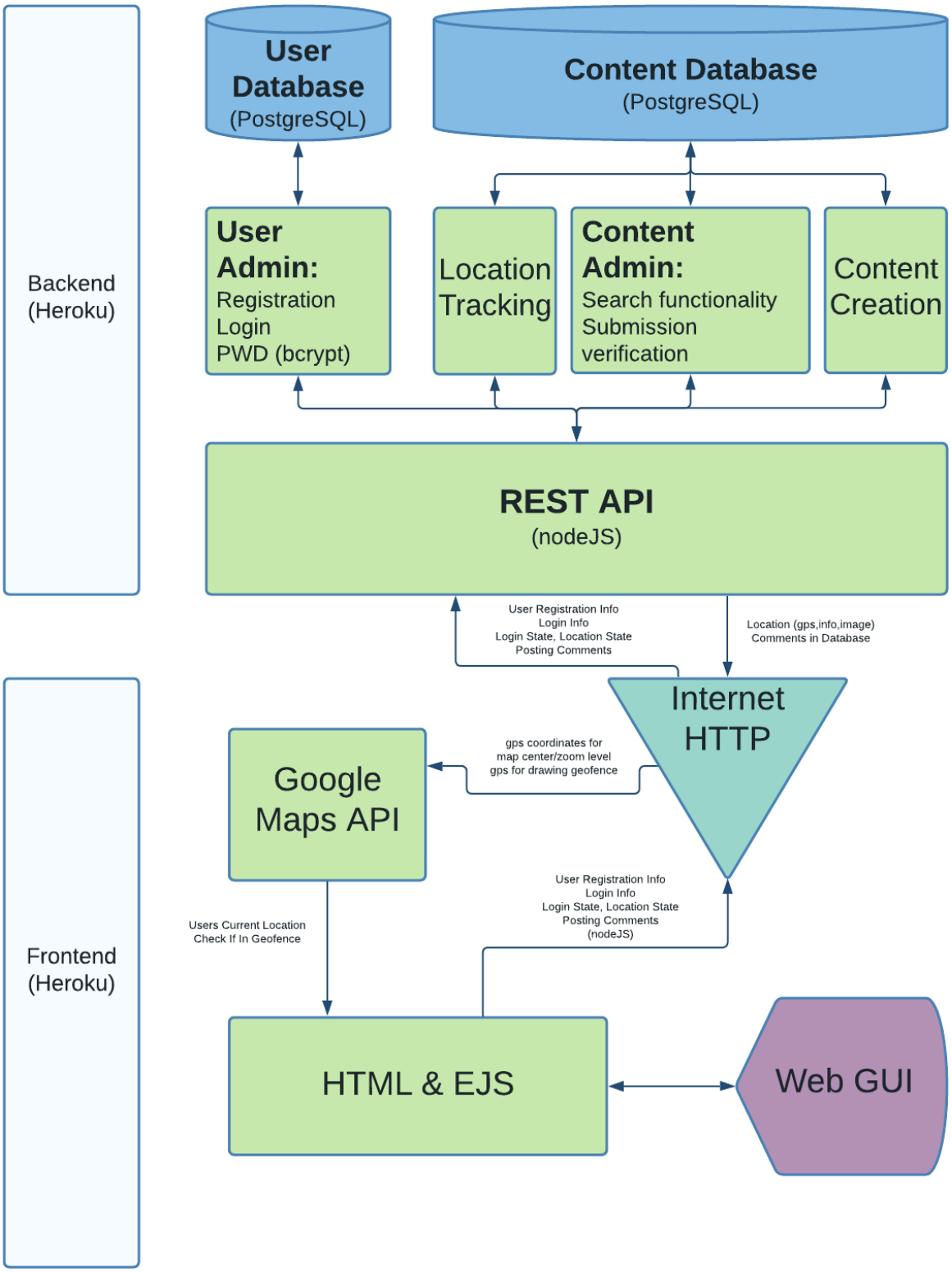
Team 13-01 (Profanity)
Milestone 4

Revised List of Features

- User can register for an account (priority HIGH)
- User can sign in to account (priority HIGH)
- Session created after sign in authentication (priority HIGH)
- “Remember Me” on sign in form extends session age to 1 week (priority HIGH)
- Forgot password lets user reset password (priority HIGH)
- Password encryption (priority HIGH)
- Google Maps API
    - Shows current location (priority HIGH)
    - Locations highlighted on map (priority HIGH)
    - Checks if current location is within geofence on map (priority MED)
- Comments based on location (priority HIGH)
- Once authenticated (unique user session created), users can comment on a location and use all other features (priority HIGH)
- If users are not logged in, commenting, liking, and upload features are disabled (priority HIGH)
- List of suggested locations (Stretch)
- User can like/favorite a location (Stretch)
- Users can report comments and reports sent to devs (Stretch)
- Users can like comments by others (Stretch)
- Users can upload images of locations (Stretch)

