ARchitects : A marker-less augmented reality application for innovative marketing of architect's portfolio

Submitted in partial fulfillment of the requirements
for the degree of

B.E. Information Technology

By

Shraddha Pawar	**151081**
Monesha Murdeshwar	**151072**
Remon Pereira	**151085**
Sahil Raut	**151090**

Supervisor
Ms. Sonali Vaidya
Assistant Professor

Department of Information Technology
St. Francis Institute of Technology
(Engineering College)

University of Mumbai
2018-2019

Contents

List of Figures

List of Tables

Chapter 1
Introduction

1.1 Introduction:

For many years, Architecture agencies have focused primarily on traditionally building models manually using plaster of paris, cardboard etc. This is a very tedious and cumbersome task. The drawback of this modus operandi is the immobility of the bulky 3D Models. To overcome this architects market their projects using 2D images in portfolios, websites and newspapers. But this resulted in losing the 3D aspect. To get the best of both worlds we are using marker & marker-less augmented reality to develop an application that showcases a 3D model of architect project on users own device.

1.2 Problem Formulation:

The project aims to develop an application that showcases a novel way of advertising architect's portfolio using augmented reality.

1.3 Description:

The main theme of this project is exhibiting architect's portfolio in a 3D view using marker-less augmented reality. With the help of AR 3D models, architects can create a lasting first impression and a more prolific networking experience. The construction companies will have their own application where these projects can be viewed. The users can download the application from the store, scan the image and view the respective detailed 3D model of the same. They can view the main structure of the building, floor plan, fully furnished room as augmented 3D models. On scanning the real site, its description, cost and location with respect to various public utilities and landmarks can be viewed.

1.4 Motivation:

The process of creating physical models involves manual labor and is very cumbersome. AR is a tool that creates an interactive experience, engaging the customer through a rich and rewarding experience. Augmented reality (AR) can turn everyday objects, places, and images, into new, interactive opportunities to engage with their customers using attractive graphics, video, animation, audio, and 3D content that brings in real results. AR can be used to offer consumers with

the latest deals and information, improved product demos, a view under the surface of an image or product, an interactive social media experience and a lot more. Moreover, when people touch or interact with a product, they get an emotional bonding with it which compels them to buy it. This project focuses and establishes the importance of augmented reality in India. Our nation is not much advance in augmented reality applications. An application that integrates Building Information Modelling and Augmented Reality may improve architectures portfolio.

1.5 Proposed Solution:

The problem with the current methodology is that it focuses on building models manually using plaster of paris or cardboard. Our architectural models will be designed and built using blender. Marker-Less technology will be used to view the projects dynamically in 3D environment. Android SDK will be used by unity top generate the android application. The performance evaluation parameters are: Rendering speed, detailing of models and high accuracy. The project incorporates three main features:

- On scanning the Architect's project portfolio image, regardless the image being in any form i.e. physical or digital, the entire 3D model of the project will be displayed on the user's screen.
- User can view the 3D model of the floor plan by scanning the architect's blueprint.
- The real-life project will be scanned and its information like location, distance from landmark will be displayed using Marker-less AR.

1.6 Scope of the project

This system will benefit the architecture field as the process of creating physical models involves manual labor and is very cumbersome. AR is a tool that creates an interactive experience, engaging the customer through a rich and rewarding experience. This project aims at developing an application that will aid the architects in advertising their portfolio in an innovative way using augmented reality. The user interface will be designed as part of the project but will contain, as a minimum, the ability to scan target images as well as the environment using the device camera to display appropriate 3D models and information on the user's display screen.

Chapter 2
Literature Review

A Service-oriented Mobile Augmented Reality Architecture for Personalized Museum Environments.
AR has also been capitalized on in many different ways such as presenting digital contents,e.g. advertising,museum interaction experiences e-commerce and many other scenarios. AR has been an extensive augmenting tool that enhances a user's experience of perceptions in real time; especially in mobile indoor AR such as those concerning interactive graphic AR applications and mobile outdoor platforms, such as location based AR or markerless AR.[1]

Application of Augmented Reality Object in Construction Project.
Previous studies have been conducted on the AR technology in the construction industry. Benefits of the AR technology in enhancing various unit works based on the technology suitability have been proposed.Also a moderate level of study has been conducted on the usefulness of the technology by using HMD device and multi-markers to present mixed reality and 3D drawings and a framework for mixed reality to provide visual information for construction workers.[2]

Building Contextual Augmented Reality Environments with semantics.
The currently available tools for authoring AR environments range from general purpose computer vision and graphics libraries, requiring advanced programming skills to develop applications, to easy-to-use point-and-click packages for mobile devices, enabling creation of simple AR presentations. These tools provide functionality for manual authoring of AR presentations – either through programming or visual design.[3]

Human lips synchronisation in Autodesk Maya.
It is described an automatic way to analyse voice, using Adobe after effects and how the result interacts to Maya, but this way is still in progress, so voice is analyzed and classified manually. Using programming language of Maya - MEL (Maya Embedded Language) the recognized phonemes are associated with mouth positions to provide visemes for computer animation of speech. Different mouth positions are created using blend shape deformers, that let you to deform a surface into the shapes of other surfaces. Lip animation is facilitated by activating facial muscles and the jaw on the given facial model created specially for animation. Highspeed natural-looking synchronized lips animation is achieved.[4]

Table 2.1: Comparison of various papers

Sr no	Title of Paper	Description	Limitation
[1]	A Service-oriented Mobile Augmented Reality Architecture for Personalized Museum Environments	In this abstract [1] the application is designed for personalized museum using marker-less technology and API is built for the same.	Providing a QRcode so that every android user will be able to download the application from google play store.
[2]	Application of Augmented Reality Object in Construction Project.	This study from [2] proposes a methodology of applying the AR technology to a construction project and a prototype system to test applicability of the AR technology in rebar construction work. CAD tools and Unifeye SDK 3.5 was used to implement augmented reality	1. Blender is free software to build 3D models whereas CAD tools are paid products. Hence blender is a better option. 2. Unity software is used to build android application which combines the real time objects with virtual objects.
[3]	Building Contextual Augmented Reality Environments with semantics.	In this abstract [3] a new method of building CARE environments is proposed. The method permits automatic generation of such environments through semantic transformation of high-level domain ontologies and using DART (Designer's Augmented Reality Toolkit) AR application is developed.	Vuforia is a popular freemium platform where using Vuforia Object Scanner, we can scan and create object targets. The recognition process can be implemented using the database (local or cloud storage).
[4]	Human lips synchronisation in Autodesk Maya	This application from abstract [4] solves the issue of how to make human lips synchronization in one of the most popular 3D modelling software - Autodesk Maya.	Autodesk-Maya is complex software to develop the 3D models. So to build 3D objects, blender software which is user-friendly will be used to save time and cost of licensing.

Chapter 3

SYSTEM ANALYSIS

3.1 Functional Requirements

- The Developer is required to create the respective model in Blender and import them to Unity for implenting the application.
- Additionally a developer can create a 3D model using AR-Core, View-AR and Autodesk 3ds Max.
- The image targets need to be uploaded to the Vuforia platform.
- Once the Models are designed and the functions are initialized the work of the developer is done.
- The user is required to hold the smartphones or tablets at a designated area or image target.
- User will be able to provide additional functionalities to the application models.

3.2 Non-Functional Requirements

- Accessibility : The application should be accessible on google play store and apple store.
- Usability : The application should be user-friendly.
- Maintainance : The application interface should be simple and concise so as it can be easily edited in future.
- Acceptance : It should meet user's requirements.
- Responsive : The function response time should be smooth.
- Modifiable : The function should be modifiable .
- Sustainable : The function should be able to be maintained at a certain rate or level.
- User Friendly Graphical Interface.

3.3 Specific Requirements

Hardware :

- CPU : 64 Bit Intel or AMD Processor .
- GPU : Minimum 1GB Graphics Memory with DX10.
- RAM : 4GB or above.
- Memory : Minimum 10 GB for installation and additional project files.
- Operating System : Windows 10 or above.
- User : With a Android phone having minimum 2GB ram and atleast 100mb free space.

Software :

- Unity 3D 2018.2.11 The GUI along with the source file of applications will be created in Unity.
- Android SDK : The Environment to run the final application for Android.
- Kudan AR SDK : Kudan AR SDK provides both tracking and rendering for location based services.
- Vuforia Engine 8.0 : For integrating.
- Blender 2.79b: For Building the models.
- Smartphone :Camera , Android 4.4 and above.

