TNPG: MegaNuclearSuperJesus
Roster: Jacob Guo, Marc Jiang, Nicole Zhou, Sam Cowan
Soft Dev
2022-12-05
Target ship date: 2022-12-21

Our Project:

College picker (search) which offers weather/climate of that region, commute or travel to that location, points of interest, and miscellaneous college info.

Search bar with dropdown on homepage which you can type a college name into. If there is a dropdown, you can select an option to autofill and search.

Query Results page gets the latitude and longitude of the college from geonames and plugs into a few apis. Bing maps finds points of interest, and lists the top five out on the page. Uses bing maps to determine travel time by car, train, and price of driving with gas buddy API. Should include a picture of campus.

Purpose is mainly to help a high school senior prepare for their first year of college, helping explain logistics of getting to and from college, etc

Program List

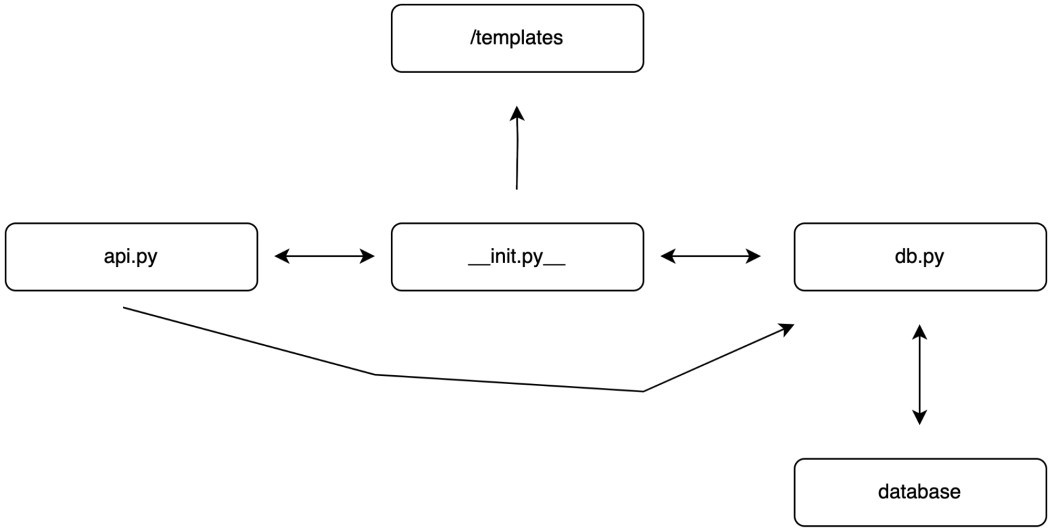
1. **__init__.py** - Basic driver with Flask and Foundation

2. static/css/ - Basic formatting that FEF doesn't take care of

3. js/ - Keeping it to a minimum

4. templates/ - our HTML files for the respective pages we need
    a. login
    b. register
    c. Home - search bar and favorites
    d. results
        i. College name, location, distance/travel cost, weather/climate of location
    e. root/main - directs you to register or login

Component Map



- **login.html**: users can login in with an existing account. If information entered is incorrect, the user stays on this page. If the correct username and password are entered, the user gets directed to the home page (home.html). If the user does not have an account, they can go to the signup page and signup.
- **register.html**: where users create an account. After creating an account, the user is redirected to home.
- **home.html**: user can search for colleges (default tab). The user can also go to the "favorite" tab and check colleges in their favorite list. After searching for a specific college, it brings the user to the results page (results.html).
- **results.html**: shows information about the specific college researched. Like button is stored here

Database Organization

2 tables:

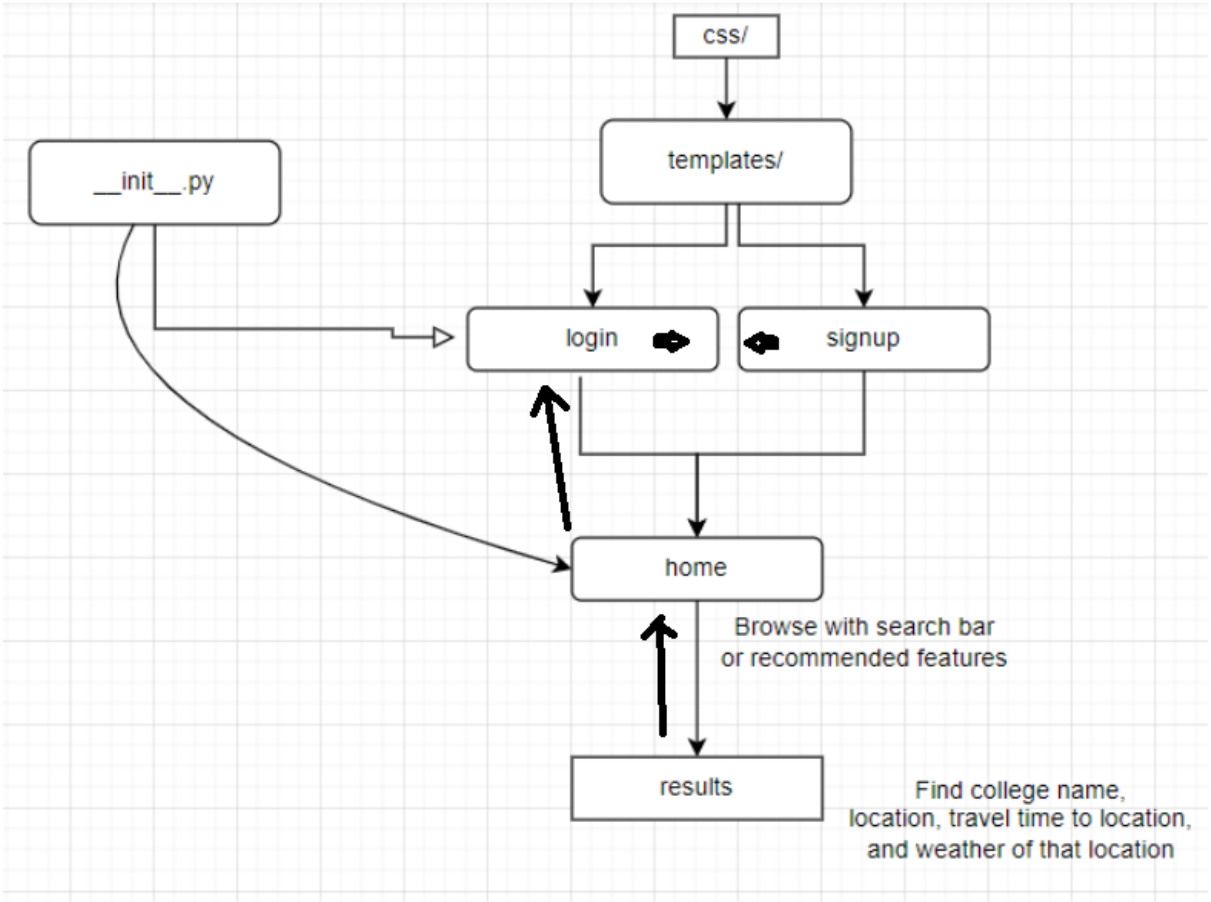
1) To store user login credentials:

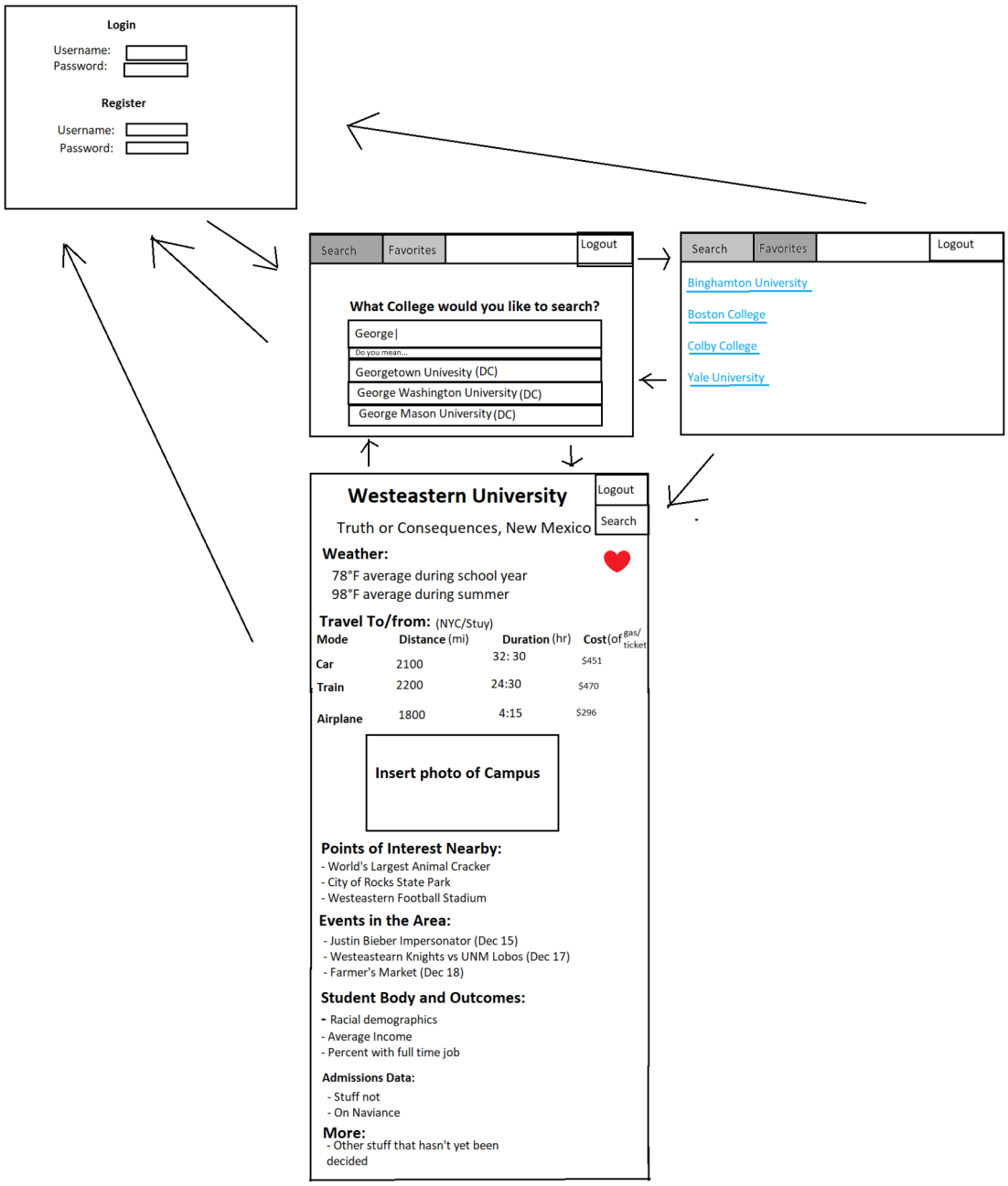
Username (TEXT_UNIQUE)	Password (TEXT)
marc123	password
jguo30	guoster123

2) To store favorited colleges:

Username	Favorited colleges (append name of college to string)
Marc	Westeastern, Hardvard, Princesston, UPeen
jguo30	Bing

Site Map (Front-end





Assignments to Members

Jacob

- PM (the best)
  - Talking to Mr. Mykolyk and organizing the group
- HTML/CSS/JS
- API kb writing, testing

Marc

- Database Engineer
    - Write database functions and understand database structure
    - In charge of favorites db and like button

Sam
- API Specialist
    - Write API kbs, learn how to use each API to get desired results
- Flask incorporator
    - Write flask things (with specialty in connection to APIs)

Nicole
- Front end framework (foundation)
    - HTML and CSS and JS in conjunction with Foundation

MVP:
- Login page w/ corresponding DB
- Search bar on homepage which works with college names
- Query page providing basic info (as presented in site map) about colleges

Goals:
- Dropdown under search bar as autofill field
- Favorites Section added to navbar on "homepage"
- Calculations for certain fields with API generated numbers (such as price of driving with current gas prices)

Stretch Goals:
- Recommend the best (easiest and most cost-effective) form of travel
- Search for college given features instead of name (requires a lot of DB tinkering)
- Keep track of past searches

API Section

1) Bing Maps
    a) Calculate distance and travel time from one point to another
        i) Walking, car, transit
2) College Scorecard
    a) Contains tons of info on colleges
        i) College name
        ii) location (latitude, longitude to be used w/ Bing Maps)

iii) type of area (urban, rural, big city, small city, etc.)
iv) General info (amount of students, cost, etc.)(much of this is actually unavailable)

3) Weather API (open-meteo)
   a) Fetch local weather and climate info based on location (latitude, longitude, or city)
4) Gas Buddy
   a) Find gas prices in the locale closest to a lat/long

FEF Features Why and How?

<u>Foundation</u>

- Search bar
- Dropdown bar
- Tabs on homepage
- Button to add colleges to the favorite list
- Card

We chose foundation because it felt conceptually easier. We also liked its grid features and felt it matched up closely with bootstrap.