

E-COMMERCE WITH AR

CIP PROJECT REPORT

Submitted by

AKIL VISHNU M 2018115010

BALASUBRAMANIAM M 2018115018

VENKAT KARTHICK P 2018115125

submitted to the Faculty of

INFORMATION AND COMMUNICATION ENGINEERING

in partial fulfillment for the award of the degree

of

BACHELOR OF TECHNOLOGY

in

INFORMATION TECHNOLOGY



DEPARTMENT OF INFORMATION SCIENCE AND TECHNOLOGY

COLLEGE OF ENGINEERING, GUINDY

ANNA UNIVERSITY

CHENNAI 600 025

May 2021

ABSTRACT

An Online Shopping Mobile Application incorporated with Augmented Reality which is aimed to enhance the traditional user experience of online shopping leveraging new technology that provides more lively experience of the products they search for. Showcasing of products is enhanced using Augmented Reality which provides a virtual experience of the products. Products sold online are made to appear in 3D format which helps user to have a better view of the product in all dimensions as if they were to be purchased in a physical store/shop. Products can also be viewed, bound to the surrounding environment or attached to any body, with the use of the cameras in the smartphones.

TABLE OF CONTENTS

| | |
|---|----------|
| ABSTRACT | ii |
| 1 INTRODUCTION | 1 |
| 1.1 TERMINOLOGIES AND PLATFORMS | 1 |
| 1.1.1 Augmented Reality | 1 |
| 1.1.2 Unity-3D | 1 |
| 1.1.3 Flutter | 2 |
| 1.2 SOFTWARE STACK | 2 |
| 1.2.1 Blender | 2 |
| 1.2.2 Figma | 3 |
| 1.2.3 Dart | 3 |
| 1.2.4 Firebase | 3 |
| 1.2.5 Unity | 3 |
| 1.2.5.1 ARFoundation | 4 |
| 1.2.5.2 ARSubsystems | 4 |
| 1.2.5.3 ARKit FaceTracking | 5 |
| 1.2.5.4 ARCore XRPlugin | 5 |
| 2 EXISTING WORK | 6 |
| 2.1 LENSKART | 6 |
| 2.2 AMAZON | 6 |
| 3 SYSTEM DESIGN | 7 |
| 3.1 DESIGN OF THE WORK | 7 |
| 3.2 USER AUTHENTICATION | 7 |
| 3.3 RETRIEVAL OF AVAILABLE PRODUCTS | 7 |
| 3.4 SHOWCASING PRODUCTS IN AR AND IN 3D | 8 |
| 3.4.1 Products in 3D | 8 |
| 3.4.2 Products in AR | 8 |
| 3.4.2.1 Wearable Products | 9 |
| 3.4.2.2 Non-wearable Products | 9 |
| 3.5 INTEGRATING AND COMMUNICATING WITH UNITY 3D FROM FLUTTER | 9 |

| | | |
|-------------------|--|-----------|
| 3.6 | UPDATE USER'S CART UPON ADDING A PRODUCT TO CART | 9 |
| 3.7 | UPDATE CHANGES IN CART AND CHECKOUT | 10 |
| 3.8 | UNITY | 10 |
| 3.8.1 | HORIZONTAL PLANE DETECTION | 10 |
| 3.8.2 | VERTICAL PLANE DETECTION | 10 |
| 3.8.3 | FACE TRACKING | 10 |
| 4 | IMPLEMENTATION OF THE WORK | 12 |
| 4.1 | AUTHENTICATION | 12 |
| 4.2 | HOME AND CART | 13 |
| 4.2.1 | Search/Category Filters | 13 |
| 4.2.2 | Display of products added to Cart | 13 |
| 4.3 | Selected Product Display and 3D view | 14 |
| 4.3.1 | 3D View of Products | 14 |
| 4.3.2 | Checkout | 15 |
| 4.4 | WEARABLE PRODUCTS | 16 |
| 4.5 | NON-WEARABLE PRODUCTS | 16 |
| 4.6 | AR VIEW (FACE TRACKING) | 16 |
| 4.7 | AR-VIEW (HORIZONTAL / VERTICAL PLANE TRACKING) | 16 |
| 5 | CONCLUSION | 20 |
| 5.1 | CONCLUSION | 20 |
| 5.2 | FUTURE WORKS | 20 |
| REFERENCES | | 21 |

CHAPTER 1

INTRODUCTION

The project aims to enhance the traditional 2D user experience with AR using 3D objects. It is aimed to be developed as an android application. 3D objects are classified as wearable and non-wearable products which are displayed to the user in their ambience.

