Team krispyKernels -- Cheryl Qian, Simon Tsui, Derek Chan, Kendrick Liang
SoftDev1 pd8
P#01 -- ArRESTed Development
2018-11-17

Ticket Krisps

Program Components

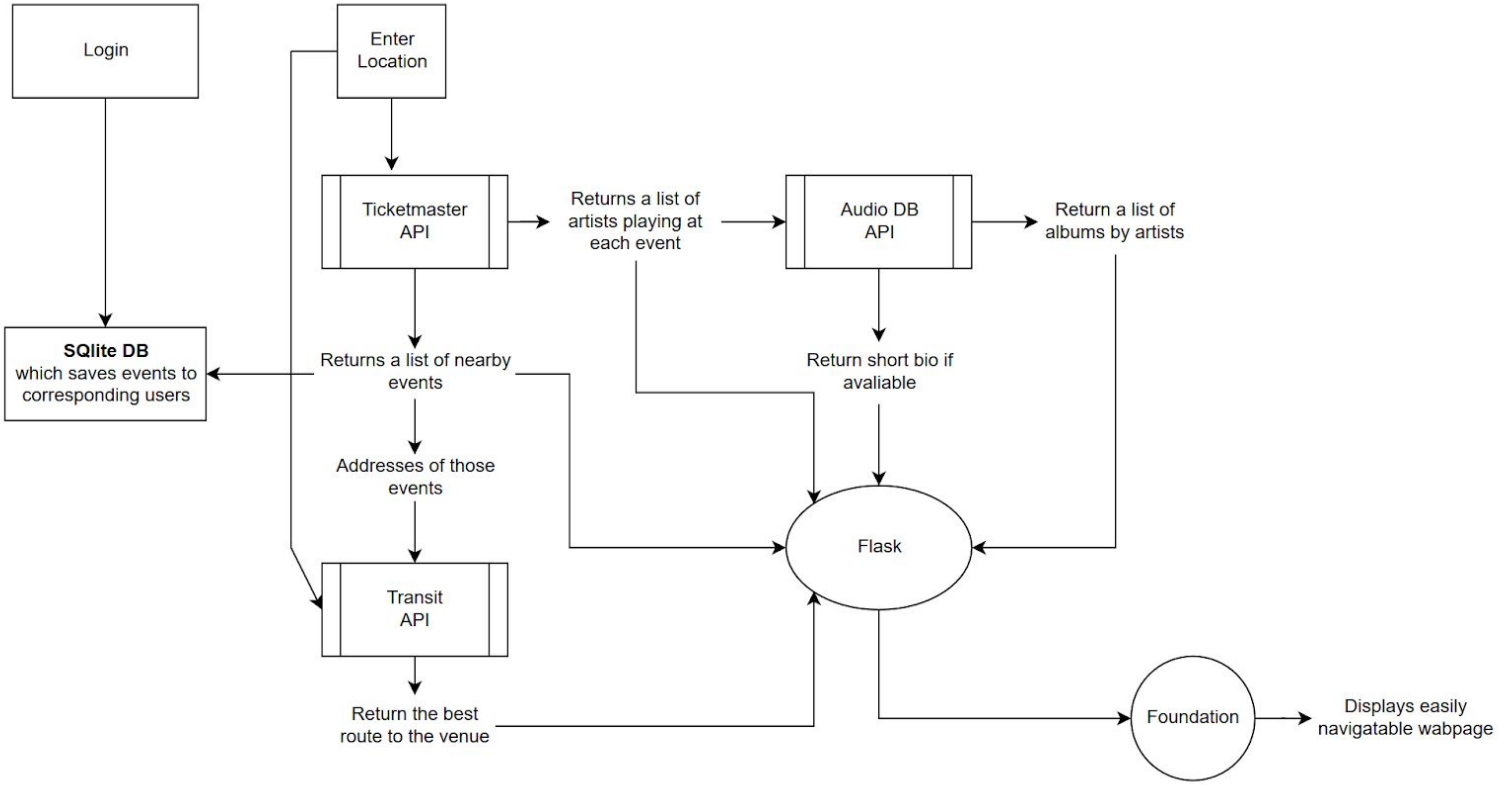
1. APIs
    a. Ticketmaster API is used to return a list of events in the user's location and accesses the list of artists at each nearby event.
    b. The Audio DB displays albums, songs, and bio of artists from the lineup of a selected event
    c. Transit/Map API generates the best route to the venue of a selected event
2. Foundation
    a. Creates our front end, and the visual layout of our site.
    b. Four main pages - login and registration, home, user settings, and saved items
3. Flask + Database
    a. Two databases, one to hold user registration info (username, pw) and one to hold saved events and saved locations

How the components relate to each other

The Ticketmaster API returns a list of nearby events based on one of user's (new or saved) locations. Each event entry in the json data includes the geo coordinates, artist lineup, date and time, and address of the event. Flask sends a query to the Audio DB for each artist in the lineup given by the Ticketmaster API and returns their albums, songs and a short bio if available. Another query is sent to the Public Transit API, which returns the best route to the venue based on the address/coordinates returned by Ticketmaster, and the user's entered location.

