Report submitted to the SASTRA Deemed to be University as the requirement for the course

ECE300: MINI PROJECT

Submitted by

RD. MANOJ **Reg. No. 124006059**

HARIHARAN M V **Reg. No. 124006058**

April 2023

SCHOOL OF ELECTRICAL & ELECTRONICS ENGINEERING

THANJAVUR, TAMIL NADU, INDIA – 613 401

SCHOOL OF ELECTRICAL & ELECTRONICS ENGINEERING

THANJAVUR – 613 401

Bonafide Certificate

This is to certify that the thesis titled "**MOBILE APPICATION BASED ON AR**" submitted in partial fulfillment of the requirements for the award of the degree of B. Tech. Electronics & Instrumentation Engineering to the SASTRA Deemed to be University, is a bonafide record of the work done by **Mr. R.D. MANOJ** (124006059), **Mr. HARI HARAN** (124006058), during the final semester of the academic year 2022- 23, in the **School of Electrical & Electronics Engineering**, under my supervision. This thesis has not formed the basis for the award of any degree, diploma, associate-ship, fellowship orother similar title to any candidate of any University.

Signature of Project Supervisor :

Name with Affiliation **:** Dr. K. Ghousiya Begum, AP III, EIE

Date **:** 18-05-2023

Project *Vivavoc*e held on _________________

Examiner 1 **Examiner 2**

THINK MERIT | THINK TRANSPARENCY | THINK SASTRA

THANJAVUR | KUMBAKONAM | CHENNAI

SCHOOL OF ELECTRICAL & ELECTRONICS ENGINEERING

THANJAVUR – 613 401

Declaration

We declare that the thesis titled "**MOBILE APPICATION BASED ON AR**" submitted by us is an original work done by us under the guidance of **Dr. K. Ghousiya Begum, AP - III, School of Electrical and Electronics Engineering, SASTRA Deemed to be University** during the final semester of the academic year 2022-23, in the **School of Electrical and Electronics Engineering**. The work is original and wherever we have used materials from other sources, we have given due credit and cited them in the text of the thesis. This thesis has not formed the basis for the award of any degree, diploma, associate-ship, fellowship or other similar title to any candidate of any University.

Signature of the candidate(s) :

Name of the candidate(s) : R.D. MANOJ (124006059)

M.V. HARI HARAN (12400658)

Date: 18.05.23

Acknowledgements

First and foremost, we are thankful to God Almighty for giving us the potential and strength to work on this project fruitfully. We express our gratitude to **Dr. S. VAIDHYASUBRAMANIAM**, Vice- Chancellor, SASTRA Deemed to be University for providing us with an encouraging platform for the course of study.

We also thank **Dr. R. CHANDRAMOULI**, Registrar, SASTRA Deemed to be University, for providing us the opportunity to work in the SEEE, SASTRA Deemed to be University for the course of the work.

We render our solicit thanks to **Dr. K. THENMOZHI**, Dean, SEEE and **Dr. K. SRIDHAR**, Associate Dean, EEE, SASTRA Deemed to be University for introducing the major project in our curriculum.

We owe a debt of earnest gratitude towards our guide **Dr. K. GHOUSIYA BEGUM**, EIE for her continued support and guidance throughout the course of our work. The intriguing questions raised by our panel members **Dr. S. VENKATESH** and **Prof. V. VIJAYABASKAR** from the department of EIE helped us in shaping this project to its worth.

We also extend our sincere thanks to our parents for their continuous moral support and friends without whose thought-provoking suggestion and encouragement, along with our guide, this work would not have been what it is today.

Abstract

This project focuses on developing a mobile augmented reality (AR) application that integrates computer vision (CV) technology and can also control things in the Internet of Things (IoT). The goal of the project is to enhance the user's AR experience by providing advanced object recognition and classification capabilities. The application uses the camera on the mobile device to capture images of real-world objects and uses pre-trained ML algorithms to identify and classify the objects in real-time. Also, this application can control things virtually. The AR application provides an interactive experience for the users by overlaying virtual objects on top of the real-world objects. The users can interact with the virtual objects through virtual objects and UI, providing a new level of interactivity and engagement. Overall, this project demonstrates the potential of integrating CV and IoT technology into AR applications to create new and exciting user experiences.

Specific Contribution

- Dealt with software application and developed an android application.
- Contributed on hardware part, connecting the hardware to cloud.
- Worked an AR navigation part.

Specific Learnings

- Learned about VUFORIA engine, and lot about Computer Vision.
- Came to know the API's available to develop the application.

Technical limitations and Ethical challenges faced

- Limitation in free version of the software and some of the plug-ins to extend the features.
- Latency in calling the API's.
- Challenging C# modules to know the functions objects and methods available.

Signature of Guide **Student Reg. No: 124006059**

Name: Dr. K. Ghousiya Begum **Name: R.D. MANOJ**

Abstract

This project focuses on developing a mobile augmented reality (AR) application that integrates computer vision (CV) technology and can also control things in the Internet of Things (IoT). The goal of the project is to enhance the user's AR experience by providing advanced object recognition and classification capabilities. The application uses the camera on the mobile device to capture images of real-world objects and uses pre-trained ML algorithms to identify and classify the objects in real-time. Also, this application can control things virtually. The AR application provides an interactive experience for the users by overlaying virtual objects on top of the real-world objects. The users can interact with the virtual objects through virtual objects and UI, providing a new level of interactivity and engagement. Overall, this project demonstrates the potential of integrating CV and IoT technology into AR applications to create new and exciting user experiences.

Specific Contribution

- Contributed on hardware part, connecting the hardware to cloud.
- Worked an AR navigation part.

Specific Learnings

- Learned about VUFORIA engine, and lot about Computer Vision.
- Learned IOT part. On how the things is connected to the internet.
- Also, learned about the API's

Technical limitations and Ethical challenges faced

- Limitation in free version of the software and some of the plug-ins to extend the features.
- Latency in calling the API's.

Signature of Guide **Student Reg. No: 124006058**

Name: Dr. K. Ghousiya Begum **Name: HARI HARAN**

Table of Contents

List of Tables

Abbreviations

AR	Augmented Reality
UI	User Interface
CV	Computer Vision
GPS	Global Positioning System
MAR	Memory Address Register
IOT	Internet Of Things
IOS	iPhone Operating System
ARCore	Augmented reality development kit Android
AR Kit	Augmented reality development kit for IOS
3D	Three Dimensional
IDE	Integrated Development Environment
SDK	Software Development Kit

CHAPTER 1
INTRODUCTION

The potential of augmented reality (AR) technology to improve the user experience in a variety of industries, including gaming, education, healthcare, and retail, has led to its tremendous rise in popularity in recent years. For this project, we used Android Studio, the well-known augmented reality development platform Vuforia, as well as various IoT-based hardware, cloud to create an AR-based mobile application. By superimposing digital 3D objects over the actual scene that was photographed by the camera on the user's mobile device, the programmed intends to give the user an engaging and immersive experience. Along with developing a user-friendly UI, the project also included adding capabilities like virtual UI control, AR navigation, object recognition and IoT device integration. The design and development process for the AR-based mobile application, along with the difficulties encountered and the successes attained, are described in this report.

The following are the primary contributions in our work:

i. Providing an UI interface on mobile AR (MAR) application
ii. Bidirectional Interaction between MAR application and IOT hardware devices using Cloud API (Blynk) to control the application virtually and analysis data real- time. Enabling real-time data presentation and analyzing from the necessary synthetic data provided on this application.
iii. Developing GPS based navigation and integrating with MAR

Following is the arrangement of the paper's remaining sections. The second chapter elaborates the existing works related to classification. The third part explains the approach behind the proposed method in detail. The next segment presents the outcomes of this work, and the chapter 5 summarizes the results with relevant figures and metrics and the final chapter presents future work which can be used to extend the work.

CHAPTER 2

LITERATURE SURVEY

In recent years, the use of Augmented Reality (AR) technology has grown in popularity, and many applications have been created in a variety of industries. In this review of the literature, we look at the current research and advancements in mobile AR applications.

In a study on [1] AR-based mobile applications have become increasingly popular due to their ability to provide users with an immersive and interactive experience. In that study they evaluated the effectiveness of AR-based mobile applications in enhancing learning outcomes in medical education. The results showed that the use of AR-based mobile applications led to improved learning outcomes compared to traditional methods. Similarly, another study explored the use of AR-based mobile applications in promoting physical activity among children. The study found that the use of AR-based mobile applications increased children's motivation to engage in physical activity.

