Ryan Rosenblatt, Matthew Jones, and Aryan Nangia

Ryan Rosenblatt, Matthew Jones, and Aryan Nangia
May 6th, 2025
CSC 211

Building Buddy - Final Report

Member Name	Member's Contribution
Ryan Rosenblatt	Final Presentation Slides, Final Report, All React Native Code, Managing Version Control, Website Hosting, Website Design, JSON Files, Floor Plan Buddy (separate app just to write JSON files), Project Ideation, Flowchart, Wireframe, Interviews
Matthew Jones	Final Presentation Slides, Final Report, Project Ideation, Storyboard, Wireframe, Flowchart, Demo Video
Aryan Nangia	Final Presentation Slides, Final Report, Writing Evaluation Form, Evaluation Results

Introduction

Interior building design is often not intuitive. Large buildings can be complex and challenging to navigate. It is a common experience to be in a new location and be confused about which room is where. This problem is especially prevalent among college students. There are dozens of unique buildings designed at different times. The buildings lack consistency in design. When lost, asking for directions can be embarrassing, especially in crowded or fast-moving places. We are all human and are prone to forgetting things from time to time, even if they were told to us 30 seconds ago. People need a better way to find obscure rooms, one that is accessible and minimizes the risk of frustration and confusion caused by human error/miscommunication.

The existing technologies either fall short or require additional technology. There are solutions like Google Maps and GPS designed to function outdoors; they have limited to no functionality indoors and require a sight line to satellites in MEO (Medium Earth Orbit) to function. There are also a variety of technologies that rely on Bluetooth, WiFi, and QR Codes. Software and hardware systems like Cisco Spaces can be prohibitively expensive or difficult for buildings to implement. There are multiple physical solutions, but they are also problem-prone. For example, paper maps cannot give real-time updates or be updated when room numbers change. There is also physical signage, which can become outdated, confusing, or put up in poor locations. Most of the time, physical signage will not be able to direct you to a specific room.

Our proposed solution uses 2-D and 3-D maps to help the user find their destination. Building Buddy would have a regular navigation feature, like Google Maps, and an indoor navigation system that uses door-to-door pathfinding. A user would first enter the building they're closest to and their destination building, using a map of their current location via their phone's GPS. The user would then input their destination building. The app will give them GPS directions to their destination. The app would remember which entrance it led the user to as well. Once the user has entered the building, they'll be prompted to enter the entrance/room they are closest to and their goal room. The app then shows a model and path from the nearest entrance to the destination room. You would get a level-by-level model with a line to show the correct direction. Our app would be akin to an indoor version of GIS. If the user gets lost, they select which room they see closest to them, and the app would pathfind again. The app would minimize the amount of time users spend being lost and unable to find their destination. We would decrease confusion and travel time, and have our users wayfinding with more confidence.

User requirement gathering and analysis

Interview
Respondents included 16 students from all four years and alumni of the University of Rochester. Most enrolled students were in the Hajim, while the rest were in AS&E.

The following questions were asked:
Have you ever gotten lost finding a room on campus?
75% of respondents said Yes

Do you find campus maps to be a valuable resource for finding buildings?
58% of respondents did not find campus maps useful

How do you determine where a room in a building is? (Short response summarized)
56% use brute force/guessing
44% try using room numbers to recognize patterns

Would you use an app with 3-D models and navigation directions?
83% of respondents said Yes

User Analysis

Affinity Diagram: The affinity diagram addresses the five major concerns from our survey. People unfamiliar with the campus and the building, either from being guests or new students often get lost. Many buildings have confusing and unintuitive layouts with rooms being difficult to find and hallways difficult to navigate. There are numerous limitations to existing solutions, including unreliable GPS, a lack of signage, and updates not being readily available. People face difficulty finding elevators, vague signage, and crowded hallways. Existing solutions remain

inadequate as hallways look similar, navigation tools are inconvenient, and campus maps lack adequate details.

Difficulty Finding Specific Rooms	**Technical Limitations**	**Lack of Real-Time Updates**	**Inefficient Routes**	**Current solutions not helpful**
New students struggle to locate classrooms	GPS is unreliable and doesn't work well indoors	Room changes aren't reflected immediately	No clear indication of the fastest route	Easy to get lost since many hallways look similar
Confusing building layouts with multiple floors	No standardised indoor positioning (WiFi, beacons)	Unexpected closures or restricted areas are unknown	Difficulty finding the elevator	Guessing, memorising, relying on patterns are inconvenient and a compromise
Room numbers are not always sequential	Takes up a lot of battery for continuous tracking	No alerts for construction or temporary detours	Can't avoid crowded hallways	Campus internal maps are not detailed enough

Persona #1:

Name: Jeff

Background:
Age: 22
Occupation: New Student
Education: Undergraduate in Computer Science
Location: UofR
Personality Traits: Anxious, punctual, self-conscious, determined, tech-savvy

User Goals:
Jeff wants to quickly find his classrooms without stress, especially in unfamiliar buildings. He hopes to avoid being late and looking lost in front of his classmates.

Narrative story:
Jeff is running late for his HCI class and only has 3 minutes left. As he is a new student, Jeff is lost in Dewey Hall and doesn't know which direction to take. He looks around, hoping to find a sign or someone to ask but everyone is too busy. Every hallway looks identical and when he pulls out his phone, the map is taking forever to load.

Roadblocks:
- The room numbers don't follow a logical pattern (in the 1000's).
- does not know any shortcuts
- He thinks asking someone might waste his time
- No clear signs or maps
- Current location is not stable

Further Information
Will it work offline in case of poor signal inside buildings?
Are the directions up-to-date?
What if the class is on a different floor?
Can I find rooms by my course name?

Persona #2:

User Goals:
- Quickly find places around campus
- Discover shortcuts and less crowded paths to avoid being late
- Get real-time location updates
- Avoid running late

Roadblocks:
- Poorly labeled buildings and confusing campus layout
- The campus map is outdated or not user-friendly
- Hesitation to ask for directions from strangers
- Getting lost in large buildings with multiple floors

Name: Alex

Background:
Age: 19
Occupation: Sophomore
Education: Undergraduate in Finance
Location: UofR
Personality Traits: Curious, Risk-taker, independent, Self-aware

Narrative story:
Alex is rushing to his first class of the semester but is lost. He thought he had plenty of time but took a wrong turn, ending up in an unfamiliar hallway with no clear signs. Checking his phone, he tries to use the campus map, but it's just a PDF and not interactive. Alex doesn't want to ask a stranger for help, so he frantically searches for directions online, wasting even more time.

Further Information
- Can the app integrate with my class schedule?
- Will it give the fastest route or shortest route?
- What if GPS accuracy is poor in areas with bad signal?
- Can I plan routes in advance?

Volere Shell 1
Description: The app will help university students quickly navigate complex indoor buildings using real-time maps and directions.

