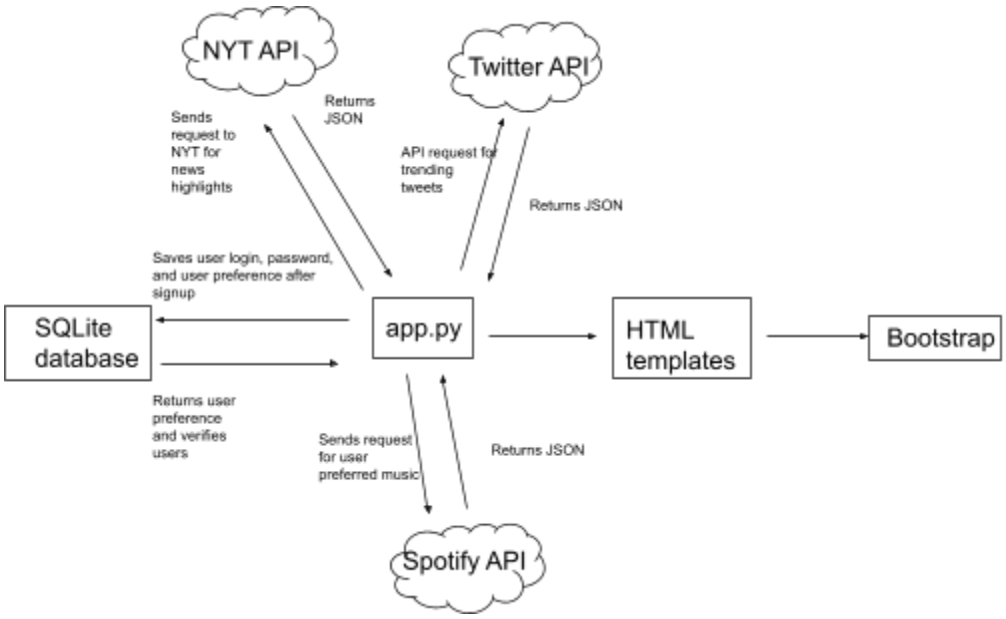
Team Fish: Madelyn Mao, Andrew Jiang, Benjamin Gallai
Softdev
P3: ArRESTed Development, JuSt in Time
2021-04-21

## Components

- README.md:
  - Clearly visible at top: **<Project Name> by <Team Name>**
  - Roster with Roles
  - Description
  - Launch Codes:
    - How to clone/install
    - How to run
- SQLite database:
  - Table 1: Users
    - Username (unique field)
    - Password
    - Preferences
  - Table 2: Saved songs
    - time saved to website
    - song name
    - artist name
    - album art
  - Table 3: Saved Tweets
    - time saved to website
    - twitter handle of tweeter
    - Tweet body
  - Table 4: Saved articles
    - time saved to website
    - article title, article itself)
- Flask App
  - Login uses cookies, so if user is already logged in, the user will not have to re-enter username password
  - connects backend files (i.e. database) to frontend files (i.e. html templates). This Python script will redirect users to another webpage based on the buttons they click and their input in HTML forms.
    - /signup, /login, /error will be same as P0
    - /home greets user, contains songs, tweets, and articles to browse + save
    - /mysongs is a page with songs the user has saved
    - /mytweets is a page with tweets the user has saved
    - /myarticles is a page with articles the user has saved
- Templates

    - signup.html: contains an HTML form that allows users to create a username and password for their account. **They will also enter their music preferences**. app.py will check if the usernames chosen by new users already exist in the Login table and verify if they fulfill other requirements (i.e. password length). Users will be redirected to response.html or error.html depending on whether their account was created successfully.
    - login.html: contains an HTML form that allows the user to enter username and password. When the user clicks the "Submit" button, app.py will check if the credentials match an entry of the Login table in the SQLite database. Users will be redirected to home.html or error.html depending on whether they logged in successfully.
    - error.html: If users fail to login or create an account successfully, they will be directed to this page and be told what went wrong. Below the error message, there is a "Sign Up" button that redirects the user to signup.html and a "Login" button that redirects the user to login.html
    - home.html: If users login or create an account successfully, they will be directed to this page. This page will have a navigation bar with options like saved songs, saved tweets, saved articles, as well as recommended songs,tweets,articles. There will also be a place to select music genres.
    - savedsongs.html displays [finish after tables are done]
    - savedtweets.html
    - savedarticles.html
- API
    - API for NYT
        - https://developer.nytimes.com/docs/most-popular-product/1/overview
    - API for Twitter
        - https://developer.twitter.com/en/docs/twitter-api
    - API for Spotify
        - https://developer.spotify.com/documentation/web-api/

