Software Design Document
for

Campus WayFinder

Version <1.0> approved

Prepared by Morayo Ogunsina, Yatri Thoria, Ju Zi Hein, Tanmay Sure

December 12, 2022

Revision History

Name	Date	Reason For Changes	Version
Ju Zi Hein	11/23/2022	Update document	1.0

1. Introduction

1.1 Purpose

The majority of people find it difficult to prospect new areas or unknown locations by themselves. The existing maps of campus can be slightly confusing, sometimes inadequate especially when the classrooms listed in GET are abbreviated. The purpose of this project is to implement a navigation system, as a mobile application that can be used on CalStateLA campus by students and staff, especially newcomers.

1.2 Document Conventions

The document contains headings, subheadings, and body text written in Times New Roman. The headings are font size 20pt and bold, subheadings are 14pt and bold, and the body text is 12pt to make the document's layout more readable. All the texts inside the body are in 1.5 line-spacing and bullet points are used throughout the document to illustrate or outline specifications for the reader to read through more interestingly.

1.3 Intended Audience and Reading Suggestions

The document is mainly written for software engineers, android developers and other IT professionals. It will include technical terminologies and diagrams, and frameworks that will be easy for the general public to comprehend. In order to read it thoroughly, it is recommended to have a general understanding of android application, augmented reality(AR), and Google mapping.

1.4 System Overview

The process or activity of accurately ascertaining one’s position and planning and following a route. The application will serve as an aid to students and staff and lead in navigation through the campus, as we selected an area in our CAL STATE LA campus for our study. We aim to propose a navigational system that combines both outdoor and indoor capabilities, which will enable a student to find their way around campus and find their classrooms.

2. Design Considerations

This section describes many of the issues which need to be addressed or resolved before attempting to devise a complete design solution.

2.1 Assumptions and Dependencies

- Related software or hardware
- Operating systems
  - Windows
  - MacOS
- End-user characteristics
  - Understanding the students' need when using our mobile app
  - Create it simple and efficient for the users to use and get familiar with technology
- Possible and/or probable changes in functionality

2.2 General Constraints

- Hardware or software environment
  - ➢ Android application that has access to the Internet or Wi-Fi
  - ➢ Runs on mobile app provided by GoogleMap
- End-user environment
  - ➢ Up-to-date mobile phone or smartphone
- Availability or volatility of resources
  - ➢ Prepopulated lists of resources from GPS data
  - ➢ Open-source libraries such as the Google Maps SDK for Android
- Interface/protocol requirements
  - ➢ Outdoor interface with the route from GPS
  - ➢ Indoor interface using AR core or textual output form
- Security requirements (or other such regulations)
  - ➢ Current location data from users(GPS)
- Memory and other capacity limitations
  - ➢ 8 GB RAM with 256GB of space
  - ➢ 16 GB RAM with 500GB or 1 TB of space
- Performance requirements
- Network communications

- ➢ The user must have a stable internet connection
- Verification and validation requirements (testing)
- Other means of addressing quality goals
- Other requirements described in the requirements specification

2.3 Goals and Guidelines

While working on the project, we set many goals and milestones for ourselves to ensure we get the best working app we can possibly achieve. It ranged from allowing users to select the desired destination to guiding them with the most efficient and fastest route for both indoor and outdoor mapping.

- Easy installation on Android phone
  - developed the software in a more general format which will be understandable by users not belonging to the technical background.
- Emphasis on speed versus memory use
- The product should work, look, or "feel" like an existing product
- The product should give the most trustful and competent result
- The users data will be secured
  - Users do not need to type their personal data and no password will be stored for validation

Guidelines :

- Follow a straight-forward and unambiguous guideline
- Mainly focus on usability
- Keep it simple to use with UI and implementation

2.4 Development Methods

The team decided to follow on agile development for the users to be friendly with little-to-no training and practice. For outdoor navigation, the team has shifted their focus on Google Maps Intent which is a feature provided by the "com.google.android.apps.maps" package which comes with the android studio library, the code is fully implemented in JAVA. Using this intent we launch Google Maps navigation with turn-by-turn directions to the address or coordinate specified. For indoor navigation, the team followed QR code scanning by using ZXing library for

Localization. Besides, for mapping, we use a full 3D map of the building floor rendered in Unity 3D whereas POI(Point of Interest) would be hardcoded into the map. Lastly, 3D AR Objects are used to navigate Mapped space for path navigation.

