Augmented Reality Application for Indoor Navigation

B.Sc. Project Report

Naif Taleb Aljadaani

Abdullah Majid Alqurashi

Abdulkarim Mohammadraffat Radwan

Abdullah Faisal Softa

Mohanad Masoud Alshalihi

DEPARTMENT OF INFORMATION TECHNOLOGY

KING ABDULAZIZ UNIVERSITY

KSA

2024

Augmented Reality Application for Indoor Navigation

Project Report Submitted in Partial Fulfillment of the Requirements for the Degree of

Bachelor of Science In Information Technology

By

Naif Taleb Aljadaani (Roll No. 2035320)

Abdullah Majid Alqurashi (Roll No. 2035175)

Abdulkarim Mohammadraffat Radwan (Roll No. 2041059)

Abdullah Faisal Softa (Roll No. 1845079)

Mohanad Masoud Alshalihi (Roll No. 1940499)

Department of Information Technology
Faculty of Computing and Information Technology
(King Abdelaziz University)

2024

Certificate

This is to certify that this project report entitled Augmented Reality Application for Indoor Navigation by **Naif Taleb Aljadaani (Roll No.2035320)**, **Abdullah Majid Alqurashi (Roll No.2035175)**, **Abdulkarim Mohammadraffat Radwan (Roll No.2041059)**, **Abdullah Faisal Softa (Roll No.1845079)**, **Mohanad Masoud Alshalihi (Roll No.1940499)**, submitted in partial fulfillment of the requirements for the degree of Bachelor of Science in Information Systems of the King Abdelaziz University, Rabigh, during the academic year 2024, is a bonafide record of work carried out under our guidance and supervision. The results embodied in this report have not been submitted to any other University or Institution for the award of any degree or diploma.

(Guide)	(External Examiner)	(Chairperson)
Dr.Khalid Dowaid		Dr.Almohannad Alorfi

ACKNOWLEDGMENT

It is a pleasure to express thanks to

Dr.Khalid Dowaid

&

Dr. Almohannad Alorfi

for the encouragement and guidance throughout the course of this project.

And

We are grateful to the faculty of computing and information technology for helping and supporting us with our project.

Supervisor's ________________________________

Chairperson ________________________________

DECLARATION

we declare that this project entitled ‘‘Augmented Reality Application for Indoor Navigation’’ is the result of our own research except as cited in the references. The project has not been accepted for any degree and is not concurrently submitted in candidature of any other degree.

DEDICATION

This project is dedicated to our families, whose unwavering support and encouragement sustained us throughout this research journey.

to our mentors and colleagues, whose guidance and insights shaped our understanding and curiosity.

ABSTRACT

Mobile devices are increasingly being used for indoor navigation, which is necessary for users who desire to locate locations inside buildings. Several applications for indoor navigation have been proposed that make use of various technologies, including Wi-Fi fingerprinting. To guide users to their goals, most of these applications employ pre-calculated paths and a set background map. Users of these systems must be familiar with reading maps in general and with the functionality of indoor maps in particular. Furthermore, before navigation begins, these system types must do intricate and precise computations to identify routing patterns, which may be impacted by erratic Wi-Fi signals. It integrates a digital compass, augmented reality (AR), and Wi-Fi fingerprinting into a single android app. And in this project, we are aiming to create an augmented reality (AR) navigation application tailored for our university campus. The application seamlessly combines outdoor navigation enriching the user experience, for students, professors, and visitors. By harnessing AR technology our app offers step by step guidance within university buildings making it simple for users to find classrooms, offices, libraries, and other amenities. Moreover, in addition to spaces our app also covers areas so that users can explore the campus grounds. Our goal, with this AR navigation app is to improve efficient movement across the FCITR premises.

Keywords : augmented reality, android app, navigation System, Indoor navigation.

TABLE OF CONTENT

Chapter IV: SYSTEM ANALYSIS AND DESIGN

Chapter V: SYSTEM IMPLEMENTATION

Chapter VI: SYSTEM TESTING

Chapter VII: CONCLUSIONS

LIST OF TABLES

LIST OF FIGURES

CHAPTER I
Introduction

1.1 Background

Portable outdoor GPS navigation apps have been invaluable to numerous individuals. Using a smartphone makes it easy to navigate and find your way in a strange place. Indoors, GPS satellite signals may not be accurately traceable for navigation. Directing individuals to precise destinations inside indoor settings presents a difficult challenge.

Operators of complex structures such as airports, hospitals, and other public buildings face challenges in efficiently navigating visitors across their premises.

Augmented reality is a technique that combines computer-generated virtual information with the actual environment to enhance consumers' immersive experiences.

Currently, AR technology is predominantly utilized in several fields including digital publishing, digital marketing, design simulation, and science research. Augmented reality-based mobile navigation applications have emerged as a new area of academic interest. The technology's driving force is to improve visual effects to help people identify objects more quickly. Thus, overlaying an interactive 3D image over a real scenario can enhance the effectiveness of training applications, such as aiding doctors in surgical demonstrations, robotic design, and more.

Indoor navigation is a topic that has been extensively researched utilizing various methods and technology. It involves a system of interconnected devices designed to locate objects or individuals within a building. There has been a consistent and rapid increase in the development and application of positioning (sometimes called localization) and navigation technology outside in recent years. Efforts to apply these technologies indoors have prompted extensive research in this area, motivating researchers to find ways to improve navigation in indoor environments for people [1].

1.1.1 A Brief History of AR

Augmented reality technology has significantly altered how we interact with content in the physical world during the past five decades. Let us review the origins of AR technology:

- In 1968: Harvard professor and computer scientist Ivan Sutherland developed the very first head-mounted display known as 'The Sword of Damocles'[2].
- 1974, Myron Kruger established a laboratory at the University of Connecticut named ‘Videoplace’ focused solely on artificial reality [2].
- 1990, Tom Caudell, a researcher at Boeing, introduced the term 'augmented reality'[2].
- 1992, Louis Rosenburg, a researcher at the USAF Armstrong's Research Lab, developed 'Virtual Fixtures’ one of the earliest fully operational augmented reality systems [2].
- 2000, Hirokazu Kato created an open-source software library named the ARToolKit. This package assists developers in constructing augmented reality software applications. The library utilizes video tracking technology to superimpose virtual pictures onto the physical environment [2].
- 2013, Volkswagen introduced the MARTA app (Mobile Augmented Reality Technical Assistance) to provide technicians with detailed repair instructions from the service handbook [2].

- 2014, Google introduced its Google Glass devices, a set of augmented reality spectacles designed for immersive experiences [2].

- 2017, IKEA launched an augmented reality application named IKEA Place, which had a significant impact on the retail sector [2].

- 2024, Apple released Apple Vision Pro, a goggle that utilizes augmented reality and virtual reality technology.

1.2 Project Aim

The aim of this project is to develop a navigation app using augmented reality specially for our university campus. The app will provide seamless indoor and outdoor navigation, allowing students, faculty, and visitors to efficiently find their way around the campus, locate buildings, and navigate to specific classrooms or offices.

1.3 Project Objectives

The objectives of the research are:

1. Develop accurate method for determining the user's position within the indoor spaces.

2. Provide clear directions to guide users from point A to point B.

3. Implement efficient routing algorithms to calculate optimal paths.

1.4 Project Scope

The scopes of this project are listed as follows:

A. We will use Figma, Lunacy, and Unity to make the 3D structure of the building and floors.
B. We will use google maps and manual measurement indoors and outdoors to build a 3D structure.
C. The application will only be for Android.
D. The project will cover the FCITR building.

1.5 Significance of this work

- Enhanced Navigation Experience:

AR navigation systems overlay digital information onto the real world, making navigation more intuitive and engaging.
Users receive real-time directions directly within their view, reducing the need to switch between maps and reality.

- Precise Indoor Wayfinding:

Traditional GPS is insufficient for indoor spaces. AR navigation fills this gap by providing accurate positioning within buildings.
Users can find specific rooms, offices, restrooms, and amenities seamlessly.

- Seamless Transitions Between Indoor and Outdoor:

AR navigation bridges the gap between indoor and outdoor contexts.
Users smoothly transition from building interiors to outdoor paths without losing guidance.

- Improved Safety and Efficiency:

AR overlays guide users along optimal routes, helping them avoid obstacles and hazards.
For drivers, AR navigation ensures they stay in the correct lane during highway driving.

1.6 Project Plan

A project plan is a detailed timeline that typically consists of charts that provide detailed information about each activity, their interrelationships, and the critical ones that can cause delays in the project, Figure 1-1.

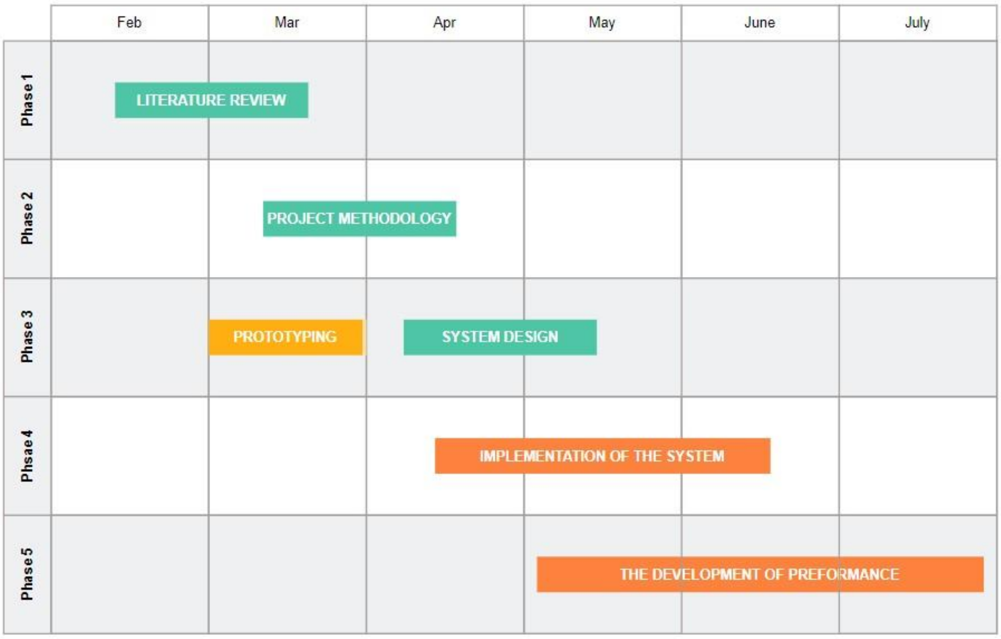

Figure 1-1: Gantt chart for the project.

1.7 Conclusion

Overall, in this era of technological advancements, Augmented Reality (AR) has emerged as a powerful tool with transformative potential across various domains. One such exciting application is indoors and outdoors navigation within educational institutions. As universities and colleges grapple with the challenges of complex layouts, inadequate signage, ongoing construction, accessibility concerns, time management, and security, AR offers innovative solutions to enhance the wayfinding experience.

CHAPTER II
Literature Review

2.1 Introduction

We have noticed the challenges faced by newcomers and even visiting staff in navigating our expanding campus. Finding specific buildings can be difficult, especially with the increasing number of buildings [3]. Traditional campus navigation systems, whether web-based or app-based, have their limitations, such as limited navigation experience, lack of high-precision localization, and poor user interface. In order to address these issues, I have conducted research on augmented reality (AR) technology and its potential in improving campus navigation [4].

2.2 Background Problems

our university is the rapid structural expansion due to the increasing number of enrolled students. As the university grows, it becomes increasingly challenging for students, parents, and newly enrolled individuals to navigate the campus effectively. The expanding infrastructure, including new buildings, pathways, and facilities, adds complexity to the campus layout, making it difficult for individuals to find their way around [3]. The sheer scale of the campus can be overwhelming, especially for newcomers who are unfamiliar with the layout. Without a reliable and user-friendly navigation system in place, students may waste valuable time and effort trying to locate specific buildings, classrooms, offices, or amenities. This can lead to frustration, delays, and even tardiness to classes and appointments.

Another problem lies in the complexity of indoor navigation. Large, multi-story buildings present unique challenges for users, as traditional navigation systems heavily rely on GPS signals, which are often weak or unavailable indoors. This limitation makes it even more difficult for students to navigate within buildings, locate specific classrooms or offices, and efficiently move from one floor to another. It can be particularly daunting for newcomers who are unfamiliar with the building layouts and may require assistance in finding their destinations [4].

2.3 Problem Analysis on Current System

2.3.1 Limited Navigation Features:

The current system provides limited navigation features, lacking interactive and immersive elements. Users are presented with static 2D maps and basic directions, which restricts their ability to navigate efficiently. Advanced features such as turn-by-turn directions, real-time updates, and personalized route suggestions are absent, limiting the system's effectiveness [5].

2.3.2 Inaccurate and Outdated Information:

The accuracy and currency of the information provided by the current system are questionable. Outdated building names, room numbers, and other details lead to confusion and incorrect directions. This problem becomes more pronounced when there have been recent campus expansions or infrastructure changes [6].

2.3.3 Insufficient Indoor Navigation Support:

Navigating within buildings poses a significant challenge in the current system. Weak or unavailable GPS signals indoors result in poor location accuracy. Users struggle to find specific rooms, offices, or facilities within buildings, leading to frustration and delays. The lack of comprehensive indoor maps and navigation guidance exacerbates this problem [7].

2.3.4 Complex User Interface:

The user interface of the current system is complex and unintuitive, hindering efficient navigation. Cluttered screens, confusing menu structures, and non-intuitive controls contribute to user frustration. Users, especially newcomers, face difficulties in understanding and utilizing the system effectively [5].

2.3.5 Limited Accessibility:

The current system does not adequately address the diverse needs of all users. Accessibility features for individuals with disabilities, such as text-to-speech or audio guidance, may be lacking. Additionally, language support may be limited, making it challenging for non-native speakers to utilize the system effectively [7].

2.4 Benefit Software Proposed:

2.4.1 Main benefit

The main benefit of our program is to help our fellow students to reach from point A to point B using augmented reality by providing real-time directions so that they can reach their respective class efficiently without losing any time going through the whole floor looking at each class.[9]

2.4.2 Ease of Use:

In real-time, an AR indoor navigation tool like our software can relay directions to the users with ease, and clear indicators on their phone screens. It can deliver information through visual representation of an arrow directing the user to follow the correct path drawn from his location to the desired class.[9]

2.4.3 Accuracy and Precision:

Maps can be difficult to follow especially in a building such as our collage buildings. However, AR-based indoor positioning technologies have an accuracy within millimeters.
They can pinpoint a user’s location and give them real-time directions to reach their desired destination, making it highly difficult to get lost.[9]

2.4.4 Improvement of the Student Experience:

With our software all types of students can save so much time whether they are new to the collage our accustomed to it, the reason for that is both types are going to have new courses each semester with each course comes a different classroom. That is why we are confident that it will dramatically improve the overall experience for our fellow students.[9]

2.5 Technology used Front End and Backend:

2.5.1 Front End:

For front end we mainly going to use 3 software which are Figma, Lunacy, and Unity. Figma and lunacy have a lot of similarities and the reason for that is both are design tools to help us design and produce an appealing and easy graphical user interface both have their own benefits such as lunacy is easy to install since it is in the Microsoft Appstore and it is more about making a sketch or prototype of our design while lunacy on the other hand can allow us as a group to design on the same file together so that we can come up with the most presentable design. as for unity since our application is going to be made on it, we can make a basic user interface then we can implement the interface that we designed on the actual application.

2.5.2 Back End:

For backend we are going to use Unity, Blender, and C#. Unity has a whole section and libraries dedicated for augmented reality this will help us a lot, these libraries and unity is programmed using C# which is helpful for us since we studied C#, as for blender we are going to use to make the 3D structure of the buildings and floors alongside unity 3D environment.

2.6 Related Works

It's important that we evaluate the existing approaches used in this field to gain a thorough understanding and knowledge the difficulties that we may end up face during this project we will discuss the already established AR navigation software or companies that provide the service since it is mostly companies that provide this service rather than a dedicated software, Table 2-1.

Title	Author	Description	Technologies
Augmented Reality BC Campus Tour App [9]	Evgeniya Kokareva, Ana Satkeeva, Robin Shaw, Bridget Wellen Supervisor: Dr. Fatma Cemile Serce	The mobile application enables users to explore Bellevue College's buildings using augmented reality (AR). Users can point their camera at building letters to identify them and access information from the college's website. The app also integrates AR features for interacting with services within selected buildings.[9]	Wi-Fi Bluetooth Ultrasound Sensors AR
Application of Augmented Reality in Campus Navigation [10]	Li Zhigang*, Qi Guanglei, Hu Wenkai, Ma Xiangyu, Guo Qinsheng	Campus navigation software requirements have increased. Traditionally, visitors use navigation apps or guides. Imagine software that enhances senses, offering a unique experience. AR technology can provide this, acting as a navigation tool and a commentator that shares the campus's history and culture.[10]	Sensors Wi-Fi QR codes Bluetooth AR
Software Requirements for Mobile Augmented	Rashidi Abd Rashid1. Halina Mohamed Dahlan	Our software analyzes the requirements needed to develop a mobile AR tourism application. The study emphasizes the increasing importance of AR in enhancing tourist experiences by providing	Wi-Fi QR codes Bluetooth AR

Reality Tourism Application.[11]		updated, relevant information and interactive feature.[11]	
AR-UTAR Kampar Campus Navigation [12]	HEW TENG WEI	This project involves the development of an AR application designed to assist with navigation around the UTAR campus.[12]	Bluetooth Apple ARKit library Sensors Wi-Fi Blender Xcode9
Campus Navigation Using Augmented Reality [13]	Mayuri Tamhane, Prathamesh Sarjekar, Chirag Gupta, Prof. V. M. Kharche	Android mobile application designed to aid navigation within a campus. The app uses AR to guide users to their desired locations by displaying augmented arrows or texts on a live feed from the smartphone camera.[13]	Bluetooth Android SDK Sensors Wi-Fi Vuforia Augmented Reality SDK
Augmented Reality Navigation [14]	Munesh Kumar Sharma, Satya Chachaundiya, Vishal	AR navigation system. The system leverages AR to overlay navigation information on a real-world view captured by a smartphone camera. It aims to enhance the user experience by providing real-time, user-centric navigation details.[14]	Bluetooth Vuforia Augmented Android SDK Sensors Wi-Fi
Indoor Navigation Using Augmented Reality [15]	Kyriaki Kalantari, Samuel B. Scherer, Thomas Liebig,	Our software use Microsoft HoloLens for indoor navigation within a university building. It evaluates the effectiveness of augmented reality in aiding users to navigate complex indoor environments, providing a detailed analysis of user experience and the accuracy of the navigation system.[15]	Microsoft HoloLens Unity Azure Spatial Anchors Indoor Mapping

Table 2-1: summary of the existing applications of related works.

2.7 Conclusion:

In conclusion the difficulties of navigating through our campus sure are not something minor, we cannot depend on the 2D static map since it is not available everywhere, and it became outdated, with augmented reality technologies, and our skills we are going to provide the solution for this problem so that students and doctors can easily and efficiently get to their class without wasting time with the convenience of their phones.

CHAPTER III
Methodology

3.1 Introduction

The methodology is a set of steps aimed at achieving a specific goal. These steps are a set of workflow methods applied throughout the life cycle of a software project. The goal of applying the methodology is to organize the work among team members so that they can develop the system in the best possible way. This chapter discuss the methodology that will be used in the development phases of the project.

