Voltron // p01 // “Planner Project”

Connor Oh, Jun Tao Lei, and Leia Park / pd9

Project Manager: Connor Oh

Necessary:

Flask will serve as your web microframework.
Multiple supporting Python3 files will be used as necessary.
Bootstrap will be used as your front-end framework.
You will provide your own customized CSS where appropriate/necessary.
You will make *meaningful* use (sum > parts) of at least three (3) REST APIs, chosen from Ye Olde SoftDev API KB.

Required APIs:

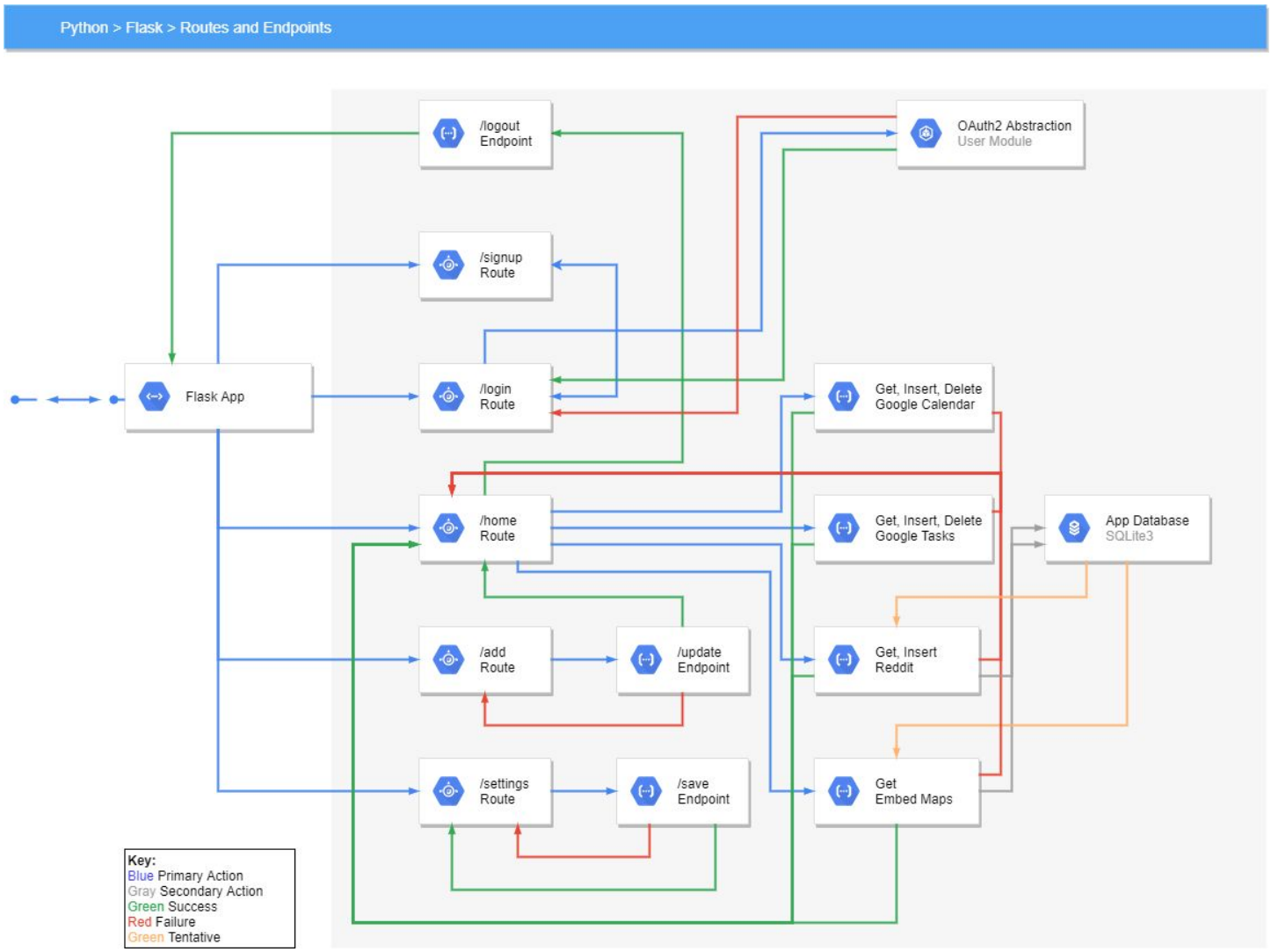
- Google Calendar API
  - See https://developers.google.com/calendar/v3/reference
- Google Tasks API
  - See https://developers.google.com/tasks/v1/reference
- Google Maps Embedded API
  - See https://developers.google.com/maps/documentation/embed/guide
- Reddit API
  - We are only using a very specific endpoint
  - Example:
    - https://www.reddit.com/r/worldnews/top.json?limit=5
    - Adding .json to the end of a url returns a subreddit’s post(s) in JSON format
    - Adding ?limit=n returns n posts from the subreddit