3.4 Use-Case Diagrams for ARchitect

Use Case diagrams, usually referred to as behavior diagrams are used to describe a set of actions that some system or systems should or can perform in collaboration with one or more external users of the system. Use case diagrams are in fact twofold - they are both behavior diagrams, because they describe behavior of the system, and they are also structure diagrams - as a special case of class diagrams where classifiers are restricted to be either actors or use cases related to each other with associations. Use case diagrams are used to specify:

- Requirements (external), required usages of a system under design or analysis (subject) - to capture what the system is supposed to do.
- The functionality offered by a subject – what the system can do.
- Requirements the specified subject poses on its environment - by defining how environment should interact with the subject so that it will be able to perform its services.

Table 3.1: Use Case Diagram Description

Actors	Description
User	The user views the AR 3D model by scanning the portfolio image using the device camera. Additional information of the Architect's project like location and pricing of the building will also be displayed on the user's screen if user selects the option.
Administrator	Administrator can make changes to the application by modifying or adding new models. An extension of the administrator's role is that he can also Add new features if required.

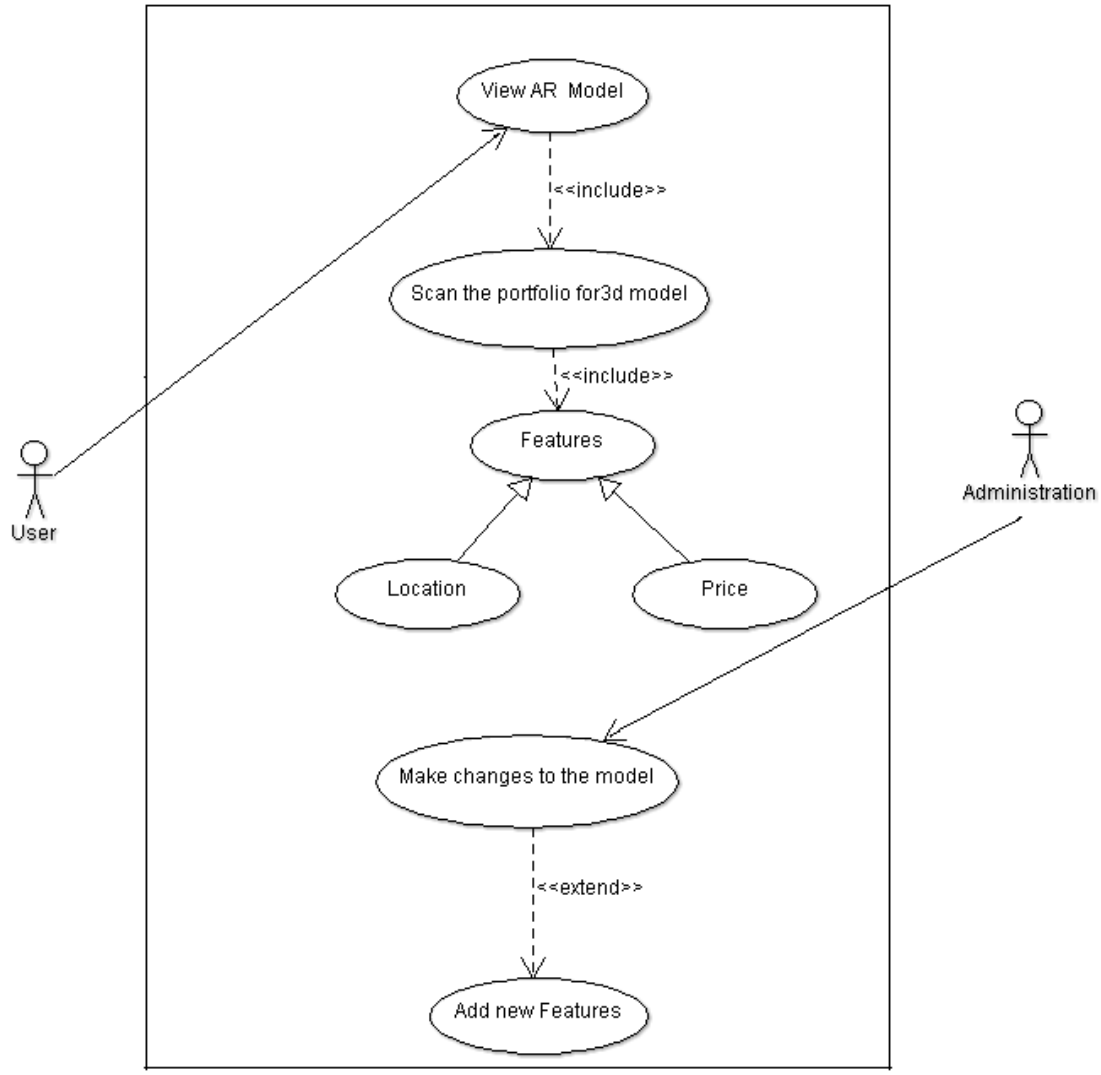

Figure 3.4.1: Use Case for ARchitect

Chapter 4
Analysis Modeling

4.1 Activity Diagrams

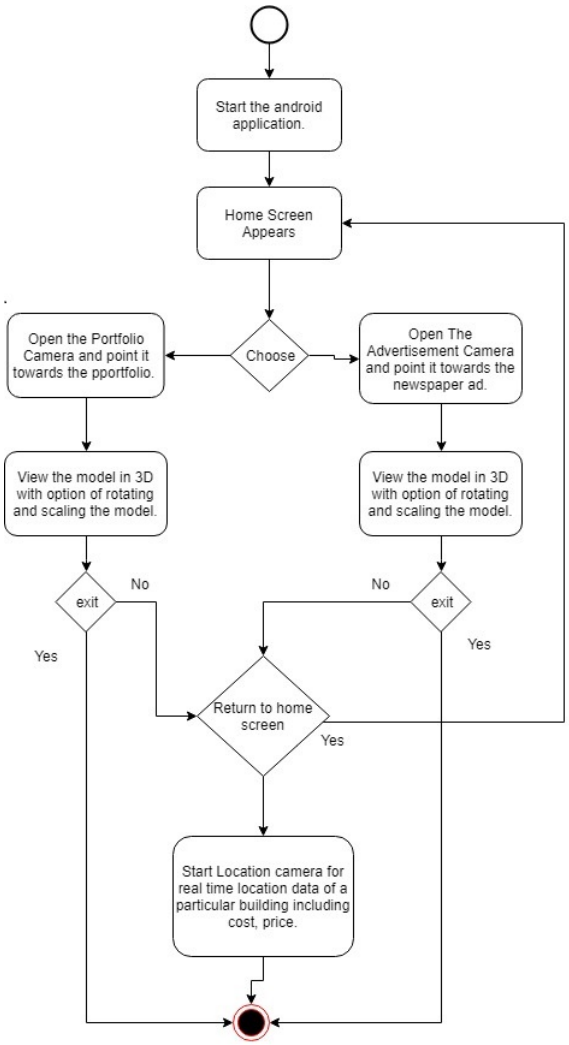

Figure 4.1.1: Activity Diagram for User

The Activity Diagram gives us a gist of the stages in the application Process. The First stage is Opening the application and focusing the camera towards the portfolio pictures. Once the model appears on the screen the user has the option to close the application or use extra features like the location and price of the services and the user can interact with the 3D model.

4.2 Functional Modeling

4.2.1 DFD:Level 0

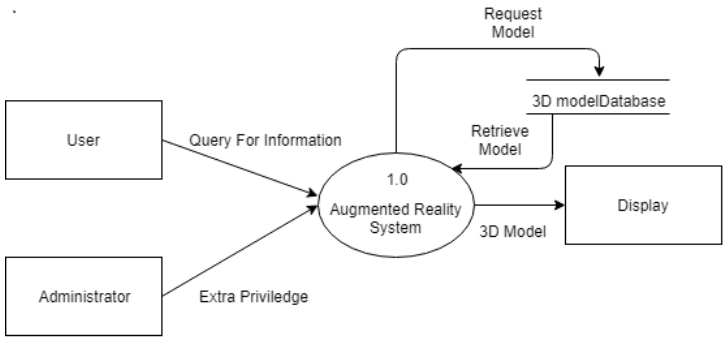

Figure 4.2.1: DFD Level 0

In the above DFD the user queries for information. The Administrator has extra privileges like changing the function and model. The system then creates the desired output and displays it on screen.