Rationale: To reduce students' stress and time spent navigating large, complex university buildings, enhancing their overall campus experience.

Source: Surveyed Students.

Fit Criterion: The app will provide accurate indoor navigation by loading directions in less than 5 seconds for 95% of users.

Volere Shell 2
Description: The app should be able to clearly and accurately display the route on a 3-D model of the building.

Rationale: To make it easier for the user to find their way around the building, especially in more complex ones.

Source: Surveyed Students.

Fit Criterion: The user should be able to visualize their route both before and while navigating to decrease their likelihood of getting lost along their route.

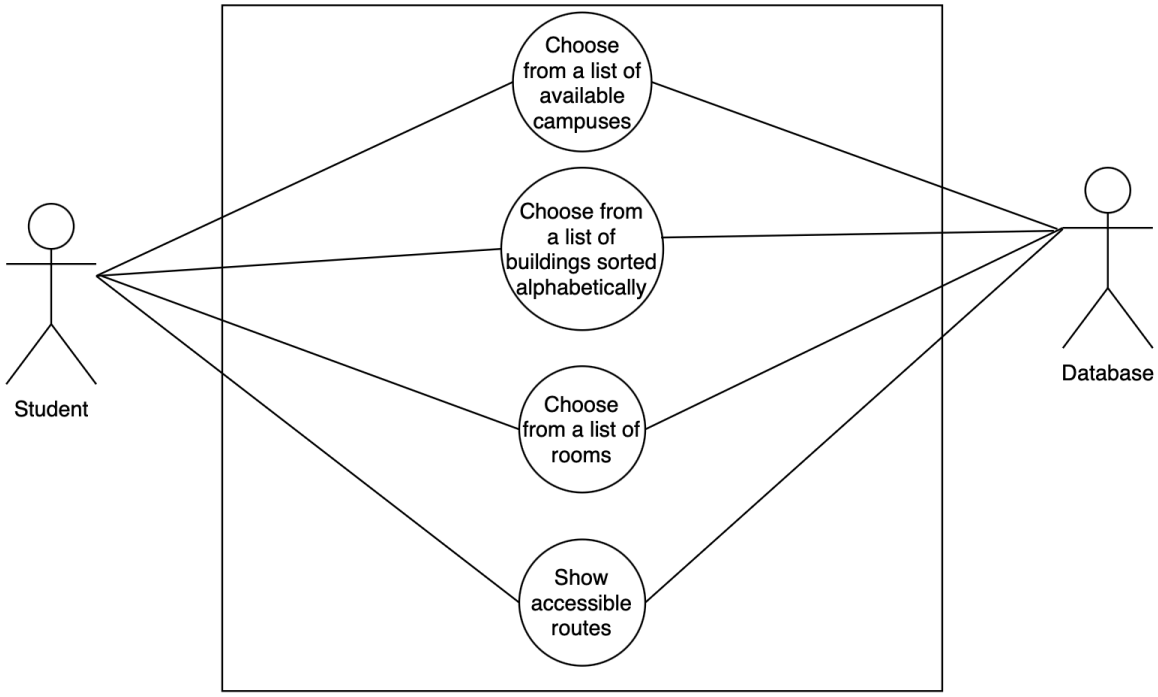
Volere Shell 3
Description: The app will provide an option to adapt the route it calculates for the user based on their accessibility needs.
Rationale: To make the app accessible to users of all abilities and direct them on a route in a way

that they can comfortably follow, given their needs.

Source: Surveyed Students.

Fit Criterion: The user should be routed on accessible routes through the building, avoiding stairs, and using elevators, ramps, door buttons, automatic doors, etc., as necessary.

Use Case Diagram



Use Case 1: Choose from a list of available campuses.

1. The app displays a list of campuses that are available to navigate.
2. The user selects a campus from the list as the target.
3. The app then pulls up the list of buildings to navigate to and from.

Use Case 2: Choose from a list of buildings sorted alphabetically.

1. The app displays a list of buildings on campus.
2. The user selects a building from the list as the target.
3. They then select the building they are nearest to; if they cannot determine that, then the app must use the location from the phone to determine the structure.
4. The app fetches the location of the selected building.
5. The app shows a button to get directions to the target building.
6. The app gives an option if they are lost, which brings them to step 3.

Use Case 3: Choose from a list of rooms.

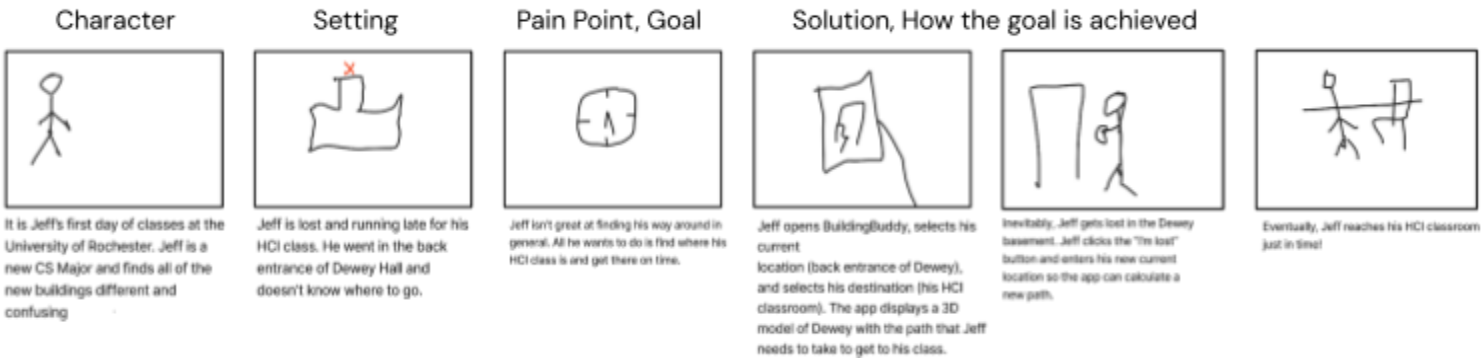
1. The app fetches rooms in the current building and displays them as a list.
2. The user selects a room from the list as the target destination.
3. The user selects which entrance or room they are closest to.
4. The app shows a button to get directions to the selected room.
5. The app gives an option if they are lost, which brings them to step 3.

Use Case 4: Show accessible routes.

1. The user selects an accessible route to avoid stairs.
2. The app fetches locations of elevators, ramps, etc..
3. The app calculates a new route designed for accessibility.
4. The app displays the new accessible route on the map.
5. The app follows steps 3-5 from Use Case 3, otherwise.

