# PROFORMA OF COVER PAGE OF THE PROJECT REPORT
## University of Mumbai

Augmented Reality Car Customize App

SUBMITTED BY

MAYURI SUNIL PATADE
4102218

DATE OF SUBMISSION

10/06/2020

Under the guidance of

Prof. DHANRAJ JADHAV

Submitted in partial fulfillment of the requirements for qualifying M.Sc. Part II Examination

***SATHAYE COLLEGE***

***PROJECT CERTIFICATE FOR M.Sc. Part II STUDENTS***

*This is to certify that the Project done at AUGMENTED REALITY CAR CUSTOMIZE APP*

*by*

*Mr./Ms. MAYURI SUNIL PATADE*

*(Seat No. 4102218 ) in partial fulfillment for M.Sc. Degree Examination has been found satisfactory. This report had not been submitted for any other examination and does not form part of any other course undergone by the candidate.*
*It is further certified that he/she has completed the Five Phases.*

| | | |
|---|---|---|
| *Signature* | *Signature* | *Signature* |
| *Lecturer Incharge* | *External Examiner* | *Head ofthe Department* |
| *Guided by* | *Examined By* | *Certified by* |

*College Stamp*

# Index

# ACKNOWLEDGEMENT

# Acknowledgement

It is a great pleasure to get this opportunity and sincerely thank all the people, who have showed me the way to create a successful project.

I am highly obliged to the teaching and non-teaching staff members of the Information Technology Department who spared efforts in making the project a successful with their guidance, appropriate advice and encouragement and of course inspiration without which the project would be ineffective.

I sincerely thank and express my profound gratitude and indebtedness to my project guide **Prof.Dhanraj Jadhav** for their timely support and prestigious guidance required for the project completion at each phase of the project development apart from inspiration and motivation.

# Synopsis of the Project

# Augmented Reality Car Customize App.

**Name \ Title of the Project** - Augmented Reality Car Customize App

**Statement about the Problem** - Many of people doesn't know about the car or the car features for that people can necessary to visit the showroom. In Before buying a car Customer needs to go car showroom to decide car. To travel to the showroom, he has to travel, he wastes time and takes time to get the car done.

**Why is the particular topic choosen ?**

Augmented reality is a technology that enables virtual images to be overlaid on views of the real world. In augmented Reality customer can feel and experience it anywhere or where he want. It's not necessary to have Real car with user we can experience in virtual car. Augmented reality systems have been used to allow experts to spatially collaborate with others at any other place in the world without traveling and thereby creating the experience of being virtually co-located. In this Augmented Reality Car Customiser App Demonstrated how a customer can have a look at a car model he is interested in & choose between various color options have a peek at the interior.

**Objective and scope of the Project –**

**Objectives of the Project -**

**Customization-**

Detect the Area for Placing the Car Model
Check the Car Model
Car Scale in Scale Out
Car Interior Experience
Change the Colour
Information About The Model of Car

## Scope of the Project –

Due to its capability to improve perception of reality, to support collaboration, a visual display of virtual objects, and to enable transitions between real and virtual environments, augmented reality (AR) technology can be used to create novel interfaces for face-to-face. In this new AR technology we are entered so this technology we gave many benifts. In this Augmented Reality Car App First Scanning the area where he want to place the virtual car. Then user can place the car model. User can move around the car feel the real car experience. User can check the features of the model . the user can customized the car there is a options to change the colors. User can open door and the check the interior features of the Model , user can check information about model of the car and change the scale car.

## Hardware and Software to be used –

### Hardware –

- RAM 8 GB
- Windows 8 & above
- Android Mobile AR Core (Supported Version 8 and Above).

### Software –

- Unity Engine 2.1.3
- ARCore Plugin
- Visual Studio 2019
- Android Studio 3.5.2

## Testing Technology –

Manual Testing is a type of software testing where testers manually execute test cases without using any automation tools. Manual Testing is the most primitive of all testing types and helps find bugs in the software system. Manual Testing requires more effort but is necessary to check automation feasibility.

**Conclusion** –

In augmented Reality customer can feel and experience it anywhere or where he want. It's not necessary to have Real car with user we can experience in virtual car. The interaction with augmented world is in two ways, using physical objects placeholder objects and by hand gestures. Created an Augmented Reality app to show how automobile companies can integrate AR in their app so that interested buyers can have a quick look on how various car models look in different colors.so buyer can quickly decide on which features he want in his new car and confirm his decision by going to the showroom just to take a test-drive this would save a lot of time.

**Further Enhancements** –

Improve interaction in augmented reality (for the field personnel)

Increase immersion and situational awareness for spatially distributed users in the virtual reality and augmented reality. Future work will focus on preparing a fully functional car simulation. We can add the feature of drive the car. We can create same feature in website so no need to download the app and add Buy Option so user can buy a car online

**Bibliography** –

https://www.researchgate.net/publication/281461043_AUGMENTED_REALITY_TECHNIQUES_FOR_VEHICLE_MAINTENANCE

https://www.researchgate.net/publication/322206805_AFFECTIVE_COMPUTING_AND_AUGMENTED_REALITY_FOR_CAR_DRIVING_SIMULATORS

# MAIN REPORT

# Objective & Scope of the Project

# Objective and scope of the Project

**Objectives of the Project –**

**Customization-**

Detect the Area for Placing the Car Model
Check the Car Model
Car Scale in Scale Out
Car Interior Experience
Change the Colour
Information About The Model of Car

**Scope of the Project** –

Due to its capability to improve perception of reality, to support collaboration, a visual display of virtual objects, and to enable transitions between real and virtual environments, augmented reality (AR) technology can be used to create novel interfaces for face-to-face. In this new AR technology we are entered so this technology we gave many benifts. In this Augmented Reality Car App First Scanning the area where he want to place the virtual car. Then user can place the car model. User can move around the car feel the real car experience. User can check the features of the model . the user can customized the car there is a options to change the colors. User can open door and the check the interior features of the Model , user can check information about model of the car and change the scale car.

# Theoretical Background

# Theoretical Background

## Application Domain :

Many of people doesn’t know about the car or the car features for that people can necessary to visit the showroom. In Before buying a car Customer needs to go car showroom to decide car. To travel to the showroom, he has to travel, he wastes time and takes time to get the car done.

Augmented Reality (AR) is the technology which superimposes an image onto a user’s view of the real world and enhances it with sound, touch, and even smell. It is a combination of the real scene viewed by the user and a virtual scene generated by the computer. AR is a technology which is going to blur the lines of reality.

Augmented Reality has moved beyond headsets and gaming and permeated into numerous industries. In general terms, Augmented Reality is increasingly being adopted for a variety of uses like assembly, maintenance, repair, education, training, retail showcasing and diagnostics.

## Why I tried Augmented Reality…

Augmented reality is a technology that enables virtual images to be overlaid on views of the real world. In augmented Reality customer can feel and experience it anywhere or where he want.

## Augmented Reality for Project

In this Augmented Reality Car Customizer App Demonstrated how a customer can have a look at a car model he is interested in & choose between various color options have a peek at the interior.

1. Detect the Area for Placing the Car Model:- Scans the ground first and then places the car object when the area is scanned.
2. Check the Car Model:- Can move object around the car and view it.
3. Car Scale in Scale Out:-Can scale in or scale out of car object.
4. Car Interior Experience:-By going inside the car object we can experience the interior
5. Change the Colour:- Can change the colour of a car object.
6. Information About The Model of Car:- Can see information of parts of car objects one by one.

## IT Domain :

Artificial Intelligence

Artificial intelligence (AI) is an area of computer science that emphasizes the creation of intelligent machines that work and react like humans. Some of the activities computers with artificial intelligence are designed for include:

Speech recognition
Learning
Planning
Problem solving

## How does AI work?

Artificial intelligence serves to simulate human intelligence through the power of computers. There are two ways that computers can gain artificial intelligence:

Machine learning – Machine learning utilizes massive amounts of data to train computers how to think. Computers can identify patterns and learn by examples. For example, machine learning keeps email inboxes spam free. The algorithms have been exposed to examples of emails classified as spam or not spam.

Deep learning – Deep learning takes machine learning to the next level by simulating the neurons in the brain. As computers and algorithms begin to process more data over a longer period of time, it continues to learn and adjust its algorithms similar to human learning. Self- driving cars utilize deep learning to recognize patterns, predict situations, and act accordingly.

## Application of Artificial Intelligence

Artificial intelligence is already ingrained into our lives. We may not even be aware of it, but it is out there serving to make our lives easier.

Artificial intelligence utilizes computer processing to complete tasks that normally require human intelligence. However, it performs actions with a greater level of accuracy and much more quickly.

## Artificial Intelligence in Project

Data is the core of artificial intelligence. With data collected at various cycles of the construction project across many different projects in construction firms, this provides valuable learning information for artificial intelligence applications. Artificial intelligence serves as a helpful tool for every phase of the construction project.

## Machine Learning

Machine learning is an application of artificial intelligence (AI) that provides systems the ability to automatically learn and improve from experience without being explicitly programmed. Machine learning focuses on the development of computer programs that can access data and use it learn for themselves

The process of learning begins with observations or data, such as examples, direct experience, or instruction, in order to look for patterns in data and make better decisions in the future based on the examples that we provide. The primary aim is to allow the computers learn automatically without human intervention or assistance and adjust actions accordingly.

I just using NEURAL NETWORKS in Machine Learning method:

We have a very large number of state-action pairs in mixed reality as feature extraction is important part as we have to know where and which place has been shared by the user and how they differ from each other using multilayer neural network for such applications is efficient choice. Tensorflow Lite handles all neural networking tasks efficiently and without any hustle we can integrate it in our model

Neurons are used for fitting linear forms, e.g., y = a + bi where i is the input (the frames in our case). Also called adrenaline rule or Widrow-Hoff rule.

