Moyo Fagbuyi, Caden Khuu, Raymond Lin, Jonathan Metzler

RACAMOJO

p01-RoH: Meme Randomization Website

2024-12-02

Time Spent: 3 hours

TARGET SHIP DATE: 2024-12-17

Website Description

Our website will allow logged-in users to generate a completely random image, and use that image to create a meme. All users can view memes other users have generated, in the order they generated them.

Components

Templates Folder

- register.html
  - Allows users to register an unused username and password to be added to the database.
- login.html
  - Allows the user to input a username and password
  - 2 users cannot have the same username or else an error message will appear.
- logout.html
  - Rendered once the user is successfully logged out.
  - Allows users to return to the homepage.
- create_meme.html
  - Users can generate a random image (Random Image API) and use that image to caption a meme and create its description (Meme Maker API) .
- Homepage.html
  - Visible to all users.
  - Showcases all user-generated memes chronologically.
  - Upvote/downvote meme feature for users.
- error.html

- ○ Rendered when an error occurs (e.g. user tries to create an account with the same username as another user).

APP Folder
- ● Build_db.py
  - ○ Use sqlite3 to create a database of user information (username and password).
- ● __init__.py
  - ○ Initializes the app.
- ● (API) keys folder
  - ○ key_GoogleFonts.txt
  - ○ key_RandomImage.txt
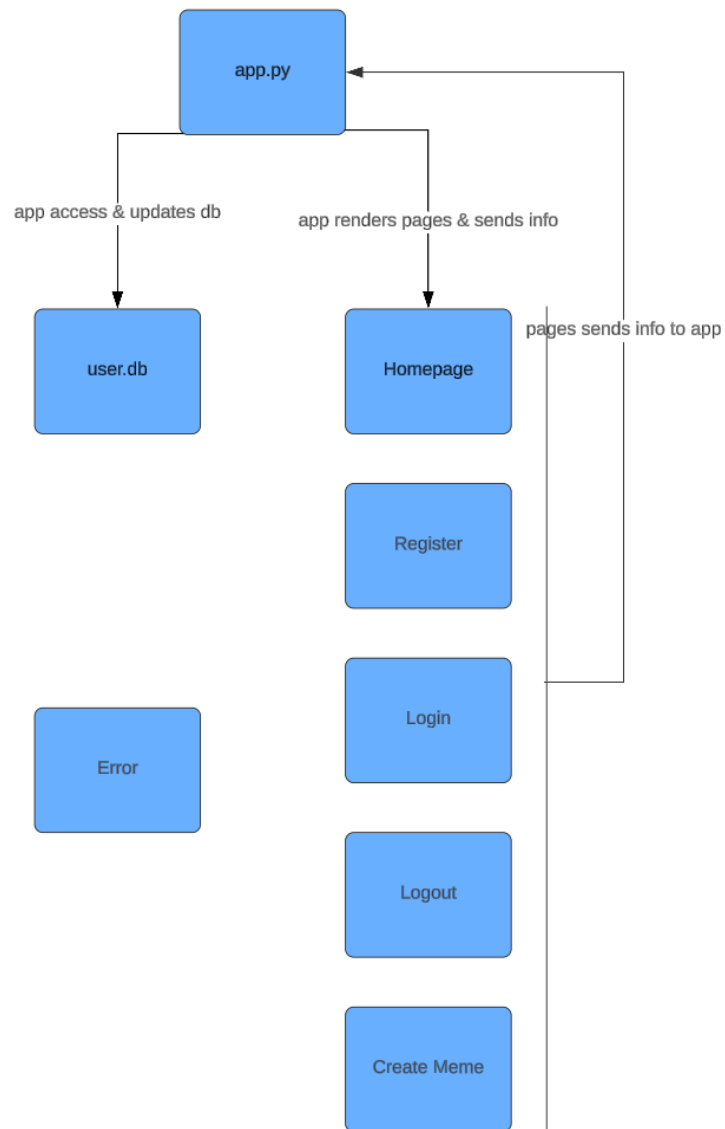 - ○ key_mememaker.txt

Application Logic
- ● __init__.py
  - ○ Flask Server, designating routes and connections between pages + their functionalities

<u>Front-end Framework</u>
- ● Tailwind
  - ○ We decided to use Tailwind because it's easy to understand and consistent. We plan to make use of its button/box customization and dark mode feature to make the user experience a bit more fun. If able, we'd like to use its image invert feature on the meme images.

