

Moyo Fagbuyi, Caden Khuu, Raymond Lin, Jonathan Metzler

RACAMOJO

p01-RoH: Meme Randomization Website

2024-12-02

Time Spent: 4 hours

TARGET SHIP DATE: 2024-12-17

Website Description

Our website will allow logged-in users to generate a completely random image and a completely random caption to combine for a meme (image with a caption). They can also save the captions that they like and they will be saved chronologically and displayed on the homepage.

Front-end Framework

- 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.

APIs

- Random Word API
 - Randomly generates different words as captions of an image.
- Picsum Photos API
 - Randomly generates images of a set pixel size.
- Font API
 - Gives users access to many google fonts to create their memes with.

Components

Templates

- 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 captions chronologically (captions that they choose to save).
 - ~~Upvote/downvote meme feature for users.~~
- ~~error.html~~
 - ~~It may be rendered when an error occurs (e.g. user tries to create an account with the same username as another user).~~

APP Folder

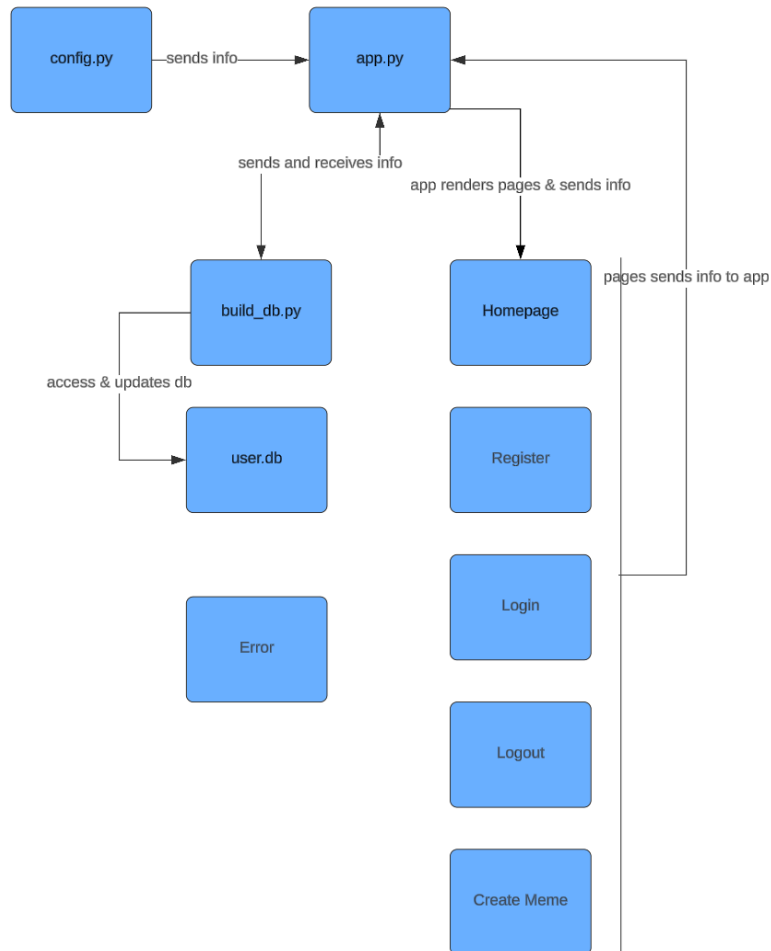
- build_db.py
 - Use sqlite3 to create a database of user information (username and password).
- __init__.py
 - Initializes the app.
- config.py
 - Checks that API keys exist and are stored in the right place.
- memes.csv
 - Stores Meme data (id, image link, creating user, etc)
- users.csv
 - Stores Users' username and password
- user.db
 - Stores the users and memes tables with respective information.
- keys folder
 - key_GoogleFonts.txt
 - readme

- Explains the state of API keys
- Templates Folder

Application Logic

- `__init__.py`
 - Flask Server, designating routes and connections between pages + their functionalities
- `config.py`
 - Securely loads API keys from their respective files. It makes sure the keys aren't hardcoded into the app.

Connection Between Components (no more create_meme page)



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.
- requirements.txt
 - Lists flask and pip installs needed to run application.

Database Organization

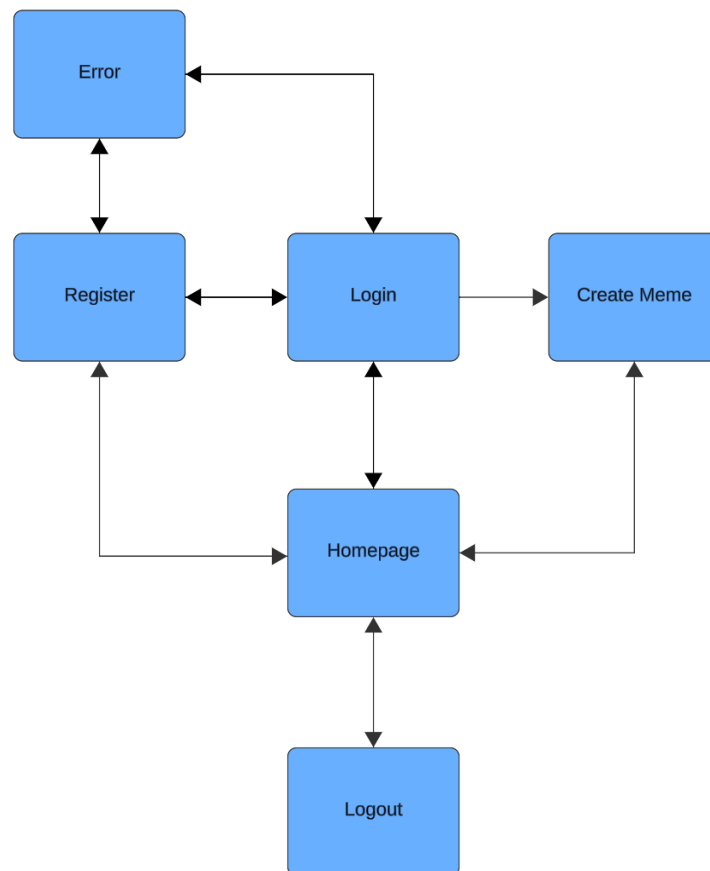
- user.db
- Tables:
 - users
 - Formatted data of usernames and passwords, each corresponding to an index by line
 - New items appended to the end
 - Cannot be deleted
 - memes
 - Indexes in table name corresponding to index in the users' table and the subsequent index of the meme in user_ table
 - Formatted 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]

Sitemap (got rid of create_meme page and everything stays on homepage)



Assignments

<u>Task</u>	<u>Assigned To</u>
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