Neurons can also be used by fitting the Q-function in a piecewise manner, where a linear fit is introduced in every piece.

**Algorithm**

Step 1: Initialize the weights of the neural network.

Step 2: Compute the output o using Σ where

w(j) is the jth weight of neuron and x(j) is the jth input.

Step 3: Update each w(i) for i = 0,1,...,k using:

$w(i) \leftarrow w(i) + \mu[target - output]x(i)$

Step 4: Increment iteration while I < iter_max

In this project suppose we are giving input as any word. For example :-Detect physical objects Use deep learning neural network to detect physical objects from camera raw input. Anchor the specific pose in the real world and keep the virtual content at the same location.

# Survey of the Literature

# Survey of the Literature

**Object detection:**

Augmented reality (AR) and machine learning (ML) are two leading-edge technologies. AR brings virtual objects into the real world by overlaying digital content above physical objects. ML, on the other hand, helps the program to recognize physical objects of the real world. By combining these two technologies, we can create some innovative projects.

**Detect physical objects**: Use deep learning neural network to detect physical objects from camera raw input.
**Classification**: The ML model pre-trained. It can recognize 80 object categories.
**Localization**: Put a bounding box around the detected object. Localized it in 2D space.

Then use Google ARCore to handle the AR parts of the demo:

**Plane detection**: Detect horizontal plane, localize the physical object in 3D by ray cast from 2D to the detected plane.
**Motion tracking**: Track the camera movement.
**Anchors**: Anchor the specific pose in the real world and keep the virtual content at the same location.

# Research Analysis & Design

# Research Analysis & Design

Creating ArCar App

**Get the ARCore SDK for Unity**

1.Download ARCore SDK for Unity 1.16.0 or later

Create a new project and import the SDK

1. Open Unity and create a new 3D project.
2. Unity 2019 only: Select Window > Package Manager and install the following packages:

- Multiplayer HLAPI (required by the CloudAnchors sample)
- XR Legacy Input Helpers (required by Instant Preview, which uses the TrackedPoseDriver)

3. Import the ARCore SDK for Unity:
   a. Select Assets > Import Package > Custom Package.
   b. Select the arcore-unity-sdk-1.16.0.unitypackage that you downloaded.
   c. In the Importing Package dialog, make sure that all package options are selected and click Import.

Open the sample scene

1. In the Unity Project window, you can find the HelloAR sample in: Assets/GoogleARCore/Examples/HelloAR/Scenes/.

Configure project settings

1. Go to File > Build Settings to open the Build Settings window.
2. Select Android and click Switch Platform.
3. In the Build Settings window, click Player Settings.

In the Settings window, configure the following:

| **Setting** | **Value** |
| --- | --- |
| Player Settings > Other Settings > Rendering | Uncheck Auto Graphics API<br>If Vulkan is listed under Graphics APIs, remove it. |
| Player Settings > Other Settings > Package Name | Create a unique app ID using a Java package name format.<br>For example, use com.example.helloAR |
| Player Settings > Other Settings > Minimum API Level | Android 7.0 'Nougat' (API Level 24) or higher (For AR Optional apps, the Minimum API level is 14.) |
| Player Settings > XR Settings > ARCore Supported | Enable |

Build and run the sample app

1. Enable developer options and USB debugging on your device.
2. Connect your device to your development machine.
3. In the Unity Build Settings window, click Build and Run. Unity builds your project into an Android APK, installs it on your device, and launches it.
4. Move your device around until ARCore starts detecting and visualizing planes.
5. Tap a plane to put an Andy Android object on it.
6. (Optional) Use Android logcat to view log messages or Android Device Monitor to analyze the device more comprehensively.

Next Steps

- Step through the HelloAR code in the HelloAR Sample App Tour.
- Learn how to Enable ARCore in your app.
- Use Augmented Images to build apps that can respond to 2D images, such as posters or logos, in the user's environment.
- Use Cloud Anchors to create shared AR experiences across iOS and Android users.
- Review Runtime Considerations.

Review Design Guidelines.

# System Planning

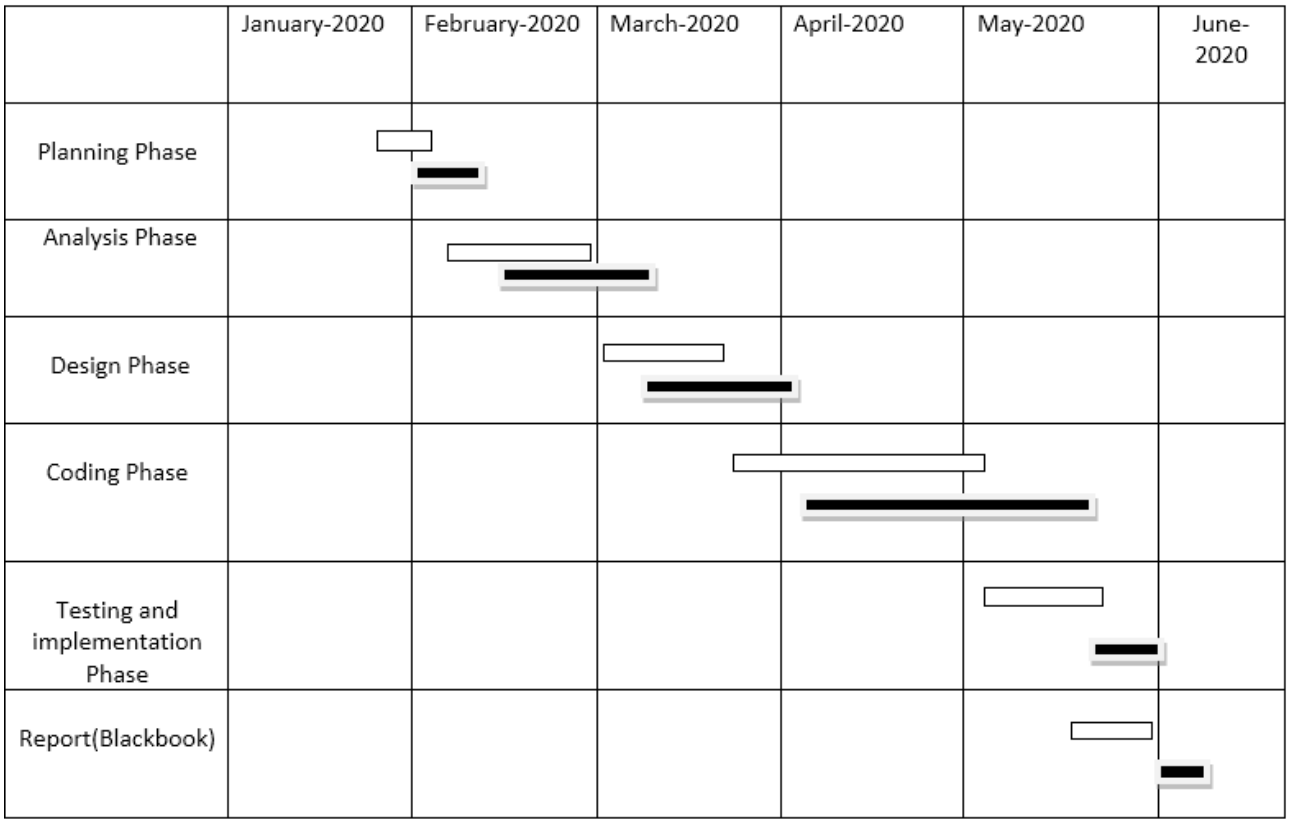
# System Planning

## Gantt chart

The Gantt chart is a horizontal bar chart developed by a production control tool in 1918 by Henry. Gantt, an american engineer and social scientist frequency used in project management . A Gantt chart provides a graphical illustration of the schedule that help to plan, co-ordinate & track specific task inproject.

Gantt chart may be simple version created using project management application such as microsoft project or excel. A Gantt chart is constructed with horizontal axis representing total time spam of project , breakdown into increments. Example. days, weeks, months, up the project horizontal bars of varying lengths represent the sequences, timings and timespam for each task.

| ID | TASK NAME | START date | FINISH date | DURATION |
|---|---|---|---|---|
| **1** | **Planning phase** | 1/2/2020 | 12/2/2020 | 12d |
| 1.1 | Define the problem | 1/2/2020 | 12/2/2020 | 12d |
| 1.1.1 | Meet the user | 1/2/2020 | 8/2/2020 | 8d |
| 1.1.2 | Determine the scope | 1/2/2020 | 10/2/2020 | 10d |
| 1.1.3 | Define the statement of need | 7/2/2020 | 12/2/2020 | 6d |
| 1.2 | Confirm project feasibility | 7/2/2020 | 11/2/2020 | 5d |
| 1.3 | Produce project schedule | 10/2/2020 | 11/2/2020 | 2d |
| 1.4 | Launch the project | 10/2/2020 | 12/2/2020 | 3d |
| **2** | **Analysis phase** | 14/2/2020 | 08/3/2020 | 24d |
| 2.1 | Gather information | 14/2/2020 | 22/2/2020 | 9d |
| 2.2 | Define system requirement | 17/2/2020 | 24/2/2020 | 8d |
| 2.3 | Prototype for feasibility & Discovery | 25/2/2020 | 08/3/2020 | 13d |
| **3** | **Design phase** | 09/3/2020 | 6/4/2020 | 29d |
| 3.1 | Design application architecture | 09/3/2020 | 18/3/2020 | 11d |
| 3.2 | Design system interface | 9/3/2020 | 3/4/2020 | 27d |
| 3.3 | Design & integrate features | 3/4/2020 | 6/4/2020 | 4d |
| **4** | **Coding phase** | 6/4/2020 | 25/5/2020 | 50d |
| 4.1 | Writing code for modules | 6/4/2020 | 25/5/2020 | 50d |
| 4.2 | Integrating modules | 15/4/2020 | 24/5/2020 | 40d |
| **5** | **Testing and implementation** | 24/5/2020 | 01/6/2020 | 6d |
| **6** | **Report (Black Book)** | 1/6/2020 | 05/6/2020 | 5d |




Planned Date:

Actual Date:

# Details of Hardware & Software used

# Software –

Windows 2008 and above

- Unity Engine 2019.2.13

  Unity is, in short, a closed-source, cross-platform game development application. You create your game by manipulating objects in 3D and attaching various components to them. Even 2D games must be manipulated in 2D/3D. Scripts are written in C# (recommended), Boo or UnityScript (some peoples mistakenly call it JavaScript) and attached to 3D objects as components.