1.1 TERMINOLOGIES AND PLATFORMS

1.1.1 Augmented Reality

It is an interactive experience of a real-world environment where the objects that reside in the real world are enhanced by computer-generated perceptual information. Any object in the real world can be enhanced to make it appear (artificially) in the space or bound to any object. The enhancement can be constructive (addition to the environment) or destructive (masking the environment).

While the physical world is three-dimensional, most data is displayed on two-dimensional pages, images and screens. This gap is closed by - Augmented reality, a set of technologies that superimposes digital data and images on physical objects and environments.

This concept is taken into account and thought of providing a new experience in online shopping replacing the traditional experience, where the products are displayed in 2-D images.

1.1.2 Unity-3D

It is a cross-platform game engine which is used to develop Android and iOS mobile games. The engine has since been gradually extended to support a variety of desktop, mobile, console and virtual reality platforms.

It provides in-built Augmented Reality packages. Those packages are customized and used in this project. Some of the packages included in the projects are ARFoundation , ARKitFaceTracking , ARCore XRplugin, ARKit XRplugin and so on.

1.1.3 Flutter

Flutter is Google's SDK for crafting beautiful, fast user experiences for mobile, web, and desktop from a single codebase. Flutter works with existing code, is used by developers and organizations around the world, and is free and open source.

Flutter's layered architecture gives you control over every pixel on the screen and its powerful compositing capabilities let you overlay and animate graphics, video, text, and controls without limitation. Flutter includes a full set of widgets that deliver pixel-perfect experiences on both iOS and Android.

1.2 SOFTWARE STACK

The listed below are the softwares used in the project.

1.2.1 Blender

Blender is the free and open source 3D creation suite. It supports the entirety of the 3D pipeline—modeling, rigging, animation, simulation, rendering, compositing and motion tracking, video editing and 2D animation pipeline.

1.2.2 Figma

Figma is a web-based graphics editing and user interface design app which is used to do all kinds of graphic design work from wireframing websites, designing mobile app interfaces, prototyping designs, crafting social media posts, and so on. It is different from other graphics editing tools mainly because, it works directly on the browser. It is used to access the projects and start designing from any computer or platform without having to buy multiple licenses or install software.

1.2.3 Dart

Dart is a client-optimized language for fast apps on any platform. It is an open-source general-purpose programming language. It is originally developed by Google and later approved as a standard by ECMA. Dart is a new programming language meant for the server as well as the browser. Introduced by Google, the Dart SDK ships with its compiler – the Dart VM. The SDK also includes a utility -dart2js, a transpiler that generates JavaScript equivalent of a Dart Script.

1.2.4 Firebase

Firebase is Google's mobile platform that helps in quickly developing high-quality apps and grow a business.

1.2.5 Unity

Unity is a game engine which is used not only for game development, but also for extended reality development.

1.2.5.1 ARFoundation

AR Foundation is a cross-platform framework that allows you to build augmented reality experiences once, then build for either Android or iOS devices. AR Foundation allows to work with augmented reality platforms in a multi-platform way within Unity. This package presents an interface for Unity developers to use, but doesn't implement any AR features itself. AR Foundation on a target device, you also need separate packages for the target platforms officially supported by Unity:

- ARCore XR Plugin on Android
- ARKit XR Plugin on iOS
- Magic Leap XR Plugin on Magic Leap
- Windows XR Plugin on HoloLens

1.2.5.2 ARSubsystems

A subsystem is a platform-agnostic interface for surfacing different types of functionality and data. The AR-related subsystems are defined in this package and use the namespace `UnityEngine.XR.ARSubsystems`. This package only provides the interface for various subsystems. Implementations for these

subsystems (called "providers") can typically be found in another package or plugin. Session, Raycasting, Camera, Plane Detection, Depth, Image Tracking, Face Tracking,

1.2.5.3 ARKit FaceTracking

ARKit FaceTracking is one of the functionality supported by ARFoundation that enables the camera to detect and track human faces in real-time. To enable Unity face detection on a specific platform, AR Foundation provides ARCore and ARkit packages. Each overlays textures and 3D models on a detected face using a face mesh. This mesh may include face landmarks, vertices, UV coordinates, facial regions and other data by which algorithms recognize a face in a video stream. ARKit face tracking package for Unity recognizes the position, topology, and facial expressions to overlay AR assets and animate a user face in real-time on iOS devices. It requires a front depth camera.

1.2.5.4 ARCore XRPlugin

ARCore is a platform for building augmented reality apps on Android. This SDK provides native APIs for all of the essential AR features like motion tracking, environmental understanding, and light estimation.