Another study in [2] Various AR development platforms are available to developers to create AR-based mobile applications. Vuforia, Unity, and ARfountation(for ANDROID users), ARKit(for IOS users) are among the popular platforms used for AR development. In compared the performance of Vuforia and Unity in developing AR-based mobile applications. The study found that both platforms performed well in terms of tracking and rendering, but Unity was more versatile in terms of developing more complex applications.

In another study [3] Integrating AR with the Internet of Things (IoT) has the potential to enhance the user experience and provide new opportunities for interaction. From the study explored the integration of AR and IoT in the context of smart homes. The study developed an AR-based mobile application that allowed users to control smart home devices through a visual interface. The results showed that the integration of AR and IoT improved the user experience and provided more intuitive control of smart home devices.

In another study [4] AR technology has been widely used in gaming, with many popular mobile games like (pokemo-go) incorporating AR features. In that study explored the use of AR in mobile gaming and its impact on user experience. The study found that the use of AR in mobile gaming led to increased engagement, immersion, and social interaction among players.

AR technology has been increasingly used in education to enhance learning outcomes and provide a more engaging learning experience. A study in[5]explored the use of AR-based mobile applications in teaching geometry to middle school students. The study found that the use of AR-based mobile applications led to improved understanding of geometric concepts and enhanced motivation among students.

These literature surveys can provide you with valuable insights into the various applications of AR-based mobile applications, and how they have been used to enhance user experience in different fields. You can use these surveys to support your research and provide context for your project.

CHAPTER 3
METHODOLOGY

3.1 Design and Planning:

Our first goal is to create an application that can interact with real-world elements that people can see with their natural eyes. Therefore, we created hardware that could be controlled over the internet, likely uploading the data to the cloud from which we could extract it. With the aid of APIs, we were also able to request data from the MAR application we created.

The second objective pertaining to the navigational component, nodes are placed at appropriate points on the map and the longitude and latitude data are obtained after which the xml format of the same (as shown in fig 3.1.1) is extracted to align the 3D models at its respective nodes.

```
ers > manoj > Downloads > data (org).xml
<ArGpsLocationData><Entry><id>0</id><lat>10.7296191422287</lat><lng>79.02322243526726</lng><altitude>0</altitude><altitudeMode>GroundRelative</
altitudeMode><name>roadmark1</name><meshId>road</meshId><movementSmoothing>0.05</movementSmoothing><maxNumberOfLocationUpdates>0</
maxNumberOfLocationUpdates><useMovingAverage>false</useMovingAverage><hideObjectUtilItIsPlaced>true</hideObjectUtilItIsPlaced></Entry><Entry><id>2</id><lat>10.
728944870158983</lat><lng>79.02324282807388</lng><altitude>0</altitude><altitudeMode>GroundRelative</altitudeMode><name>roadmark3</name><meshId>road</
meshId><movementSmoothing>0.05</movementSmoothing><maxNumberOfLocationUpdates>0</maxNumberOfLocationUpdates><useMovingAverage>false</
useMovingAverage><hideObjectUtilItIsPlaced>true</hideObjectUtilItIsPlaced></Entry><Entry><id>3</id><lat>10.72916677904407</lat><lng>79.02323974773145</lng><altitude>0</
altitude><altitudeMode>GroundRelative</altitudeMode><name>roadmarker3</name><meshId>road</meshId><movementSmoothing>0.05</
movementSmoothing><maxNumberOfLocationUpdates>0</maxNumberOfLocationUpdates><useMovingAverage>false</us
hideObjectUtilItIsPlaced></Entry><Entry><id>4</id><lat>10.728709775725733</lat><lng>79.02324600248329</
altitudeMode><name>roadmark4</name><meshId>road</meshId><movementSmoothing>0.05</movementSmoothing><max
maxNumberOfLocationUpdates><useMovingAverage>false</useMovingAverage><hideObjectUtilItIsPlaced>true</hideObjectUtilItIsPlaced></Entry><Entry><id>5</id><lat>10.
728519291862526</lat><lng>79.02324764926192</lng><altitude>0</altitude><altitudeMode>GroundRelative</altitudeMode><name>roadmark5</name><meshId>road</
meshId><movementSmoothing>0.05</movementSmoothing><maxNumberOfLocationUpdates>0</maxNumberOfLocationUpdates><useMovingAverage>false</
useMovingAverage><hideObjectUtilItIsPlaced>true</hideObjectUtilItIsPlaced></Entry><Entry><id>6</id><lat>10.728283339853249</lat><lng>79.02322777172424</lng><altitude>0</
altitude><altitudeMode>GroundRelative</altitudeMode><name>roadmark6</name><meshId>road</meshId><movementSmoothing>0.05</movementSmoothing><maxNumberOfLocationUpdates>0</
maxNumberOfLocationUpdates><useMovingAverage>false</useMovingAverage><hideObjectUtilItIsPlaced>true</hideObjectUtilItIsPlaced></Entry><Entry><id>7</id><lat>10.
728059881056666</lat><lng>79.0232516679215</lng><altitude>0</altitude><altitudeMode>GroundRelative</altitudeMode><name>roadmakrer</name><meshId>road</
meshId><movementSmoothing>0.05</movementSmoothing><maxNumberOfLocationUpdates>0</maxNumberOfLocationUpdates><useMovingAverage>false</
useMovingAverage><hideObjectUtilItIsPlaced>true</hideObjectUtilItIsPlaced></Entry><Entry><id>10</id><lat>10.729522682873153</lat><lng>79.02322909116526</
lng><altitude>0</altitude><altitudeMode>GroundRelative</altitudeMode><name>roadmarker</name><meshId>road</meshId><movementSmoothing>0.05</
movementSmoothing><maxNumberOfLocationUpdates>0</maxNumberOfLocationUpdates><useMovingAverage>false</useMovingAverage><hideObjectUtilItIsPlaced>true</
hideObjectUtilItIsPlaced></Entry><Entry><id>11</id><lat>10.729367054963816</lat><lng>79.02323402586796</lng><altitude>0</altitude><altitudeMode>GroundRelative</
altitudeMode><name>roadmarker</name><meshId>road</meshId><movementSmoothing>0.05</movementSmoothing><maxNumberOfLocationUpdates>0</
maxNumberOfLocationUpdates><useMovingAverage>false</useMovingAverage><hideObjectUtilItIsPlaced>true</hideObjectUtilItIsPlaced></Entry><Entry><id>12</id><lat>10.
729665508297856</lat><lng>79.0232187575275</lng><altitude>0</altitude><altitudeMode>GroundRelative</altitudeMode><name>vashishta</name><meshId>vashishta</
meshId><movementSmoothing>0.05</movementSmoothing><maxNumberOfLocationUpdates>0</maxNumberOfLocationUpdates><useMovingAverage>false</
useMovingAverage><hideObjectUtilItIsPlaced>true</hideObjectUtilItIsPlaced></Entry><Entry><id>13</id><lat>10.729562893662248</lat><lng>79.02127707537664</
lng><altitude>0</altitude><altitudeMode>GroundRelative</altitudeMode><name>av</name><meshId>av</meshId><movementSmoothing>0.05</
movementSmoothing><maxNumberOfLocationUpdates>0</maxNumberOfLocationUpdates><useMovingAverage>false</useMovingAverage><hideObjectUtilItIsPlaced>true</
hideObjectUtilItIsPlaced></Entry><Entry><id>14</id><lat>10.72918307493272</lat><lng>79.0213207271508</lng><altitude>0</altitude><altitudeMode>GroundRelative</
altitudeMode><name>av</name><meshId>av</meshId><movementSmoothing>0.05</movementSmoothing><maxNumberOfLocationUpdates>0</
maxNumberOfLocationUpdates><useMovingAverage>false</useMovingAverage><hideObjectUtilItIsPlaced>true</hideObjectUtilItIsPlaced></Entry><Entry><id>18</id><lat>10.
72887609166l462</lat><lng>79.02205460194801</lng><altitude>0</altitude><altitudeMode>GroundRelative</altitudeMode><name>mega</name><meshId>mega</
meshId><movementSmoothing>0.05</movementSmoothing><maxNumberOfLocationUpdates>0</maxNumberOfLocationUpdates><useMovingAverage>false</
useMovingAverage><hideObjectUtilItIsPlaced>true</hideObjectUtilItIsPlaced></Entry><Entry><id>19</id><lat>10.72740630380477</lat><lng>79.01919788204992</lng><altitude>0</
altitude><altitudeMode>GroundRelative</altitudeMode><name>vv</name><meshId>vv</meshId><movementSmoothing>0.05</movementSmoothing><maxNumberOfLocationUpdates>0</
maxNumberOfLocationUpdates><useMovingAverage>false</useMovingAverage><hideObjectUtilItIsPlaced>true</hideObjectUtilItIsPlaced></Entry><Entry><id>20</id><lat>10.
727453329283264</lat><lng>79.01855028772422</lng><altitude>0</altitude><altitudeMode>GroundRelative</altitudeMode><name>jvc</name><meshId>jvc</
meshId><movementSmoothing>0.05</movementSmoothing><maxNumberOfLocationUpdates>0</maxNumberOfLocationUpdates><useMovingAverage>false</
useMovingAverage><hideObjectUtilItIsPlaced>true</hideObjectUtilItIsPlaced></Entry><Entry><id>21</id><lat>10.727438867545885</lat><lng>79.01837629073384</
Tokenization is skipped for long lines for performance reasons. This can be configured via editor.maxTokenizationLineLength .
Ln 1, Col 1   Spaces: 4   UTF-8   CRLF   XML   Prettier
```


Fig. 3.1.1 longitude and latitude data in form of xml format

Fig 3.1.1 shows xml format the longitude and latitude data of the necessary node to be directed. these data are used in the application to find the points destination.

