Study on Campus

CS 326 Group 11
Milestone #4 | March 28th, 2025

Note: GIFs in Code descriptions were converted to still images, please click here to view them animated.

Links: Repository & Milestone #4

Project Overview

Working with a study partner or group can provide increased understanding, accountability, routine, camaraderie, and productivity. However, finding an effective one can be difficult to students for a multitude of reasons: large classes, social anxiety, mismatched goals, unclear expectations, and lacking a method of contact. **As a solution, we propose a web application that offers a platform to help students more easily connect with each other, find study locations that promote productivity, and organize or find study groups.**

Key Features

- **Study group posting:** Students can post open invites for study groups, along with their goals, preferences, and location. Users can browse and filter for groups that match their needs.
- **Communication between users:** Users can discuss meeting details and coordinate study group plans through public comments on invite posts and/or private messages.
- **User-reported location crowding score:** Users can rate how crowded popular study spots are to help others make more informed decisions.

Team

Erika Elston (Project Manager)

Primary Contributions: Completed interface mockups for **location browsing** and **user profile page**.

Julia Farber (Time Keeper)

Primary Contributions: Completed interface mockups for **post browsing and search page** and the **post viewing page**. Implemented both UI's in HTML & CSS. Created two future tasks in repo.

Anastasia Isakov (Project Organizer/Documentation Lead)

Primary Contributions: Completed interface mockups for **post creation page, login page, and sign up page**.

Ashley Kang (Quality Control)

Primary Contributions: Completed interface mockup for **settings page**.

Historical Development Timeline

M	T	W	Th	F
		3/5 **Team meeting** Initial concept design & wireframing (Figjam)	3/6	3/7 **Deadline** Milestone #3 (Application Design)
3/10	3/11 **Team meeting** Discuss Milestone #4 requirements **Development** Began location browsing page mockup	3/12	3/13 **Development** Began user profile page mockup	3/14
3/17	3/18	3/19	3/20	3/21
3/24 **Development** Began post creation, browsing/search page mockups	3/25 **Team meeting** Milestone #4 progress check-in **Development** Began post viewing page mockup	3/26	3/27 **Development** Opened PR for location browsing page mockup Began login and sign up, settings page mockup **Planning** Created future task: User Profile Icon Generation	3/28 **Deadline** Milestone #4 (Interface Mockup) **Development** Opened PR for post creation, login, sign up, post browsing/search, post viewing page, user profile, settings page mockups Closed PRs for all pages **Planning** Created future tasks: Post Browsing Search Filtering, Password Change Page

Erika Elston

Work Summary

Issues:

- User Profile Screen (UI)
- Location Browsing and Crowding Score Reporting Screen (UI)
- Location and Crowding Information Data Structure (Data)
- Project Manager (Team)

Commits:

- **659b89d** (3/11): Set up frontend folder for Milestone #4 submission. Set up frontend/location-browsing-page folder, along with initial mockup for location browsing page (HTML & CSS). Progress and remaining tasks noted here.
- **48fd813** (3/11): Set up frontend folder for Milestone #4 submission (for main branch).
- 936ccd9 (3/13): Initial mockup for user profile page (HTML & CSS). Notes here.
- **b89d254** (3/27): Refined interface mockup for location browsing page. Notes and demo here.
- cbd50f6 (3/28): Refined user profile page interface mockup. Notes and demo here.

Pull Requests:

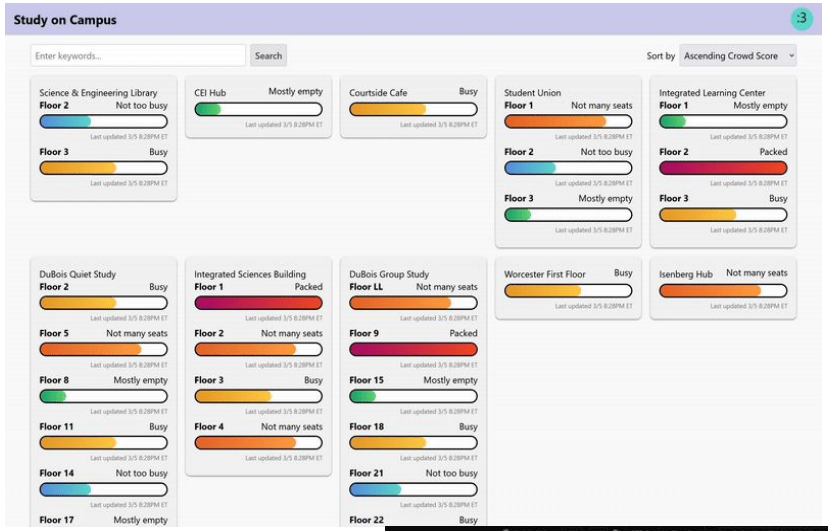
- **#21** (3/11): Add frontend folder to main branch. Closed without review (due to simplicity).
- **#25** (3/27): Implementation of location browsing page interface mockup. Required team review.
- **#34** (3/28): Implementation of user profile page mockup. Required team review.

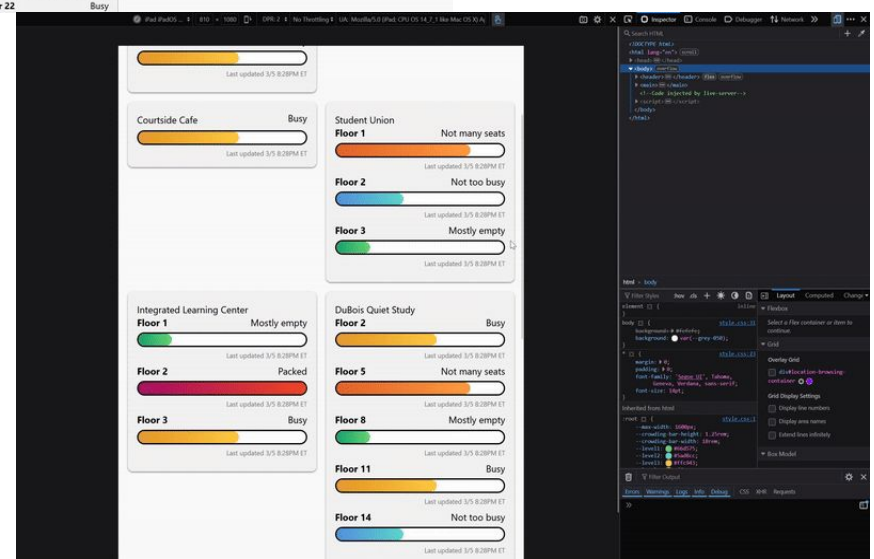
UI Components: Location Browsing

Interface mockup demo in full screen (2560x1600), tablet, phone and various window widths.