  That said, once you've created a game with Unity, deployment is a cinch. With a couple of clicks, you can export your game to mobile, desktop and/or web (web currently requires the Unity player app to be installed). If you have the right license, you can even deploy to gaming consoles like Xbox, Playstation and Wii.

- ARCore Plugin

  ARCore is a software development kit developed by Google that allows for augmented reality applications to be built.

  ARCore uses three key technologies to integrate virtual content with the real environment:

1. **Motion Tracking:** it allows the phone to understand its position relative to the world.

2. **Environmental understanding:** It allows the phone to detect the size and location of all type of surfaces, vertical, horizontal and angled.

3. **Light Estimation:** it allows the phone to estimate the environment's current lighting conditions

- Visual Studio

  **Microsoft Visual Studio** is an integrated development environment (IDE) from Microsoft. It is used to develop computer programs, as well as websites, web apps, web services and mobile apps. Visual Studio uses Microsoft software development platforms such as Windows API, Windows Forms, Windows Presentation Foundation, Windows Store and Microsoft Silverlight. It can produce both native code and managed code.

- Android Studio 3.5.2

  Android Studio is the official integrated development environment (IDE) for Google's Android operating system, built on JetBrains' IntelliJ IDEA software and designed specifically for Android development. It is available for download on Windows, macOS and Linux based operating systems.

# Hardware–

- RAM 8Gb

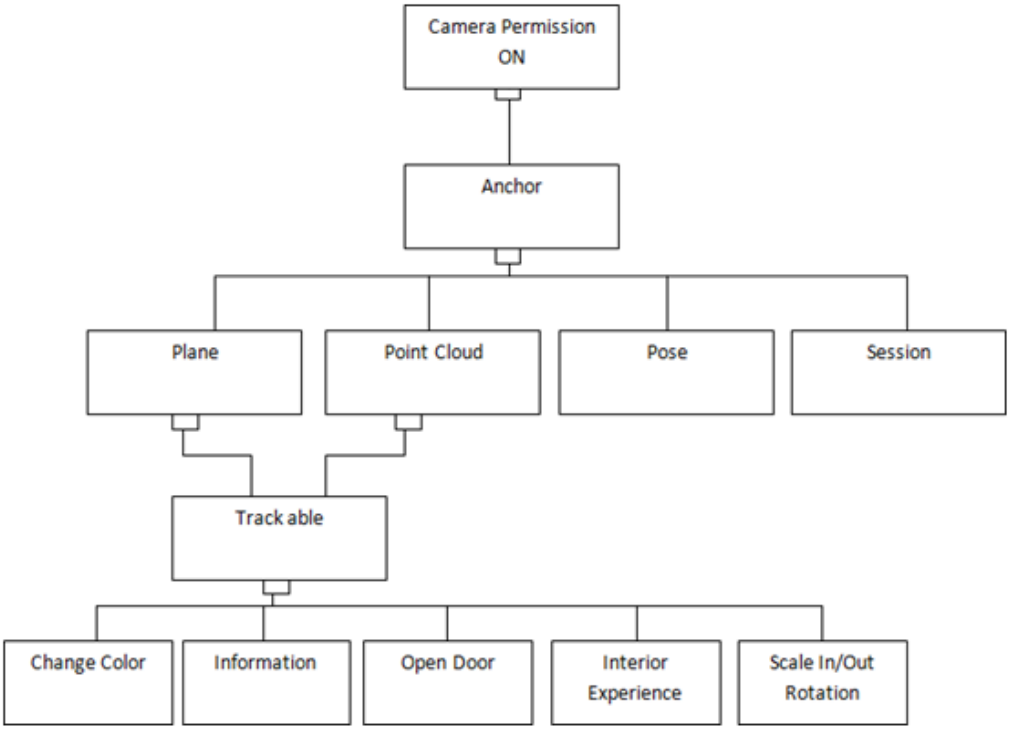
Android Mobile AR Core (Supported Version 8 and Above).

# Detailed Life Cycle of the Project

# Detailed Life Cycle of the Project

## Work Breakdown Structure

A work breakdown structure (WBS) is a key project deliverable that organizes the team's work into manageable sections. The Project Management Body of Knowledge defines the work breakdown structure as a "deliverable oriented hierarchical decomposition of the work to be executed by the project team."



# Process Model

## Iterative Model

Software Development Life Cycle (SDLC) is extremely vast and full of various development and testing activities, methodologies, techniques, tools, and more. It involves intense planning and management, calculation and preparation. It is only after combining all these efforts of the software engineers that a software or application is successfully developed. **Iterative Model** is too a part of Software Development Life Cycle. It is a particular implementation of a software development life cycle that focuses on an initial, simplified implementation, which then progressively gains more complexity and a broader feature set until the final system is complete. In short, iterative development is a way of breaking down the software development of a large application into smaller pieces.

## What is Iterative Model in Software Development Life Cycle (SDLC)?:

An iterative life cycle model does not start with a full specification of requirements. In this model, the development begins by specifying and implementing just part of the software, which is then reviewed in order to identify further requirements. Moreover, in iterative model, the iterative process starts with a simple implementation of a small set of the software requirements, which iteratively enhances the evolving versions until the complete system is implemented and ready to be deployed. Each release of Iterative Model is developed in a specific and fixed time period, which is called iteration.

## Process of Iterative Model:

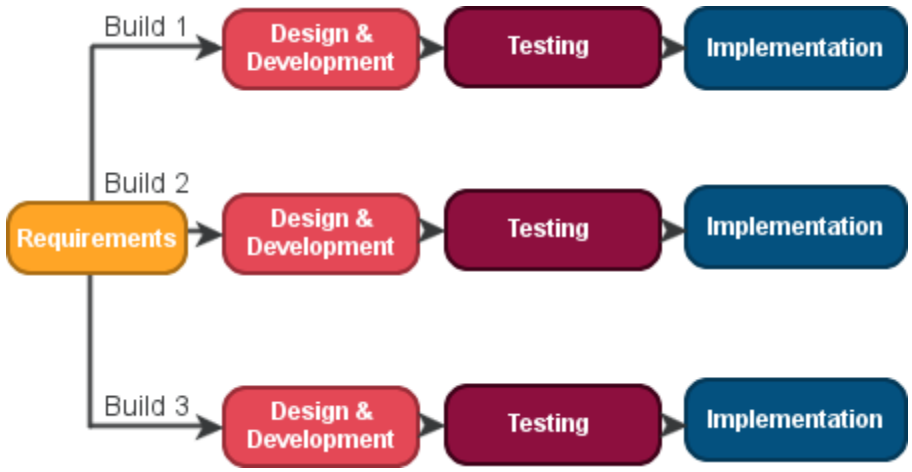
I am using Iterative Model in my project because the process of Iterative Model is cyclic, unlike the more traditional models that focus on a rigorous step-by-step process of development. In this process, once the initial planning is complete, a handful of phases are repeated again and again, with the completion of each cycle incrementally improving and iterating on the software.

**Uses of Iterative Model in the Project**

I have followed Requirements of the complete system are clearly defined and understood.

Major requirements must be defined; however, some details can evolve with time. When the project is big.

**Diagram :**



Other phases of the iterative model are described below:

1. **Planning Phase:** This is the first stage of the iterative model, where proper planning is done by the team, which helps them in mapping out the specifications documents, establish software or hardware requirements and generally prepare for the upcoming stages of the cycle.

2. **Analysis and Design Phase:** Once the planning is complete for the cycle, an analysis is performed to point out the appropriate business logic, database models and to know any other requirements of this particular stage. Moreover, the design stage also occurs in this phase of iterative model, where the technical requirements are established that will be utilized in order to meet the need of analysis stage.
3. **Implementation Phase:** This is the third and the most important phase of the iterative model. Here, the actual implementation and coding process is executed. All planning, specification, and design documents up to this point are coded and implemented into this initial iteration of the project.

4. **Testing Phase:** After the current build iteration is coded and implemented, testing is initiated in the cycle to identify and locate any potential bugs or issues that may have been in the software.

5. **Evaluation Phase:** The final phase of the Iterative life cycle is the evaluation phase, where the entire team along with the client, examine the status of the project and validate whether it is as per the suggested requirements.

Advantages of Iterative Model:

- Some working functionality can be developed and early in the software development life cycle (SDLC).
- It is easily adaptable to the ever changing needs of the project as well as the client.
- It is more cost effective to change the scope or requirements in Iterative model.
- Parallel development can be planned.
- Testing and debugging during smaller iteration is easy.
- Risks are identified and resolved during iteration; and each iteration is an easily managed.
- In iterative model less time is spent on documenting and more time is given for designing.