CHAPTER 2

EXISTING WORK

The concept is inspired from lenskart and amazon.

2.1 LENSKART

Lenskart is the leading e-commerce portal for eyewear in India. It has revolutionised the eyewear industry in the country with its omni-channel approach using Augmented Reality. Lenskart uses Augmented Reality technology to show the frames on the face.

This project is inspired by that concept and it has been extended and used for the entire app which is not available elsewhere. Lenskart showcased their spectacles' frames in AR. This project tries to showcase every possible product in the user's reality which gives a near realistic experience.

2.2 AMAZON

Amazon (Amazon.com) is the world's largest online retailer and a prominent cloud services provider. Amazon uses three dimensional model viewer for some of their products to establish enhanced user experience. This project inspires that concept and tries to provide 3-dimensional view of every possible objects.

Thus the project resembles as an upgradation to the current version of Online E-Commerce shopping apps enriched with virtual products in user's vicinity and 3D product viewer.

CHAPTER 3

SYSTEM DESIGN

3.1 DESIGN OF THE WORK

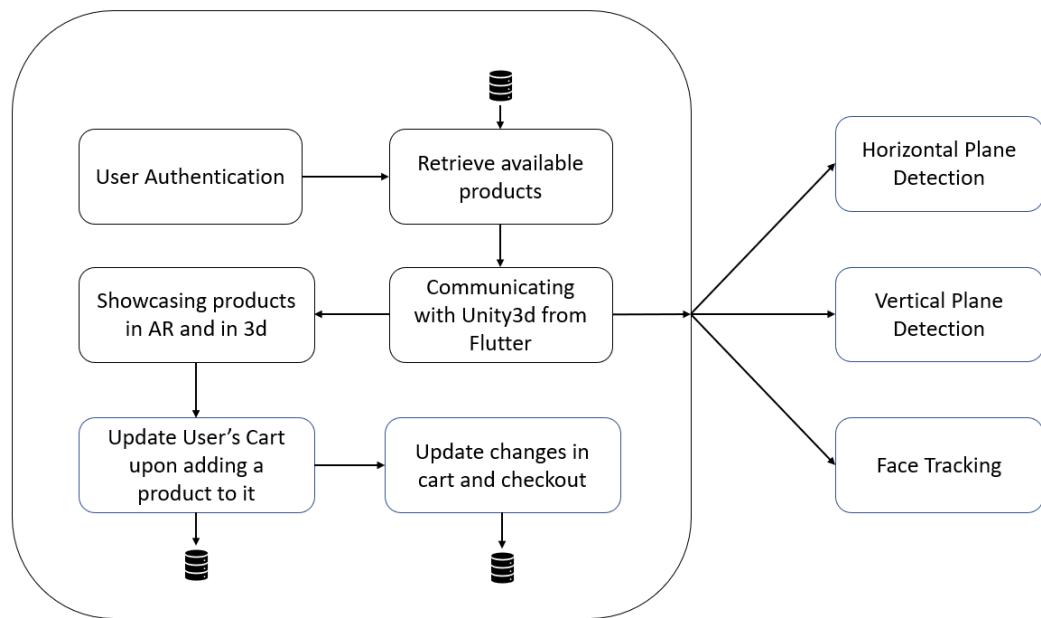


Figure 3.1: Design

3.2 USER AUTHENTICATION

Authenticity of the user is checked in this module. User data for creating an account is validated and handled all exceptions in case of any errors in storing data into firebase. User login details are also validated before checking the authentication details in database and then further exceptions like incorrect password, invalid username, no such account are handled and displayed as a toast.

3.3 RETRIEVAL OF AVAILABLE PRODUCTS

All the products available for sale is displayed in this module. All the available products are segregated into different categories. One of the categories is "All" which shows all the available products. The other categories shows products particular to that category.

This module has a search bar to filter the category of the product they require. Cart details can be viewed from this module.

3.4 SHOWCASING PRODUCTS IN AR AND IN 3D

Products are showcased in 3D as well as in AR.

3.4.1 Products in 3D

All the products' 3D models are prepared and converted to a ".glb" extension. These models are placed in the asset of the Flutter application. These models are shown in each product's description page using 3DModelViewer plugin. 360 degree view of all products is provided. Products displayed in 3D view can be scaled and rotated in all dimensions.

3.4.2 Products in AR

Products to be displayed in AR are classified into two categories.