Screen width responsiveness for search functions implemented with flexbox. Responsiveness for location cards implemented with grid.

Navbar is accessible across the web app, including this page. Users are provided filtering options via keyword search or parameter sorting. Users may click on a card to expand the post for details (indicated by darkened color and cursor change on hover).



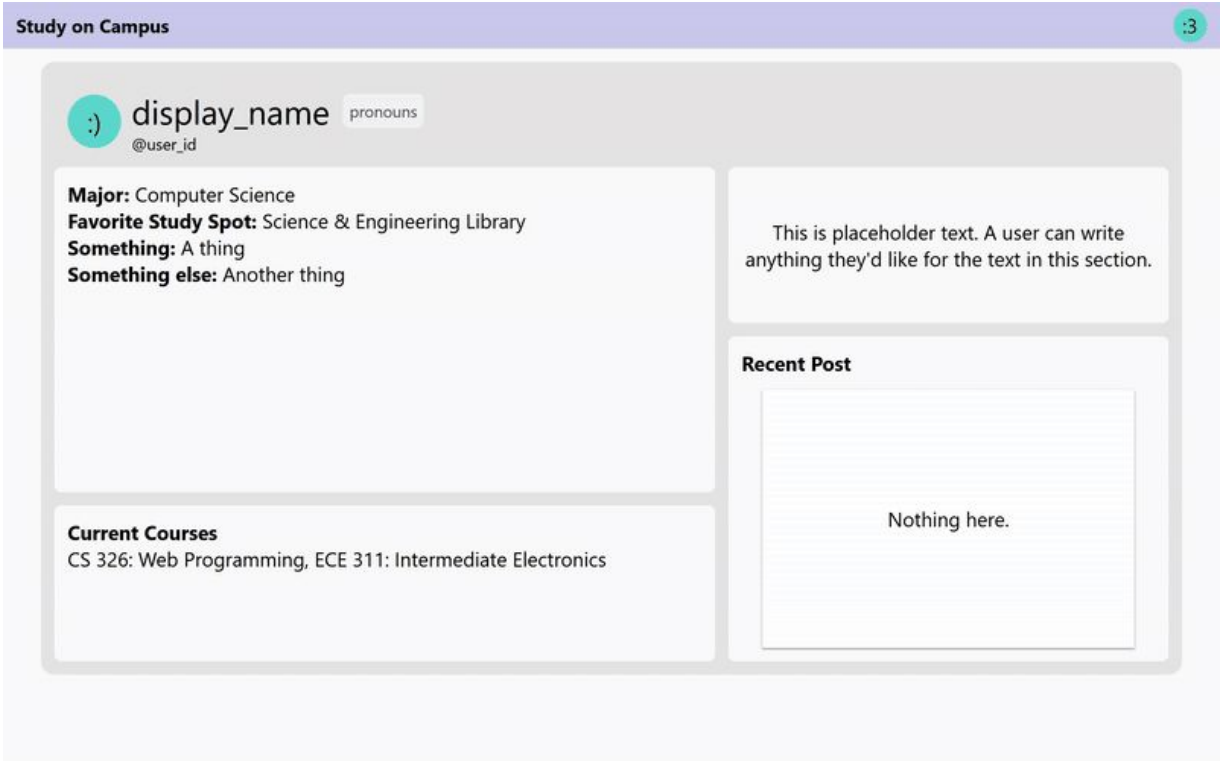


UI Components: User Profile

Interface mockup demo in full screen (2560x1600), tablet, and various window widths.

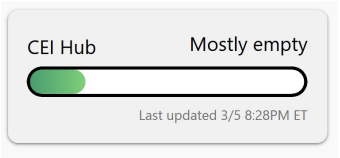
Screen width responsiveness for profile “header” implemented with grid (wider screen width) and flexbox + flex-wrap (smaller screen widths). Responsiveness for profile subsections implemented with grid.

Navbar is accessible across the web app, including this page. Users may access the profile owner’s favorite study spot via a link (indicated by underline on hover in GIF), or view their most recent post by clicking on the embed.



Code

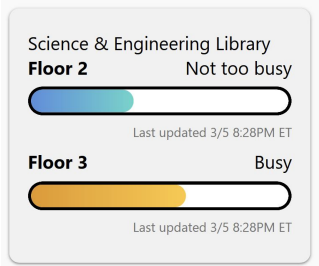
A key task in implementing the interface mockup for the location browsing screen was determining the HTML structure of each card.




```html
<div class="location-card-container">
    <div class="location-name"><span>Science & Engineering Library</span></div>

    <div class="floor-info-container">
        <div class="location-info">
            <div class="floor-name"><strong>Floor 2</strong></div>
            <div class="crowding-score-hint">Not too busy</div>
        </div>

        <div class="crowding-score-bar-container">
            <div id="level2" class="crowding-score-bar"></div>
        </div>
        <div class="crowding-score-updated-time">Last updated 3/5 8:28PM ET</div>
    </div>

    <div class="floor-info-container">
        <div class="location-info">
            <div class="floor-name"><strong>Floor 3</strong></div>
            <div class="crowding-score-hint">Busy</div>
        </div>

        <div class="crowding-score-bar-container">
            <div id="level3" class="crowding-score-bar"></div>
        </div>
        <div class="crowding-score-updated-time">Last updated 3/5 8:28PM ET</div>
    </div>
</div>
```

```html
<div class="location-card-container">
    <div class="location-info">
        <div class="location-name"><span>CEI Hub</span></div>
        <div class="crowding-score-hint">Mostly empty</div>
    </div>

    <div class="crowding-score-bar-container">
        <div id="level1" class="crowding-score-bar"></div>
    </div>
    <div class="crowding-score-updated-time">Last updated 3/5 8:28PM ET</div>
</div>
```

I found myself initially struggling to achieve the look I was aiming for. In particular, I wanted the crowding score hint (e.g., “Mostly empty”, “Not too busy”) to appear next to the location name if there was only a single floor, but next to the floor name if there was more than one.