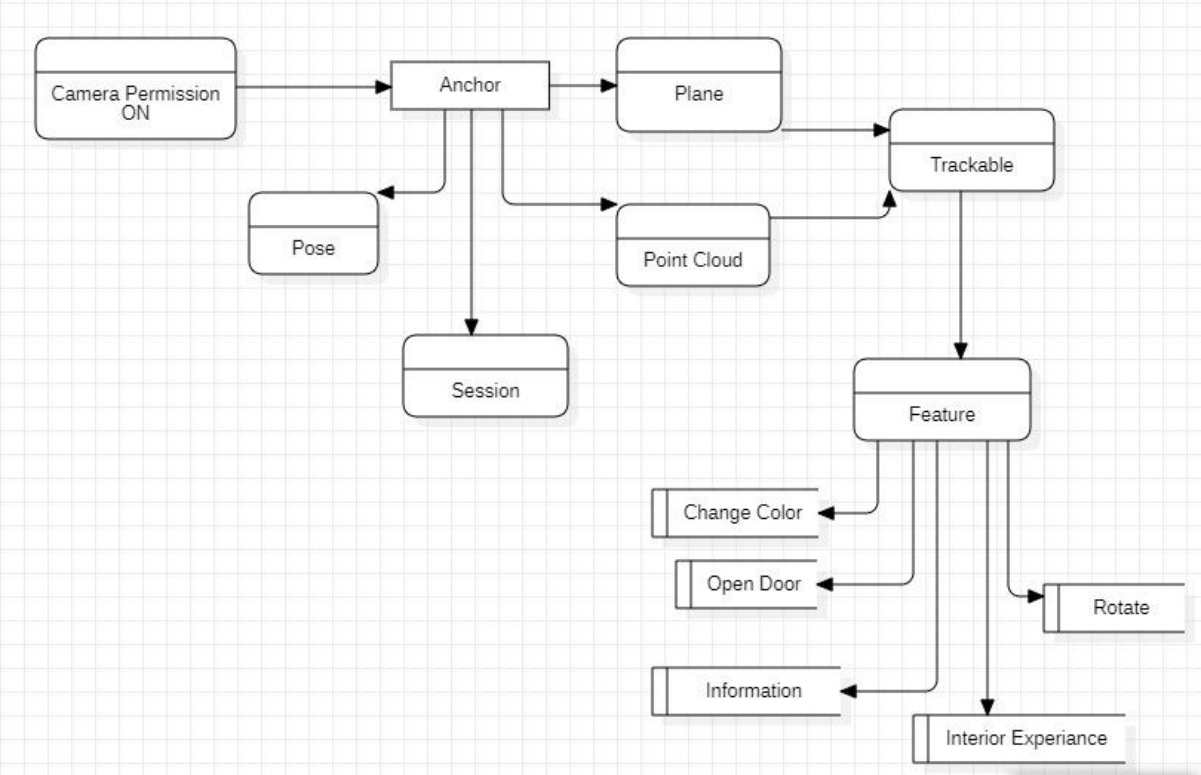
Disadvantages of Iterative Model:

- More resources may be required.
- Although cost of change is lesser, but it is not very suitable for changing requirements.
- More management attention is required.
- It is not suitable for smaller projects.
- Highly skilled resources are required for skill analysis.
- Project progress is highly dependent upon the risk analysis phase.
- Defining increments may require definition of the complete system.

# DataFlow Diagram

# Data Flow Diagram

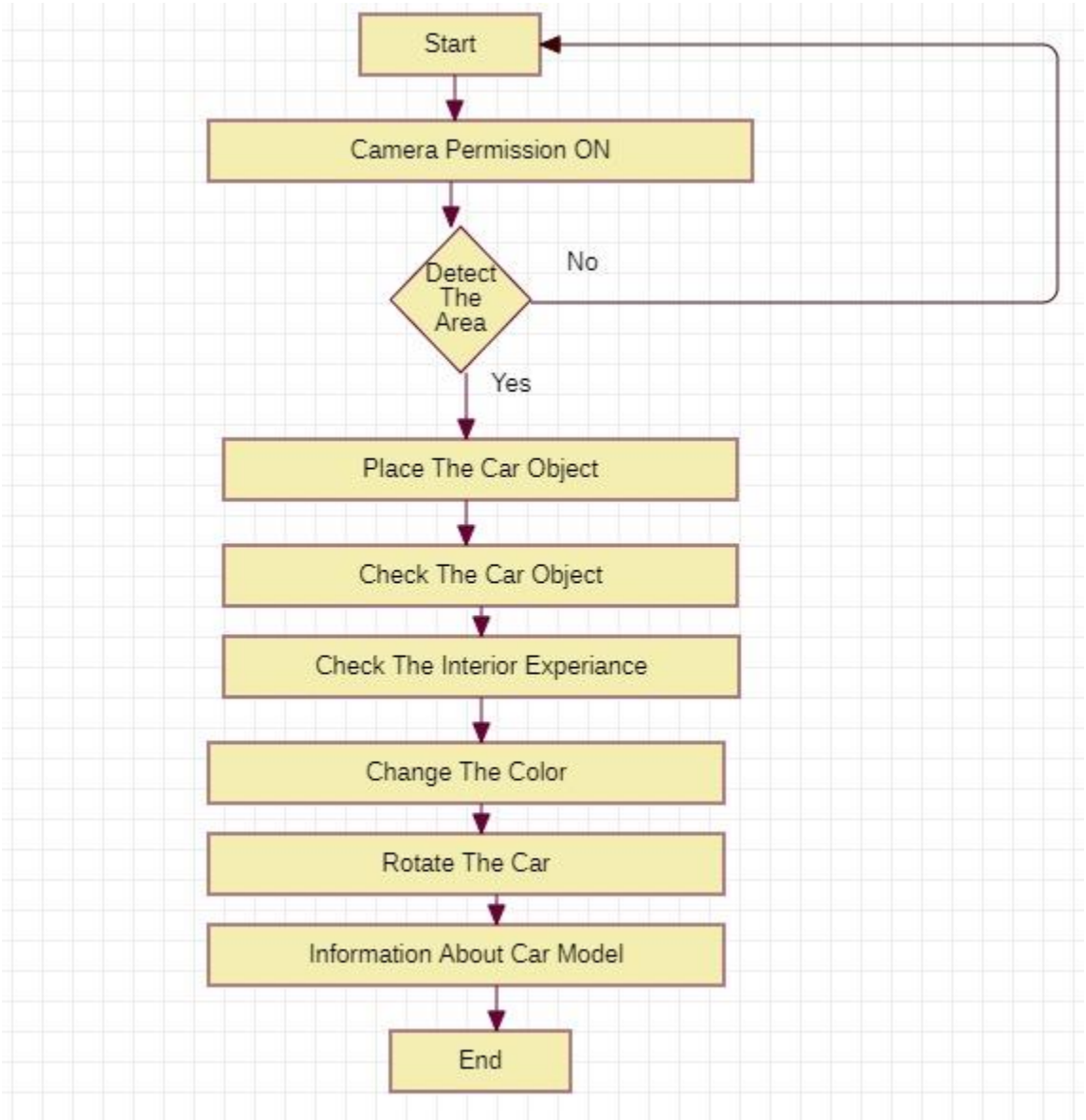
A **data flow diagram** (**DFD**) is a graphical representation of the "flow" of data through an information system, modeling its *process* aspects. A DFD is often used as a preliminary step to create an overview of the system without going into great detail, which can later be elaborated. DFDs can also be used for the visualization of data processing.



# Flow Diagram

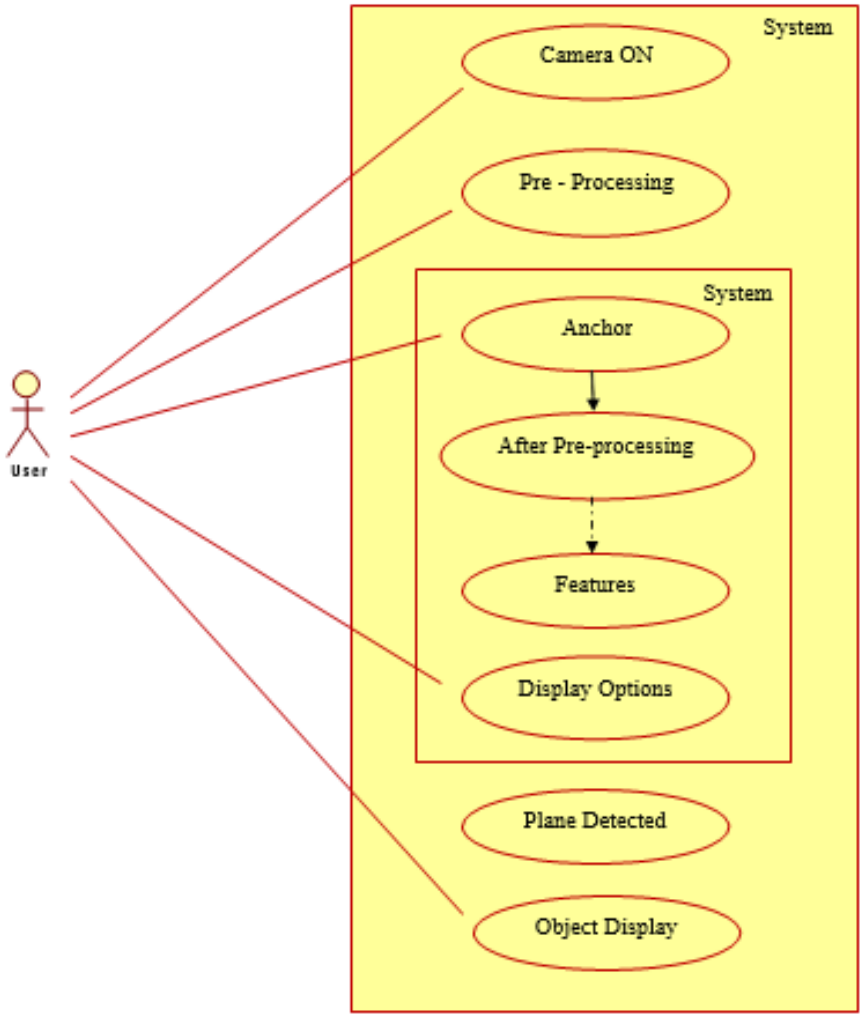
## 1. FLOW DIAGRAM

**Flow diagram** is a collective term for a diagram representing a flow or set of dynamic relationships in a system. The term flow diagram is also used as a synonym for flowchart, and sometimes as a counterpart of the flowchart.



