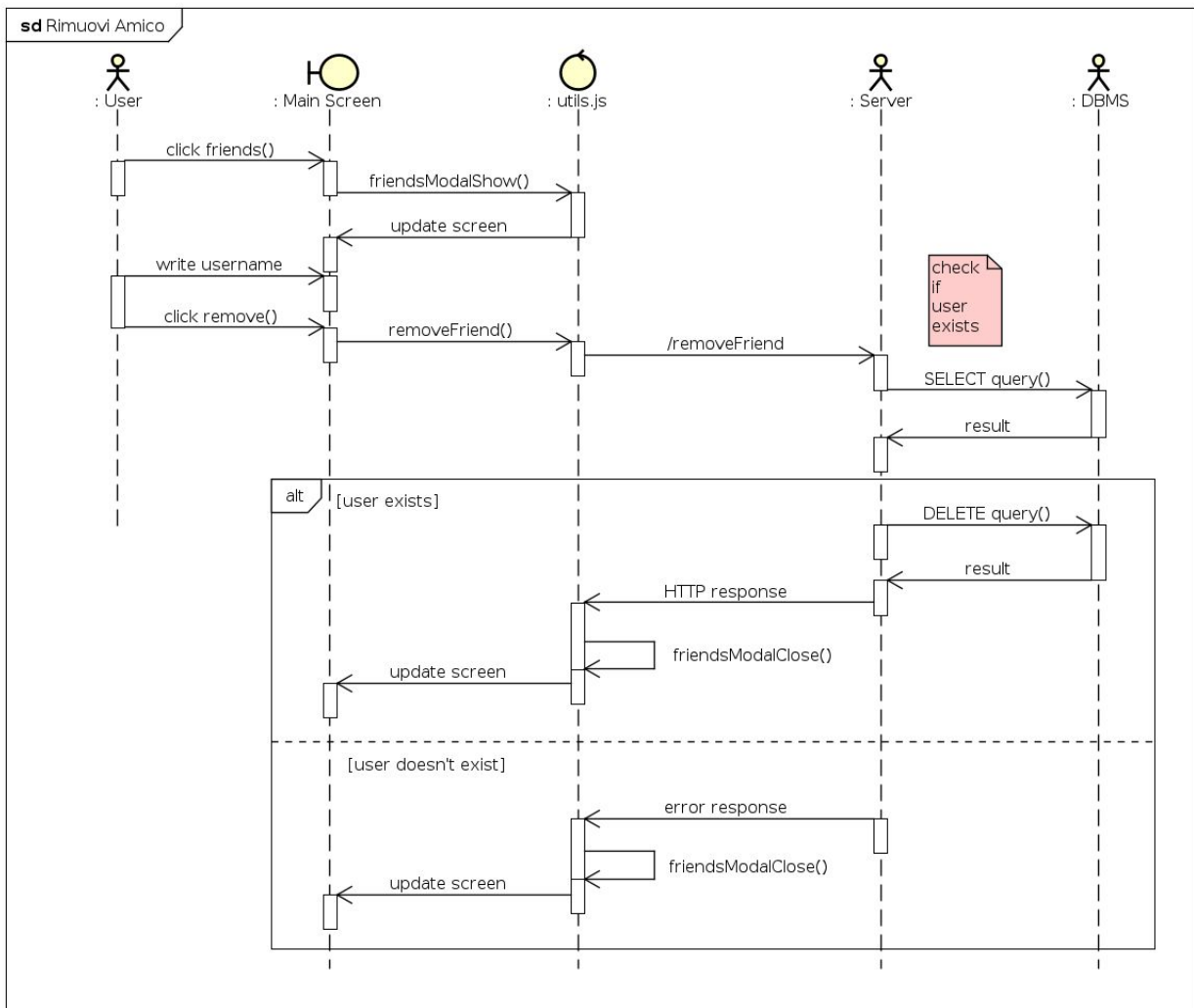
Attori	Utente, Server, DBMS
Precondizioni	<ul style="list-style-type: none"> - L'utente ha completato il caso d'uso "Login" - Il sistema sta mostrando la "Main Screen"
Sequenza degli eventi	<ol style="list-style-type: none"> 1. L'utente clicca su "FRIENDS" nella <i>NavBar</i>. 2. Il sistema mostra il <i>Friends Modal</i>. 3. L'utente inserisce nell'input field il nome dell'utente che vuole aggiungere come amico, quindi clicca <i>Add</i>. 4. Il sistema invia al server la richiesta di aggiunta. 5. Il server si interfaccia con il DBMS per l'accesso al database. 6. Se non esiste un utente con quel username: <ol style="list-style-type: none"> a. Il server invia un messaggio di errore. b. Il Sistema mostra un alert con il messaggio di errore 7. Altrimenti: <ol style="list-style-type: none"> a. Il server registra nel database l'aggiunta dell'utente come amico. b. Il server invia un messaggio di risposta al sistema.
Postcondizioni	<ul style="list-style-type: none"> - Il sistema ha registrato l'aggiunta come amico dell'utente.

