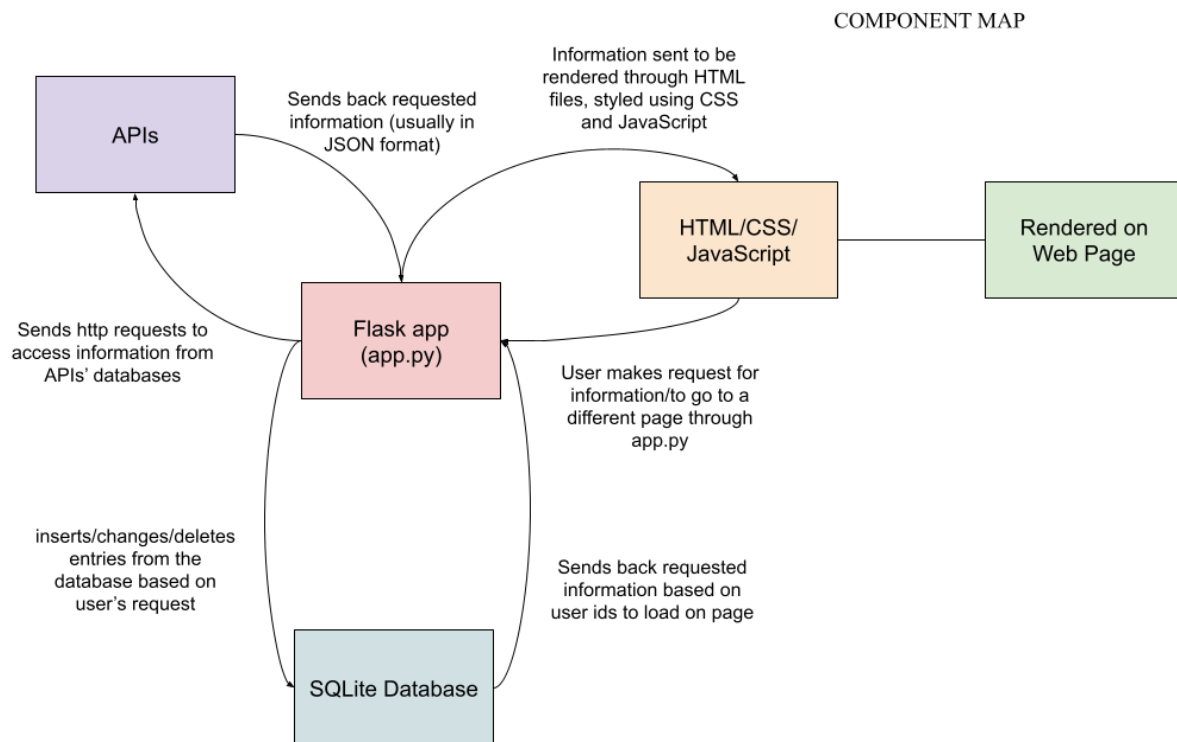Team lame (Amelia Chin, Ethan Shenker, Liam Kronman, Maddy Andersen)
SoftDev — Rona Ed.
P3 — ArRESTed Development, JuSt in Time
2021-04-29

Topic: Study Buddy that includes a to-do list, a notepad, date, time, a piece of advice, holiday of the day, weather, and a cute dog picture.

Program Components and Role of Each
- **SQLite Database**: holds our data and information so we can access it to display on the site
    - keep track of users and user content
    - to-do list content/entries
- **Flask**: the framework for our web-based app, connecting and managing the routes and functions
- **Python**: works with Flask, directs our functions and renders templates
    - Pulls information from SQLite and sends to Flask app to be displayed on page
- **HTML/CSS/Foundation**: styling for our front-end pages
- **JavaScript**: allows users to manipulate/interact with the webpage in real-time
- **RESTful APIs**: retrieve data from external databases and applications, utilize function calls defined within those applications, transmits data back after receiving an http(s) request