4.2.2 DFD:Level 1

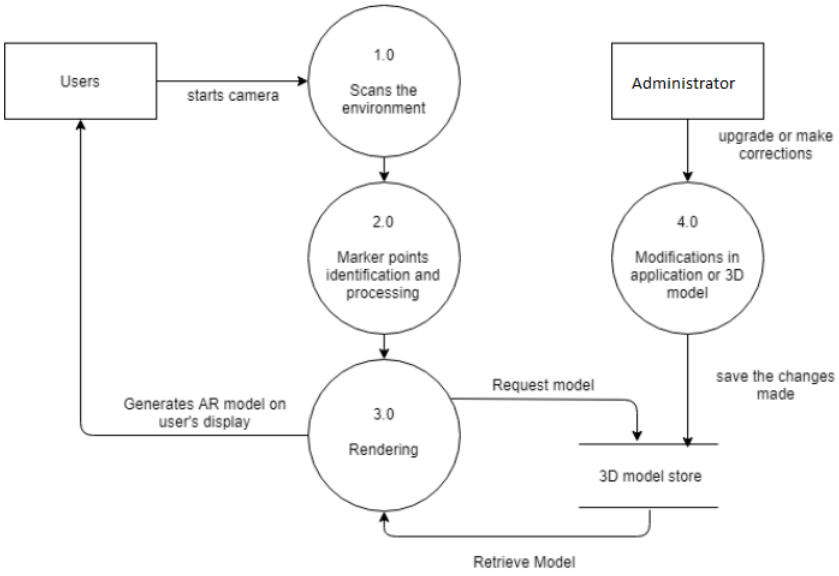

Figure 4.2.2: DFD Level 1

In the above DFD level 1 the user uses his smartphone or tablet to scan the particular environment.
The camera then maps the points according to the image target and render the model in the given area.
Once the Model is rendered the user has the option to edit it on his own way and also the admin has privileges to update and make any corrections in the application.

4.3 TimeLine Chart

	Name	Duration	Start	Finish
1	**ARchitect**	**194 days?**	**26/7/18 8:00 AM**	**23/4/19 5:00 PM**
2	**Project proposal**	**5 days?**	**26/7/18 8:00 AM**	**1/8/18 5:00 PM**
3	Presentation	5 days?	26/7/18 8:00 AM	1/8/18 5:00 PM
4	Proposal Submission	5 days?	26/7/18 8:00 AM	1/8/18 5:00 PM
5	IEEE Papers	5 days?	26/7/18 8:00 AM	1/8/18 5:00 PM
6	**Project Guide**	**5 days?**	**2/8/18 8:00 AM**	**8/8/18 5:00 PM**
7	Meeting Project Guide	5 days?	2/8/18 8:00 AM	8/8/18 5:00 PM
8	Group project File	5 days?	2/8/18 8:00 AM	8/8/18 5:00 PM
9	**Planning**	**5 days?**	**9/8/18 8:00 AM**	**15/8/18 5:00 PM**
10	Literature Review	5 days?	9/8/18 8:00 AM	15/8/18 5:00 PM
11	Deciding Modules	5 days?	9/8/18 8:00 AM	15/8/18 5:00 PM
12	**UML Diagrams**	**5 days?**	**16/8/18 8:00 AM**	**22/8/18 5:00 PM**
13	USE Case Diagram	5 days?	16/8/18 8:00 AM	22/8/18 5:00 PM
14	Approval of Diagrams	5 days?	16/8/18 8:00 AM	22/8/18 5:00 PM
15	Soft Copy Of synopsis	5 days?	16/8/18 8:00 AM	22/8/18 5:00 PM
16	**Component and Software Finalize**	**11 days?**	**23/8/18 8:00 AM**	**6/9/18 5:00 PM**
17	Study of Blender	11 days?	23/8/18 8:00 AM	6/9/18 5:00 PM
18	Study Of markerless AR and Unity	11 days?	23/8/18 8:00 AM	6/9/18 5:00 PM
19	**Modelling And Designing**	**16 days?**	**13/9/18 8:00 AM**	**4/10/18 5:00 PM**
20	Drawing Timeline and WBS	16 days?	13/9/18 8:00 AM	4/10/18 5:00 PM
21	Modelling In blender	16 days?	13/9/18 8:00 AM	4/10/18 5:00 PM
22	Final GUI	16 days?	13/9/18 8:00 AM	4/10/18 5:00 PM
23	**Submission**	**1 day?**	**10/10/18 8:00 AM**	**10/10/18 5:00 PM**
24	Submission Of Whitebook	1 day?	10/10/18 8:00 AM	10/10/18 5:00 PM
25	**Location Based Module Research**	**62 days?**	**11/10/18 8:00 AM**	**4/1/19 5:00 PM**
26	KudanAR Research	36 days?	11/10/18 8:00 AM	29/11/18 5:00 PM
27	Detailed Study of Plugin	26 days?	30/11/18 8:00 AM	4/1/19 5:00 PM
28	**Meeting with Guide**	**9 days?**	**7/1/19 8:00 AM**	**17/1/19 5:00 PM**
29	Detailed Discussion on Work Done so far	3 days?	7/1/19 8:00 AM	9/1/19 5:00 PM
30	Discussion of Implementation Phase	4 days?	9/1/19 8:00 AM	14/1/19 5:00 PM
31	Submission of Rough Draft of Poster for Colloquium IT for Society	3 days?	14/1/19 8:00 AM	16/1/19 5:00 PM
32	Submission of Final Poster for Colloquium IT for Society	2 days?	16/1/19 8:00 AM	17/1/19 5:00 PM
33	**Poster Presentation**	**2 days?**	**18/1/19 8:00 AM**	**21/1/19 5:00 PM**
34	Building sample application for poster presentation	2 days?	18/1/19 8:00 AM	21/1/19 5:00 PM
35	**Project Implementation**	**25 days?**	**22/1/19 8:00 AM**	**25/2/19 5:00 PM**
36	Refine Scope of Project	2 days?	26/1/19 8:00 AM	29/1/19 5:00 PM
37	Implementation of Marker-Less Feature	11 days?	22/1/19 8:00 AM	5/2/19 5:00 PM
38	UI design and Geolocation	3 days?	5/2/19 8:00 AM	7/2/19 5:00 PM
39	Addition of Rotation, Scaling and Translation Feature	3 days?	7/2/19 8:00 AM	11/2/19 5:00 PM
40	Building 3D Model of Floor Plan In Blender	6 days?	11/2/19 8:00 AM	18/2/19 5:00 PM
41	Screenshot Feature	5 days?	19/2/19 8:00 AM	25/2/19 5:00 PM
42	**Mid Sem Evaluation**	**4 days?**	**26/2/19 8:00 AM**	**1/3/19 5:00 PM**
43	Evaluation of code	4 days?	26/2/19 8:00 AM	1/3/19 5:00 PM
44	Discussion about Documentation	1 day?	1/3/19 8:00 AM	1/3/19 5:00 PM
45	**Documentation**	**31 days?**	**4/3/19 8:00 AM**	**15/4/19 5:00 PM**
46	Result Analysis	4 days?	4/3/19 8:00 AM	7/3/19 5:00 PM
47	Update timeline chart and Diagrgams	3 days?	7/3/19 8:00 AM	11/3/19 5:00 PM
48	Technical paper Publish	8 days?	11/3/19 8:00 AM	20/3/19 5:00 PM
49	Final Report Documentation	3 days?	20/3/19 8:00 AM	22/3/19 5:00 PM
50	PPT of Work	16 days?	25/3/19 8:00 AM	15/4/19 5:00 PM
51	**Testing**	**5 days?**	**16/4/19 8:00 AM**	**22/4/19 5:00 PM**
52	Unit testing	4 days?	16/4/19 8:00 AM	19/4/19 5:00 PM
53	Integrate testing	3 days?	16/4/19 8:00 AM	18/4/19 5:00 PM
54	Black Box testing	3 days?	16/4/19 8:00 AM	18/4/19 5:00 PM
55	Alpha testing	5 days?	16/4/19 8:00 AM	22/4/19 5:00 PM
56	**Submission**	**1 day?**	**23/4/19 8:00 AM**	**23/4/19 5:00 PM**
57	Final Black Book	1 day?	23/4/19 8:00 AM	23/4/19 5:00 PM
58	Project Submission	1 day?	23/4/19 8:00 AM	23/4/19 5:00 PM