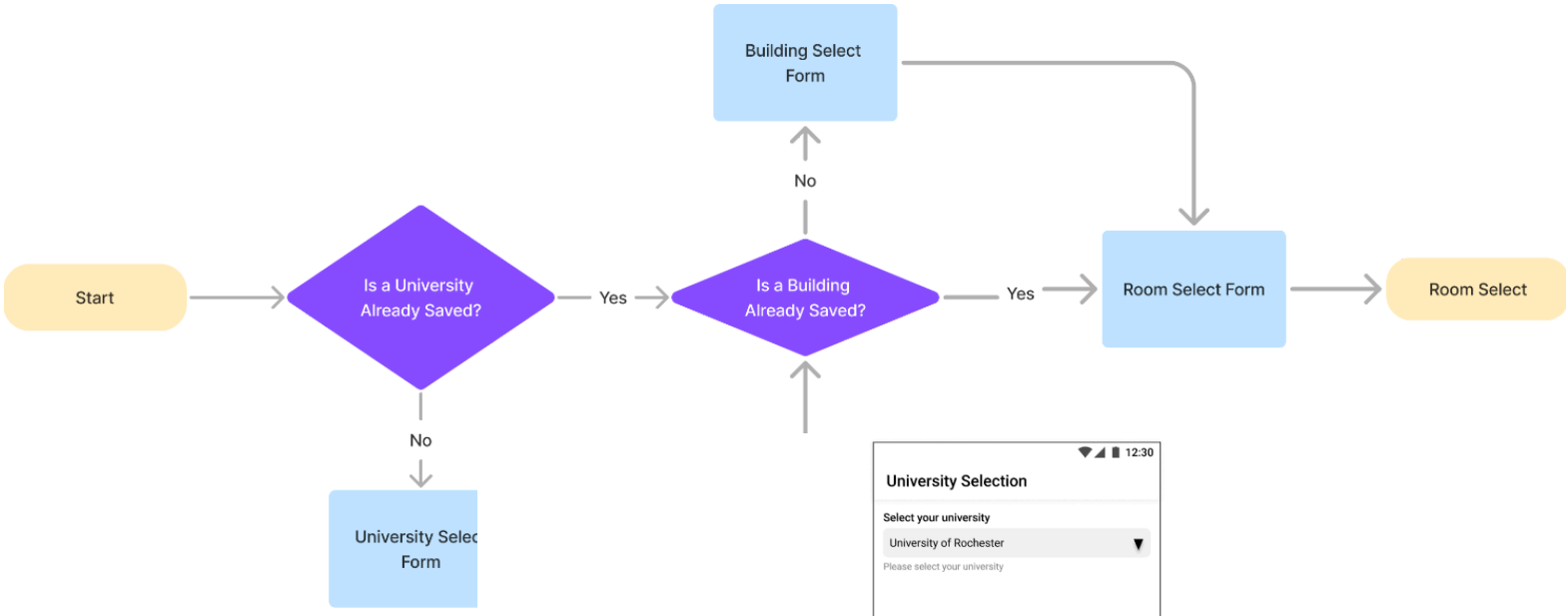
Low-fidelity Prototyping

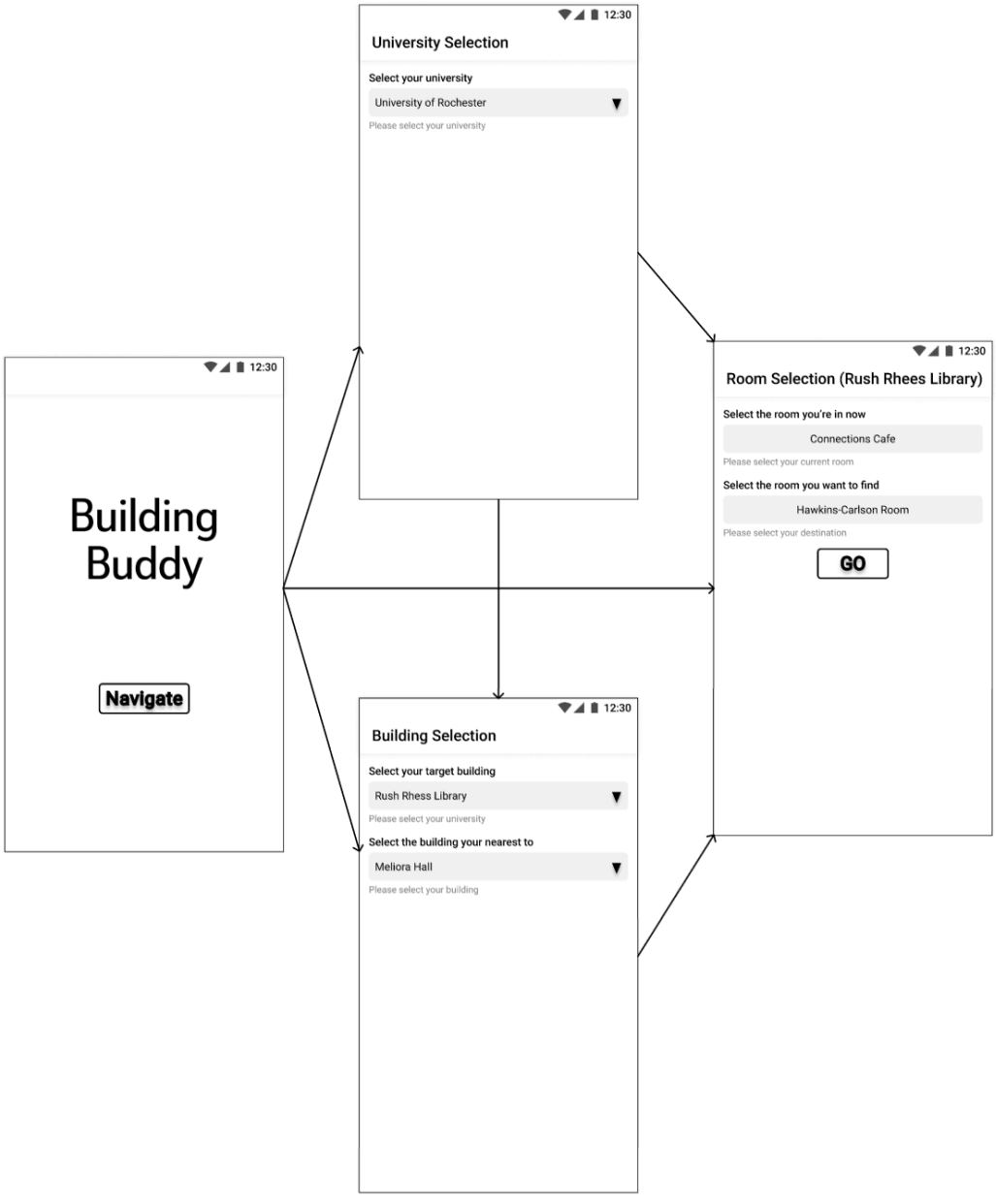
Storyboard



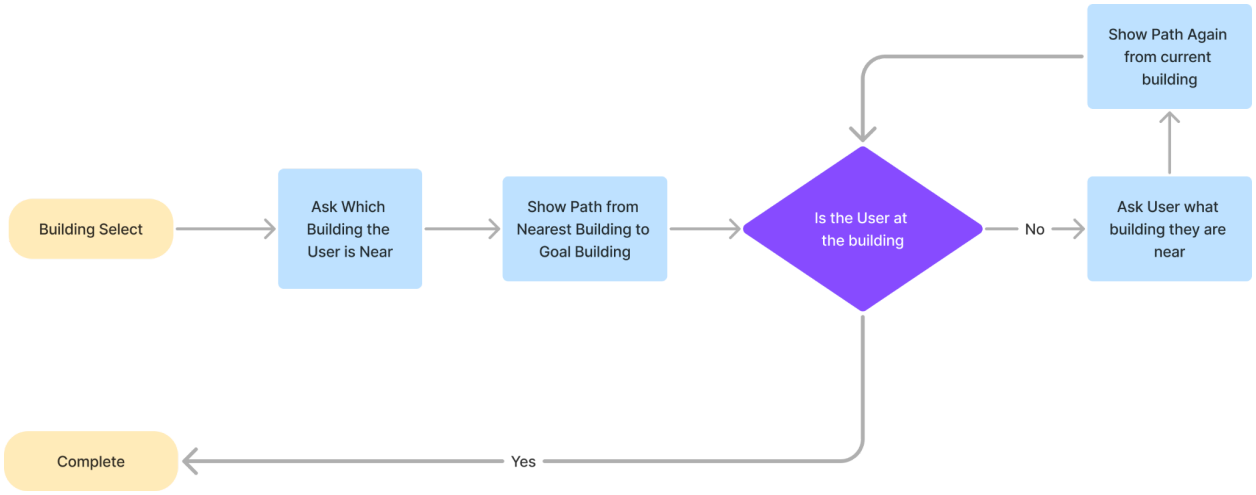
