A Project Report on

AR App for College Campus Navigation

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

Bachelor of Engineering

in

Information Technology

by

Sameer Sawant(19104054)
Raj Shisode(19104070)
Prathmesh Shrirao(19104016)

Under the Guidance of

Prof.Mandar Ganjapurkar
Prof.Shweta Mahajan

Department of Information Technology
NBA Accredited
A.P. Shah Institute of Technology
G.B.Road,Kasarvadavli, Thane(W), Mumbai-400615
UNIVERSITY OF MUMBAI

Academic Year 2022-2023

Approval Sheet

This Project Report entitled ***"AR App for College Campus Navigation"*** Submitted by ***"Sameer Sawant"(19104054), "Raj Shisode"(19104070), "Prathmesh Shrirao"(19104016)*** is approved for the partial fulfillment of the requirement for the award of the degree of ***Bachelor of Engineering*** in ***Information Technology*** from ***University of Mumbai***.

(Prof.Shweta Mahajan)
Co-Guide

(Prof.Mandar Ganjapurkar)
Guide

Dr. Kiran Deshpande
Head of Department of Information Technology

Place: A.P. Shah Institute of Technology, Thane
Date:

CERTIFICATE

This is to certify that the project entitled ***"AR App for College Campus Navigation"*** submitted by ***"Sameer Sawant" (19104054)***, ***"Raj Shisode" (19104070)***, ***"Prathmesh Shrirao" (19104016)*** for the partial fulfillment of the requirement for award of a degree ***Bachelor of Engineering*** in ***Information Technology***,to the University of Mumbai,is a bonafide work carried out during academic year 2022-2023.

(Prof. Shweta Mhajan)
Co-Guide

(Prof.Mandar Ganjapurkar)
Guide

Dr. Kiran Deshpande
Head Department of Information Technology

Dr. Uttam D.Kolekar
Principal

External Examiner(s)

1.

2.

Place:A.P.Shah Institute of Technology, Thane
Date:

Acknowledgement

We have great pleasure in presenting the report on **AR App for College Campus Navigation.** We take this opportunity to express our sincere thanks towards our guide **Prof Mandaar Ganjapurkar** & Co-Guide **Prof. Shweta Mahajan** Department of IT, APSIT thane for providing the technical guidelines and suggestions regarding line of work. We would like to express our gratitude towards his constant encouragement, support and guidance through the development of project.

We thank **Dr. Kiran B. Deshpande** Head of Department,IT, APSIT for his encouragement during progress meeting and providing guidelines to write this report.

We thank **Prof. Sonal Jain** BE project co-ordinator, Department of IT, APSIT for being encouraging throughout the course and for guidance.

We also thank the entire staff of APSIT for their invaluable help rendered during the course of this work. We wish to express our deep gratitude towards all our colleagues of APSIT for their encouragement.

Sameer Sawant
(19104054)

Raj Shisode
(19104070)

Prathmesh Shrirao
(19104016)

Declaration

We declare that this written submission represents our ideas in our own words and where others' ideas or words have been included, We have adequately cited and referenced the original sources. We also declare that We have adhered to all principles of academic honesty and integrity and have not misrepresented or fabricated or falsified any idea/data/fact/source in our submission. We understand that any violation of the above will be cause for disciplinary action by the Institute and can also evoke penal action from the sources which have thus not been properly cited or from whom proper permission has not been taken when needed.

Sameer Sawant(19104054)

Raj Shisode(19104070)

Prathmesh Shrirao(19104016)

Date:

Abstract

Indoor positioning has gained popularity recently due to its potential to be used in the increasing complexity of indoor environments. Although GPS is widely used for navigation purposes, GPS uses satellite links for pinpointing the location, GPS is not recommended for indoor navigation because satellite signals may be difficult to penetrate tall buildings. The Augmented Reality (AR) navigation systems can provide a new experience compared to conventional navigation on 2D maps. Nowadays AR finds its involvement in various fields along with its smart application software which can be used in different fields including education, entertainment, navigation, and many more. The proposed system initially localizes and maps the nodes deployed using computer vision and the node graph of the indoor structure. It then implements an A* path-finding algorithm to route the shortest distance between the user and the destination. The proposed system helps us navigate within large buildings using cameras mounted on our smartphones.

Contents

List of Figures

List of Abbreviations

AR:	Augmented Reality
GPS:	Global Positioning System
OCR:	Optical Recognition System
VRML:	Virtual Reality Modeling language
RSSI:	Received Signal Strength Indicator
SDK:	Software Development Kit
SLAM:	Simultaneous localization and mapping
MLKit:	Machine learning kit

Chapter 1

Introduction

In recent years, the technology industry has increasingly invested in augmented reality (AR), which has been tested and applied in a number of sectors including gaming and retail.

With the ability to interact with the immediate surroundings, register, and connect real and virtual items, augmented reality combines the virtual and real worlds together. Building on this idea, augmented reality (AR) refers to the real-time integration of digital data with live video on the environment of the user. Due to its growing use in the business world, augmented reality is continually evolving. The demand for AR chips has soared exponentially as a result, particularly in industries like medicine, entertainment, education, retail, etc. Due to the increasing opportunities in the AR field which happens to be largely untapped until now, top firms in the sector including Qualcomm and NXP have released unique AR-powered graphic chips to meet end-users' evolving expectations.

With augmented reality (AR), computer-generated graphics are superimposed on the user's actual view to provide the impression that the virtual and the real world are coexisting. The advancement of the visual effect that might help people more easily assume specific objects is the driving factor behind this technology. In many large buildings, including train stations, malls, hospitals, and governmental structures, indoor navigation systems have been installed as a result of technological advancements in the field of computer science and business potential. Users of mobile navigation apps can choose a starting point and an ending point within the program. The system will then choose the shortest course to take in order to arrive at the destination. Nowadays, the most commonly used user interface of navigation applications is a 2D map with a route. Users are provided with navigation instructions, such as turn left, turn right, and go straight when they are close to an intersection. However, due to the limitations of a 2D navigation map, it could add an additional cognitive load for users to construct the relationship between the 2D navigation map and the real environment. Additionally, users may experience increased mental stress and become disoriented. Therefore, it's crucial for system design to get rid of any potential user confusion. A number of research projects have been devoted to building an interior navigation system using augmented reality (AR) technology in order to produce a positive user experience. This project's major objective is to show people directions on a screen that is superimposed over actual environments as seen by a smartphone or headset camera.

