Dry Popcorn: Jason Lin, Jiajie Mai, Raymond Wu, Jerry Ye
SoftDev1 pd07
P01 -- TuneyForecast
2017-11-21

Tuney Forecast

Project Manager: Jerry Ye

Description

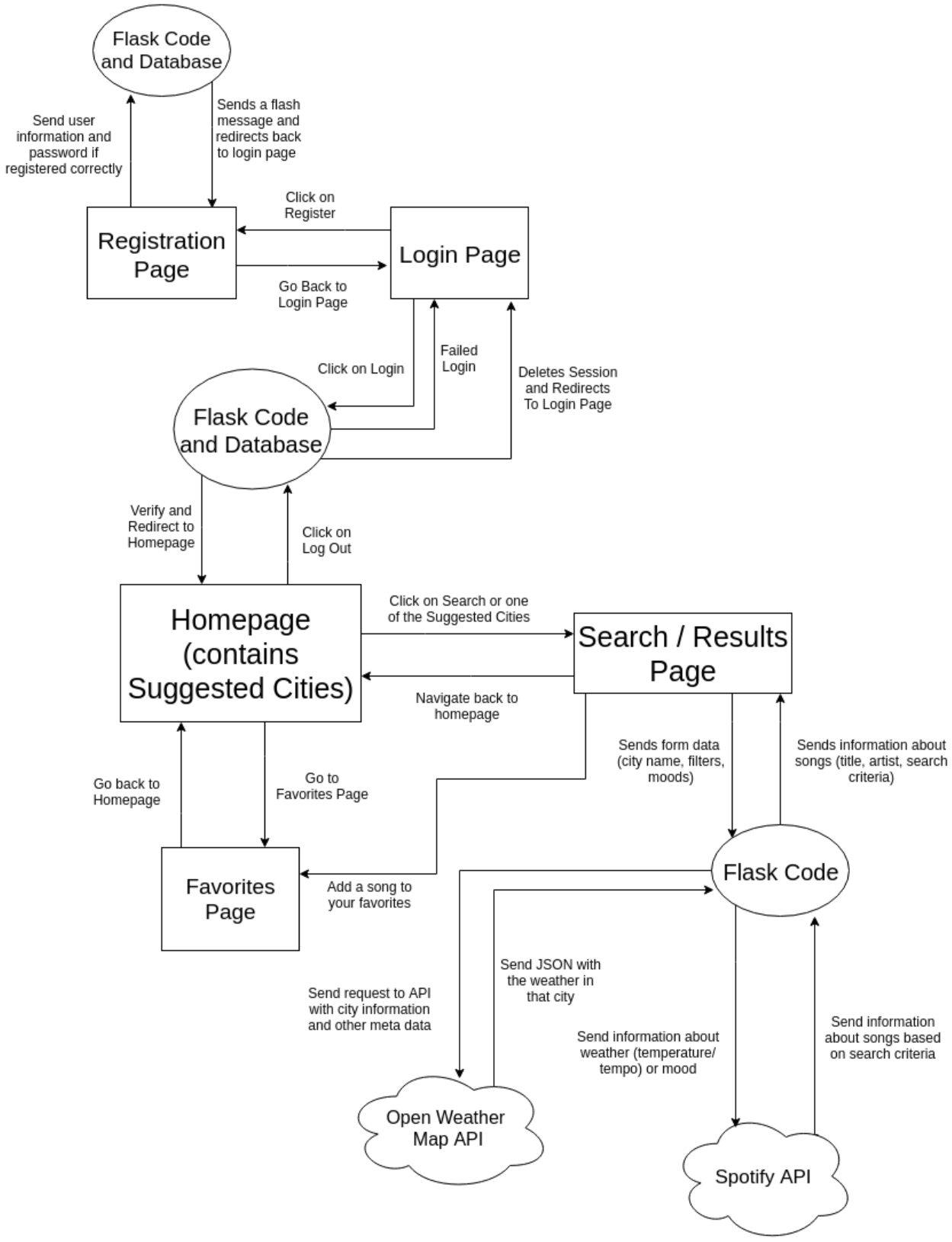
Our project will take advantage of the Spotify API and the Open Weather Map API to recommend for a user based on their current location. The User will be able to enter a location as a prompt, and our project will take that location and use the Open Weather Map API to generate the weather from that location at the current time. Our project would then call on a self-developed algorithm to take the current weather and recommend song choices based on that.

We decided to use Bootstrap because of its great grid system. It also has great documentation and great developer support. It seems to be more customizable and is more popular, leading to more possibilities.