My solution was to keep the structure of single-floor locations the same, with the “location-info” div containing the name and hint. The crowding score bar container and score timestamp existed outside of this. For multi-floor locations, the location name exists on its own. Instead, a “floor-info-container” div exists for each floor, with “location-info” containing the floor name and hint, along with the floors corresponding score bar and timestamp.

This HTML structure will likely be generated using JS, with a conditional checking for the number of floors a location has. Additional thoughts here.

In the future, clicking on a location card will cause the card to “expand”, via DOM manipulation, providing the user with additional information and the option to report on the crowding score, for other users to reference.

Later on, references to specific locations on different pages of the app will lead to the location browsing page, filtered for that specific location. This will allow users to easily check on the crowding status of that study spot.

Code

```
:root {
    /* defining variables */
    --max-width: 1600px; /* main content max width */
    --crowding-bar-height: 1.25rem;
    --crowding-bar-width: 18rem;

    /* crowding bar colors */
    --level1: #66d575;
    --level2: #5ad8cc;
    --level3: #ffc943;
    --level4: #ff9e42;
    --level5: #f24822;

    /* component colors */
    --grey-050: #fafafa; /* lightest */
    --grey-100: #f2f2f2;
    --grey-200: #e4e4e4; /* card hover grey */
    --grey-300: #b9b9b9;
    --grey-400: #757575; /* drop shadow grey */
    --grey-500: #424651; /* darkest */
}
```

While implementing each interface mockup, I found myself using and reusing many colors for various components. I defined a tentative color scheme to help with design consistency and cohesiveness. I did have some difficulty at times with the different shades of grey; I kept having issues with contrast and feeling like I needed more to work with. For this milestone, I stuck to six shades, but these may be altered later on.

Going forward, we will work as a team to standardize our UI design across each page of the web app. Sitewide formatting, e.g., main.css, could be a potential starting point.

Challenges and Insights

I've had some experience messing with HTML and CSS prior to this project, but my previous approach to design was primarily based on trial and error. This was my first time working with a true design goal, set before even opening VS Code. Having a design as reference made outlining the HTML and beginning the CSS go by quicker. Using the Inspect Element tool made flexbox and grid structures easier to understand. The MDN Web documentation also provided a lot of support.

Navigating interactivity, media queries, and design responsiveness with pages featuring multiple sections was a bit of a challenge. Aiming to prioritize readability and intuitive design helped frame my approach to the CSS, particularly for the location browsing page.

Implementing the interface mockup as a team has highlighted the significance of having multiple perspectives; a lot of great ideas for future tasks and design goals have come out of discussing our ideas with each other.

Future Improvements and Next Steps

Quite a few design elements require Javascript to achieve the functionality and/or visual our team has in mind. One of the first next steps will be to refine our UI using JS.

Implementing the interface mockup also led to ideas for future features and refinements:

- **Login and signup screens (#22):** While working on the user profile page mockup, I realized that we didn't yet have a screen for account creation. This goal was refined and completed within the Milestone 4 deadline by Anastasia! :)
- **Visual preferences (#36):** In a similar vein as user profile icon generation (based on display name or username), additional customization could come in the form of allowing users to adjust how the app looks to them, as well as how their profiles appear to others! This could be a fun UI goal to pursue after we complete some baseline functionality.
    - Potential customization options: site background/theme (dark mode?), navbar colors, customized profile icon colors, user profile background/subsection colors

Anastasia Isakov

Work Summary

Issues worked on in this Milestone:

- Post Creation Screen (UI)
- Login Screen (UI)
- Sign Up Screen (UI)

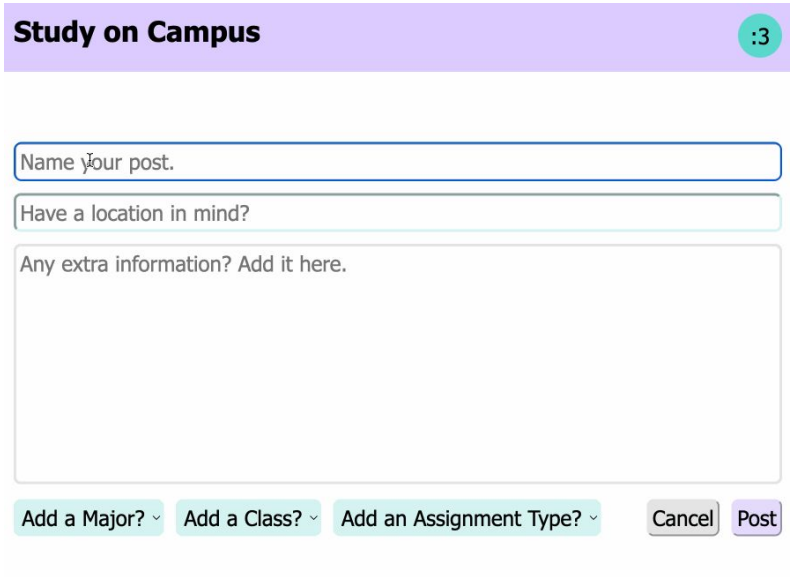
Commits:

- 3d46f5 (3/28): Initial mockups for the post creation page (HTML & CSS).
- f95ac27 (3/28): Updated post creation page diagram to better reflect mockups.
- 12e8704 (3/28): Initial mockups for the login and sign up pages (HTML & CSS), updated README with new issue for login and sign up pages, added diagrams for login and sign up pages.
- e3d53b5 (3/28): Fixed spelling mistake in README and updated diagrams.
- 40b0a11 (3/28): Updated README with new title for team role, update login/sign up diagrams to better reflect mockups.
- 19baf8a (3/28): Renamed files and updated CSS to be more compact.
- c58b9eb (3/28): Renamed files, disabled autocomplete, and fixed overflow for login/sign up pages.
- 162760e (3/28): Fixed overflow for post creation page.