3.2. Development Process:

In order to create a mobile application, we must either select an application development software in order to develop an application. We require Android Studio for Android and Xcode IDEs for iOS. Eventually, we require the following software's and add-ons.

- UNITY SOFTWARE: Creating the UI for the application
- VUFORIA ENGINE: To integrate AR with mobile application MAR
- VISUAL STUDIO: For scripting
- BLYNKS CLOUD: Data logs and communication between IOT devices and MAR
- Arduino IDE : Scripting for Blynk edge end

These are software's we worked with in application development process.

And demonstrating the AR utilities in IOT enabled devices, we develop an hardware which has temperature and humidity sensor, and some actuators like led, fan, motor. Here we used.

- ESP8266 (WIFI module)
- Fans, LED, Relay, Motor

The data is acquired and sent to the cloud, as well as problems obtaining the longitude and latitude data, when it came to the coding portion of the project. We also had to provide logic for how the user interfaces should operate and behave on user's input. Some of the necessary plugins have premium versions only available. Therefore, we had to come up with a different strategy to overcome the difficulties.

BLOCK DIAGRAM

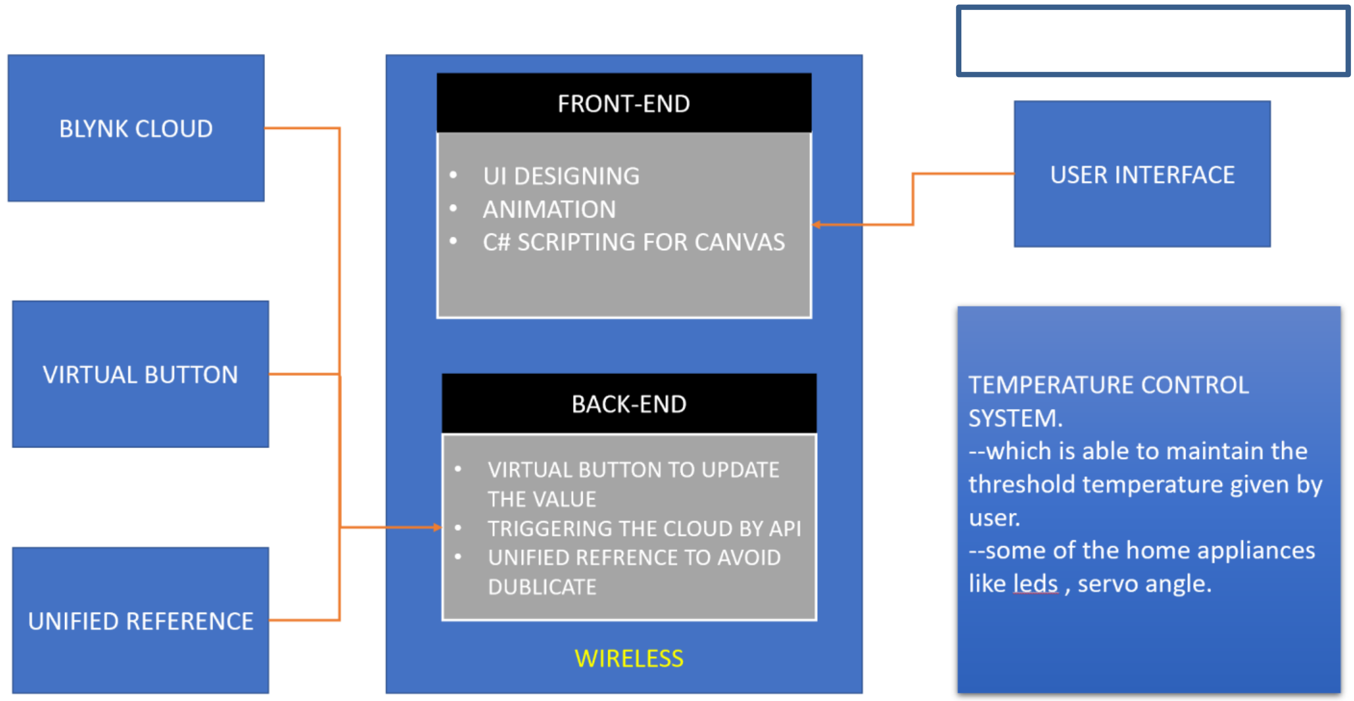


Fig 3.2: Block diagram on how the application works.

3.3. AR Development Platform:

Unity is a cross-platform game engine and development environment used to create 2D and 3D interactive applications, including games, architectural visualizations, and simulations. Unity allows developers to create applications for multiple platforms, including desktops, mobile devices, and virtual reality and augmented reality (VR/AR) devices. Unity provides a range of tools and features for developing interactive applications, including a visual editor, scripting tools, physics engine, animation tools, and a wide range of assets and resources. Unity also supports the development of AR and VR applications, with built-in support for many AR/VR platforms, such as Vuforia, AR Core, and Oculus. Unity is widely used by developers and companies of all sizes, from indie game developers to large studios, and is considered one of the most popular and versatile game engines available. Its popularity is due in part to its ease of use and flexibility, making it accessible to developers of varying skill levels and experience. Additionally, Unity has a large and active community of developers and users who provide support, resources, and assets to the community. We require separate add-on of developing an AR application. A software development kit for augmented reality apps called Vuforia is available for mobile devices. It recognizes and tracks planar pictures as shown in figure 3.3.1 and 3D objects in real time using computer vision technologies. When virtual things are observed through the camera of a mobile device, such as 3D models and other media, this image registration functionality allows developers to position and orient them in reference to actual items. The virtual object then continuously monitors the image's location and orientation to ensure that the target and viewer's perspectives are same. Thus, it seems as though the virtual item is a component of the actual picture.

3.4 UI DESIGNING:

For designing the UI, we used the following method. First, we create a plan as a layout before starting to design the UI, plan out the layout of the screens and the navigation between them. This will ensure that the UI is intuitive and easy to use. Use the Canvas Component Unity’s Canvas component is used to define the user interface of an application. Use the Canvas component to position and layout UI elements such as buttons, images, text, and input fields. Choose Appropriate Font and Color Choose a font that is legible and appropriate for the application's style and tone. Use colors that are consistent with the application's branding and theme. Use a grid to layout the UI elements to ensure that they are aligned and spaced consistently. This will improve the overall look and feel of the UI. Create custom assets such as buttons and icons that match the application's style and branding. This will give the application a unique and professional look. Add animation to UI elements to make the application more engaging and interactive. For example, add animations to buttons to make them appear to "pop" when clicked. Test the UI with actual users to ensure that it is intuitive and easy to use. Make any necessary changes based on user feedback. By following steps, we could able to create a user interface that is both visually appealing and easy to use.

