

INDY-5 CHOA MAP APPLICATION

Section 05

Fall 2024

Harrison, Lucas, Eddie, Zach

Professor: Arthur Choi

11/19/2024

Website: [https://indy-5-team.github.io/Map Navigation Application/](https://indy-5-team.github.io/Map%20Navigation%20Application/)

GitHub: [https://github.com/INDY-5-Team/Map Navigation Application](https://github.com/INDY-5-Team/Map%20Navigation%20Application)

Number of Lines of Code: ~2500

Number of Project Components/Tools: 3

Total Man Hours: 60

Table of Contents

Introduction	3
Requirements.....	4
Project Goals	4
Design Constraints	5
Environment	5
User Characteristics	5
System.....	5
Analysis	6
Design	7
Classification	7
Responsibilities	7
Composition.....	7
Uses/Interactions.....	7
Resources.....	7
Processing	8
Development	8
Path Finder - Development Overview	8
Tools and Technologies Used	8
Major Components.....	8
Minor Components.....	8
Widgets.....	8
Key Development Steps.....	9
Test Plan and Report.....	10
Version Control	11
Summary	12
Appendix	13

Introduction

Our app uses Dart and Flutter, and its purpose is to be able to map the interior of a building, say a hospital or airport and make it easier for users to navigate to where they need to go quickly and efficiently. The main goal was to have a functioning app with a map and a functioning navigation algorithm to find the shortest path with the later inclusion of QR code functionality. This idea came from the knowledge of a new CHOA hospital opening and from my mother who mentioned it is difficult to navigate from there the idea came to form, but after a few difficulties we had to resort to the map application as proof of concept rather than just a hospital navigation app. It would help make the stress of the hospital visit just a little less as you do not need to worry about where you are going as you can simply follow the path to your destination. It may not be much but our overall goal with this app is to make people's work, travel, or daily tasks much easier by subtracting one less thing they must worry about.

Requirements

Project Goals

We plan to have a fully functional application that will allow a user to input their location and destination, and it will provide them with a route. It will not track location but will provide a list of directs to take

Design Constraints

There are very few design constraints other than making it have a fast loading and interactable UI for the user.

Environment

We plan on the application environment to be on IOS and Android and the application itself will be used for navigating the Atrium building but will be easily transferable to any other building.

User Characteristics

We want the application to have a very responsive and friendly User interface to limit any kind of hinderance and we want it to be able to keep user info (if any) secure and for there to be user support integrated into the application.

System

Our system will have a map that is navigable by our algorithm to guide the user to the destination as quickly as possible, there will also be a help menu and settings. There may also be a history / favorite page where they can easily access locations again in the future if needed

Analysis

Throughout development there were some points for concern regarding the success of our program, namely being able to acquire a map layout of the hospital and taking on a large project in an unfamiliar coding language while also needing to meet deadlines. Our first step towards developing our plan was acquiring the hospital layout for CHOA. Unfortunately, we were denied a map layout due to security concerns and so we had to adjust our plans. The new plan from then was to model the app layout based on the Kennesaw State University atrium building on the Marietta campus, while still maintaining the same concepts for our building navigation app. Moving forward, starting a project in a new coding language such as dart was an adjustment for all of us which was a risk for us to take on a large project and meet deadlines. This issue, however, wasn't a large one, as the team was dedicated and diligent to learning new things and delivering results in a timely manner with fewer problems than anticipated.

Design

Classification

- **Subsystems:** UI/UX, Data Management, QR Code Processing, Network Communication.
- **Modules:** Each subsystem will be broken down into modules to manage specific functionalities, such as the QR code scanner module, map display module, etc.

Responsibilities

- **UI/UX Subsystem:** Provide a responsive, user-friendly interface.
- **Data Management Subsystem:** Handle the storage and retrieval of map data.
- **QR Code Scanning Subsystem:** Accurately scan and interpret QR codes.
- **Network Communication Subsystem:** Manage secure communication with the server.

Composition

- The system will be composed of independent yet interrelated subsystems, each focusing on a specific aspect of the application's functionality (UI, data management, QR code processing, network communication).

Uses/Interactions

- The UI subsystem interacts with users to capture inputs and display outputs.
- The QR code scanning subsystem interfaces with the camera hardware.

Resources

- Utilizes mobile device hardware (camera for QR code scanning, local storage for map data).

Processing

- QR code scanning triggers real-time data retrieval and navigation path updates.
- User inputs on the map interface led to corresponding actions (zoom, navigate, etc.).

Development

Path Finder - Development Overview

Tools and Technologies Used

Path Finder is a mobile indoor map traversal application developed using Flutter and Dart. Flutter is a modern UI toolkit by Google, enabling efficient cross-platform mobile app development, and Dart is the programming language used with Flutter.

Major Components

1. **Map Page:** Displays the map visuals and incorporates the navigation algorithm. It includes gesture support for zooming and panning.
2. **About Us Page:** Provides information about the developers and project goals. It features a clean design matching the app's theme.
3. **Hamburger Menu:** A drawer-style menu for seamless navigation between pages, providing a compact navigation solution.
4. **Settings Page:** Includes settings like dark mode toggle and a font size slider for enhanced user experience.