3. Architectural Strategies

This chapter deals with different approaches on how to create an intuitive application for a mobile device to support a person on campus in different situations.

As described in the introduction chapter there are two main goals for this application prototype. The first goal is to provide the user with an on campus navigation system that helps him to get from his current position to a designated building or room. The second feature of this application focuses on presenting different kinds of useful information in an augmented reality view.

3.1 Use of a particular type of product

A route on campus of the University of Calgary can be very long and thus very confusing for a person who is not familiar with this area. To simplify the route, it has to be broken down into smaller segments that provide an easy overview. According to the resource and limited timeframe, the team decided to map out only one or two buildings to navigate. Before we started with our app, we had to search for the floor plan on campus which we basically had to populate on our map view. In order to launch Google map intent on our app, the team used Google Map Intent developed by JAVA. The main platform we use to implement our app is Android Studio. Also, to do indoor navigation, the team mainly used Unity3D Software, Android Studio and compatible libraries. For the main platform, we use the same languages Java and XML on Android Studio.

3.2 Reuse of existing software components to implement the system

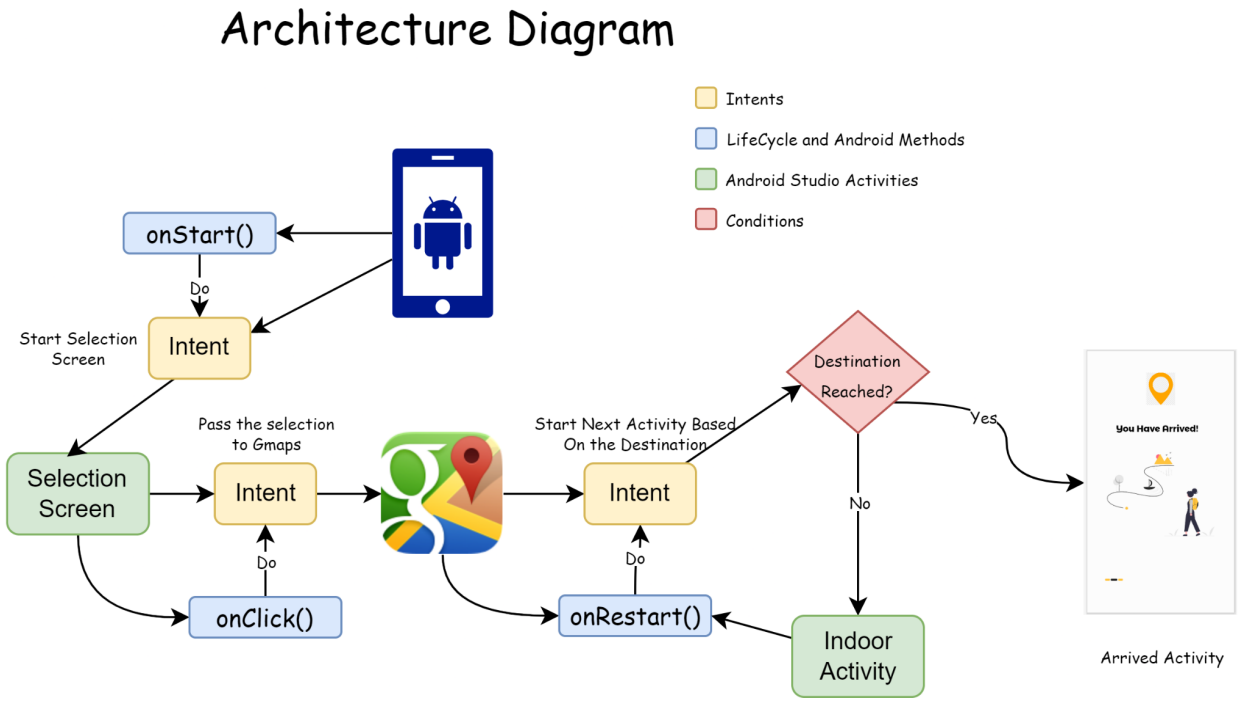
The team made extensive use of the Geographical date intent provided by Google. We had the challenge of determining a usable indoor position using Augmented Reality(AR). Aberrations and measurement errors are identified and tried to reduce. In the end the result of this technique is compared to the location information based on GPS provided natively by the Android operating system.