3.2 Methodology

This section presents the proposed methodology for the development of indoor navigation systems for mobile devices. It integrates AR and Semantic Web technologies as methods for navigation and contextual information consumption. According to the components stated in the related work for the development of indoor navigation systems, this proposal is based on seven phases presented in a sequential order in which the methodology is performed: (a) project planning, (b) Project requirement, (c) design the project, (d) Coding and development, (e) Testing, (f) Updating and maintenance.

3.3 Methodology Chosen

The waterfall methodology is a model for software development projects, commonly known as the software development life cycle (SDLC). It is utilized by software and product development teams to effectively manage projects. This methodology divides the project into various phases, outlining the necessary activities and steps for each phase. For instance, in the initial stages, the waterfall methodology focuses on gathering all the requirements from stakeholders, which the project team will later utilize to design and implement the product. The waterfall methodology is a widely employed project management technique that follows a sequential process. It requires the completion of each phase before progressing to the next one. Although there are various project management approaches available, the waterfall method is particularly suitable for projects with well-defined objectives from the start [17].

3.3.1 Waterfall Methodology

The waterfall method consists of six distinct phases, with each phase depending on the deliverables of the previous phase. This methodology is especially effective for predetermined outcomes, such as software development projects. Figure 3-1 shows the waterfall methodology stages.

Advantages [18]:

- The methodology follows a clear sequential model for each phase and requirements are clearly defined in the initial stages and remain constant throughout the development life cycle.
- Each phase has specific outcomes and this ensures clarity in the project team's workflow.
- A clear structure and well-defined goals and deliverables make it easy to manage projects using the waterfall methodology.
- Early planning ensures a clear understanding of what needs to be achieved and how to approach it.
- Potential problems during implementation can be easily predicted.

Disadvantages [18]:

- It may be expensive and inflexible.
- Changes or additions made late in the process may require going back to all previous stages, resulting in increased costs.
- This methodology restricts the ability to conduct reviews or reflect on progress until each phase is completed.
- Testing is postponed to the final stage of system testing in the waterfall methodology.

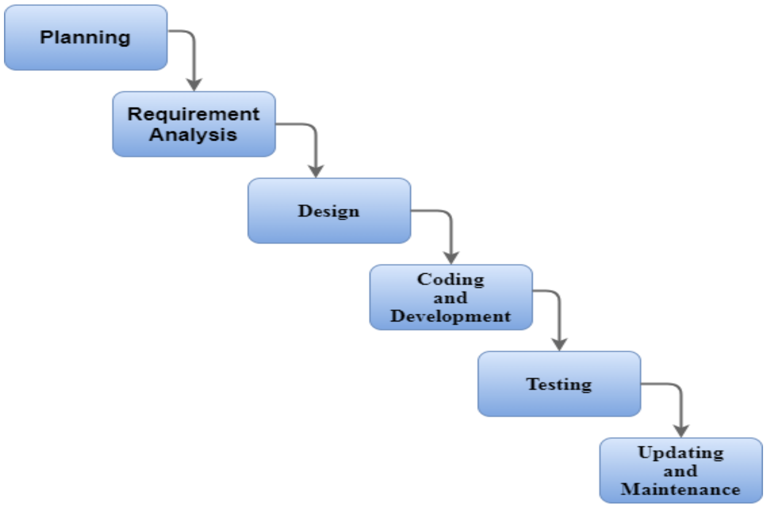

Figure 3-1: Waterfall Methodology Stages.

3.3.1.1 Project Planning Phase

In this phase, objectives and goals are set to be done by a specific time for every single one. Setting a timeline for every part of the project, needs and requirements will gather to design and code the application, The project team establishes a robust project plan that not only defines the scope and expected deliverables but also identifies the critical resources required for successful implementation.

3.3.1.2 Analysis of Project Requirement Phase

In this phase, the gathered requirements are carefully examined, the team will gather information needed such as the map, the offices and classrooms numbers and location, from the university projects agency, also measuring the distance between the rooms. the application should provide the best experiment for the user.

3.3.1.3 Design Phase

Once we knew our requirements, we moved to the design phase. We designed the system to the user with multiple choices, from setting the location to choose the destination, User experience design is a crucial component, the system shows a path while interaction from the user the instructors change, using Figma, Lunacy, and Unity to make the 3D structure of the building and floors,

3.3.1.4 Coding and Development Phase

With a clear plan and design in place, we proceed to make the plan on realty, Using C# coding and developing the Navigation System, our team work its best to make it an easy useful app to use , The objective is to provide the system with the information and algorithms necessary to locate the user location and choose the short route to lead into the destination .

3.3.1.5 Testing Phase

During this stage, after we had a functional prototype. We wanted to make sure it was user-friendly, that it worked well, and that it met everyone's expectations, the final software of the application system undergoes testing. The application will be used many times to see how it probably works and if there any issue to be fixed, the team will do unexpecting situations to see how the system response.

3.3.1.6 Updating and Maintenance Phase

After testing the Navigation System, there are any system errors that may arise, which could include upgrading the application or resolving software bugs. It marks the final phase of the software development process, aimed at enhancing the system's performance and stability.

3.4 Justification of Choosing Methodology

The waterfall model is recognized for its structured and effective nature, enabling all team members to thrive when following it. Project managers widely embrace the waterfall model for its strong and dependable approach, leading to streamlined workflows and increased productivity within teams.
Waterfall projects exhibit a high level of process definition and minimal output variability. Additionally, the waterfall model is a suitable choice when projects are constrained by cost or time. Projects based on the waterfall model are well-defined, predictable, and accompanied by specific documentation. The waterfall methodology represents a project management approach that emphasizes a sequential progression from the project's initiation to its completion. This methodology, commonly utilized by engineers, prioritizes thorough planning, detailed documentation, and consecutive execution [19].

3.5 Summary

This section we presented a detailed explanation of the methodology used in developing the proposed project, explaining the reasons for its use, its advantages and disadvantages, and an explanation of each stage.

CHAPTER IV
SYSTEM ANALYSIS AND DESIGN

4.1 Introduction

In this chapter, we will delve into the system analysis and design phase of our project. This phase is crucial for understanding the requirements of the software, identifying functional and non-functional requirements, as well as interface requirements. By thoroughly analyzing and designing the system, we lay the foundation for developing an effective AR navigation solution.[20]

4.2 Result of Software Requirement

Before proceeding with the analysis and design, it is essential to summarize the software requirements gathered during the requirements elicitation phase. These requirements serve as the basis for the subsequent steps in the system development process. The software requirement document includes both functional and non-functional requirements, as well as interface requirements. [21]

4.2.1 Functional Requirement

Functional requirements outline the specific functionalities and features that the AR navigation system needs to provide. These requirements are typically derived from user needs and expectations. Here the functional requirements identified for the AR navigation project:

1. GPS-based navigation: The system should accurately track the user's location and provide real-time navigation guidance using augmented reality overlays. And ask the user for permission to access their location [22].

2. Route planning and optimization: The system should analyze the user's destination and current location to determine the most efficient route and provide step-by-step guidance [22].

3. Spatial mapping and object recognition: The system should utilize computer vision techniques to map the physical environment and recognize beacon in real-time, ensuring accurate placement of AR elements [22].

4. Camera Permissions: The system should be able to ask the user for permission to access their camera.

4.2.2 Non-Functional Requirement

Non-functional requirements define the qualities and constraints that the AR navigation system must possess. the non-functional requirements identified for the AR navigation project include:

1. Performance: The system should provide smooth and responsive AR overlays, with minimal latency between the user's movements and the rendered visuals [23].

2. Usability: The system should have an intuitive and user-friendly interface, making it easy for users to interact with the AR navigation features and settings [23].

3. Reliability: The system should be reliable and capable of functioning in various environmental conditions, such as low-light or crowded areas [23].

4. Security: The system should ensure the privacy and security of user data, including location information and personalized settings.

5. Compatibility: The system should be compatible with a wide range of mobile devices and operating systems, ensuring broad accessibility for users.

4.2.2.1 Security for our App

1. Secure Backend Infrastructure
   - Securing the backend infrastructure, including servers and databases, with regular security patches, firewalls, and intrusion detection systems.
   - Implementing strong access controls to protect against attacks. [25]

2. Secure AR Content
   - Implementing content moderation techniques to filter and review user-generated AR content.
   - Preventing the display of malicious or inappropriate content.[26]

3. Secure User Inputs
   - Validating and sanitizing user inputs to prevent common security vulnerabilities.
   - Protecting against attacks such as SQL injection, cross-site scripting (XSS), and cross-site request forgery (CSRF).[25]

4. Regular Security Audits and Testing
   - Conducting regular security audits and penetration testing to identify and address vulnerabilities.
   - Manual and automated testing techniques to ensure the effectiveness of security controls.[25]

5. Implement Privacy Controls:
   - Provide users with granular control over camera permissions within your app's settings. Allow them to enable or disable camera access as per their preferences.[25]

4.2.3 Interface Requirement

Interface requirements define the interactions between the AR navigation system and its users, as well as any external systems or devices. These requirements focus on the usability and integration aspects of the system. The interface requirements identified for the AR navigation project include:

1. User interface (UI): The system should have an intuitive and visually appealing UI that provides easy access to navigation controls, and information [24].

2. Integration with external systems: The system should be able to integrate with external data sources, such as GPS services, mapping APIs, or POI databases [24].

3. Input methods: The system should support different input methods, such as touch gestures, or physical buttons, to accommodate user preferences.

4. Error handling and feedback: The system should provide clear and informative error messages, as well as visual or auditory feedback to guide users during navigation [24].

4.3 Use Case Diagram

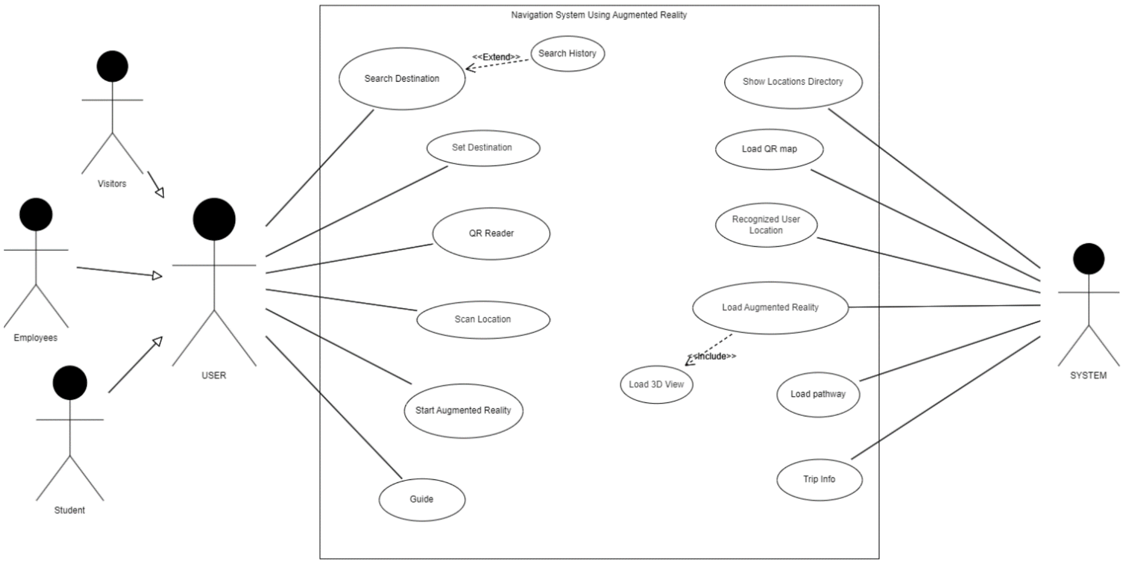

Figure 4-1: The Use Case diagram.

4.3.1 Use Case Description (figure 4-1)

- Search Destination: the user writes the destination name such as building name, classrooms, offices and so on.
- Search History: in this option the user can find what they have searched for before.
- Set destination: the user will set the destination.
- Start augmented reality: the user will start to view the augmented reality of the map or the path instructions to the destination selected.
- QR Reader: The user scans QR of the campus, Buildings, libraries, map etc.
- Scan Location: the user can get benefit of this feature, once he seems to be lost, he scans the area, and the system will recognize the location
- Guide: the user can read instructions on how to use the application.
- Show location directory: the system will show to the user the locations and suggestions based on what he writes.
- Trip Info: The system will provide the user with the distance to the location and the estimated time to arrive.
- Load QR: the system will load to the user the map or the destination in the QR.
- Recognized user location: the system will scan the user location and recognize the place to set the go point
- Load Augmented Reality: The system displays the augmented reality of the selected map or destination and makes it available for the users to view.
- 3D View: by using AR the system will show the user the path and in a 3D view.
- Load Pathway: The system will show to the user the path of the chosen destination accurately in real-time directions and personalized route suggestions as in Figure 4-1.

4.4 Analysis Phase

The Analysis Phase in software development is a crucial step where the system's requirements are formulated and formalized. we will focus on three outlines:

- Observation: it provides valuable and accurate information that other methods, such as interviews, surveys, or workshops may not capture. By conducting observations of users in their natural environments, you could acquire valuable insights into their behaviors and motives.

- Risk Assessment: we will assess the likelihood and impact of potential risks and determine the most appropriate and cost-effective risk treatment method.

- Prototyping: by prototyping will be able to gather information about the system and its requirements, also it enhances speed, communication, and validation during the Analysis Phase.

4.4.1 Our Prototype

We designed a prototype and first thing we did is to draw it on the papers because it particularly useful during this early stage and then we used Figma to improve it. Here the first prototype and the last one that we did using Figma, as shown in Figure 4-2:

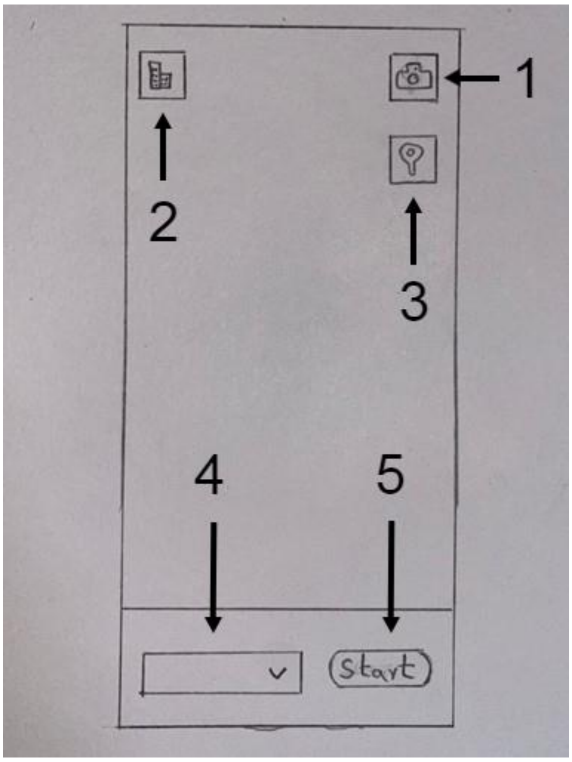

Figure 4-2: The first prototype.

1. Camera button: to scan QR codes.
2. Building button: to choose the floor level.
3. Location button: shows the current location.
4. Drop List section: to choose the classroom or office of the chosen floor.
5. Start button: when clicked the navigation start.

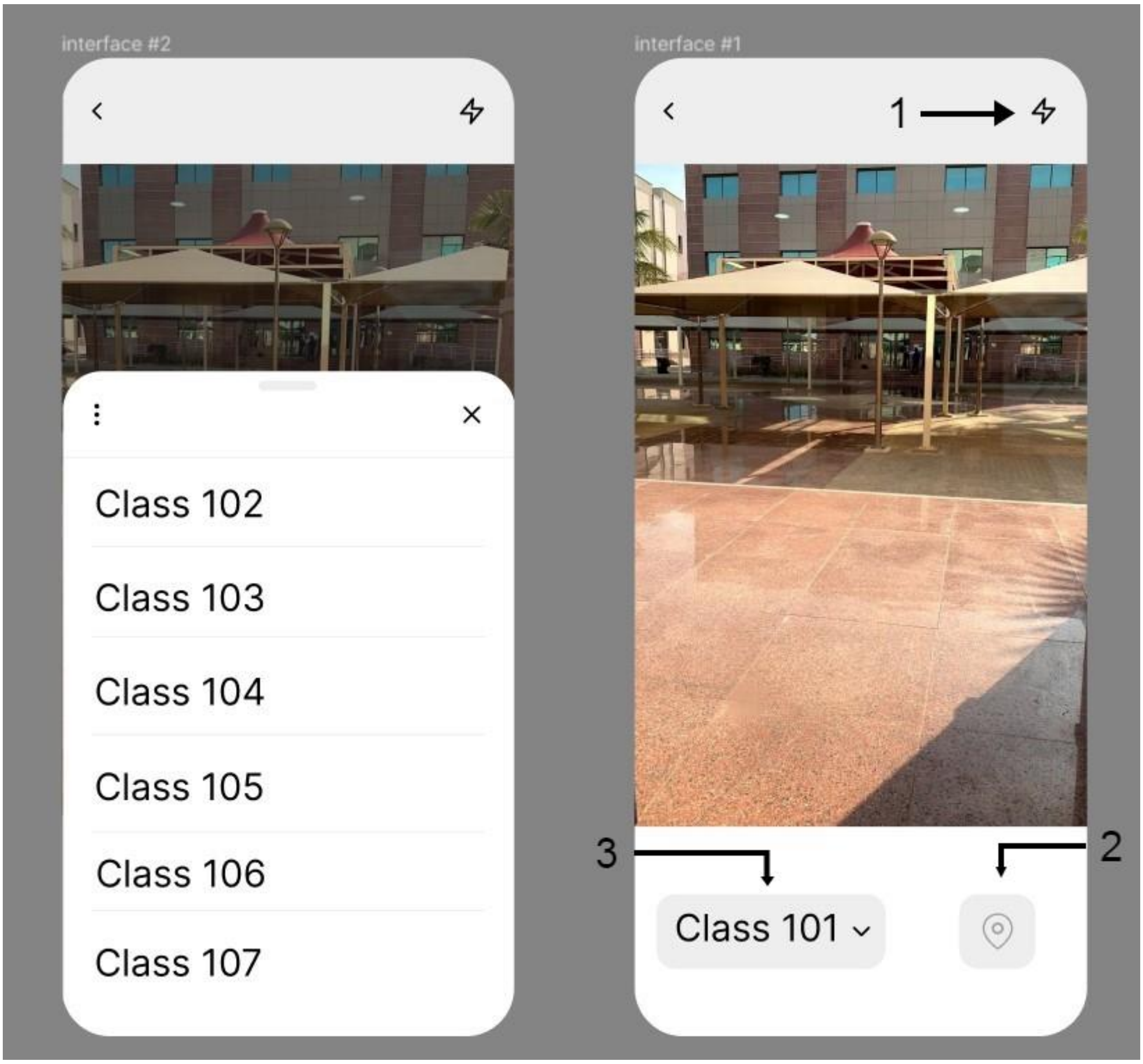

Figure 4-3: The Figma prototype.

We can notice the improvement in the interface's effectiveness as we developed the destination drop list, as well as when user selects the destination, the application starts without the need to press an additional button as shown in Figure 4-3.

1. Flashlight button: turn on the flashlight.
2. Location button: shows the current location.
3. Drop list section: to choose the classroom or office of the chosen floor.