Figure 4.3.1: Time Chart Tasks

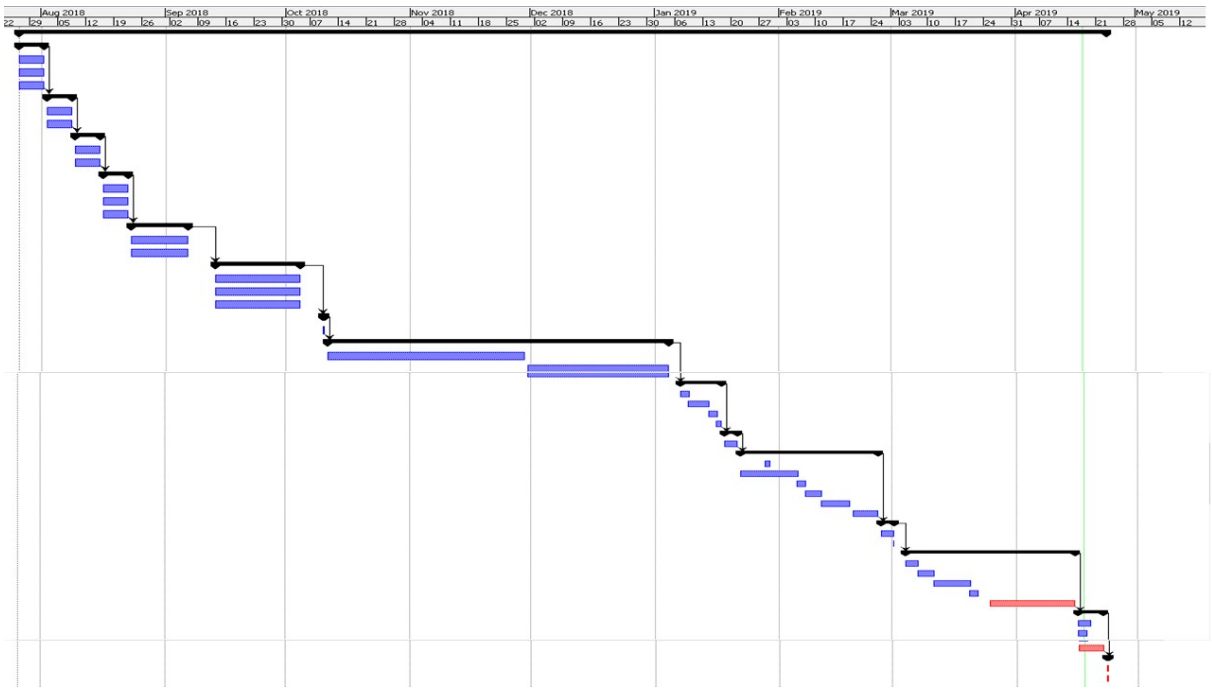

Figure 4.3.2: Timeline From July 2018 - October 2018

Chapter 5

Design

5.1 System Architecture

The given System Flowchart explains the entire Architectural Design of the application. It shows how exactly the data or commands will be processed starting from the user or the developer.

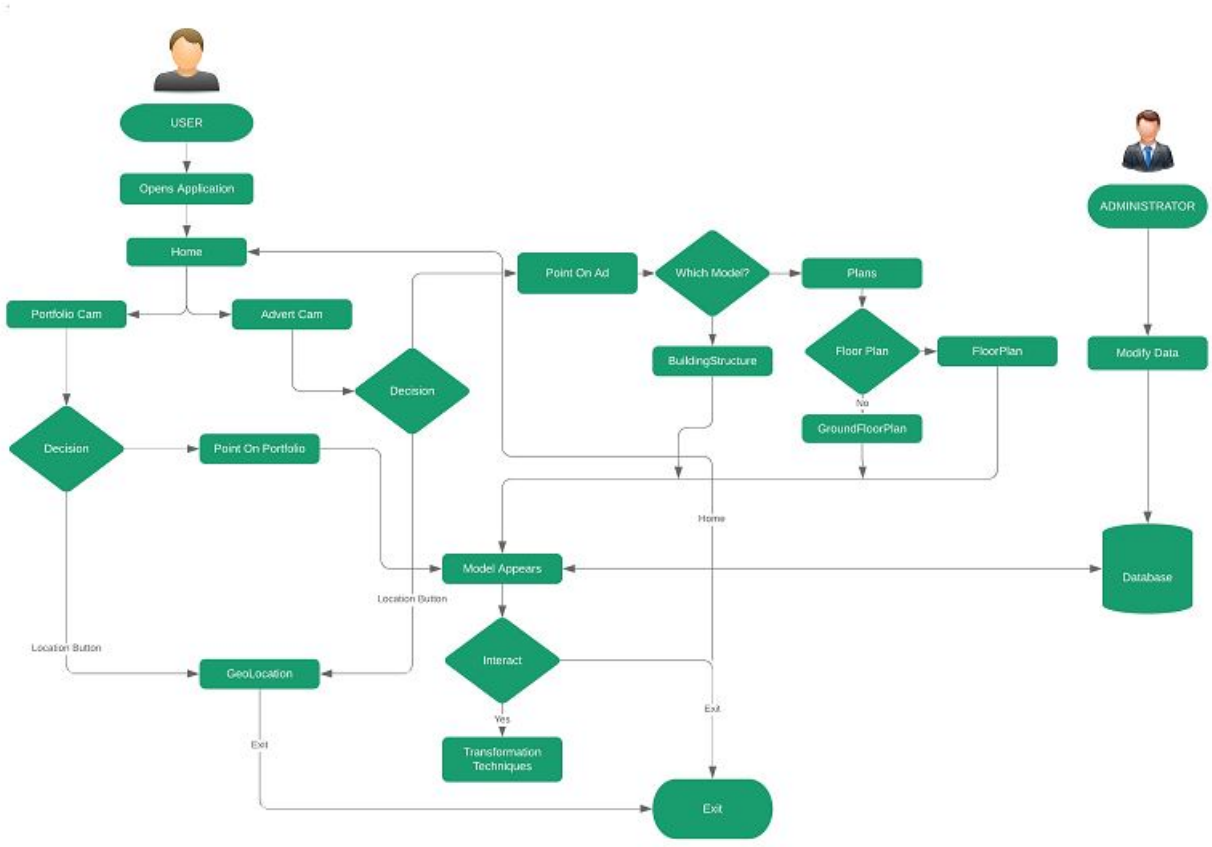

Figure 5.1.1: System Architecture

The System is mainly divided into two parts that are the user side and administrator side. The user and the administrator have specific privileges in the system. Users are the individuals who make use of the basic functionalities of the application. Administrators are the developers of the

system having privileged of the backend functionalities like modifications in the application and 3D models. In the following flowchart, the user accesses the interface of the application wherein the AR camera comes into play.The System comprises of two sides the portfolio camera and the advertisement camera.After selection of the specific camera the user has to point the camera towards a target Image set as a reference in the database or the user can point towards a plane surface where the user can use the functionality of the button to render an object in an open space. Once the object is rendered from the database connected to the application various tasks such as scale, translate, rotate can be executed.The user also has an option of geolocation service wherin the user needs to point out the camera in the location of the building for seeing information of the specific building such as price,distance from station.

5.2 Graphical user Interface Design

Figure 5.2.1: Portfolio first page

This is the first page of the Architect's portfolio.

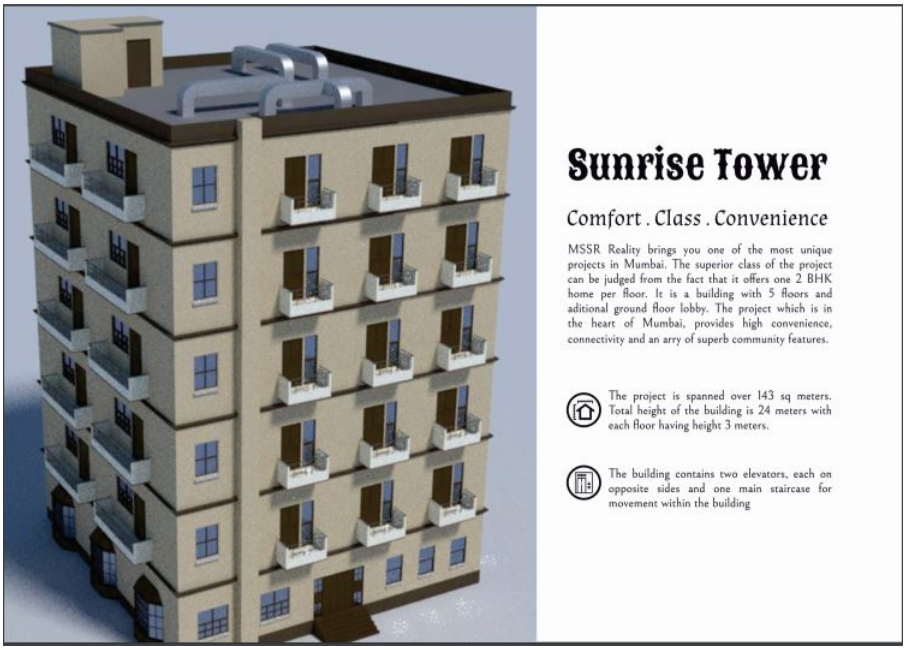

Figure 5.2.2: Portfolio second page

This is the second page which consist of the Building model image.

