A

Project Report

on

INDOOR NAVIGATION USING AUGMENTED REALITY

Submitted in partial fulfillment for the requirements for the award of the degree of

BACHELOR OF ENGINEERING

in

INFORMATION TECHNOLOGY

By

Y. Manoj Kumar Reddy (2451-19-737-132)

P. Likhith (2451-19-737-158)

K. Revan (2451-19-737-178)

Under the guidance of

S. Ch. Vijaya Bhaskar

Assistant Professor Department of IT

MATURI VENKATA SUBBA RAO ENGINEERING COLLEGE

(An Autonomous Institution)

Department of Information Technology,

(Affiliated to Osmania University, Hyderabad)

Nadergul, Hyderabad, TELANGANA – 501510

Academic Year: 2022-23

MATURI VENKATA SUBBA RAO(MVSR) ENGINEERING COLLEGE

(An Autonomous Institution)

(Affiliated to Osmania University, Hyderabad, Recognized by AICTE)

Nadergul, Saroornagar Mandal, Hyderabad-501510

DEPARTMENT OF INFORMATION TECHNOLOGY

CERTIFICATE

*This is to certify that the project work entitled "***Indoor Navigation using Augmented Reality***" is a bonafide work carried out by* ***Mr. Y. Manoj Kumar Reddy (2451-19-737-132), Mr. P. Likhith (2451-19-737-158), Mr. K. Revan (2451-19-737-178)*** *in fulfillment of the requirements for the award of degree of* **Bachelor of Engineering** *in* **Information Technology** *from* **Maturi Venkata Subba Rao Engineering College,** *affiliated to OSMANIA UNIVERSITY, Hyderabad, during the Academic Year 2022-23. under our guidance and supervision.*

The results embodied in this report have not been submitted to any other university or institute for the award of any degree or diploma.

Signature of Project Coordinator

Signature of Guide

Signature of Head, ITD

Signature of External Examiner

DECLARATION

This is to certify that the work reported in the present project entitled "Indoor Navigation using Augmented Reality" is a record of bonafide work done by us in the Department of Information Technology, Maturi Venkata Subba Rao Engineering College, Osmania University. The reports are based on the project work done entirely by us and not copied from any other source.

The results embodied in this project report have not been submitted to any other University or Institute for the award of any degree or diploma to the best of our knowledge and belief.

Roll Number:	Student Name:	Signature of the Student
(2451-19-737-132)	Y. Manoj Kumar Reddy	
(2451-19-737-158)	P. Likhith	
(2451-19-737-178)	K. Revan	

ACKNOWLEDGMENT

We would like to express our sincere gratitude and indebtedness to our project guide **Mr. S. Ch. Vijaya Bhaskar, Assistant Professor** for her valuable suggestions and interest throughout the course of this project.

We would like to extend our gratitude to Project Coordinator **Dr. Chepuri Samson, Professor,** Department of Information Technology, Maturi Venkata Subba Rao Engineering College, for his valuable suggestions and timely help during the course of the project.

Finally, we would like to take this opportunity to thank our family for their support through the work. We sincerely acknowledge and thank all those who gave directly or indirectly their support in completion of this work.

Y. Manoj Kumar Reddy (2451-19-737-132)
P. Likhith (2451-19-737-158)
K. Revan (2451-19-737-178)

Maturi Venkata Subba Rao Engineering College

Department of Information Technology

COURSE NAME: Project Work II

COURSE CODE: PW861IT

VISION

To impart technical education to produce competent and socially responsible engineers in the field of Information Technology.

MISSION

a. To make teaching learning process effective and stimulating.
b. To provide adequate fundamental knowledge of sciences and Information Technology with positive attitude.
c. To create an environment that enhances skills and technologies required for industry.
d. To encourage creativity and innovation for solving real world problems.
e. To cultivate professional ethics in students and inculcate a sense of responsibility towards society.

PROGRAM EDUCATIONAL OBJECTIVES(PEOS)

The Program Educational Objectives of undergraduate program in Information Technology are to prepare graduates who will:

I. Apply knowledge of mathematics and Information Technology to analyze, design and implement solutions for real world problems in core or in multidisciplinary areas.

II. Communicate effectively, work in a team, practice professional ethics and apply knowledge of computing technologies for societal development.

III. Engage in Professional development or postgraduate education to be a life-long learner.

(A) PROGRAM OUTCOMES(POs)

At the end of the program the students (Engineering Graduates) will be able to:

1. **Engineering knowledge:** Apply the knowledge of mathematics, science, engineering fundamentals, and an engineering specialization to the solution of complex engineering problems.
2. **Problem analysis**: Identify, formulate, review research literature, and analyse complex engineering problems reaching substantiated conclusions using first principles of mathematics, natural sciences, and engineering sciences.
3. **Design/development of solutions:** Design solutions for complex engineering problems and design system components or processes that meet the specified needs with appropriate consideration for the public health and safety, and the cultural, societal, and environmental considerations.
4. **Conduct investigations of complex problems:** Use research-based knowledge and research methods including design of experiments, analysis and interpretation of data, and synthesis of the information to provide valid conclusions.
5. **Modern tool usage:** Create, select, and apply appropriate techniques, resources, and modern engineering and IT tools including prediction and modeling to complex engineering activities with an understanding of the limitations.
6. **The engineer and society:** Apply reasoning informed by the contextual knowledge to assess societal, health, safety, legal and cultural issues and the consequent responsibilities relevant to the professional engineering practice.

7. **Environment and sustainability:** Understand the impact of the professional engineering solutions in societal and environmental contexts, and demonstrate the knowledge of, and need for sustainable development.
8. **Ethics:** Apply ethical principles and commit to professional ethics and responsibilities and norms of the engineering practice.
9. **Individual and team work:** Function effectively as an individual, and as a member or leader in diverse teams, and in multidisciplinary settings.
10. **Communication:** Communicate effectively on complex engineering activities with the engineering community and with society at large, such as, being able to comprehend and write effective reports and design documentation, make effective presentations, and give and receive clear instructions.
11. **Project management and finance:** Demonstrate knowledge and understanding of the engineering and management principle and apply 6 these to one’s own work, as a member and leader in a team, to manage projects and in multidisciplinary environments.
12. **Lifelong learning:** Recognize the need for, and have the preparation and ability to engage in independent and life-long learning in the broadest context of technological change.

(B) PROGRAM SPECIFIC OUTCOMES (PSOs)

1. **Hardware design:** An ability to analyse, design, simulate and implement computer hardware / software and use basic analogue/digital circuits, VLSI design for various computing and communication system applications.
2. **Software design**: An ability to analyse a problem, design algorithm, identify and define the computing requirements appropriate to its solution and implement the same.

COURSE OBJECTIVES AND OUTCOMES

Course Objectives

1. To enhance practical & Professional skills.
2. To familiarize the tools and techniques of symmetric literature survey and documentation.
3. To expose students to industry practices and teamwork.
4. To encourage students to work with innovative and entrepreneurial ideas

Course Outcomes

On successful completion of this course student will be

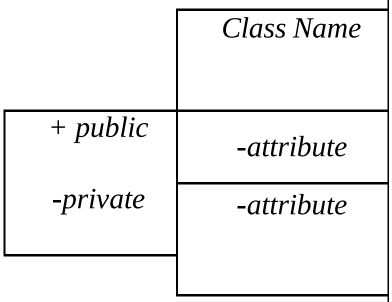
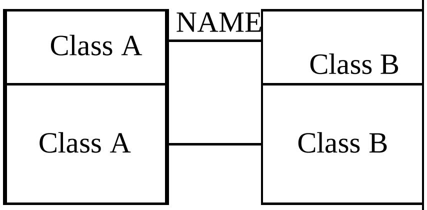
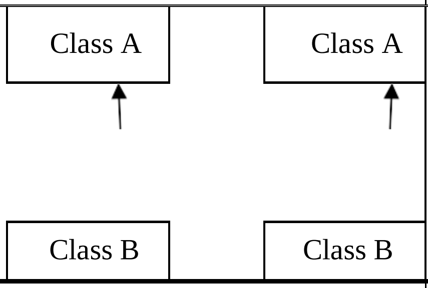
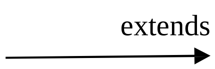
1. Define a problem of the recent advancements with applications towards society.
2. Outline requirements and perform requirement analysis for solving the problem.
3. Design and develop a software and/or hardware, based solutions within the scope of project using contemporary technologies and tools.
4. Test and deploy the applications for use.
5. Develop the Project as a team and demonstrate the application, with effective written and oral communications

ABSTRACT

Navigation Systems are very much essential for en-route assistance, indoor positioning etc. For outdoor environment navigation is still better as compared to complex indoor environment. In this project, we have focused on building an indoor navigation application which uses augmented reality to assist people in navigating at complex buildings and also making a cloud platform (Content Management System) from where the administrator of a particular building can be able to modify and manage the navigation path. We have used Unity 3d framework to develop the AR based mobile application. The application can be run on smartphones. It has been seen that this augmented reality-based application provides better interface and experience than the traditional 2D maps or the paper maps that are displayed outside buildings to help in the navigation. To evaluate the concept proposed in the project, technical evaluations were performed at the college.