4.4.2 **Sequence Diagram** (figure 4-4)

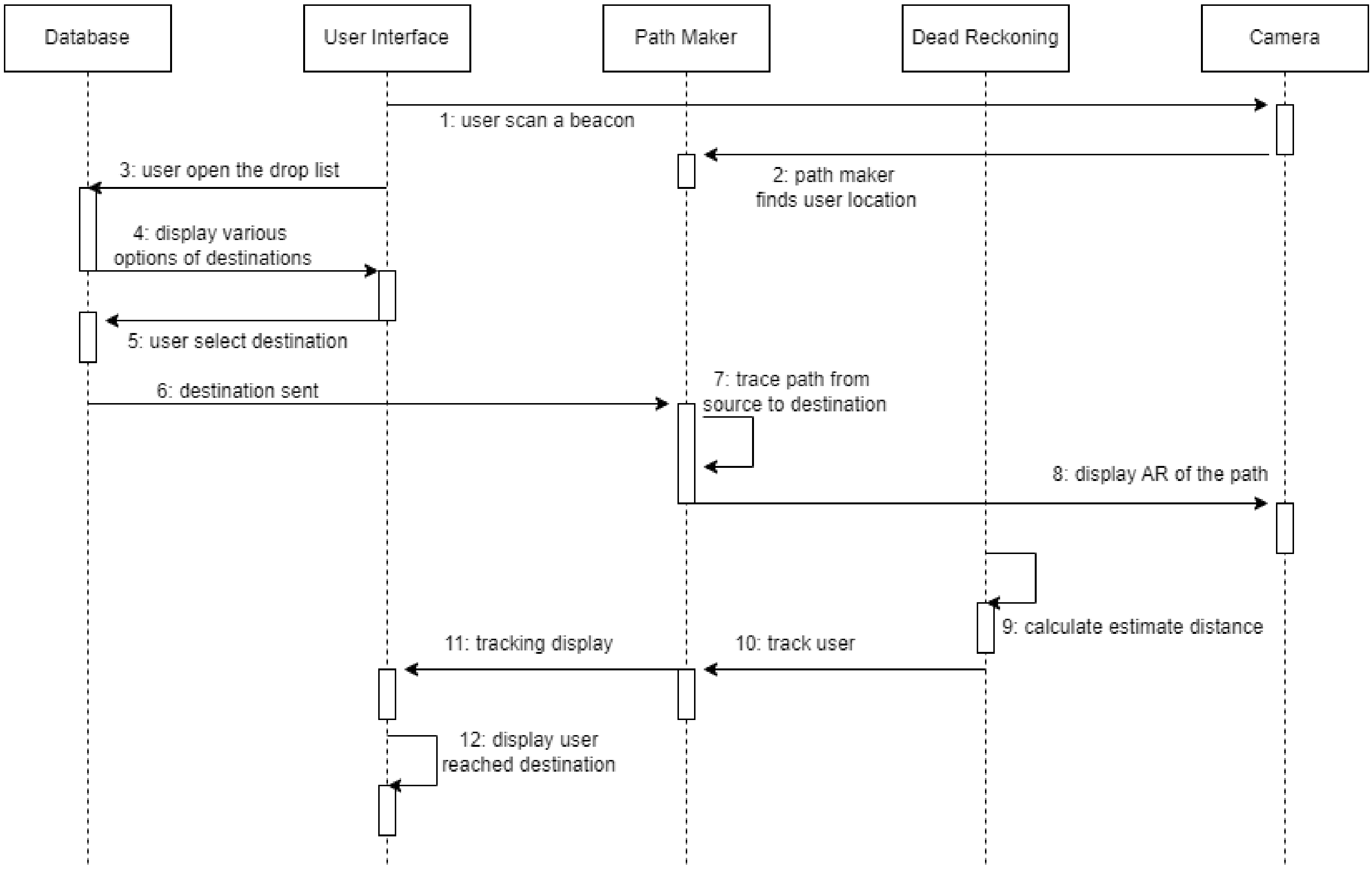

Figure 4-4: The Sequence Diagram.

4.5 Architecture Design Phase

Architecture design for a college building with an AR navigation system involves integrating augmented reality technology and spatial considerations to enhance navigation and wayfinding. The design encompasses infrastructure for AR devices, accurate indoor mapping, real-time tracking, intuitive user interfaces, and visual cues. Points of interest and accessibility considerations are incorporated, along with integration with existing systems and reliable connectivity. Testing, maintenance, and regular updates ensure the system remains accurate and user-friendly, ultimately enhancing the navigation experience for students, faculty, and visitors as they navigate the college building.

4.5.1 System Architecture and Design

For example, we design a navigation map for our college building that show how the system will use beacons to find user and set a path via the beacons or points, here a simple map we did and some paths that the user may take as shown in Figure 4-5:

Figure 4-5: The ground floor of FCITR building.

4.5.1.1 paths examples (table 4-1)

Source	Destination	No. Beacons	Beacons Path	Distance
A	F	4	3-4-5-6	14m
C	G	5	10-9-8-7-6	26m
B	E	5	3-11-10-9-8	26m
A	B	1	3	0m
D	A	4	9-10-11-3	24m

Table 4-1: Some Paths that may the user take.

4.5.2 Database Design

The design of our database consists of the number of halls, buildings, and the number of beacons or points. We have prepared a simple example of how we will write our database in the system in table 4-2:

Buildings	No. Classes	No. Beacons	No. Other Rooms
Building 1	10	20	28
Building 2	10	18	26
Building 3	12	24	25

Table 4-2: simple example of our database design.

4.6 Summary

In summary, this phase plays a pivotal role in software development. It serves as the foundation for understanding software requirements and collecting precise information. Additionally, a key focus is on thoroughly analyzing user behavior and motivations. By doing so, we can lay the groundwork for successful software design and implementation.

CHAPTER V
SYSTEM IMPLEMENTATION

5.1 Introduction

This chapter centers on the results and discussions on the evolution and deployment of our Android augmented reality (AR) navigation system. The primary goal is to carefully review the data acquired during the installation process in order to evaluate the performance, usability, and general efficiency in reaching our initial project goals of app.

First, we will start by looking at the implementation details that is, the particular technical aspects and techniques applied all during the development process. This covers the basic instruments and technologies that were absolutely necessary to realize the project.

We will then go over the development environment, including particular details on the hardware and software choices that helped our work be supported. An outline of the development tools, programming languages, and outside libraries included into the project will be given in this part.

We will next shortly provide a succinct overview of the whole system development process. Beginning from requirement analysis and design and working through the whole system development lifecycle, this part will help you to reach coding. It will especially stress the waterfall method we have selected to employ.

Our focus then will turn to the evolution of the user interface. To provide a user-friendly and understandable experience, we will investigate the design ideas and elements influencing our choices. Covered in this part will be the specific user interface (UI) components and feedback mechanisms created to enhance user engagement.

We will use application screenshots to offer a graphic picture of our written explanations. These pictures help to graphically show the salient features and functions of the program.

5.2 Implementation

Creating the AR navigation app for Android was a multifarious and challenging project including several important phases and application of several modern technologies. This is a comprehensive review of our approach for creating the app:

Using Unity's strong augmented reality features, we developed and designed a complete 3D model of the Faculty of Computing and Information Technology from first scratch. Integration of ARCore features was very beneficial using the AR Foundation of Unity.

With their ARCore software development kit (SDK), Google controlled the augmented reality (AR) elements of the application, including motion tracking and ambient awareness.

C# is used in development of the backend services. This setup guaranteed dependability, scalability, and effective data storage.

Using Unity's UI framework, the UI creation guarantees a consistent and flexible design among many Android devices.

Real-time updates were sent using QR codes, therefore assuring that user had the most current routing information without any lag.

5.3 Development Environment

The development environment, which consists of the tools, technologies, and configurations used to create the application, is vital to every software project. For our AR navigation software, we purposefully selected a development environment that would enable efficient coding, testing, and deployment. Here is a detailed breakdown of the components that make up our development environment:

5.3.1 Hardware

High-performance PCs with robust GPUs helped us manage the heavy graphics processing required for AR development. Smooth running of Unity and other development tools depended on these machines, which also were absolutely essential.

Testing Devices: The app was tested using a range of Android handsets. This covered hardware capabilities to guarantee compatibility and performance throughout a broad spectrum of models as well as devices with various resolutions.

5.3.2 Software

Unity was the main development platform AR navigation app used on. Its strong AR capabilities and AR Foundation architecture helped us to quickly create and apply AR experiences.

Writing and backend code management took place in Visual Studio. Its compatibility with Unity and support of C# made it the perfect solution for our current development requirements.

5.3.3 Libraries

Implementing the AR functionality required Google's ARCore SDK. It gave the required instruments for light estimate, ambient awareness, and motion tracking.

Different Unity plugins were applied to improve the capacity of the application. These included extras AR features, performance improvements, and UI design plugins as shown in Figure 5-1.

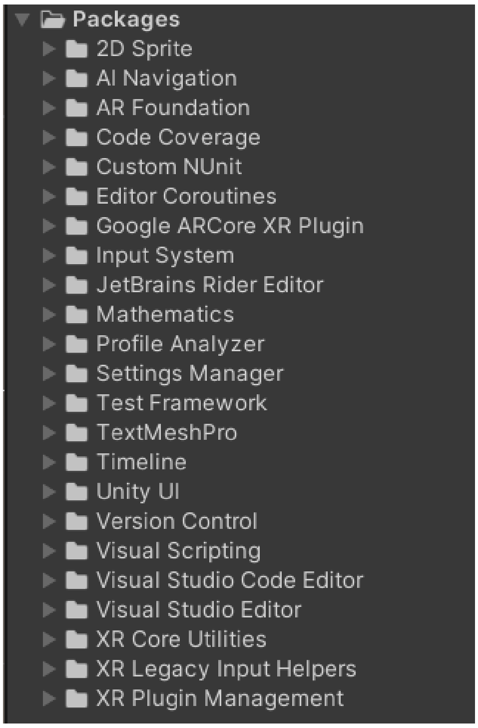

Figure 5-1: A look at the packages that were used.

5.4 Overall System Development

The development of the AR navigation app for Android was a thorough and methodical procedure comprising gathering requirements and developing user-friendly interface as well as coding and including 3D modeling. The following summarizes our approach to it:

We started by gathering exact needs. This included a focus group to find out what consumers required from an augmented reality navigation system. The knowledge acquired guided our definition of the main characteristics and capabilities of the app.

The application developed using C# utilizing Microsoft Visual Studio 2022 and unity 3D modeling for the faculty building was Maintaining high standards, regular code reviews and testing were carried out.

5.5 User Interface Development

The face of the app is the user interface (UI), therefore we aimed to make it as user-friendly and understandable as feasible. We addressed the UI development as follows:

Our main objective was to produce a straightforward and understandable UI. We concentrated on simple design ideas to keep the app clutter-free and guarantee users' easy navigation. Important components were meant to be easily reachable and comprehensible.

We used basic navigation controls so that people may begin their trip with easy motions. Users travelled along their path guided by visual cue including a red line. Designed to be simple and responsive, this cue offers real-time input as users traverse their surroundings.

5.6 Snapshots

In this section, we provide visual representations of the development process of the AR navigation app within Unity. These snapshots illustrate key stages and components of the app's development, offering a clear view of how the app was built and refined.
These snapshots show the Unity Editor interface, highlighting the various panels and tools used during development in Figure 5-2 to 5-12:

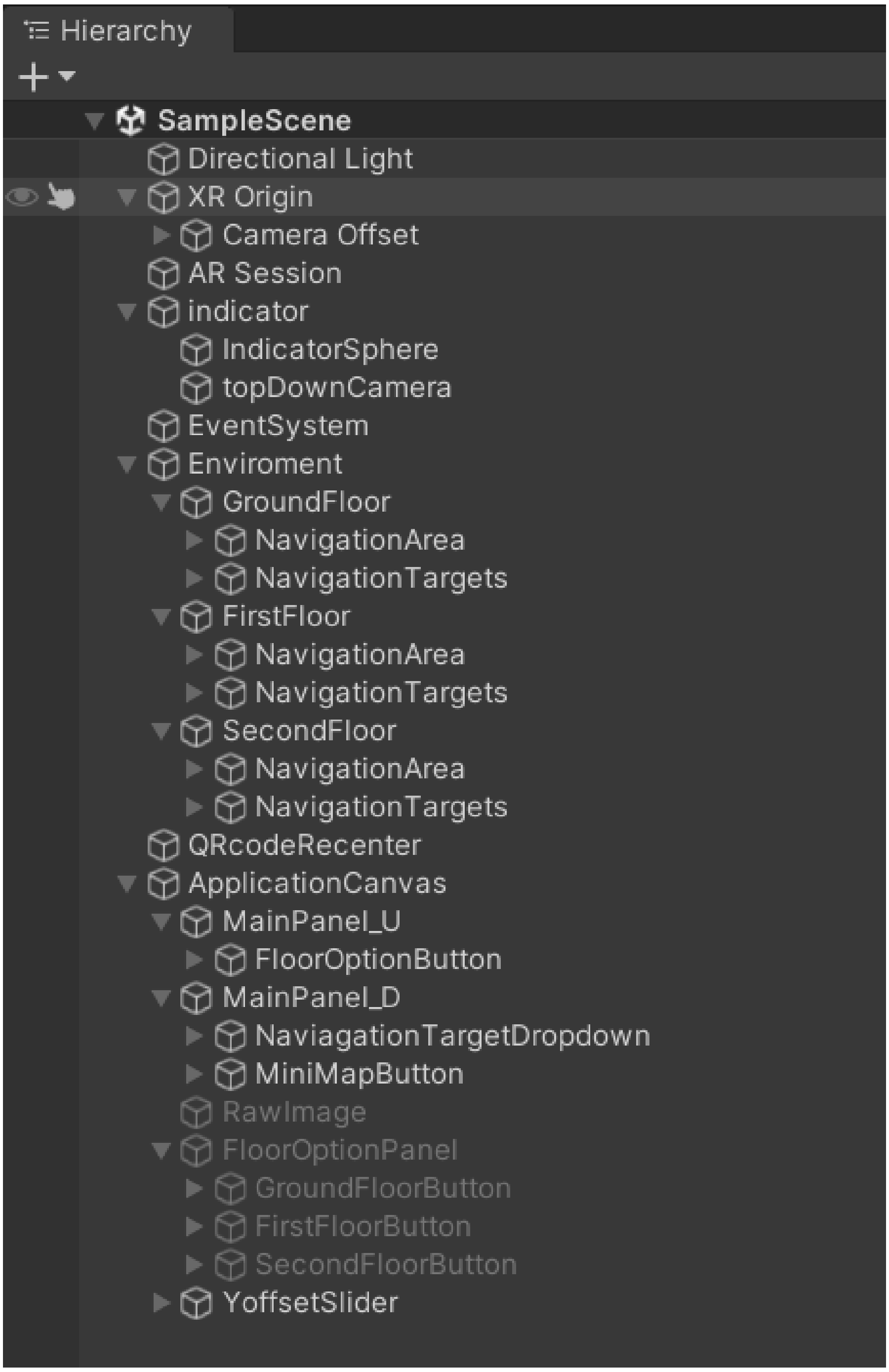

Figure 5-2: The objects hierarchy inside unity.

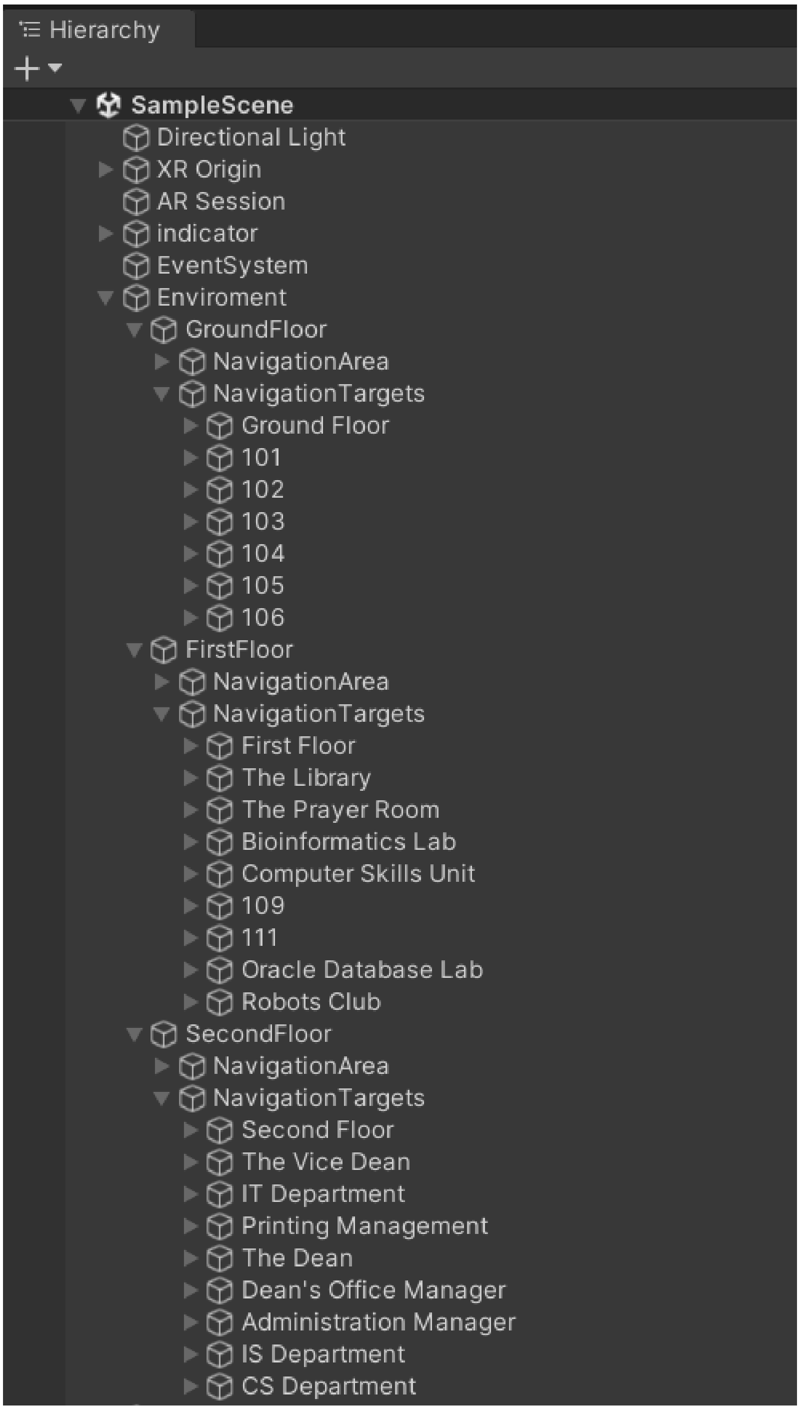

Figure 5-3: An expanded view of the navigation targets.