sd Aggiungi Amico



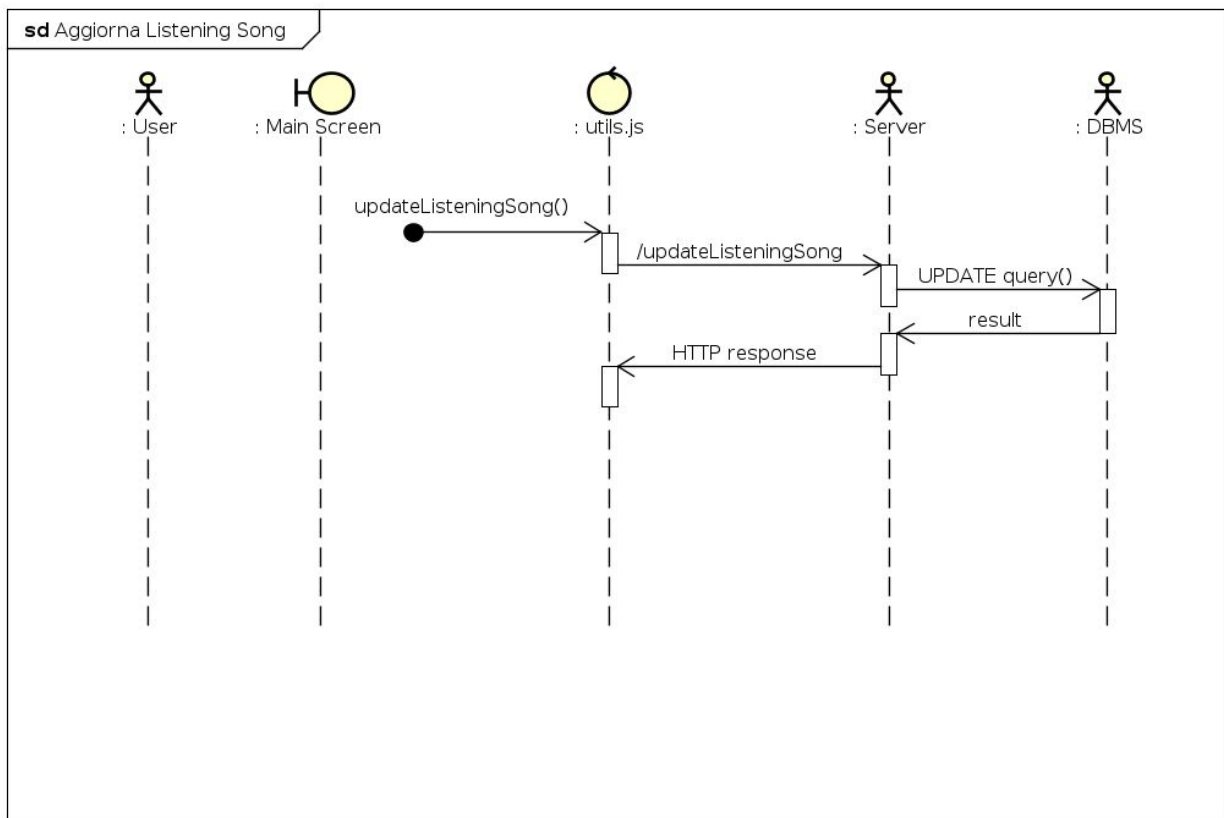
3.13 Rimuovi Amico

Attori	Utente, Server, DBMS
Precondizioni	<ul style="list-style-type: none">- L'utente ha completato il caso d'uso "Login"- Il sistema sta mostrando la "Main Screen"
Sequenza degli eventi	<ol style="list-style-type: none">1. L'utente clicca su "FRIENDS" nella <i>NavBar</i>.2. Il sistema mostra il <i>Friends Modal</i>.3. L'utente inserisce nell'input field il nome dell'utente che vuole rimuovere dagli amici, quindi clicca <i>Remove</i>.4. Il sistema invia al server la richiesta di rimozione.5. Il server si interfaccia con il DBMS per l'accesso al database.6. Se non esiste un utente con quel username:<ol style="list-style-type: none">a. Il server invia un messaggio di errore.b. Il Sistema mostra un alert con il messaggio di errore7. Altrimenti:<ol style="list-style-type: none">a. Il server registra nel database la rimozione dell'utente dagli amici.b. Il server invia un messaggio di risposta al sistema.