LIST OF FIGURES

LIST OF SYMBOLS

S.NO	NOTATION NAME	NOTATION	DESCRIPTION
1.	Class		Represents a collection of similar entities grouped together.
2.	Association		Association represents static relationships between classes. Role represents the way the two classes see each other.
3.	Actor		It aggregates several classes into a single class.
4.	Aggregation		Interaction between the system and external environment
6.	Relation (extends)		Extends relationship is used when one use case is similar to another use case but does a bit more.
7.	Communication		Communication between various use cases.

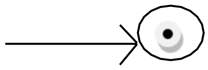
8.	State		State of the processes.
9.	Initial State		Initial state of the object
10.	Final state		Final state of the object
11.	Control flow		Represents various control flow between the states

TABLE OF CONTENTS

CHAPTER 1

INTRODUCTION

Significant developments in locational technologies and in the power and size of mobile devices have brought advanced portable location-aware systems into our everyday life. Navigation systems are one of the more common location-aware services, and many people use them on a daily basis. Providing location information through a user-friendly interface is crucial in a navigation system, as is the accurate determination of user locations and target locations. Most past and current indoor navigation systems are electronic sensor-based, relying on infrared, ultrasound, radio frequency, and so on, all of which necessitate an extra installation cost. Augmented Reality (AR) is particularly well-suited as a user interface for location-aware. AR is a variation of the virtual. The recent appearance of smartphones featuring GPS receivers, digital compasses, and accelerometers has made it more feasible to build practical AR applications for mobile devices. One of the benefits of using AR in location-aware applications is that it can provide an intuitive and user-friendly interface. This is possible because an AR interface dynamically superimposes interactive computer graphics images onto objects in the real world. In this project we are using QR code to determine the user's current location and collaborating internal Point Of Interest (POI) data to provide indoor navigation information. We also present a scenario-based approach to develop a mobile AR indoor navigation system which could be used in defined areas such as campuses or shopping malls.

1.1 PROBLEM STATEMENT

When compared to the outdoor positioning, indoor positioning faces multiple problems and has specific requirement. Satellite signals such as GPS are more difficult to track in indoors because radio frequency signals are attenuated by barriers such as walls in a building. In comparison, indoor positioning is commonly associated with more specific locations than outdoor positioning.

1.2 OBJECTIVES

To create an android mobile application, which can be used by new visitors to navigate through the complex building structures.

1.3 MOTIVATION FOR AUGMENTED REALITY

AR allows to superimpose specific digital objects such as pictures, text, or sounds over real-world objects. Thus, it helps businesses create highly interactive and robust platforms. Even though AR has many amazing applications, one of its most effective applications is in navigation. Through this technology, you can easily identify and navigate through places such as shopping malls, museums, and parks. AR facilitates path logistics along with visual orientation for effective navigation. AR applications can highlight and superimpose directions on what your camera sees, along with instruction points for easy readability. You can also use AR for finding routes in case of an emergency. Augmented Reality is more effective in case of catastrophe since it uses optimization modes for efficient steering. This is better than standard navigation methods since it reduces cognitive overload. You can also use AR for measuring speed.

1.4 EXISTING SYSTEM –

Bluetooth Beacons

The Bluetooth Beacons are transmitters for hardware (Satan et.al. 2018). They are low energy Bluetooth systems that broadcast their identifier to neighbouring portable electronic devices. When they are close to the beacons, the hardware in these beacons enables laptops, smartphones, and other gadgets to perform acts. Bluetooth Beacons use Bluetooth low energy proximity sensing to relay a globally specific identifier.

RSSI-based Wi-Fi Fingerprinting

Using RSSI, the power present inside the received radio signal may be determined. Standard printing of fingers is also based on RSSI, although there is a minor distinction. In conventional fingerprinting, there are many access points and it was clearly focused on capturing the signal intensity from these access points, which are in range. This information is then stored in a database along with the known coordinates of the client computer in an offline process. This knowledge that is preserved can be probabilistic or deterministic.

1.5 PROPOSED SYSTEM

The development environment selected for the project is Unity because of the NavMesh components and its advantages. The proposed indoor navigation system is mainly divided into four basic modules, namely the AR Core based localization, the QR-code repositioning, the unity navmesh navigation, and finally the AR path showing. QR-codes are placed at different destinations inside the building. The user can scan the QR code at a specific location and choose the location he or she wants to visit. The path to get to the destination is shown as a line-render inside the mini-map, which is placed at the bottom.

1.6 SCOPE OF THE PROJECT

- The Application can be used for indoor navigation of complex building structures, where GPS and network connectivities are unreliable.
- This solution can be further improved and extended for complex multi-storied buildings.

1.7 OUTLINE OF THE RESULTS

A QR-Code is provided to the visitors, which must be downloaded and installed to navigate to the point of interests using a mobile application. The visitor can navigate to the point of interest by choosing in the drop-down menu of the application and navigate through the unknown building without using GPS or any network connectivity.

CHAPTER 2

LITERATURE SURVEY

S.NO	Yr. Of Pub	Author	Technique	Findings	Limitation
1	2013	Filonenko et al.	RF based Wifi Access Points	Surrounding WiFi can be positioned even if it is not connected.	Not functional without WiFi connectivity and must be in a networked state.
2	2015	Subedi and Pyun	RF based Bluetooth Beacons	No signal or connectivity required, just Bluetooth devices and has good compatibility with devices.	Need Application Management system or client-based solution.
3	2017	Kaushal et al.	Optical Wireless Positioning	Less infrastructure is required, as it utilizes the user's smartphone processing power and sensors.	Privacy issues with the users, as the information is stored in the server. Affected by dull light, bright light and motion blur etc. which causes error resulting in poor precision.
4	2019	Weinmann	Computer Vision	CV uses techniques that infer information about a scene by analysing images of that scene.	Long latency, poor accuracy and no height detail makes it difficult to scale markers and visual recognition.

Filonenko et al.2013 examined how well an indoor positioning system can work where publicly accessible WIFI access points are used without changing their standard communication protocols in which 60 GHz range millimetre waves were used in an office setting with a combination of human traffic. Related meter-level accuracies can be accomplished with commercial WIFI infrastructures, according to other studies. He reached an accuracy of 1.5 metres using fingerprinting, thus showing that both the offline and the online process can be carried out with a smartphone.

Subedi and Pyun, 2017 suggested a fingerprinting indoor positioning device based on BLE beacons. Additionally, the authors used a technique in the method called Weighted Centroid Localization (WCL) which is equivalent to multilateration. The authors were able to minimise the required number of deployed BLE beacons by eliminating noisy RSS values and using both fingerprinting and Weighted Centroid Localization approaches at the same time to achieve the same degree of precision compared to the case where only fingerprinting was used. This machine was able to obtain an accuracy of 1–2 metres in a hallway and an office room, where the BLE beacons were fitted to the ceiling, tests were carried out.

Weinmann, 2019 figured out that given a marker's known real-world scale and how perspective projects it onto a 2D image, the camera's relative direction and orientation to the marker can be easily determined using linear algebra. However, the output knowledge from this method is discrete in the sense that it defines which signpost the consumer is looking at. This technique cannot place the viewer if there are no markers in the camera's vision. As a consequence, the question is whether machine vision can offer continuous indoor navigation.