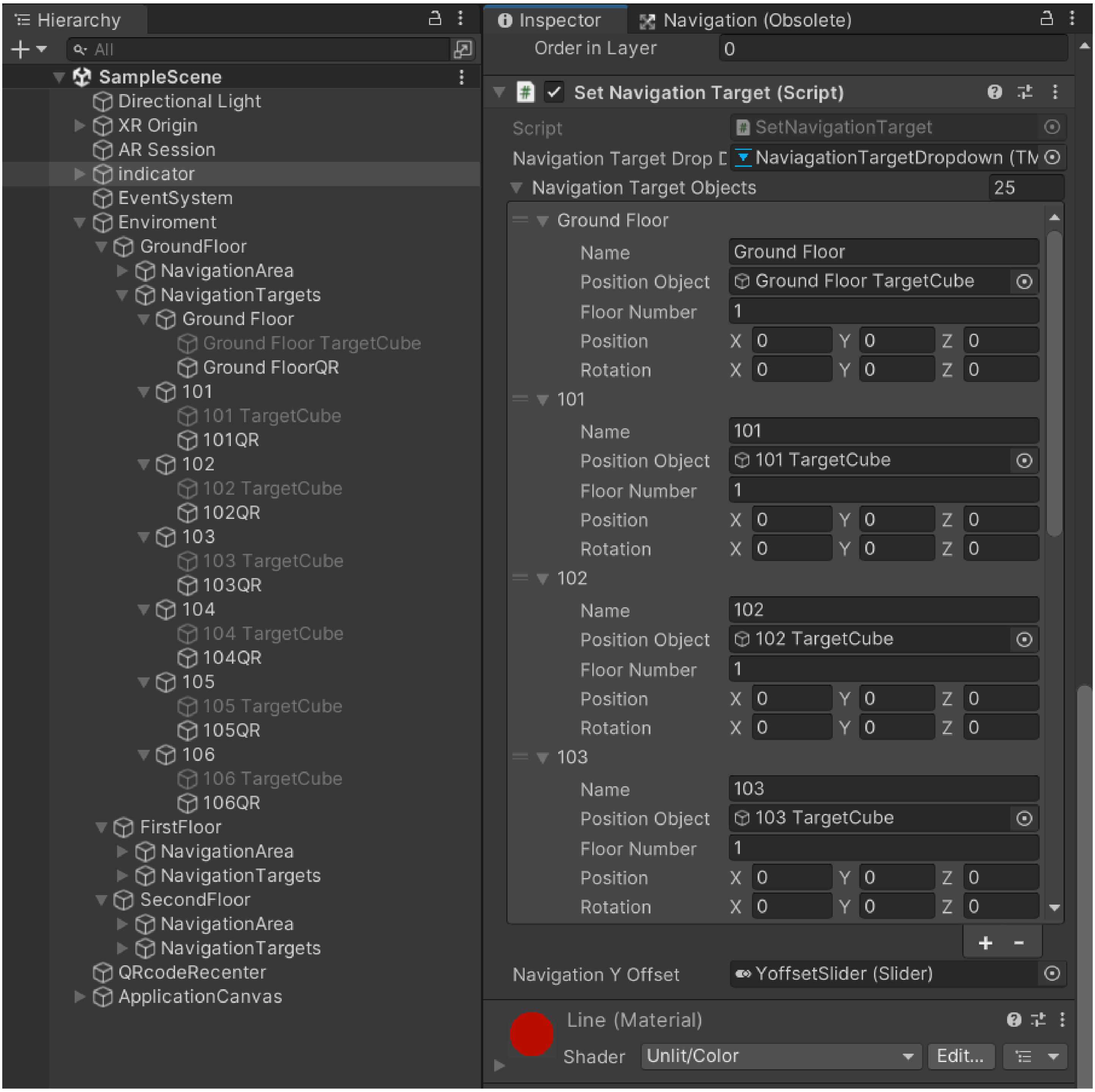

Figure 5-4: An expanded view of the indicator.

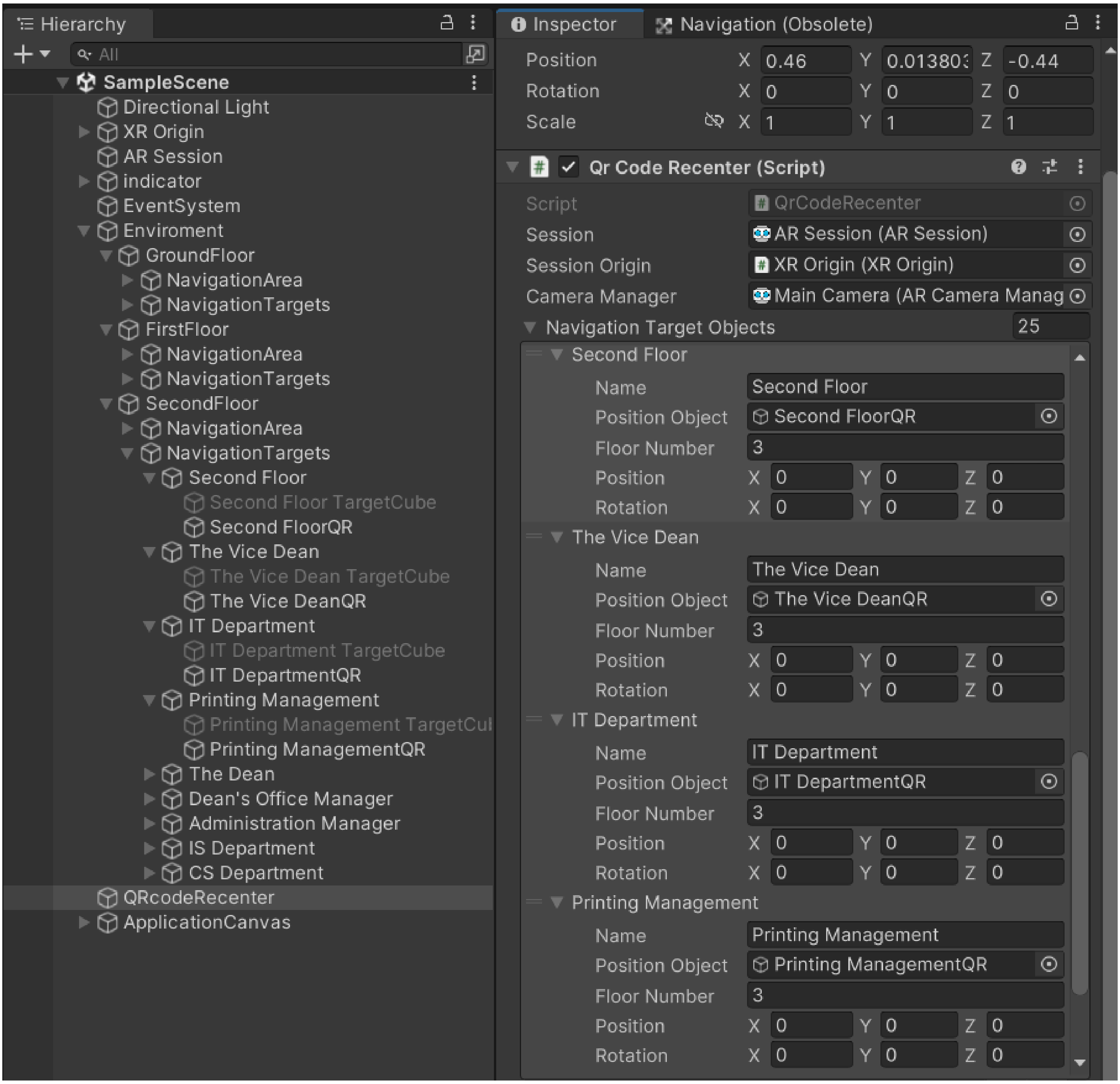

Figure 5-5: An expanded view of the QR code recenter.

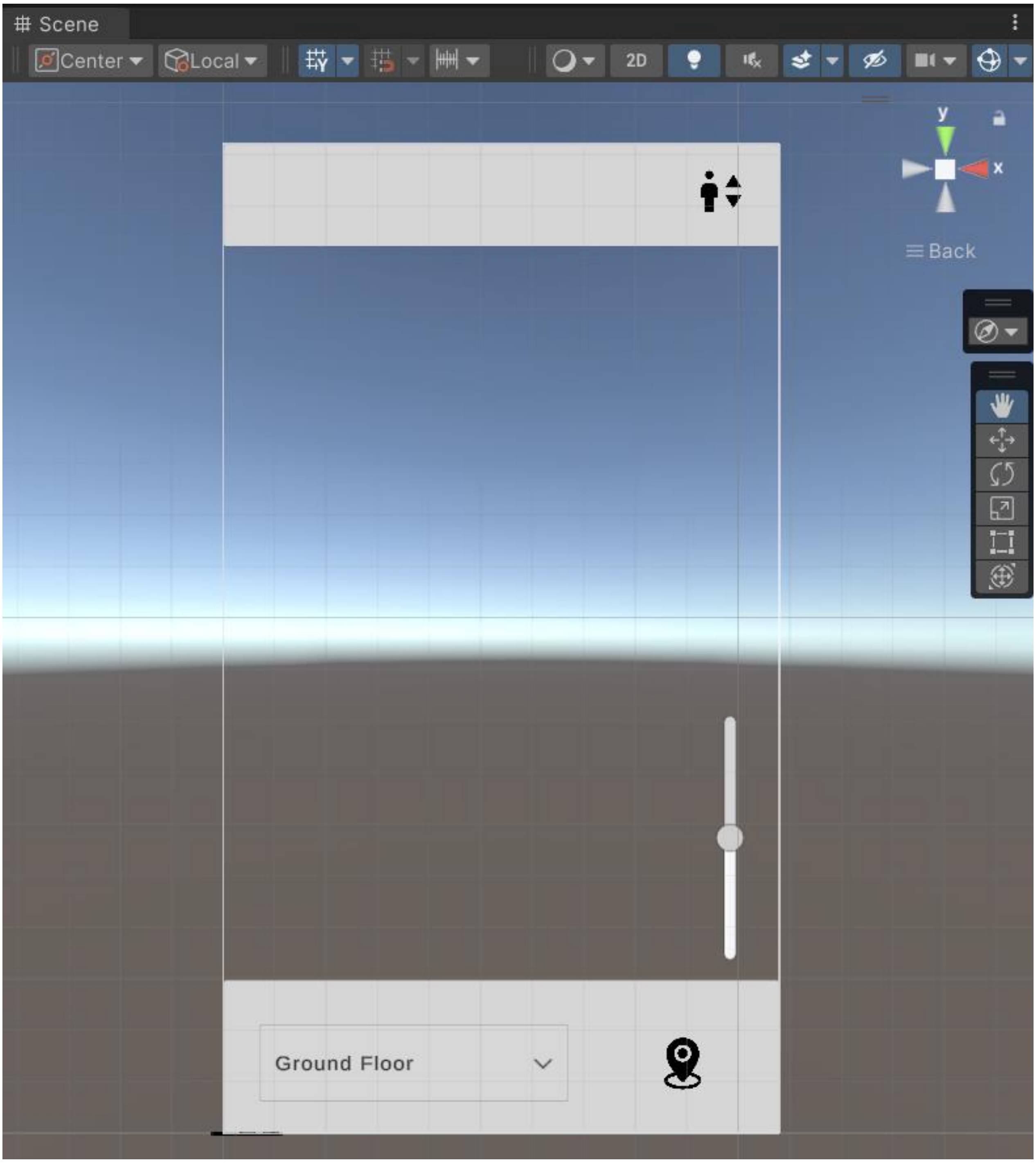

Figure 5-6: The user interface design.

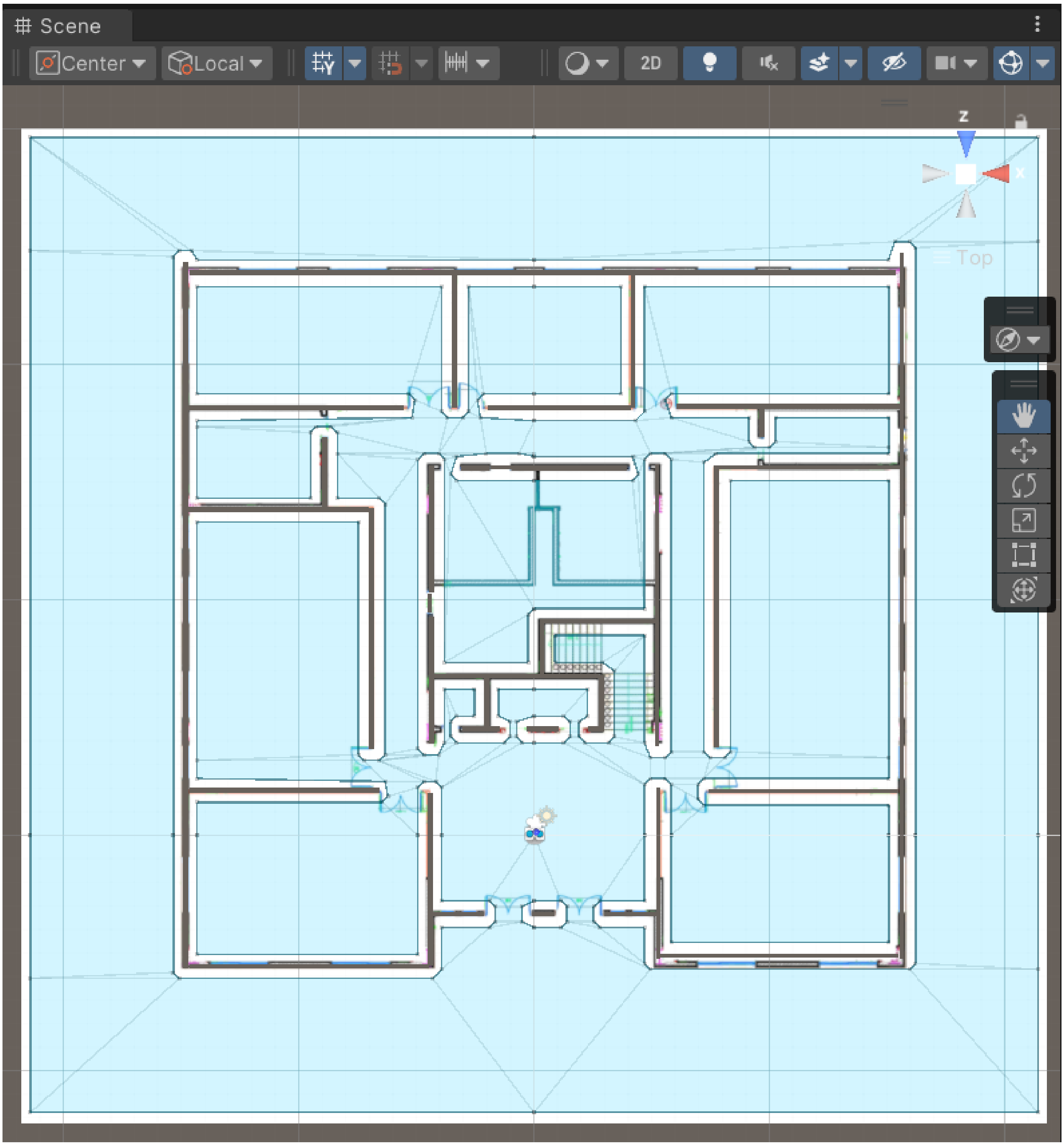

Figure 5-7: The ground-floor navigation design.

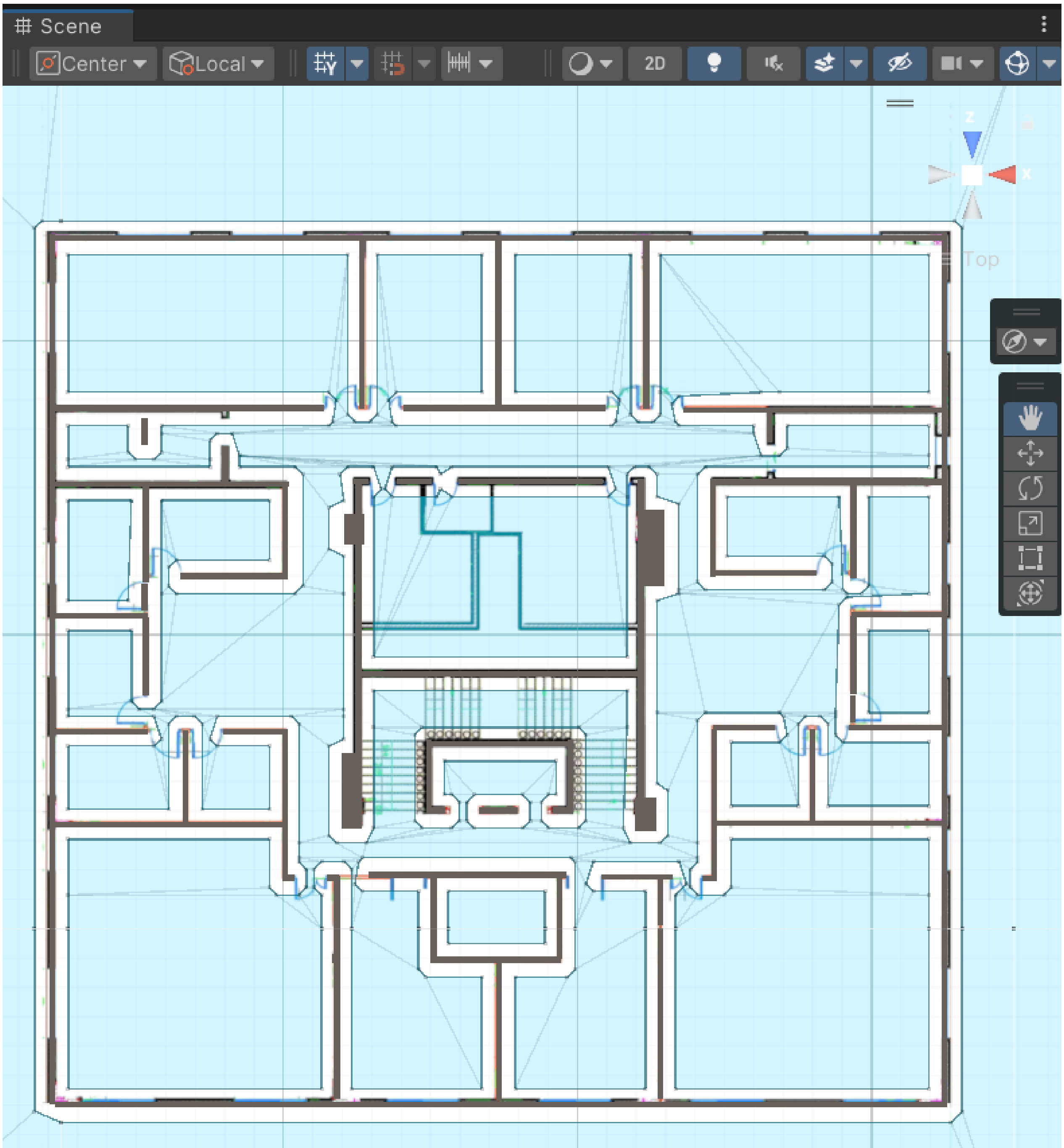

Figure 5-8: The First-floor navigation design.