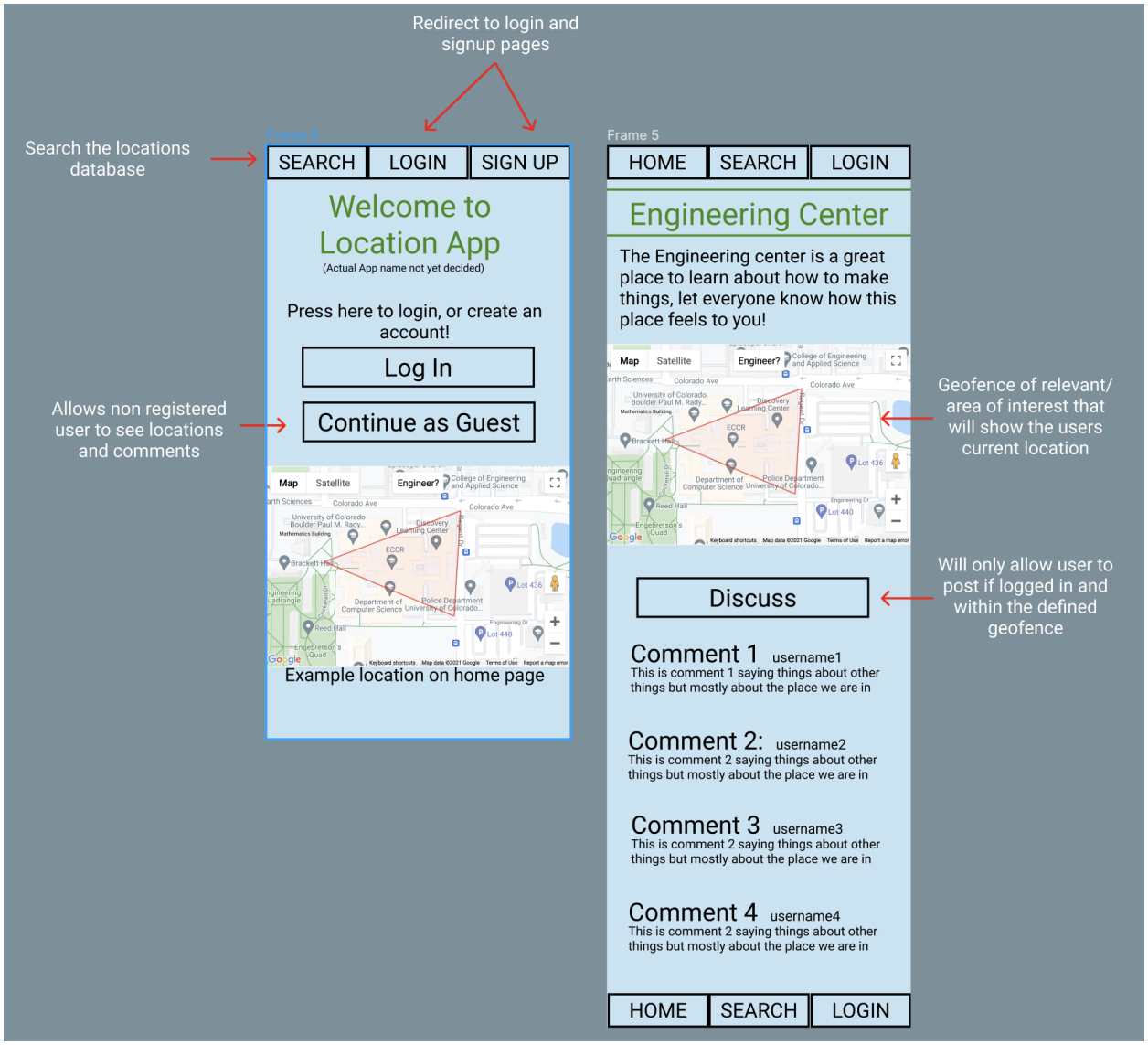
Architecture Diagram

Team Profanity 013-01

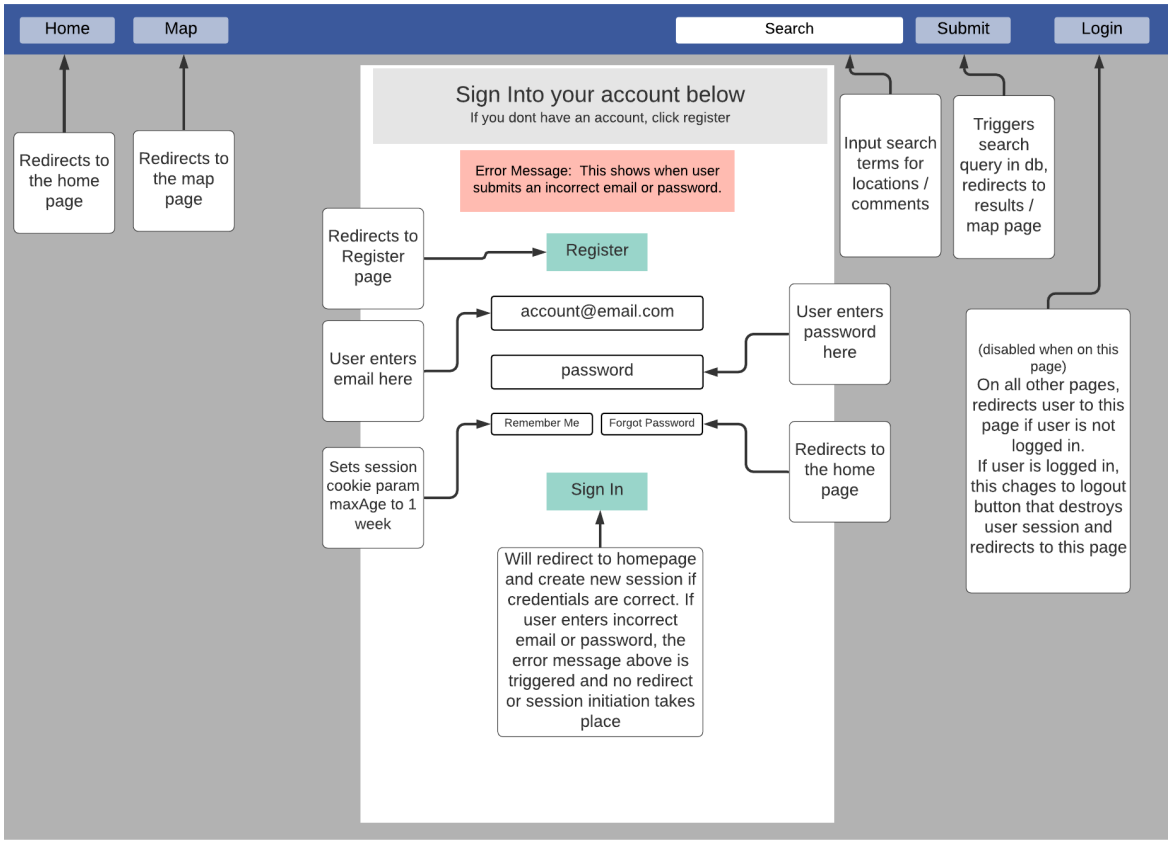


Front End Design

Home Page/Location Page



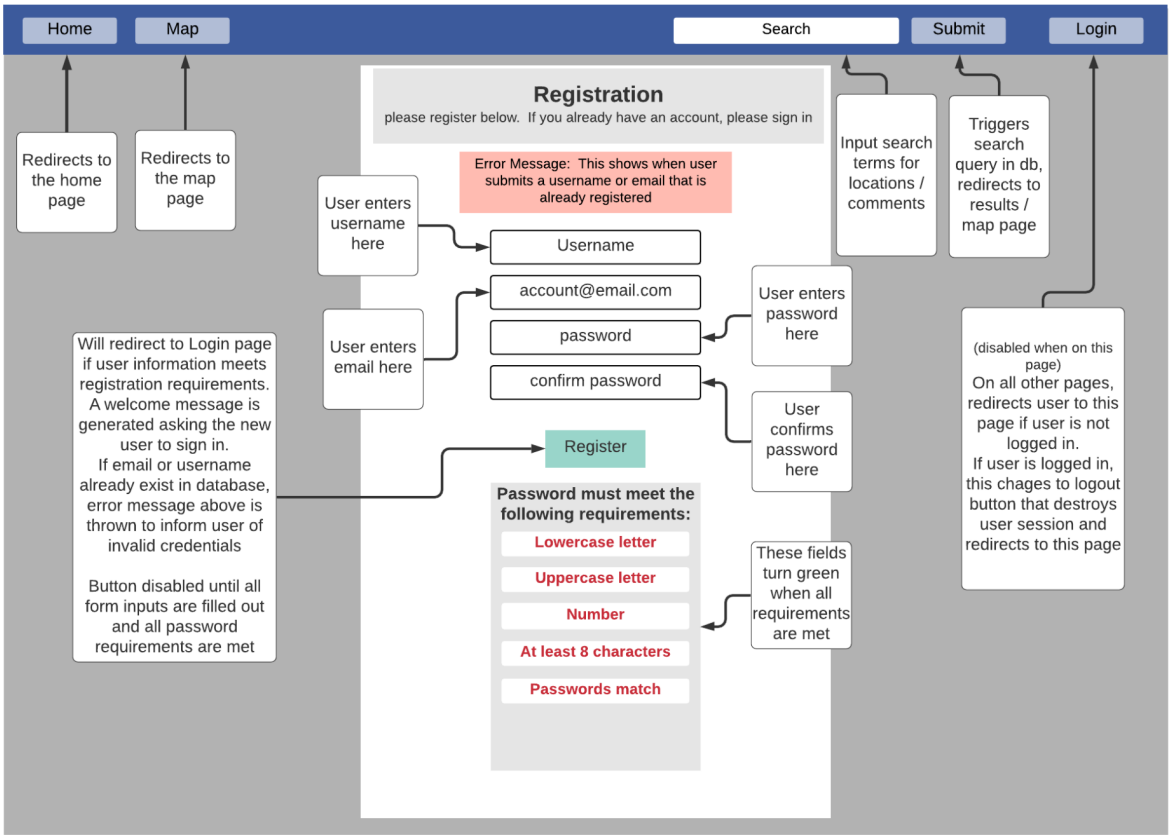
Sign In Page



Registration Page

Registration Page Diagram

Owen Arnold | October 29, 2021

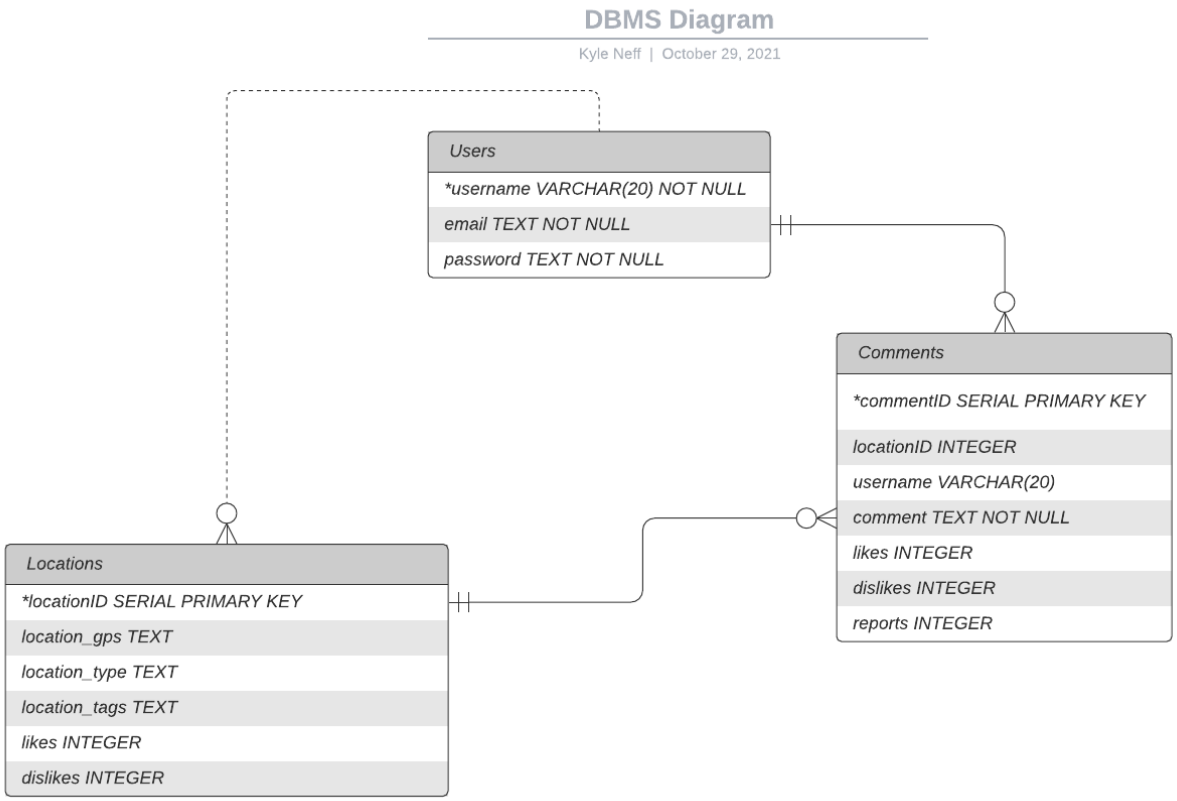


Web Service Design

We are using the google maps javascript api to handle our user locations and drawing the geofence of the location/area of interest. There is also a method to check whether the user's location falls within the specified area. The data passed to the api is gps coordinates that define the center of the map, and the bounding box (geofence) of the location of interest, as well as an integer that sets the zoom level