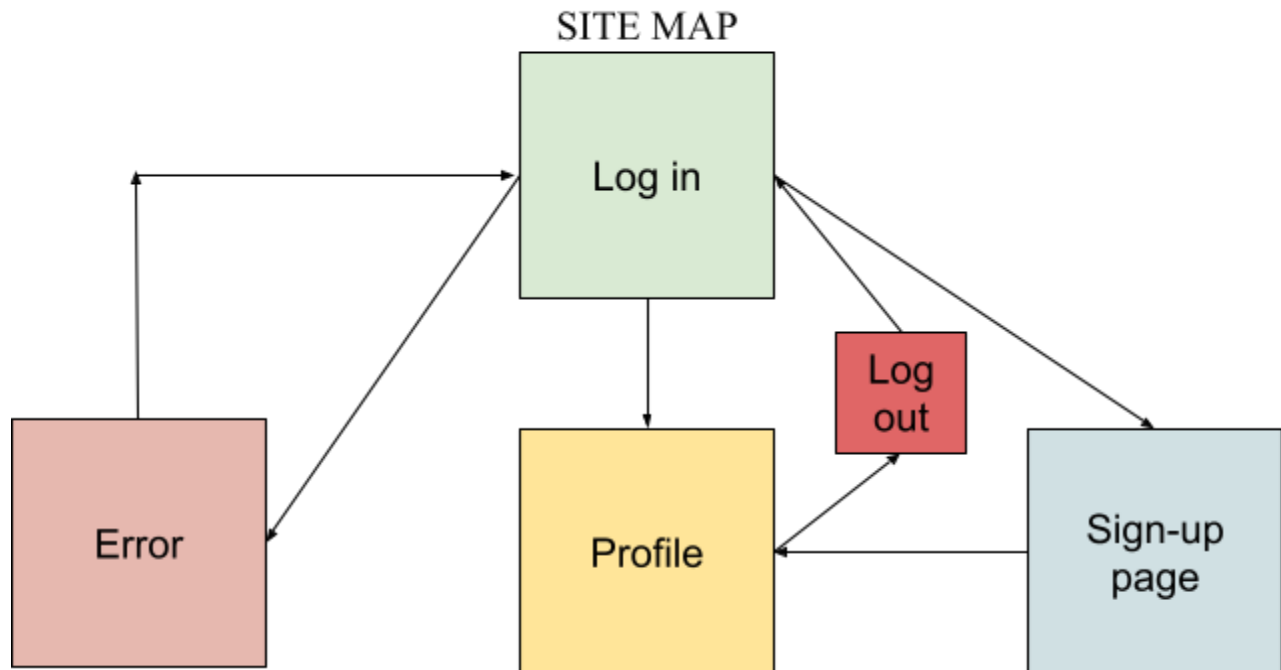COMPONENT MAP

Database Tables

User table (users)

user_id (text, primary key)	username (text, unique)	password (text)
(uuid, ex: 890c238f-e34b-478f-880c-b4f05dfd9d10)	user	123pass

User Notes Table (user_note)

user_id (text)	content (text)
(uuid, ex: 890c238f-e34b-478f-880c-b4f05dfd9d10)	user

To-Do List table (todo)

user_id (text)	item_id (text)	title (text)	body (text)	Date_time (datetime)
(uuid, ex: 890c238f-e34b-478f-880c-b4f05dfd9d10)	(uuid, ex: 890c238f-e34b-478f-880c-b4f05dfd9d10)	calculus homework	pages 3-4, due tomorrow, notes, etc.	Apr 29 2020 2:41:40

SITE MAP

Profile Page:

Task Breakdown
- <u>Login/Logout</u> -- *Ethan*
    - Registering users (setting up database of users and passwords)
    - Set up login page, register page, and error page
    - Check login info with SQLite table of users and passwords
    - Link login page with error page
    - Link login page with register page
    - Link login and register pages with profile
    - Link profile with logout page, link logout page with login page
    - Add session to keep track of the logged in user
    - Error handling -- *Ethan, Maddy*

- Profile/Home Page
    - Create personalized greeting with username, ex: "Hello, <username>" -- *Liam*
        - Change greeting message depending on time (good morning/afternoon/night) -- *Ethan*
    - Show time and date using JavaScript or Python datetime -- *Liam*
    - Display a piece of advice (Advice Slip API) -- *Amelia*
    - Display holiday (Public Holiday API) -- *Amelia*
    - Display today's weather (Open Weather API) -- *Liam*
    - Display dog picture (Dog CEO API) -- *Maddy*
        - Add dog fact (Dog Facts API) if we have time. *Update: We tried this, but the Dog Facts API does not work.*
    - Create a to-do list -- *Ethan*
        - Add entries
        - Delete entries
        - Clear all entries
    - Create a notepad -- *Ethan*
        - Edit entry
        - Save entry (can edit same entry next time)
- Static Front-End! using Foundation, HTML, CSS -- *Amelia, Maddy, Liam*
    - login.html
    - register.html
    - error.html
    - user_page.html
- Add API cards for Public Holiday and Open Weather APIs -- *Amelia*

Timeline
- Working Flask app with login and logout functionality -- *4/26*
- Personalized greeting, date and time, weather, holiday, advice, dog picture, notepad -- *4/27*
- To-do list, front-end work -- *4/28*
- Tie up loose ends, API cards -- *4/29*