List of Program Components:

- Front-end (HTML and CSS) using Bootstrap
    - Pages include:
        - Homepage
        - Register(additional feature)
        - Login page(additional feature)
        - Search page
        - Suggested Cities page(same as homepage)
        - Favorites page(additional feature)
        - Music suggestions page
        - Add moods page(additional feature)
- Back-end Flask code
    - Use of Spotify API and Open Weather Map API
    - Flask code to connect the use of both APIs
    - Login and user creation functionality(additional feature)

Component Map


Flask Code and Database
Send user information and password if registered correctly
Sends a flash message and redirects back to login page
Click on Register
Registration Page
Login Page
Go Back to Login Page
Click on Login
Failed Login
Deletes Session and Redirects To Login Page
Flask Code and Database
Verify and Redirect to Homepage
Click on Log Out
Homepage (contains Suggested Cities)
Click on Search or one of the Suggested Cities
Search / Results Page
Navigate back to homepage
Sends form data (city name, filters, moods)
Sends information about songs (title, artist, search criteria)
Go back to Homepage
Go to Favorites Page
Favorites Page
Add a song to your favorites
Flask Code
Send JSON with the weather in that city
Send request to API with city information and other meta data
Send information about weather (temperature/ tempo) or mood
Send information about songs based on search criteria
Open Weather Map API
Spotify API

Database Schema

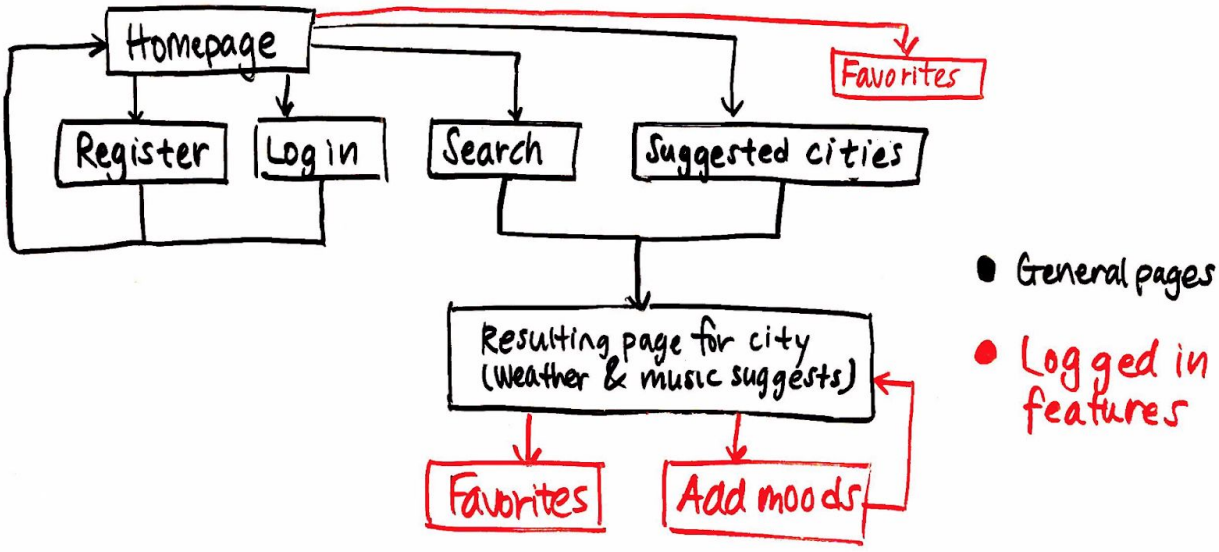
Users:

id	username	password
Integer	String	String

Favorites:

id	song_link	userId	song_name
Integer	String	Integer	String

Front End Site Map



Breakdown of Core Tasks:

1. Create the breakdown of tasks(Jerry)
2. Decide which APIs to use(Entire team)
3. Write up the project description(Jiajie and Jerry)
4. Write up a list of programs components (Jerry)
5. Design Component Map (Jason)
6. Design Front-end sitemap(Raymond)
7. Research and report on Spotify API(Raymond + Jerry)
8. Research and report on Open Weather Map API(Jason + Jiajie)

9. Write up HTML (Jiajie)
10. Write up CSS (Raymond + Jerry)
11. Develop an algorithm for generating the music using the weather(Entire group)
    a. Temperature Related to Tempo
    b. Mood-related to Mood of Song
    c. Filter songs based on certain categories (genres, length, artist)
12. Putting the entire backend together (Entire group)