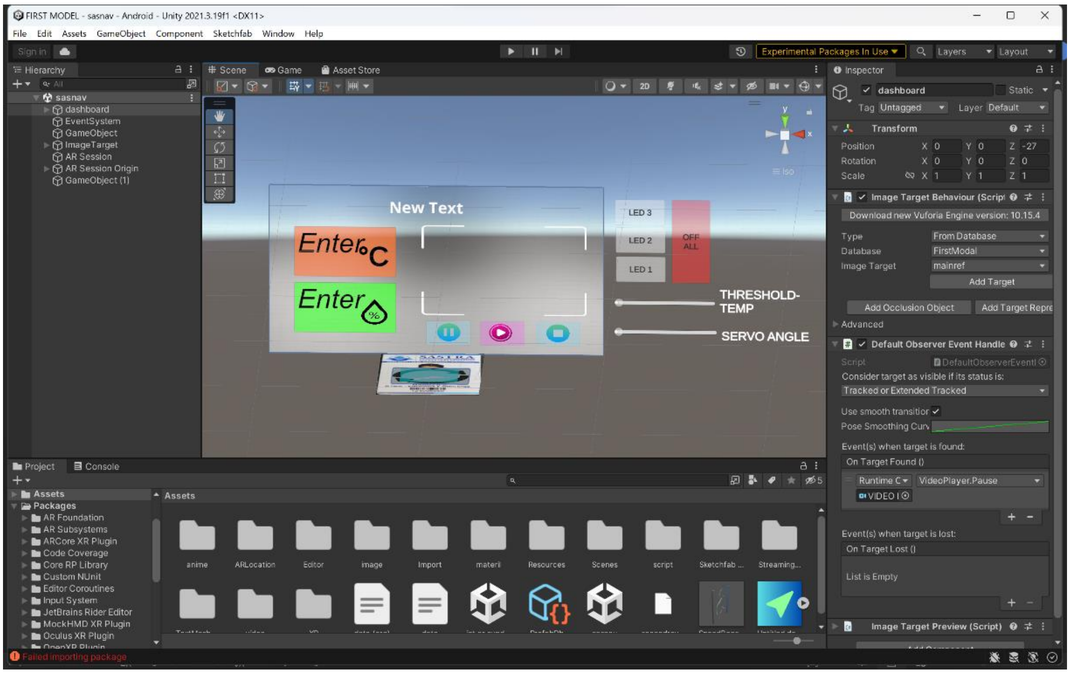


Fig 3.4.1: UI we designed for controlling the IOT device.

Fig 3.4.2: we can have multiple image targets in a scene.

In Fig 3.4.1: this the UI we designed to turn the led on or off and sliders to set the temperature threshold and servo motor angle. place holder to store the data which we request from the web and display it there. and we have a video player in middle where the demo of our hardware process is explained in there, in fig 3.4.2: it's another image target to show case that we can use multiple image target in a single scene environment.

3.5 Develop the AR Components:

We created an image reference that the AR application will use to track. This can be any image, such as a logo or a pattern. The image reference should be unique and easily recognizable by the AR software. Next, the AR software will use image recognition algorithms to detect the image reference in the real world. Once the image reference is detected, the software will use it as a base for augmenting the scene with virtual objects. The AR software will then overlay 3D models or other virtual objects onto the real-world scene. These objects can be used to provide additional information, navigation, or interactive elements. In our case, we are using 3D models for navigation. This means that when the user points their device at the image reference, the AR software will display a 3D model on top of the longitude and latitude point which we feed through the script. This model can be used to guide the user through a physical space, providing information and directions. Overall, image reference tracking is a powerful technique for creating AR applications that interact with real-world objects. By using 3D models for navigation, you can create immersive and interactive experiences that guide users through physical spaces. Just be sure to choose an image reference that is easily recognizable and has enough visual features for reliable tracking.

Fig3.5.1: image target nodes.

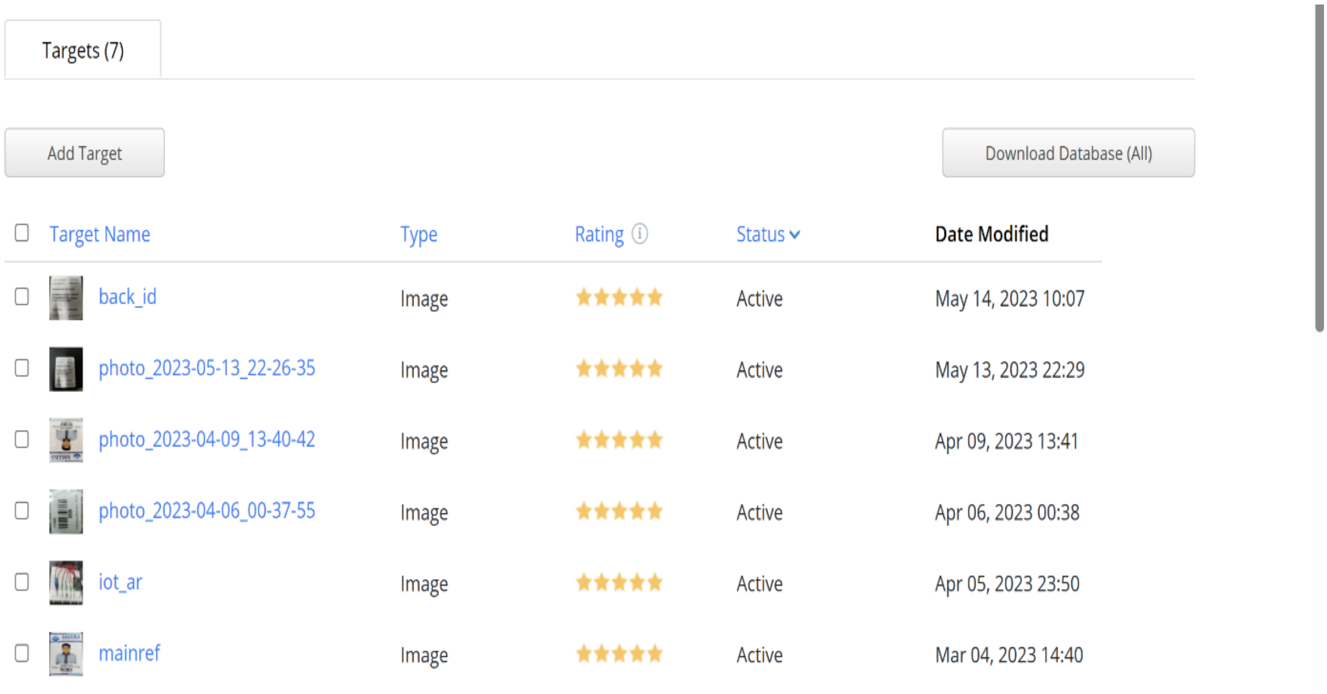


Fig 3.5.2: target database manager.

In the Fig 3.5.1 is the image reference which is fed to the Vuforia's AR camera known to be image target where the camera from the mobile collects the frame data and compare with the nodes of the target. If the target of the nodes match with the camera frames the target confidence increases. Once the target is identified the tracking process starts and its corresponding confidence (computer vision) will increase to a certain level such that it will be tracked to the image orientation and the user interface move along with the image in our case, (we use ID card as image reference). As, Image target confidence level increases proportionally tracking confidence level also increases. Fig 3.5.2: It'sa data base where we can edit the image target and manage our target nodes.

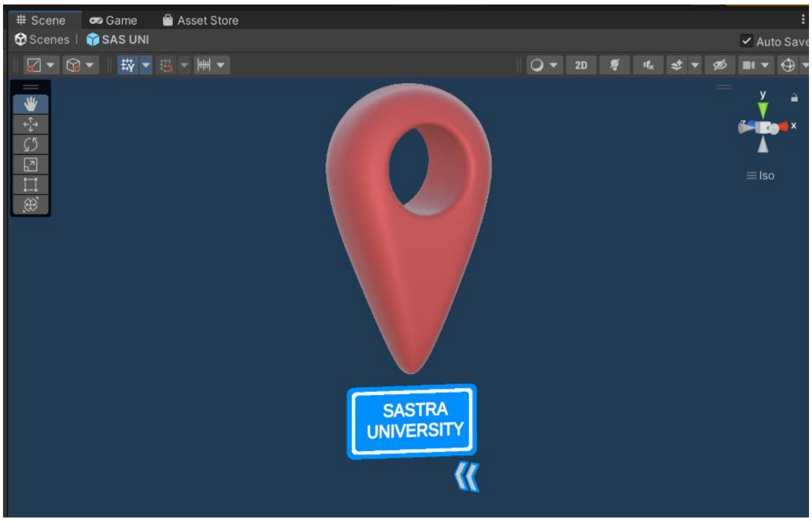


Fig 3.5.3: it’s the 3D model for land mark.

In Fig 3.5.3: we have an 3D model which will align itself in the longitude and latitude which we feed on the script from xml data sheet. One of the major factors on developing a mobile application is to make the application Optimize the performance of the AR components to ensure that they run smoothly on the devices at that user use. This may include reducing the complexity of the models or using more efficient tracking algorithms. Testing the AR components thoroughly and debug any issues that arise. This may include testing the accuracy of the tracking, the quality of the 3D models, and the performance of the AR components.