User Flow

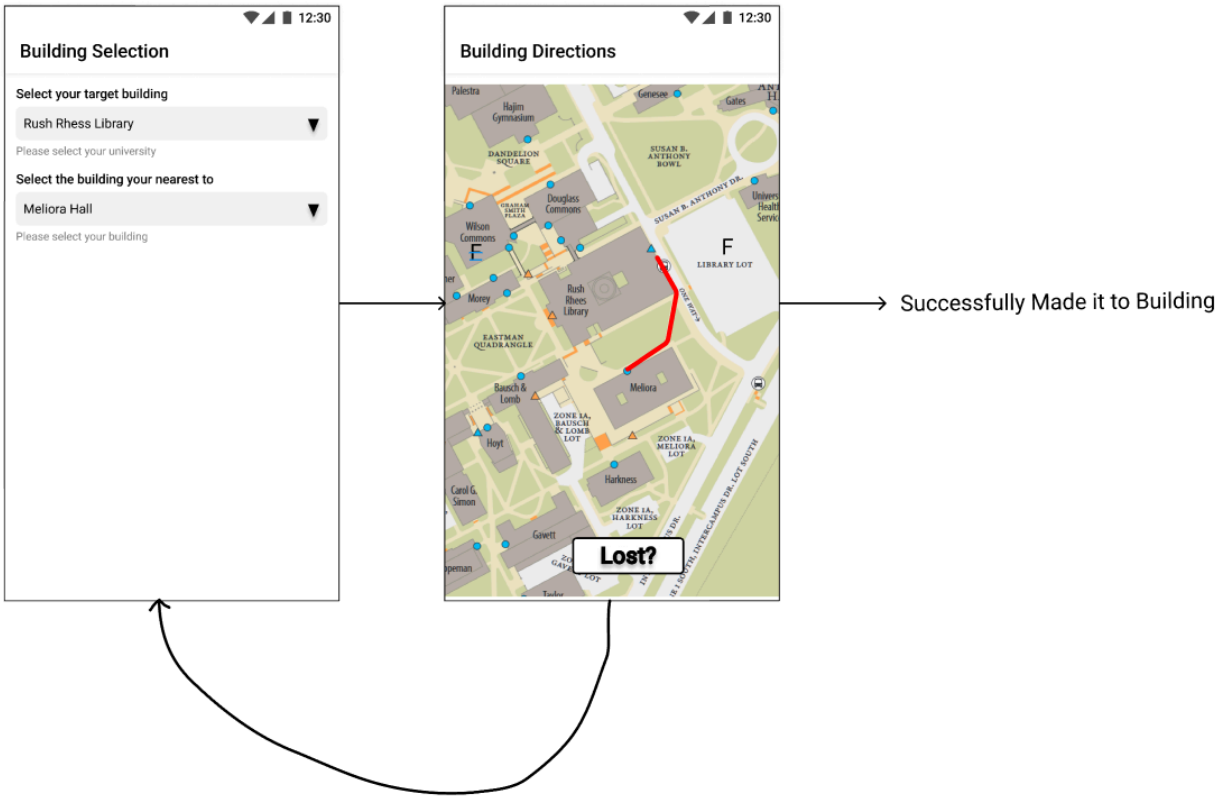
Use Case 1: Show directions from the current location to a desired room.