About/Features:

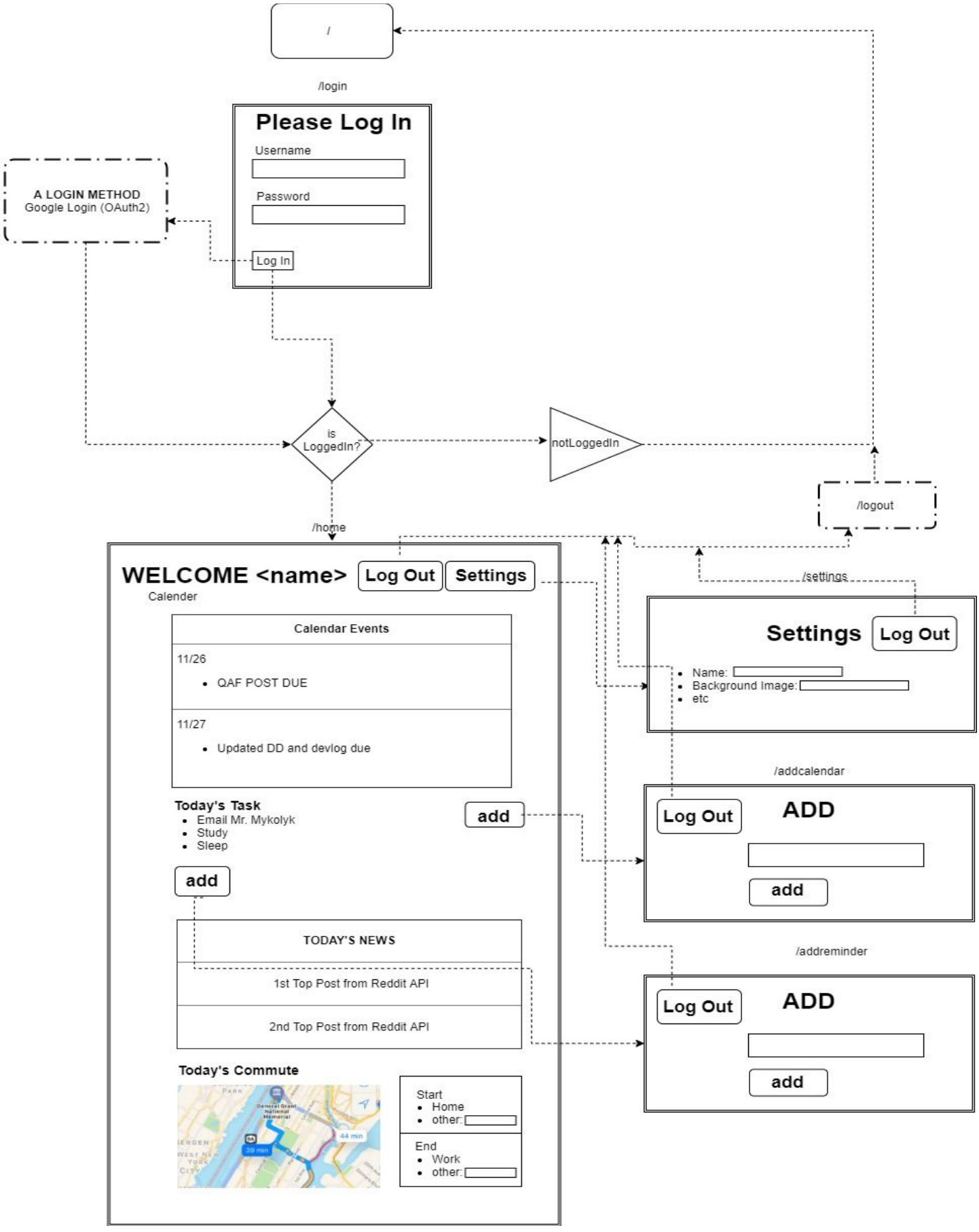
- PERSONAL PLANNER WEBSITE
  - Calendar Events (Google Calendar API)
  - Tasks (Google Tasks API)
  - Displays the Current Traffic Information for Commute (Maps Embedded API)
  - News Feed (Reddit API)
- REGISTRATION
  - User should be able to login using his/her Google account in place of registering an account
  - The app would store some basic information about the user as outlined in the database diagram later
- AFTER LOGIN
  - User should be able to access their account, provided they have already created one, using the same unique username and password
  - Edit/delete/add events and reminders
  - Display the tasks in the positions returned by the Task API