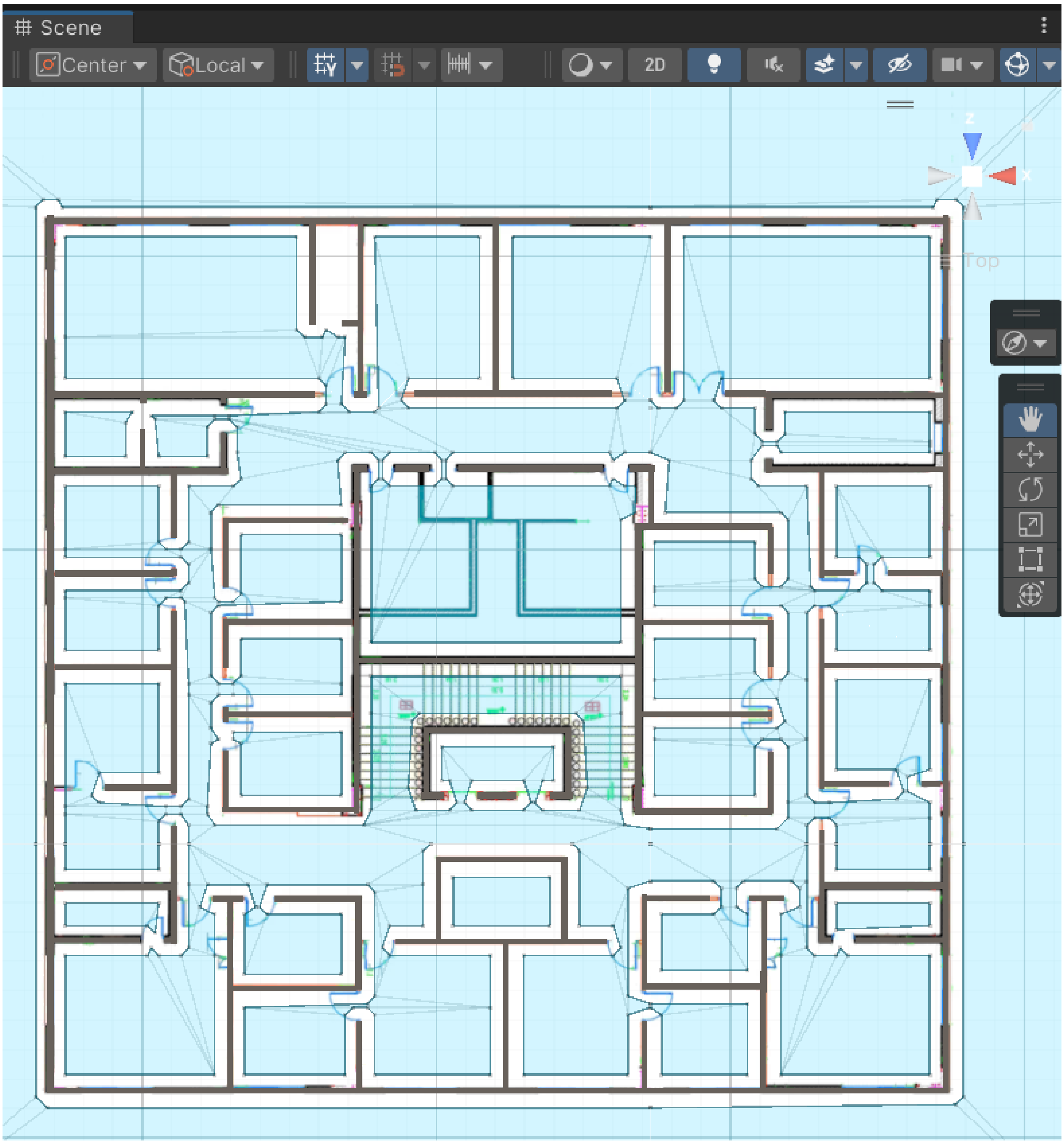

Figure 5-9: The second-floor navigation design.

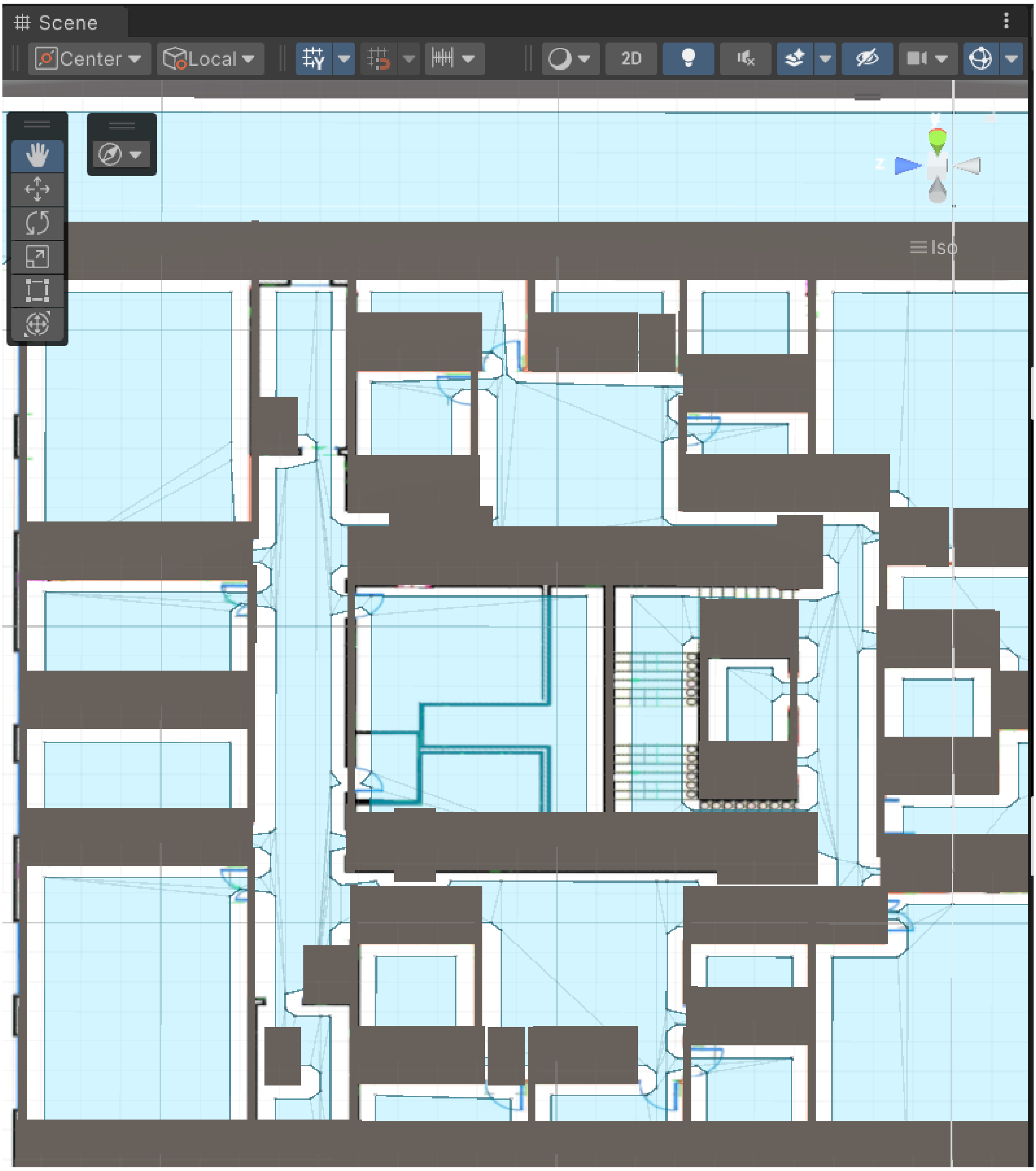

Figure 5-10: The first-floor navigation design in 3D.

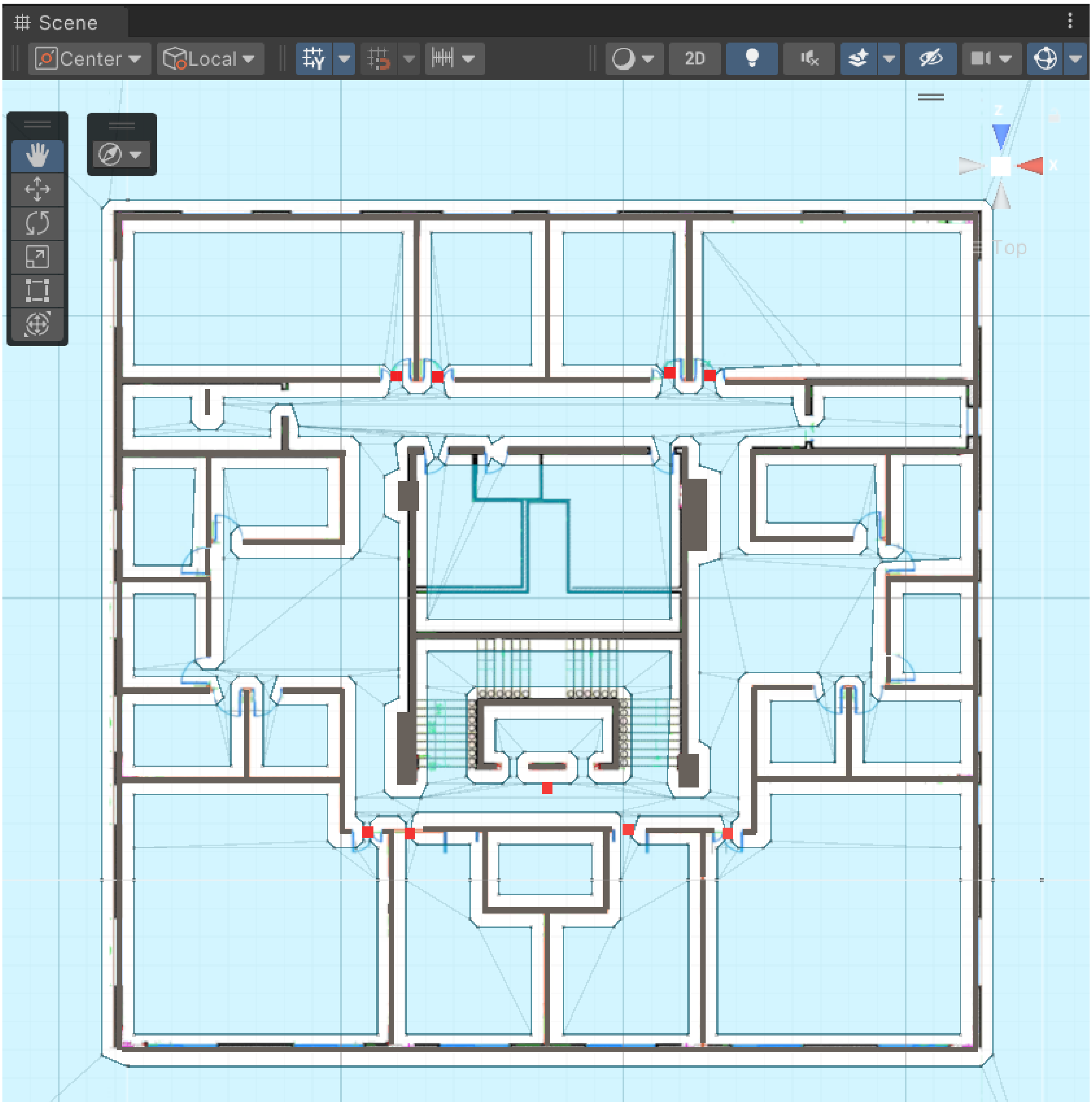

Figure 5-11: The first-floor navigation target objects.

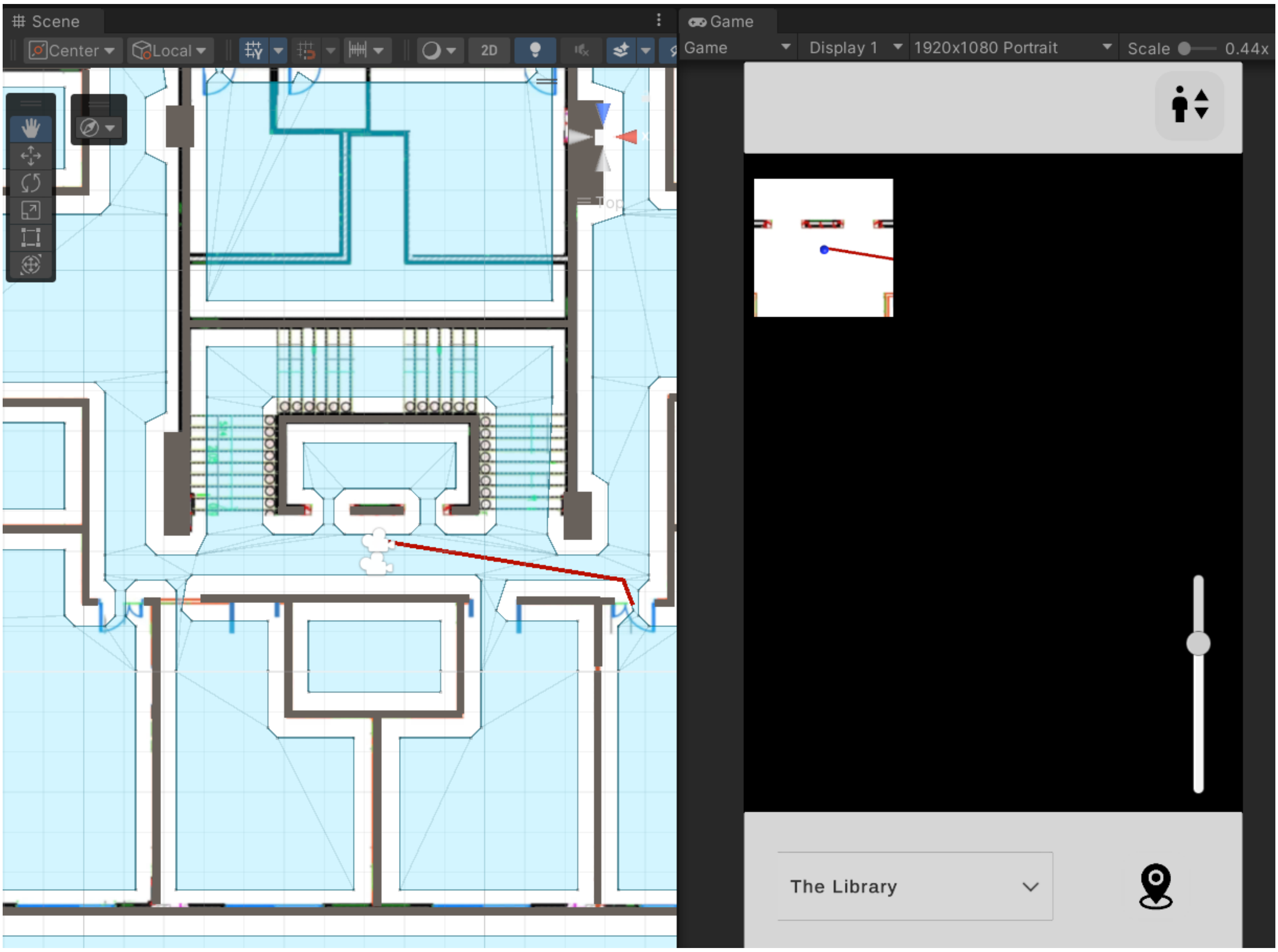

Figure 5-12: Simple run test inside unity.

5.7 Code Implementation

We have in this part thorough codes that were really essential in creating the AR navigation app. Future development or troubleshooting may find this part helpful as well as a reference for the technical execution as shown in Figure 5-13.

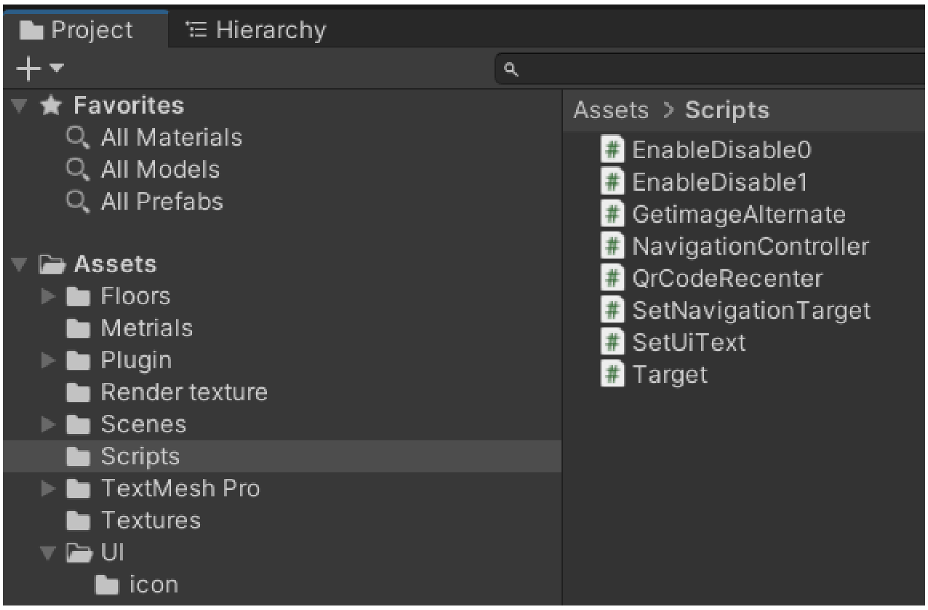

Figure 5-13: A view of the project's script files.

5.7.1 Navigation Target

```
using System.Collections.Generic;
using TMPro;
using UnityEngine;
using UnityEngine.AI;
using UnityEngine.UI;

public class SetNavigationTarget : MonoBehaviour
{
    [SerializeField]
    private TMP_Dropdown navigationTargetDropDown;
    [SerializeField]
    private List<Target> navigationTargetObjects = new List<Target>();
    [SerializeField]
    private Slider navigationYOffset;
    private NavMeshPath path;
    private LineRenderer line;
```

```
private Vector3 targetPosition = Vector3.zero;
private int currentFloor = 1;
private bool lineToggle = false;
private void Start()
{
    path = new NavMeshPath();
    line = transform.GetComponent<LineRenderer>();
    line.enabled = lineToggle;
}
private void Update()
{
    if (lineToggle && targetPosition != Vector3.zero)
    {
        NavMesh.CalculatePath(transform.position, targetPosition, NavMesh.AllAreas, path);
        line.positionCount = path.corners.Length;
        Vector3[] calculatedPathAndOffset = AddLineOffset();
        line.SetPositions(calculatedPathAndOffset);
    }
}
public void SetCurrentNavigationTarget(int selectedValue)
{
    targetPosition = Vector3.zero;
    string selectedText = navigationTargetDropDown.options[selectedValue].text;
    Target currentTarget = navigationTargetObjects.Find(x => x.Name.Equals(selectedText));
    if (currentTarget != null)
    {
        if (!line.enabled)
        {
            ToggleVisibility();
        }
     targetPosition = currentTarget.PositionObject.transform.position;
```

```
    }
}
public void ToggleVisibility()
{
    lineToggle = !lineToggle;
    line.enabled = lineToggle;
}
public void ChangeActiveFloor(int floorNumber)
{
    currentFloor = floorNumber;
    SetNavigationTargetDropDownOptions(currentFloor);
}
private Vector3[] AddLineOffset()
{
    if (navigationYOffset.value == 0)
    {
        return path.corners;
    }
    Vector3[] calculatedLine = new Vector3[path.corners.Length];
    for (int i = 0; i < path.corners.Length; i++)
    {
        calculatedLine[i] = path.corners[i] + new Vector3(0, navigationYOffset.value, 0);
    }
    return calculatedLine;
}
private void SetNavigationTargetDropDownOptions(int floorNumber)
{
    navigationTargetDropDown.ClearOptions();
    navigationTargetDropDown.value = 0;
    floorNumber = currentFloor;
    if (line.enabled)
```

```
{
    ToggleVisibility();
}
if (floorNumber == 1)
{
    navigationTargetDropDown.options.Add(new TMP_Dropdown.OptionData("Ground Floor"));
    navigationTargetDropDown.options.Add(new TMP_Dropdown.OptionData("101"));
    navigationTargetDropDown.options.Add(new TMP_Dropdown.OptionData("102"));
    navigationTargetDropDown.options.Add(new TMP_Dropdown.OptionData("103"));
    navigationTargetDropDown.options.Add(new TMP_Dropdown.OptionData("104"));
    navigationTargetDropDown.options.Add(new TMP_Dropdown.OptionData("105"));
    navigationTargetDropDown.options.Add(new TMP_Dropdown.OptionData("106"));
}
if (floorNumber == 2)
{
    navigationTargetDropDown.options.Add(new TMP_Dropdown.OptionData("First Floor"));
    navigationTargetDropDown.options.Add(new TMP_Dropdown.OptionData("The Library"));
    navigationTargetDropDown.options.Add(new TMP_Dropdown.OptionData("The Prayer Room"));
    navigationTargetDropDown.options.Add(new TMP_Dropdown.OptionData("Bioinformatics Lab"));
    navigationTargetDropDown.options.Add(new TMP_Dropdown.OptionData("Computer Skills Unit"));
    navigationTargetDropDown.options.Add(new TMP_Dropdown.OptionData("109"));
    navigationTargetDropDown.options.Add(new TMP_Dropdown.OptionData("111"));
    navigationTargetDropDown.options.Add(new TMP_Dropdown.OptionData("Oracle Database Lab"));
    navigationTargetDropDown.options.Add(new TMP_Dropdown.OptionData("Robots Club"));
}
if (floorNumber == 3)
{
    navigationTargetDropDown.options.Add(new TMP_Dropdown.OptionData("Second Floor"));
    navigationTargetDropDown.options.Add(new TMP_Dropdown.OptionData("The Vice Dean"));
    navigationTargetDropDown.options.Add(new TMP_Dropdown.OptionData("IT Department"));
    navigationTargetDropDown.options.Add(new TMP_Dropdown.OptionData("Printing Management"));
```

```
            navigationTargetDropDown.options.Add(new TMP_Dropdown.OptionData("The Dean"));
            navigationTargetDropDown.options.Add(new TMP_Dropdown.OptionData("Dean's Office Manager"));
            navigationTargetDropDown.options.Add(new TMP_Dropdown.OptionData("Administration
Manager"));
            navigationTargetDropDown.options.Add(new TMP_Dropdown.OptionData("IS Department"));
            navigationTargetDropDown.options.Add(new TMP_Dropdown.OptionData("CS Department"));
        }
    }
}
```