## 2.USE CASE DIAGRAM

A use case diagram at its simplest is a representation of a user's interaction with the system that shows the relationship between the user and the different use cases in which the user is involved. A use case diagram can identify the different types of users of a system and the different use cases and will often be accompanied by other types of diagrams as well.



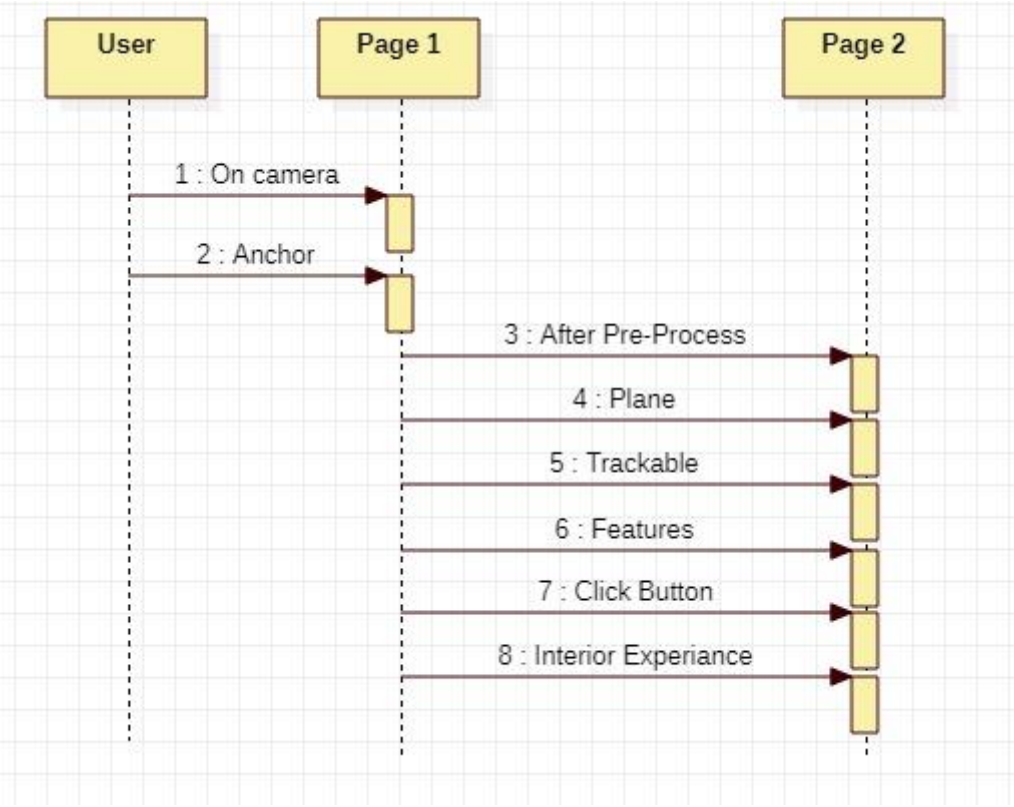
## 3.SEQUENCE DIAGRAM

Shows sequence of interactions between objects and of events in a single use case. Focuses on message details.

Used more frequently in industry.

Actor represented by stick figure- person (or role) that "interacts" with by entering input data and receiving output data Objects notation is rectangle with name of object underlined - shows individual object and not class of all similar objects.

Lifetime is vertical line under object or actor to show passage of time for object. Message use arrows to show messages sent or received by actor or system.



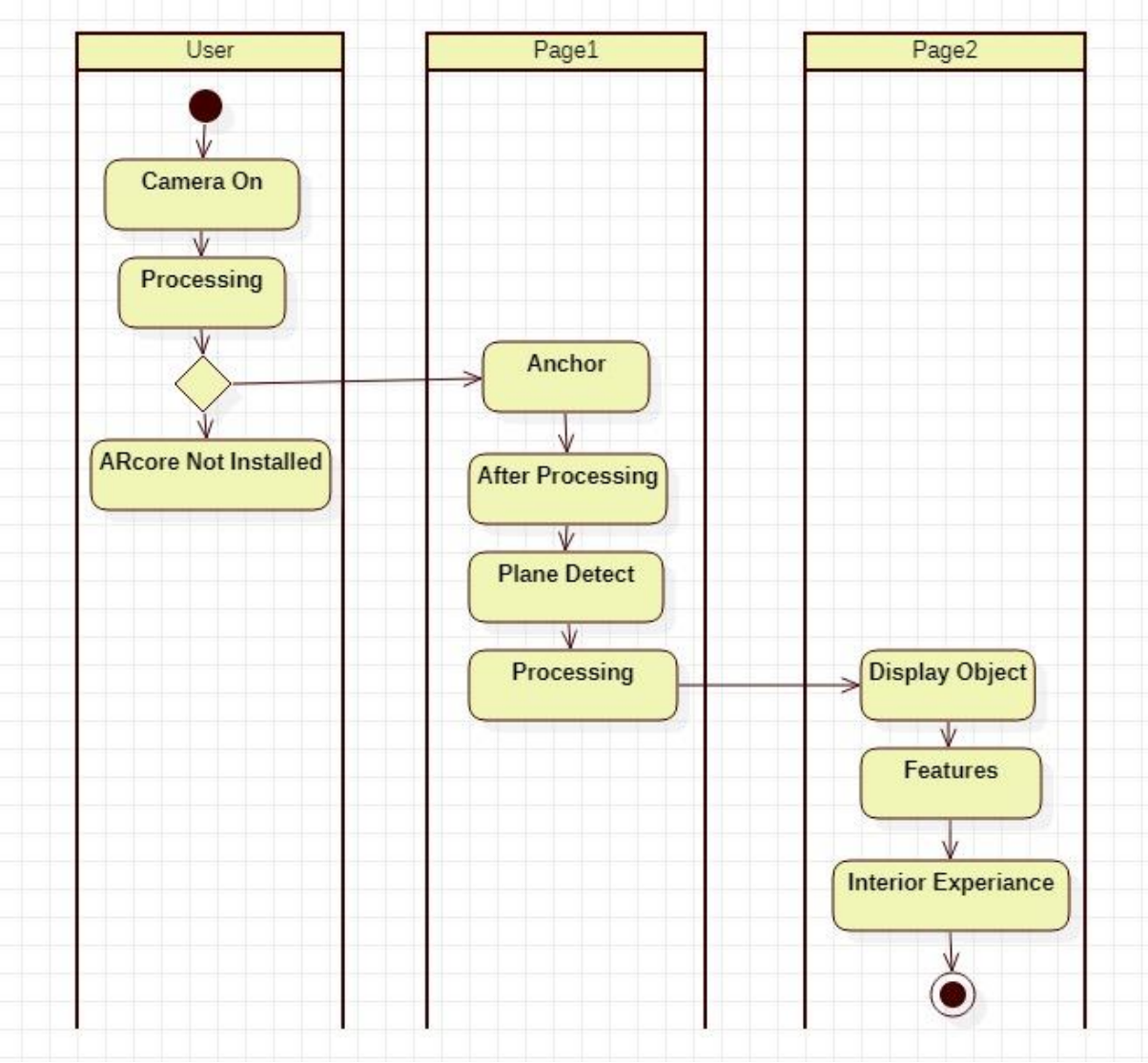
## 4.Activity Diagram

An Activity Diagram has the following features:

It resembles a flow chart.

It illustrate the dynamic nature of a system by modeling the flow of control from activity.

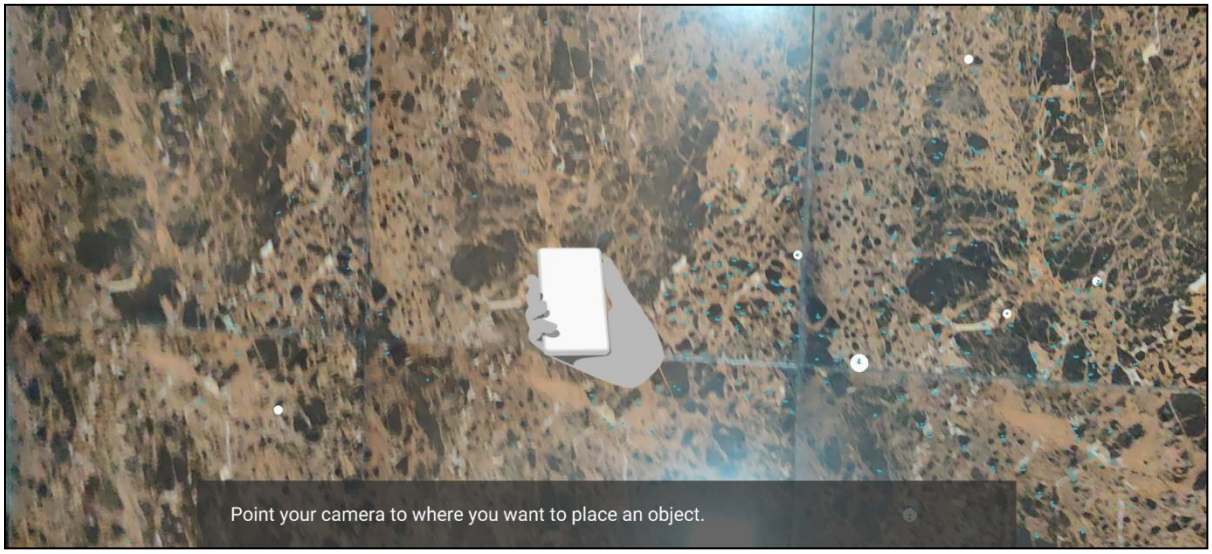
An activity can represent an operation on some class in the system that results in a change in the state of the system.