Chapter 2

Literature Review

The primary goal of this literature review is to gain an understanding of the existing research on Augmented Reality and to understand the area of study. The literature review helped in selecting the appropriate methodology for the positioning and navigation processes to achieve efficient results..

1. In Paper [1], “Indoor Navigation system using visual position system with augmented reality” [1], Published in 2018 by Ravinder Yadav, Vandit Jain, and Himanika Chugh. This study suggests a development strategy for an augmented reality and image processing-based interactive indoor navigation system. In this project, computer-produced 3D graphics are superimposed on the user’s mobile screen using the AR Toolkit (open-source AR software). Then the VRML model (virtual object) is shown on the identified marker as the basic operating principle of the AR tool kit. Two crucial tasks are performed by the AR toolkit: perspective tracking and virtual object interaction. The camera angle and location can be determined in real-time using the AR toolkit. After determining the position of the actual camera, an AR tool kit is used to line up the virtual camera with the actual camera before drawing 3D computer graphics precisely over the actual marker. The suggested navigational strategy makes use of AR, which also aids in triangulating the position depending on the discovery of markers in the surrounding area.

2. In Paper [2], “AR-based Navigation Using Hybrid Map”, Published in 2022 by Yanlei Gu, Woranipit Chidsin , and Igor Goncharenko [2], This research suggests a marker-free indoor navigation system based on augmented reality. The suggested system uses SLAM (Simultaneous Localization and Mapping) to create a point cloud map, and it executes positioning and navigation using a cutting-edge hybrid map that combines 3D point cloud data and an indoor environment floor map. Positioning and navigation are controlled by the data in the floor maps and point clouds, respectively. In order to demonstrate the viability of the suggested navigation system, tests are conducted in the hallway of a substantial construction.

3. In Paper [3], “Design of a mobile Augmented Reality – based indoor navigation system”.Published 2022 by Low Chee Huey, Woan Ning Lim [3], Here AR core and Indoor Atlas Android SDKs were integrated into Android Studio to create the AR indoor navigation mobile application. The venue data was kept in a local SQLite database. While AR core SDK is used to provide augmented reality (AR) guidance on top of a real-world camera image, Indoor Atlas SDK combines magnetic fields, Wi-Fi signals, and inertial sensors to

deliver indoor positioning and path-finding services. The sensor data in the mobile device was acquired and delivered to the Indoor Atlas Cloud server when the user begins the navigation. The position and way-finding data were calculated, and the device receives feedback. The user was directed to the location by an augmented reality navigation guide after receiving the position and route.

4. In Paper [4], "An Indoor Navigation and localization system.", published 2022, by Qi – wei Bao, Chris Papachristou, Frank Wolf [4]. In this study, a composite strategy that not only combines the WiFi RSSI fingerprint with the PDR method but also automatically generates a self-adaptive system using the WiFi signals that are received to adjust the user's stride estimation based on PDR approaches is presented. The method of indoor navigation in this research paper has provided numerous components. Pre-processing of the system, which includes choosing the base station and sample sites as well as the location and sensor Approach database construction. The whole process utilizes a grid-based approach based on the Wi-Fi Received Signal Strength Indicator to gather information about the visitor's Wi-Fi signal strength (RSSI). By detecting the user's step and determining the direction and stride of each one, the PDR method is used to estimate the user's position. Here the last step is to Employ the K-Nearest Neighbour (KNN) method on the measured Wi-Fi signal strength data and the PDR-estimated Wi-Fi signal strength data in the area.

5. In Paper [5], "Feature-Based Indoor Navigation System Using Augmented Reality", Published 2019, Manas Patil, Ajay Kadam, Kalpit Choudhari. The suggested method makes use of many SDKs to produce an interior graphic representation of a building's or infrastructure's map, such as Map box or Place note. After the virtual interior has been generated using a building plan or map, it is combined with an AR support system, such as Google's AR Core. After that the integrated map is scaled to the real-world level and then the markers are inserted. Some locations on the accompanying visual map have visual markers dropped on them. Visual indicators are planted on some predefined location inside the map. In order to determine the route from the user's position to their destination, the location of the drop markers is then saved in a database. The user is then led to the goal along this path with the help of visuals that resemble an arrow.

6. In Paper [6], "An Indoor Navigation Robot Using Augmented Reality", published 2019, by Austin Corotan, Jianna Jian Zhang, Irgen-Gioro. In this paper, Technologies like AR core, JAQL and OpenGL SDK are used which are written in Kotlin and SQLite to store data and Firebase to store cloud Anchors. Dijkstra's Algorithm was used for path planning and estimating robot position and orientation. The mobile device is used both as a sensor and controller for the robot. For measuring the effectiveness, three different variables were used during the routing process based on the final location using the Q-learning algorithm and JAQL system and the real-world final location of the robot. The system was tested 10 times to check the final offset of each trial

7. In Paper [7], "Indoor Mapping and Positioning Using Augmented Reality ", published 2019, by ˙Ibrahim Alper Ko¸c, Tacha Serif, Sezer G¨oren, George Ghinea [7], In the first step, a virtual 3D coordinate system is initiated by the user's device utilizing a cell phone's camera. The device's sensors and camera can track the user's movement as well as their distance (measured in AR coordinates) from surrounding objects. The user must utilize ARKit and the device's sensors to mark each corner of the room one at a time in order

to create a map of an indoor environment. When a user launches an application, ARKit creates an ARWorld and begins polling the camera and device sensors. A Kit determines the corresponding 3D coordinate when a user marks a corner of the environment. After all, the edges are indicated, and the gadget saves the ARWorld data. Position Detection: ARKit is initialized first, and then the device is examined for AR World data. The data of an ARWorld that has already been created are loaded After that, ARKit creates the ARWorld and begins polling the sensors. The user's coordinate is recalculated using the sensor data that has been received. Route Construction. The marked region has two beacons deployed. The starting point is where the first beacon is located, and the ending point is where the second beacon is located. To begin route construction, the user must be near the first beacon. Following this, the path is drawn as the user explores the map. Up until the user reaches the second beacon, the route is sketched.