If a new user is registered, that user's login info will be added to the user directory table. If a user saves or removes an event, the event info will be added to their specific table in the database. Foundation displays the info our flask app gets from our database and APIs.

Component Map



Database Schema

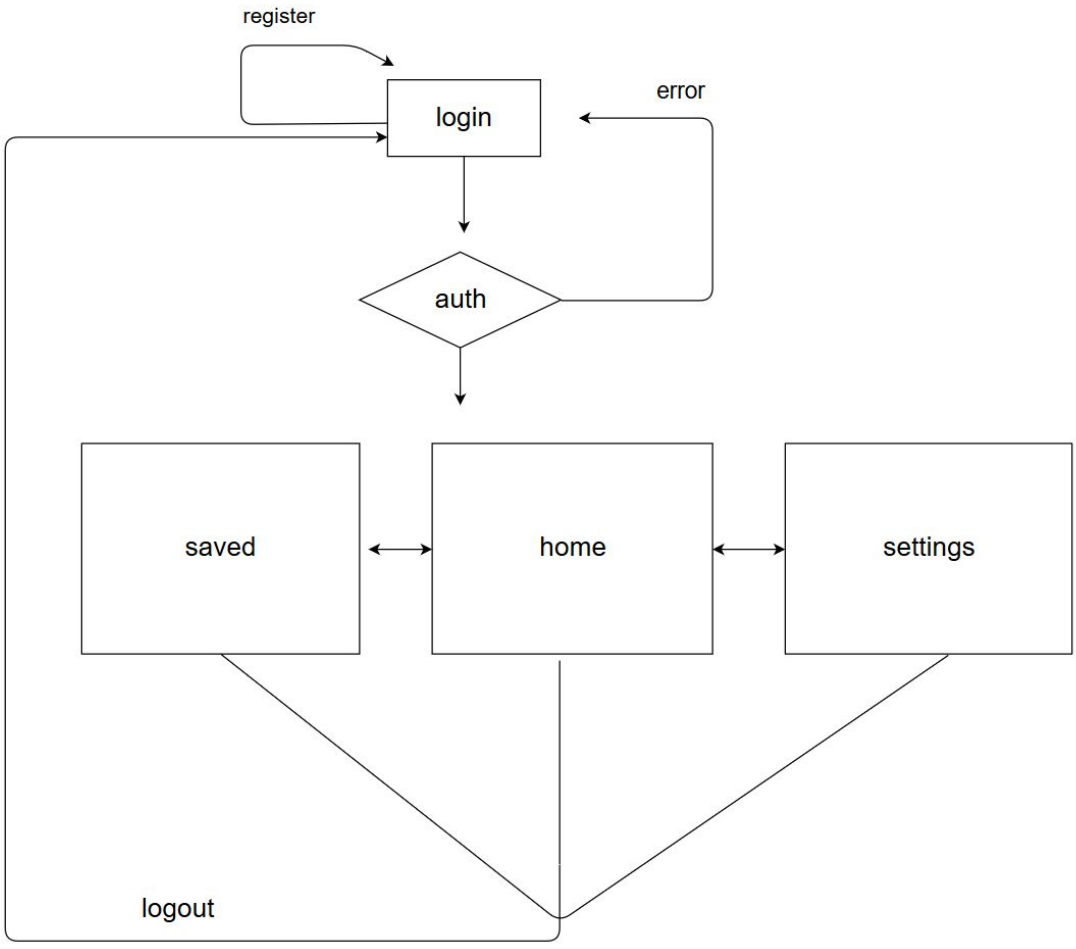
Account Info (Directory of users)

Username Stores usernames of users' accounts	**Password** Stores passwords of registered users

User specific info table (One table generated for each user)

Username Stores usernames of users' accounts	**Event** Stores name of event searched	**Location** Stores location of user	**Timestamp** Stores time that event was searched

Site Map



Breakdown of Tasks

1. Preparation
    a. Knowledge of how to access APIs will be required by all members. It is essential that we know how the website is sending requests to the API and the data received.
    b. A thorough understanding of the site and component maps linking the different pieces of our application. Certain tasks can be divided, others will need coordination in the linking of individual code blocks.
2. Placeholders
    a. Front-end template will be built. Lorem Ipsum will take the place of the future display of data. (**Cheryl Qian**)
    b. Back-end data, SQL databases will be operational with hard coded data. This tests the data retrieval and insertion methods, so they will be fully functional in short time. (**Simon Tsui**)
        i. Flask app linkage, test sites will be built so the routes are in place. (same person as task B)
    c. Back-end data, API calls will be placed on data of interest to the site's functionality. Selecting which parts of the API are to be featured in the website and the REST calls will be shared by both people (**Derek Chan & Kendrick Liang**)
3. Implementation
    a. On the website, Lorem Ipsum will be slowly replaced by Ticketmaster, The Audio DB, and transit map API. (**Kendrick Liang**)
    b. In flask, the app will call appropriate REST calls and get/pull from the database. (**Simon Tsui**)
    c. Both functionality and design will be worked on
4. Testing (**Derek Chan & Cheryl Qian**)
    a. Doing our best to break the website
    b. Adapting our website to withstand stupidity