## Component Map

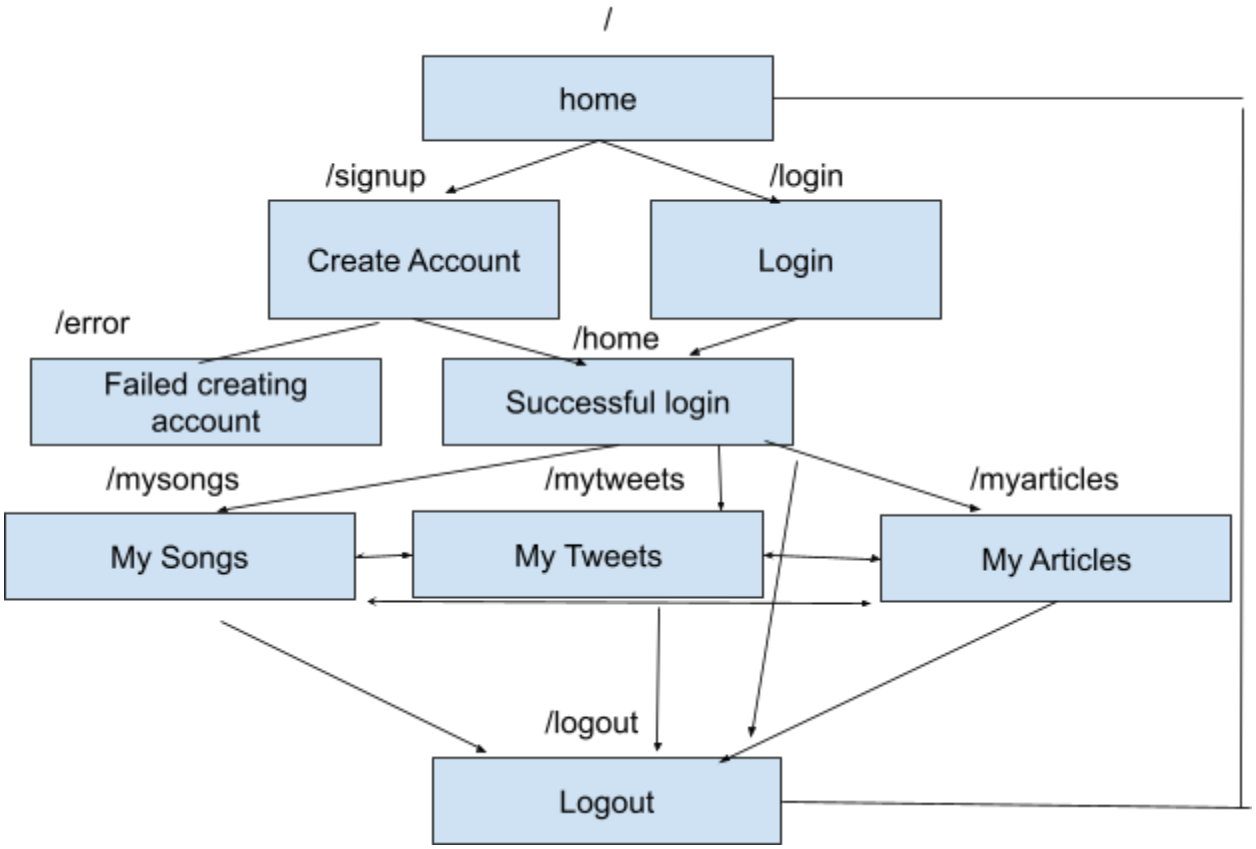



## Database Organization

- Users table:
    - There will be a row for each user.
    - Tweet IDs are are 64-bit integers represented as strings
    - Spotify IDs are base-62 song identifiers

| ID | Username | Password | Music preferences | Saved Tweets | Saved Music | Saved Articles |
|---|---|---|---|---|---|---|
| 0 | andrew | jiang | emo, rap, ... | 829...32, 430...87,... | a4T...8q, 9R3...mk, ... | https://www.nytimes.com/…, https://www.nytimes.com/…, ... |
| <non-negative integer> | <username> | <password> | <array of genres> | <array of Tweet ID> | <array of Spotify IDs> | <array of urls> |
| INTEGER | TEXT | TEXT | TEXT | TEXT | TEXT | TEXT |

## Site Map for frontend




**Tasks**

Andrew: spotify, deployment (bootstrap)
Benjamin: data base, twitter, nyt
Madelyn: flask app + html

**Timeline**

- April 23: finish development documentation
- April 25: basic html and login functionality
- April 26: Finish design documentation
- April 28: API requests, sqlite table
- April 30: css/js
- May 1: deployment