2.1 PROPOSED SYSTEM

Indoor Navigation using Augmented Reality

Significant developments in locational technologies and in the power and size of mobile devices have brought advanced portable location-aware systems into our everyday life. Navigation systems are one of the more common location-aware services, and many people use them on a daily basis. Providing location information through a user-friendly interface is crucial in a navigation system, as is the accurate determination of user locations and target locations. Global Positioning System (GPS), which offers maximum coverage, has been widely used to provide location information since Selective Availability (SA) was turned off in 2000 (Yanying Gu et al. 2009). However, it is well-known that GPS performs too poorly inside buildings to provide usable indoor positioning (Kjærgaard et al. 2010; Yanying Gu et al. 2009; Hui Liu et al. 2007; Ran et al. 2004). It is for that reason that indoor navigation systems are considered much more challenging than outdoor navigation systems (Mautz 2009). Commercial and research-based organisations have therefore carried out a large number of indoor positioning and navigation projects (some of which I will present in the paper) (Kray & Baus 2003; Hui Liu et al. 2007). Most past and current indoor navigation systems are electronic sensor-based, relying on infrared, ultrasound, radio frequency, and so on (Mautz 2009; Yanying Gu et al. 2009; Hui Liu et al. 2007), all of which necessitate an extra installation cost. I believe that not all cases should require high-resolution indoor positioning systems.

On the other hand, proving a user-friendly interface in a navigation system is vital to reducing navigation errors, as well as the time required for a user to understand that navigation information. Augmented Reality (AR) is particularly well-suited as a user interface for location-aware applications (Hollerer et al. 1999; Reitmayr and Schmalstieg 2004). AR is a variation of the virtual environment (Azuma 1997; Höllerer and Feiner 2004). The recent appearance of smartphones featuring GPS receivers, digital compasses, and accelerometers has made it more feasible to build practical AR applications for mobile devices (Jang and Hudson-Smith 2010). One of the benefits of using AR in location-aware applications is that it can provide an intuitive and user-friendly interface (Wagner et al. 2005). This is possible because an AR interface dynamically superimposes interactive computer graphics images onto objects in the real world (Poupyrev et al. 2002).

AR CORE

Fundamentally, AR Core is doing two things: tracking the position of the mobile device as it moves, and building its own understanding of the real world. AR Core's motion tracking technology uses the phone's camera to identify interesting points, called features, and tracks how those points move over time. With a combination of the movement of these points and readings from the phone's inertial sensors, AR Core determines both the position and orientation of the phone as it moves through space.

In addition to identifying key points, AR Core can detect flat surfaces, like a table or the floor, and can also estimate the average lighting in the area around it. These capabilities combine to enable AR Core to build its own understanding of the world around it.

AR Core's understanding of the real world lets you place objects, annotations, or other information in a way that integrates seamlessly with the real world. You can place a napping kitten on the corner of your coffee table, or annotate a painting with biographical information about the artist. Motion tracking means that you can move around and view these objects from any angle, and even if you turn around and leave the room, when you come back, the kitten or annotation will be right where you left it.

AR Core provides SDKs for many of the most popular development environments. These SDKs provide native APIs for all of the essential AR features like motion tracking, environmental understanding, and light estimation. With these capabilities you can build entirely new AR experiences or enhance existing apps with AR features.

NAV MESH

A navigation mesh is a collection of two-dimensional convex polygons (a polygon mesh) that define which areas of an environment are traversable by agents. In other words, a character in a game could freely walk around within these areas unobstructed by trees, lava, or other barriers that are part of the environment. Adjacent polygons are connected to each other in a graph.

Pathfinding within one of these polygons can be done trivially in a straight line because the polygon is convex and traversable. Pathfinding between polygons in the mesh can be done with one of the large number of graph search algorithms, such as A*. Agents on a navmesh can thus avoid computationally expensive collision detection checks with obstacles that are part of the environment.

Representing traversable areas in a 2D-like form simplifies calculations that would otherwise need to be done in the "true" 3D environment, yet unlike a 2D grid it allows traversable areas that overlap above and below at different heights. The polygons of various sizes and shapes in navigation meshes can represent arbitrary environments with greater accuracy than regular grids can.

2.2 ALGORITHM

A* Algorithm

- A* is an informed search algorithm, or a best-first search, meaning that it is formulated in terms of weighted graphs: starting from a specific starting node of a graph, it aims to find a path to the given goal node having the smallest cost.
- It does this by maintaining a tree of paths originating at the start node and extending those paths one edge at a time until its termination criterion is satisfied.

CHAPTER 3

SYSTEM REQUIREMENTS AND SPECIFICATIONS

HARDWARE REQUIREMENTS

- Processor: Minimum 1.7 GHz; recommended 2GHz (or more)
- Hard Drive: 100GB
- Memory (RAM): 4GB or more
- Android Smartphone

SOFTWARE REQUIREMENTS/ TOOLS AND TECHNOLOGIES

- OS : Windows 10 / MacOS / Linux
- Unity 2021.3.12f1
- Visual Studio Code
- Microsoft Paint
- Android Version 7.0 or above

CHAPTER 4

SYSTEM DESIGN

ARCHITECTURE

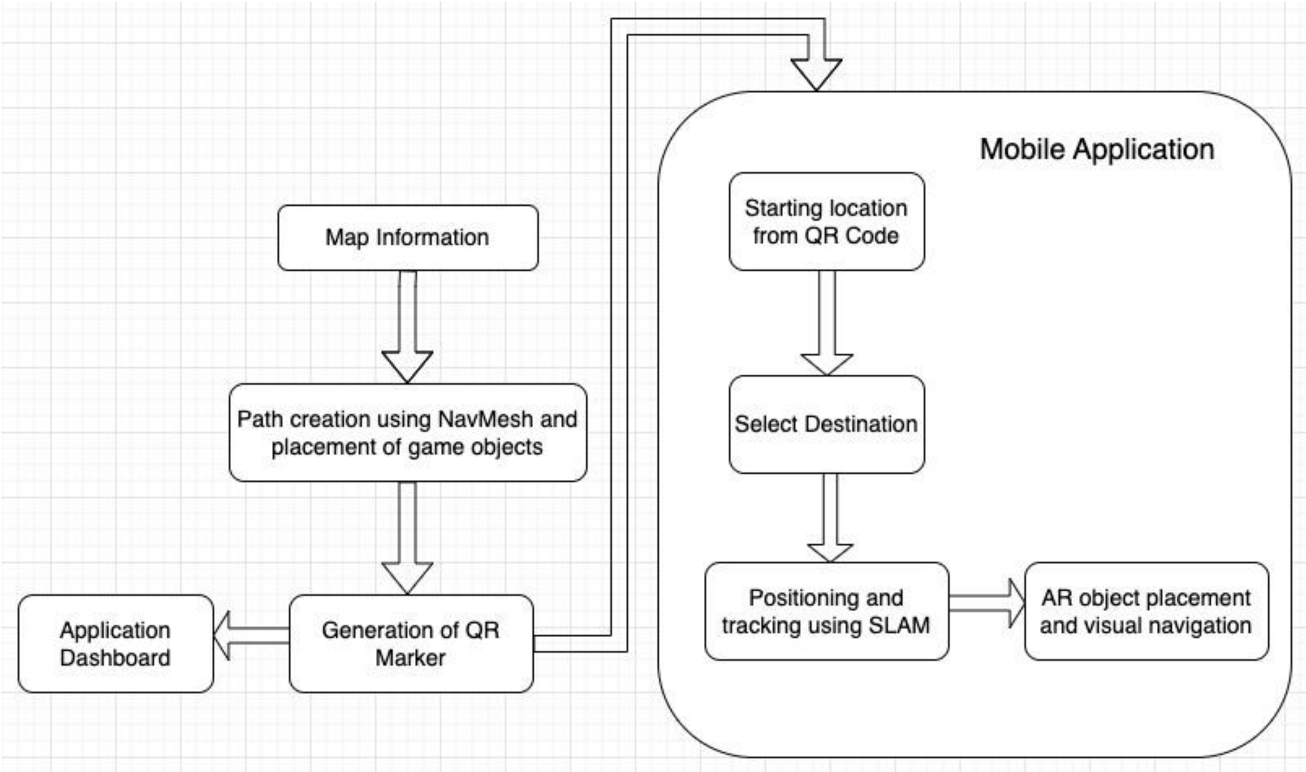


Figure 4.1 : Architecture

- Mobile Application block hosts the front end and user interface of the application.
- The users need to download and install the application using the QR-code provided.
- After opening the installed application, user can see their camera getting started.
- User can choose the point of interest to navigate the indoors of the building.
- The application then projects the shortest path to the point of interest by means of arrow indications.
- Since the map information is stored in the application there will be no need of network after downloading and installing the application.
- The path generated by the application uses NavMesh and placement of the game objects in the unity editor.