2.1 Objectives

The objectives of an augmented reality (AR) based indoor navigation system are:

- Enhance user experience: The use of AR technology in indoor navigation systems can significantly enhance the user experience by providing real-time, interactive and immersive visual aids.

- Using OCR: Implementing OCR to identify classroom numbers, can improve the efficiency and accuracy of navigation systems and help individuals find their destination more easily.

- Increase efficiency: AR-based indoor navigation systems can help users navigate through complex indoor environments more efficiently by providing the shortest and most optimized route to their destination.

- Render directions: Once the route has been calculated, the navigation system then renders directions on the screen in a clear and user-friendly way. This may include turn-by-turn directions and visual cues such as arrows and landmarks.

Chapter 3

Project Design

The project's goal is to use augmented reality (AR) technology to give university students, professors, and visitors a fresh and cutting-edge method to explore the campus structures. Users can more easily find their way around the university buildings owing to the project's implementation of augmented reality navigation following a previously built corridor graph.

3.1 Existing System

Whenever we visit any big structural premises, we often find ourselves struggling to find an accurate location to our destination without taking any help from external sources. We may also try to ask for direction from someone who is well-versed with the location. If the exact same situation had popped up in any outdoor open area we could have taken assistance from apps like google maps or any other web mapping platform. In the present conditions, there are very few indoor navigation systems available. To add on, Most indoor navigation systems use 2d navigation for pathfinding. In 2d navigation, we constantly have to map the location inside the system with our real-time location. This can be time-consuming as well as confusing. Also if there is any huge Fest, or Event gathering inside an indoor building it becomes very chaotic to ask for directions to our destination. So, how do we solve these existing problems with our application? We would assist the user by providing an augmented reality-based interface. So that they can use their mobile phone camera to navigate through their directions in an indoor compound without any interference. To achieve this we would provide direction from the user's location to the destination(probably the shortest path) through the mobile camera as the user travels through the predefined nodes until the decided final location is reached. This is far more convenient than the preexisting system because the proposed navigation app doesn't need any sort of remapping location with the current location and as well any involvement from the outside.

3.2 Proposed System

It is necessary to confirm the detected classroom number and its location in 3D space because there is a chance that the planes around the user could be determined incorrectly. This makes sure that the user is beginning the journey at the appropriate place. To ascertain the user's present location in relation to the graph of saved classrooms, initialization is also necessary.

The user can choose the route's start and finish points when initialization is finished. The user is then shown the path that was created using the SceneView AR framework in 3D space. The shortest path between the chosen start and finish sites is determined using the A* algorithm. This guarantees that users can get there in the fastest amount of time.

Bezier curves are also used for path smoothing to make the route appear more streamlined and natural. This enhances the user experience and makes it simpler to follow the route. The application administrator manually creates the classroom graph using a unique interface. This makes it possible to customize the navigation paths and guarantees that they are precise and relevant.

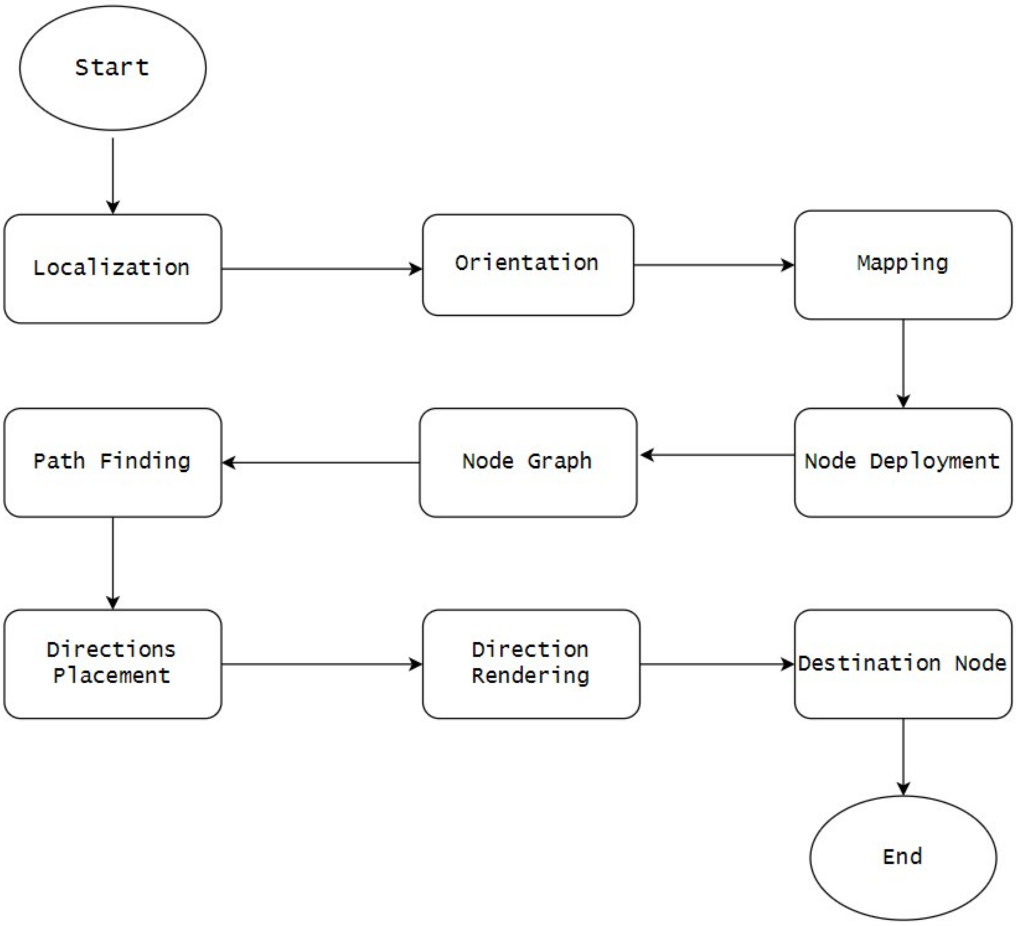


Figure 3.1: Flow-Diagram

A flow diagram, also known as a flowchart or process diagram, is a graphical representation of a process or system using symbols and arrows to show the sequence of steps or stages. As shown in Figure 3.1, the system performs simultaneous localization and mapping(SLAM) and deploys a node grid for pathfinding and direction rendering to navigate the user to the destination.