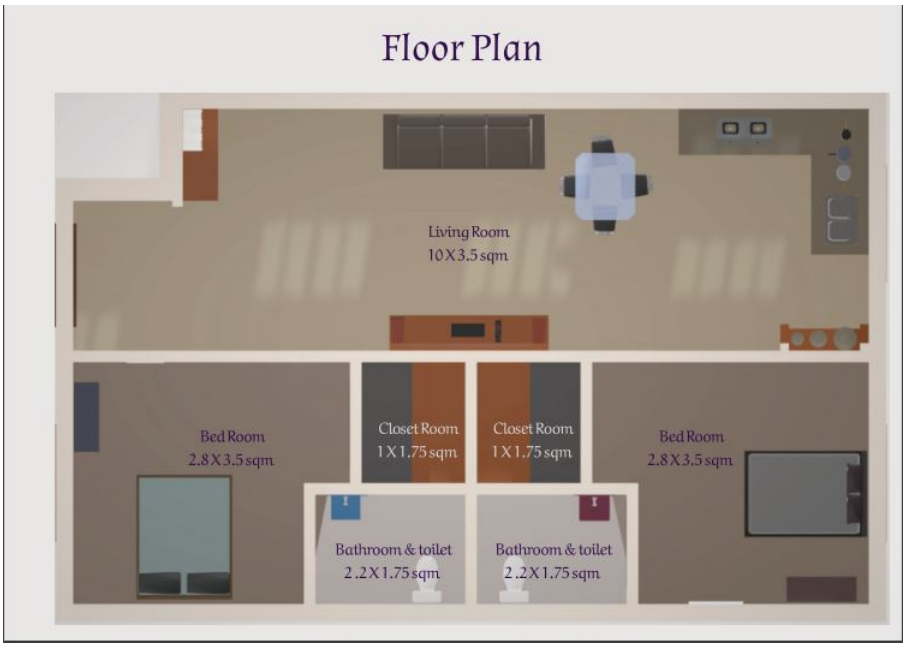

Figure 5.2.3: Portfolio third page

This is the third page which consist of the Building floorplan model image.

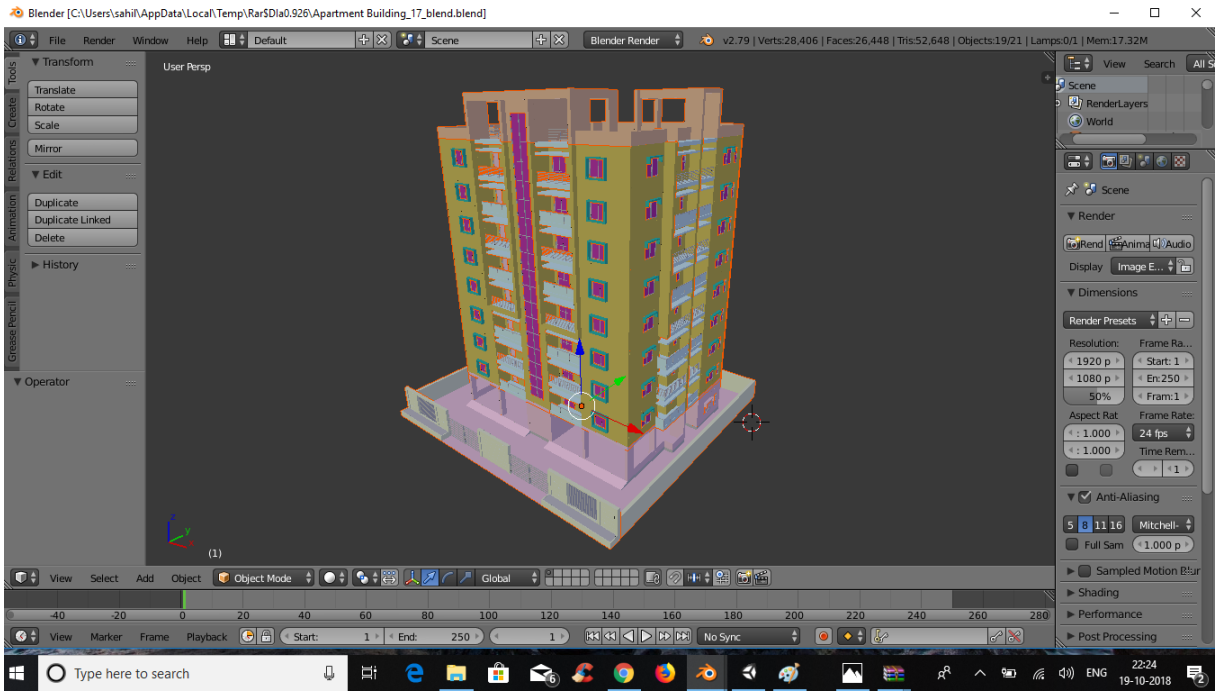

Figure 5.2.4: Model Design in Blender

The Modelling of the Architect's Portfolio Building is done in Blender environment.

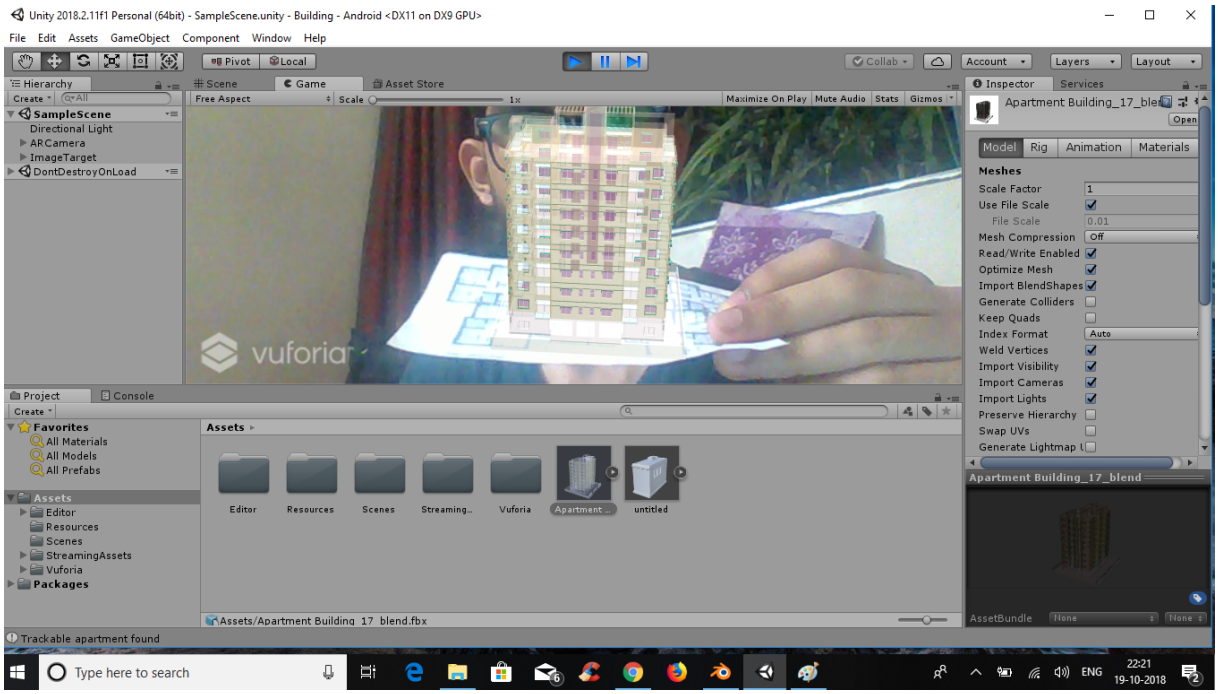

Figure 5.2.5: Implementation in Unity using Vuforia license

The Blender Model is implemented in Unity Environment and using Vuforia database.