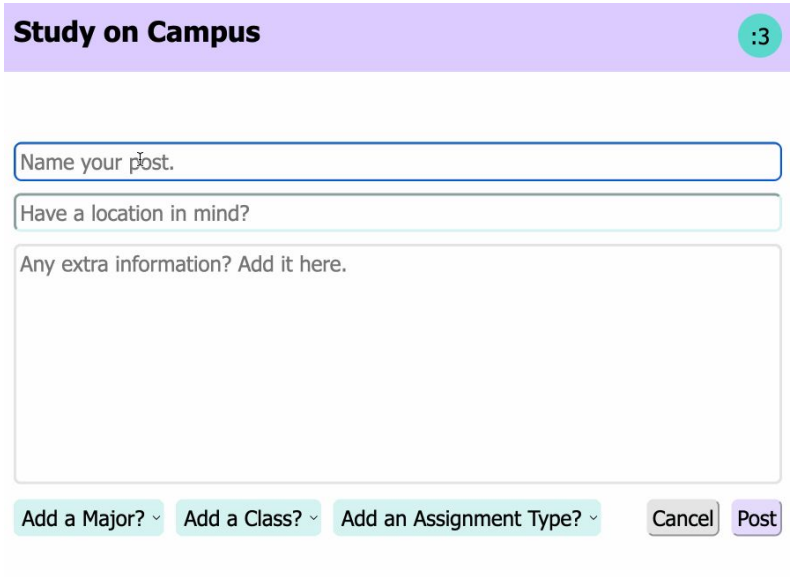
Pull Requests:

- #29 (3/28): Implementation of Post Creation Screen HTML and CSS.
- #30 (3/28): Implementation of Login and Sign Up Screens HTML and CSS.

UI Components: Post Creation Screen



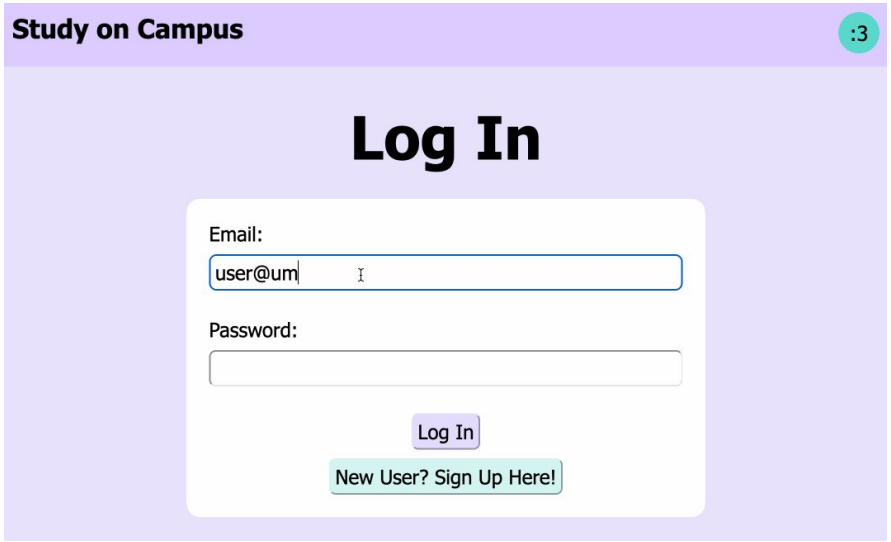
Send Post Use Case



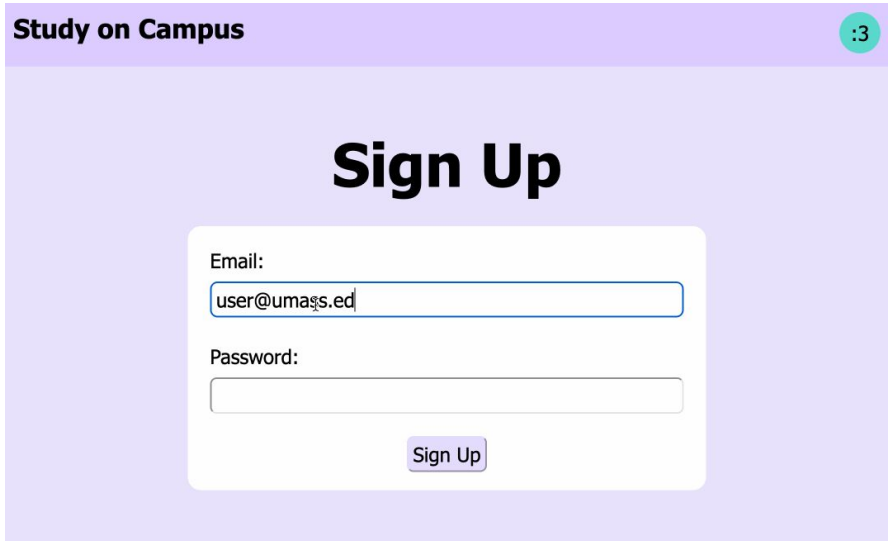
Cancel Post Use Case

In the Post Creation Screen, the user can add a title, location, and additional information for their post that will be visible on the post browse screen. They can add additional tags from dropdown menus, and they can either send their post (which will submit a form to the database) or cancel, which will take them back to post browsing.

UI Components: Login and Sign Up Screens



Log In Use Case



Sign Up Use Case

On the Login Screen, the user can enter the email and password for their associated account and click the Log In button. If their credentials are correct they will be redirected to the Post Browsing Screen. If they are a new user, they can click the New User button, which will redirect them to the Sign Up page. Here, the user can enter an email and password and sign up, which will redirect them to the User Profile page.