3.3 Future plans for extending or enhancing the software

Indoor location tracking in the current version of the application is not very accurate. So, the team has thought of enhancing this work with more exact information about the location of the access points, a more accurate indoor position could be estimated. A long term perspective could offer users better interaction with their environment in the augmented reality view. Whereas in

the current version of the prototype, the team has chosen only some specific location to map out during the certain period of time, in later releases with an accurate method to determine the orientation of the device indoors and full evacuation map view for path planning and route estimation for all buildings and landmarks on campus.

4. System Architecture



Indoor infrastructure is a 3D plan for constructing the indoor structure of the college. Indoor infrastructure is the main requirement for indoor location tracking systems. Based on the infrastructure to develop the indoor application on the android platform. It is used to find the indoor location.

4.1 Level 0 DFD

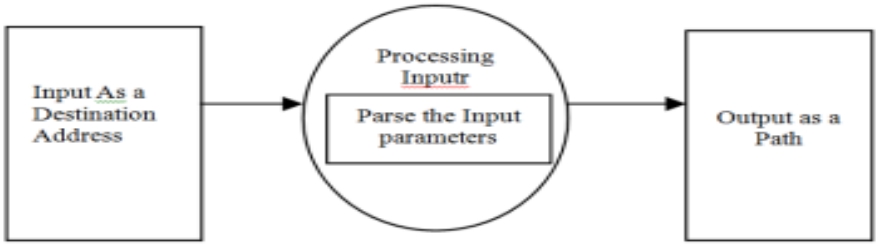


Fig 1 Level 0 DFD

4.2 Level 1 DFD

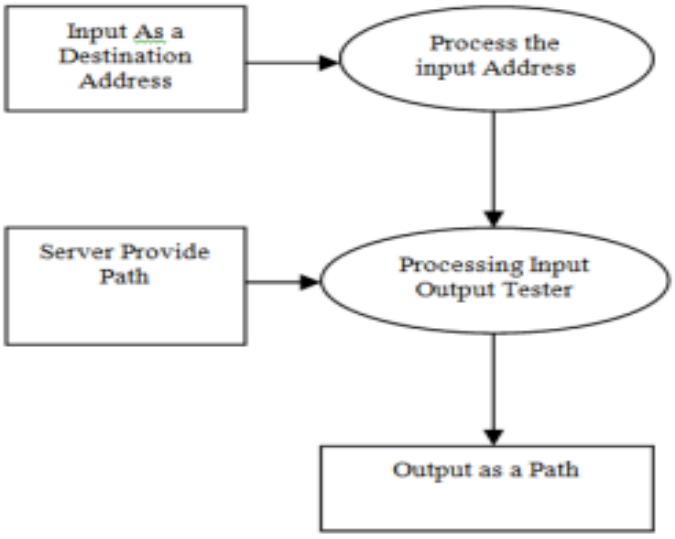


Fig 2 Level 1 DFD

5. Policies and Tactics

5.1 Choice of which specific products used

- IDE
    - Android
- Software
    - Android Studio
    - 3D software
    - Unity engine
- Programming Language
    - Java
    - XML
- API
    - Mapbox API

5.2 Plans for ensuring requirements traceability

We will be using GitHub because it's fast and accessible. Every time one of the group members makes an "update," the other group members will know and have access to it.

5.3 Plans for testing the software

The plan for testing the software is to aim for professionalism. The goal is to provide the user the precise route and combine augmented reality and 3D model mapping together, that way, we will have more extended accessibility to indoor navigation.

5.4 Plans for maintaining the software

The android application will be maintained by adding more location and routes for new buildings and landmarks when needed to maintain efficiency.