Connection Between Components

Documentation and Assets

- design.pdf
    - This document. Devos should refer to and edit it often.
- flag.jpg
    - Our team's flag. Proudly portending greatness.

- README.md
  - Roles, description of website, how to, etc. Please read it.
- devlog.txt
  - Updated by each group member after they push to github and make changes. Maintained by PM. The most recent entry goes on the bottom detailing changes and bugs.

APIs used

- Random Image API
  - Provides the randomly generated image of a set pixel size.
- Meme Generator API
  - Allows users to post an image with a description, to its global database. Memes are organized by ID.
- Font API
  - Gives users access to many google fonts to create their memes with.

Database Organization

- user.db
- Tables:
  - user_information
    - Formatted data of usernames and passwords, each corresponding to an index by line
    - New items appended to the end
    - Cannot be deleted
  - user_private
    - Index in table name corresponding to index in the user table
    - Formatted data of the user's meme (title), each corresponding to an index by line
    - New memes appended to the end
    - Cannot be deleted
  - user_memes
    - Indexes in table name corresponding to index in the users' table and the subsequent index of the meme in user_ table
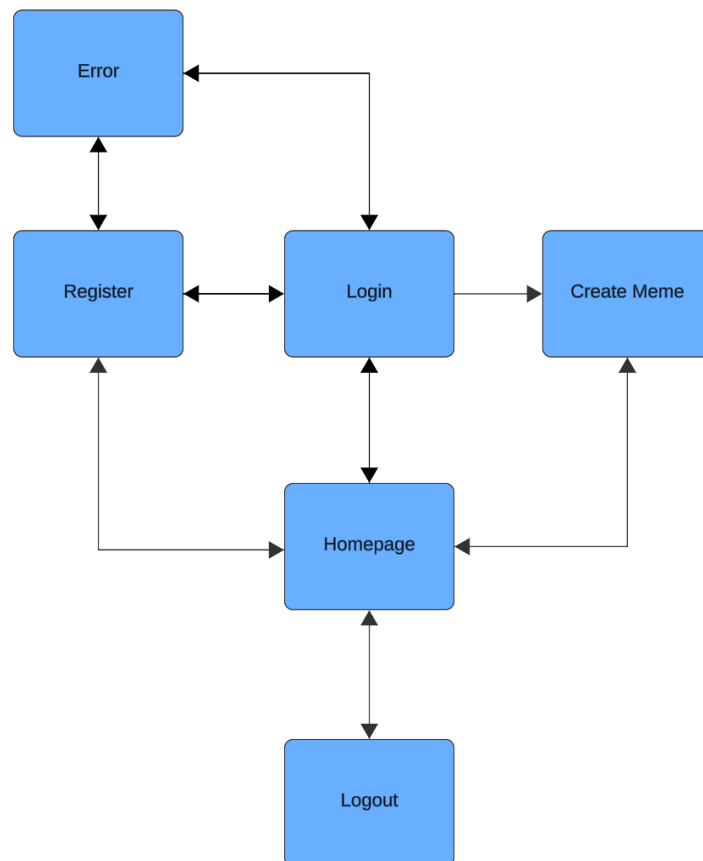
- Formatted data of the user's meme entries (title and text), each corresponding to an index by line
- New entries are appended to the end
- Cannot be deleted

Database Table

Meme #	Image	Caption	Size	Upvotes	Author
int	BLOB	string	int,int	integer	User class

Username	Password	Memes created
String	String	array[images]

<u>Sitemap</u>

Assignments

Task	Assigned To
Working on the frontend, helping with the database and Flask	Moyo (PM)
Database Engineering	Jonathan
Flask	Caden
Helping with Flask, database, and CSS	Raymond
Design Document + Devlog	All members