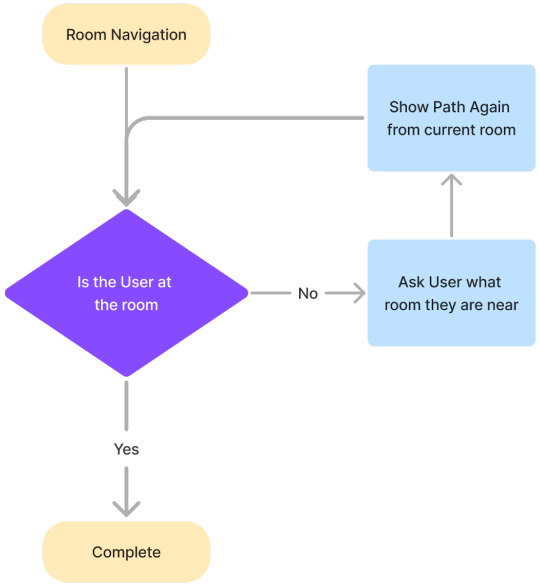


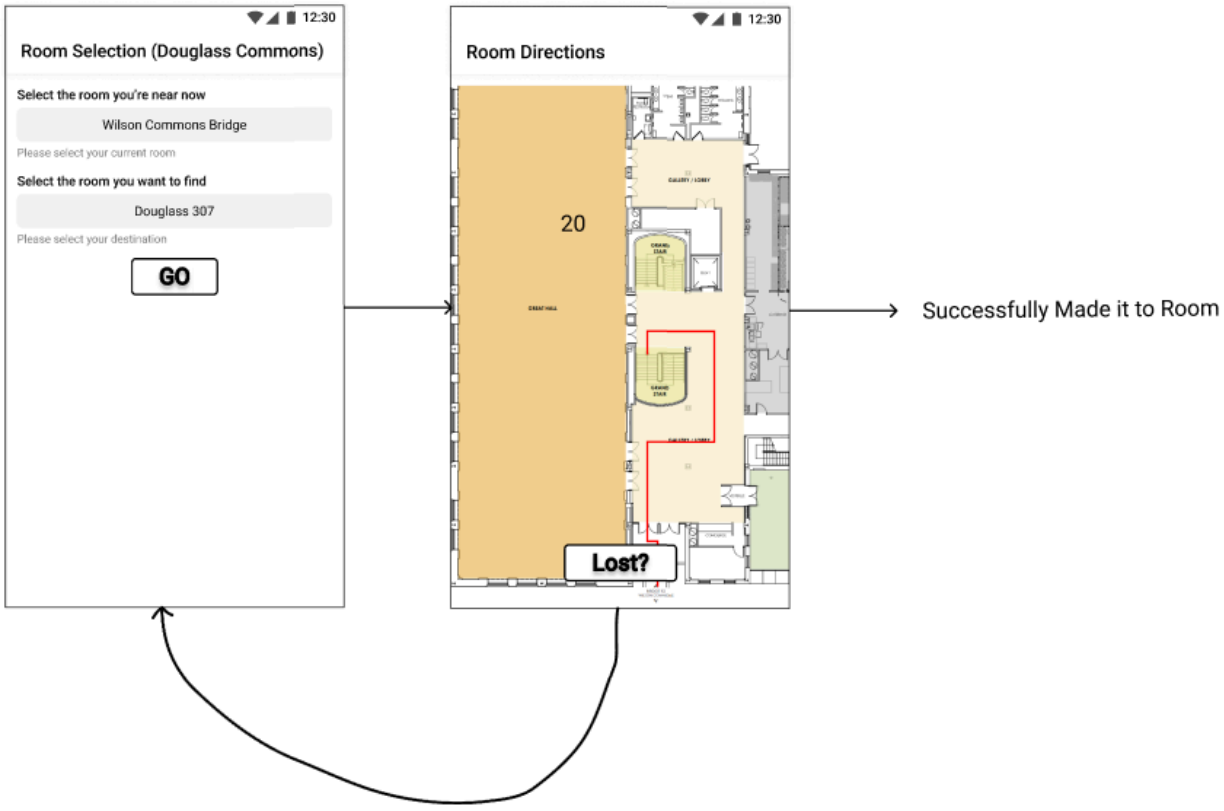
Use Case 2: Set directions from the nearest building to the target building and show the path.