# Input Output Screen Design

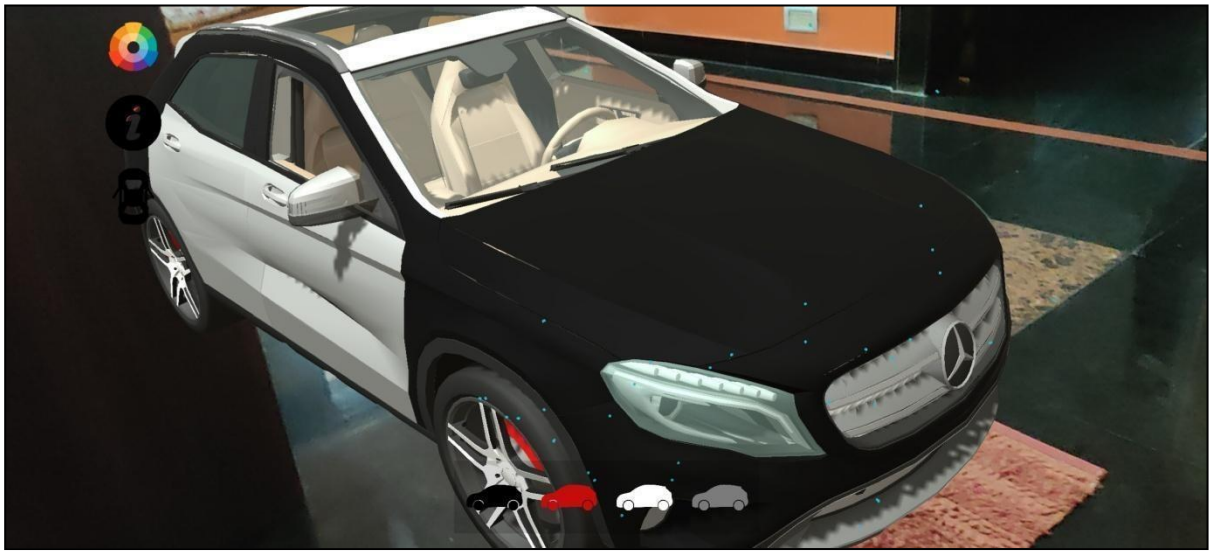
# Input Output Screen Design

Ground Scanning:-

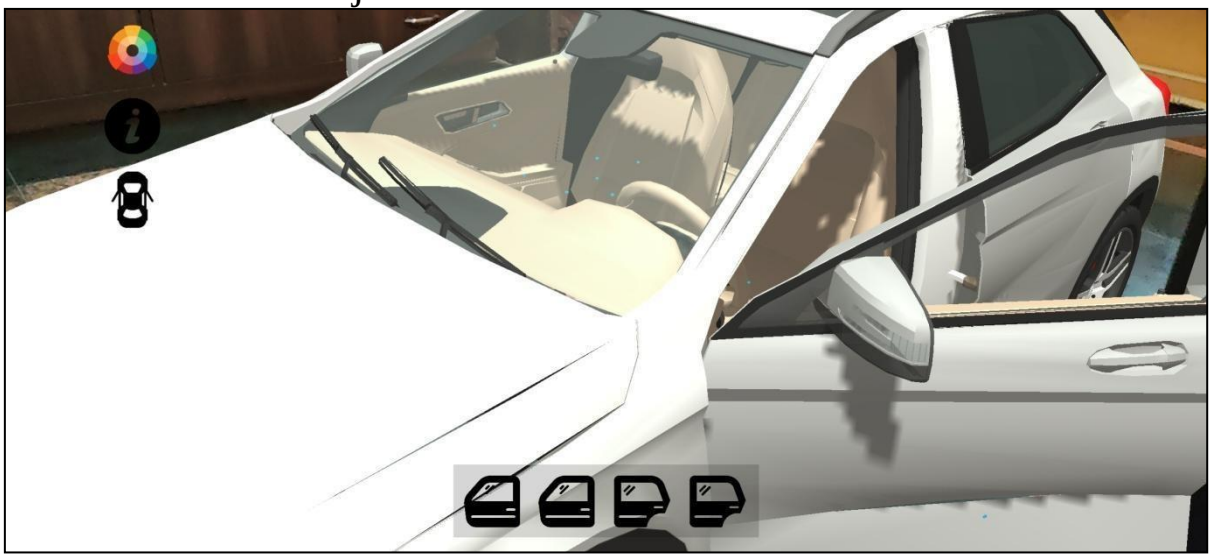



Place Car Object:-

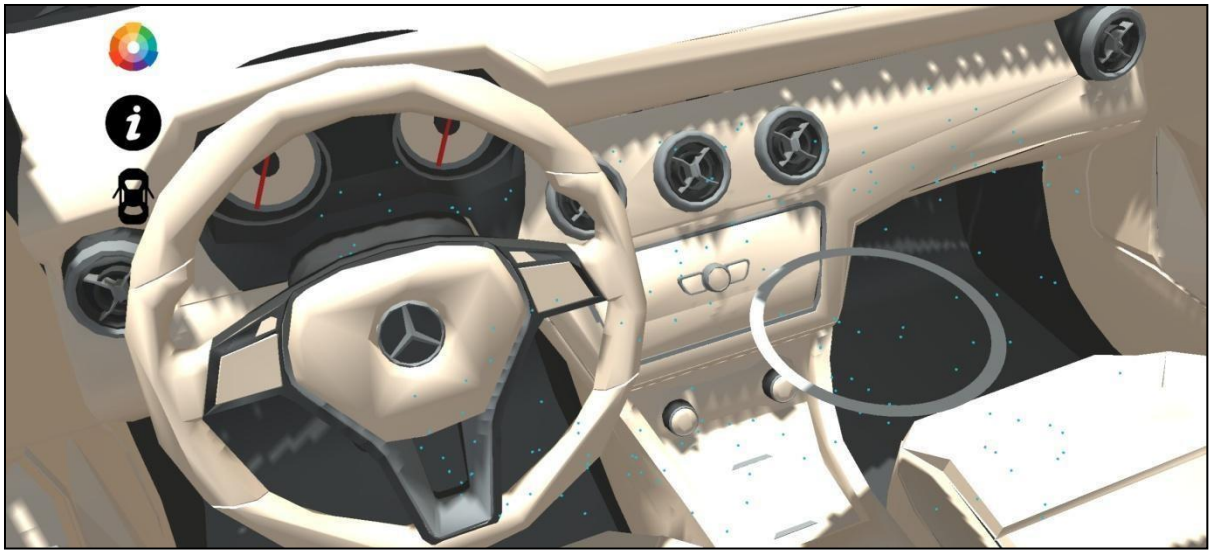
Change Color:-

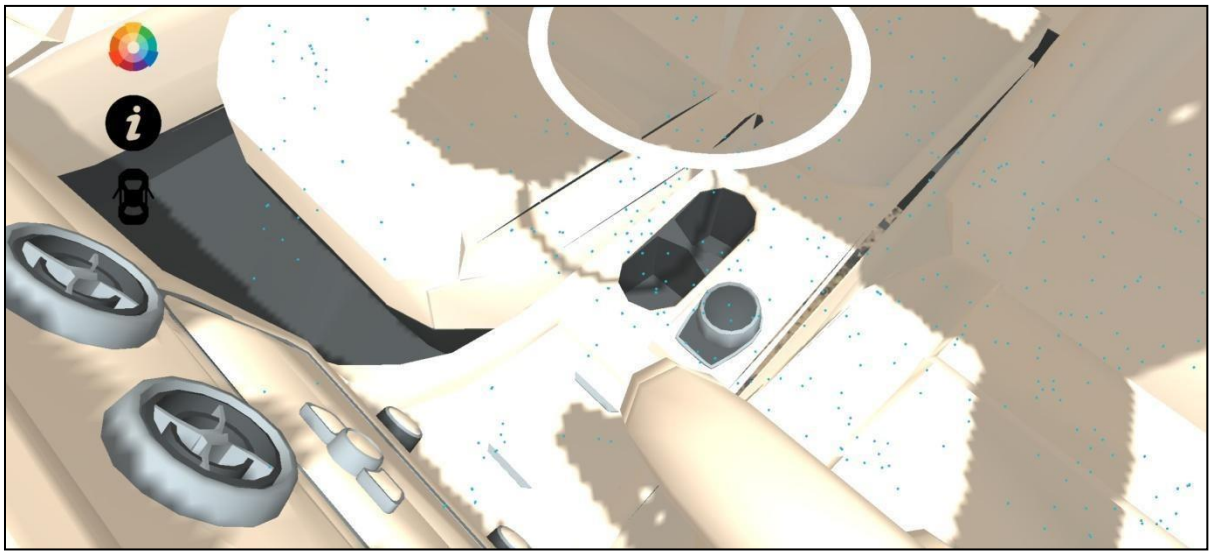


Interact With Car object:-



Interior Experience:-





Car Rotate zoom in zoom out:-

Car part Information:-


# Methodology Used For Testing

# Manual Testing

**Manual Testing** is a process of finding out the defects or bugs in a software program. In this method the tester plays an important role of end user and verifies that all the features of the application are working correctly. **The tester manually executes test cases without using any automation tools**. The tester prepares a test plan document which describes the detailed and systematic approach to testing of software applications. Test cases are planned to cover almost 100% of the software application. As manual testing involves complete test cases it is a time consuming test.