5.7.2 QR Code Recenter

```
using System.Collections.Generic;
using Unity.Collections;
using Unity.XR.CoreUtils;
using UnityEngine;
using UnityEngine.XR.ARFoundation;
using UnityEngine.XR.ARSubsystems;
using ZXing;

public class QrCodeRecenter : MonoBehaviour
{
    [SerializeField]
    private ARSession session;
    [SerializeField]
    private XROrigin sessionOrigin;
    [SerializeField]
    private ARCameraManager cameraManager;
    [SerializeField]
    private List<Target> navigationTargetObjects = new List<Target>();
    private Texture2D cameraImageTexture;
    private IBarcodeReader reader = new BarcodeReader(); // create a barcode reader instance
```

```
private void Update()
{
    if (Input.GetKeyDown(KeyCode.Space))
    {
        SetQrCodeRecenterTarget("101");
    }
}
private void OnEnable()
{
    cameraManager.frameReceived += OnCameraFrameReceived;
}
private void OnDisable()
{
    cameraManager.frameReceived -= OnCameraFrameReceived;
}
private void OnCameraFrameReceived(ARCameraFrameEventArgs eventArgs)
{
    if (!cameraManager.TryAcquireLatestCpuImage(out XRCpuImage image))
    {
        return;
    }
    var conversionParams = new XRCpuImage.ConversionParams
    {
        inputRect = new RectInt(0, 0, image.width, image.height),
        outputDimensions = new Vector2Int(image.width / 2, image.height / 2),
        outputFormat = TextureFormat.RGBA32,
       transformation = XRCpuImage.Transformation.MirrorY
    };
     int size = image.GetConvertedDataSize(conversionParams);
    var buffer = new NativeArray<byte>(size, Allocator.Temp);
    image.Convert(conversionParams, buffer);
```

```
        image.Dispose();
        cameraImageTexture = new Texture2D(
            conversionParams.outputDimensions.x,
            conversionParams.outputDimensions.y,
            conversionParams.outputFormat,
        false);
        cameraImageTexture.LoadRawTextureData(buffer);
        cameraImageTexture.Apply();
        buffer.Dispose();
        var result = reader.Decode(cameraImageTexture.GetPixels32(), cameraImageTexture.width,
cameraImageTexture.height);
        if (result != null)
        {
            SetQrCodeRecenterTarget(result.Text);
        }
    }
    private void SetQrCodeRecenterTarget(string targetText)
    {
        Target currentTarget = navigationTargetObjects.Find(x =>
x.Name.ToLower().Equals(targetText.ToLower()));
        if (currentTarget != null)
        {
            session.Reset();
            sessionOrigin.transform.position = currentTarget.PositionObject.transform.position;
            sessionOrigin.transform.rotation = currentTarget.PositionObject.transform.rotation;
        }
    }
    public void ChangeActiveFloor(string floorEntrance)
    {
        SetQrCodeRecenterTarget(floorEntrance);
    }
}
```

5.7.3 Navigation Controller

```
using UnityEngine;
using UnityEngine.AI;

public class NavigationController : MonoBehaviour {
   public Vector3 TargetPosition { get; set; } = Vector3.zero;
   public NavMeshPath CalculatedPath { get; private set; }
   private void Start() {
      CalculatedPath = new NavMeshPath();
   }
   private void Update() {
      if (TargetPosition != Vector3.zero) {
         NavMesh.CalculatePath(transform.position, TargetPosition, NavMesh.AllAreas, CalculatedPath);
      }
   }
}
```

5.7.4 Get Image Alternative

```
using TMPro;
using UnityEngine;
using UnityEngine.XR.ARFoundation;
using ZXing;

public class GetImageAlternative : MonoBehaviour
{
   [SerializeField]
   private ARCameraBackground arCameraBackground;
   [SerializeField]
   private RenderTexture targetRenderTexture;
   [SerializeField]
   private TextMeshProUGUI qrCodeText;
```

```
    private Texture2D cameraImageTexture;
    private IBarcodeReader reader = new BarcodeReader();
    private void Update()
    {
        Graphics.Blit(null, targetRenderTexture, arCameraBackground.material);
        cameraImageTexture = new Texture2D(targetRenderTexture.width, targetRenderTexture.height,
TextureFormat.RGBA32, false);
        Graphics.CopyTexture(targetRenderTexture, cameraImageTexture);
        var result = reader.Decode(cameraImageTexture.GetPixels32(), cameraImageTexture.width,
cameraImageTexture.height);
        if (result != null)
        {
            qrCodeText.text = result.Text;
        }
    }
}
```

5.7.5 Enable Disable Map button

```
using System.Collections;
using System.Collections.Generic;
using UnityEngine;
using UnityEngine.UI;
public class EnableDisable0 : MonoBehaviour
{
    public GameObject Rawimage;
    public bool isEnabled = true;
    public void ButtonClicked0()
    {
        isEnabled = !isEnabled;
        Rawimage.SetActive(isEnabled);
    }
}
```

5.7.6 Enable Disable Floor button

```
using System.Collections;
using System.Collections.Generic;
using UnityEngine;
using UnityEngine.UI;

public class EnableDisable1 : MonoBehaviour
{
    public GameObject FloorOptionPanel;
    public bool isEnabled = true;
    public void ButtonClicked()
    {
        isEnabled = !isEnabled;
        FloorOptionPanel.SetActive(isEnabled);
    }
}
```

5.7.7 Target script

```
using System;
using UnityEngine;
[Serializable]
public class Target
{
        public string Name;
        public GameObject PositionObject;
    public int FloorNumber;
    public Vector3 Position;
    public Vector3 Rotation;
}
```

5.7.8 SetUiText script

```
using TMPro;

using UnityEngine;

public class SetUiText : MonoBehaviour

{

    [SerializeField]

    private TMP_Text textField;

    [SerializeField]

    private string fixedText;

    public void OnSliderValueChanged(float numericValue)

    {

        textField.text = $"{fixedText}: {numericValue}";

    }

}
```

5.8 Conclusion

We examined the several facets in the implementation of an Augmented Reality (AR) Navigation System for indoor spaces. Our discussion commenced with an examination of the architectural design, highlighting the significance of a sturdy and adaptable system that smoothly interacts with the current infrastructure.

Overall, the detailed examination of these sections underscores the complexity and potential of AR-based indoor navigation systems. By addressing each aspect methodically, we can develop a solution that enhances user experience and operational efficiency within indoor environments.

The development of the AR navigation app demonstrated its potential to provide accurate and user-friendly navigation. And the implementation was successful.

CHAPTER VI
SYSTEM TESTING

6.1 Introduction

An important stage in the software development life, system testing ensures that the whole and integrated system satisfies the designated criteria by means of thorough testing of the whole system. This phase seeks to confirm under several circumstances the dependability, performance, and usefulness of the system. Comprehensive system testing helps us identify and address any problems so that the software runs as intended, so guaranteeing a flawless user experience.

Our goal in doing extensive system testing was to make sure the AR navigation software not only satisfied the given criteria but also gave a dependable and fun user interface. The knowledge acquired at this stage was priceless in pointing up areas needing development and maximizing the accessibility and efficiency of the software.

We will discuss the specifics of the system testing procedure in the next sections, including each of the tests carried out, the results obtained from them. This study will clearly show whether or not the app is for usage and how well it could satisfy consumer needs.

6.2 System Testing

System testing for the AR navigation app involved several tests including: Version 1.0 test and Version 2.0 test. After testing each version, we worked hard to develop the application’s performance and usability as well as the user interface.

Furthermore, we will witness significant improvements in the development of the user interface, ultimately resulting in its definitive state in version 3.5.

6.2.1 Version 1.0

In this initial version we made sure that the destinations were accessible smoothly, by testing all the destinations. Then we tested the readability of the QR codes unfortunately QR scanner didn’t work as expected.

When the user starts the program, he can choose the floor he is on by clicking the “Show Floor Options” button, then he chooses the floor. After that, he can choose the destination he wants on this floor by clicking the “Show Class Option” button, and then the program starts showing the navigation line for the selected room, as shown in Figure 6-1 to 6-3.

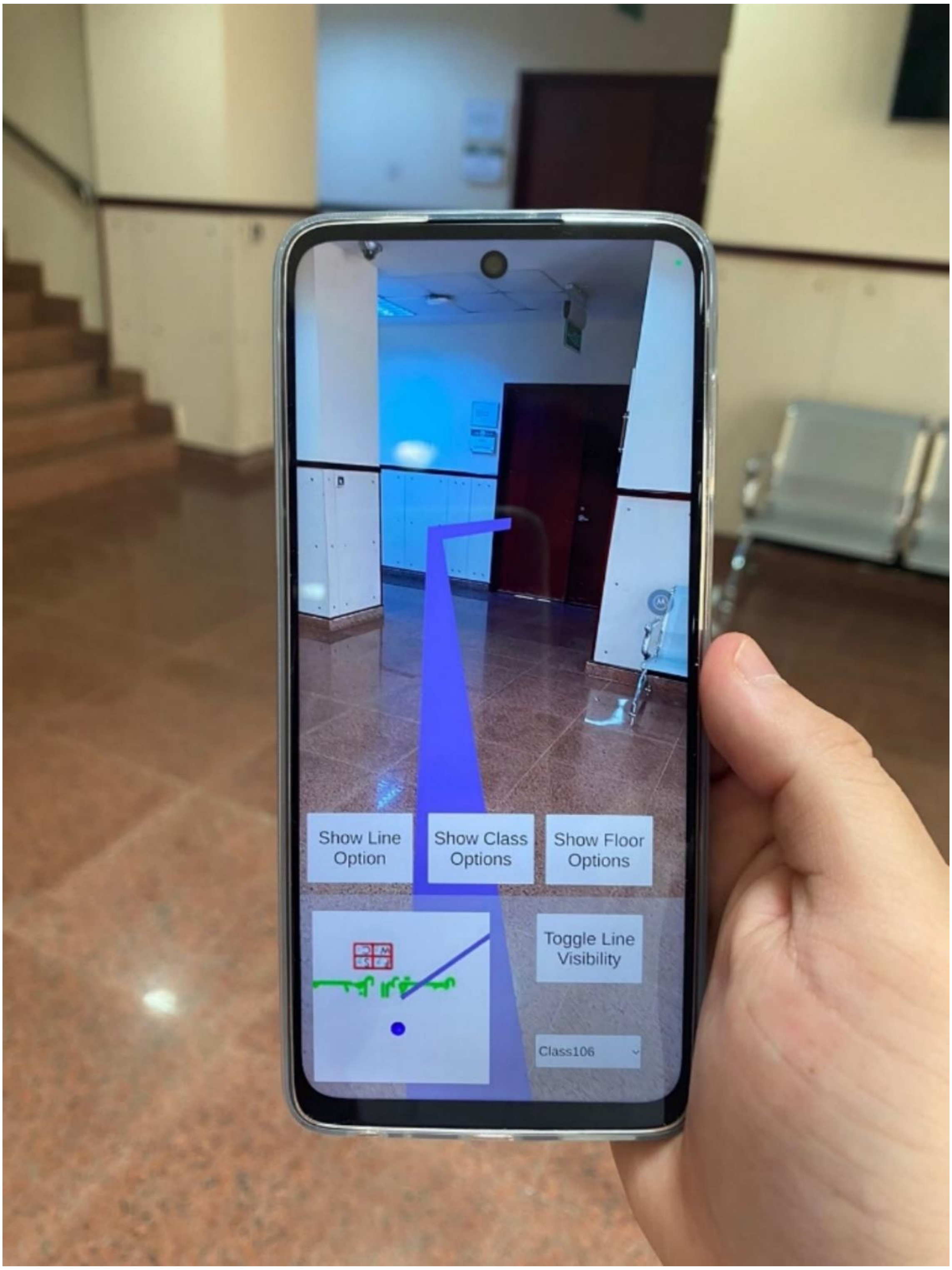

Figure 6-1: Version 1.0 first test.

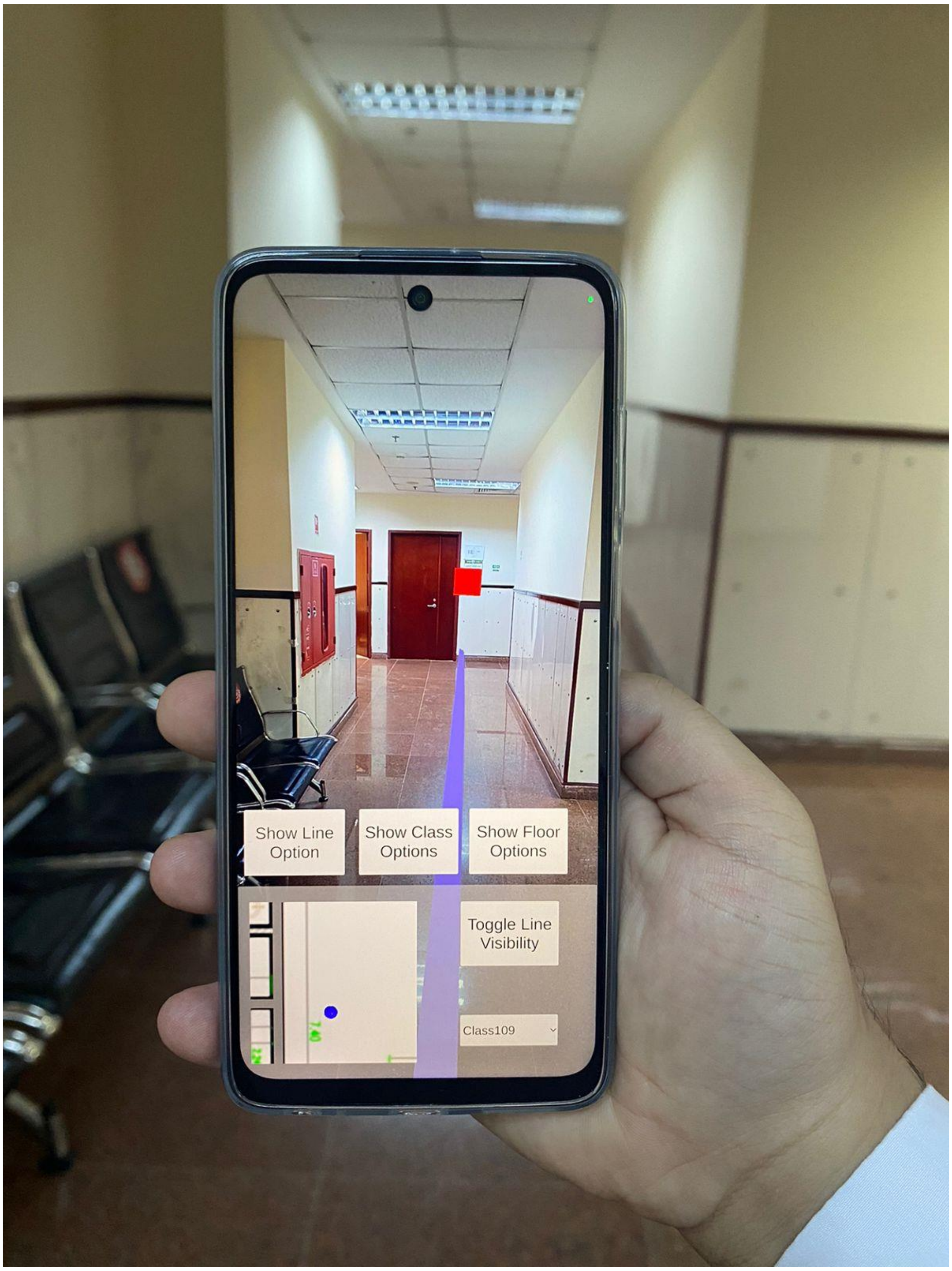

Figure 6-2: Version 1.0 navigate to 109.

