

Team Flop-Fish: Madelyn Mao, Andrew Jiang, Benjamin Gallai, William Li  
Softdev  
P5: This is the End  
2021-05-25

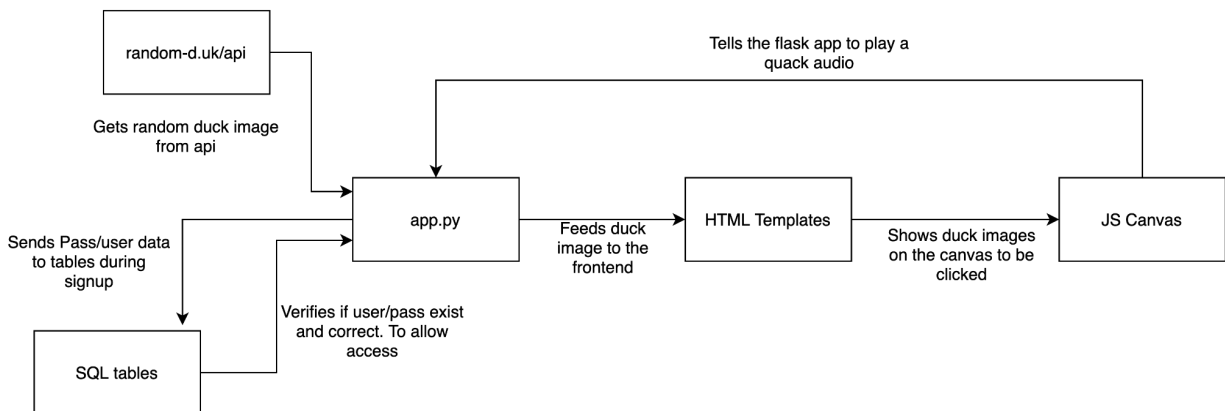
## Components

- 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
    - 15Duck Moves high score
    - Ducksu high score
- 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 game options and user high score
    - /ducksu
      - User clicks on ducks that appear on the canvas, if clicked on time, QUACK! and user's score increases. If user misses three times in a row, HONK!
    - /15duck
      - User clicks on buttons to rearrange numbers 1-15 on board.
- Templates
  - signup.html: contains an HTML form that allows users to create a username and password for their account.. 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.
- ducksu.html: displays canvas with appearing ducks, tracks user’s mouse clicks and displays user’s updated score
- 15duck.html: displays a table with randomly placed numbers 1-15. The user can click on any of 4 buttons to move the numbers.
- Bootstrap:
  - Neatly displays the two game options once the user logs in
- API
  - API for Spotify
    - <https://developer.spotify.com/documentation/web-api/>
  - API for ducks
    - Link: <https://random-d.uk/api>

## Component Map



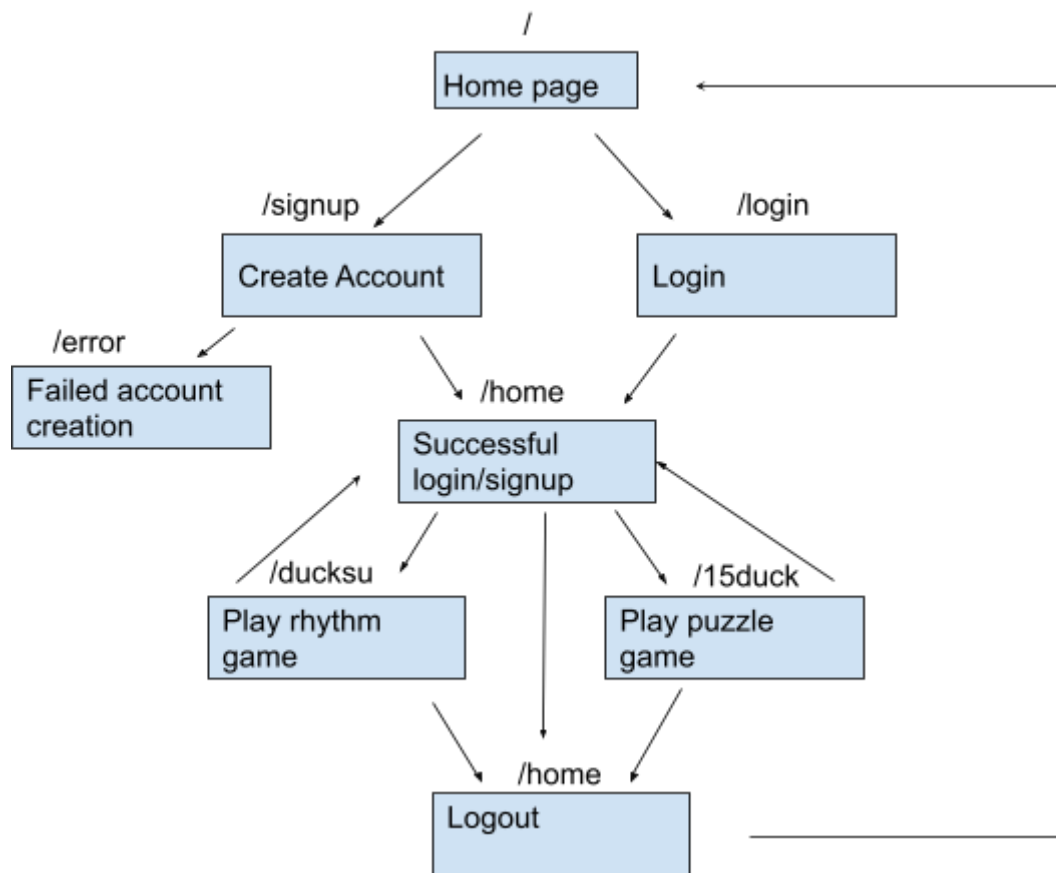
## Database Organization [NEEDS TO BE UPDATED]

- Users table:
  - There will be a row for each user.

ID	Username	Password	15Duck Moves	Ducksu Moves
----	----------	----------	--------------	--------------

0	andrew	jiang	20	20
<non-negative integer>	<username>	<password>	<non-negative integer>	<non-negative integer>
INTEGER	TEXT	TEXT	INTEGER	INTEGER

## Site Map for frontend



## Tasks

Andrew: duck API, spotify API

Benjamin: all the math behind ducksu and connecting canvas to script

Madelyn: Login, html table for 15DUCK + arrow buttons

William: JS, getting ducks to appear on canvas, user clicking

## Timeline

May 26: Fix up design doc

May 28: Get Login set up, tables set up

May 29: APIS, canvas working for ducksu // tables displaying randomly ordered numbers for 15duck

May 30: tracking user clicks for ducksu // loops to move numbers working for 15duck

June 1: connect quack sound for ducksu // record high score for 15duck

June 3: fix bugs in ducksu // better the animations for 15duck

May 31:

**Possible additions if time warrants:**

Instead of the numbers 1-15, user has to reconstruct a duck image.