Minor Components

- **Gear Button:** Positioned in the bottom-right corner for quick access to the settings page.
- **Theme Settings:** Implements a blue and white theme throughout the application.
- **Dark Mode Toggle:** Allows switching to dark mode for better usability in low-light conditions.
- **Font Bar:** Adjustable font size for improved accessibility.

Widgets

Widgets are the building blocks of Flutter applications. They can be classified as Stateless Widgets (which do not change) and Stateful Widgets (which maintain state).

Key Development Steps

1. **Project Setup:** Initialized the project in Flutter, set up dependencies for navigation and state management.
2. **Map Algorithm Development:** Implemented navigation logic and integrated it with map visuals using CustomPainter.
3. **Page Design:** Designed individual pages with a consistent theme using Flutter's widget tree.
4. **Settings Implementation:** Added dark mode and font adjustment features on the settings page.
5. **Testing:** Ensured the seamless performance of user interactions, map responsiveness, and settings functionalities.

Test Plan and Report

For our test plan we focused mainly on the functionality of the app and the mapping algorithm as these were the main components of our project. We specifically tested the apps functionality including how the widgets functioned and if they created any conflicts with each other, we tested our mapping algorithm involving several different routes, and our QR code reading implementation, and we tested to make sure functionality stayed the same across different operating systems.

We tested the app's main functionality by going through all our pages and different settings and features to check if any conflicting errors were caused. Such as if changing to dark mode would make other parts of the page inaccessible, or if altering the text size would make the map too hard to read. We also tested the algorithm itself to ensure that it would find and display the optimal route from the user's current position to their destination and to make sure that the algorithm wouldn't go through existing rooms and walls to reach the destination. For the most part our test environment was inside our application running on an Android emulator where we would test the different functions for any unexpected results

Version Control

GitHub was our version control for our project, as it is a simple and effective medium for version control and cooperation within our project. Members of our group would use pull requests for each feature added. We reviewed each pull request to avoid overriding any work and to ensure stable version control for our application. After each pull request had been reviewed, it was merged in. We created branches for features, for example, adding the routing algorithm and nodes. We also had branches for some bug fixes but small fixes and edits to the website were mostly merged directly into the master branch.

Summary

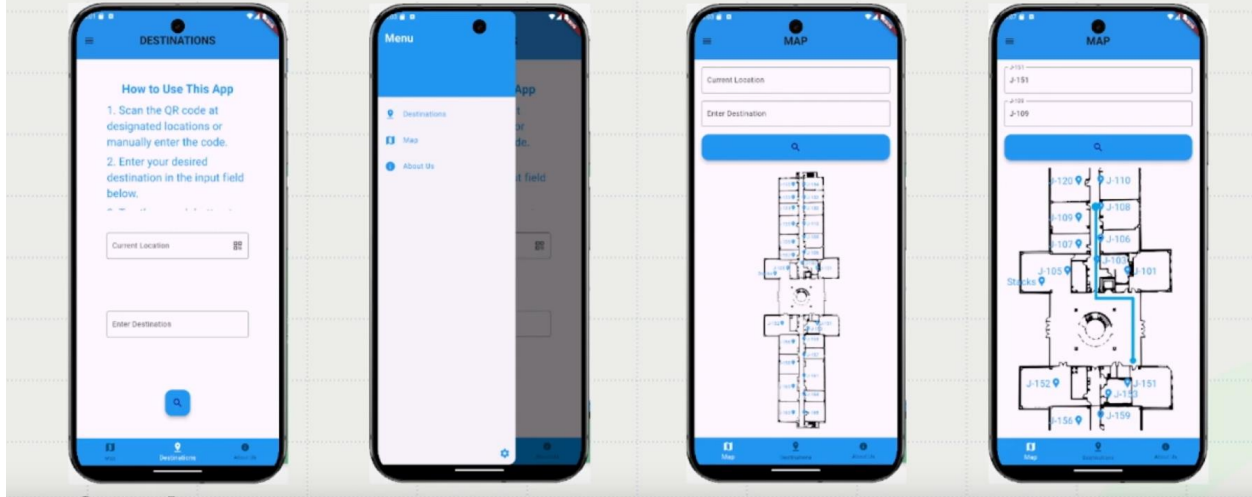
For our project in terms of success and objective requirements, we would classify this as a success. Our application was able to take a map with nodes and different rooms and determine the shortest route from A to B. From this we learned that building a map and placing the nodes by "hand" is a dauntingly time consuming task and have future plans to create an algorithm that can make future maps and place nodes using a heat map to determine where walls are and to assign location names to the nodes at a later date, saving time. We would also like to include more settings to allow for greater accessibility and customization for the users.

When it came to our app demonstrations at C-Day, everyone who viewed our project and saw our app work in real time seemed impressed and liked the prospect of the app and its potential uses. These uses include use in hospital buildings, airports, and many more as it could be used to assist contractors get to their work location at a location. All in all, our demonstrations gave positive feedback regarding our application.

Appendix

Simplicity was a significant goal for when the app was in development

App Flow



As shown in the images above, the app features tabs to view the map, input locations, and the about page, having as little clutter as possible creating a stress-free user experience

Directions are as follows:

1. Open the app
2. Navigate to either the map or destination tab
3. Input manually or click the QR scanner button and allow camera access
4. Input desired destination and press search
5. Follow directions on map