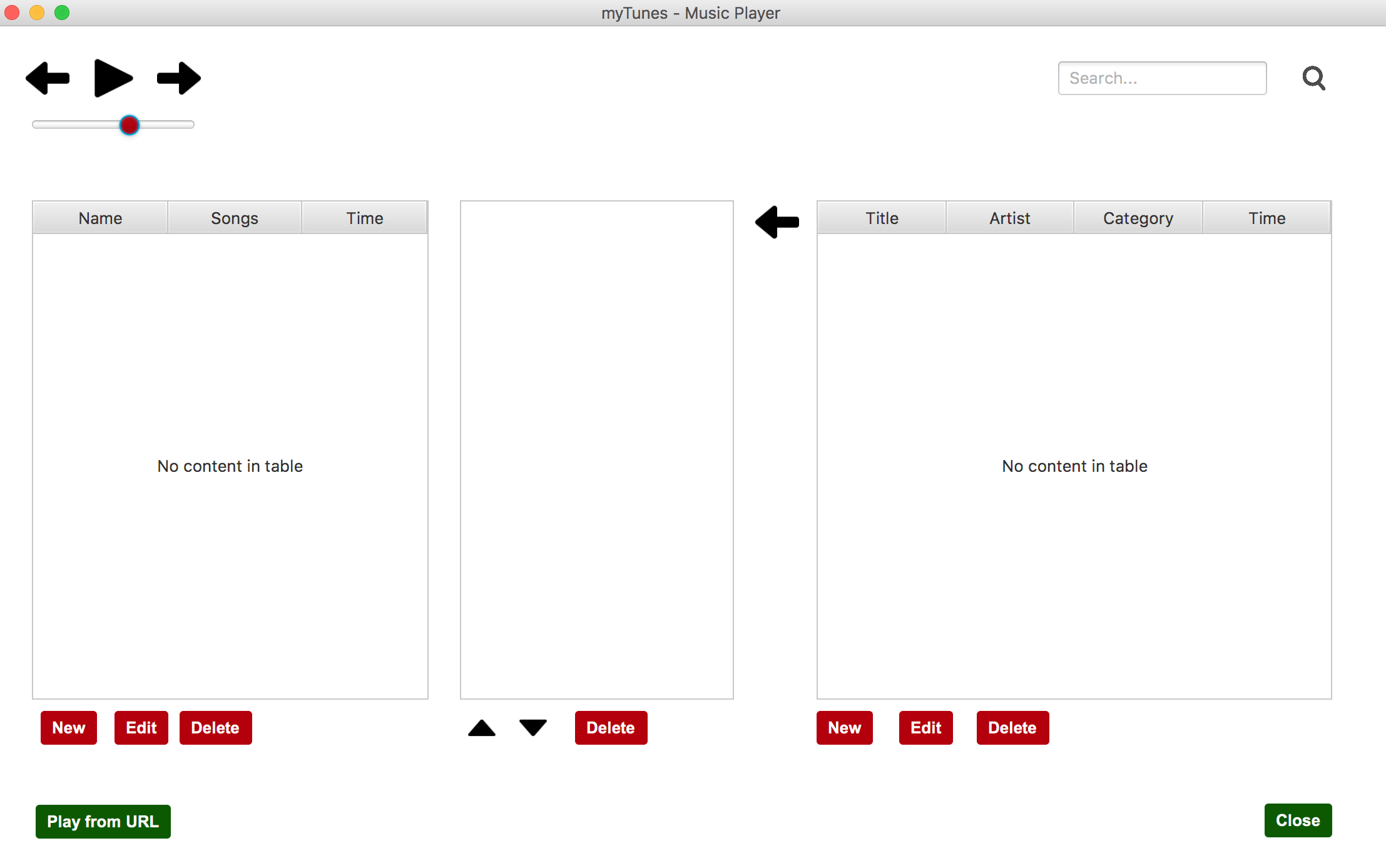
MyTunes - Compulsory Assignment #4



**Source code:** [**https://github.com/MapeSVK/myTunes**](https://github.com/MapeSVK/myTunes)

**Handed-in by**

1. **Dominik Nagy**
2. **Péter Sebők**
3. **Bence Mátyási**
4. **Michal Moravik**

**Date: 10.12.2017**

1. **State of delivery**

We were tasked with designing and constructing a java desktop application that could be used for playing music and storing its information to a database.

We needed to create an application that is backed up with a database that stores all the information about the imported music and the playlists. We had to make it so that the user can Add, Edit and Delete the imported music, their information, and the playlists. There are also options for playing and pausing music, adjusting the playback volume, filtering the music that we added, changing the order of playing in the playlists, adding and deleting music from the playlists and the option to open links from the internet.

The architecture of our application follows the three-layer model, so the GUI layer handles the interactions with the user, the Business Logic layer contains the business logic, the Data Access layer that controls the access to the database and the information stored in the music files and the Business Entities package that contains all our business entities.

1. **Application structure**
2. **Data Storage**

Our application stores its data to a database located on the school’s server, so it can only be accessed from the school network (as far as we know).

The database has three tables:

* The “Music” table for the music information, like the id, the name of the artist, the title of the song, its genre, the song length in seconds and the path of the file.
* The “Playlist” table for the playlist information, like its id and name.
* The “MusicInList” table that connects to previous two. It stores the songs which were put into playlists by storing the music’s id and the list’s id.

The ids in our tables are automatically generated and stored as an integer.

1. **Implementation details**
2. **Source Control**