3.6 Integrating IoT-based Devices to Application:

For integrating IOT with AR we set up the hardware components. This includes connecting the ESP8266 WiFi module to your microcontroller and configuring it for use with Blynk cloud. Next, we integrate Blynk cloud into your AR application we used Blynk Rest APIs to achieve it. Blynk is a mobile app platform that allows you to control your hardware components remotely. You can use Blynk to create custom user interfaces and control the ESP8266 module from within your AR application. After the hardware components are set up and integrated with Blynk, you can use them to collect and process data from the real world. For example, we used temp and hum sensors connected to the ESP8266 module to monitor environmental conditions and feed that data into your AR application. Finally, we could able to collect data from my hardware components to create AR visualizations within our application. For example, we used temperature sensors to visualize temperature changes within a room, or use motion sensors to create interactive AR objects that respond to movement. Test and debug, hardware components to ensure they are functioning correctly. This may involve monitoring data streams in real-time, testing the

performance of your AR visualizations, and ensuring that the hardware is responding as expected.
We scripted an logic how to hardware should react with the received data from the sensor , in our the fan with be triggered when the temperature reached the threshold value which is controlled in the UI interface discussed in the UI designing session.

3.7 AR navigation in mobile:

Our AR navigation system uses longitude and latitude data to provide accurate and reliable guidance to users in physical spaces. The system features a 3D model of the space, augmented with AR elements such as directions and points of interest. This allows users to easily navigate the space and find their way to their destination. We have created an AR campus navigation where we get data from map as shown is the Fig 3.7.1.

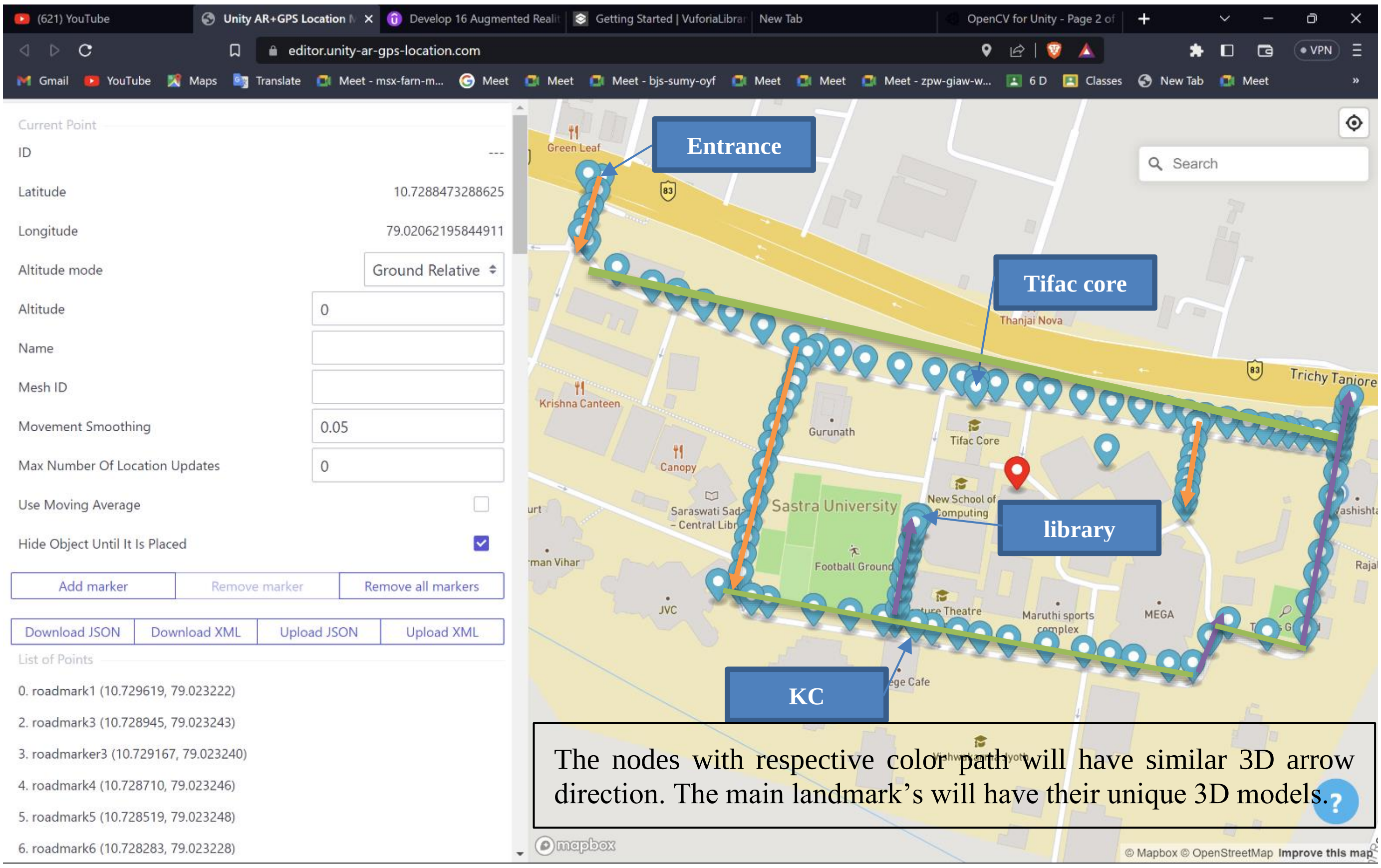
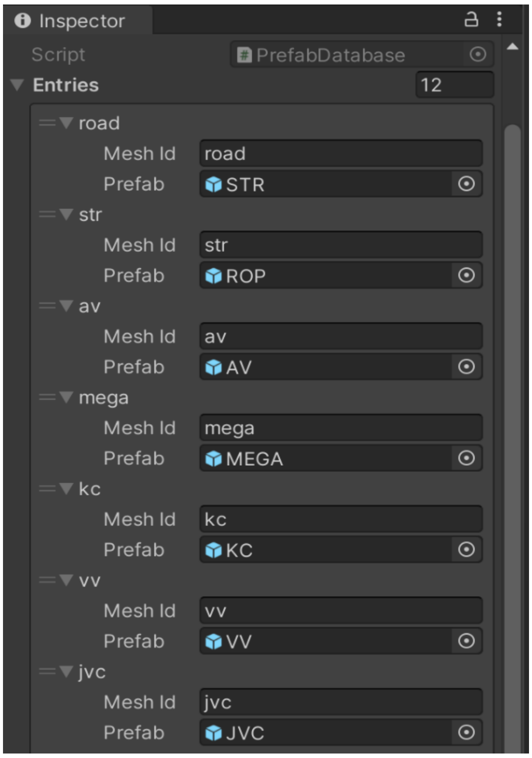


Fig 3.7.1: this is the point we used for the navigation, aligned as per the campus roadmap. This is the list of points and its longitude and latitude values



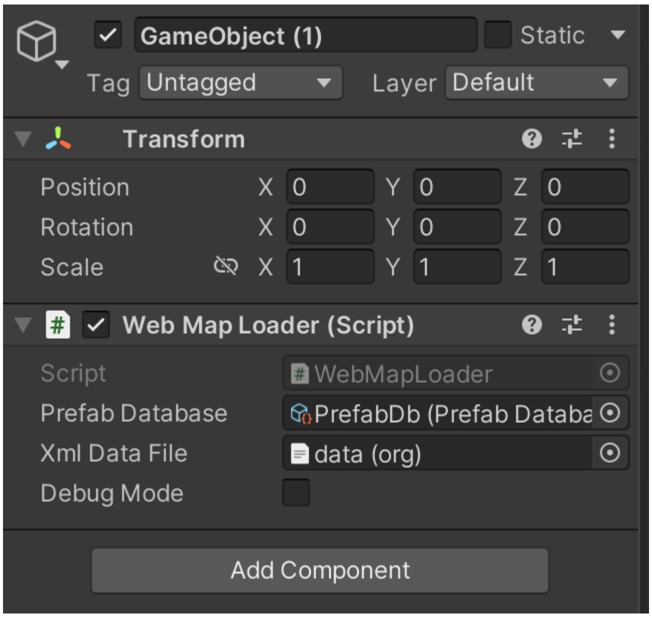


Fig 3.7.2: the XML data is feed to the C# script Web Map Loader.

Fig3.7.3: 3D model for each mesh ID.