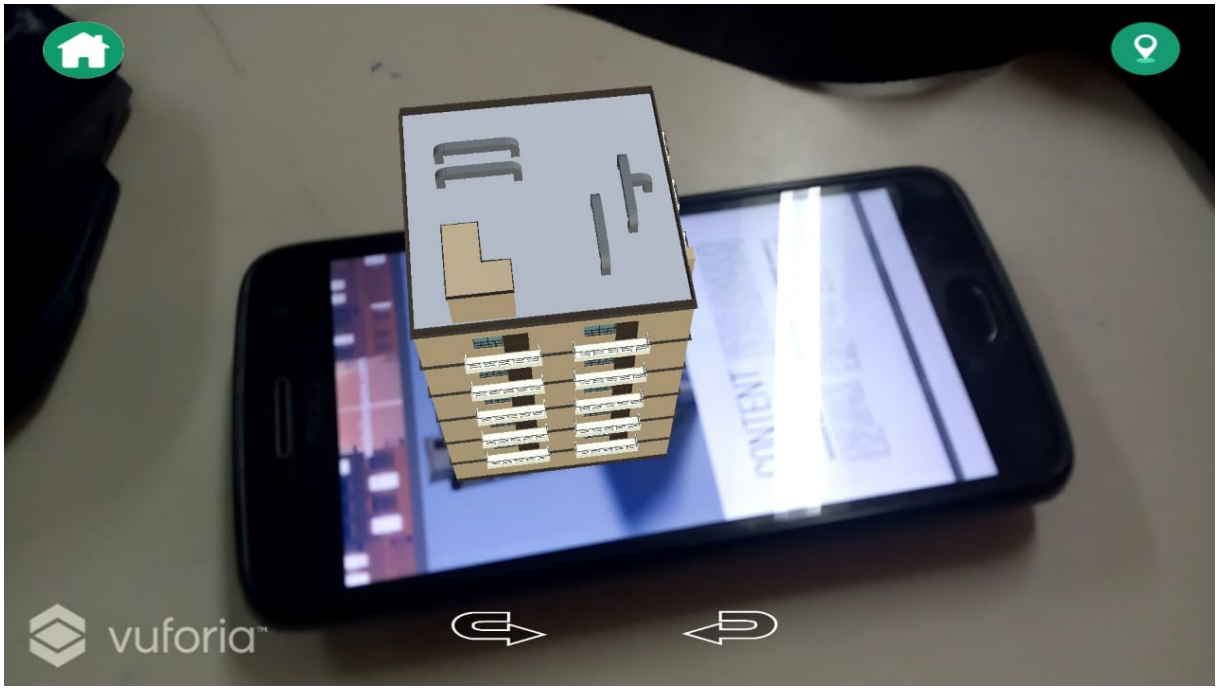

Figure 5.2.6: Application displaying Building model

The Modelling of the Architect's Portfolio Building is done in Blender environment.

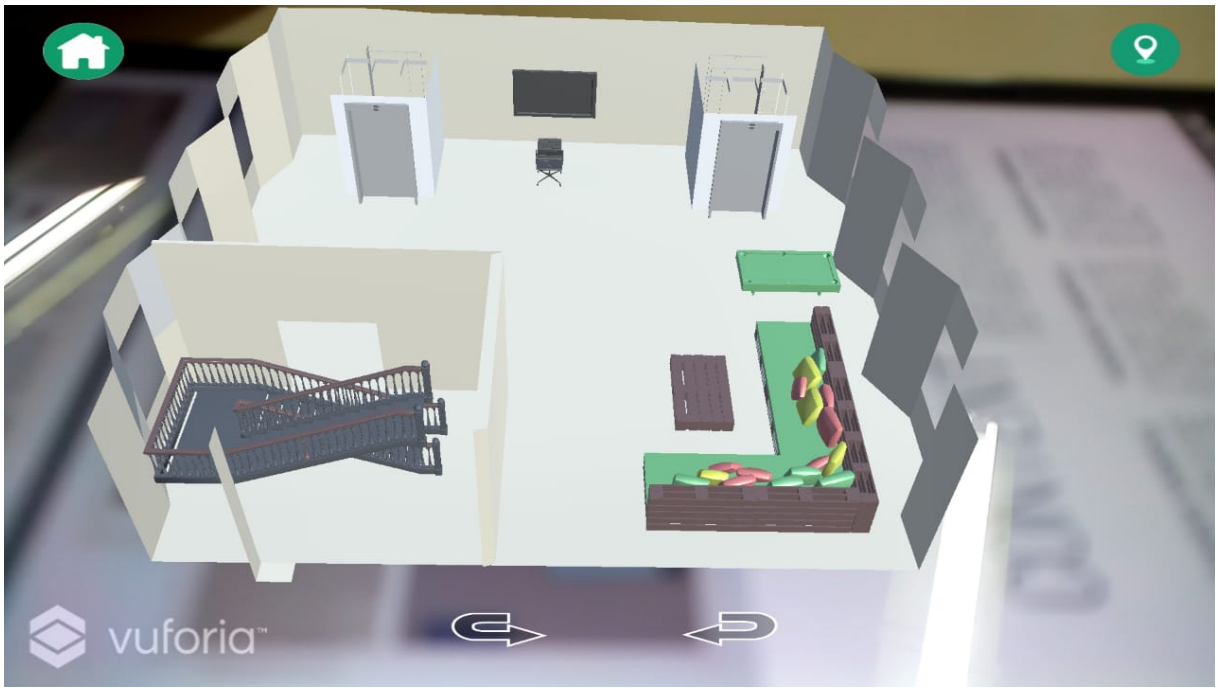

Figure 5.2.7: Application displaying ground floor model

The Blender Model is implemented in Unity Environment and using Vuforia database.

Chapter 6
Implementation

6.1 Working of the Project

The proposed system uses augmented reality to enhance the user experience for viewing an architect's work. Augmented reality creates almost an exact copy of how the structure would look in the real world. The references are image based targets in real life and also any given area. Digital information is overlayed over real objects.

Implementation involves the following steps:
Step 1: Data gathering
First the architectural data is gathered for which the application is to be made. According to the data gathered the models are designed.

Step 2: Model Designing
As this project deals with architect designs, building structures and layouts are the basic design. To create 3D models Blender and unity is used. Blender v2.79 is an opensource software which provides extensive functionalities in the field of modelling, animation, simulations, composting.

Step 3: Importing models and setting up unity
Once the model is developed it is imported to unity. To make the models visually appealing in the application various lightings and camera angles need to be set in unity.

Step 4: Uploading image targets and setting up vuforia
Vuforia database is used to save the image targets. Vuforia package is imported in unity which contains all the necessary prefabs. The license key of the database has to be entered in unity.

Step 5: Coding the functions
All the necessary functions are implemented using C# scripts.

Step 6: Building the application
Once android is set up and all the settings are done the application is compiled and is ready to run.

Step 7: Testing
The targets are scanned via a smartphone/tablets camera and the object gets loaded. With the help of marker-less technology the models should be placed manually and adjusted.

The application is implemented in several phases:

Phase 1: Model Creation

In this step,3D model of building is created using Blender, floor plan models are designed using Unity3D and ProBuilder plugin. These models are created relevant to the topic textual material. Blender and Unity3D allows the developer to provide textures, colors and shapes in multiple dimensions.

Phase 2: Light and Shadow Module

In this phase, the 3D models are given characteristic lighting, shadows,color effects, etc. to depict the relevant processes and mechanisms. These steps are done using Blender.

Phase 3: Integration and Database Connectivity

In this step, all the 3D models are stored in vuforia database and these models are imported to Unity3D by proving vuforia license.

Phase 4: 3D Model Generation and Marker-less Recognition

Once the Unity3D and vuforia connectivity is completed, various features like scaling, rotation, animations and its size relative to the device screen are defined.The Marker-less technology feature which allows you to show the models even after removing the target image is imported. UI for the same is designed on Unity3D.

Phase 5:Application Development

In this phase, Android connectivity with Unity3D is done by importing Android NDK, SDK files. Player Settings of Unity3D are set as per the application requirements and finally Android (.apk) is builded.

6.2 Application Performance

The user requires a smartphone with the application that works as follows:

- Target Detection:

  When the user clickes on the button for viewing 3D advertisement i.e. magazine or newspaper,the respective AR camera is activated which scans the target image.

- Model Generation:

  After scanning the target image respective 3D model appears on the screen in an augmented view along with its characteristics visible on the screen.

- Marker-less and Marker based Recognition:

  Marker-less technology allows user to view models even if the target image is removed, which gives advantage of spectating 3D models as per user's choice location. Whereas Marker based technology allows user to viw the 3D model while scanning target image once the target image is removed user won't be able to see the models.