The differences between actual and desired results are treated as defects. The defects are then fixed by the developer of software application. The tester retests the defects to ensure that defects are fixed. The goal of Manual testing is to ensure that application is defect & error free and is working fine to provide good quality work to customers.

Procedure of Manual Testing

- Requirement Analysis
- Test Plan Creation
- Test case Creation
- Test case Execution
- Defect Logging
- Defect Fix & Re-Verification

Following are the testing techniques that are performed manually during the test life cycle:

- Acceptance Testing
- White Box Testing
- Black Box Testing
- Unit Testing
- System Testing
- Integration Testing

## SYSTEM TESTING

System Testing is a series of different tests whose primary purpose is to fully exercise the computer-based system. Although each test has different purpose, all work to verify that the system elements have been properly integrated and performs the allocated function.

## WHITE BOX TESTING

White Box Testing, sometimes also called 'Glass-Box' Testing, is predicted on close examination of procedural details. Logical paths through the software are tested by providing test cases that exercise using this testing that uses the control structure of procedural design to derive test cases.

## BLACK BOX TESTING

Black Box Testing also called "Behavioral Testing", focuses on the functional requirements of the software. Black Box Testing is used to demonstrate that software functions are operational, that input is properly accepted and output is correctly produced and that the integrity of external information is maintained.

## UNIT TESTING

Unit testing is defined as the smallest collection of code, which can be tested. Typical this would be a source file, a package or a non-trivial objects class. Unit Testing Is primarily focused on the implementation- Does the code implement what the designer intended!

For each conditional statement, is the condition correct? Unit Testing was carried out to test the validity and functionality of the following areas of code:

The combo-boxes, which were used to retrieve data from the database based on certain conditions, were tested to make sure that the test conditions specified retrieved the correct data.

Each piece of code where database connection was used, was tested to ensure that it was delivering the correct results.

The data validation statement were checked to make sure that they did not allowed any bogus or invalid data to pass into the systems.

## INTEGRATION TESTING

The individual components are combined with other components to make sure that necessary communication links and data sharing occur properly. There are 3 basic Integration Test methods:

## BOTTOM-UP

The Bottom-up Testing involves individual testing of each module using a driver routine that calls the module and provides with needed resources. It is a more intuitive approach to testing that usually finds error in critical modules. However, in a new system many modules must be integrated to produce a system level behavior, thus interface errors surface rate in the process.

## TOP-DOWN

The Top-down testing fits prototyping environment that establishes an initial skeleton that fills in individual modules is completed. The method tends itself to more structured organization that plans out the entire test process. Modules can be found later that you would like. What all this implies is that a combination of low-level Top-Down modules provides an early working program that can give management and the users more confidence in results.

# Test Cases

# Test Cases

| Sr. no | Test case name | Test case Description | Input | Expected Output |
|---|---|---|---|---|
| 1. | Camera Permission on | Camera on for scan the plane | Allowing camera permissi on | Camera on |
| 2. | Scaning plane | Scaninng plane for place the object on ground | Searchin g plane area | Popout the scanning area |
| 3. | Place object on plane | Place the object where user want to place it. | Tap on screen to place the object | Place the objet on plane area |
| 4. | Changing color | Changing color of object to custumised | Tap on button to change color | Object change the color |
| 5. | Display information about car | To see the information about the car | Tap on button to see informati on | Display the information. |
| 6. | Check animation of door | User can interact with object. | Tap on button to open the doors. | Opening the doors. |
| 7. | Scale in Scale out | To scale the object. | Pinch to scale in scale out | Object get scale in scale out. |
| 8. | Rotation | To rotate the object. | Swipe to rotate. | Object get rotate. |
| 9. | Interior experien ce | To get the interior experience | Inside to object To see | Interior experience get |

# Test Reports

# Test Report

| Sr. no | Test case name | Test case Description | Input | Expected Output | Actual Output | Remar k |
|---|---|---|---|---|---|---|
| 1 | Camera Permission on | Camera on for scan the plane | Allowing camera permissio n | Camera on | Not get permissio n option | Failed |
| 2 | Camera Permission on | Camera on for scan the plane | Allowing camera permissio n | Camera on | get permiss ion option and camera on. | Passed |
| 3 | Scaning plane | Scaninng plane for place the object on ground | Searching plane area | Popout the scanning area | Not detec ted plan e | Failed |
| 4 | Scaning plane | Scaninng plane for place the object on ground | Searching plane area | Popout the scanning area | Plan e detec ted | Passed |
| 5 | Place object on plane | Place the object where user want to place it. | Tap on screen to place the object | Place the objet on plane area | Object not place properl y | Failed |
| 6 | Place object on plane | Place the object where user want to place it. | Tap on screen to place the object. | Place the object on plane area | Object place prop erly | Passed |
| 7 | Changing color | Changing color of object to custumised | Tap on button to change color | Object change the color | Object change the | Passed |

| | | | | | | |
|---|---|---|---|---|---|---|
| | | | | | color | |
| 8 | Display information about car | To see the information about the car | Tap on button to see informatio n | Display the information. | Display informat ion | Passed |
| 9 | Check animation of door | User can interact with object. | Tap on button to open the doors. | Opening the door | Doo r is ope n | Passed |
| 10 | Scale in Scale out | To scale the object | Pinch to scale in scale out | Object get scale in scale out. | Not scaling. | Failed |
| 11 | Scale in Scale out | To scale the object | Pinch to scale in scale out | Object get scale in scale out. | Sca led obj ect | Passed |
| 12 | Rotation | To rotate the object. | Swipe to rotate. | Object get rotate. | Not rotate | Failed |
| 13 | Rotation | To rotate the object. | Swipe to rotate. | Object get rotate. | Rotated | Passed |
| 14 | Interior experience | To get the interior experience | Inside to object To see | Interior experience get | Interior experie nce get | Passed |

# Code of the Project

**Arcore Session.cs:-**

```
//_____________________________________________________
// <copyright file="ARCoreSession.cs" company="Google">
//
// Copyright  Google LLC. All Rights Reserved.
//
// Licensed under the Apache License, Version 2.0 (the "License");
// you may not use this file except in compliance with the License.
// You may obtain a copy of the License at
//
// http://www.apache.org/licenses/LICENSE-2.0
//
// Unless required by applicable law or agreed to in writing, software
// distributed under the License is distributed on an "AS IS" BASIS,
// WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
// See the License for the specific language governing permissions and
// limitations under the License.
//
// </copyright>
//_____________________________________________________

namespace GoogleARCore
{
    using
    System.Collections.
    Generic; using
    GoogleARCoreInte
    rnal; using
    UnityEngine;

    /// <summary>
    /// A component that manages the ARCore Session in a Unity scene.
    /// </summary>
    [HelpURL("https://developers.google.com/ar/reference/unity/class/GoogleARCore/AR
    CoreSession")] public class ARCoreSession : MonoBehaviour
    {
        /// <summary>
        /// The direction of the device camera used by the session.
        /// </summary>
        /// <remarks>
        /// Note that changing this value will trigger a re-initialization of session. ARCore
        /// tracking data (e.g. Trackables) are not shared between cameras.
        /// </remarks>
        [Tooltip("The direction of the device camera used by the session.")]
        public DeviceCameraDirection DeviceCameraDirection =
        DeviceCameraDirection.BackFacing;

        /// <summary>
        /// A scriptable object specifying the ARCore session configuration.
        /// </summary>
        [Tooltip("A scriptable object specifying the ARCore session
        configuration.")] public ARCoreSessionConfig
```

```
SessionConfig;

/// <summary>
/// The camera configuration filter object that defines the set of
/// properties desired or required by the app to run.
/// </summary>
[Tooltip("Configuration options to select the camera mode and features.")]
public ARCoreCameraConfigFilter CameraConfigFilter;

private OnChooseCameraConfigurationDelegate m_OnChooseCameraConfiguration;

/// <summary>
/// Selects a camera configuration for the ARCore session being resumed.
/// </summary>
/// <param name="supportedConfigurations">
/// A list of supported camera configurations. The size is dependent on
/// <see cref="ARCoreSession.CameraConfigFilter"/> settings.
/// The GPU texture resolutions are the same in all configs.
/// Currently, most devices provide GPU texture resolution of 1920 x 1080,
/// but devices might provide higher or lower resolution textures, depending
/// on device capabilities.
/// The CPU image resolutions returned are VGA, 720p, and a resolution matching the GPU
/// texture, typically the native resolution of the device.</param>
/// <returns>The index of the camera configuration in <c>supportedConfigurations</c> to
be
/// used for the ARCore session. If the return value is not a valid index (e.g. the value
/// -1), then no camera configuration will be set and the ARCore session will usethe
/// previously selected camera configuration or a default configuration if noprevious
/// selection exists.</returns>
public delegate int OnChooseCameraConfigurationDelegate(
   List<CameraConfig> supportedConfigurations);

/// <summary>
/// Unity Awake.
/// </summary>
[SuppressMemoryAllocationError(Reason = "Could create new
LifecycleManager")] public virtual void Awake()
{
   if (SessionConfig != null &&
      SessionConfig.LightEstimationMode !=
      LightEstimationMode.Disabled &&
      Object.FindObjectsOfType<EnvironmentalLight>().Length
      == 0)
   {
      Debug.Log("Light Estimation may not work properly when
         EnvironmentalLight is not" + " attached to the scene.");
   }

   LifecycleManager.Instance.CreateSession(this);
}

/// <summary>
/// Unity OnDestroy.
/// </summary>
```

```
[SuppressMemory
AllocationError(
   IsWarning = true, Reason = "Requires further
investigation.")] public virtual void OnDestroy()
{
   LifecycleManager.Instance.ResetSession();
}

/// <summary>
/// Unity OnEnable.
/// </summary>
[SuppressMemory
AllocationError(
   Reason = "Enabling session creates a new ARSessionConfiguration")]
public void OnEnable()
{
   LifecycleManager.Instance.EnableSession();
}

/// <summary>
/// Unity OnDisable.
/// </summary>
[SuppressMemory
AllocationError(
   IsWarning = true, Reason = "Requires further
investigation.")] public void OnDisable()
{
   LifecycleManager.Instance.DisableSession();
}

/// <summary>
/// Unity OnValidate.
/// </summary>
public void OnValidate()
{
   if (DeviceCameraDirection == DeviceCameraDirection.FrontFacing && SessionConfig
   != null)
   {
      if (SessionConfig.PlaneFindingMode != DetectedPlaneFindingMode.Disabled)
      {
         Debug.LogErrorFormat("Plane Finding requires back-facing camera.");
      }

      if (SessionConfig.LightEstimationMode ==
            LightEstimationMode.EnvironmentalHDRWitho
            utReflections ||
         SessionConfig.LightEstimationMode ==
            LightEstimationMode.EnvironmentalHDR
            WithReflections)
      {
         Debug.LogErrorFormat("LightEstimationMode.{0} is
            incompatible with" + "front-facing (selfie) camera.",
            SessionConfig.LightEstimationMode);
      }
```

```
            if (SessionConfig.CloudAnchorMode != CloudAnchorMode.Disabled)
            {
                Debug.LogErrorFormat("Cloud Anchors require back-facing camera.");
            }

            if (SessionConfig.AugmentedImageDatabase != null)
            {
                Debug.LogErrorFormat("Augmented Images require back-facing camera.");
            }
        }

        if (DeviceCameraDirection ==
            DeviceCameraDirection.BackFacing &&
            SessionConfig != null &&
            SessionConfig.AugmentedFaceMode ==
                AugmentedFaceMode.Mesh)
        {
            Debug.LogErrorFormat("AugmentedFaceMode.{0} requires front-facing
                (selfie) camera.", SessionConfig.AugmentedFaceMode);
        }

        if (SessionConfig == null)
        {
            Debug.LogError("SessionConfig is required by ARCoreSession.");
        }

        if (CameraConfigFilter == null)
        {
            Debug.LogError("CameraConfigFilter is required by
                ARCoreSession. " + "To get all available
                configurations, set CameraConfigFilter to " +
                "a filter with all options selected.");
        }
    }

    /// <summary>
    /// Registers a callback that allows a camera configuration to be selected from a list of
    /// valid configurations.
    /// The callback should be registered before the ARCore session is enabled
    /// to ensure it is triggered on the first frame update.
    /// The callback will then be invoked each time the ARCore session is resumed,
    /// which can happen when the <see cref="ARCoreSession"/> component is enabled or the
    /// Android app moves from a state of 'paused' to 'resumed' state.
    ///
    /// Note: Starting in ARCore 1.12, changing the active camera config will make existing
    /// anchors and trackables fail to regain tracking.
    /// </summary>
    /// <param name="onChooseCameraConfiguration">The callback to register for selecting a
    /// camera configuration.</param>
    public void RegisterChooseCameraConfigurationCallback(
        OnChooseCameraConfigurationDelegate
        onChooseCameraConfiguration)
    {
        m_OnChooseCameraConfiguration = onChooseCameraConfiguration;
```

```
        }

        internal OnChooseCameraConfigurationDelegate
        GetChooseCameraConfigurationCallback()
        {
            return m_OnChooseCameraConfiguration;
        }
    }
}
```