- Wearable Products
- Non-wearable Products

3.4.2.1 Wearable Products

Products that can be worn by people are made to appear as if they were wearing it. This is done using AR Face tracking. [subsection 3.8.3]

3.4.2.2 Non-wearable Products

All the products that are placed on floors and walls falls under non-wearable category. These products are made to appear on the user's surrounding space. Users are allowed to select a place to where the product should be placed using a marker that appears on the detected plane. On placing the target in correct position and on clicking the target marker, the object is rendered and displayed over the marker. [subsection 3.8.1], [subsection 3.8.2]

3.5 INTEGRATING AND COMMUNICATING WITH UNITY 3D FROM FLUTTER

In this module, flutter User Interface and Unity AR models are integrated using Flutter-Unity-Widget plugin which establishes communication between flutter and unity.

A Flutter-Unity-Widget package is loaded in Unity. A C-Sharp script is created which switches the scenes in unity.

Flutter provides the name of the unity scene to this script as a parameter and the script loads the respective scene which is rendered to the flutter app and the unity scene opens inside the flutter app. A Loading Scene is rendered until the actual scene is loaded hiding the loading time to the user.

3.6 UPDATE USER'S CART UPON ADDING A PRODUCT TO CART

Cart details are updated in this module. Cart of the user is updated on addition of a product to cart.

3.7 UPDATE CHANGES IN CART AND CHECKOUT

Any changes in the cart like altering quantity, changing variant of the product are checked and updated to the cart.

3.8 UNITY

Augmented Reality part of the project is entirely done in Unity 3D. Wearable and non-wearable products are developed in AR in this module.

3.8.1 HORIZONTAL PLANE DETECTION

In this module, horizontal plane is detected using ARPlane Manager and ARRayCast Manager scripts which are provided by ARFoundation package. Horizontal plane is detected and objects like laptop, musical instruments, Sofa are placed on the detected plane. A marker is placed and when it is touched, an AR object is made to spawn above the marker which is controlled by a C-Sharp script.

3.8.2 VERTICAL PLANE DETECTION

In this module, Vertical plane is detected using ARFoundation package. Vertical plane is detected and objects like Television and paintings are placed on the detected plane.

3.8.3 FACE TRACKING

In this module, Face is detected using ARKit Facetracking and ARKit XRPlugin packages provided by unity. Face is detected here and objects like sunglasses, cap, mask and power glasses are placed above the detected face. This module comprises wearable products. Random face model is taken and a custom object is placed above and then the face model is made transparent making the object to appear above the face.

CHAPTER 4

IMPLEMENTATION OF THE WORK

4.1 AUTHENTICATION

Users are allowed to register to create an account in the application. Users can then login ensuring the authentication of the user. User can login to view or purchase products. The Registration and login process is done as follows:

E-mail and password authentication is done. Validation of the content is done.

The authentication data is stored in firebase. In case of any error in storing the authentication information, a toast (popup) is shown displaying the appropriate message.

While logging in, if no such account exists and if the password is not wrong, a toast with appropriate message is shown to users. All such exceptions in registering and logging in are handled effectively. [Figure 4.1]