• Activity Diagram

An activity diagram is a flowchart that represents the dynamic processes of a system. The activity here refers to a system action, and it represents the flow of the diagram from one activity to the next. Though unlike the flowchart, it includes additional flows such as branching, parallel flow, concurrent flow, and so on. It helps in understanding the flow and sequence of activities, making it easier to plan and communicate how something should be done.

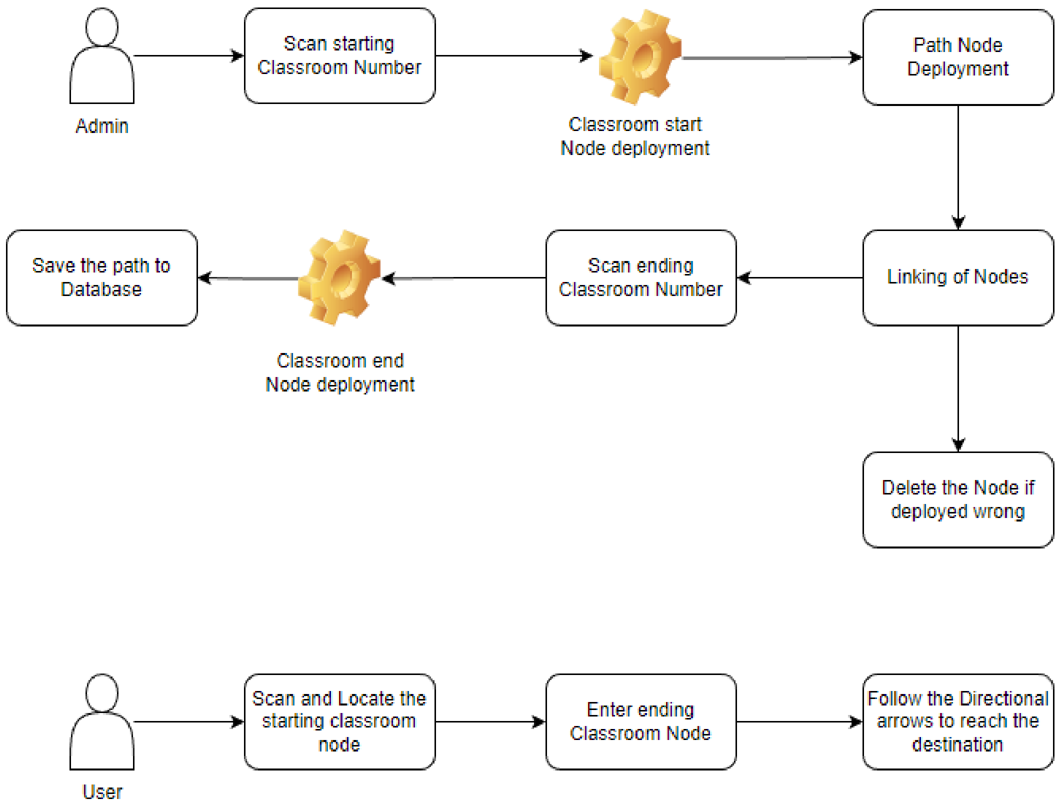


Figure 3.2: System Architecture

The Structure and Flow of our app are represented in Figure 3.2, The process of initializing the system in Admin mode involves several steps that must be carried out correctly to ensure that the system is accurate and effective. First, the developer scans the classroom number, which serves as the starting point for the navigation system. From there, the pathfinding algorithm deploys various nodes along the path to develop a route for the user to follow. These nodes can be linked together to create a seamless path, and they can also be deleted if they do not meet user expectations or are found to be unnecessary.

Once the path is complete, the ending classroom node is scanned and deployed, finalizing the route. The entire path is then saved in the database for the end user experience, ensuring that the same route can be accessed and followed by multiple users.

- Use Case Diagram

A use case diagram illustrates the various ways that a user may interact with a system and presents the details of the interaction in a compact form. These diagrams are very helpful in illustrating system-user interactions and their goals, organising system requirements, and displaying a simplified flow of events that will take place. In simple terms, a use case diagram illustrates the various actions that users can take and the system's responses, helping to understand how the system is used and how it should behave from the user's perspective.

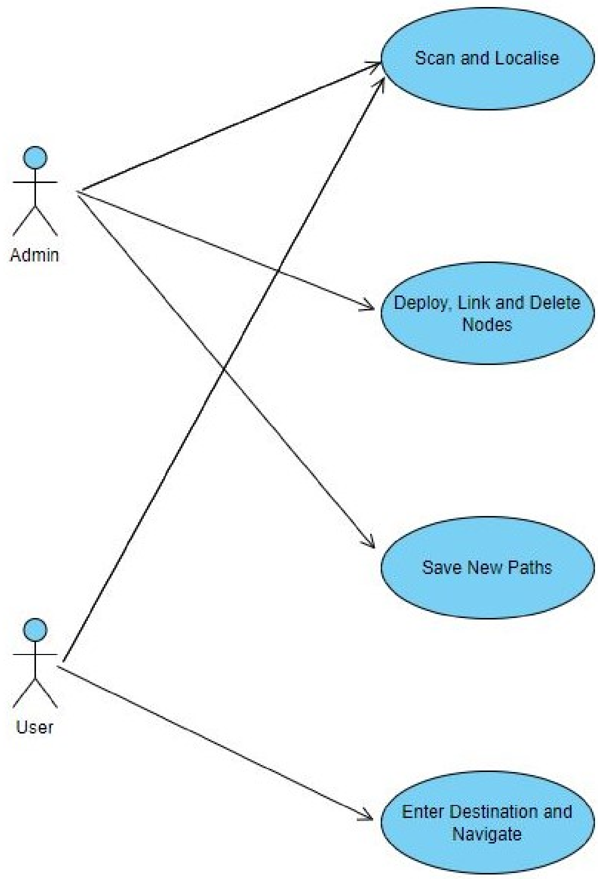


Figure 3.3: Use Case Diagram

Figure 3.3 provides an overview of the functionality available to Admins and users. The Admin can scan and localize to deploy and link new nodes or delete unnecessary nodes. By using the newly created node grid, the admin can save new paths to the database ensuring it can be accessed and followed by new users. For the end user, the process of using the system involves first scanning the classroom number to localize themselves within the system. They then select their final destination, and the navigation system superimposes directional arrows onto the user's device or screen to guide them to the required classroom using A* path-finding algorithm on the nodes deployed by the admin. By following the arrows and the path laid out by the system, the user can easily and accurately navigate from one classroom to another. In summary, the AR navigation project across university buildings is a practical and useful tool that makes it easy for users to go around the campus using the latest tools like Google MLKit and the SceneView library.

