P00: Half-Quick
Vedant, (Naf) Nafiyu, Ben, Chloe
Team Minerals
Pd. 5
TARGET SHIP DATE: 2024-11-27

Overview:

Our web app features a front end powered by HTML templates and CSS for layout and styling, while the back end utilizes a Flask app to manage web pages, user sessions, and database interactions for storing user data and blog articles. The site includes multiple pages such as the Main Page, Login Page, and Home Page, with categories for blog posts like Sports, Politics, Economics/Business, Lifestyle, and Arts. Logged-in users can post blogs, view random blog posts, and navigate through categories, with seamless session management and login/logout functionality integrated throughout the site.

Program Components:

Front end:
- HTML templates: provide layout for web pages, display blog articles from databases
- CSS files: provide style for HTML templates

Back end:
- Flask app: host web pages, manage user sessions
- Database module: store user information and blog articles

Component Relationships:

Flask ↔ Python Middleware
- Flask routes will call Python functions to handle various operations such as retrieving or modifying data.
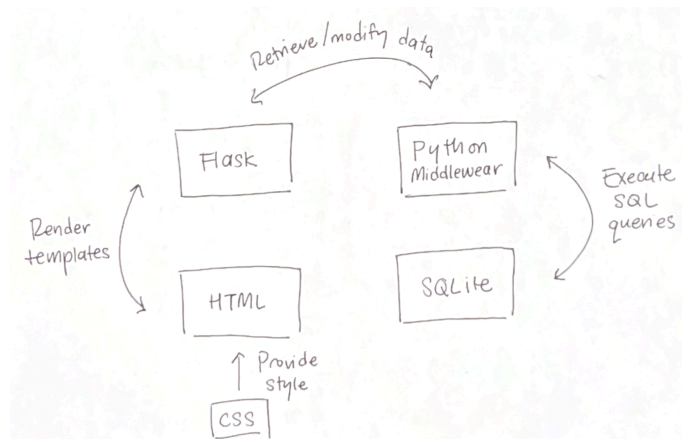
Python Middleware ↔ SQLite
- Python middleware interacts with SQLite to execute SQL queries for CRUD operations and passes results back to Flask for rendering.

HTML/CSS ↔ Flask
- Flask will render HTML templates dynamically based on the data passed by Python middleware, allowing the user to interact with the web interface.
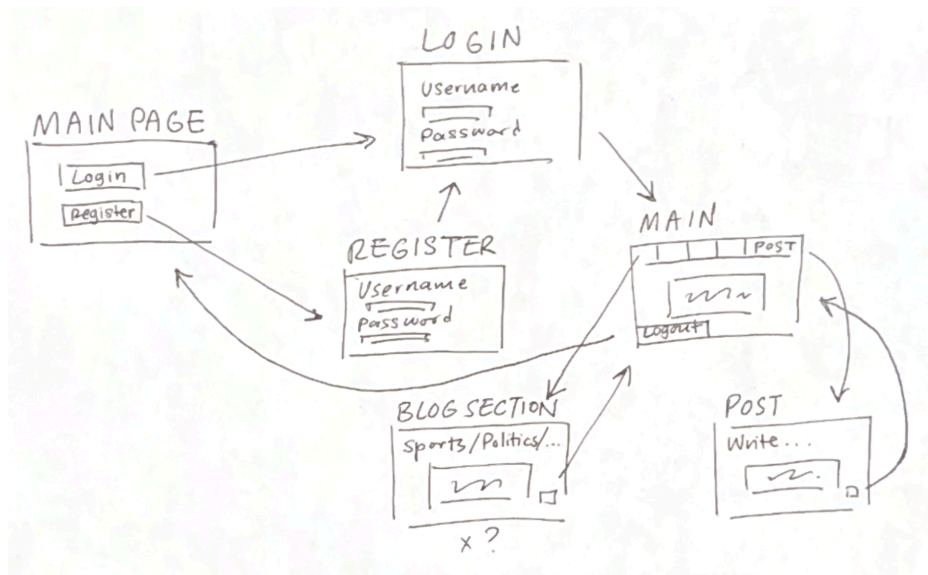
Component Map:

Blog Tabs:
- Main Page:
  - Introductory page that gives a brief description of the site. A LOGIN and REGISTER button will be presented. REGISTER will lead to the Registration Page, whilst LOGIN will lead to the Login Page.
  - If the user is logged in: Redirects to Home Page.
- Login Page:
  - Follows similar styling as the main page. Contains inputs for LOGIN information and a LOGIN button. Incorrect credentials reloads the page. Correct credentials lead to the Home Page. A Register button is also present. This button leads to the Registration Page.
- Registration Page:
  - Similar styling as the main page. Contains inputs for REGISTRATION information and a REGISTER button. Invalid/Taken inputs such as usernames pop up an error. Valid inputs register a user and the REGISTER button will direct to the Login Page.
- Home Page:
  - If the user is not logged in: Redirects to Main Page
  - Top middle of the page has a text input box with a 'POST BLOG' button. Essentially like Facebook's.
  - Below that, a random blog post will be presented in a somewhat big view.
- Sports:
  - Tab with blog posts only relating to sports current events
- Politics:
  - Tab with blog posts only relating to politics current events
- Economics/Business:
  - Tab with blog posts only relating to economics/business current events
- Lifestyle:
  - Tab with blog posts only relating to lifestyle current events
- Arts:
  - Tab with blog posts only relating to arts current events

Every page that a user is logged in for will have a LOGOUT button at the top navbar. This will lead to the Main Page. All pages listed under Home Page will include categories of the other pages below it in the navbar. Clicking each page directs to its respective page. All pages have the logo (home button), clicking it leads to the Home Page if logged in, Main page if logged out.

Database Organization:
- Table for users: id, username, password
- Table for categories: id, category
- Table for blog posts: id, author id, category id, article
- Blog posts can be traced back to their authors and categories via their author and category ids

Site Map:

Task Designations:
Naf: Flask app
Ben: Database module
Vedant: HTML Templates
Chloe: CSS