Team GPStuy (William Lu, Jason Tung, Jeffrey Wu, Ivan Zhang)
SoftDev pd7
P05 -- Fin
M 2019-05-13

Project Name: GPStuy

Using **Bootstrap**

Project Description:

- Google Maps system for travel throughout Stuyvesant High School
- No account necessary
    - Guests can input 2 locations (i.e. room numbers) and we will map directions between those places
        - Directions can be:
            1. On the same floor: shortest path on the same floor
            2. On separate floors: map the instructions including staircases, escalators, elevators
                - Include options for if escalators OR elevators are NOT to be used
        - Directions will be imaged on 2D map layout of Stuy
        - Directions will be given in terms of left/right turns, distance to walk
- Students can make accounts
    - Will be able to input their schedules
        - Map instructions from one class to another
        - Map estimated travel time between classes
    - Allow users to adjust their travel times (i.e. how fast/slow they walk)
        - Use these travel times to give users more accurate results
- Extra features:
    - 3D imaging of Stuy (i.e. Google Maps Street View)
    - Add detailed descriptions and images for each Stuy room (i.e. department, courses taught)
    - Combine Mr. Brook's Schedule Choice with webpage so students know what period it is
        - i.e. account for A or B day schedules
    - Implement Parent accounts for PTC
        - Parent accounts can be linked to student accounts
    - Add directions to restaurants by Stuy