UML DIAGRAMS

UML stands for Unified Modeling Language. UML is a standardized general-purpose modeling language in the field of object-oriented software engineering. The standard is managed, and was created by, the Object Management Group.

The goal is for UML to become a common language for creating models of object-oriented computer software. In its current form UML is comprised of two major components: A Meta-model and a notation. In the future, some form of method or process may also be added to; or associated with, UML.

The Unified Modeling Language is a standard language for specifying, Visualization, Constructing and documenting the artifacts of software system, as well as for business modeling and other non- software systems.

The UML represents a collection of best engineering practices that have proven successful in the modeling of large and complex systems.

The UML is a very important part of developing objects-oriented software and the software development process. The UML uses mostly graphical notations to express the design of software projects.

USE CASE DIAGRAM

Use case diagrams are a set of use cases, actors, and their relationships. They represent the use case view of a system.

A use case represents a particular functionality of a system. The Indoor Navigation application has three actor's user, admin and system. User and system both have the same view of the application where the main interface of the application could be seen. The admin has the full privileges to create, update and delete the map of the mobile application.

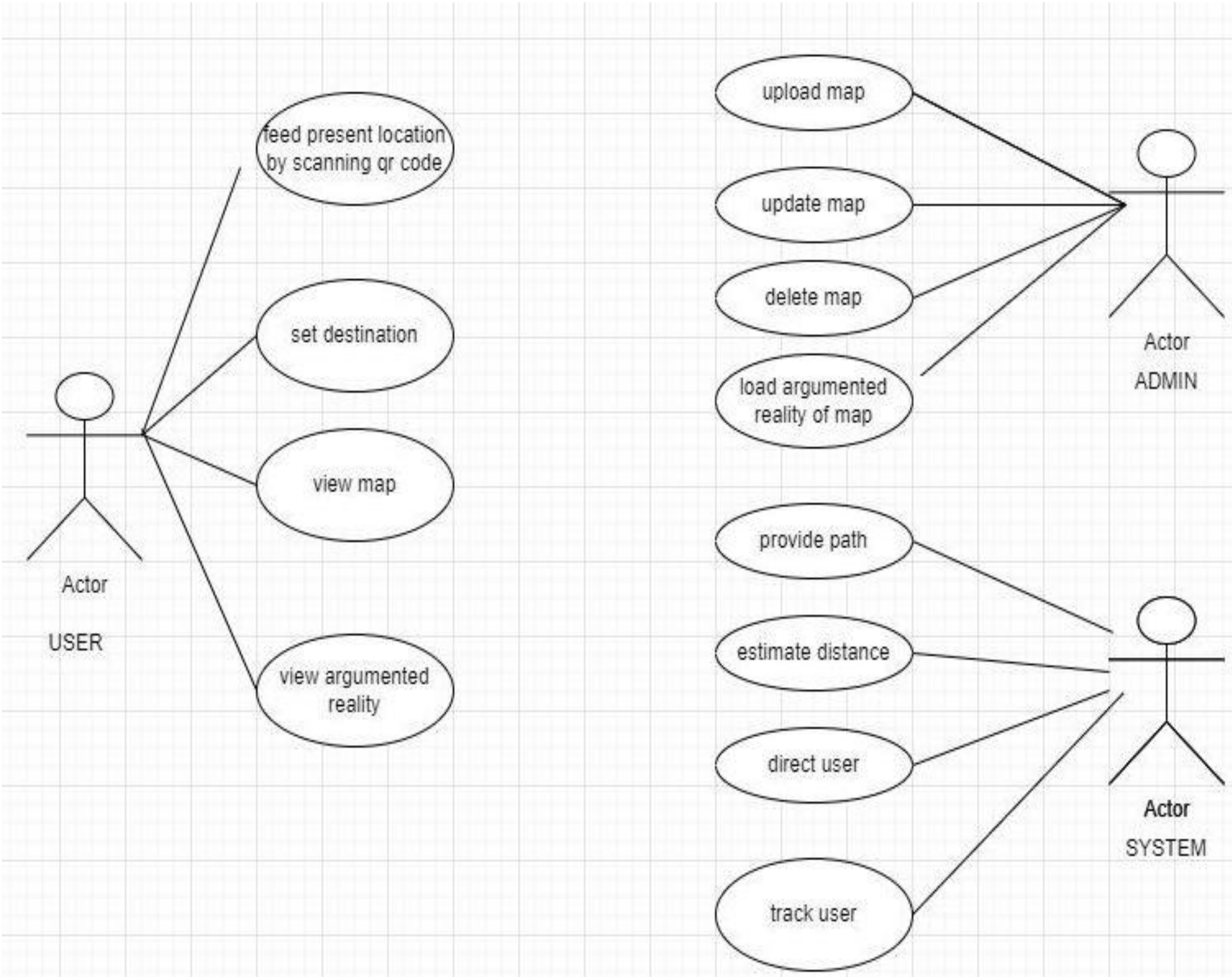


Figure 4.2: Use Case Diagram

SEQUENCE DIAGRAM

A sequence diagram in Unified Modelling Language (UML) is a kind of interaction diagram that shows how processes operate with one another and in what order. It is a construct of a Message Sequence Chart. Sequence diagrams are sometimes called event diagrams, event scenarios, and timingdiagrams.

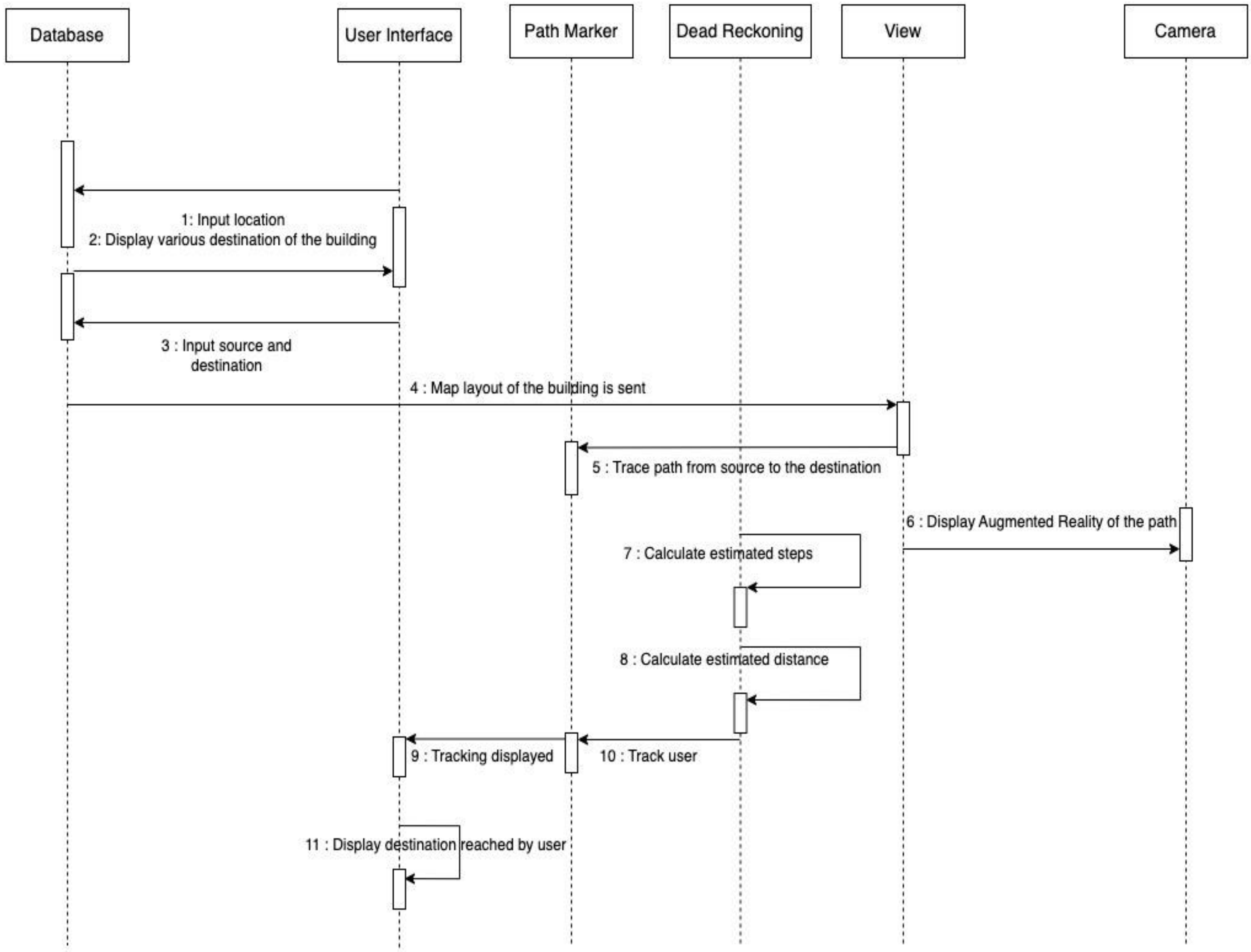


Figure 4.3: Sequence Diagram

ACTIVITY DIAGRAM

Activity diagram describes the flow of control in a system. It consists of activities and links. The flow can be sequential, concurrent, or branched. Activities are nothing but the functions of a system. Numbers of activity diagrams are prepared to capture the entire flow in a System.