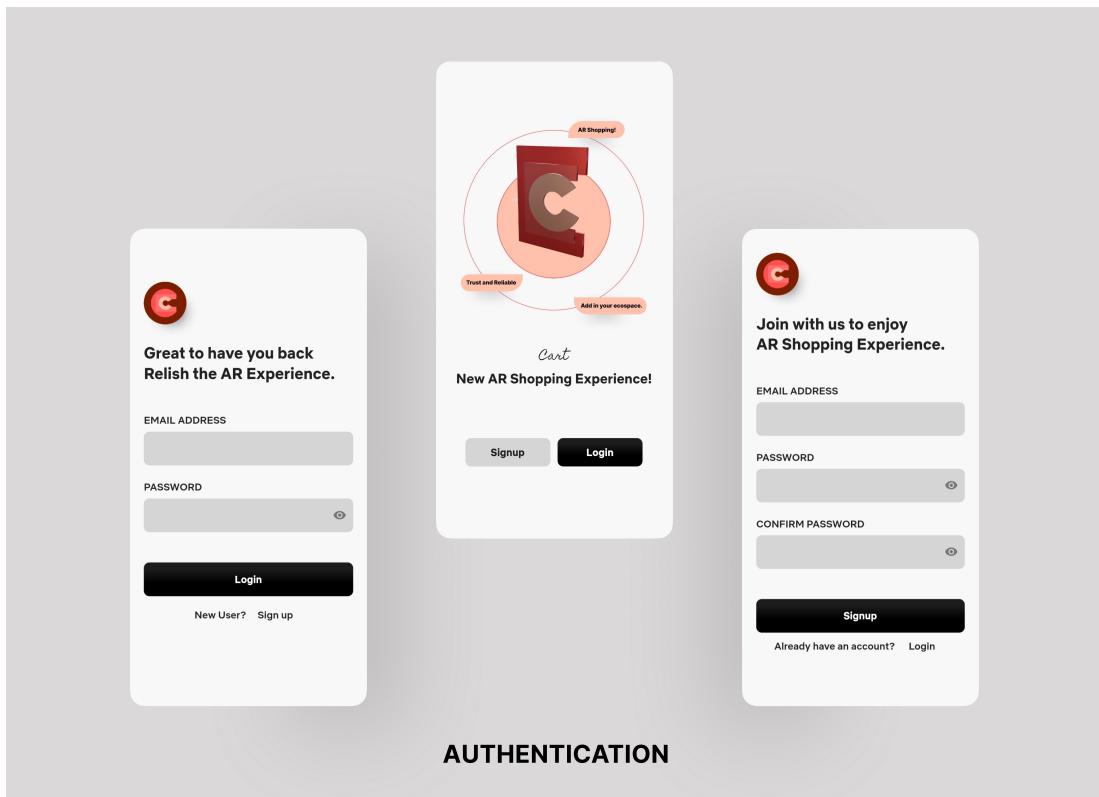


Figure 4.1: Authentication

4.2 HOME AND CART

This is the module where all the available products are displayed to the user. All the available products are segregated into different categories like Eye Wear, Smart Phones, Laptops, Caps, Instruments, Masks and so on. Initially all products are displayed with "All" category selected. Users can then navigate to any other category and search through the products they require.

4.2.1 Search/Category Filters

A search bar is also provided where users can filter the products based on the categories they look for.

4.2.2 Display of products added to Cart

All the products that are added to cart are displayed here. Users can increase or decrease the quantity of the products present in the cart. A summary of the products and their quantities and price details are also displayed. Users can then proceed to payment. [Figure 4.2]

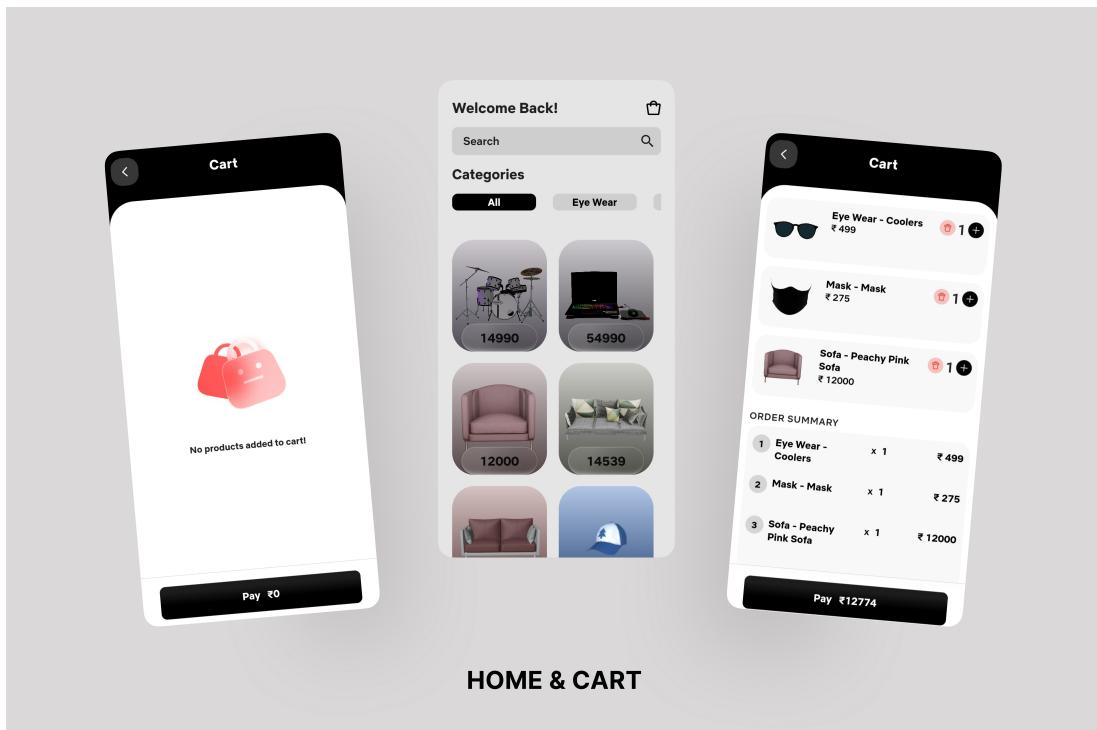


Figure 4.2: Home and Cart

4.3 Selected Product Display and 3D view

The Product selected by the user is displayed in this module. The Interface is made in such a way that information is organised so that it is easy for users. The aim is to give consistent look and feel for the product. The background of the product is made dynamic. Dominant color from the product is picked and the lighter and darker variants of the color is used to create a gradient. Font size and opacity is adjusted for optimal readability and accessibility.

