|  |
| --- |
| **Introduction to Internet and Web** |
| **Project Report #2: Final Report** |

Enter the following for **ALL** team members!

|  |  |  |
| --- | --- | --- |
| **Name** | **Student ID** | **Department** |
| Yegizbayev Zholan | 202255634 | Computer Science and engineering |
| Yi Hyun | 201924551 | Computer Science and engineering |
|  |  |  |

|  |  |
| --- | --- |
| **Team Number** | (you can find your team number in the Project Announcement pdf file)  7 |

## <Note>

* Write you answer right below each question and submit this document on PLATO.
* Only one submission per group is required.
* You need to submit the followings
* Final project report (this document) written in English
* All project files (html, css, js, image/video, etc.) in a single zip file
* Recorded video for presentation and demo
  + The video should be 3~5 minutes long
  + You don’t need to record your face in the video, but you do need to explain your project in English.
  + Please use a PPT for presentation
  + Only the following formats are allowed: \*.avi and \*.mp4.
  + The video should include: 1) introduction to the project, 2) how to launch the web site (see [Q 2]), 3) explanation on the overall structure/layout of the web page, 4) use-cases/scenarios with demo.
  + Check the video before submitting to see if the recorded screen is clear enough to read the text therein, and if your voice is loud and clearly enough to understand.

## <Assignments>

|  |  |
| --- | --- |
| **[Q 1] Introduction**  Introduce your project. | |
| **Question** | **Your answer** |
| Introduce your project and web site in one paragraph. | Our project is a music player that integrated Spotify and YouTube services.  When a user searches for an artist or song on Spotify, it is designed to play the corresponding trac on YouTube based on those results.  This project leverages Spotify’s powerful recommendation algorithms and playlists to discover new music and plays the songs via YouTube. |

|  |  |
| --- | --- |
| **[Q 2] Launching the web site**  To launch the web site, what to do? Please write your answer in detail and in a step-by-step manner so that anyone can easily follow. Sample answers are:   * Simply double click “main.html” (or load “main.html” on a web browser), or * 1) launch a web server by “node server.js”, and then 2) launch a web browser and type “localhost:3000” | |
| **Question** | **Your answer** |
| Explain how to launch the web site | When you run index.html, you will see the screen in the picture attached below. First, you need to set up the Spotify API. After creating a Spotify account, enter the “Spotify Developer Dashbord” link and create an app, you can get the Client ID and Client Secret values. |
| Attach the screenshot of the front page (i.e., the very first web page of the web site) |  |

|  |  |
| --- | --- |
| **[Q 3] Project minimum requirement**  As announced, there are some project minimum requirements to satisfy. | |
| **Question** | **Your answer** |
| What are the three use-cases (or scenarios)? Explain in detail. | 1. When you search for an artist, you can see popular songs recommended by Spotify’s algorithms, as well as the artist’s albums, a list of other artists related to the artist, and recommended playlists. 2. You can save your own playlists based on your search results. The playlist can be saved a deleted in IndexedDB, and when you click the ‘My Playlist’ button, it shows the song stored in the database. 3. When you click on an album icon, it plays the corresponding songs as YouTube videos. |
| What is the name of the css file you have made and used in your project? | The UI of the music player webpage is created and used in the style.css |
| What is the name of the JS file you have made and used in your project? | We’re using jquery-3.7.1.min.js for Ajax calls, and the functionalities for each service are separated into spotify.js and youtube.js files.  We’re functionalizing the repetitive HTML generation code and storing it in util.js for use. |
| What is the Open API you have used in your project? | <Specific URL to the Open API introduction page>  <https://developer.spotify.com/documentation/web-api/reference/search>  https://docs.piped.video/docs/api-documentation/  <Which information/data can you get by using the Open API>  You can utilize Spotify’s search results and recommended playlist data by leveraging the Spotify API.  You can leverage the API of the piped.video service to get VideoIDs to play on YouTube.  <What is the format of the returned data>  JSON  <Attach the screenshot of the source code calling Open API request>  API code used to get the VideoID    Code that utilized the Spotify API |
| Attach the screenshot of the source code taking input from user | Since everything works through the getSearchInput function, so if the input value is empty, it will output an error message and stop working. |

Now, submit 1) this document, 2) all project files (in a single zip file), and 3) a video file for presentation on PLATO.

**THE END**