- Live Location Statistics:

  This feature allows user to scan real environment and shows surrounding information of area in an augmented view by using GPS.

- User Interaction:

  The user is provided with several on-click functionalites to interact with the models and view additional information. For any given model, the user will just have to scan the target image for the model to appear on screen. Once model is generated user need not to stand steadily and can move freely. The generated 3D model is accompanied with an interactive GUI that provides on-click functionalities to user.

  Functionalities included in the application are:

- Scale: This includes zoom in and zoom out opeartions by using gestures.
- Rotate: The generated model can be rotated to the right or left as per the user's requirement by using provided buttons.
- Translation: This includes moving the 3D model wherever the user wants.
- Geolocation: User can see the various building related statistics like distance from railway station, nearest bus station, price of flat etc. on scanning the real environment of the building.

6.3 Code

UserInterfaceButtons.cs

```
using UnityEngine;
using System.Collections;
using System.IO;

public class UserInterfaceButtons : MonoBehaviour
{
public float scalingSpeed = 0.03f;
public float rotationSpeed = 70.0f;
public float translationSpeed = 5.0f;
// public GameObject Model;
bool repeatScaleUp = false;
bool repeatScaleDown = false;
bool repeatRotateLeft = false;
bool repeatRotateRight = false;
bool repeatPositionUp = false;
bool repeatPositionDown = false;
bool repeatPositionLeft = false;
bool repeatPositionRight = false;

void Update ()
{
if (repeatScaleUp) {
ScaleUpButton ();
}

if (repeatScaleDown) {
ScaleDownButton ();
}

if (repeatRotateRight) {
RotationRightButton();
}

if (repeatRotateLeft) {
RotationLeftButton();
}

if (repeatPositionUp) {
PositionUpButton();
}

if (repeatPositionDown) {
PositionDownButton();
}

if (repeatPositionLeft) {
PositionLeftButton();
}

if (repeatPositionRight) {
PositionRightButton();
}

```

```
54  }
55
56  public void CloseAppButton ()
57  {
58  Application.Quit ();
59  }
60
61  public void RotationRightButton ()
62  {
63  // transform.Rotate (0, -rotationSpeed * Time.deltaTime, 0);
64  GameObject.FindWithTag ("Model").transform.Rotate (0, -rotationSpeed
        * Time.deltaTime, 0);
65  }
66
67  public void RotationLeftButton ()
68  {
69  // transform.Rotate (0, rotationSpeed * Time.deltaTime, 0);
70  GameObject.FindWithTag ("Model").transform.Rotate (0, rotationSpeed
       * Time.deltaTime, 0);
71  }
72
73  public void RotationRightButtonRepeat ()
74  {
75  // transform.Rotate (0, -rotationSpeed * Time.deltaTime, 0);
76  repeatRotateRight=true;
77  }
78
79  public void RotationLeftButtonRepeat ()
80  {
81  // transform.Rotate (0, rotationSpeed * Time.deltaTime, 0);
82  repeatRotateLeft=true;
83  }
84
85  public void ScaleUpButton ()
86  {
87  // transform.localScale += new Vector3(scalingSpeed, scalingSpeed,
       scalingSpeed);
88  GameObject.FindWithTag ("Model").transform.localScale += new Vector3
        (scalingSpeed, scalingSpeed, scalingSpeed);
89  }
90
91  public void ScaleUpButtonRepeat ()
92  {
93  repeatScaleUp = true;
94  Debug.Log ("Up");
95  }
96  public void ScaleDownButtonRepeat ()
97  {
98  repeatScaleDown = true;
99  Debug.Log ("Down");
100 }
101 public void PositionDownButtonRepeat ()
102 {
103 repeatPositionDown = true;
104 }
105 public void PositionUpButtonRepeat ()
106 {
107 repeatPositionUp = true;
```

```
108 }
109 public void PositionLeftButtonRepeat ()
110 {
111 repeatPositionLeft = true;
112 }
113 public void PositionRightButtonRepeat ()
114 {
115 repeatPositionRight = true;
116 }
117
118 public void ScaleUpButtonOff ()
119 {
120 repeatScaleUp = false;
121 Debug.Log ("Off");
122 }
123 public void ScaleDownButtonOff ()
124 {
125 repeatScaleDown = false;
126 Debug.Log ("Off");
127 }
128
129 public void RotateLeftButtonOff ()
130 {
131 repeatRotateLeft = false;
132 Debug.Log ("Off");
133 }
134
135 public void RotateRightButtonOff ()
136 {
137 repeatRotateRight = false;
138 Debug.Log ("Off");
139 }
140 public void PositionRightButtonOff ()
141 {
142 repeatPositionRight = false;
143 Debug.Log ("Off");
144 }
145 public void PositionLeftButtonOff ()
146 {
147 repeatPositionLeft = false;
148 Debug.Log ("Off");
149 }
150 public void PositionUpButtonOff ()
151 {
152 repeatPositionUp = false;
153 Debug.Log ("Off");
154 }
155 public void PositionDownButtonOff ()
156 {
157 repeatPositionDown = false;
158 Debug.Log ("Off");
159 }
160
161 public void ScaleDownButton ()
162 {
163 // transform.localScale += new Vector3(-scalingSpeed, -scalingSpeed,
      -scalingSpeed);
```

```
164  GameObject.FindWithTag ("Model").transform.localScale += new Vector3
         (-scalingSpeed, -scalingSpeed, -scalingSpeed);
165  }
166
167  public void PositionUpButton ()
168  {
169  GameObject.FindWithTag ("Model").transform.Translate (0, 0, -
         translationSpeed * Time.deltaTime);
170  }
171
172  public void PositionDownButton ()
173  {
174
175  GameObject.FindWithTag ("Model").transform.Translate (0, 0,
         translationSpeed * Time.deltaTime);
176  }
177
178  public void PositionRightButton ()
179  {
180  GameObject.FindWithTag ("Model").transform.Translate (-
         translationSpeed * Time.deltaTime, 0, 0);
181  }
182
183  public void PositionLeftButton ()
184  {
185  GameObject.FindWithTag ("Model").transform.Translate (
         translationSpeed * Time.deltaTime, 0, 0);  // backward
186  }
187
188  public void ChangeScene (string a)
189  {
190  Application.LoadLevel (a);
191  }
192
193  public void AnyButton ()
194  {
195  Debug.Log ("Any");
196  }
197  }
```

DetectLocation.cs

```
1  using UnityEngine;
2  using System.Collections;
3  using UnityEngine.UI;
4  using Kudan.AR.Samples;
5
6  public class DetectLocation : MonoBehaviour
7  {
8
9  private Vector2 targetCoordinates;
10 private Vector2 deviceCoordinates;
11 private float distanceFromTarget = 0.4f;
12 private float proximity = 0.001f;
13 private float sLatitude, sLongitude;
14 //Set Values foor targewt location
```

```
15 public float dLatitude = 19.253445f, dLongitude = 72.854569f;
16 private bool enableByRequest = true;
17 public int maxWait = 10;
18 public bool ready = false;
19 public Text text;
20 public SampleApp sa;
21
22 void Start()
23 {
24 targetCoordinates = new Vector2(dLatitude, dLongitude);
25 StartCoroutine(getLocation());
26 }
27
28 IEnumerator getLocation()
29 {
30 LocationService service = Input.location;
31 if (!enableByRequest && !service.isEnabledByUser)
32 {
33 Debug.Log("Location Services not enabled by user");
34 yield break;
35 }
36 service.Start();
37 while (service.status == LocationServiceStatus.Initializing &&
       maxWait > 0)
38 {
39 yield return new WaitForSeconds(1);
40 maxWait--;
41 }
42 if (maxWait < 1)
43 {
44 Debug.Log("Timed out");
45 yield break;
46 }
47 if (service.status == LocationServiceStatus.Failed)
48 {
49 Debug.Log("Unable to determine device location");
50 yield break;
51 }
52 else
53 {
54 text.text = "Target Location : " + dLatitude + ", " + dLongitude + "\
       nMy Location: " + service.lastData.latitude + ", " + service.
       lastData.longitude;
55 sLatitude = service.lastData.latitude;
56 sLongitude = service.lastData.longitude;
57 }
58 //service.Stop();
59 ready = true;
60 startCalculate();
61 }
62 void Update()
63 {
64
65 }
66
67 public void startCalculate()
68 {
69 deviceCoordinates = new Vector2(sLatitude, sLongitude);
```

```
70 proximity = Vector2.Distance(targetCoordinates, deviceCoordinates);
71 if (proximity <= distanceFromTarget)
72 {
73 text.text = text.text + "\nDistance : " + proximity.ToString();
74 text.text += "\nTarget Detected";
75 sa.StartClicked();
76 }
77 else
78 {
79 text.text = text.text + "\nDistance : " + proximity.ToString();
80 text.text += "\nTarget not detected, too far!";
81 }
82 }
83 }
```