6. Detailed System Design

The following figure (Fig.3) depicts the project flowchart. It describes the activity plan of the project. The activities will be carried out in the same order.

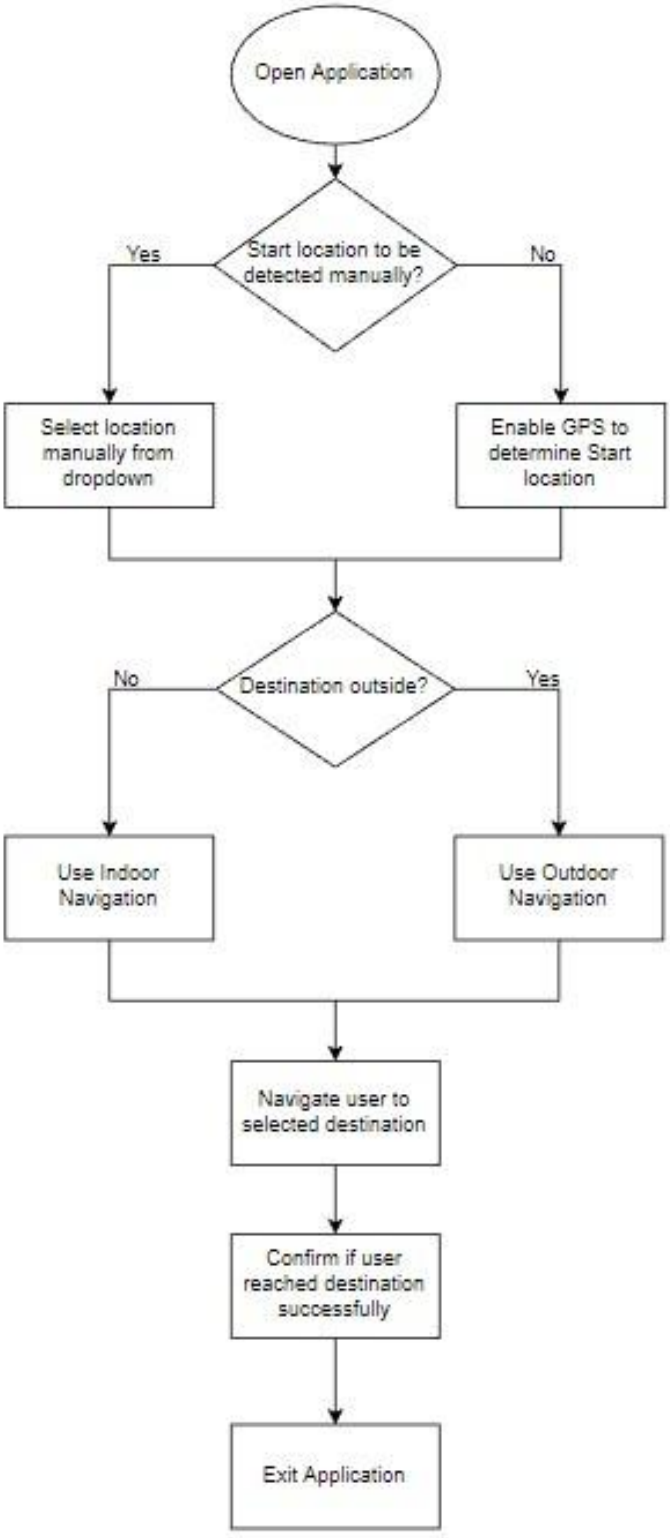


Fig.3 Flowchart Diagram

6.1 Outdoor Navigation

6.1.1 Responsibilities

The main features of the application are browsing maps and routing on campus. Connected to the server, the application offers users, after defining two places on campus, a route which leads to the destination point.

6.1.2 Constraints

Due to the limited amount of schedule and resources, the number of features that can feasibly be implemented will be limited. To overcome this constraint a comprehensive schedule of implementation and maintaining time will be created to ensure that all necessary features will have sufficient time allocated to their development.