of the map. We also use the API to get the user's current location and check to see if that location falls within the geofence.

Database Design

- We are using PostgreSQL to store our application data



Challenges

- One challenge we are currently facing is how to host our site using Heroku. This challenge shouldn't persist too much however, after our lab next week on Heroku. If we continue to struggle with it after that we'll go to office hours and get help with a TA.
- One major challenge faced this past week was sign in authentication and session creation. The pg-promise dependency made it difficult to work out the code for these features, but Owen managed to figure it out and get the features working. The next roadblock we will face for these features is transferring from a locally hosted app to a fully operational web hosted app.
- Another issue is finding actual content for our site, such as the locations and that will be on it. This is a relatively easy thing to solve, but could prevent us from moving forward as we get closer to finishing the site.
- Another challenge that lies ahead is user submission verification and the comment report feature. We have not begun this yet, and it will be a time crunch to get it done before the deadline.

Individual Contribution

Nolan: Currently working on trying to get Heroku working, so that we can have our site up and running. Helped update the front end design and built the user database. Will also start working on filling the content database with locations.
CU-CSCI-3308-Fall-2021/CSCI-3308-Fall21-013-01 at nolan (github.com)

Owen: Fully implemented user registration and sign in authentication. These features are connected to our backend database and user passwords are successfully encrypted using bcrypt for nodeJS. Upon successful authentication, a new session is created for the user using express-session, creating a session id from the user's unique id stored in the users table in the database.
Link to latest commit:
https://github.com/CU-CSCI-3308-Fall-2021/CSCI-3308-Fall21-013-01/commit/1930a1b2209fd84a5cce87896274ea442490111b

Kyle: Successfully connected comments backend to a temporary frontend page where there is a table with all the comments, and below is a submission box where if you add a comment it will load into the table and display on the webpage.