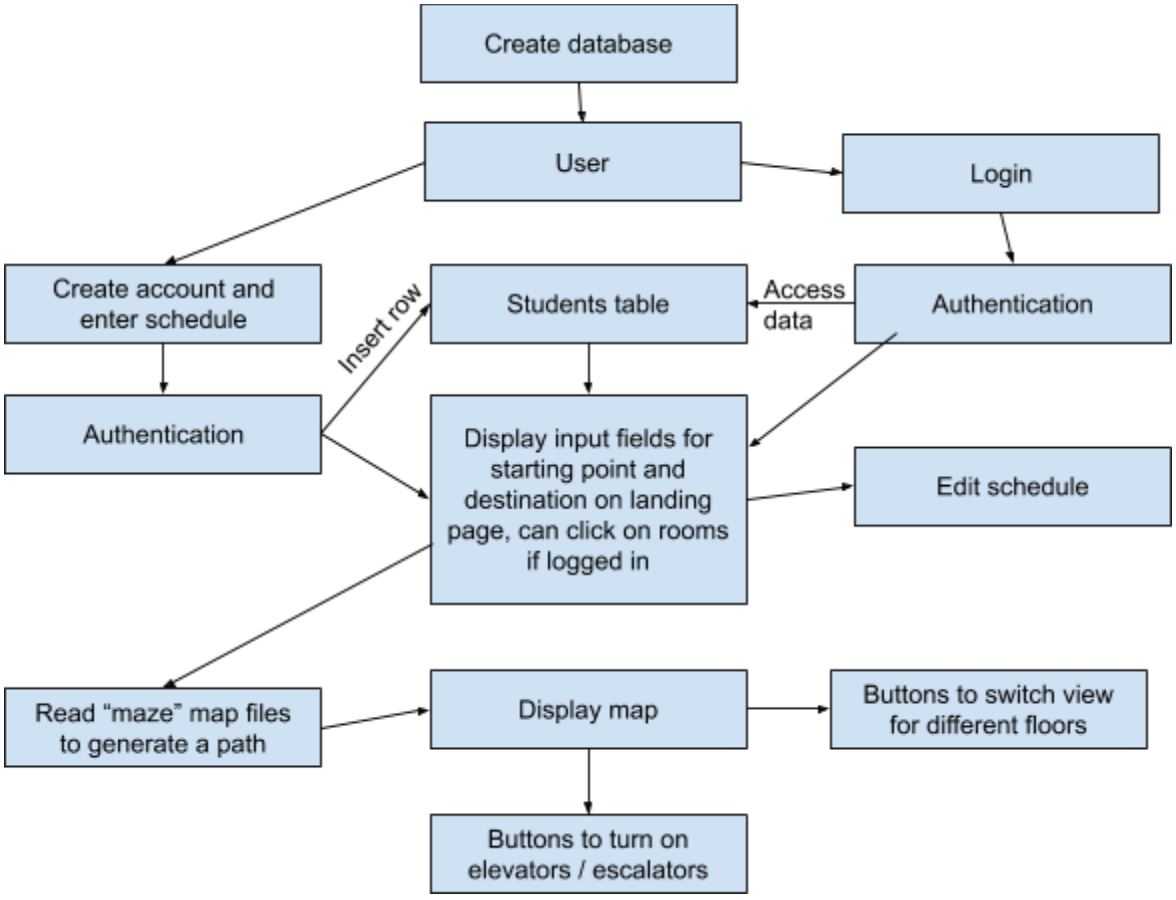
Project Manager: William Lu

<u>List of Program Components:</u>

- Computerized layouts for each floor of Stuy
    - i.e. ASCII maps of Stuy

- Algorithm to calculate best route between rooms: has to account for elevators, stairs, and escalators
    - Add parameters to algorithm to make NO escalator and NO elevator routes possible
- Landing page
    - Allow guests to enter two rooms and map the best route between them
    - Tell logged in Users their current class and route to next class
- Create SQLite database which stores the data about:
    - Each user and their login information
    - Each user's schedule
- Authentication allows only logged-in users to save a schedule
    - Logged-in users can input custom parameters (i.e. their travel time between 2 rooms) so that system can return accurate estimated travel times
    - Logged-in users can edit their schedules
- Logout capability
- Page templating to show
    - step by step instructions
    - next to a 2D layout map with route to end
- Non-core features
    - Making the website aesthetic to make the site user-friendly
    - Add accounts so that parents can inherit child's schedule
    - Add 3D street view-esque imaging of Stuy
    - Implement schedules and period timing (i.e. Brook's Schedule Choice) with website
    - Add direction functionality to restaurants by Stuy
    - Add system for admins/users (via crowdsource) to add warnings (i.e. room changes, congested stairwells, broken escalators)
    - Add descriptions and images for each room

<u>Component map/Explanation visualizing relationships between components:</u>

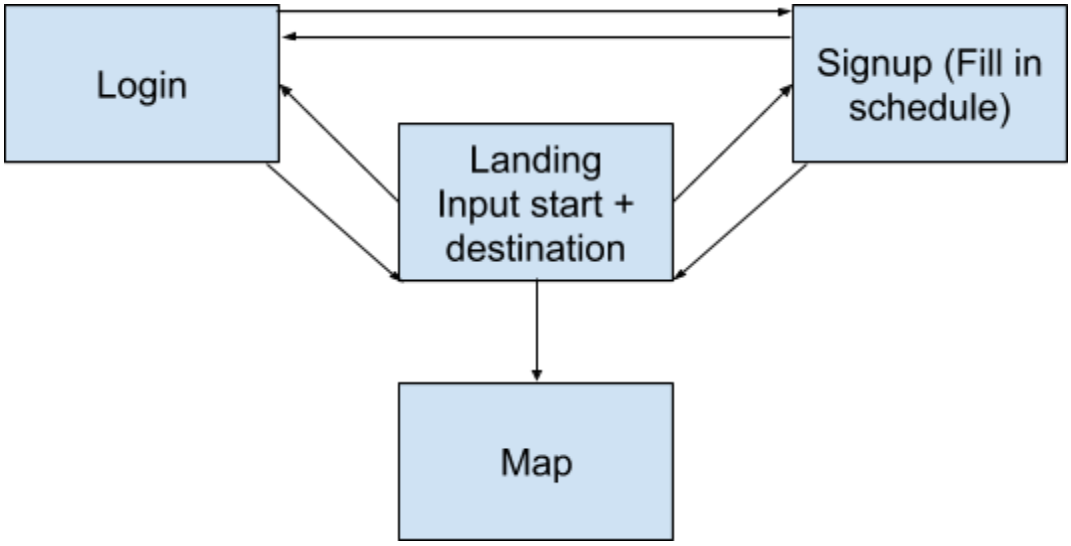


Database Schema:

Students	
Field	**Type**
Name	TEXT
Email	TEXT (Unique)
Guardian Email	TEXT (Unique)
Username	TEXT
Password	TEXT (Hash)
Period 1	TEXT (not all rooms are numbered, such as Lecture Hall A)
Period 2	TEXT
...	

Period 10	TEXT

Front End Site Map:



Breakdown of Tasks and Group Member Assignments, ordered in priority:

1. ~~Create a GitHub repo (William)~~
2. ~~Start with Flask Starter Kit (William)~~
3. Get layouts of Stuy (Ivan)
4. Transform visual layouts (i.e. PDF) into ASCII replicas (Ivan)
5. Create algorithm to calculate best route between rooms (Jason)
6. Create landing page for Guests
    a. Implement 2D map with correct route (William)
    b. Add step by step instructions (Jason)
7. Add options for No Elevator and No Escalator (Jeffrey)
8. Create database to store student accounts with correct fields (Jeffrey)
9. Add User sign up, log in, log out pages on the website (Jeffrey)
    a. Account creation on the website, Sign Up Form and Page
    b. Account login and authentication, Sign In Form and Page
10. Create database to store student schedules (Ivan)
11. Allow students to input their schedules on the website (Ivan)
12. Allow students to edit their schedules (Ivan)

Extra Features

13. Add descriptions and images to each room (William)
14. Combine Mr. Brook's Schedule Choice timing (Jason)
15. Admin Accounts (Ivan)
    a. Increase to scalability
    b. Allow admins to upload layouts of schools
    c. Admins can mass email schedules

    d. Admins can directly email one student

~~16. Implement Parent accounts (NO LONGER NECESSARY)~~

    a. Instead of Parent accounts, we will allow admins (and maybe Big Sibs) the ability to mass email OR email a specific student the schedules and directions for their parents (if there is a guardian email associated)

17. Add directions to nearby restaurants (Jeffrey)
18. Allow users to submit requests (i.e. broken escalator, congested stairwell, change of room) (Ivan)

<u>Tasks being done concurrently:</u>

1. Maintaining devlog (Everyone)
2. Revising Design Doc (William)
3. Entries in devlog addressing mid-development modifications (William)