LIST OF NODES AND ITS LONGITUDE AND LATITUDE VALUES

1.road (10.729619, 79.023222)
2.road (10.728945, 79.023243)
3.road (10.729167, 79.023240)
4.road (10.728710, 79.023246)
5.road (10.728519, 79.023248)
6.road (10.728283, 79.023228)
7.road (10.728060, 79.023252)
8.road (10.728762, 79.023265)
9.road (10.728064, 79.023476)
10. road (10.729523, 79.023229)
11. road (10.729367, 79.023234)
12. vashishta (10.729666, 79.023219)
13. av (10.729563, 79.021277)
14. av (10.729183, 79.021321)
15. mega (10.728876, 79.022055)
16. vv (10.727406, 79.019198)
17. jvc (10.727453, 79.018550)
18. jvc (10.727439, 79.018376)
19. lib (10.728305, 79.019874)
20. lib (10.728308, 79.019912)
21. lib (10.728306, 79.019895)
22. road (90) (10.729751, 79.023221)
23. road (90) (10.729815, 79.023214)
24. road (90) (10.729886, 79.023211)
25. road (90) (10.729665, 79.017762)
26. road (90) (10.729703, 79.017524)
27. road (90) (10.729728, 79.017314)
28. road (90) (10.729960, 79.016707)
29. road (90) (10.729446, 79.018704)
30. road (90) (10.729299, 79.018655)
31. road (90) (10.729182, 79.018646)
32. road (90) (10.729057, 79.018621)
33. road (90) (10.728924, 79.018587)
34. road (90) (10.728790, 79.018568)
35. road (90) (10.728644, 79.018559)
36. road (90) (10.728472, 79.018552)
37. road (90) (10.728318, 79.018552)
38. road (90) (10.728145, 79.018541)
39. road (90) (10.727964, 79.018530)
40. road (90) (10.727839, 79.018528)
41. road (90) (10.727705, 79.018535)
42. road (90) (10.727553, 79.018546)
43. road (90) (10.727385, 79.018811)
44. road (90) (10.727383, 79.018675)
45. road (90) (10.727430, 79.019502)
46. road (10.727461, 79.020415)
47. road (10.727470, 79.020602)
48. road (10.727482, 79.020841)
49. road (10.727506, 79.021165)
50. road (10.727530, 79.021473)
51. road (10.727534, 79.021692)
52. road (10.727546, 79.021896)
53. road (10.727562, 79.022195)
54. road (10.727599, 79.022392)
55. road (10.728007, 79.022605)
56. road (10.727981, 79.022942)
57. sv (10.730499, 79.016708)
58. sv (10.730507, 79.016592)
59. road (-90) (10.730383, 79.016697)
60. road (-90) (10.730246, 79.016672)
61. road (-90) (10.730128, 79.016657)
62. road (-90) (10.730022, 79.016629)
63. road(-90) (10.727527, 79.019885)
64. road (-90) (10.727639, 79.019884)
65. road (-90) (10.727773, 79.019877)
66. road (-90) (10.727880, 79.019879)
67. road (90) (10.727983, 79.019884)
68. road (180) (10.728094, 79.019881)
69. road (180) (10.728192, 79.019882)
70. road (180) (10.728254, 79.019875)
71. road (180) (10.727844, 79.022487)
72. tc (10.729511, 79.020138)
73. tc (10.729446, 79.020138)
74. road (10.729531, 79.022018)
75. road (180) (10.729393, 79.022024)
76. road (180) (10.729264, 79.022033)
77. road (180) (10.729155, 79.022034)
78. road (180) (10.729055, 79.022039)
79.road (180) (10.728960,79.022048)

** all road tag share common 3D models but according to the direction of part we need to rotate its orientation to make a meaningful path flow.**

CHAPTER 4
RESULTS AND DISCUSSION

At first, we manage our packages according so that it could able to run on any android devices. In the unity we specified necessary building setting to run the application without any error. Installing necessary develop kit, and with USB debugging we installed directly to the mobile. The benefit of using USB debugging it automatically install the required SDK to mobile directly without losing any SDK modules. With this we could export the application without any bugs.

So, in this chapter we will discuss the test attempts and challenges and way we rectified the bugs.
The outcome of the project.

4.1. PHASE 1 - ATTEMPT IN MAR APPLICATION:

In our first application we managed to display and vertical interface which show the information and contact detail, the UI in the Fig 4.1.1 has a video player with play, stop, pause button which the video and responsive. Two social media Buttons will director to the URL which I scripted to direct. And a simple ON/OFF button to turn the LED on and off respective to the button triggered.
We faced some difficulty in scripting an proper logical code to interfacing the hardware and mobile application.

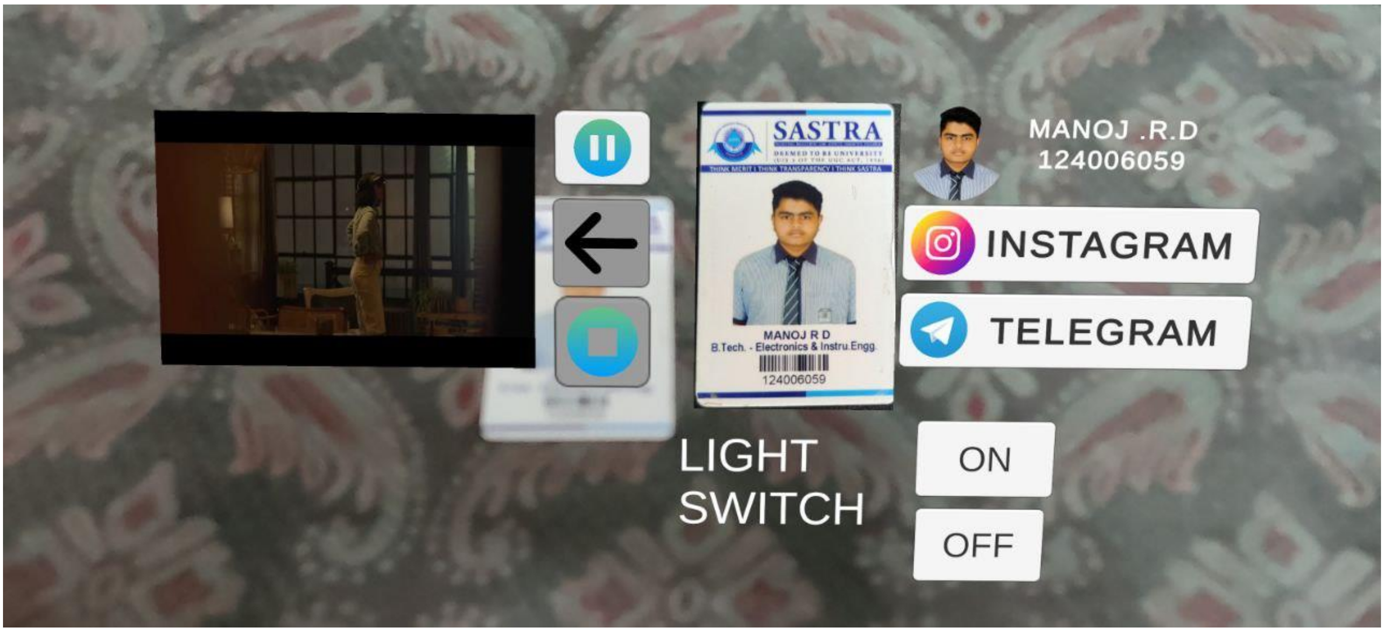
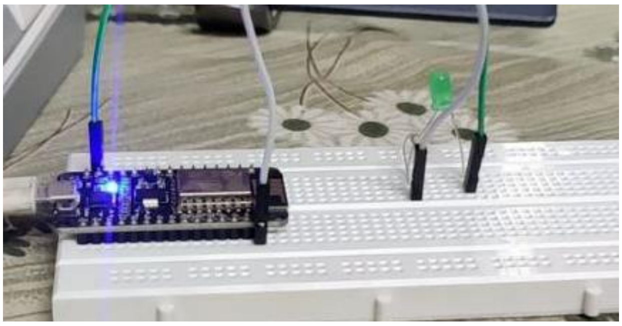


Fig 4.1.1: attempt on first full functioning MAR.



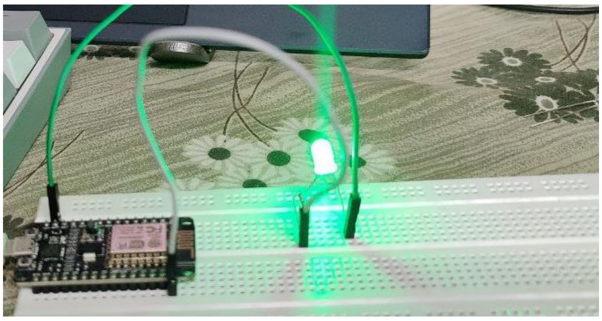


Fig 4.1.2 LED OFF when no button pressed.   Fig 4.1.2 LED on when the UI ON button pressed.

In the phase on we able to successfully interface our application with the hardware and able to make the UI responsive.