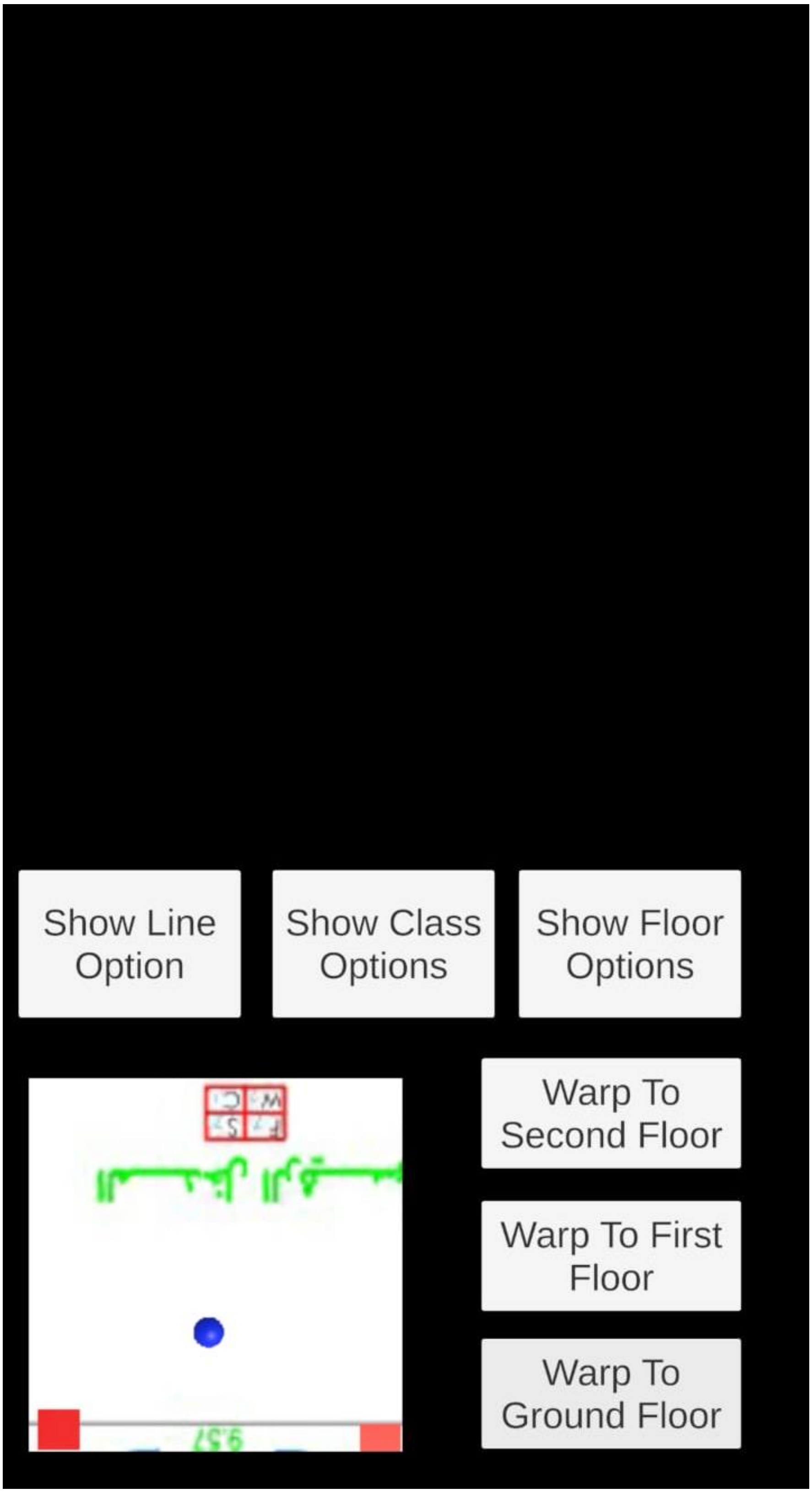

Figure 6-3: Version 1.0 user interface.

6.2.2 Version 2.0

In this version, we tested the QR code for all destinations on all floors of the building to verify the correct positioning of the user after reading the QR code, taking into account the correct angle for each QR code. This is important because the barcode must be on the wall's side, and the application must accurately read the user's movements after it reads the QR code, illustrated in Figure 6-4 to 6-6.

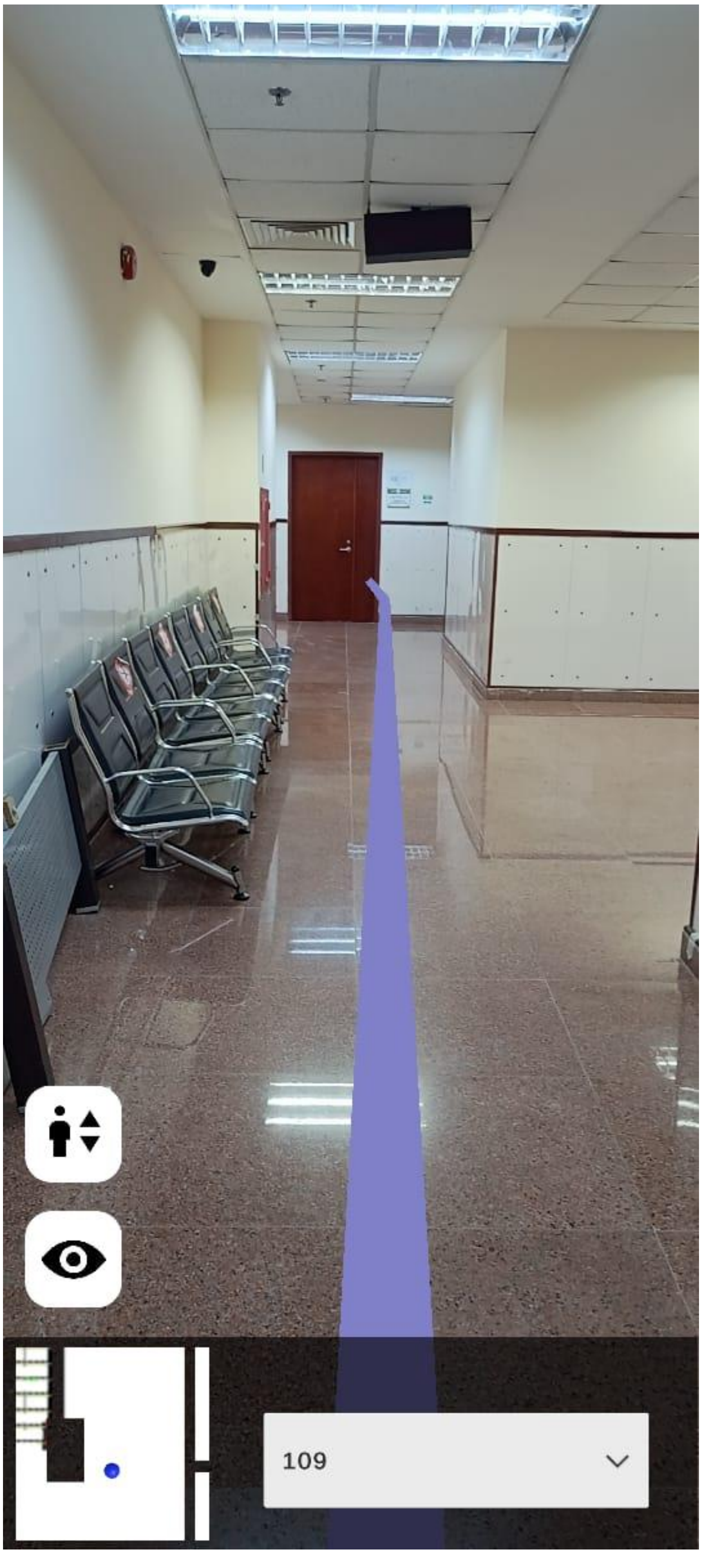

Figure 6-4: Version 2.0 navigate to 109.

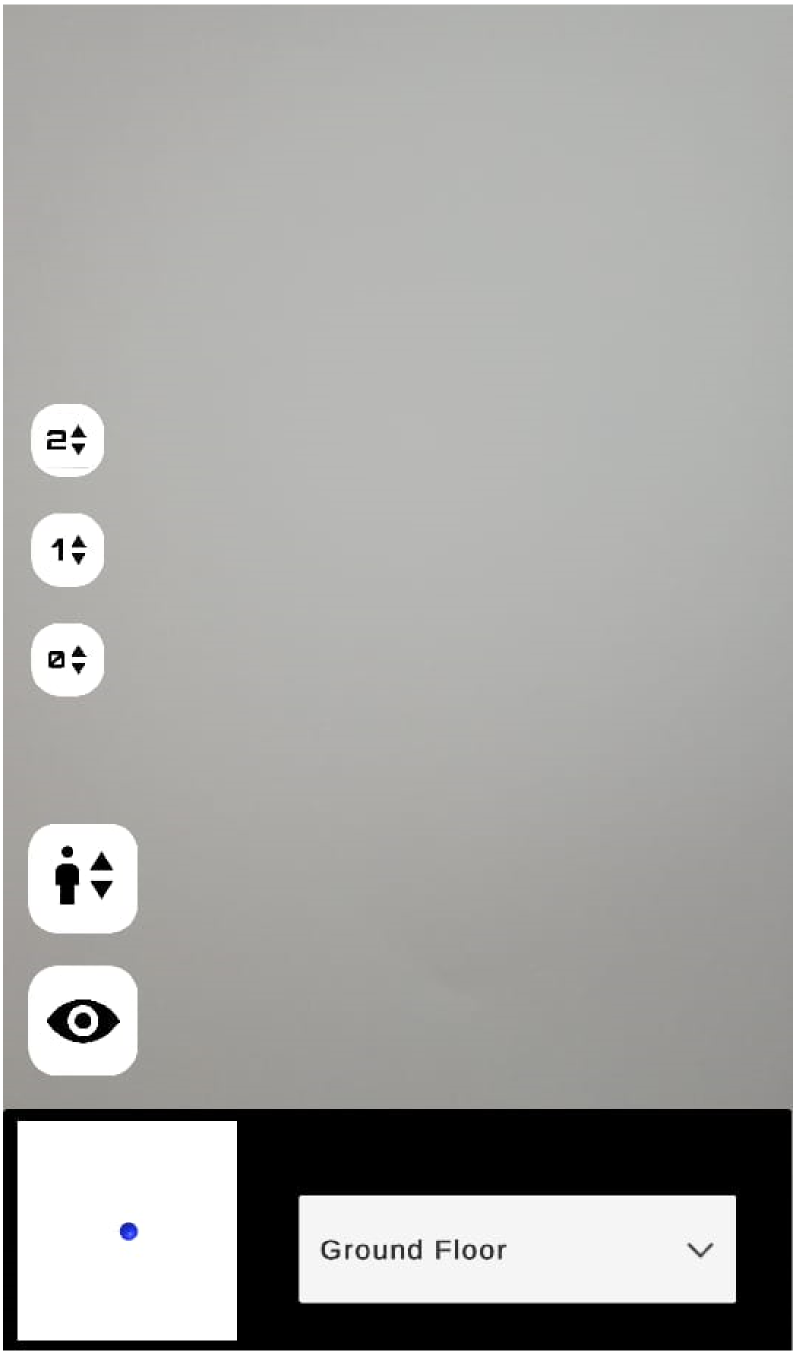

Figure 6-5: Version 2.0 interface after clicking The Floor button.

a. The 0-icon button: allow the user to access the ground-floor map navigation targets.
b. The 1-icon button: allow the user to access the first-floor map navigation targets.
c. The 2-icon button: allow the user to access the second-floor map navigation targets.

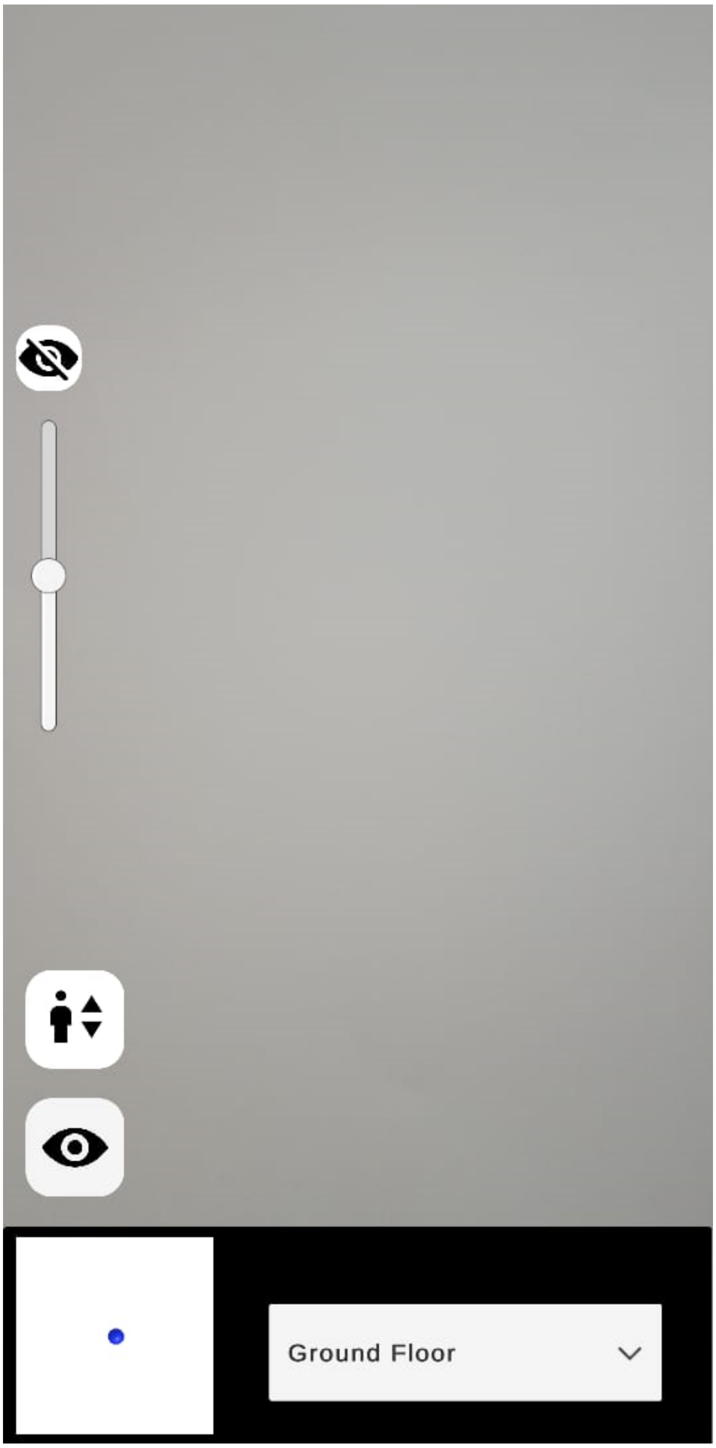

Figure 6-6: Version 2.0 interface after clicking the line visibility button.

a. Hidden eye button: allow the user to make the line only visible in the mini map.
b. The bar-line: control the height of the navigation line.

6.2.3 Version 3.0

The progress of the user interface can be observed starting with version 2.0. The interface has been streamlined to enhance user experience by incorporating a button that hides the mini-map and seamlessly integrating the buttons with the upper and bottom panels in Figure 6-7 and 6-8.

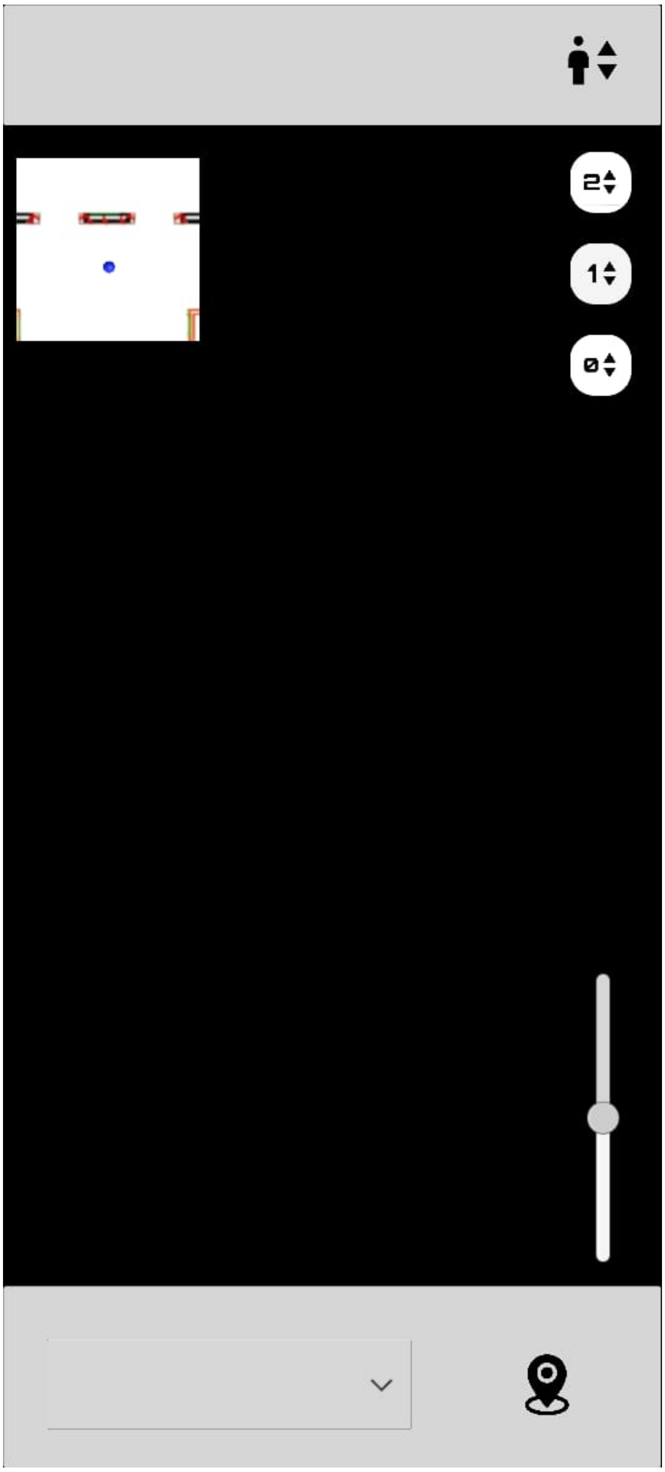

Figure 6-7: Version 3.0 user interface improvement.

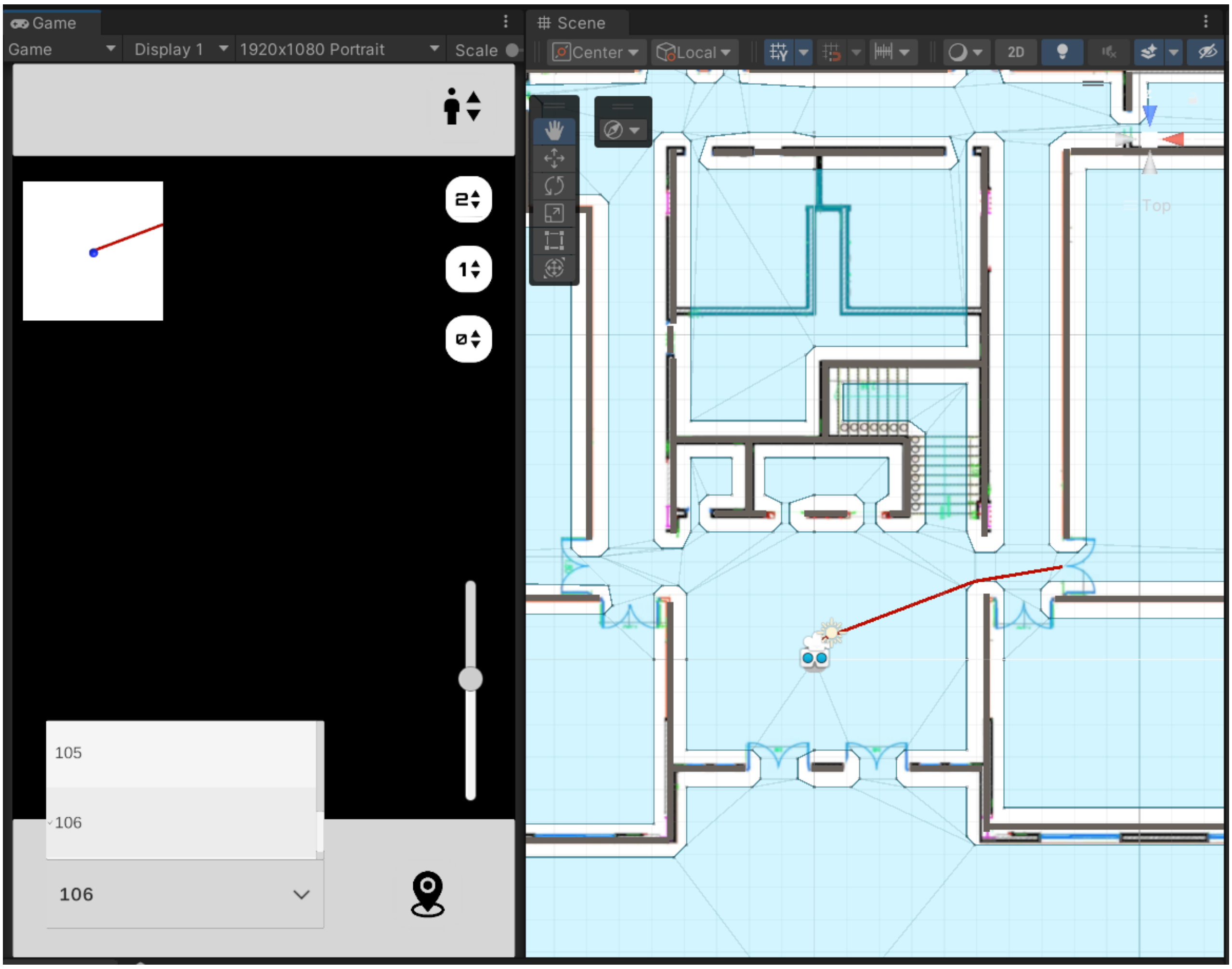

Figure 6-8: Version 3.0 navigate to 106.