A user/visitor must scan the QR-Code to download and install the mobile application for indoor navigation. The application then begins preprocessing of the indoor map and initializes the camera of the smartphone. SLAM is used as part of the AR Core functionality in order to determine the user's location and project the shortest distance to the user/visitor's point of interest.

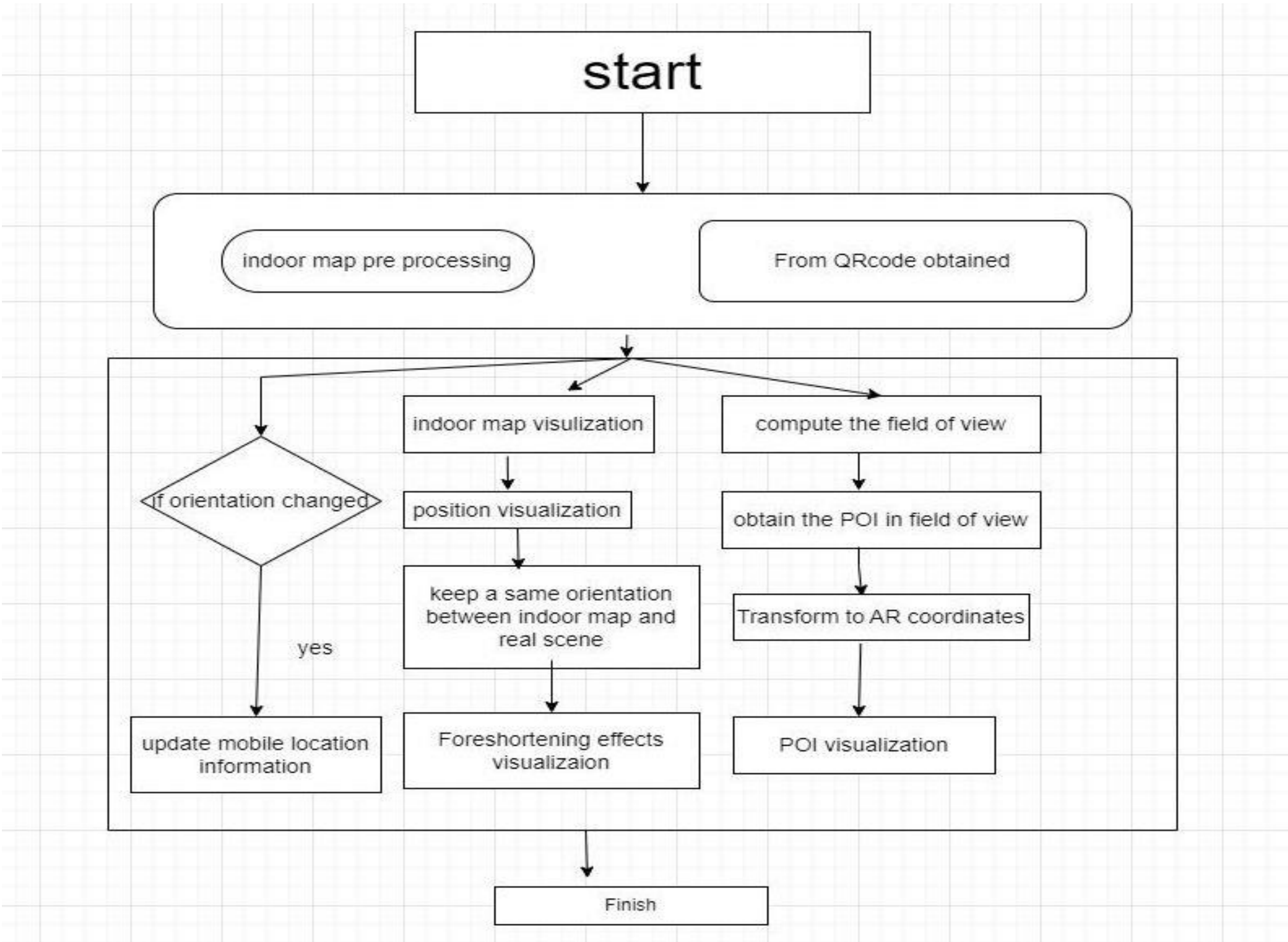


Figure 4.4: Activity Diagram

CHAPTER 5

IMPLEMENTATION

ENVIRONMENTAL SETUP

Installing Visual Studio Code:

1. To download and install VS Code visit the official website of VS Code https://code.visualstudio.com/

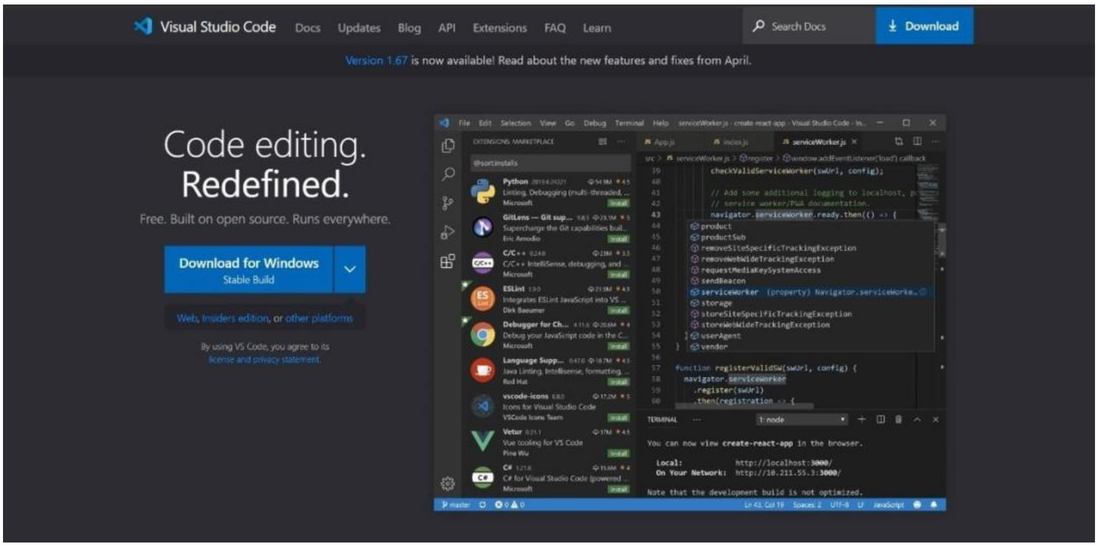


Figure 5.1: VS Code

2. Once the download is complete, run the exe for installing VS Code. Now click on install now.
3. You can see VS Code installing at this point.
4. When it finishes, you can see a screen that says the setup was successful.