        - This is usually the importance of the task.
    - View through news feed
    - Set their home address and other locations that they commute to
- NOT LOGGED IN
    - Users can do nothing but stare at login screen

Components Map



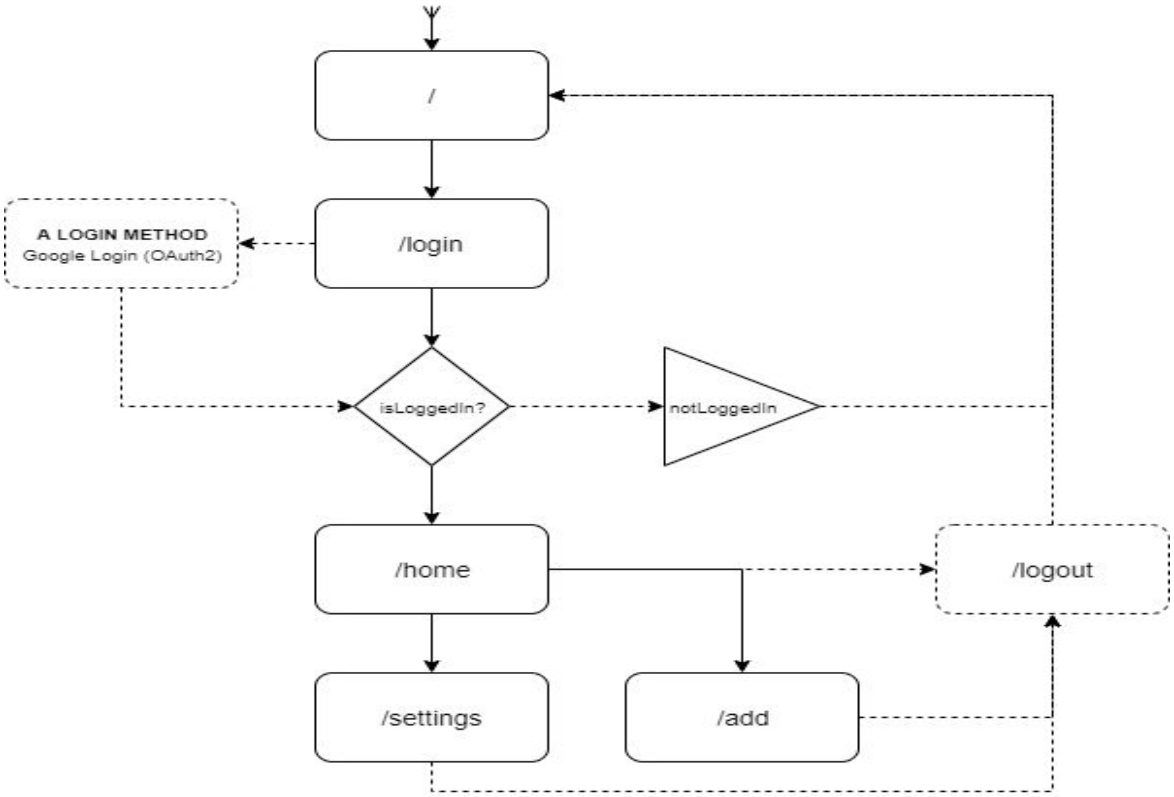
A More Visual Component Map


/
/login
Please Log In
Username
Password
Log In
A LOGIN METHOD
Google Login (OAuth2)
is LoggedIn?
notLoggedIn
/logout
/home
WELCOME <name>
Log Out
Settings
Calender
Calendar Events
11/26
QAF POST DUE
11/27
Updated DD and devlog due
Today's Task
Email Mr. Mykolyk
Study
Sleep
add
add
TODAY'S NEWS
1st Top Post from Reddit API
2nd Top Post from Reddit API
Today's Commute
Start
Home
other:
End
Work
other:
/settings
Settings
Log Out
Name:
Background Image:
etc
/addcalendar
Log Out
ADD
add
/addreminder
Log Out
ADD
add

Website Pages

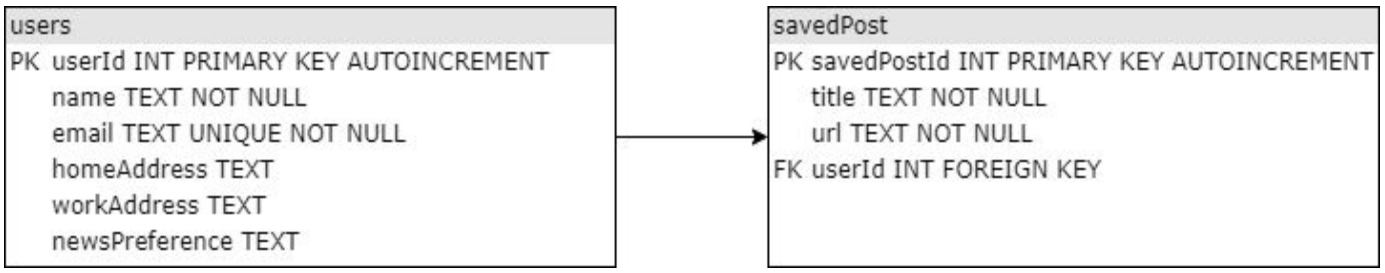
- LOGIN page
    - Login using the user's Google Account
    - Registers the user's basic info as outlined by the database diagram to the database
- HOME page
    - Buttons
        - Logout button -> routes to login page
        - Settings button -> routes to settings page
    - Shows list of reminders & "to do" tasks
    - Shows a news feed
    - Shows current traffic for commute
- SETTINGS page
    - Enables edit of home page setup
        - Edit name
        - Change homeAddress
        - Change workAddress
        - Change newsPreference
    - Personalize user's home page
- ADD page
    - Add a task or reminder

Site Map