**Pawnmanipulator.cs**

```
//_____________________________________________________
// <copyright file="PawnManipulator.cs" company="Google">
//
// Copyright  Google LLC. All Rights Reserved.
//
// Licensed under the Apache License, Version 2.0 (the "License");
// you may not use this file except in compliance with the License.
// You may obtain a copy of the License at
//
// http://www.apache.org/licenses/LICENSE-2.0
//
// Unless required by applicable law or agreed to in writing, software
// distributed under the License is distributed on an "AS IS" BASIS,
// WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied//
See the License for the specific language governing permissions and
// limitations under the License.
//
// </copyright>
//_____________________________________________________

namespace GoogleARCore.Examples.ObjectManipulation
{
    using
    Google
    ARCor
    e;
    using
    UnityE
    ngine;

    /// <summary>
    /// Controls the placement of objects via a tap gesture.
    /// </summary>
    public class PawnManipulator : Manipulator
    {
        /// <summary>
        /// The first-person camera being used to render the passthrough camera image (i.e. AR
        /// background).
        /// </summary>
        public Camera FirstPersonCamera;
```

```
/// <summary>
/// A prefab to place when a raycast from a user touch hits a plane.
/// </summary>
public GameObject PawnPrefab;

/// <summary>
/// Manipulator prefab to attach placed objects to.
/// </summary>
public GameObject

ManipulatorPrefab;

private bool

StopSpawn=true;

/// <summary>
/// Returns true if the manipulation can be started for the given gesture.
/// </summary>
/// <param name="gesture">The current gesture.</param>
/// <returns>True if the manipulation can be started.</returns>
protected override bool CanStartManipulationForGesture(TapGesture gesture)
{
   if (gesture.TargetObject == null)
   {
      return true;
   }

   return false;
}

/// <summary>
/// Function called when the manipulation is ended.
/// </summary>
/// <param name="gesture">The current
gesture.</param> protected override void
OnEndManipulation(TapGesture gesture)
{
   if (gesture.WasCancelled)
   {
      return;
   }

   // If gesture is targeting an existing
   object we are done. if
   (gesture.TargetObject != null)
   {
      return;
   }

   // Raycast against the location the player touched to
   search for planes. TrackableHit hit;
   TrackableHitFlags raycastFilter = TrackableHitFlags.PlaneWithinPolygon;
```

```
            if (Frame.Raycast(
                gesture.StartPosition.x, gesture.StartPosition.y, raycastFilter, out hit))
            {
                // Use hit pose and camera pose to check if hittest is from the
                // back of the plane, if it is, no need to
                create the anchor. if (StopSpawn ==
                true)
                {
                    StopSpawn = false;

                    if ((hit.Trackable is DetectedPlane) &&
                        Vector3.Dot(FirstPersonCamera.transform.position -
                        hit.Pose.position,
                            hit.Pose.rotation * Vector3.up) < 0)
                    {
                        Debug.Log("Hit at back of the current DetectedPlane");
                    }
                    else
                    {
                        // Instantiate game object at the hit pose.
                        var gameObject = Instantiate(PawnPrefab, hit.Pose.position, hit.Pose.rotation);

                        //
                        Instantiate
                        manipulat
                        or. var
                        manipulat
                        or =
                            Instantiate(ManipulatorPrefab, hit.Pose.position, hit.Pose.rotation);

                        // Make game object a child of the
                        manipulator.
                        gameObject.transform.parent =
                        manipulator.transform;

                        // Create an anchor to allow ARCore to track the hitpoint as understanding of
                        // the physical world evolves.
                        var anchor = hit.Trackable.CreateAnchor(hit.Pose);

                        // Make manipulator a child of the
                        anchor.
                        manipulator.transform.parent =
                        anchor.transform;

                        // Select the placed object.
                        manipulator.GetComponent<Manipula
                        tor>().Select();
                    }
                }
            }
        }
    }
}
```

# Conclusion

# Conclusion

In augmented Reality customer can feel and experience it anywhere or where he want. It's not necessary to have Real car with user we can experience in virtual car. The interaction with augmented world is in two ways, using physical objects placeholder objects and by hand gestures. Created an Augmented Reality app to show how automobile companies can integrate AR in their app so that interested buyers can have a quick look on how various car models look in different colors.so buyer can quickly decide on which features he want in his new car and confirm his decision by going to the showroom just to take a test-drive this would save a lot of time

# Future Enhancement

# Future Enhancement

Improve interaction in augmented reality (for the field personnel) Increase immersion and situational awareness for spatially distributed users in the virtual reality and augmented reality. Future work will focus on preparing a fully functional car simulation. We can add the feature of drive the car. We can create same feature in website so no need to download the app and add Buy Option so user can buy a car online

# List of Figures

# List of Tables

# REFERENCES
# & BIBLIOGRAPHY

# References & Bibliography

## Research paper–

Title-AUGMENTED REALITY TECHNIQUES FOR VEHICLE MAINTENANCE
Author- Marcin Januszka, Wawrzyniec Panfil
Date of Paper- January 2015

Title- A Survey On Augmented Reality Applications Using Deep Learning
Author- Cengiz Güngör, Kenan Zengin
Date of paper- December 2017

Title- Recreation of history using augmented reality
Author- Nilam Desai
Date of Paper- February 2018

Title- Augmented Reality based Platform to share virtual worlds
Author-Prof. Dhananjay Gaikwad1, Akash Chikane2, Shrikrishna Kulkarni3, Aishwarya Nhavkar4
Date of Paper- Jan-2018

# Bibilography


- https://www.researchgate.net/publication/281461043_AUGMENTED_REALITY_TECHNIQ UES_FOR_VEHICLE_MAINTENANCE
- https://www.researchgate.net/publication/322206805_AFFECTIVE_COMPUTING_AND_AUGMENTED_REALITY_FOR_CAR_DRIVING_SIMULATORS
- https://www.researchgate.net/publication/322332639_A_Survey_On_Augmented_Reality_Ap plications_Using_Deep_Learning


**Sites Referred:**


https://unity.com/
https://developers.
google.com/ar

https://sketchfab.com/3d-models/sketchtab-5129dd11ad864158b80f553c7a2a8343