6.1.3 Composition

We first tried to use GMap Navigation SDK but since it is in paid version, we changed to a new feature called Google Map Intent.

```
    //openDestination();
    Intent intent = new Intent(Intent.ACTION_VIEW,
            Uri.parse("google.navigation:q=34.06787402629378,-118.16584018494166&mode=w"));
    intent.setPackage("com.google.android.apps.maps");

    if (intent.resolveActivity(getPackageManager()) != null) {
        startActivity(intent);
        //     }
    };

}
```

6.1.4 Uses/Interactions

There are three main criteria that define important waypoints on a campus route on which it would be useful to divide the route into different segments:

- Entering or leaving a building
- On height change (If the path uses more than one floor it has to be split, because it is not possible to display more than one floor on a 2D map at the same time. Therefore, the route is segmented at every elevator or staircase.)
- Landmarks such as sculptures, big signs

6.1.5 Resources

6.1.5.1Google Map Intents

6.1.6 Interface/Exports

6.1.6.1 Visual Output

6.2 Indoor Navigation

6.2.1 Responsibilities

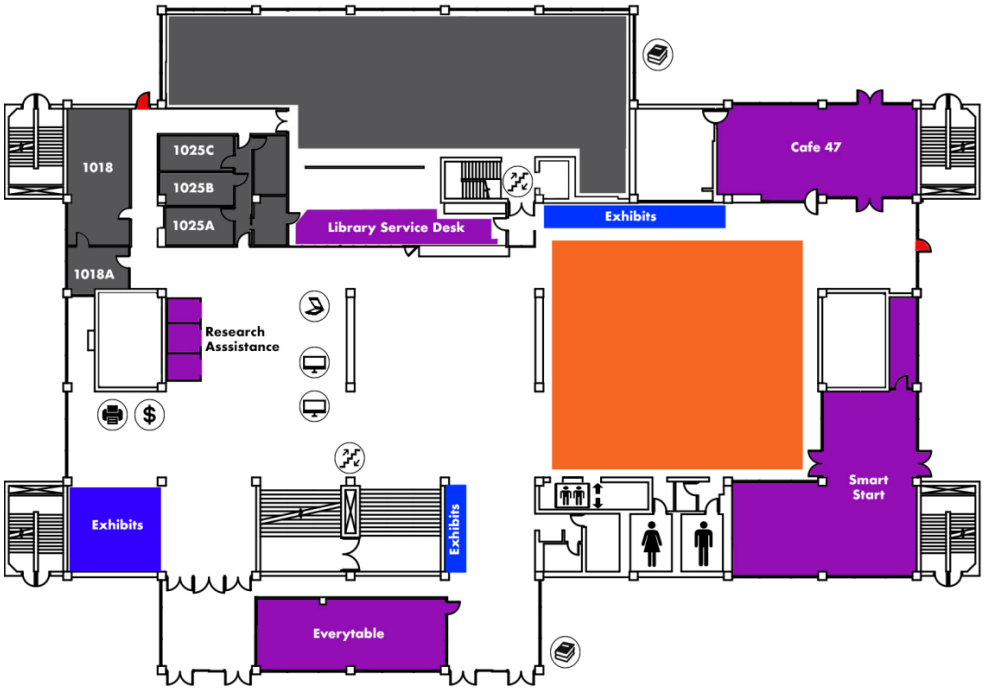
The main features of the application are browsing maps and routing on campus. Connected to the Unity Engine, the application offers QR code for location, after defining two indoor places on campus, a route which leads to the destination point.

6.2.2 Constraints

Due to the limited amount of schedule and resources, the number of features that can feasibly be implemented will be limited. To overcome this constraint a comprehensive schedule of implementation and maintaining time will be created to ensure that all necessary features will have sufficient time allocated to their development.

6.2.3 Composition

- Localization - Position marking using QR code scanning, ZXing library
- Mapping - Full 3D map of building floor rendered in Unity3D, Point of Interests (POIs) hardcoded into map.
- Path Navigation - 3D AR Object is used to navigate Mapped space



6.2.4 Resources

6.2.4.1 Unity 3D Engine

6.2.4.2 Exporting QR

6.2.5 Interface/Exports

6.2.5.1 Textual/Visual output

7. Detailed Lower level Component Design

Does not apply

8. Database Design

No database was used in this project; therefore, this does not apply

9. User Interface

9.1 Overview of User Interface

This provides the wireframe layouts of the overview elements of the system from the user's perspective.