4.3.1 3D View of Products

Products are shown in 3D view where users can have a 3D experience of the product within the application. objects can be rotated to any angle and viewed according to their convenience.

4.3.2 Checkout

Users can checkout their products which shows an action sheet. Users can checkout the product immediately or add it to the cart. Action sheet provides the product variant, product quantity details and "Buy Now", "Add to Cart" buttons. It also provides a summary of the product and the total cost of the product. [Figure 4.3]

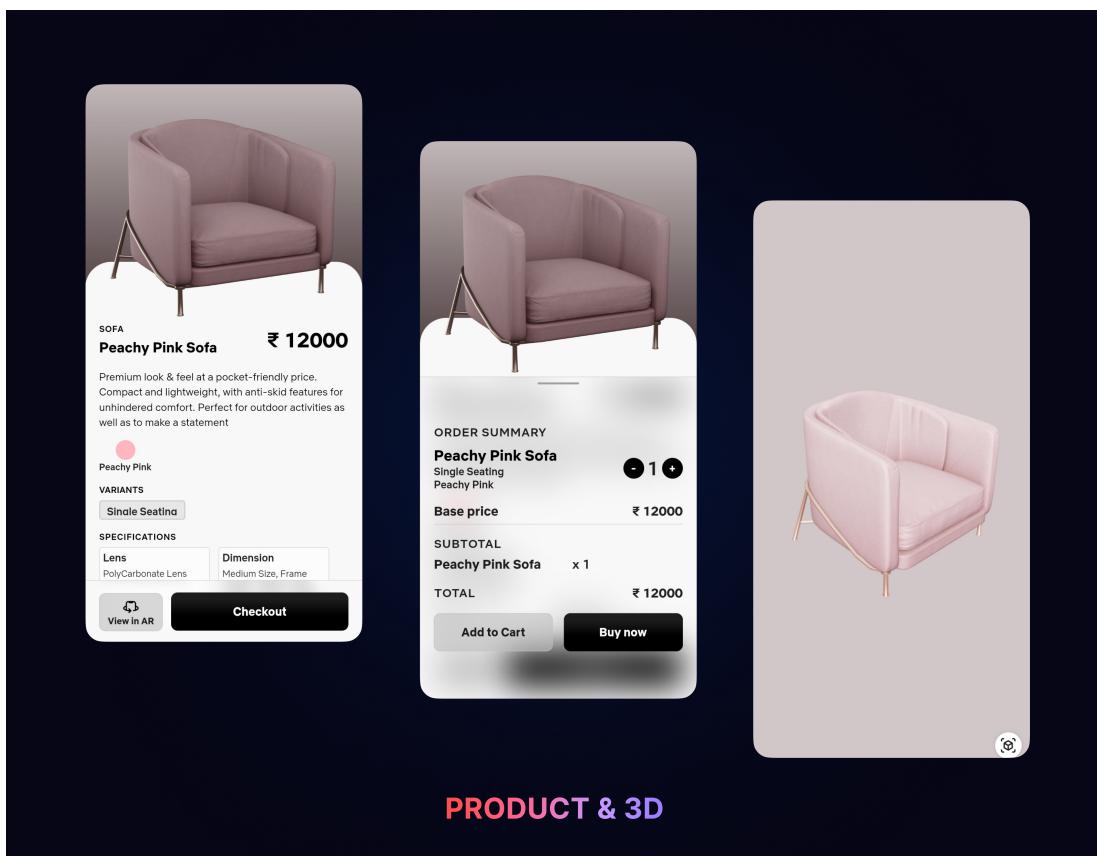


Figure 4.3: Product and 3D display

4.4 WEARABLE PRODUCTS

Products are classified as wearable and non-wearable during the design phase. Wearable products are made to appear in the user's surrounding environment. Some of the wearable products which are shown in the application are sofa, instruments, laptop and so on. These products are made to appear on the user's space with the help of the following unity packages ARFoundation, ARCore XRPlugin, ARKit XRPlugin, ARSubsystems.

4.5 NON-WEARABLE PRODUCTS

Non-Wearable products are made to appear in the user's body or face. Some of the non-wearable products which are displayed in the application are medical mask, sunglasses, cap, power glasses which are made to appear on the face and T-shirt which are made to appear above the body. Some of the packages used are ARKit FaceTracking and ARKit XR Plugin.