Installing Unity Hub:

1. To download and install Unity Hub visit the official website of Unity https://unity.com/

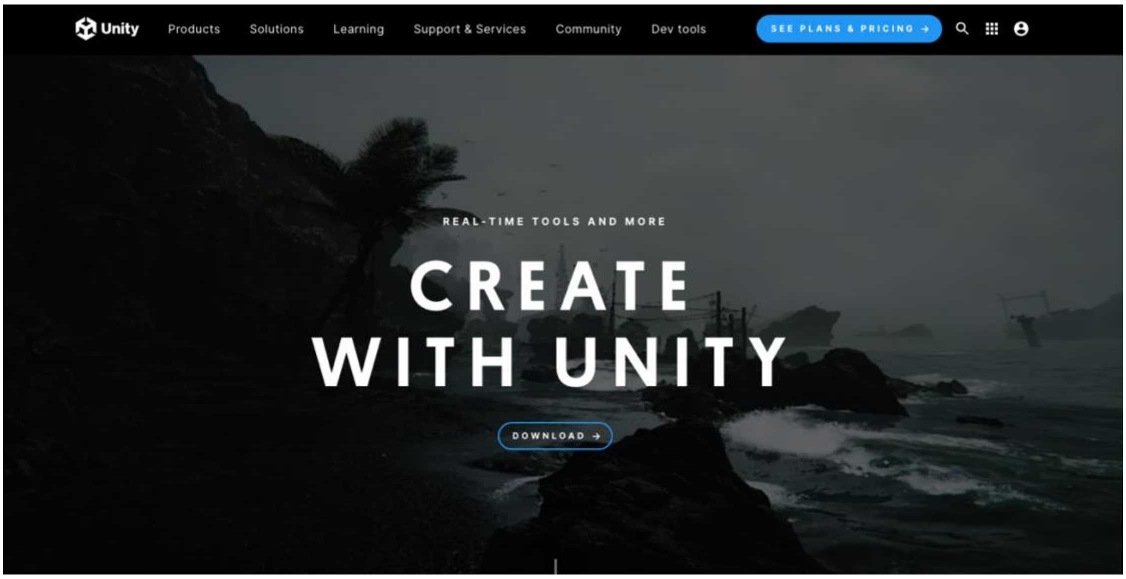


Figure 5.2: Unity Hub

2. Once the download is complete, run the exe for installing Unity Hub. Now click on install now.
3. You can see Unity Hub installing at this point.
4. When it finishes, you can see a screen that says the setup was successful.

Installing Unity Editor:

1. To download and install Unity Editor go to "Install Editor" and install the version 2021.3.24f1.

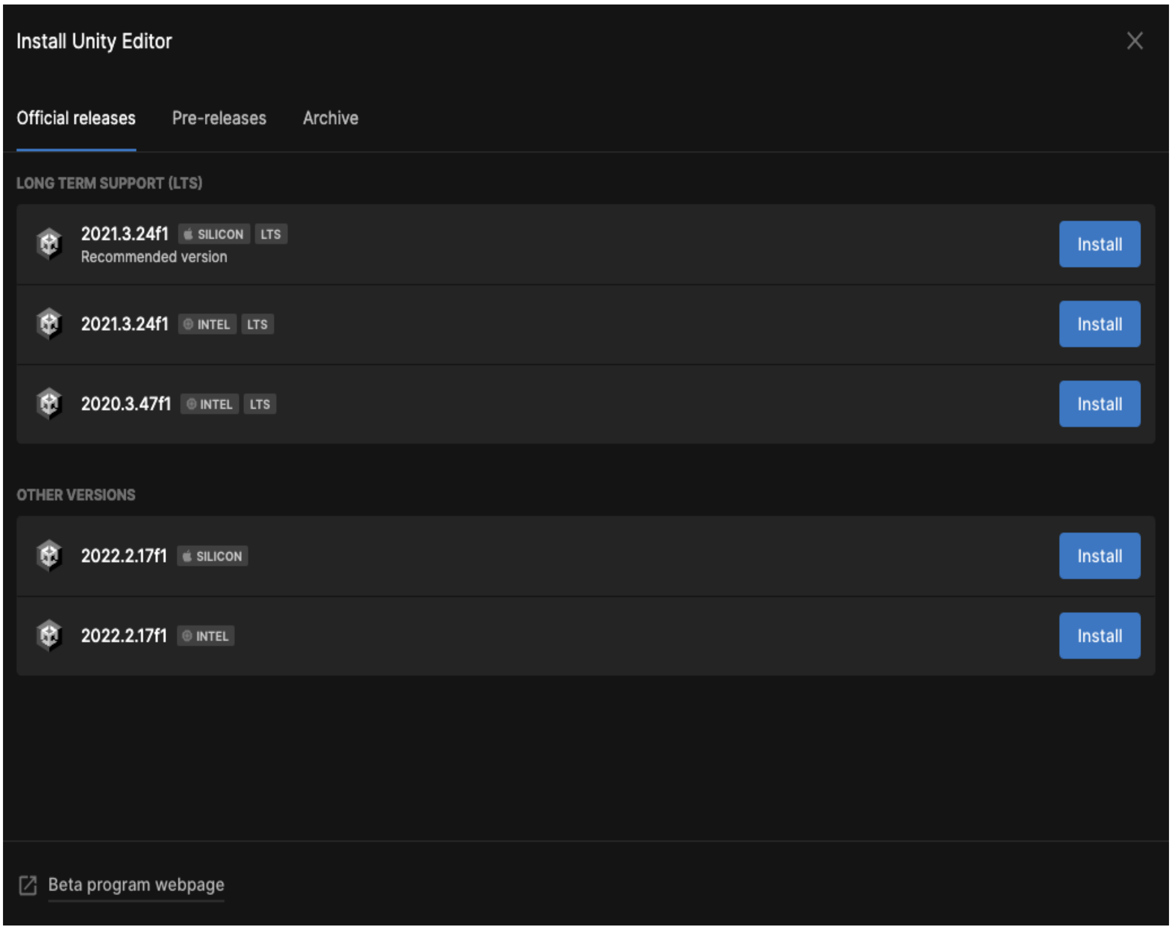


Figure 5.3: Unity Editor

2. Once the installation is complete, the project can be opened to customize the elements of indoor navigation.

Installing AR Foundation Package:

1. To install the AR Foundation package Window -> Package manager.
2. Select AR Foundation to install and import all the functionalities of AR Core.

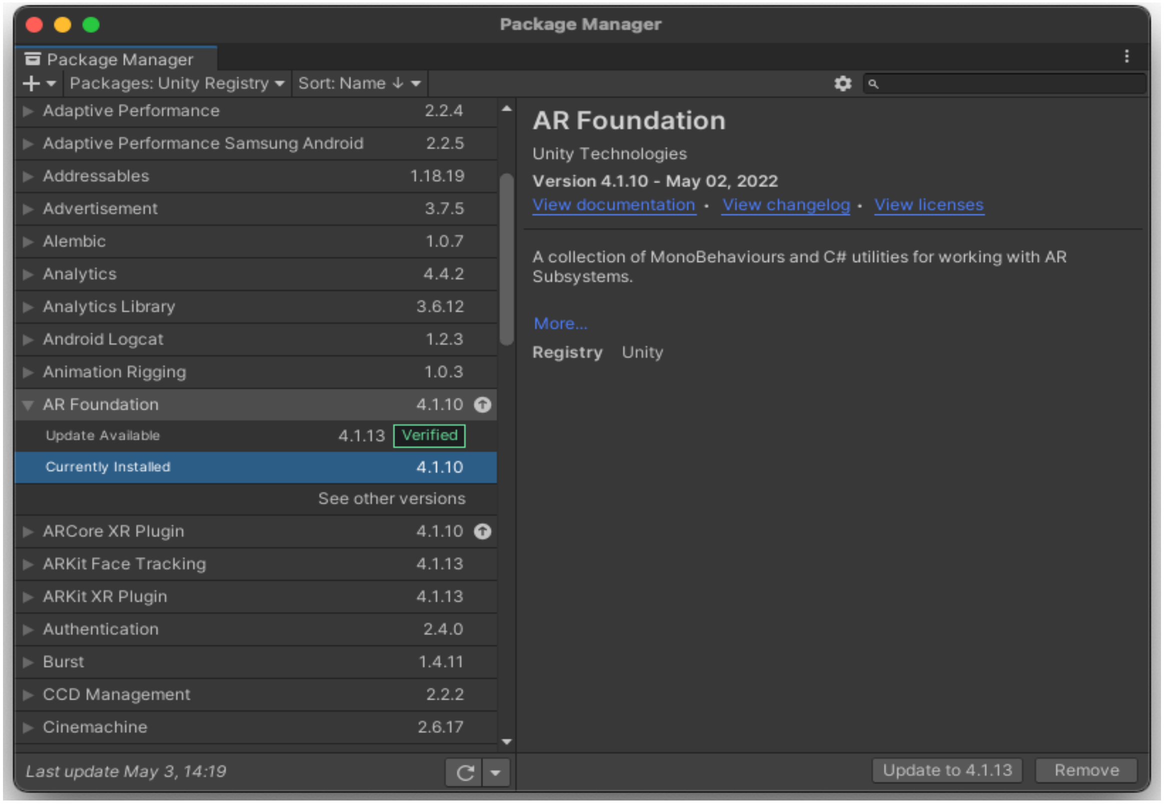


Figure 5.4: AR Foundation package

3. After installing the package AR Core modules are imported into the project file.

MODULE DESCRIPTION

1. **SCAN QR - CODE**

   The user is required to scan the QR code in order to install the android mobile application and start off with the user's current location to start navigating the indoors of the buildings.

