**SYNC**

**Description**

The web application is a collaborative music player in which users with a Spotify premium Spotify account can login, join a music room and listen to the same music other users in the room are listening to. The user would also have an option to contribute to the playlist.

There is a concept of a queue where a user in the room can add tracks and it would be queued. If many users like a track, they have an option to up-vote that track so that the track comes up faster. Users in the same room can also talk to each other via the group chat feature.

A user can also save a track he likes in his/her favorites list and this list can be retrieved from any user session and across any room.

**Deployment**

* Create an application on the Spotify Site and fetch the CLIENT\_ID and CLIENT\_SECRET.
* These 2 values need to be added to the auth.js file in the react-redux application.
* For the front-end react and redux app run an npm install command to install all the dependencies.
* Run the command npm run dev to start the react-redux application.
* Run the player application on another port e.g 3001 and the music can be transferred to this player.
* Start the chat server using the command node server/chatserver.js from the react-redux application.
* For the backend stack run the WebAPI application. It should run on port 5000.
* Hit the URL localhost:5000/api/system/init. This will create the default Tracks database with 4 dummy data of users.

**Features**

* Users with the same taste in music can listen to music in sync in real time.
* Users in the same room can communicate with each other with the group chat feature.
* Users can up-vote a song added to a queue based on their personal preferences.
* If no user has put a song in the queue, a bot is configured to play Spotify suggested songs in the room.
* There is an option to transfer the music to another device configured to that Spotify account.
* A user can save favorite tracks and can fetch them later in another session.

**Author**

* Meven Edwin DCunha