4.6 AR VIEW (FACE TRACKING)

Face tracing is done in Unity using ARKit FaceTracking. ARKit FaceTracking detects the user's face and paints with the sample material above the face.

A custom face model is taken and it is made transparent. Objects to be placed over face is placed above the transparent model and it is made the child of the transparent face model in the hierarchy and the object is converted into a prefab. The converted prefab is passed into the Game Object. Thus, the user will feel that the object is above their face. Similarly T-shirts and trousers are made to appear on the user.



Figure 4.4: Face Tracking

4.7 AR-VIEW (HORIZONTAL / VERTICAL PLANE TRACKING)

Plane tracking is done in Unity using ARRay Cast Manager and AR Plane Manager. AR PlaneManager detects the plane and ARRayCast Manager provides the positional and rotational co-ordinates to the C-Sharp script.

A self-designed marker is made and it is made to appear above the tracked plane. If the user touches on the marker, the positional and rotational co-ordinates of the user's space are send to the script and the designed objects

are coded to spawn in those positional and rotational co-ordinates which gives us the illusion of objects above the marker. This part of the app is fully designed and coded in Unity. This Unity part is integrated with the flutter app using Flutter-Unity widget. When AR button in the product page is pressed, the unity part is opened and the products appear in the user's ambience.

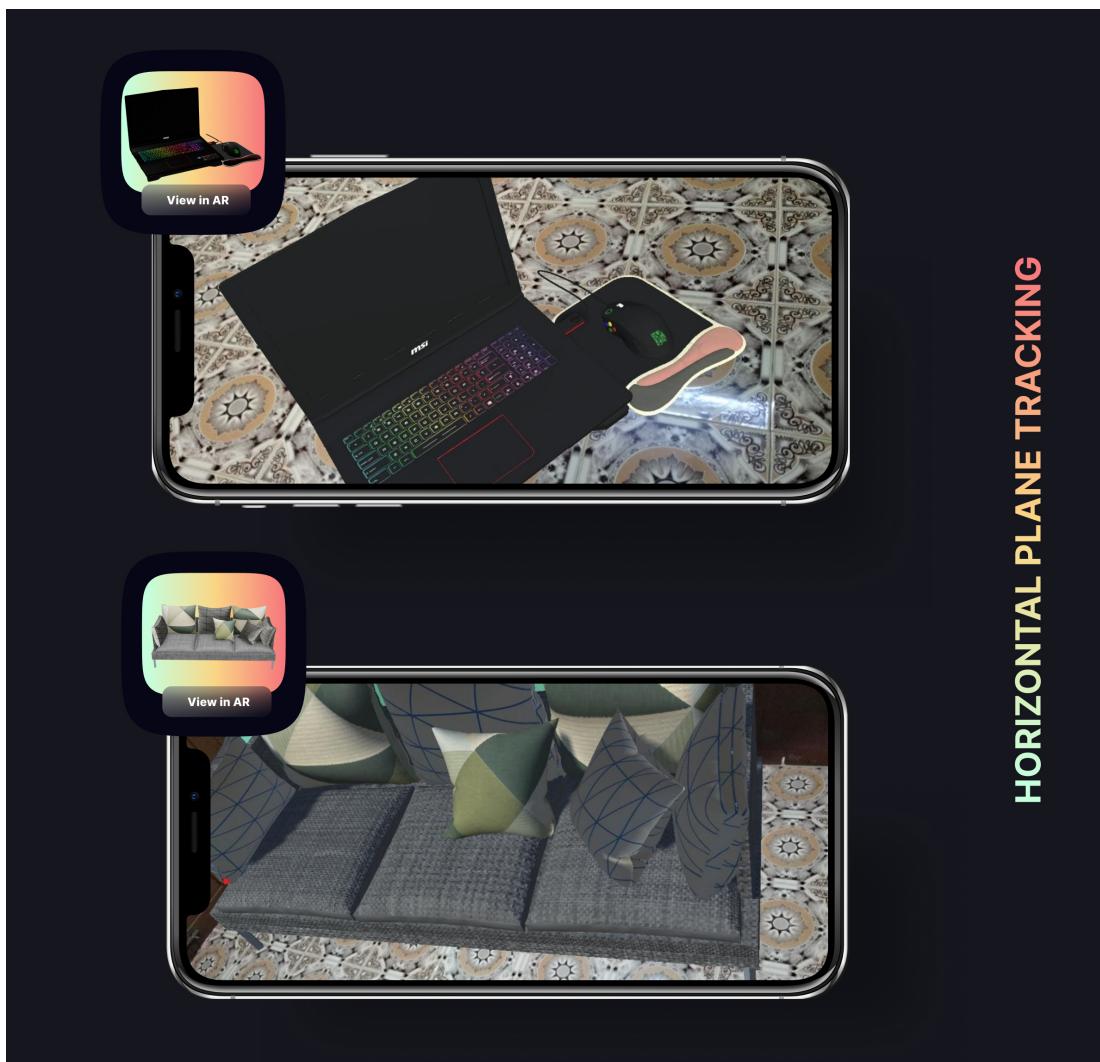


Figure 4.5: Horizontal Plane Tracking