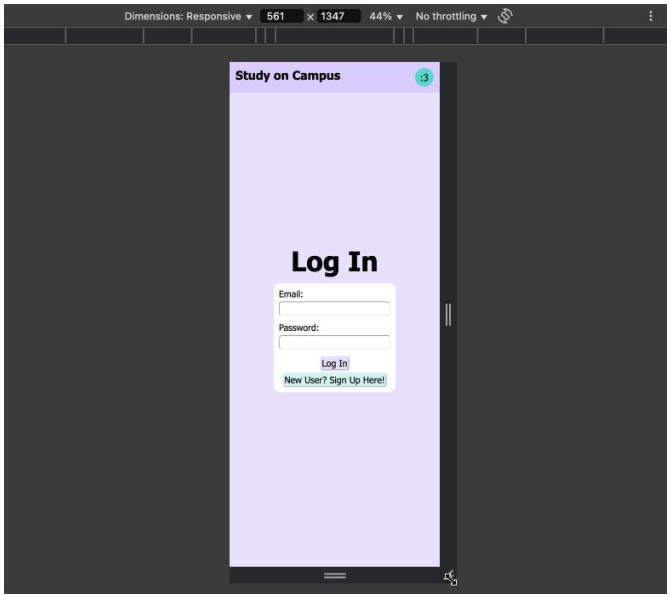
Code

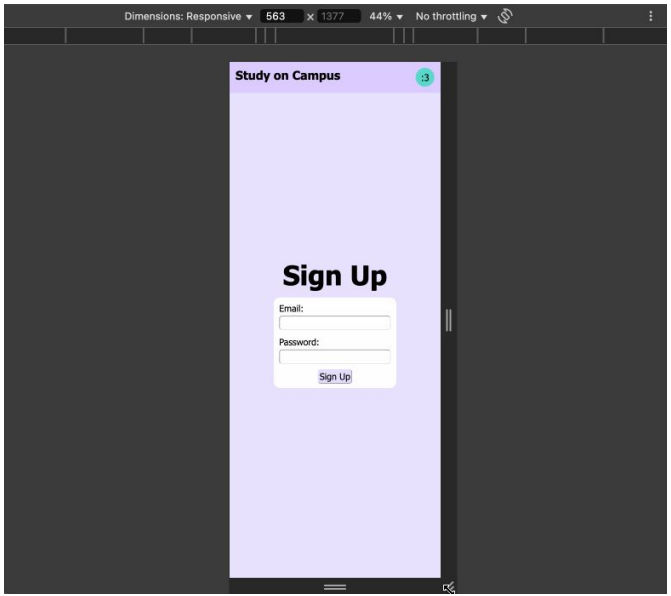
The code on the left comes from the stylesheet for the Login and Sign Up pages. Since the pages are very similar in terms of design, I decided to use one stylesheet for both. The main difference is that the Login page has a new user button and the Sign Up page doesn't, but formatting that is trivial as seen by the code at the bottom of the page.

```
/*******************************
*********** MAIN *************
*******************************/

main {
    display: flex;
    flex-direction: column;
    justify-content: center;
    align-items: center;
    height: calc(100vh - 11vh);
    overflow: hidden;
}

.entry-container{
    width: 60%;
    display: flex;
    flex-direction: column;
    block-size: fit-content;
}

/****** LOG-IN/SIG-UP BOXES *******/

.entry-heading{
    padding: 2vw;
    font-size: 2.25rem;
    text-align: center;
}

.entry-box{
    display: flex;
    flex-direction: column;
    block-size: fit-content;
    margin: 1vw;
    background-color: white;
    border-radius: 1rem;
}

.email, .password {
    display: flex;
    flex-direction: column;
    margin: 2vw;
}
```

```
.email input, .password input {
    padding: 0.5vw;
    border-radius: 0.5rem;
    border-color: #e4e4e4;

}

.email input, .email label, .password input, .password label{
    margin: 0.5vw;
    font-size: 1.5rem;
}

.buttons{
    display: flex;
    flex-direction: column;
    margin: 2vw;
    block-size: fit-content;
    align-items: center;
    cursor: pointer;
}

.new-user, #sign-up-submit, #log-in-submit{
    width: fit-content;
    padding: 0.5vw;
    border-radius: 0.5rem;
    font-size: 1.5rem;
}

.new-user, #log-in-submit{
    margin: 0.5vw;
}
```

Code



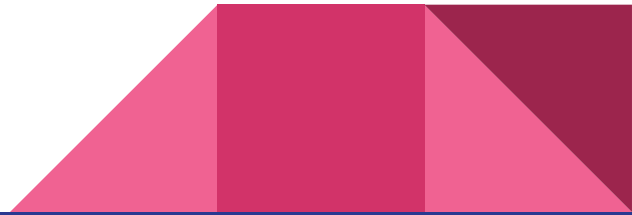


The code from the previous page also makes it so that the Login and Sign Up pages can be viewed from devices with different screen sizes. The white box remains centered in the middle of the page and adjusts size while keeping the text understandable.

Challenges and Insights

A gained a lot of insight into CSS formatting and how to debug as a result of the mockups. I ran into a lot of situations where certain elements weren't positioning correctly or were overlapping. I learned to essentially put borders around everything to debug my code and track how elements were interacting with each other. I also used the Computed tab in Chrome DevTools to get an understanding of how margins and padding was working, as well as the Toggle Device Toolbar to check whether my page would render correctly on different devices.

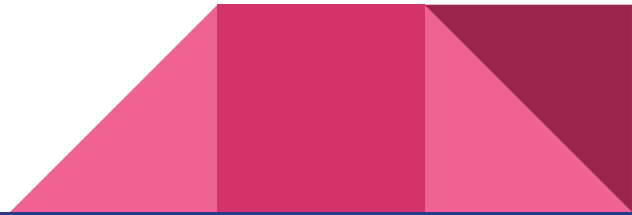
I also gained some insight from my team members on how to implement certain UI features that I wasn't sure about. On the Post Creation page, I wanted to add in placeholder text but wasn't sure how to. Julia let me know that HTML input has a placeholder attribute, which I added to my code.



Future Improvements and Next Steps

#32 **Password Reset Screen(s)**: Currently we have the base Login and Sign Up screens. However, to avoid running into the issue where users forget their passwords and get locked out of their account, we should add a screen where users can reset their passwords. The base use case is that a user presses a link to the password reset screen on the Login page ad they are taken to the Password Reset page. They can then enter their email and receive an authentication code in their mailbox, which they can then use to reset their password. Depending on the implementation, this might be multiple screens.

#35 **UI Standardization**: Currently all of our pages have their own corresponding stylesheets that were made largely independently of each other. We need to work on creating a base “style” for the different pages so that the application looks consistent.



Julia Farber

Work Summary

Issues:

- Time Keeper (Team)
- Post Browsing and Search Screen (UI)
- Post Viewing Screen (UI)
- Post Data Structure (Data)

Commits:

- 342a829 (3/28): Created Post Browsing and Search Screen. Notes and demo.
- 1856706 (3/28): Debugged UI for Post Browsing and Search Screen. Fixed bug of title not hiding overflow and other formatting issues with the title. Created a realistic example and unique post cases.
- 3a0fd32 (3/28): Implemented Post Viewing Screen UI. Notes and demo.