2. **PERMISSIONS**

   Upon installation of the mobile application, the user will be prompted to give required permissions for accessing the camera and settings of the smartphone as part of the augmented reality functionalities to be working.

3. **HOME INTERFACE**

   The user after he/she grants the required permissions, will be taken to the home interface of the mobile application where the user can see the camera getting started along with the mini-map displayed at the bottom along with a drop-down menu.

4. **POINT OF INTEREST**

   The user can now access the drop-down menu to select the respective point of interest to navigate the location in the shortest path possible without any GPS or network connectivity.

5. **TOGGLE PATH**

   After the user selects the point of interest to navigate, a toggle line visibility button is provided to toggle the path via Augmented Reality i.e., to provide a project of arrow indications along the way until the destination is reached by the user.

CHAPTER 6
RESULTS (SCREENSHOTS)

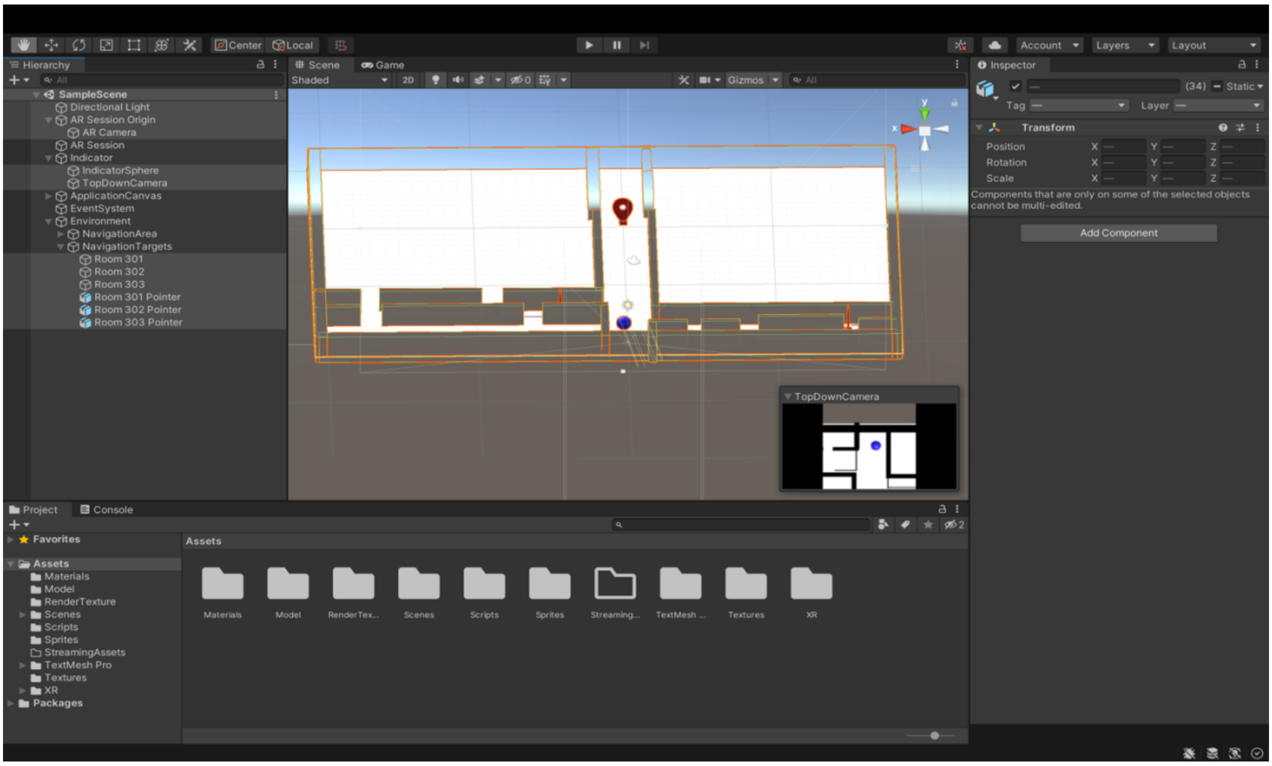


Figure 6.1:Unity Home Page

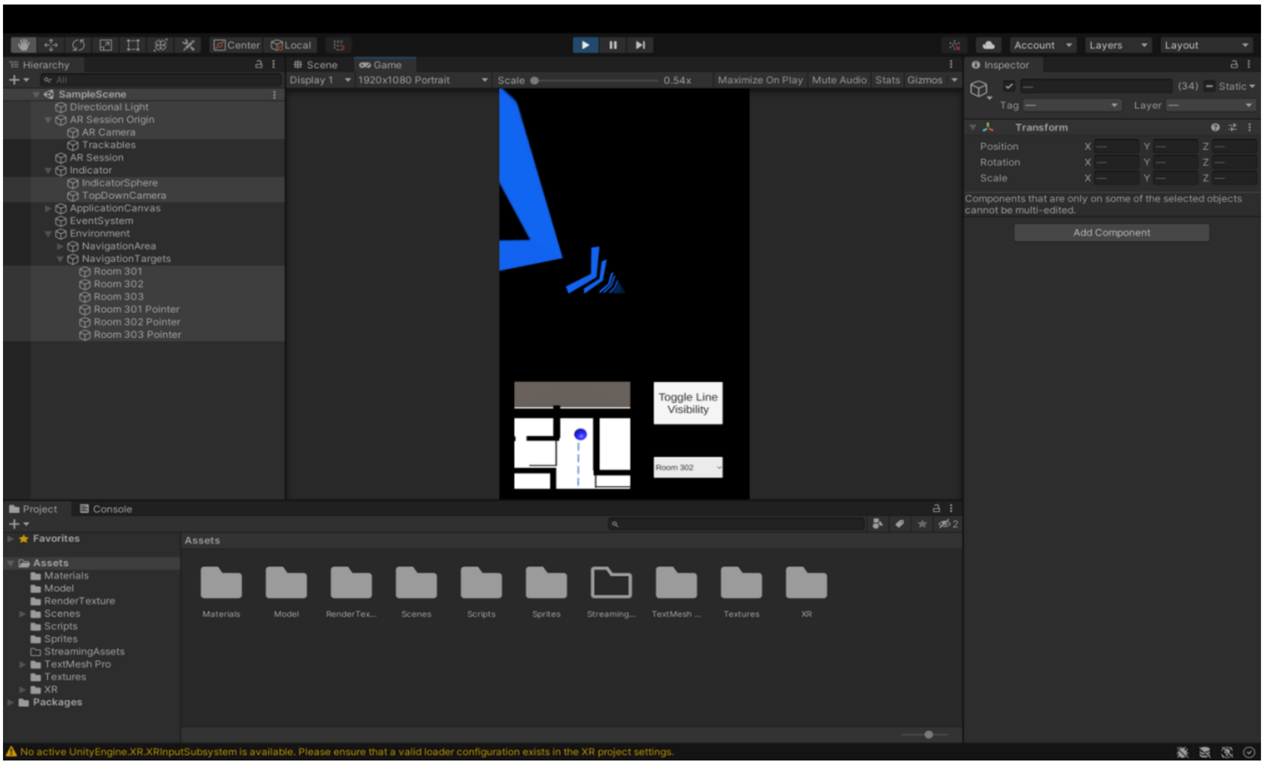


Figure 6.2: Unity Debug Page

Figure 6.3: QR Code to install application

Figure 6.4: Application Start-up Page

Figure 6.5: Selecting Point of Interest

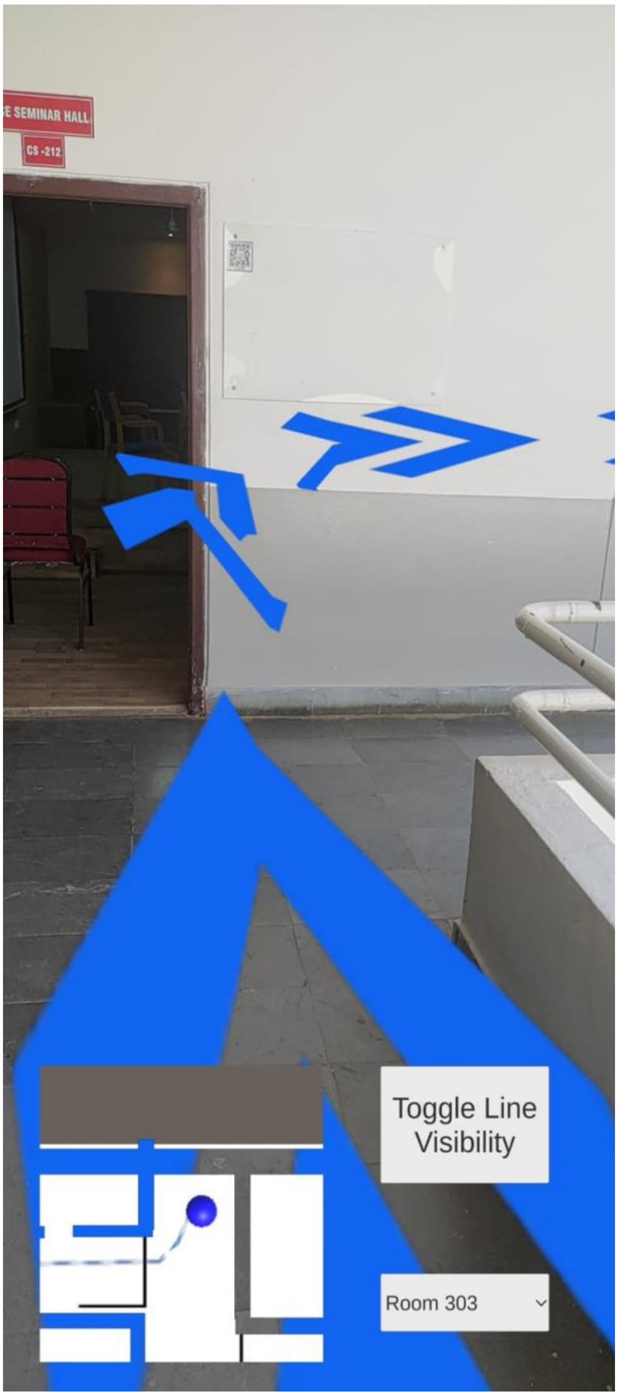


Figure 6.6: Projected path

Figure 6.7: Reaching Destination

CHAPTER 7

CONCLUSION

In this project, a major problem for navigating inside the buildings is the main focus. The conventional system majorly uses GPS technology for navigation, which is unreliable inside the buildings. To tackle these, Augmented Reality is being used for the purpose of indoor navigation. An AR interface could be a new visualization method for Geographical Information System data such as point of interest locations, directions, and so on. In addition, an AR navigation system can be efficiently implemented within defined areas such as university campuses, shopping centers, museums, and so on. In cases where high-accuracy positioning is not required, QR codes could be the cheapest and easiest positioning method for an indoor navigation system. Today's mobile application ecosystem allows users to download mobile applications ubiquitously. The project is still in the designing phase, for further implementation the creation of an android application using Unity for client, deployment of NavMesh to handle the designed map and ARCore to enhance the user interaction will be the future goals.

FUTURE ENHANCEMENTS

- This prototype is designed for same floor level navigation only, future works can be done to expand it to multi-floor navigation.
- Additional features such as audio navigation can also be implemented, which could be prompted by using voice assistant like Google, Siri, etc.

REFERENCES

[1] M. B. Kjærgaard, H. Blunck, T. Godsk, T. Toftkjær, D. L. Christensen, and K. Grønbæk, "Indoor Positioning Using GPS Revisited," in Pervasive 2010.

[2] D. Macias-Valadez, R. Santerre, S. Larochelle, and R. Landry, "Improving vertical GPS precision with a GPS-over-fiber architecture and real-time relative delay calibration," GPS Solutions, vol. 16, no. 4, pp. 449–462, 2012.

[3] C. Yang and H. R. Shao, "WiFi-based indoor positioning," IEEE Communications Magazine, vol. 53, no. 3, pp. 150–157, 2015.

[4] C. Perey and T. Miyashita," Indoor positioning and navigation for mobile AR," 2011 IEEE International Symposium on Mixed and Augmented Reality - Arts, Media, and Humanities, 2011, pp. 1-1, doi: 10.1109/ISMAR-AMH.2011.6093646.