4.2 PHASE -2 IN ATTEMPT TO GET DATA FROM THE SENSOR:

Our next attempt was to get data from sensor and write text in UI interface. It was bit challenging because, earlier we will set the data through API it will send the data to the device no matter whether the device is online or not. But in this case, we need to read the data from the sensor and application should no crash if it could not fetch the data from the internet. After lot of research and attempts to add the sensor data we could able to add the data to UI with the help of will optimize C# script to run the application bug free.

Fig 4.2.1: showing sensor data in the UI.

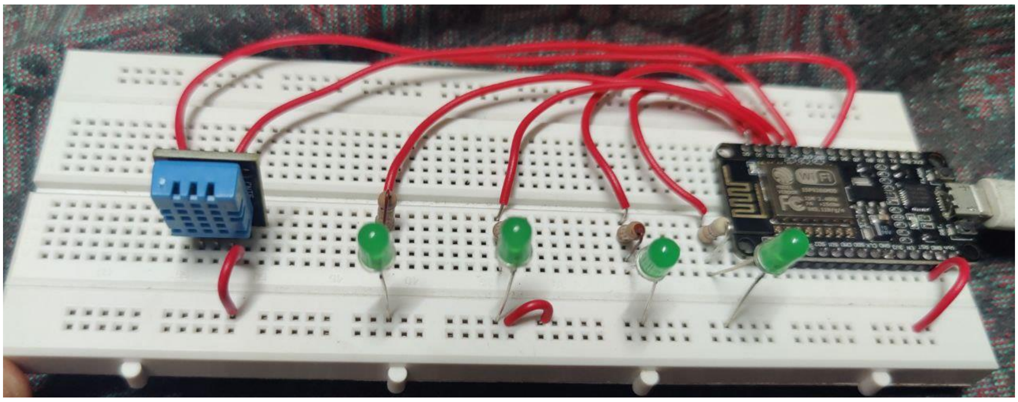


Fig 4.2.2: second hardware build with series of light and sensor.

4.3 phase -3 attempt on adding virtual button:

AR's most interesting thing it able to trigger something which it sees in the real world. In our case we included a virtual button which we placed over the target whenever the virtual button is pressed in real world it triggers something. In the phase -3 of the project, we improved the User Interface and scripting we did much optimization and light file size we reduce CPU load in the mobile. So that low-end mobile devices also could able to run the application without any struggles. We organized our Virtual PINS to look neat and presentable as shown in the fig4.3.1. here also we faced many difficulties in the codding part, we need to specify the virtual buttons behavior. And getting the virtual buttons behaviors was challenging. with lot of studies, we could able to understand how the game object in the unity works.

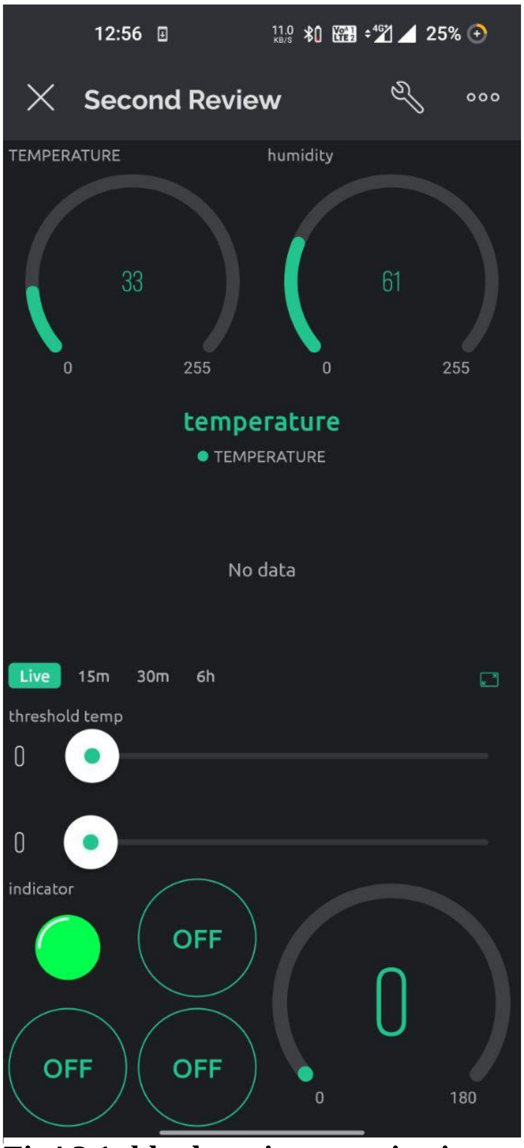


Fig4.3.1: blynks vpins organization

In Fig 4.3.2: the refresh symbol above the id card in the is the virtual button which trigger on pressing the button over the it card.

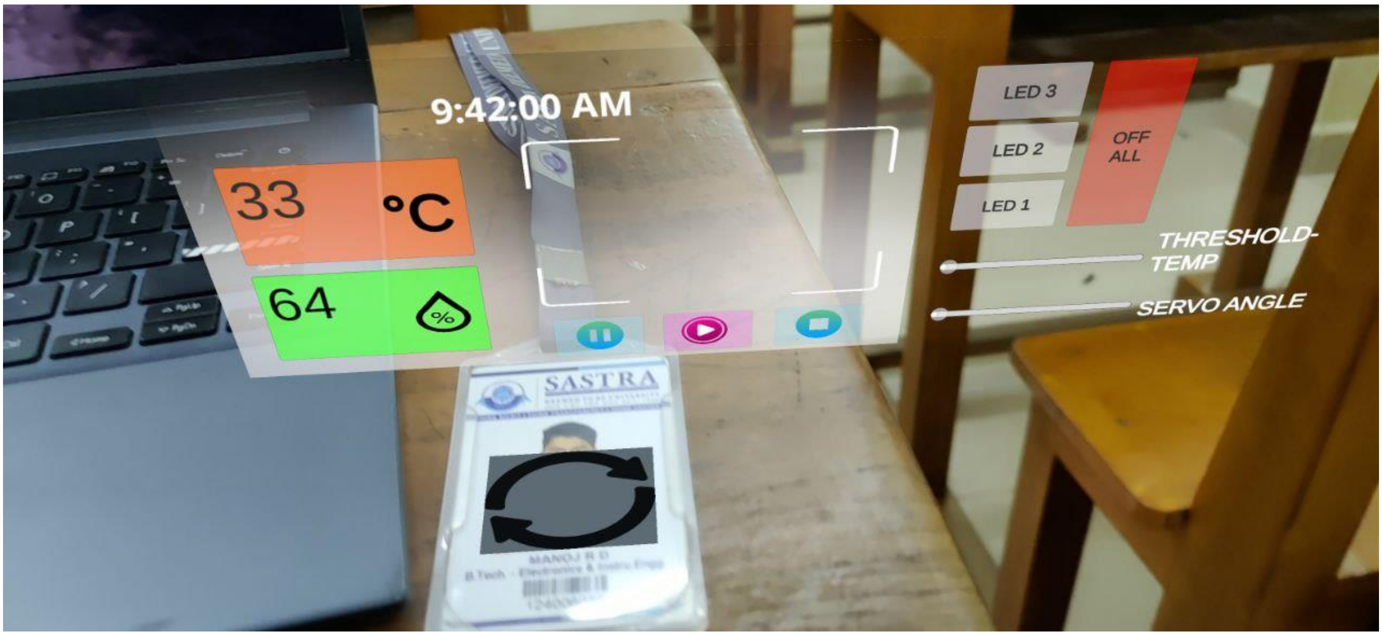


Fig 4.3.2: phase 3 version of the application.