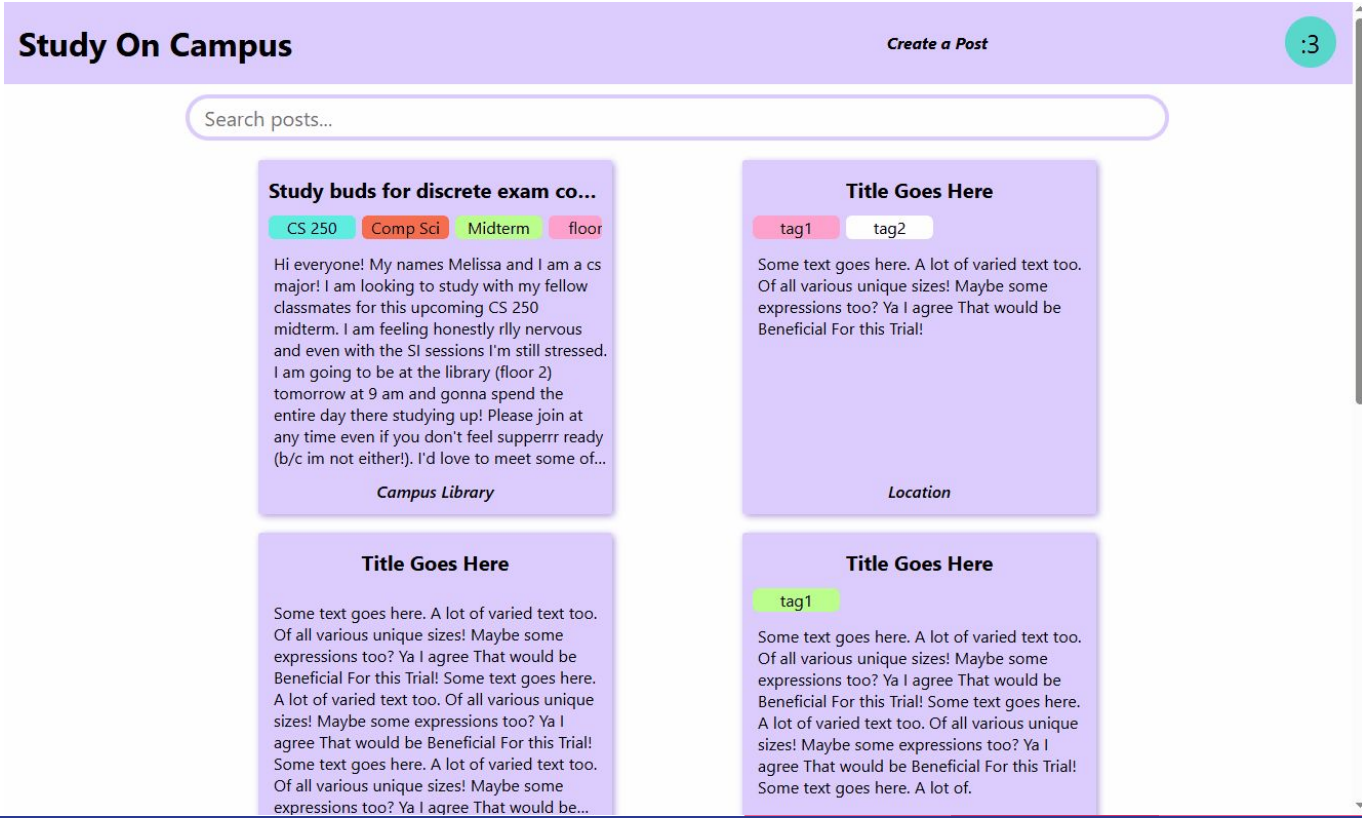
Pull Requests:

- #27 (3/28) Pull request to integrate post viewing HTML & CSS to main branch for tag + submission.
- #28 (3/28) Pull request to integrate post browsing HTML & CSS to main branch for tag + submission.

UI Components: Post Browsing and Search Screen

In the gif, please ignore the blue square background highlight upon clicking certain elements, it's just chrome device toolbar.

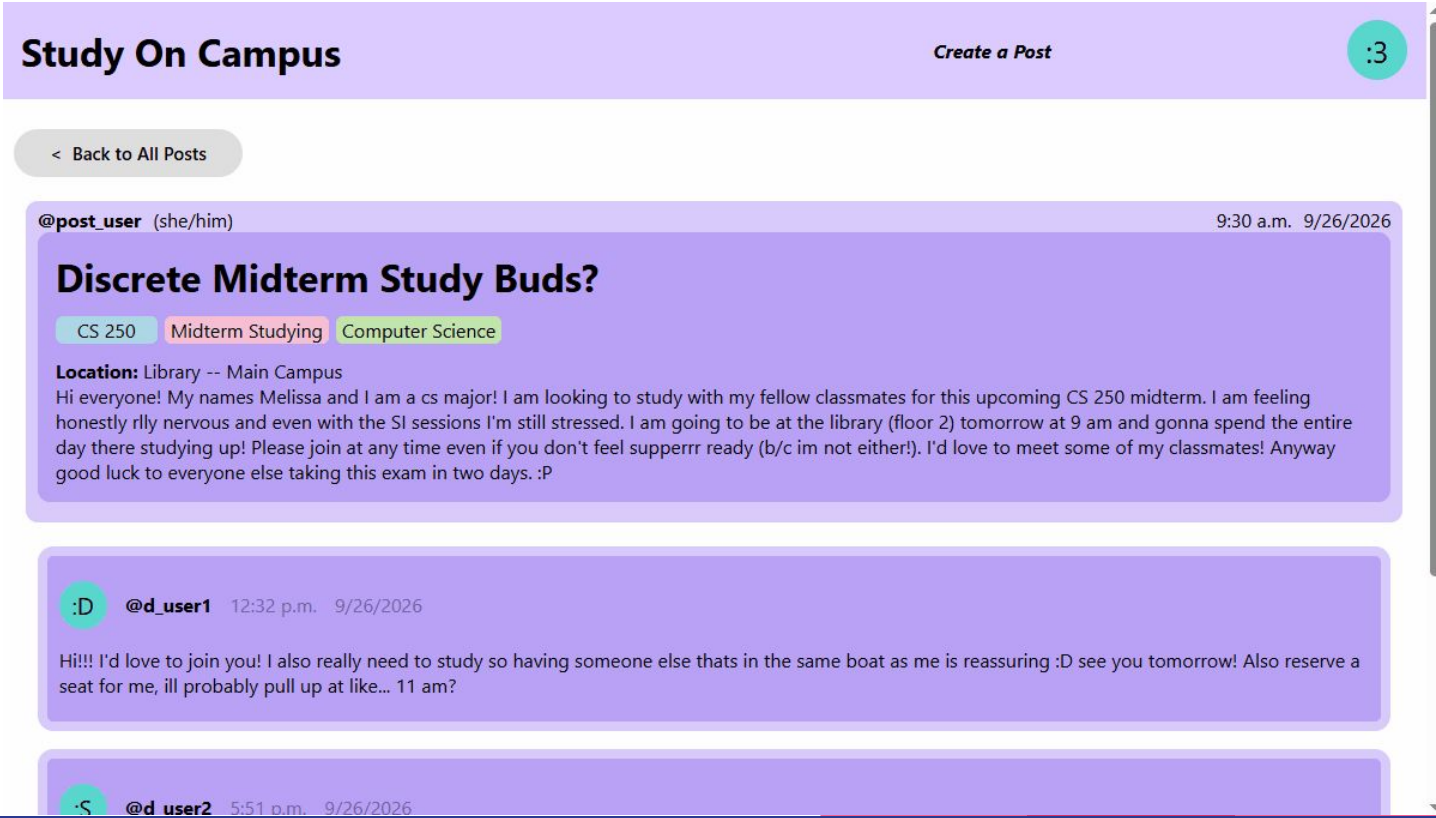
Demonstrates viewing all posts published with the search bar to filter the posts. Shows navigation bar and various resolution experiences. Also shows navigation bar usability.