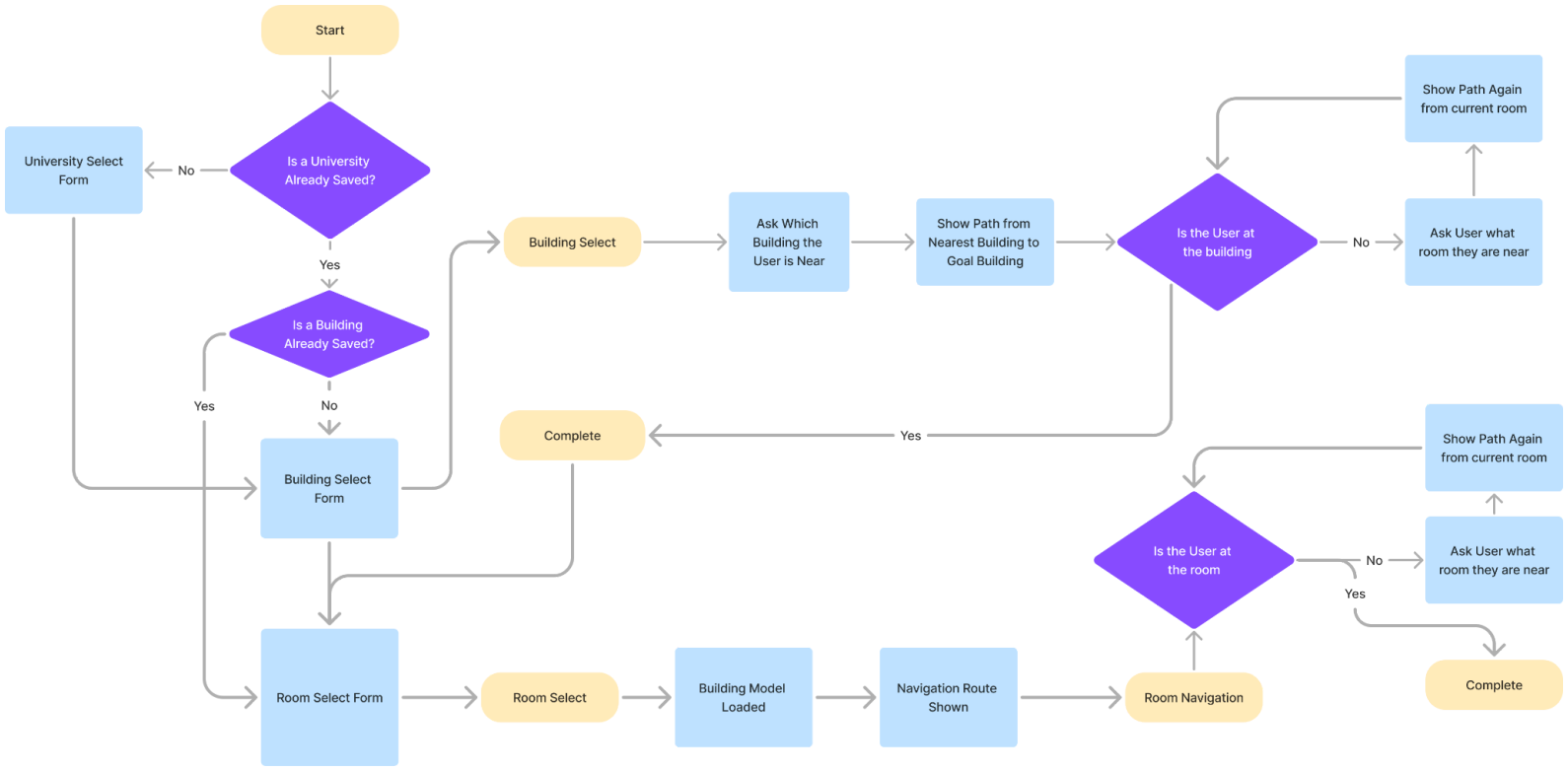


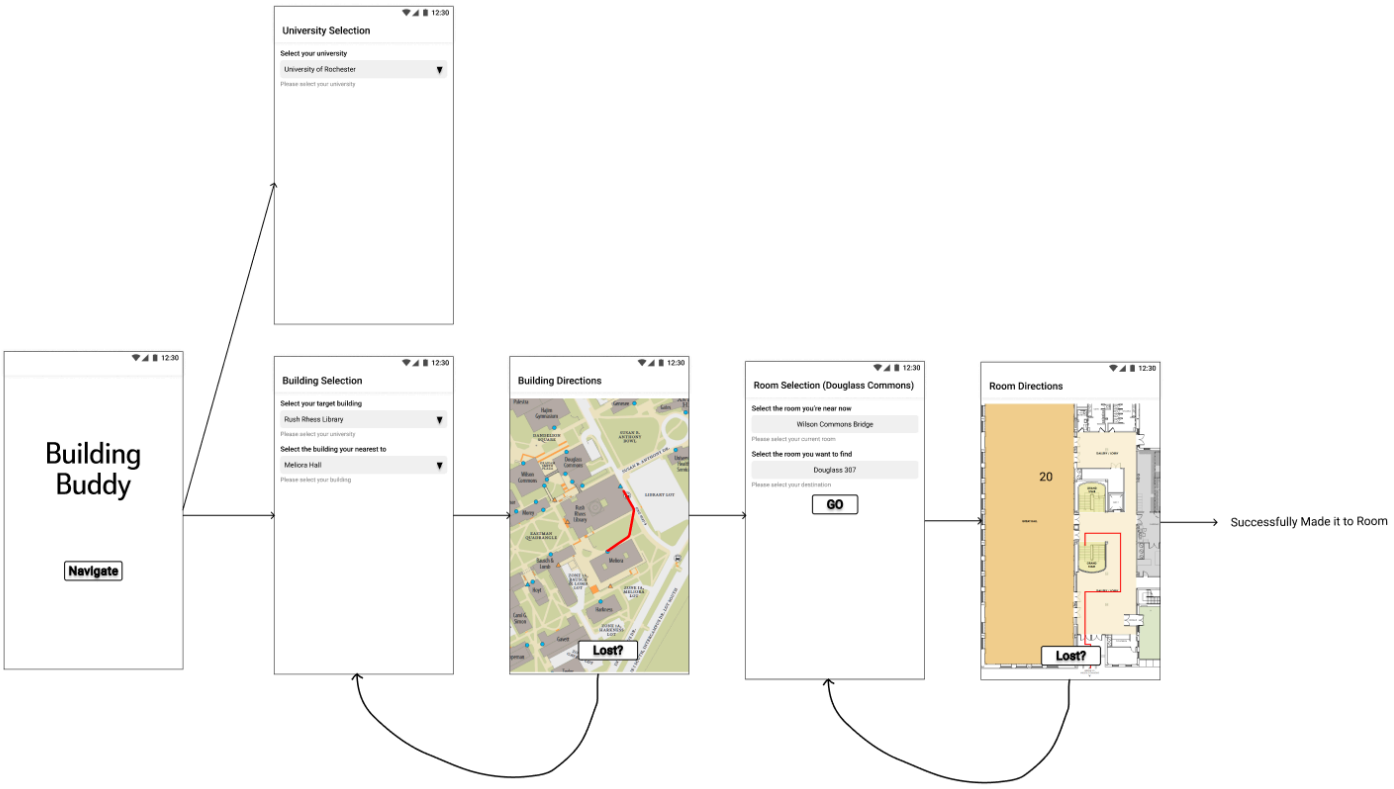
Use Case 3: Update map directions/current location based on which room is nearest.





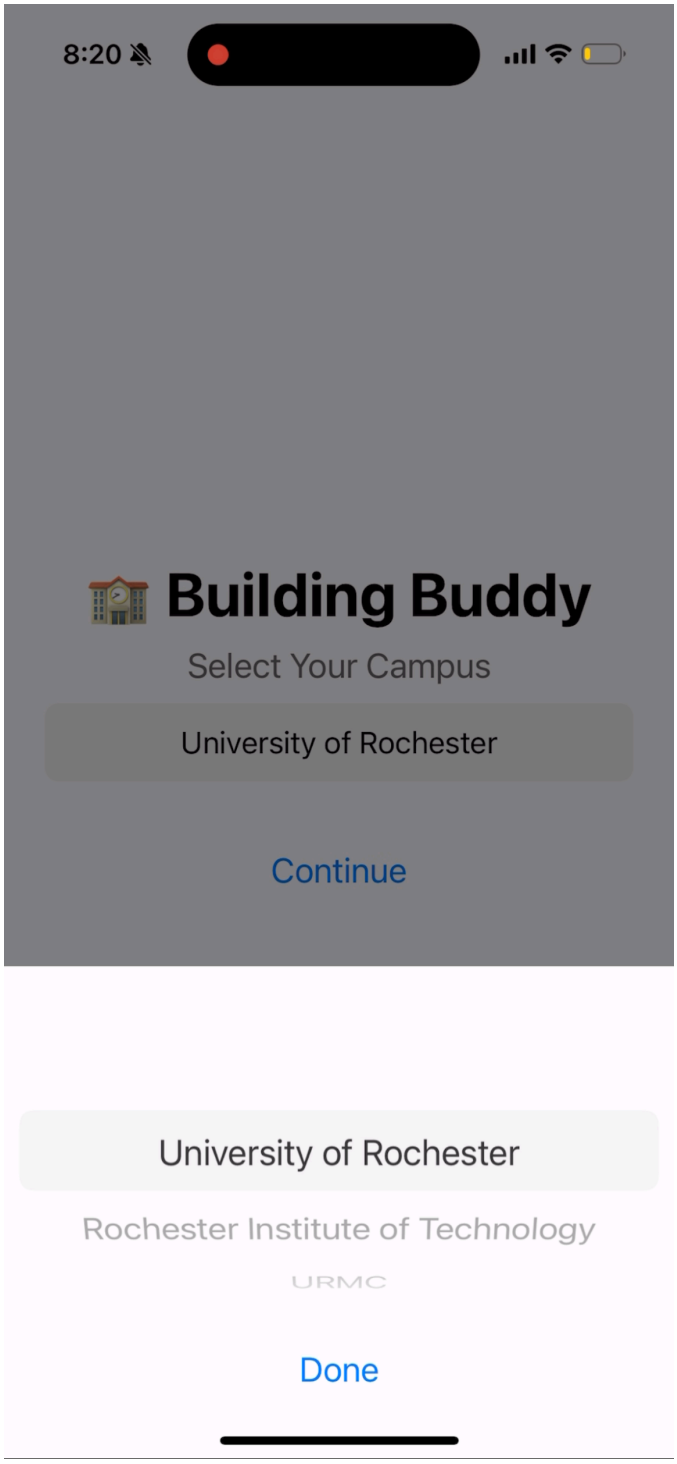
Complete Use Case:



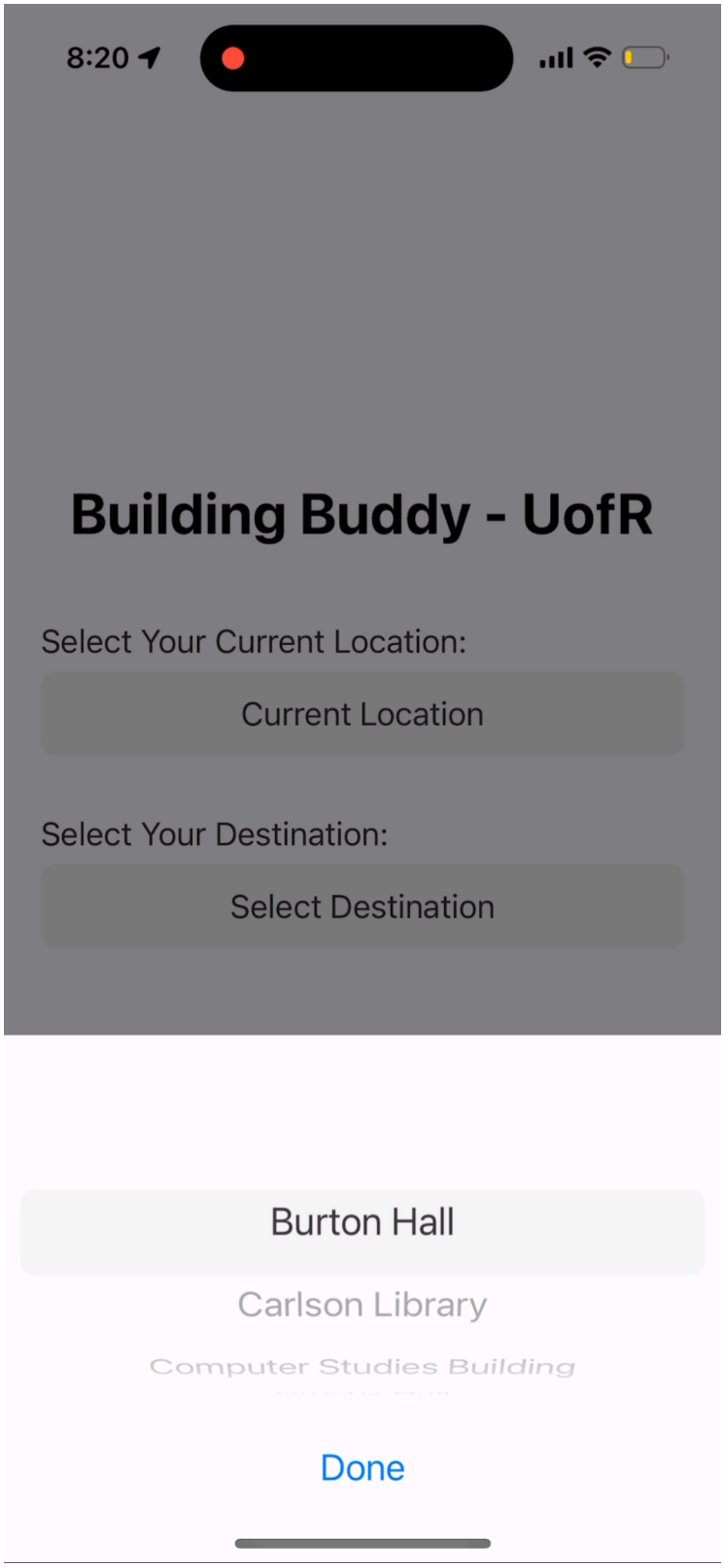


Working Prototype and Implementation

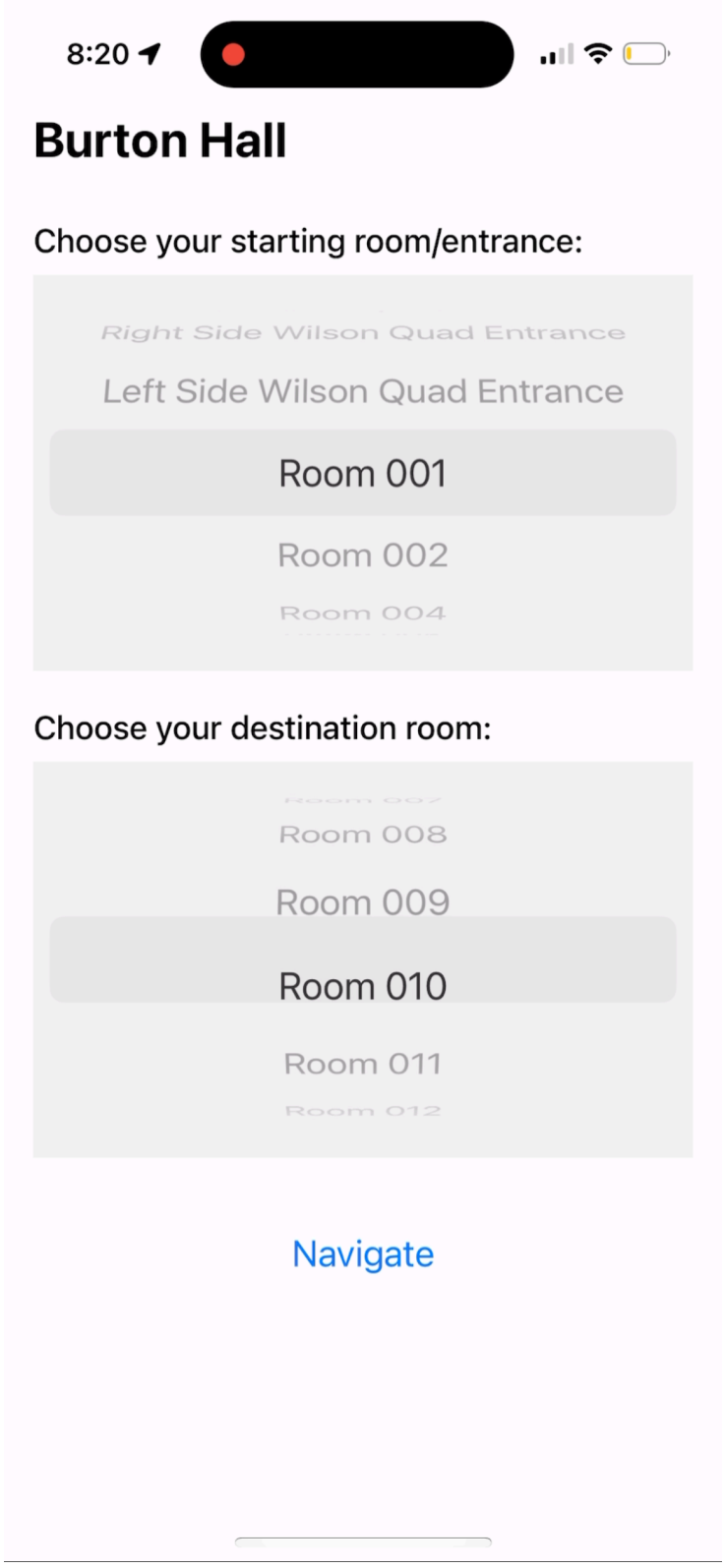
Use Case 1: Choose from a list of available campuses.