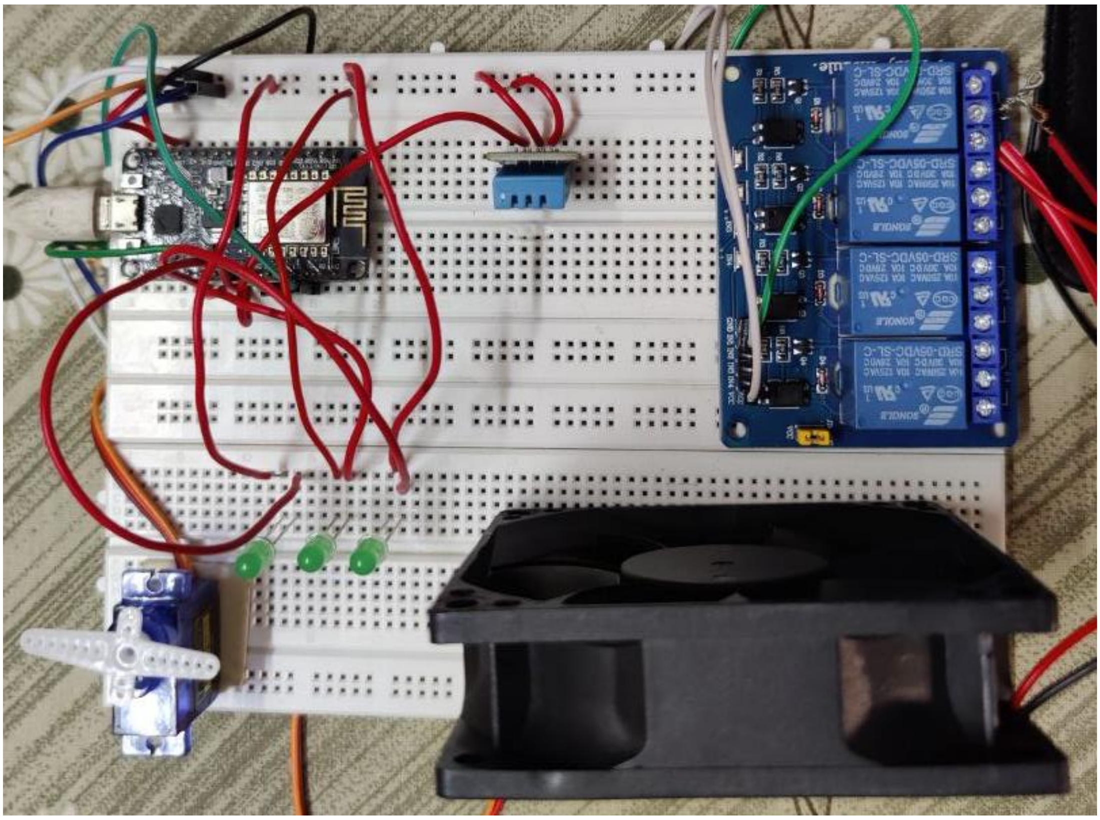


Fig 4.3.3: This the hardware setup for IOT-AR demonstration.

In Fig 4.3.3: This is the hardware setup which has ESP8266 WIFI module for send the sensor data to the Blynk Cloud from the BlynkEdgent , which is coded from the Arduino IDE.

In phase-3 we constructed a practical temperature control and home appliance enabled hardware and UI to control the hardware by the user. All the data in this phase in real time no data is synthetic data. The time clock which it refreshes for each and every second and the sensor data refreshed ever 5 mins and can update the data whenever the virtual button is pressed.

4.4 phase-4 AR navigation:

AR navigation is one of the main features of AR which virtually navigate to a certain destination, with out anyone's help. So, in order to navigate, there are many ways to navigate some will be suitable for indoor navigation some will be suitable for out door navigation. Our motive was to develop a AR navigation campus tour. So, we choose to use GPS based AR navigation which is suitable for outdoor navigation. if in case of indoor navigation, we can use 3D rendering the whole house and design the interface. In order to make application work on GPS device should have access to GPS and internet use the application effectively.

Here are the results of AR navigation,

Fig 4.4.1: result of the AR navigation at location 1.

Fig 4.42: result of AR navigation at location 2.

Conclusion

Mark-4 of the MAR application we could able to make responsive UI and which is also able to interact with the IOT enabled devices and bult-in AR campus navigation. In conclusion, we have developed a mobile AR application that leverages IoT technology and AR navigation to provide users with a unique and immersive experience. Our application allows users to navigate through physical spaces using AR markers and a 3D model, while also leveraging IoT sensors to provide real-time data about their surroundings. We used Unity software to develop the AR components of the application, including image tracking and 3D modeling, and integrated IoT-based hardware components such as the ESP8266 WIFI module and Blynk module to provide real-time data. Through testing and debugging, we could make sure that our application is highly intuitive and user-friendly, providing accurate and reliable navigation guidance. Users were able to easily orient themselves in physical spaces and navigate to their desired locations, while also receiving real-time data about their surroundings. In terms of future work, we see potential for further development of our application, including integration with additional CV technology and functionality, as well as expanding to new markets and use cases.

Overall, our project has demonstrated the potential of mobile AR applications that combine IoT and AR navigation technology, providing a platform for future innovation and development in this exciting field.

Signature of the Guide: Student Reg. No: 1224006059

Name: Dr. K. Ghousiya Begum Name: MANOJ R. D

Conclusion

Our application uses IoT sensors to offer real-time data about the environment while allowing users to navigate across physical places using AR markers and a 3D model. To create the augmented reality (AR) application, which included image tracking and 3D modelling, we used Unity gaming engine for this project. To give real-time data, we also incorporated IoT-based hardware elements such as the ESP8266 WIFI module and Blynk module. We could ensure that our programme is very optimized and user-friendly and intuitive through testing and debugging, offering precise and accurate navigation advice. In addition to collecting real-time information about their surroundings, users were able to travel to their preferred places and effortlessly orient themselves in physical areas. We see possibilities for furthering the development of our application in terms of future effort.

Signature of the Guide:

Name: Dr. K. Ghousiya Begum

Student Reg. No: 124006058

Name: HARI HARAN

FUTURE WORK

the potential applications for AR are vast and varied, with opportunities for innovation and development across a range of industries and domains. By continuing to explore and develop this technology, we can unlock its full potential and revolutionize the way we interact with the world around us. We can use then in any field in medical, education, military, industry, manufacturing, in my future work on developing this application by integrating AR with OpenCV. By integrating OpenCV with your AR navigation functionality, you can potentially add features such as object recognition, gesture recognition, and facial recognition, among others. Object recognition can be used to recognize physical objects in the environment and provide additional information or functionality related to those objects, while gesture recognition can be used to detect hand movements and gestures, allowing users to interact with the AR environment in new ways. Facial recognition can be used to recognize individual users and provide personalized experiences and information. Integrating OpenCV with AR technology requires advanced programming skills and a deep understanding of both technologies. However, with careful planning and development, it can lead to a more robust and advanced AR application that provides users with even more engaging and immersive experiences.

Overall, continuing to develop and refine your AR application with OpenCV integration is a promising area of future work that has the potential to push the boundaries of what is possible with AR technology.

REFERENCES

[1]. Cheng et al., 2020 Q. Cheng, S. Zhang, S. Bo, D. Chen, H. Zhang Augmented reality dynamic image recognition technology based on deep learning algorithm

[2] Tim Pedure. (2017). Applications of Augmented Reality[Online]. Availabe: https://www.lifewire.com/applications-of-augmented-reality-2495561

[3]. Junwei Yu, Lu Fang and Chuanzheng Lu. (2016). Key technology and application research on mobile augmented reality[Online]. Available: http://ieeexplore.ieee.org.libaccess.sjlibrary.org/document/7883129/

[4]. Stan Kurkovsky, Ranjana Koshy, Vivian Novak, Peter Szul. (2012). Current Issues in Handheld Augmented Reality[online]. Available:http://ieeexplore.ieee.org.libaccess.sjlibrary.org/stamp/stamp.jsp?arnumber=62 85844

[5]. Zunaira Ilyas Bhutta, Syedda Umm-e-Hani, Iqra Tariq. (2015). The next problems to solve in augmentedreality[Online].Available:http://ieeexplore.ieee.org.libaccess.sjlibrary.org/stamp/stamp.jsp?arnumber=74 69490

[6].ingfeng Zhang1 , Weilong Chu1 , Changhong Ji2 , Chengyuan Ke3 , Yamei Li1. (2015). An implementation of generic augmented reality in mobile devices[Online]. Available: http://ieeexplore.ieee.org.libaccess.sjlibrary.org/document/7065112/

[7]. Dimitris Chatzopoulos,Carlos Bermejo,Zhanpeng Huang,Pan Hui.(26 April 2017). Mobile Augmented Reality Survey: Fom Where We Are to Where We Go [Online]. Availale: http://ieeexplore.ieee.org/document/7912316

[8]. V. Ferrari, T. Tuytel aars, an d L. Van Gool, “Markerless Augmented Reality with a Real-Time Affine Region Tracker,” IEEE and ACM Intl. Sym. on Augmented Reality, v. I, 87-96, Oct. 2001.

[9]. A. Henrysson, M. Billinghurst and M. Ollila, “Virtual Object Manipulation Using a Mobile Phone, ” Int. Conf. on Augmented Telexistence, Christchurch, New Zealand, 164-171, Dec. 5-8, 2005

[10]. Augmented reality meets computer vision: efficient data generation for urban driving scenes Alhaija et al., 2018 H.A. Alhaija, S.K. Mustikovela, L. Mescheder, A. Geiger, C. Rother