- Sequence Diagram

A sequence diagram is a visual representation that shows the interaction and order of messages exchanged between objects or components in a system. It illustrates the flow of events and the sequence in which these events occur in a time-ordered manner. In simpler terms, a sequence diagram helps to understand how different parts of a system communicate with each other and the specific order in which they do so. It visually represents the interactions and collaborations between objects or components, making it easier to understand the system's behavior and the flow of information between its various parts. Figure 3.4

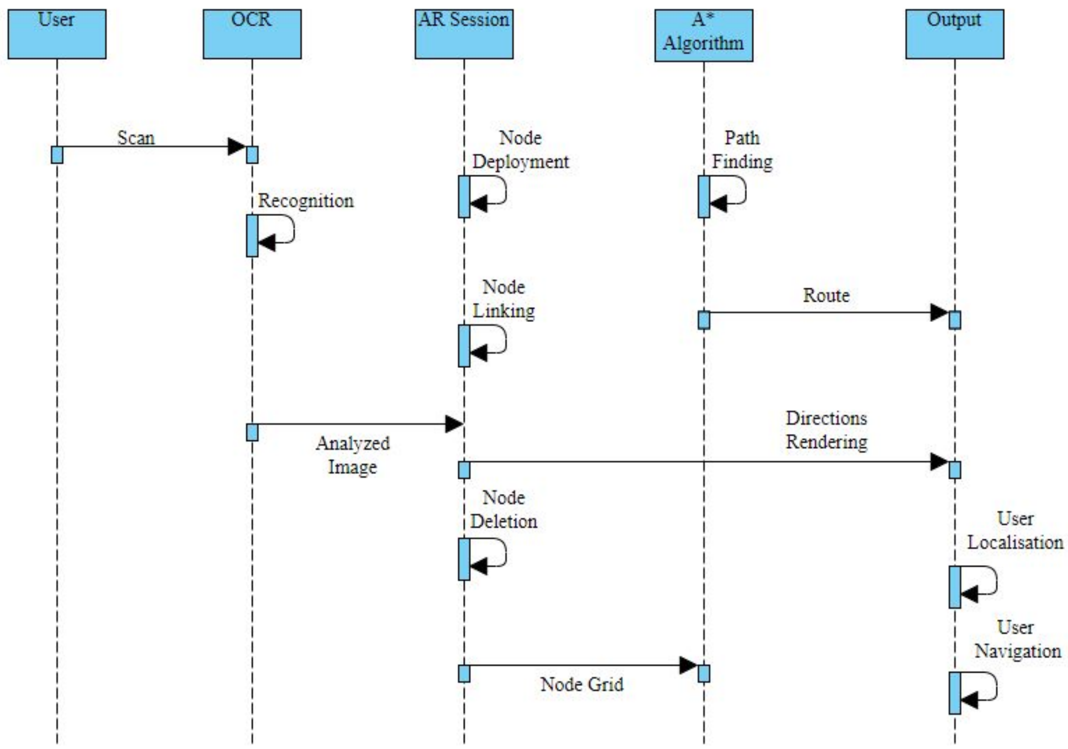


Figure 3.4: Sequence Diagram

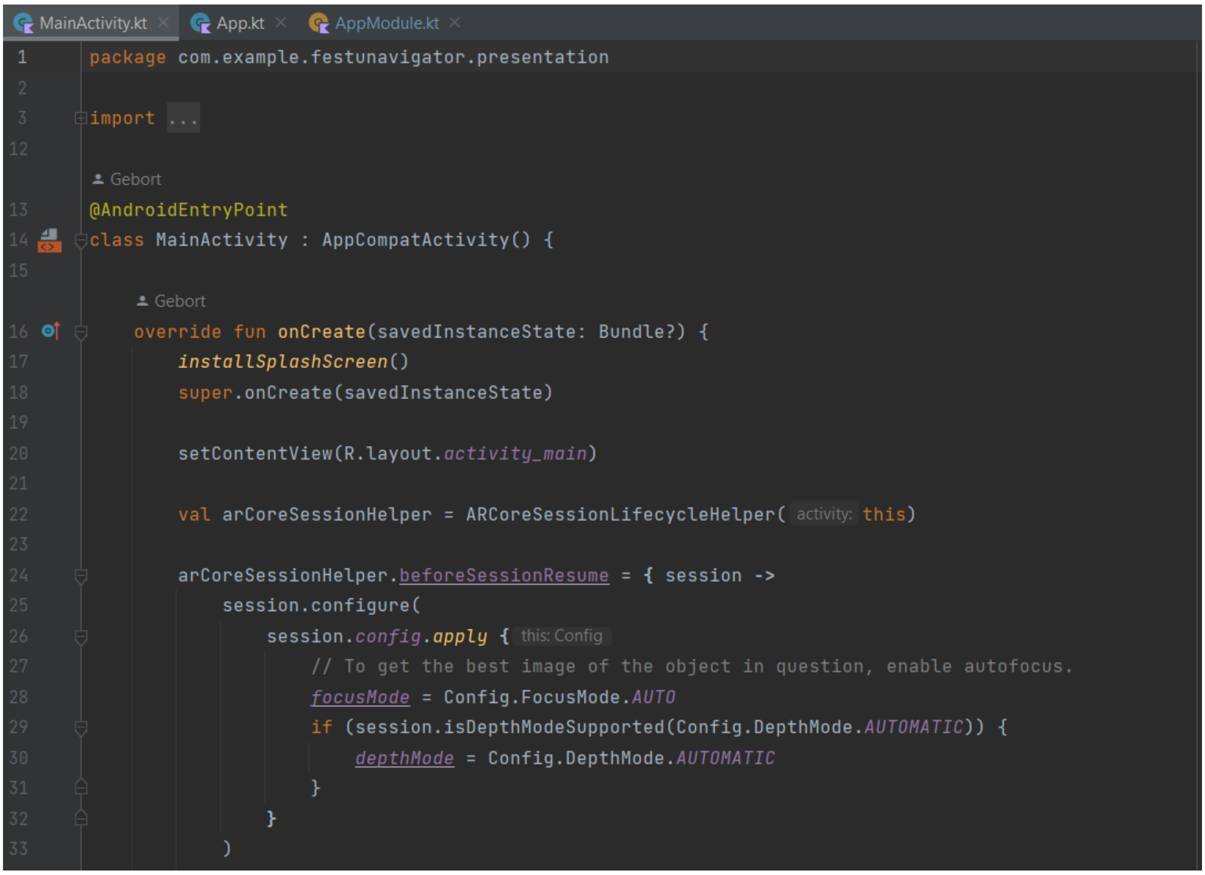
represents the order in which the whole augmented reality system works. Before setting up nodes on the college map to initiate the indoor navigation system, we first scan and determine the location of the indoor area, which in this case is the classroom number using the OCR Recognition functionality. An AR session is then created for Node Deployment, linking, and deletion to form a Node Grid. The user is asked to input their current position and final destination before the A* algorithm is used to calculate the fastest route between the two points. To enhance the precision of path-finding, the grid nodes are recursively scanned using the A* algorithm to provide the route to the user. To guide the user, the application generates graphical directing arrows that are superimposed on the user's mobile screen using augmented reality technology. And thus the system provides User localization and Navigation.