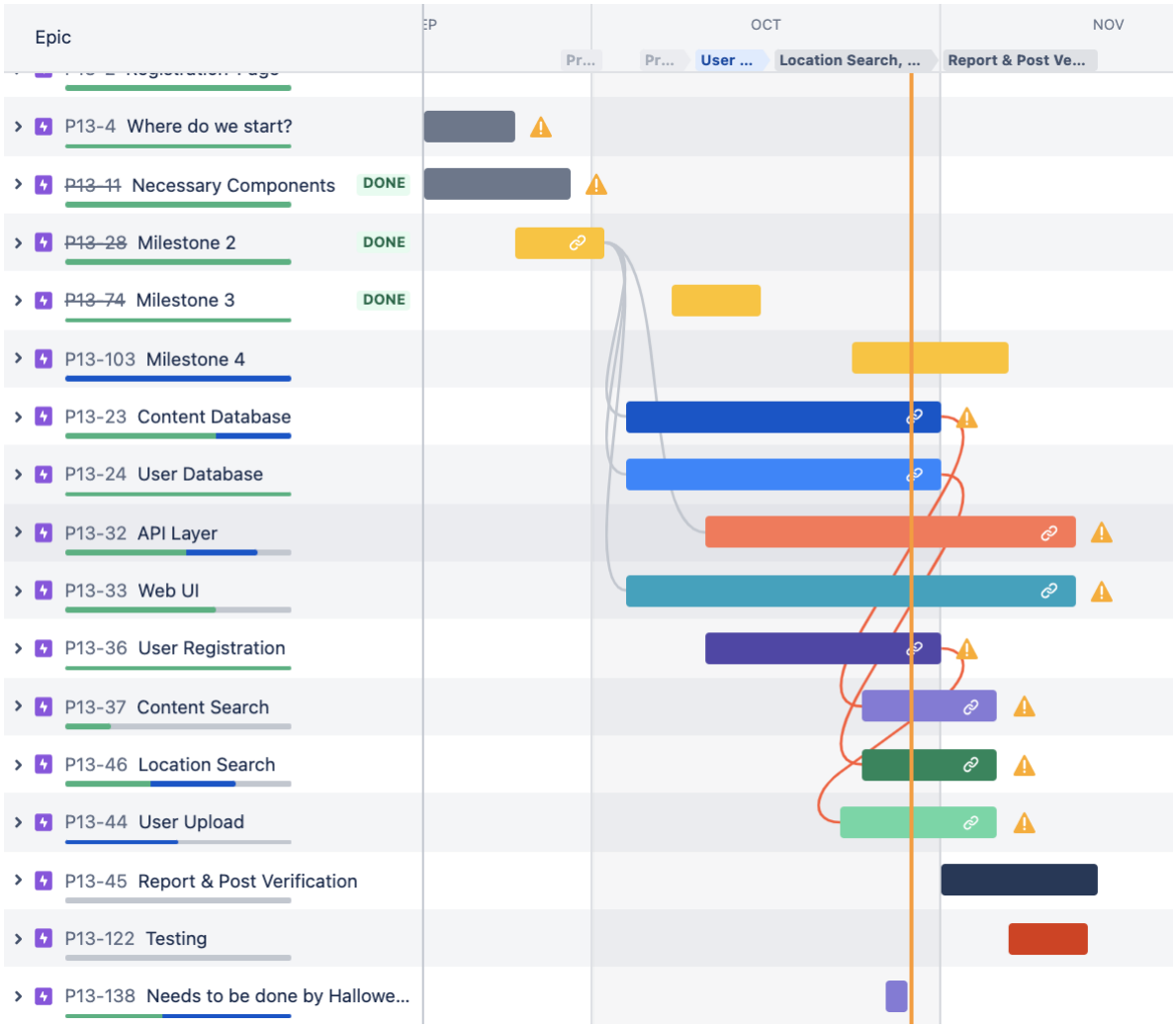
Link to latest commit:
https://github.com/CU-CSCI-3308-Fall-2021/CSCI-3308-Fall21-013-01/tree/main/webpage_code

Even: Continuing to work on getting locations database working with google maps api, got the maps code to work in the ejs file and linking to the database of locations.
Re-designed the locations database to contain gps coordinates of geofence ot be drawn.
Finalized the application architecture diagram and finalized the description of the google maps API.

Latest Commit: (will merge once I get location geofence to draw from db, I am aware of NPM bloat from getting maps api working)
https://github.com/CU-CSCI-3308-Fall-2021/CSCI-3308-Fall21-013-01/commit/ef0c0774d83b0150a3a1d2841bf3dc885e6a3d14

Yefan: Still working on the searching function for searching comments and locations, and trying to figure out the way to add searching results into the Web UI in an aesthetic fashion. And help to list the revising list of features. All I work is on my pc, haven't uploaded it yet.

JIRA Board:


Epic
EP
OCT
NOV
Pr...
Pr...
User ...
Location Search, ...
Report & Post Ve...
P13-4 Where do we start?
P13-11 Necessary Components
DONE
P13-28 Milestone 2
DONE
P13-74 Milestone 3
DONE
P13-103 Milestone 4
P13-23 Content Database
P13-24 User Database
P13-32 API Layer
P13-33 Web UI
P13-36 User Registration
P13-37 Content Search
P13-46 Location Search
P13-44 User Upload
P13-45 Report & Post Verification
P13-122 Testing
P13-138 Needs to be done by Hallowe...