Postcondizioni	<ul style="list-style-type: none">- Il sistema ha registrato la rimozione dell'utente dagli amici.



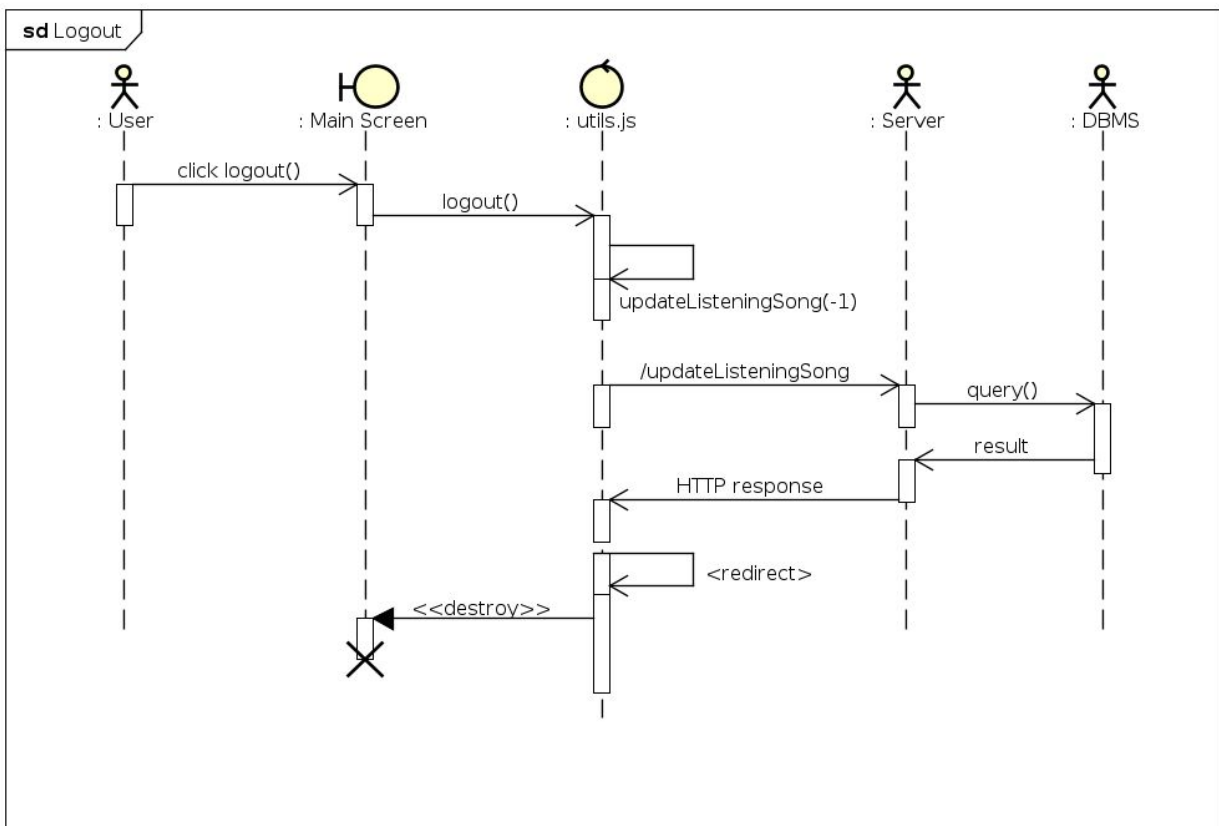
3.14 Aggiorna Listening Song

Attori	Utente, Server, DBMS
Precondizioni	<ul style="list-style-type: none"> - L'utente ha completato il caso d'uso "Login" - Il sistema ha iniziato o terminato di riprodurre un brano.
Sequenza degli eventi	<ol style="list-style-type: none"> 1. Il sistema invia una richiesta al server per la registrazione del nuovo brano che il sistema sta riproducendo (usato per sapere cosa sta ascoltando un utente in un determinato momento). 2. Il server si interfaccia con il DBMS per l'accesso al database. 3. Il server registra la modifica. 4. Il server invia al client un messaggio di risposta il cui codice di stato descrive l'esito dell'operazione di modifica della tabella del database.
Postcondizioni	<ul style="list-style-type: none"> - Il sistema ha registrato la nuova canzone che l'utente sta riproducendo.