Chapter 4

Project Implementation

4.1 Code Snippets

Figure 4.1 demonstrates initializing the main activity of the Navigator app and sets up the ARCoreSessionLifecycleHelper to handle ARCore session events. The code defines the main activity and imports required classes and packages including AppCompatActivity and SplashScreen.



MainActivity.kt App.kt AppModule.kt

```
package com.example.festunavigator.presentation

import ...

Gebort
@AndroidEntryPoint
class MainActivity : AppCompatActivity() {

    Gebort
    override fun onCreate(savedInstanceState: Bundle?) {
        installSplashScreen()
        super.onCreate(savedInstanceState)

        setContentView(R.layout.activity_main)

        val arCoreSessionHelper = ARCoreSessionLifecycleHelper( activity: this)

        arCoreSessionHelper.beforeSessionResume = { session ->
            session.configure(
                session.config.apply { this: Config
                    // To get the best image of the object in question, enable autofocus.
                    focusMode = Config.FocusMode.AUTO
                    if (session.isDepthModeSupported(Config.DepthMode.AUTOMATIC)) {
                        depthMode = Config.DepthMode.AUTOMATIC
                    }
                }
            )
```

Figure 4.1: code1

Figure 4.2 represents setting up the ARCore camera configuration for an app and a CameraConfigFilter is created by passing in a session object. This filter is used to specify the camera's facing direction as BACK.

```
            val filter = CameraConfigFilter(session)
                .setFacingDirection(CameraConfig.FacingDirection.BACK)
            val configs = session.getSupportedCameraConfigs(filter)
            val sort = compareByDescending<CameraConfig> { it.imageSize.width }
                .thenByDescending { it.imageSize.height }
            session.cameraConfig = configs.sortedWith(sort)[0]
        }

        //lifecycle.addObserver(arCoreSessionHelper)

    }

}
```

Figure 4.2: code2

Figure 4.3 represents the code that sets up the App class for dependency injection using Hilt and defines constants to specify the mode of the application and check if it is in admin or user mode.

```
package com.example.festunavigator.data

import android.app.Application
import com.example.festunavigator.BuildConfig
import dagger.hilt.android.HiltAndroidApp

Gebort +1
@HiltAndroidApp
class App : Application() {

    Gebort +1
    companion object {
        const val ADMIN_MODE = "admin"
        const val USER_MODE = "user"
        const val mode = BuildConfig.FLAVOR
        const val isAdmin = mode == ADMIN_MODE
        const val isUser = mode == USER_MODE
    }
}
```

Figure 4.3: code3

Figure 4.4 represents the code which sets up the DomainModule to provide the necessary dependencies for the domain layer of the Navigator app, such as pathfinding algorithms, data repository implementations, and machine learning components.

```
package com.example.festunavigator.domain.di

import android.app.Application
import androidx.room.Room
import com.example.festunavigator.data.data_source.Database
import com.example.festunavigator.data.ml.classification.TextAnalyzer
import com.example.festunavigator.data.pathfinding.AStarImpl
import com.example.festunavigator.data.repository.GraphImpl
import com.example.festunavigator.data.repository.RecordsImpl
import com.example.festunavigator.domain.ml.ObjectDetector
import com.example.festunavigator.domain.pathfinding.Pathfinder
import com.example.festunavigator.domain.repository.GraphRepository
import com.example.festunavigator.domain.repository.RecordsRepository
import com.example.festunavigator.domain.tree.Tree
import com.example.festunavigator.domain.use_cases.*
import dagger.Module
import dagger.Provides
import dagger.hilt.InstallIn
import dagger.hilt.components.SingletonComponent
import javax.inject.Singleton
```

Figure 4.4: code4

Figure 4.5 represents the code that this sets up the database and provides a repository for accessing graph data. The code defines a Dagger Hilt module named "AppModule" that provides dependencies for the application.

```
Gebort *
@Module
@InstallIn(SingletonComponent::class)
object AppModule {

    private const val DATABASE_NAME = "nodes"
    private const val DATABASE_DIR = "database/nodes.db"

    Gebort *
    @Provides
    @Singleton
    fun provideDatabase(app: Application): Database {
        return Room.databaseBuilder(app, Database::class.java, DATABASE_NAME)
            .allowMainThreadQueries()
            .addMigrations()
            .build()
    }

    Gebort
    @Provides
    @Singleton
    fun provideGraphRepository(database: Database): GraphRepository {
        return GraphImpl(database)
    }
```

Figure 4.5: code5

Figure 4.6 represents the code that shows the definition of four Dagger Hilt-provided methods with the @Provides annotation, which are used to provide dependencies for the app.

```kotlin
@Provides
@Singleton
fun providePathfinder(smoothPath: SmoothPath): Pathfinder {
    return AStarImpl(smoothPath)
}

Gebort
@Provides
@Singleton
fun provideFindWay(pathfinder: Pathfinder): FindWay {
    return FindWay(pathfinder)
}

Gebort
@Provides
@Singleton
fun provideObjectDetector(): ObjectDetector {
    return TextAnalyzer()
}

Gebort
@Provides
@Singleton
fun provideImageAnalyzer(objectDetector: ObjectDetector): AnalyzeImage {
    return AnalyzeImage(objectDetector)
}
```