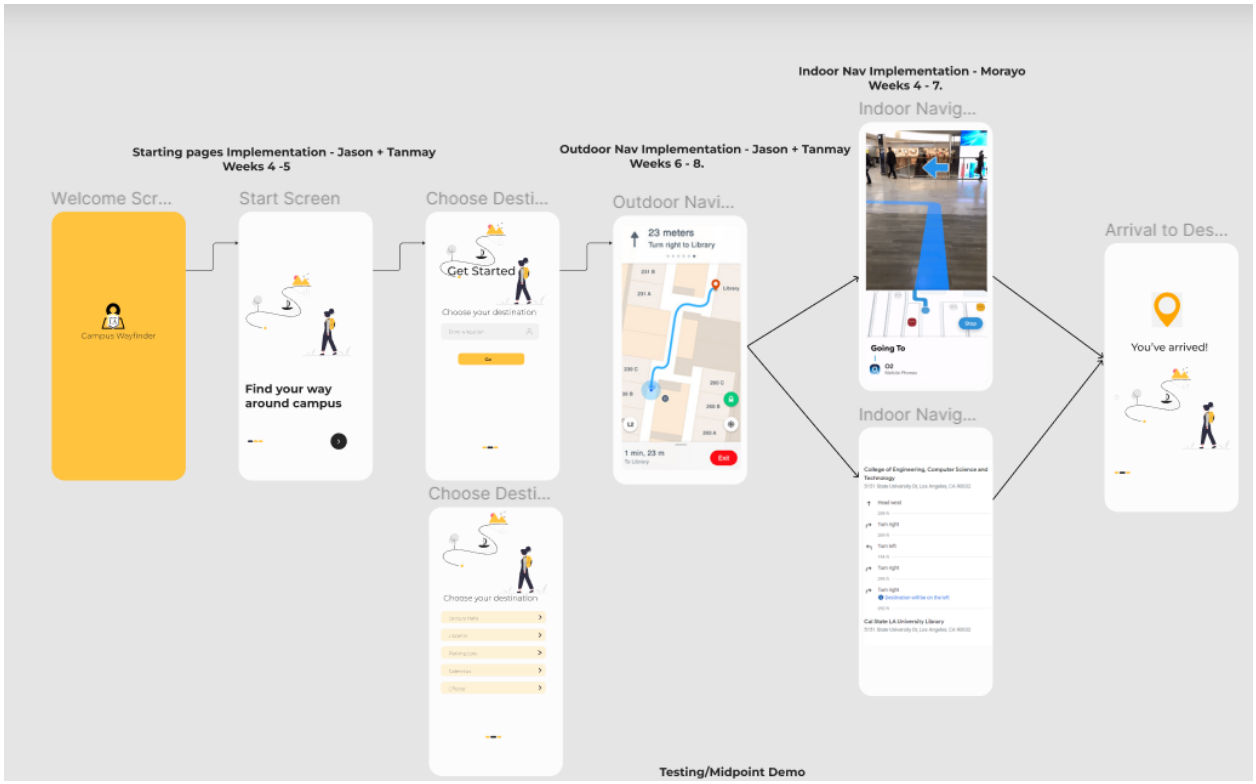
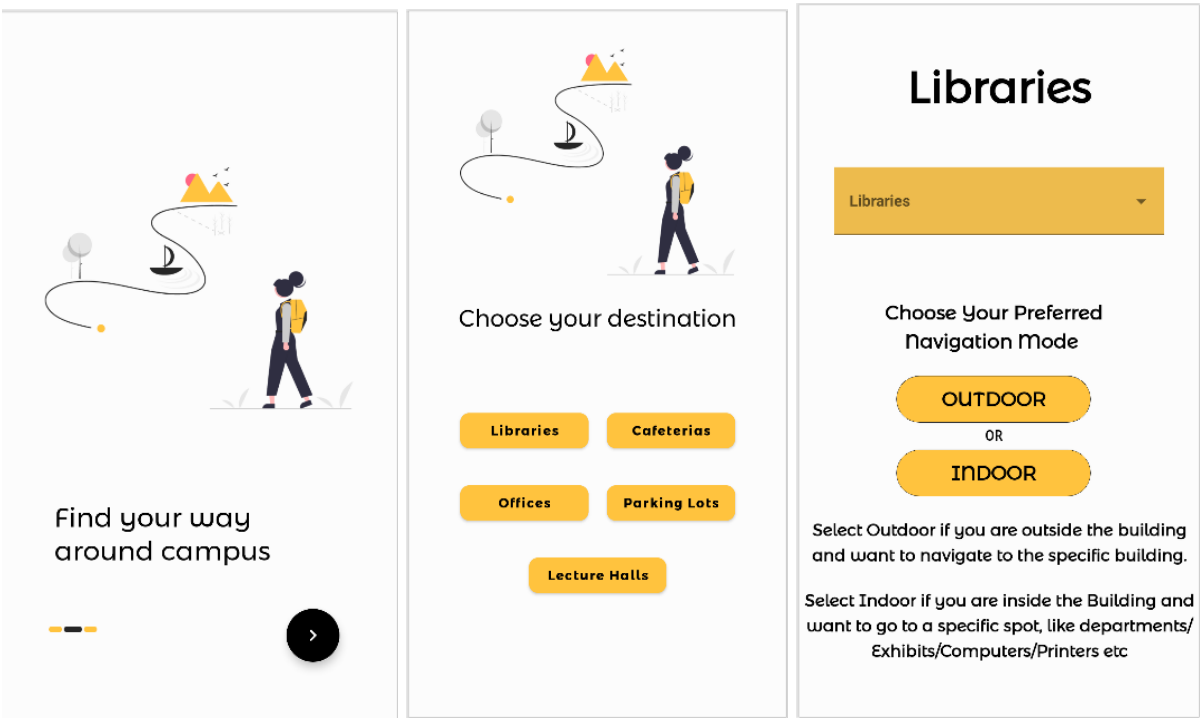