[5] Mehdi Mekni and Andre Lemieux," Augmented reality: Applications challenges and future trends", Applied Computational Science— Proceedings of the 13th International Conference on Applied Computer and Applied Computational Science (ACACOS '14), pp. 205- 209, 2014.

[6] SA Habsi, M Shehada, M Abdoon, A Mashood, H Noura. Integration of a Vicon camera system for indoor flight of a Parrot AR Drone. 2015 10th International Symposium on Mechatronics and its Applications (ISMA). Sharjah. 2015.

APPENDIX

CODE:

Navigation Target Script -

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


public class SetNavigationTarget : MonoBehaviour
{
    [SerializeField]
    private TMP_Dropdown navigationTargetDropDown;
    [SerializeField]
    private List<Target> navigationTargetObjects = new List<Target>();


    private NavMeshPath path;
    private LineRenderer line;
    private Vector3 targetPosition = Vector3.zero;


    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;
      line.SetPositions(path.corners);

    }
    }



    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) {
      targetPosition = currentTarget.PositionObject.transform.position;

    }
    }



    public void ToogleVisibility() {
    lineToggle = !lineToggle;
    line.enabled = lineToggle;
    }
}
```

QR Code Recenter Script –

```
using Unity.Collections;
using UnityEngine;
using UnityEngine.XR.ARFoundation;
using UnityEngine.XR.ARSubsystems;
using ZXing;

public class QrCodeRecenter : MonoBehaviour {

[SerializeField]
private ARSession session;
[SerializeField]
private ARSessionOrigin sessionOrigin;
[SerializeField]
private ARCameraManager cameraManager;
[SerializeField]
private TargetHandler targetHandler;
[SerializeField]
private GameObject qrCodeScanningPanel;
private Texture2D cameraImageTexture;
private IBarcodeReader reader = new BarcodeReader(); // create a barcode reader instance
private bool scanningEnabled = false;

private void OnEnable() {
cameraManager.frameReceived += OnCameraFrameReceived;
}

private void OnDisable() {
cameraManager.frameReceived -= OnCameraFrameReceived;
}

private void OnCameraFrameReceived(ARCameraFrameEventArgs eventArgs) {

if (!scanningEnabled) {
return;
```

```
}

if (!cameraManager.TryAcquireLatestCpuImage(out XRCpuImage image)) {
return;
}

var conversionParams = new XRCpuImage.ConversionParams {
// Get the entire image.
inputRect = new RectInt(0, 0, image.width, image.height),

// Downsample by 2.
outputDimensions = new Vector2Int(image.width / 2, image.height / 2),

// Choose RGBA format.
outputFormat = TextureFormat.RGBA32,

// Flip across the vertical axis (mirror image).
transformation = XRCpuImage.Transformation.MirrorY
};
int size = image.GetConvertedDataSize(conversionParams);

// Allocate a buffer to store the image.
var buffer = new NativeArray<byte>(size, Allocator.Temp);

// Extract the image data
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();

// Detect and decode the barcode inside the bitmap
var result = reader.Decode(cameraImageTexture.GetPixels32(), cameraImageTexture.width,
cameraImageTexture.height);

if (result != null) {
SetQrCodeRecenterTarget(result.Text);
ToggleScanning();
}
}

private void SetQrCodeRecenterTarget(string targetText) {
TargetFacade currentTarget = targetHandler.GetCurrentTargetByTargetText(targetText);
if (currentTarget != null) {
// Reset position and rotation of ARSession
session.Reset();

// Add offset for recentering
sessionOrigin.transform.position = currentTarget.transform.position;
sessionOrigin.transform.rotation = currentTarget.transform.rotation;
}
}

public void ChangeActiveFloor(string floorEntrance) {
SetQrCodeRecenterTarget(floorEntrance);
}

public void ToggleScanning() {
scanningEnabled = !scanningEnabled;
qrCodeScanningPanel.SetActive(scanningEnabled);
}
}
```