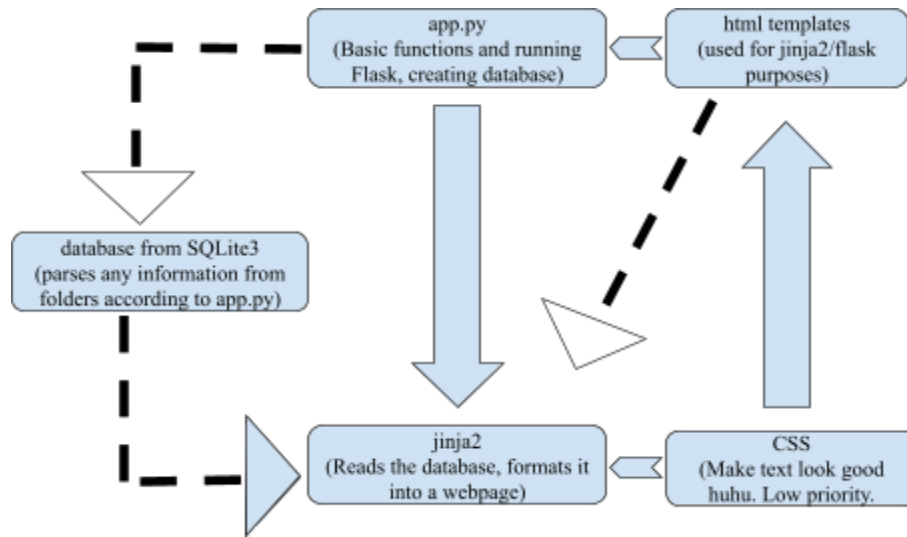


Eviss Wu, Thamidur Rahman, Ricky Lin  
SoftDev  
P00:Half-Quick  
Oct 22nd 2025  
TARGET SHIP DATE: {2025-12-01}

## Program Components:

- Frontend
  - All static HTML files that aren't created
  - Register Page
  - Login Page
  - Profile Page
  - Home Page
  - Each individual page will be an appended item on the database, then created by Flask
- Database (Through SQLite3)
  - Users (Username, Bio, Password, Pages changed)
  - Pages (Link, Name, Content)
- Flask Functionality (Through Jinja2)
  - Creating a new page
  - Viewing a page
  - Editing a page (Must check whether someone is a user)
  - Logging in/out
  - See a profile
  - Create an account
  - Save "cookies" with a private Hash
- Templates (HTML)
  - Template Template (Header, Navigation Bar, Body, etc...)
  - Homepage to see all pages
  - Template for a single page
  - Register Template
  - Login Template
  - Profile Template
  - Editing Template? (Can be added to viewing functionality)
- build\_db.py that will create the database through Flask
- Utility functions in Python (app.py) to use Jinja2 and create a Database
- CSS (Do last, not as important)

## Map:



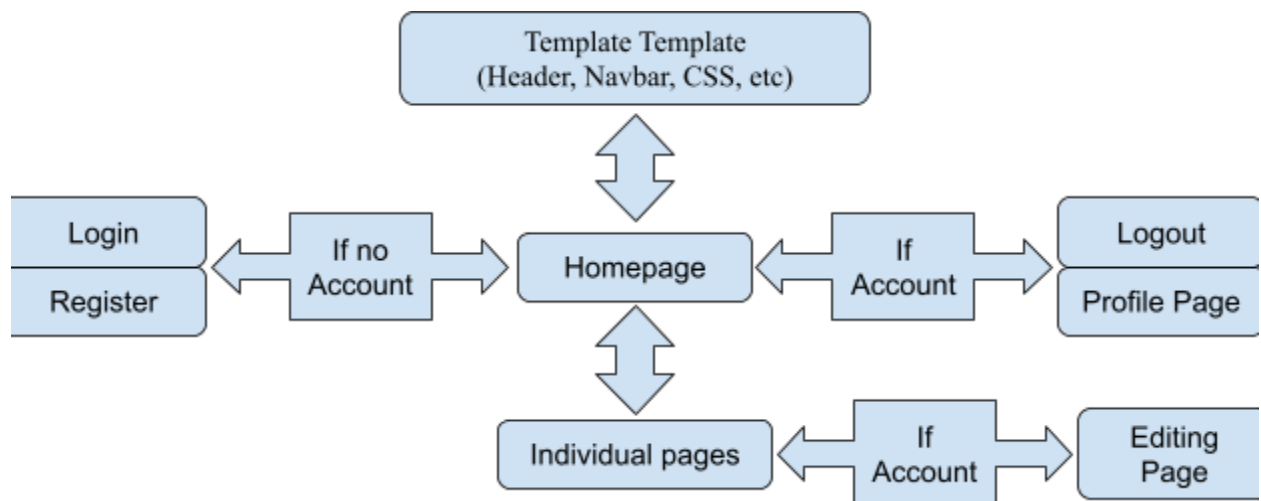
## Database Organization:

We plan to use tables to store the data for each corresponding page and user. As for our navigation, we will most likely have a navigation bar that looks like the following.

Users		
String	username	No duplicates, unique value
String	password	Password of user (hidden)
HashSet	Identifier (Cookie)	Cookie of user(hidden)
String	edited_pages	Name of edited pages with timestamp
String	bio	Information of the user

Wiki Page		
String	title	Name of the page
Link	link	Name of directory from the root
String	info	Database info to put in template
String[]	edit_history	List of usernames

## Web Map:



- Template Template (Header, Navigation Bar, Body, etc...)
- Homepage to see all pages
- Single page
- Register
- Login
- Logout
- Profile Page

- Home page
- Editing? (Can be added to viewing functionality)

## Navigation Bar:

Team Flag	Edit this page	CoalMiners Wiki	Logout/Login	Profile Page/Register
Sends user to home page	Only available for some pages, and for registered users	Not interactable, but a title	Leads to corresponding pages	Leads to corresponding pages

## Steps:

- 1) Creation of HTML literals (All static templates, navbar) - Eviss
- 2) Database creation (Handling how database is stored within [app.py](#), and how it is logged) - Thamidur
- 3) Flask Functionality (Handling opening and pathing of websites, as well as editing) - Ricky
- 4) Utility Functions (To be determined what we need) - Ricky & Thamidur
- 5) CSS (Visual changes to each page)- Eviss
- 6) Bug Fixes - Eviss & Ricky & Thamidur