6.3 Conclusion

Functional testing indicates that all features, including accurate navigation, efficient QR code scanning, and a straightforward UI, are functioning as intended. Through the implementation of performance testing, the app was ensured to effectively handle real-world usage scenarios, hence maintaining stability and responsiveness even when subjected to heavy usage. These tests will provide a satisfactory user experience and instill confidence that the application is prepared for utilization.

We have ensured the software's robustness, reliability, and user satisfaction by meticulously evaluating both its functional and performance aspects. Future advancements can leverage these solid foundations, hence enhancing the capability and user satisfaction of the software.

Here the first-floor plan before and after we added the necessary modifications, such as some walls and simplified the plans, in the figure 6-9 and 6-10.

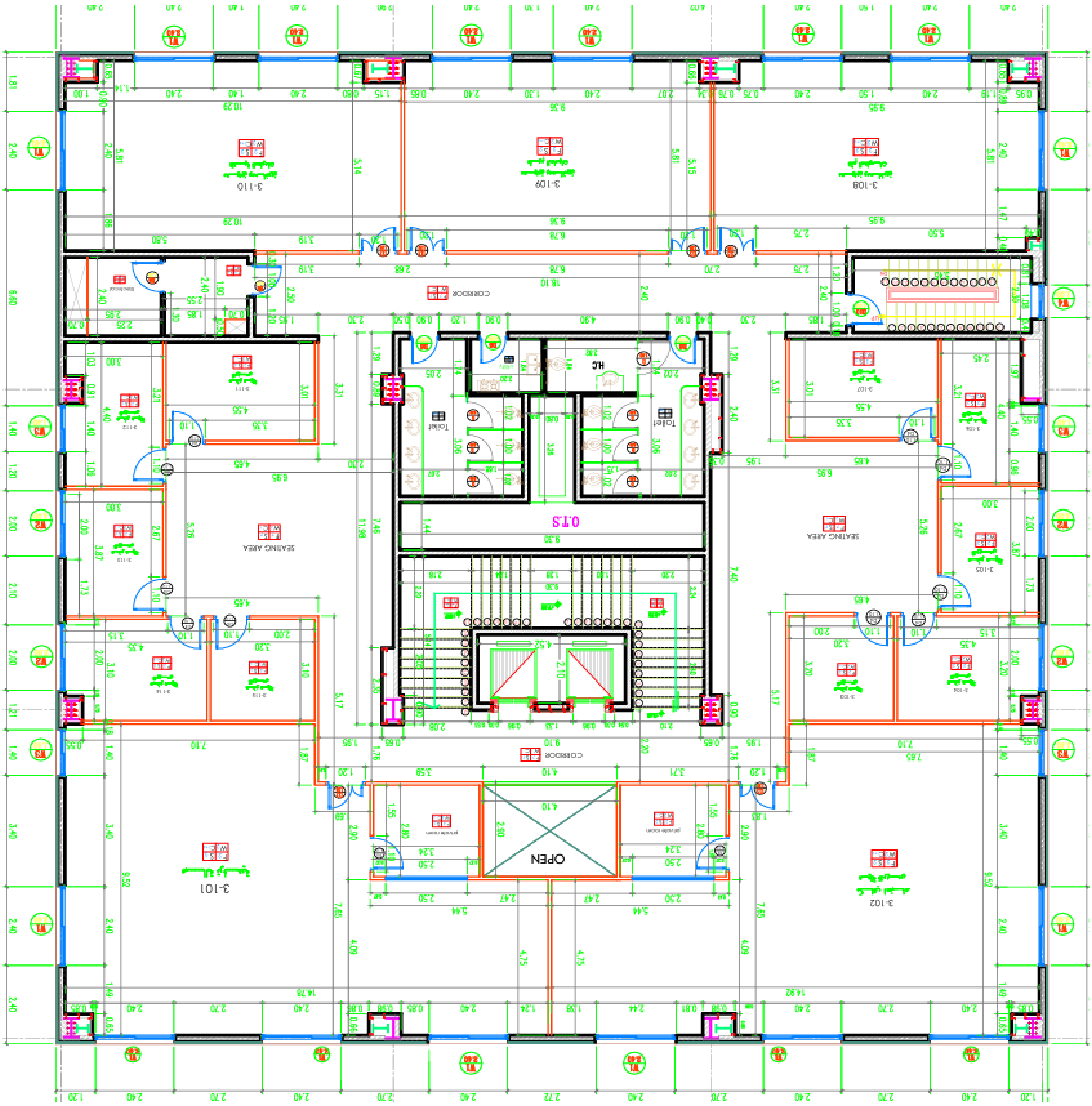

Figure 6-9: The building plan before modification.

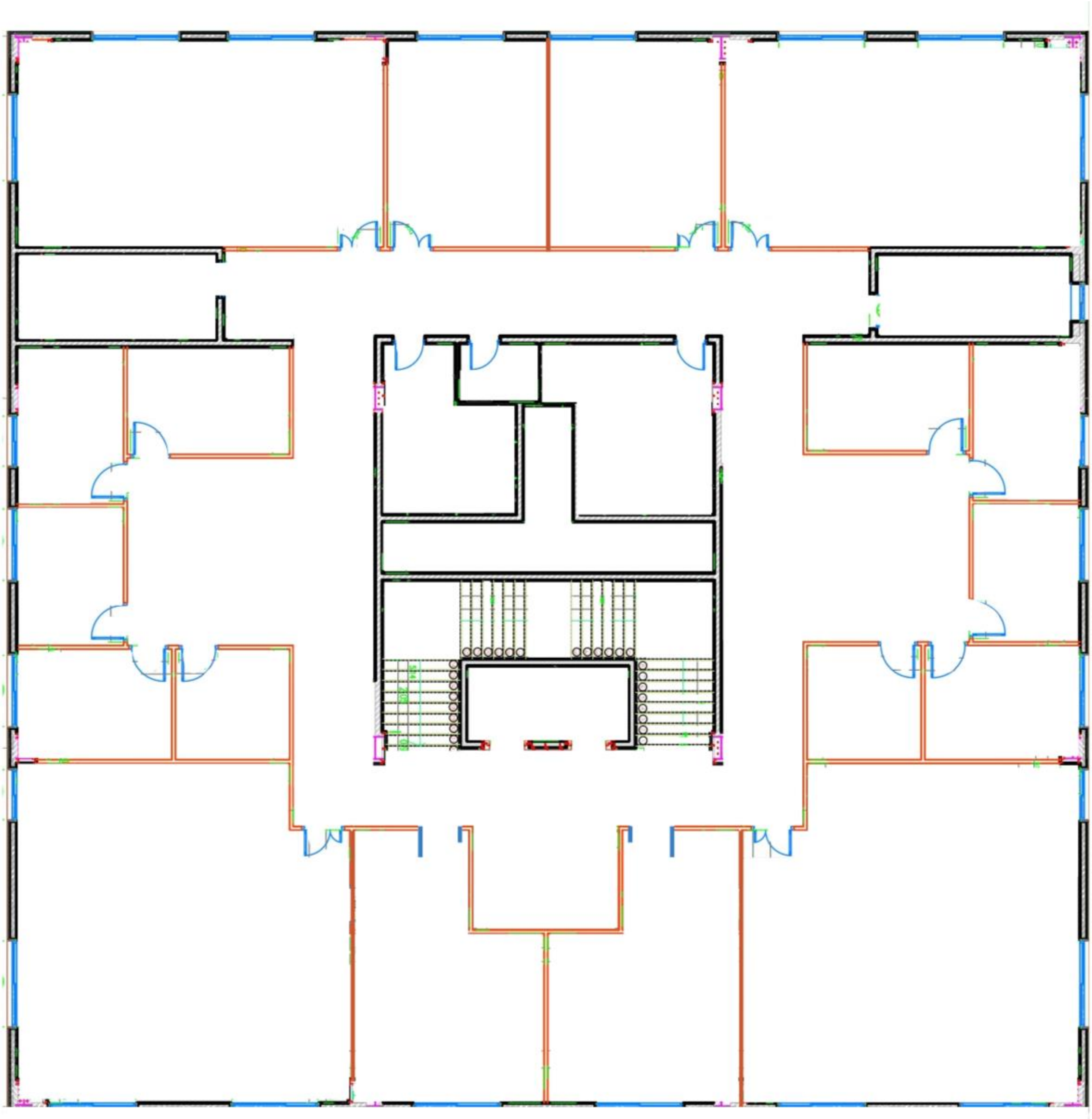

Figure 6-10: The building plan after modification.

CHAPTER VII
CONCLUSIONS

7.1 Introduction

This chapter marks the end of our trip in creating the AR navigation software; it offers a whole picture of the results of the project, points up areas needing work, and suggests future developments. This chapter's main objectives are to consider the achievements, assess the success of the created system, and sketch a road map for next development.

Throughout the project, we aimed to create an original AR navigation system that would improve interior navigation experiences. Using augmented reality, we wanted to provide consumers with precise and simple navigation directions within complex indoor areas. The many phases of development requirement analysis, design, implementation, and comprehensive testing all contribute to the project's overall success.

7.2 Achievements

The difficulty of having to look for the exam room after arriving late was the motivation for us to develop an application that all visitors to the building, including first-time visitors, students, and staff, can use to easily reach their goal. Using all of the knowledge and expertise that we had accumulated throughout our studies, we wanted to create a system that was straightforward and easy to use.

For the purpose of developing this application and achieving the goals of the project, we were required to use Unity to generate a three-dimensional design for each floor of the building. In addition to this, we were required to thoroughly plan the navigation area by making use of the offered plan as well as manual measurements in order to guarantee precisely accurate adjustments. The design of the user interface was a time-consuming procedure. In order to arrive at its ultimate form, the application went through a number of different adjustments. In order to do this, the design team and the development team had to hold a number of meetings, during which they conducted extensive searches for ideas that could be used to inspire comparable work in the design.

The main tool of the program, AR navigation, underwent considerable testing and came out to be quite accurate. With an 88% to 90% accuracy rate, users could follow virtual red line layered on the actual surroundings to reach their targets. This great degree of accuracy guaranteed users could depend on the app for exact faculty navigation, illustrated in table 7-1 and the formula next page.

Another very important component of the app was QR code scanning. The program showed a 100% success rate in scanning and deciphering QR codes put at different sites during testing. By just scanning a QR code, this function let users rapidly relocate user position inside the application, so improving the whole user experience.

The user interface was intended to be simple and understandable. Users of the interface, according to tests, found it simple to operate with responsive buttons and unambiguous visual signals. Users using the simple navigation controls and minimalistic design reported great degrees of pleasure, which helped to create a good user experience.

The Destination	The Distance From Target
101	35 cm
102	15 cm
103	12 cm
104	0 cm
105	0 cm
106	23 cm
The Library	0 cm
The Prayer Room	10 cm
Bioinformatics Lab	8 cm
Computer Skills Unit	0 cm
109	0 cm
111	15 cm
Oracle Database Lab	0 cm
Robots Club	0 cm
The Vice Dean	10 cm
IT Department	18 cm
Printing Management	10 cm
The Dean	0 cm
Dean's Office Manager	0 cm
Administration Manager	7 cm
IS Department	10 cm
CS Department	12 cm

Table 7-1: shows the distance from targets

To calculate the accuracy first we need to know the average distance from the targets here how we did it:

The sum of distances: (35+15+12+23+10+8+15+10+18+10+7+10+12) = 185

And the number of rooms is 22 and the average distance: 185/22 = 8.41 cm

Using 70 cm as the maximum acceptable distance:

Accuracy (%) = (1 – Average Distance from Target / Maximum Acceptable Distance) * 100

Accuracy (%) = (1 – 8.41/70) * 100

Accuracy (%) = (1 – 0.12) * 100

Accuracy (%) = 0.88 * 100

Accuracy (%) = 88%

7.3 Weakness of Developed System

Although the AR navigation software showed good user satisfaction and performance, some flaws were found throughout development and testing stages. Improving the general functioning and user experience of the app depends on addressing these flaws. The main areas in which the app may use improvement are these:

Ensuring precise GPS positioning—especially in interior environments—was one of the main difficulties experienced during the creation of the AR navigation software. Although the app's navigation accuracy was generally good, it suffered in situations with weak GPS signal, such subterranean locations or thick-walled structures. In several cases, this restriction affected the dependability of the navigation direction.

It is also relevant to note that the three floors are isolated from one another, and that each floor has its own unique destinations. It is not possible to go from one floor to another, such as from the ground floor to the first floor, or vice versa. Due to a lack of time, we were not able to merge all of the floors into a single building. This means that the user cannot select, for instance, 109, which is located on the first floor, and the user are on the ground floor, meaning they will not be able to navigate from different floors. We built a button designed to travel to the floor on which the user is currently located so that he could access the destinations for that floor. This was done so that we could link the floors together in a single application.

7.4 Future Enhancement

Here several upcoming improvements that provide an even better user experience help to guarantee the ongoing success and improvement of the AR navigation app:

Advanced locating techniques such Bluetooth beacons, Wi-Fi triangulation, or integration of indoor mapping technologies assist to increase navigation accuracy in interior environments.

Voice-guided navigation can also give users auditory directions, therefore enabling them to navigate without always staring at their screens. Those who prefer hands-free navigation or visually challenged users may especially find this function helpful.

Apart from an in-app feedback system, which facilitates users' reporting of problems and comments. This can include choices for app rating, idea submission, bug reporting.

Here is a look at the application icon design, as well as a sample of the QR codes that will be placed on the doors of the FCITR building rooms, as shown in figure 7-1 and 7-2.

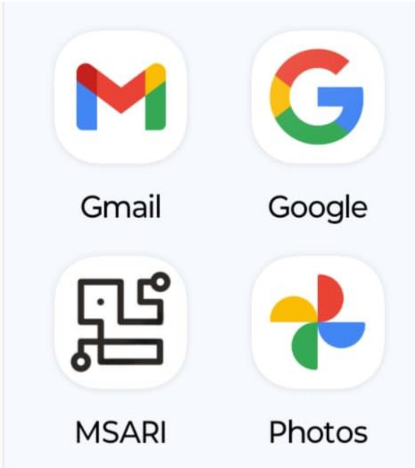

Figure 7-1: (MSARI) application icon.

Figure 7-2: IT Department QR code.

7.5 Reference

1. Rubio-Sandoval, J.I.; Martinez-Rodriguez, J.L.; Lopez-Arevalo, I.; Rios-Alvarado, A.B.; Rodriguez-Rodriguez, A.J.; Vargas-Requena, D.T. An Indoor Navigation Methodology for Mobile Devices by Integrating Augmented Reality and Semantic Web. Sensors 2021, 21, 5435. https://doi.org/ 10.3390/s21165435
2. https://www.g2.com/articles/history-of-augmented-reality
3. Jacob, R.; Zheng, J.; Ciepłuch, B.; Mooney, P.; Winstanley, A.C. Campus guidance system for international conferences based
4. on OpenStreetMap. In Proceedings of the International Symposium on Web and Wireless Geographical Information Systems, Maynooth, Ireland, 7–8 December 2009.
5. Wenrui, S.; Liansun, Z.; Zhen, Y. School based on Android platform Garden guide software design. *Electron. Des. Eng.* **2012**, *20*, 26–28.
6. Anpat, V.; Shewale, A.; Bhangale, Y. Campus Navigation on Android Platform. *Int. J. Sci. Technol. Eng.* **2016**, *2*, 452–458.
7. Yu, K.-M.; Chiu, J.-C.; Lee, M.-G.; Chi, S.-S. A mobile application for an ecological campus navigation system using augmented reality. In Proceedings of the International Conference on Ubi-Media Computing (UMEDIA), Colombo, Sri Lanka, 24–26 August 2015; pp. 17–22.
8. Nextech3D.ai The Unprecedented Benefits of Indoor Navigation Using Augmented Reality
9. Augmented Reality BC Campus Tour App Evgeniya Kokareva, Ana Satkeeva, Robin Shaw, Bridget Wellen Supervisor: Dr. Fatma Cemile Serce CS / Bellevue College
10. Application of Augmented Reality in Campus Navigation Li Zhigang*, Qi Guanglei, Hu Wenkai, Ma Xiangyu, Guo Qinsheng
11. Software Requirements for Mobile Augmented Reality Tourism Application Rashidi Abd Rashid1. Halina Mohamed Dahlan
12. AR-UTAR Kampar Campus Navigation,HEW TENG WEI
13. Campus Navigation Using Augmented Reality ,Mayuri Tamhane, Prathamesh Sarjekar,Chirag Gupta, Prof. V. M. Kharche
14. Augmented Reality Navigation, Munesh Kumar Sharma, Satya Chachaundiya, Vishal
15. Indoor Navigation Using Augmented Reality: Investigation of Microsoft HoloLens for Indoor Navigation in a University Building, Kyriaki Kalantari, Samuel B. Scherer, Thomas Liebig, Nikolaos Gkatzoflias, Ann-Kristin Smailies, Daniel Vögele, Eleni A. Kourti, Elena E. Kordella, Oliver Wendt, Tim A. Majchrzak, Petra Knoth, Jörg P. Müller
16. Saeed, S., Jhanjhi, N. Z., Naqvi, M., & Humayun, M. (2019). Analysis of software development methodologies. International Journal of Computing and Digital Systems, 8(5), 446-460.

17. Built in, "What Is the Waterfall Methodology?", https://builtin.com/software-engineering-perspectives/waterfall-methodology , [Last accessed April 25, 2024]
18. Lvivity, "Waterfall Methodology: Advantages, Disadvantages And When to Use It?" , Available at: https://lvivity.com/waterfall-model , [Last accessed April 25, 2024]
19. Adenowo, A. A., & Adenowo, B. A. (2013). Software engineering methodologies: a review of the waterfall model and object-oriented approach. International Journal of Scientific & Engineering Research, 4(7), 427-434.
20. Sommerville, I. (2016). Software Engineering (10th ed.). Pearson.
21. Pressman, R. S. (2014). Software Engineering: A Practitioner's Approach (8th ed.). McGraw-Hill Education.
22. Augmented Reality for Developers (Johnathan Linowes, Krystian Babilinski 2017)
23. Brito, C. de la N. A. (2015). Augmented reality applied in tourism mobile applications.In 2nd International Conference on eDemocracy and eGovernment (ICEDEG) .IEEE.
24. Rehman, U., & Cao, S. (2016). Augmented-reality-based indoor navigation: A comparative analysis of handheld devices versus google glass. IEEE Transactions on Human-Machine Systems, 47(1), 140-151.
25. AR Security: The Challenges and Solutions https://www.wikitude.com/blog-ar-security-the-challenges-and-solutions/
26. Mobile Application Security by Himanshu Dwivedi, Chris Clark, and David Thiel

^_^