Landscape.cs

```
1 using UnityEngine;
2
3 public class Example : MonoBehaviour
4 {
5 // Start in landscape mode
6 void Start()
7 {
8 Screen.orientation = ScreenOrientation.Landscape;
9 }
10 }
```

Portrait.cs

```
1 using UnityEngine;
2
3 public class Example : MonoBehaviour
4 {
5 // Start in landscape mode
6 void Start()
7 {
8 Screen.orientation = ScreenOrientation.Portrait;
9 }
10 }
```

Chapter 7

Testing

7.1 Test Cases

Table 7.1: Test case for rotation of the model

Test Id	141030
Test Name	Rotate the model
Test Steps	User rotates the model by clicking ”Rotate Left” and ”Rotate Right” buttons to spectate the 180 degree view of the model.
Expected Result	Model is rotated as per the user’s requirements.
Actual Result	As expected
Fail/Pass	Pass

Table 7.2: Test case for scaling the size of model

Test Id	141031
Test Name	Scale the model
Test Steps	User scales the model by pinching it using two fingers to experience the various sizes of model.
Expected Result	Model size is changed.
Actual Result	As expected
Fail/Pass	Pass

Table 7.3: Test case for detecting live location

Test Id	141032
Test Name	Geolocation
Test Steps	User scans the live location of a site by AR camera through application.
Expected Result	Information regarding the site as well as surroundings is displayed in 3D.
Actual Result	User's location is detected using GPS and information is shown.
Fail/Pass	Pass

Table 7.4: Test case for critical environment conditions

Test Id	141033
Test Name	Testing under critical conditions
Test Steps	User scans the target image with less brightness and in a dark room.
Expected Result	Model should be generated.
Actual Result	No model is generated.
Fail/Pass	Fail

Table 7.5: Test case for rendering differnt models on-click

Test Id	141034
Test Name	On-click models
Test Steps	On-clicking different buttons for news paper advertisement, as per the clicks various models should be rendered with minimum amount of time.
Expected Result	On-clicking various models are generated as per the user's requirements.
Actual Result	As expected.
Fail/Pass	Pass

7.2 Testing Methods Used

7.2.1 Unit Testing

- Unit testing is a software testing method by which individual units of souce code, set of one or more computer program modules together with associated contrl data,usage procedures and operating procedures are tested to determine whether they are fit for use.
- Here, we have used unit testing to test and efficient working of indivual AR 3D models. On tapping icons, the various models are rendered and functions regarding to respective models are performed. We have performed these steps for each and every function and can conclude that every model is displayed correctly as per our need.

7.2.2 Integration Testing

- Integratin testing is the phase in software testing in which individual software modules combined and tested as a group. It occurs after unit testing and before validation testing. Integration testing takes as its input modules that have been unit tested, groups them in larger aggregates, applies tests defined in an integration test plan to those aggregates, and delivers s its output the inegrated system ready for system testing.
- We have used integration testing to verify individual modules such as home screen, geolocation for scanning real life environment. we also tested the marker-less feature of the application.

7.2.3 Usability Testing

- Usability testing is a testing method that measures an application's ease-of-use from the end-user perspective. The goal is to determine whether or not the visible design and aesthetics of an application meet the intended workflow for various processes, such as logging into an application. Usability testing is a great way for teams to review separate functions, or the system as a whole is intuitive to use.g.
- Usability testing was used to verify whether the application is easy to understand and use from the users perspective and also test if all the functions are working properly.

7.2.4 Smoke and Sanity Testing

- Sanity testing determines whether it is reasonable to proceed with further testing. Smoke testing consists of minimal attempts to operate the software, designed to determine whether there are any basic problems that will prevent it from working at all. Such tests can be used as a build verification test.
- This testing method was done to see if any errors are existing in the current functionalities and whether it is feasible to build the application further.

Chapter 8
Results and Discussion

The main purpose of ARchitect was to innovatively market the architect's portfolio thus helping to convert the application users into potential buyers. The application was successfully built on the Android platform considering the operating system of the majority smartphones in India. ARchitect has an attractive and user-friendly UI that helps boost the appeal of the project as a whole. It has two separate modes, one used when the user has the architect's portfolio and the other when the user only has the advertisement image. The homescreen UI has two separate buttons to initialize the two modes. The application detects the target and generates the corresponding Marker-Less AR model. It also generates live location statistics of the architectural project that is under construction. A user can interact with the model generated using the functionalities like scale, rotate and on-click model details. The size of the application is relatively small that is around 50MB with a model retrieving speed of 1ms. The models displayed are highly detailed with efficient accuracy on appearing over the specified target image. One limitation of the application is its inability to detect the target images under varied environmental conditions like low lightings, reflection on images and dark room.

Chapter 9
Conclusion

This system will benefit the architecture field as the process of creating physical models involves manual labor and is very cumbersome. AR is a tool that creates an interactive experience, engaging the customer through a rich and rewarding experience. This project develops an application that aids the architects in advertising their portfolio in an innovative way using augmented reality. All the architect's plans for that project will be converted to 3D using detailed 3D modelling. There will be an application for iPhones and Android based systems along with the corresponding user interface. The application will have the minimum ability to scan target images as well as the environment using the device camera to display appropriate 3D models and information on the user's display screen using markerless augmented reality. It is assumed that the project is one application per architect's project and all the architect's portfolio images can be realised into 3D models.

Biblography

[1] Sasithorn Rattanarungrot,Martin White, “A Service-oriented Mobile Augmented Reality Architecture for Personalized Museum Environments”, International Conference on Virtual Systems and Multimedia (VSMM), Electronic ISBN: 978-1-4799-7227-2.

[2] Hyeon-seung Kim, Chang-hak Kim , Hyoun-seok Moon , So-yeong Moon , Younghwan Kim , Leen-seok Kang “Application of Augmented Reality Object in Construction Project” ,Third World Congress on Information and Communication Technologies ,Electronic ISBN: 978-1-4799-3230-6.

[3] Krzysztof Walczak, Dariusz Rumi´nski, Jakub Floty´nski,“Building Contextual Augmented Reality Environments with semantics”,Published in: 2014 International Conference on Virtual Systems and Multimedia (VSMM),DOI: 10.1109/VSMM.2014.7136656.

[4] Antonio Moura ; Ingrida Mazonaviciute ; Joao Nunes ; Justinas Grigaravicius ” Human lips synchronisation in Autodesk Maya” : 2012 International Conference on Information Management, Innovation Management and Industrial Engineering.

[5] M. Chu, J. Matthews and P. Love. 2017. Integrating mobile Building Information Modelling and Augmented Reality systems: An experimental study. Automation in Construction, vol.85,pp.305-316,2018.Available:10.1016/j.autcon.2017.10.032.

[6] A. Clark, A. Dünser, and R. Grasset.2011. An interactive augmented reality coloring book. SIGGRAPH Asia 2011 Emerg. Technol. - SA ’11, pp.

[7] L. Pombo and M. M. Marques.2017. Marker-based augmented reality application for mobile learning in an urban park: Steps to make it real under the EduPARK project. International Symposium on Computers in Education (SIIE).

[8] Article title: What is Unity? - Definition from Techopedia
Website title: Techopedia.com
URL: https://www.techopedia.com/definition/28545/unity.

[9] Author Blender Foundation Article title: blender.org - Home of the Blender project - Free and Open 3D Creation Software
Website title: blender.org
URL: https://www.blender.org/

[10] Article title: Vuforia Augmented Reality SDK
Website title: En.wikipedia.org
URL:https://en.wikipedia.org/wiki/Vuforia_Augmented_Reality_SDK

Acknowledgement

We are extremely grateful to our college St. Francis Institute of Technology for the confidence bestowed in us and entrusting out project entitled ”ARchitects : A marker-less augmented reality application for innovative marketing of architect’s portfolio”. We express our sincere gratitude to our re- spected director Bro. Jose Thuruthiyil, our principal Dr. Sincy George and our HOD Dr. Joanne Gomes for encouragement and facilities provided to us. We owe our profound gratitude to our project guide Ms. Nazneen Ansari, who introduced us to the methodology of work and timely completion of various stages of our project. Many people especially all team members and classmates have made valuable comments on this proposal which gave us inspiration to improve our assignment.