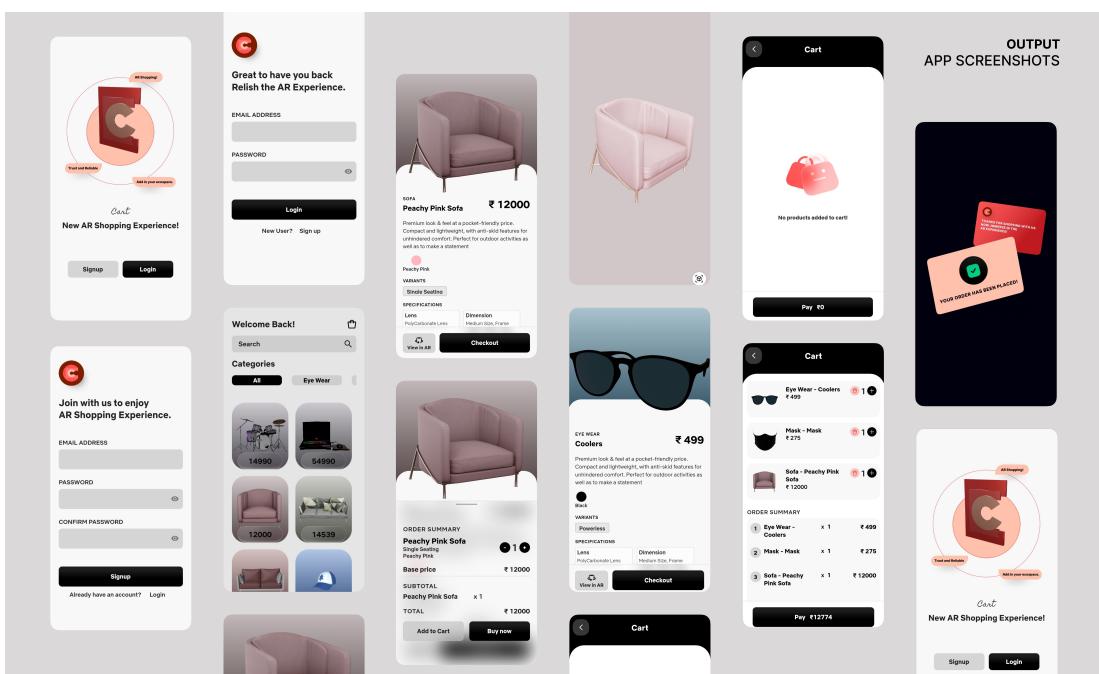


Figure 4.6: Collection of all pages

CHAPTER 5

CONCLUSION

5.1 CONCLUSION

The core idea of the project is to narrow the gap between Reality shopping and E-Shopping and also make it much more interactive. Augmented Reality (AR) is used to superimpose digital 3D models in your environment. Mobile Application is developed as an interface where the user can look for variety of products and bring it to reality using AR. Mobile application is made with all basic functions of E-Shopping. Users are allowed to virtually try out a variety of merchandise before buying. User interface is designed to provide easy to use experience and visually pleasing. Products can be rotated and viewed in all dimensions. This ensures customers to make better and confident decisions and adds value to business.

5.2 FUTURE WORKS

The project can have efficient body tracking models and can have VR experience as a further improvement.

REFERENCES

- [1] https://pub.dev/packages/flutter_unity_widget. *Flutter dependency*.
- [2] https://pub.dev/packages/model_viewer. *Flutter dependency*.
- [3] <https://github.com/juicycleff/flutter-unity-view-widget/issues/87>.
Flutter-unity integration.
- [4] <https://docs.unity3d.com/Packages/com.unity.xr.arfoundation@4.0/manual/plane-management.html>.
AR Plane Management.
- [5] <https://developers.google.com/ar/develop/java/augmented-faces/developer-guide>.
Augmented Faces in Android.
- [6] <https://docs.unity3d.com/Packages/com.unity.xr.arfoundation@4.0/manual/raycast-management.html>.
AR Raycast Management.