Programming Usable Interfaces
Final Project Write-Up

Nicolas Ettlin • nettlin

Part 1: Website Description

The goal of this project is to provide an interactive map of the CMU campus, including an indoor view of the buildings.

Before this project, there was no simple way to locate a room on campus or know the layout of the rooms in a building. CMU Map straightforwardly provides this information.

This makes this tool much more useful than the existing alternatives, such as the official CMU map (which doesn't feature indoor maps), and the CMU floor plans (which are difficult to browse).

The target audience is the people who study and work at CMU.

Part 2: Interactions

- **Browse the map**
  When they open the website (**cmumap.com**), users see a map of the campus. They can pan and zoom the map as on other map services such as Google Maps, by using the scroll wheel (or a pinch zoom on a trackpad) and drag/drop.
  When they zoom in on a specific building, the floor plan appears. By zooming in more, the room numbers appear as well.
  Example: zoom in on Wean Hall until the floor plan appears, then zoom in more to see the room numbers.

- **Select a different floor**
  When a floor plan appears, users can use the floor switcher component that appears at the bottom of the screen to change the floor that is currently being displayed. They can either go one floor above or below, or select a particular floor by using the menu that appears when tapping the current floor number.
  Example: when a floor plan is visible, press the ▼ and ▲ buttons on the box that appears at the bottom of the screen. Press the current floor number to open the quick switcher.
- **Search for a place**
  Users can search for a particular building or room by using the search box.
  Example: try the following search queries in the search box: "gates", "ghc", "ghc5222", "la prima". Press the desired building or room in the search results to display it on the map.

Part 3: External Tools

React

- **What?** React is a popular library used to write front-end JavaScript code. It allows developers to define reusable interface components that will automatically update when the application state changes.
- **Why?** I chose to use React because it is very helpful to use a library of its kind when writing complex apps.
- **How?** My project is based on React, and uses many of its features such as components and hooks.

Next.js

- **What?** Next.js is a front-end framework based on React. It provides a structure for the application's code and many features.
- **Why?** I chose to use Next.js because it is commonly used by React developers.
- **How?** My project is based on the default Next.js project.

MapKit JS

- **What?** MapKit JS is the library published by Apple that allows third-party developers to integrate Apple Maps on their web pages.
- **Why?** Apple Maps provides beautiful client-side-rendered maps and offer the customization options I needed.
- **How?** The map displayed on CMU Map uses Apple Maps to display the map background and render the custom elements (building and room shapes, building names, markers...).

mapkit-react

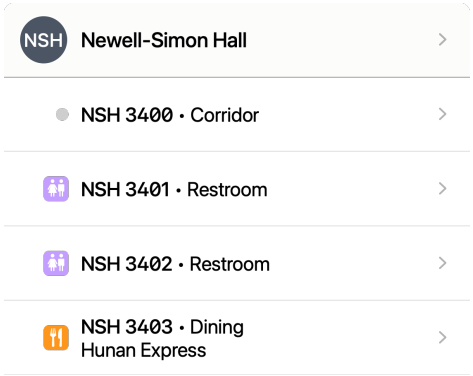
- **What?** mapkit-react is a wrapper that allows to use MapKit JS objects as React components.
- **Why?** When working on another project that uses React (CMUEats), I had to add a map but couldn't find a wrapper for MapKit JS that could do what I needed. So, I decided to develop my own. I extended it later to support the features that CMU Map needs (e.g. polygons and custom annotations).
- **How?** CMU Map uses the API provided by mapkit-react to display the map and its contents.

Part 4: Iterations

My design slightly evolved from the original mockups. The main changes are:

- The quick floor switch was inlined in the current floor box, instead of being a dropdown menu. This saves space on the screen and is easier to use, as it is easier to reach controls at the bottom of the screen on mobile phones. The ellipsis on its button was also changed so that it indicates the number of floors available and the position of the current one.
- The design of the search results evolved so that it also features the room type and its marker icon.
- I didn't implement the room details view to have more time to focus on the other parts. This view could be more useful later when more features will be added, but is not strictly necessary for now.

The new floor switcher design.



The new search results design.

Part 5: Challenges

The main challenge I had was to get usable floor plan data. I had to find a way to the locations and shapes of the CMU campus rooms in a machine-readable format. This was done by writing a program that reads the Autodesk floor plan files published by CMU and identifies room shapes and room names. This was challenging, as the files are not structured in a way that makes them easily parsable.

Part 5: Screen sizes

CMU Map is fully responsive. Some of the screen sizes that I tested the project with are mobile (414 × 736 pixels), tablet (1024 × 768 pixels) and desktop (1440 × 900 pixels).

Part 6: Accessibility