Final Project Write-up

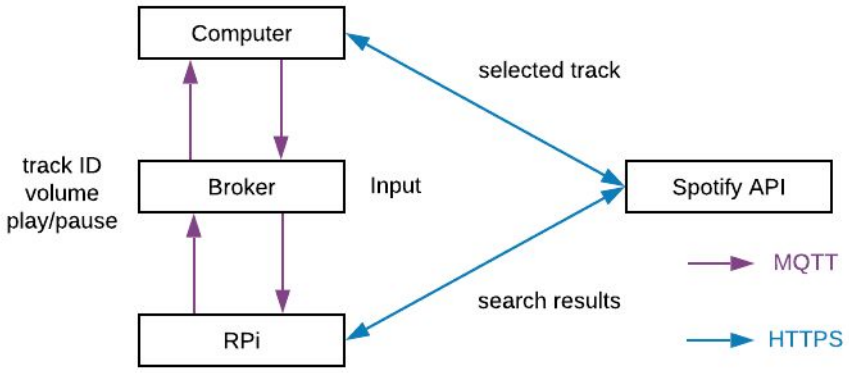
Team members

Jaylen Jackson, Haowen Liu

Project description

Our project uses the Spotify Web API to access a user's Spotify account and allow the user to search and play songs.

Using the terminal on your VM, the user can type keywords they want to search for, hit enter, and the RPi will display the results. Then the user can scroll through RPi results with the rotary encoder and press the button to select songs to play. Then the Spotify client on the computer will play the selected song. When a song is playing, the user can use the rotary encoder to adjust volume and short-press the button to play and pause. To go back to the main menu and start again, the user can long-press the button.



Implementation details

Components used:

- Rotary encoder
- Button
- LCD RGB display

Protocols used:

- HTTPS: Access Spotify API
- MQTT: Communication between RPi and computer

Processing techniques:

- Rotary encoder: polling
- Button: use polling to measure the duration of the button pressed, and use the duration to detect short press and long press

Design decision:

- We decided to have the rotary encoder only recognize multiples of 5 for volume. That is to RPi from generating too many MQTT messages, which increases latency.

Reflection

While writing spotify_api.py and spotify_int.py we encountered some issues accessing specific information about each artist or album. So, we did use pretty printing (json.dumps) to visualize the path that we wanted to use to access specific information like an artist or a track name. Furthermore, spotipy and the Spotify Web API have two options for authorization one of which requires that the user enter the client id and secret id at the time of running, as well as the verification of a redirect URI. That approach was not working for us as we encountered issues with caching. So we decided to rework the approach and focus on the documentations' other form of authorization, Client Credentials Flow, which relies less on the use of the redirect URI and allowed us to encode the username, client id, and secret id within the Spotify object. Nevertheless, this project was a great demonstration of my increased ability to create a project based on documentation, something that was not as simple to me at the beginning of the semester.