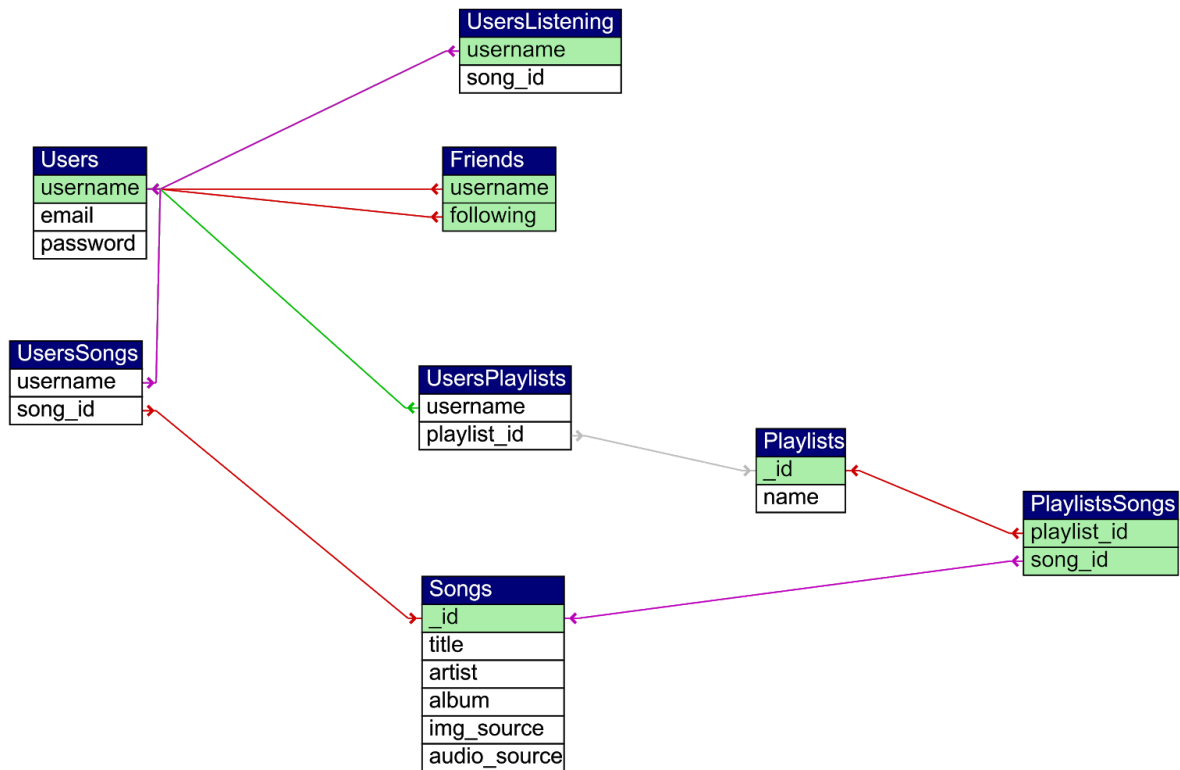


3.14 Logout

Attori	Utente, Server, DBMS
Precondizioni	<ul style="list-style-type: none"> - L'utente ha completato il caso d'uso "Login" - Il sistema sta mostrando la "Main screen".
Sequenza degli eventi	<ol style="list-style-type: none"> 1. L'utente clicca su logout nella <i>Navbar</i>. 2. Il sistema invia una richiesta al server impostare a -1 la listening song. 3. Il server si interfaccia col DBMS per la modificare il database. 4. Il server ritorna un messaggio la cui codice di stato descrive l'esito dell'operazione di modifica del database. 5. Il sistema resetta il cookie dell'utente e lo redireziona alla schermata di login.
Postcondizioni	<ul style="list-style-type: none"> - L'utente è scollegato dal sistema. - Il sistema sta mostrando la schermata di login.