Use Case 2: Choose from a list of buildings sorted alphabetically.



Use Case 3: Choose from a list of rooms.



Implementation Methods

Development Environment
To build our app, our group used Git/GitHub (for version control), React Native (JavaScript and TypeScript-based platform-agnostic framework), Expo, Expo Router, and React Webview. We used iOS and Android devices to test the app. All of the code was written in VSCode with text editors like gedit and nano for quickly writing JSON data.

Frontend Development
We used React Native as our frontend framework. Expo also handles some of the native elements on the frontend.

Backend Development
The backend of our app uses Expo and JSON data files. Additionally, before displaying the map to the user, React Webview first renders the map in the background using OpenStreetMap with Leaflet for interactivity.

Extra Use Cases

Extra Use Case 1: Automatically navigate based on current location.
Our original use case did not cover whether a person did not know any buildings or rooms near them, as it would select the building that was closest to them. This use case checks your location for the building routing and for the room routing. If you don't know your location, the app will automatically guess and use that to guide you, which could help people who have trouble finding signs.

Evaluation of Prototype

Our evaluation aimed to identify areas for improvement in user experience and determine the app's usability for our target audience. Our evaluation will answer three key questions:

1. Can users efficiently locate and select a desired room from the dropdown list?
2. Can users navigate to their target building using the map-based directions?
3. Can users find their current location and regenerate a new route if they are lost indoors?

We had 10 undergraduate students from the University of Rochester, aged between 18 and 25. These students are a mix of males and females and are regular smartphone users. The students are from a variety of majors. To prevent external factors like memorization, we have ensured the students have varying degrees of familiarity with the campus layout.

The evaluation will consist of 5 evaluation tasks based on the original use cases:

1. Select a specific building from the navigation page of the university campus
2. Rate, on a scale of 1-5, the success and usability of the indoor navigation feature
3. Determine the entrances of the target building through the map interface
4. Choose a destination room using the dropdown list of rooms in a particular building
5. Use the indoor map, which shows navigation within a building, to reach a target room

Our test procedure is broken down into three separate phases:

Pre-Test: We checked the background of participants and whether they closely represent our intended target audience. A short briefing was provided on the tasks that will be performed by the participants, and they can opt out of the test at this stage.

Test: Participants then used the prototype application to complete the five evaluation tasks independently. During the test, a screen recording will also be active.

Post-test: After the test, participants completed a 5-minute questionnaire on Google Forms and answered questions like what worked well, what was annoying, and potential improvements. They also had the opportunity to ask questions in the post-study interview.

Data collected during the evaluation includes a questionnaire, a post-study interview, and a screen recording. It was measured for the following three aspects of usability:

Effectiveness: how many users completed the tasks, will they use the app in the future, how easy was it to reach the intended destination or locate a building/room from the dropdown list
Efficiency: task completion time, number of errors, time taken to fix errors, observation notes (e.g, confusion, stress)
Satisfaction: app rating from a scale of 1-10, favorite feature, suggestions for future deployment

We calculated the median of quantitative data, such as ratings, as a mean that would not be accurate, given a small sample size of only 10 respondents. The median was a 4/5 for all our use cases and 9/10 for an overall rating of the app. This suggests most participants found the app easy to use and satisfied our usability testing. For qualitative data, we grouped together responses to identify common categories. This was achieved by providing MCQs and responses in a dropdown list to generate a pie chart based on how many respondents select a particular description. The data we have collected will help refine our app and lead to a better UX for future builds. Below lists what users liked/disliked, & potential improvements.

Unique Features: Simple layout, user-friendly with informational text, domain-specific
Disliked Features: Bigger font size for readability, reset the map, restrict maximum zoom in/out

Improvements: saving favorite buildings/directions, informing about construction zones, and map search

In addition, 80% of respondents did not get lost and could understand the app immediately. From our observational notes, users found the interface self-explanatory and didn't require assistance. Respondents stated that "[UR provided] campus maps can be confusing for some students," and this app can fix this gap by helping "students find their way between buildings". The evaluation shows that our use cases are satisfied, and the app has a simple, user-friendly design. Collecting user data for our assessment proved challenging since many students were busy and had little time to participate in testing, especially in finals week. The evaluation process also had many moving parts, which became overwhelming and highly technical for some users. Lastly, participants often give short answers and do not elaborate much, which makes it harder to draw useful insights from open-ended questions.

We learned that a questionnaire was the most effective way to collect feedback, as students find this to be less intimidating and can therefore provide honest answers. For verbal feedback, we learned that asking follow-up questions will enable participants to elaborate and provide more relevant feedback. Lastly, we continually worked to simplify the evaluation as much as possible to make responses more likely.

Conclusion and Future Work

Summary

Overall, we made a lot of progress towards achieving our goal of making buildings easier to navigate. Based on our user evaluation study, over three-quarters of participants successfully understood how to use the app on their first try and got immediate benefits from it. This is a massive step in the right direction, but we still have a bit more to go towards making our app (and thus building wayfinding) universally accessible and beneficial to all users.

Limitations

The first proposal for this project envisioned it as being able to wayfind through and display a route on an accurate 3-D model of a building. However, as our group began writing code, we quickly realized that pathfinding would be the most challenging part. Pathfinding on a 2-D floor plan was already going to be tricky enough, let alone doing it on a 3-D model. Additionally, the time needed to generate the building interior data was quite large, which led us to not create as many interiors as we would've liked.

Key Learnings

Throughout this project, our group's biggest takeaway was coming to grips with how many people share a similar frustration with us (navigating buildings) and want a solution. Going into the project, we figured that while there would at least be some people who had difficulty

navigating buildings, there would be more who were good at wayfinding and had an easier time navigating buildings. Yet, that was not the case, at least to the degree we expected it to be.

Next Steps
Given more time, we could devote more to making our original vision of 3-D pathfinding & models a reality. We could also explore ways to integrate AR functionality, such as by using it to display directions. If we were to do this project over again, we would probably choose a different programming language for our app (such as Kotlin), since some of React Native's quirks caused issues for us during the development process. Particularly, we had problems with our map (we were initially going to use Mapbox, and had a 3-D map of campus and the tunnel system ready to go, but discovered that React Native couldn't render it efficiently), the fact that our app doesn't consistently work on Android, and an Expo SDK update breaking our app at one point. Additionally, we could collect feedback more effectively by coding functionality that prompts participants for feedback directly within the app, instead of or in addition to a survey that they have to leave the app to complete.

Media

Website: Building Buddy Website
Demo Video: **BBDemo2.mp4**
Evaluation Highlights: (see website)

Final Presentation

CSC 211 - Final Term Presentation_S25 (link to our first slide)