Figure 4.6: code6

Chapter 5

Testing

5.1 Unit Testing

Unit testing is the first level of testing, which is typically performed by the developers themselves. At the code level, it is the process of ensuring that individual components of the software are functional and work as intended. Unit testing can be done manually, however, automating the process will reduce delivery times and boost test coverage. Because flaws will be detected earlier in the testing process and will take less time to fix than if they were discovered later, debugging will be easier as a result of unit testing. It helped us understand the desired output of each module, which we had broken down into separate units. It helped us in classifying the cry categories on the basis of the algorithm that we used. The main objective of unit testing is to isolate written code to test and determine if it works as intended. Unit testing is an important step in the development process because if done correctly, it can help detect early flaws in code which may be more difficult to find in later testing stages.

5.1.1 Various Test Cases

Test Case No.	Test Condition	Test Steps / Procedure	Expected Results	Actual Results	Pass/Fail
1.	Splash Screen	Launching the Android Application	The system should show the expected Layout of Application	Layout of application is shown successfully	Pass
2.	Optical Character Recognition	Scanning the classroom number	The system should recognize the classroom number	Classroom number is successfully recognized	Pass
3.	Node Deployment	Deploying and linking the nodes	Nodes should be deployed and linked with each other	Nodes deployed and linked successfully	Pass

Figure 5.1: Test cases

Chapter 6

Result

The method described in this research paper uses optical character recognition (OCR) to identify classroom numbers and generates nodes for augmented reality (AR) navigation. The system takes pictures of classroom doors with a smartphone camera, which are then analyzed with OCR to identify the classroom number. An AR navigation system can then be used to direct students to their classrooms by using the recognized number to construct a node in the system.

Figure 6.1: Scanning the classroom number

The two primary parts of the proposed system are OCR and AR navigation. The collected image is sent into an OCR engine during the OCR stage in order to identify the classroom number. The identified classroom number is then utilized to generate a node in an AR navigation system, which can be used to navigate students to their classrooms, in the stage of AR navigation.

Figure 6.2: Node Deployed

The captured image is utilized to do OCR in order to identify the classroom number. OCR is a complex activity that requires a number of processes, including character recognition, image segmentation, and post-processing. The image of deployed Node is displayed in Figure 6.2.

An interface is provided to interact with the AR Session. The Interface provides various buttons such as PATH for Node Deployment, LINK for linking Nodes, DELETE for deleting nodes, and ENTRY to save the Node Grid displayed on the user screen to the Database. Below that, two Drop Down Menus are provided for entering the Starting and Final Destination for Directions Rendering and Navigation. Figure 6.3 displays an example of connected grid nodes.

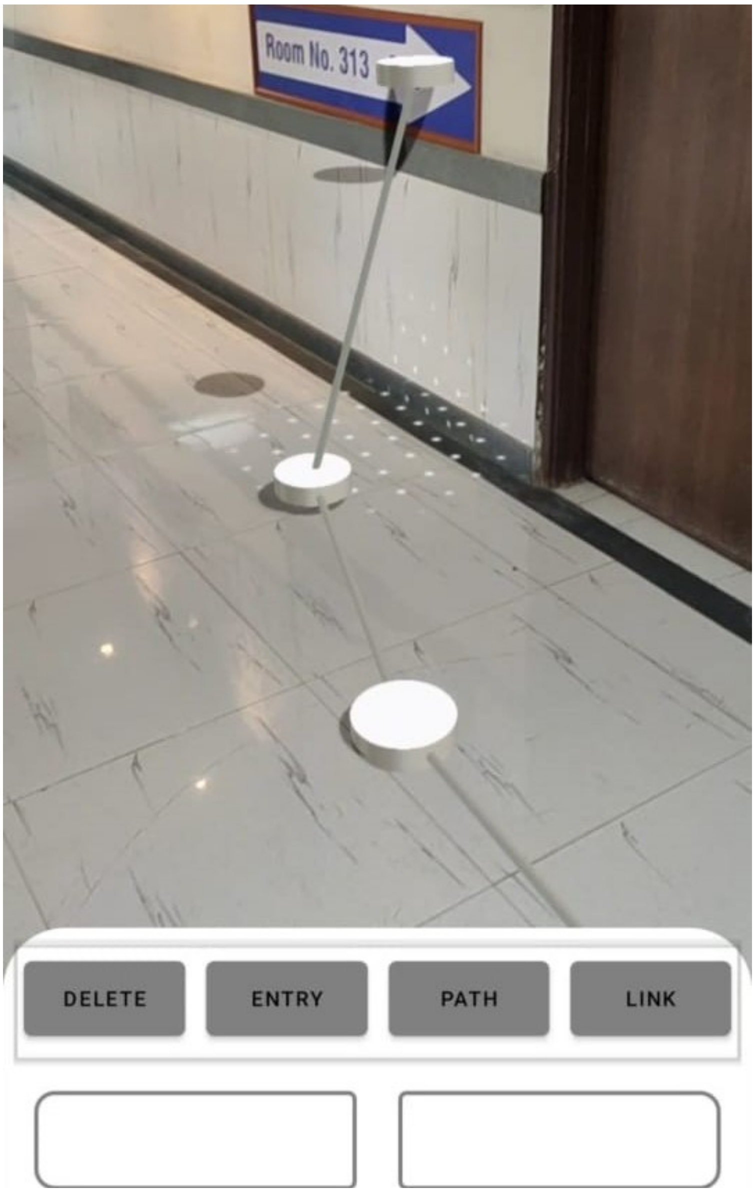


Figure 6.3: Node connected

Figure 6.4 displays an example of the rendered directional arrows superimposed on the user interface using the AR session for efficient Navigation across curved edges, turns, corners, etc. The arrows are rendered continuously and centered using the algorithm to avoid collision and ensure the safety of the end user.

Figure 6.4: Superimposed Arrows Showing directions

Figure 6.5: Superimposed Arrows Showing directions

Figure 6.5 represents the process of displaying a built route in 3D space using the Scene View AR library. The shortest route is determined using the A* algorithm, which is an efficient path-finding algorithm commonly used in navigation systems. Once the route is established, path smoothing is performed using Bezier curves, which help to create a smoother and more visually appealing path for the user. Overall, this approach allows for accurate and efficient navigation in augmented reality environments.

