**MusicDB**

**Project Proposal**

**Group 7**

**Members:**

* Duy Vu
* Nam Nhat Luu
* Jaejin Lee

**The Project’s Background:**

* We Group 7 love music.
  + We listen to online music from sources such as: Spotify, Pandora … and have a music library on each of those service.
  + We each also have a music library/database that we manage on local drive
* The problem arise when we have a song and need to find if we already have it in one of the library
  + In order to find that one song, we would need to search on each online services, then if it isn’t there, we need to find on the local drive.
  + The whole process is very inconvenient
  + Not to mention having multiple libraries of music.
* We want a service that can store our library’s info with maybe an addition of telling us where we can find the songs in the library.

**The Proposal:**

* We propose a musicDB (similar to IMDB but for music), which act as a global **music info** catalog.
  + A user of the service will have a library which is a subset of the global music info catalog
  + The users manage their library by adding songs from the global catalog to their library, remove songs from their library, creating playlists…
    - When adding the song to their library, the user can also add the source of which the song can be found. This source should only be visible by that user only.
  + The user can also comment, rate the song (this makes sharing or finding new song easier for the users)

**Project’s Description:**

* The project aims to create a central place for users to manage their music database online. We will refer to it as MusicDB library and others music libraries (Spotify, local ITunes) will be referred as local library. They can search for a song **across** **all** their online/offline music library
* The way to achieve this is:
  + Every time a user add/purchase a song and add it to their local libraries
  + The user will also have to manually go to our website (where there is “supposedly” a huge catalog of music info) search and add that song to their MusicDB.
  + The manually adding process is hard for those who don’t usually spending time managing their music library, and in fact they are not the kind of user that the service focuses on.
  + Later on, we can add a functionalities to synchronize the user’s library with others service APIs or importing local’s libraries by csv file format.
* The project will have a web GUI interface which allow user’s interactions.
* The service will allow users to create an account, login and add the songs to their personal account. The user can also comment, like, rate the song.
* The music database will have a library of songs’ info such as: song’s name, artist, album…
* The user database will have fields such as: name, password, comments, rating…

**Technical Aspects:**

* We will be using MySQL to build a relational database.
* The project consists of a web front-end, a server back-end and a database
* Front end will be developed using AngularJS, HTML, CSS, Jquery
* Back end will be developed using NodeJS
* The database will be MySQL with the use of nodejs-mysql module to connect it with the server
* We will be using Spotify API for the music the catalog. When a user decide to add the song to their library, or comment on it … we will then save that info in our database.
* There should be no machine restriction on the project.

***We Group 7 is a team of 3 members. As described above, the project need a web-server to work/achieve the intended purposes. Aside from designing a database for the project, we each members need to learn how to create a full-stack web-server. This can be challenging as we only has some experiences with front end development. Therefore we would like to process this project as a 3-member-team.***