NOTE: the dashed line show reroutes to routes that display nothing but instead do actions. The solid lines show reroutes to actual sites. Also, the line that goes from signup to login represents both the user going back to login or successfully signing up.

Database Layout Diagram



- Users
    - Each user will have his/her own user unique ID using SQLite3 autoincrement function.
    - Name represents the name of the user obtained through Google.
    - Email represents the email of the user through Google.
    - Home address represents the home address of the user that will be used in the Maps section.
    - Work address works in a similar way to home address.
    - News preference represents the preferred news provider for the user.
        - May not necessarily be implemented
- Saved Posts
    - Posts that are saved by the users have their own unique IDs.
    - Title refers to the title of the article.
    - The URL refers to the web URL of the article.
    - A user ID can be used as a foreign key that links the article to a specific user.

Tasks & Roles

- Front-end Appearance and UI (*Leia Park & Connor Oh)*
    - Bootstrap
        - Login, Home pages
- Back-end
    - Google (*Jun Tao Lei)*
        - OAuth2
        - Google API
            - Google Calendar API
            - Google Tasks API
                - Filter & personalization system
            - Google Maps Embedded API
    - Reddit API (*Connor Oh)*
        - No key necessary for pulling posts from subreddits
    - Users database (*Leia Park)*

        - Creation of database
        - Input to database from Login page
        - Accessing information from database (in order to login and register)
- Update design doc and devlog and facilitate workflow(*Connor Oh*)