Chapter 7

Conclusions and Future Scope

The primary goal of this project is to give a simple and unhindered way to move inside enormous, complex buildings without the need for outside assistance. Prior to deploying nodes on the college map to start the entire indoor navigation, we first scan and localize the indoor area(in this example, the classroom number). It asks for the user's present position and the final destination location before using the A* algorithm to map the quickest route between the two spots. The accuracy of path finding was increased by recursively scanning the grid nodes with the A* algorithm. The application guides the user by using graphically created directing Arrows that are then superimposed on the user's mobile screen.

AR indoor navigation can be used in various different real-life scenarios. Indoor navigation systems can be very helpful in large, complex areas such as shopping malls, corporate offices, airports, hospitals, and universities. In such places, visitors can easily get lost or confused, which can lead to frustration and waste of time. Indoor navigation systems provide users with real-time information on their location, direction, and the shortest route to their destination. Indoor navigation systems can also be useful for planning big events such as exams, fests, and conferences. During such events, thousands of people gather in a confined space, and it can be challenging to navigate through the crowd. Indoor navigation systems can provide attendees with real-time information on the location of different stalls, events, and amenities. This can enhance the overall experience of the attendees and make the event more efficient. Indoor navigation systems can be used to engage visitors with relevant offers. If someone walks toward a Coffee Shop, that person can receive a notification with a relevant offer, such as a discount coupon. This can enhance the user experience and increase revenue for the business. Overall, indoor navigation systems can have a wide range of applications and benefits, from enhancing user experience to saving costs and making smarter decisions. As technology continues to advance, we can expect indoor navigation systems to become even more sophisticated and beneficial for a variety of industries.

Bibliography

[1] “Indoor Navigation system using visual position system with augmented reality”, 2018 International Conference on Automation and Computational Engineering (ICACE - 2018).

[2] Yanlei Gu, Woranipit Chidsin ,Igor Goncharenko “AR-based Navigation Using Hybrid Map”, 2021 IEEE 3rd Global Conference on LifeSciences.

[3] Low Chee Huey,Woan Ning Lim “Design of a mobile Augmented Reality – based indoor navigation system.”, 2020 4th International Symposium on Multidisciplinary Studies and Innovative Technologies (ISMSIT)

[4] Qi – wei Bao, Chris Papachristou, Frank Wolf “An Indoor Navigation and localization system.”, 2019 IEEE National Aerospace and Electronics Conference (NAECON)

[5] Manas Patil, Ajay Kadam, Kalpit Choudhari, “Feature-Based Indoor Navigation Using Augmented Reality”, 2019 9th International Conference on Cloud Computing, Data Science Engineering (Confluence)

[6] Austin Corotan ,Jianna Jian Zhang ,Irgen-Gioro , “An Indoor Navigation Robot Using Augmented Reality”, 2019 5th International Conference on Control, Automation and Robotic

[7] Austin Corotan ,Jianna Jian Zhang ,Irgen-Gioro , “An Indoor Navigation Robot Using Augmented Reality”, 2019 5th International Conference on Control, Automation and Robotic

[8] Ibrahim Alper Ko¸c, Tacha Serif, Sezer G¨oren , George Ghinea, “Indoor Mapping and Positioning Using Augmented Reality.” 2019 7th International Conference on Future-IOT

[9] L. Rosenblum, ”Virtual and augmented reality 2020,” in IEEE Computer Graphics and Applications, vol. 20, no. 1, pp. 38-39, Jan-Feb. 2000, DOI: 10.1109/38.814551

[10] ”New realities: a systematic literature review on virtual reality and augmented reality” by Yung, Ryan, Khoo-Lattimore, Catheryn, https://doi.org/10.1080/13683500.2017.1417359

[11] Y. Kuang and X. Bai, ”The Research of Virtual Reality Scene Modeling Based on Unity3D,” 2018 13th International Conference on Computer Science and Education (ICCSE), 2018,pp. 1-3, doi: 10.1109/ICCSE 2018.8468687

[12] S. Wang, Z. Mao, C. Zeng, H. Gong, S. Li and B. Chen, ”A new method of virtual reality based on Unity3D,” 2010 18th International Conference on Geoinformatics, 2010, pp.15,doi:10.1109/GEOINFORMATICS.2010.5567608

[13] S. L. Kim, H. J. Suk, J. H. Kang, J. M. Jung, T. H. Laine and J. Westlin, "Using Unity3D to facilitate mobile augmented reality game development, 2014 IEEE World Forum on Internet of Things (WF-IoT), 2014, pp. 21-26, doi: 10.1109/WF-IoT 2014.6803

[14] "Introducing the IEEE Virtual Reality 2020 Special Issue" in IEEE Transactions or Visualization and Computer Graphics, vol. 26, to 5, by K. Mueller and D. Bom, 77 v, MAY 2020, DOI:10.1109/TVCG 2020,2978971

[15] K. Mueller and D. Schmalstieg, "Introducing the IEEE Virtual Reality 2019 Special Issue," in IEEE Transactions on Visualization and Computer Graphics, vol. 25, no. 5, pp.iv-iv, May 2019, DOI: 10.1109/TVCG 2019,2902964.

Appendices

Appendix-I: Android Studio

- Download and Install Android Studio Electric Eel from https://developer.android.com/studio
- Git Clone the GitHub repository
- Import the cloned folder to Android Studio using VCS
- Setup Gradle JDK in build tools
- Gradle Building
- Set build variants to adminDebug or adminRelease
- Remove line - .createFromAsset(DATABASE DIR): in /domain/di/ModuleApp.kt
- Build the Project
- Enable USB/WiFi debugging in developer options in Mobile
- Connect the Mobile to the Desktop using USB/WiFi
- Run the Application

Publication

Paper entitled **“AR App for College Campus Navigation”** is presented at **“International Conference on Contemporary Challenges in Science and its Engineering Applications (IC3SEA 2023) on May 5th and 6th 2023”** by **”Sameer Sawant, Raj Shisode, Prathmesh Shrirao, Prof. Mandar Ganjapurkar, and Prof. Shweta Mahajan”.**