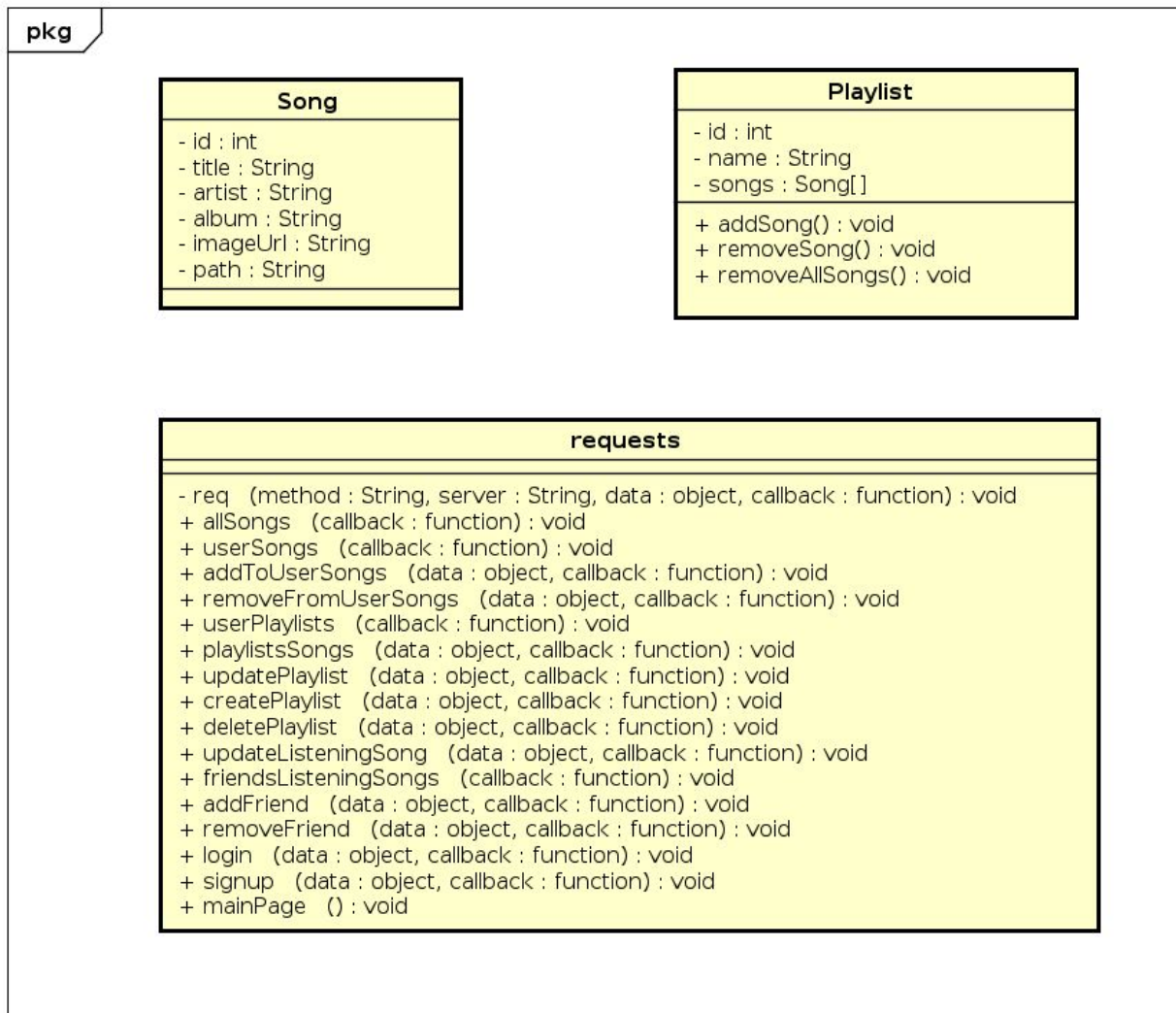


4. Struttura del database



Le tabelle il cui nome è composizione dei nomi di altre 2 tabelle (es. *UsersSongs*) schematizzano la relazione molti a molti che lega quelle due tabelle.

5. Diagramma degli oggetti



Gli oggetti **Song** e **Playlist** sono stati implementati in modo da rendere specifiche proprietà ed attributi non modificabili una volta creati ed inizializzati. Sono definiti all'interno del file `public/js/protos.js`

requests è invece un modulo che fornisce al lato client un'interfaccia attraverso la quale effettuare e gestire le richieste al server. L'implementazione del modulo è fatta nel file `public/js/requests.js`