Figure 4: mock layout for navigation

9.2 Screen Frameworks or Images

These are some actual screenshots of the various UI screens and popups. It includes from selecting the desired destination either indoor or outdoor to arriving at the target using the map provided on the app.



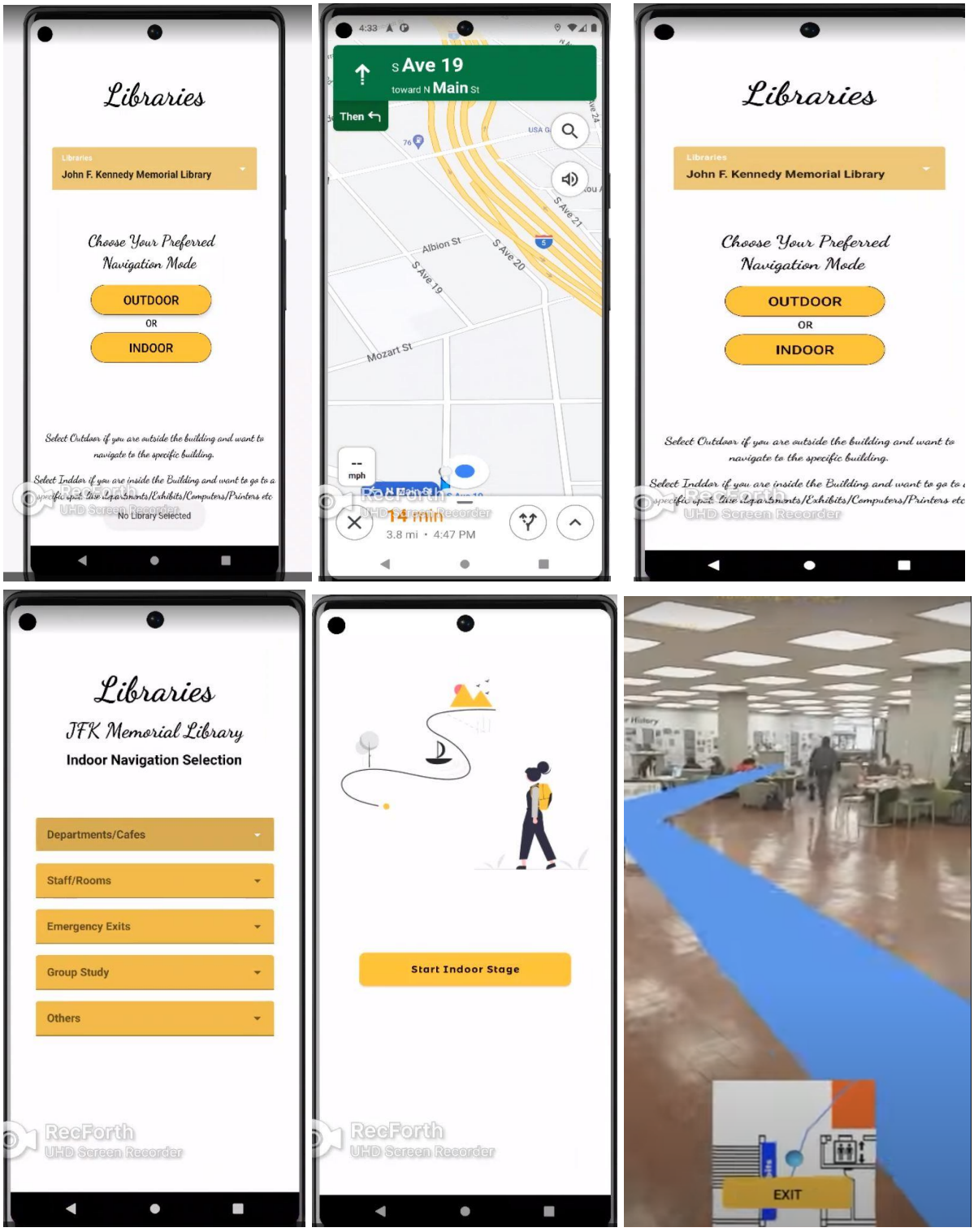


Figure 5: Actual work flow of both outdoor and indoor navigation

10. Requirements Validation and Verification

Since the project does not include validating and verifying the users' credentials on map,then, no requirements validation and verification was used in this project.

11. Glossary

API - stands for application programming interface, which is a set of definitions and protocols for building and integrating application

AR - is an interactive experience that combines the real world and computer-generated content.

Android Studio - is the official Integrated Development Environment (IDE) for Android app development, based on IntelliJ IDEA

GPS - The Global Positioning System (GPS) is a satellite constellation supporting highly accurate positioning, navigation and timing (PNT) measurements worldwide.

JAVA - Java is a widely-used programming language for coding web applications. It has been a popular choice among developers for over two decades.

3D - or three dimensional, refers to the three spatial dimensions of width, height and depth. The physical world and everything that is observed in it are three dimensional.

3D engine - are software that enable real-time generation of 3-dimensional graphics (in contrast to pre-rendered animation that appears in movies).

Mapping - The term “navigation map” represents the route instructions, as it were, throughout the conceptual model.

Localization - The process of identifying the position and orientation of a device relative to some reference point.

Unity - is a cross-platform build engine and it builds real-time 3D projects for various industries across games, animation, automotive, architecture, and more

UI - The user interface (UI) is the point of human-computer interaction and communication in a device

QR code - a machine-readable code consisting of an array of black and white squares, typically used for storing URLs or other information for reading by the camera on a smartphone.

Mapbox API - is an advanced and flexible map service that can be integrated into mobile apps.

XML - Extensible Markup Language (XML) is a markup language and file format for storing, transmitting, and reconstructing arbitrary data.

12. References

[1]https://ijarcce.com/wp-content/uploads/2012/03/IJARCCE8H-A-pritam-the-campus-navigator.pdf

[2] Susovan Jana, Matangini Chattopahyay “An Event Driven University CampusNavigation System based on Android Platform”IEEE Conference on application and innovation in mobile computing, pp-182-187, 2015

[3]https://www.researchgate.net/publication/312057435_Design_and_Implementation_of_a_Navigational_System_for_the_University_of_Ilorin