UI Components: Post Viewing Screen

In the gif, please ignore the blue square background highlight upon clicking certain elements, it's just chrome device toolbar.

Demonstrates viewing a post and the comments, being able to press on navigation bar buttons and usernames and posting a comment. Also demonstrates mobile (responsive dimension) display.



Code

Two key pieces of code that I wrote are in CSS for the tags and the post description on the post browsing page. The same css is used in post viewing screen for tag-container and tag, as it defines the behavior well and ensures wrapping occurs. The two tag css code integrates into the larger UI architecture as it can be reused for any visual representation of the tag data, if tags of any form will be used anywhere else. The challenges faced during implementation of the tags was ensuring the border rounding occurred on the left and right, as without rounding, the overflow would abruptly cut off. Debugging of the format also occurred, as this is my first front-end experience.

Also, the post-text css code is impactful as it can be reused to truncate text in the way we need to manipulate it in the future, it can be reused on the profile page showing snippets of previous posts on the user's profile, and more. This was challenging to implement, as it is my first time learning about webkit box and its use case here to properly truncate text with ellipsis cut off to respect the div size.

Overall, these pieces of css code are good for reusability in other UI's in the application, and took time for me to realize how to get them.

```css
.tag-container {
    display: flex;
    flex-direction: row;
    align-items: center;

    margin: 12px 0px;

    gap: 6px;
    max-width: 100%;
    border-radius: 6px;

    font-size: 1rem;
    flex-wrap: nowrap;
    overflow: hidden;
}

.tag {
    background-color: white;
    text-align: left;
    border-radius: 6px;
    padding: 1px 5px;
    min-width: 75px;
    text-align: center;
    overflow:hidden;
    white-space: nowrap;
}
.post-text {
    padding: 1px 5px;
    font-size: 1rem;
    text-overflow: ellipsis;
    overflow: hidden;
    width:100%;
    align-self: flex-start;

    display: -webkit-box;
    -webkit-line-clamp:10;
    -webkit-box-orient: vertical;
```

Challenges and Insights

As this is my first time working on a project with front-end, there was a bit of a learning curve. However, this learning experience only made me a more efficient programmer and better understand html and css. With each interface element I implemented, my pace increased and I became more efficient. I got to experiment and debug my interface and figure out how to get the interface to behave how I wanted it. It was challenging, yet fun to figure out what was causing bugs. Some examples of these were, ensuring the formatting of the posts had a sticky note like format and to ensure that the location was always at the bottom not just under the text, how to get the post description to cut off with a '...' at the end, and ensuring that the title started in the middle of the post, but if it got too long, it cut off to the right and did not cut off on both sides. Ensuring the post looked and behaved exactly the way it was supposed is critical, and I am glad I spent the time to ensure that no bugs will be faced down the line due to formatting issues. I improved at manipulating flexbox and @media to super small resolutions to tweak the design to what I wanted. I also learned that I should use the @media not to cater towards popular smartphone resolutions, but rather to what my design needs and benefits from. Working with my teams also made me realize how critical team input is, as sometimes as developers we might be blinded by our own perspectives on designs or how functions should work. Specifically, they helped me realize that there should be a search bar, not an advanced filter search, and gave UI post format suggestions.

Future Improvements and Next Steps

- #24 is about User Profile Icon Generation. The team has discussed how to ensure that comments are better distinguishable from one another and what icons that the users should have, since the team has agreed to do something besides profile pictures that the user chooses. Colorful background color assigned at account creation and profile picture mimicking ASCII emoticons (like :D :P ;3 :U) through a colon or semicolon paired with the first letter of the user's username/display name. This will create fun customization, while allowing users to better distinguish themselves in comments and profiles. A familiar, yet fun spin on default icons across the internet.
- #31 is about Post Browsing Search Filtering. It is integral to the post browsing page to have a post search filtering. With each word/keystroke in the search bar, the posts would filter. This heavily depends on JavaScript and DOM manipulation. Also, it is a way for users to find relevant posts easier, rather than searching through all of them by scrolling. A search button might also be beneficial to confuse users less.
- Optimize further: test the wrapping of the location before proceeding to javascript to ensure long locations work to avoid bugs in the future.

Ashley Kang

Work Summary

Issues:

- Settings Screen Page (UI)
- Course Catalog Data Structure (Data)
- Quality Control (Team)

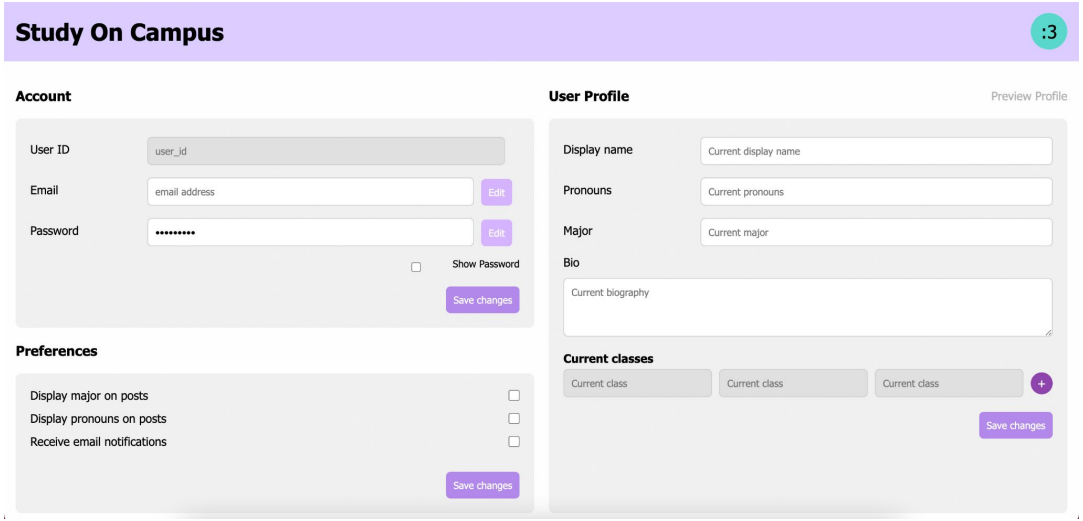
Commits:

- 44a1221 (3/28) Implemented the HTML and CSS for the settings screen page

Pull Requests:

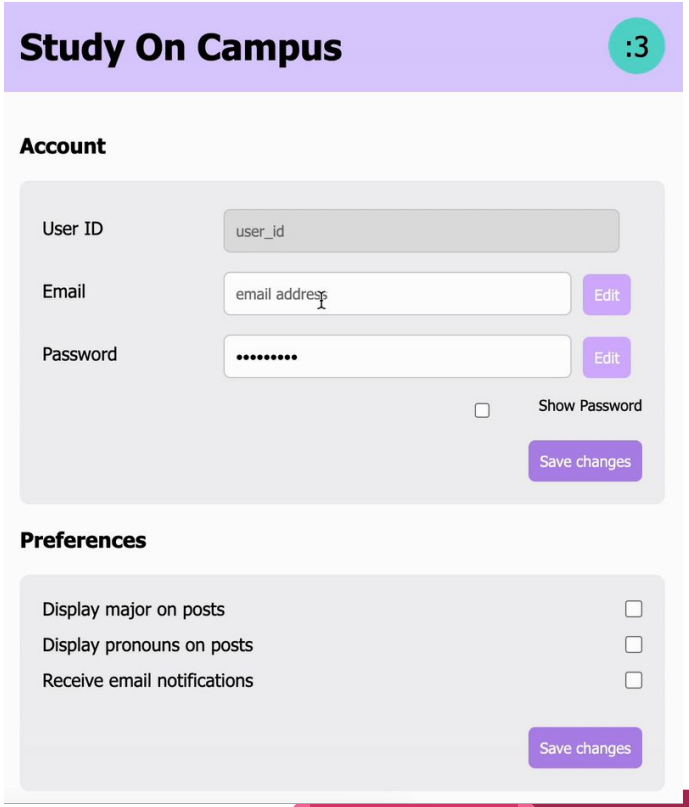
- #33 (3/28) Implemented the HTML and CSS for the settings screen page

UI Components



Desktop view

Users can make any changes to their account, profile, or preferences in this page. They can also preview their profile before saving any changes. The interface works with both desktop and mobile screens.



Mobile View with demonstration

Code

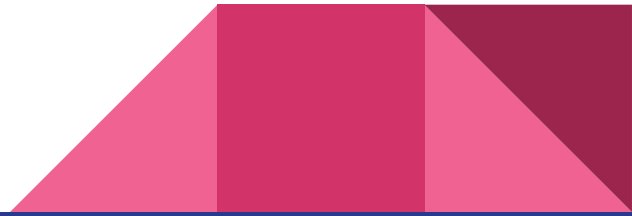
```html
<section class="account-section">
    <label>User ID <input type="user_id" placeholder="user_id" disabled /></label>
    <label>Email <input type="email" placeholder="email address" /><button class="edit-btn">Edit</button></label>
    <label>Password <input type="password" value="123456789" /><button class="edit-btn">Edit</button></label>
    <div class="show-password-container">
        <input type="checkbox" onclick="myFunction()">
        <small>Show Password</small>
    </div>
    <button class="save-btn">Save changes</button>
</section>
```

Some key aspects of the HTML and CSS for the settings page screen were the edit buttons and the show password checkbox in the account section. The edit buttons allow users to switch between editing and viewing their email or password. They are also responsive when hovered over and change appearance. The show password checkbox allows users to see their password if needed.

```css
.account-section .edit-btn {
    margin-left: auto;
    width: fit-content;
}
.show-password-container {
    display: inline-flex;
    justify-content: flex-end;
    align-items: center;
    width: 100%;
}
.edit-btn {
    background-color: #d6b3ff;
    color: white;
    border: none;
    cursor: pointer;
    width: fit-content;
}

.edit-btn:hover {
    background-color: #b388eb;
    cursor: pointer;
}
```

Challenges and Insights

Given that this was my first time working with HTML and CSS on a project, there was a learning curve in order to implement the design based on the diagram. There were issues with creating the different sections within the settings page and making sure everything was aligned properly. After completing the implementation, I became better at organizing the layout of a website and more comfortable with using CSS to create interactive features and handle various screen sizes.



Future Improvements and Next Steps

- #20 **Course Catalog:** In order to avoid multiple instances of the same class (but with varying spellings), the course catalog would allow users to choose the course department (eg. CS, CICS) and input the number of the course. This would create a tag that would allow users to put on their profiles and use to browse posts.
- Currently, there is no way to add or remove classes from one's profile. This requires both CSS, HTML, and JavaScript to implement
- Profile Preview: This would also require both frontend and backend to implement as we want users to have the ability to see their changed profile without changing their profile.