AUGMENTED REALITY FURNITURE APPLICATION

Submitted in partial fulfillment of the requirements for the award of
Bachelor of Engineering degree in Computer Science and Engineering

By

SAI BHAVAN GUBBALA (Reg. No - 39110353)
DHRUV NAIDU ALTI (Reg. No – 39110272)

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

SCHOOL OF COMPUTING

SATHYABAMA

INSTITUTE OF SCIENCE AND TECHNOLOGY
(DEEMED TO BE UNIVERSITY)
Accredited with Grade "A" by NAAC | 12B Status by UGC | Approved by AICTE
JEPPIAAR NAGAR, RAJIV GANDHI SALAI,
CHENNAI - 600119

APRIL - 2023

SATHYABAMA
INSTITUTE OF SCIENCE AND TECHNOLOGY
(DEEMED TO BE UNIVERSITY)
Accredited with Grade "A" by NAAC | 12B Status by UGC | Approved by AICTE
www.sathyabama.ac.in

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

BONAFIDE CERTIFICATE

This is to certify that this Project Report is the bonafide work of **SAI BHAVAN GUBBALA (Reg. No - 39110353) and DHRUV NAIDU ALTI (Reg. No - 39110272)** who carried out the Project Phase-2 entitled **"AUGMENTED REALITY FURNITURE APPLICATION"** under my supervision from January 2023 to April 2023.

Internal Guide
Mrs. S. R. SRIVIDHYA M.E., Ph.D.

Head of the Department
Dr. L. LAKSHMANAN, M.E., Ph.D.

Submitted for Viva voce Examination held on 20.04.2023

Internal Examiner **External Examiner**

DECLARATION

I, **SAI BHAVAN GUBBALA (Reg. No- 39110353),** hereby declare that the Project Phase-2 Report entitled **"AUGMENTED REALITY FURNITURE APPLICATION"** done by me under the guidance of **Mrs. S. R. SRIVIDHYA, M.E., Ph.D.** is submitted in partial fulfillment of the requirements for the award of Bachelor of Engineering degree in **Computer Science and Engineering**.

DATE:20.04.2023

PLACE: Chennai **SIGNATURE OF THECANDIDATE**

ACKNOWLEDGEMENT

I am pleased to acknowledge my sincere thanks to **Board of Management** of **SATHYABAMA** for their kind encouragement in doing this project and for completing it successfully. I am grateful to them.

I convey my thanks to **Dr. T. Sasikala M.E., Ph. D**, **Dean**, School of Computing, **Dr. L. Lakshmanan M.E., Ph.D.,** Head of the Department of Computer Science and Engineering for providing me necessary support and details at the right time during the progressive reviews.

I would like to express my sincere and deep sense of gratitude to my Project Guide **Mrs. S. R. SRIVIDHYA M.E., Ph.D,** for her valuable guidance, suggestions and constant encouragement paved way for the successful completion of my phase-2 project work.

I wish to express my thanks toall Teaching and Non-teaching staff members of the **Department of Computer Science and Engineering** who were helpful in many ways for the completion of the project.

ABSTRACT

In the modern digital transformation era, providing customers with confidence in the suitability of the furniture they choose for various situations in which it will be used is the major problem in online furniture sales. While buying furniture, purchasers must ensure the furniture's model, size, and color are appropriate for and complement the space in which they will be put. The flexibility and appropriateness of the furniture product when placed in a particular area may be seen by prospective buyers with suitable instruments. For furniture to be sold online correctly, these issues must be resolved. By using Augmented Reality, this is feasible, to do. When it comes to online product customization, certain application models in the furniture sector employ Augmented Reality and mobile technology. A 3D product visualization tool, product pricing, and an enhanced renderer are all included. For the purpose of creating AR-based furniture technology, Google AR is used in the development process. It provides a framework for people or teams to creatively and flexibly address challenging tasks and issues in order to generate and deliver the best performance. The study is finished using Android Studio and Google AR services. Based on the results of the studies, it is advised that this application be enhanced to have a greater degree of performance. This might be achieved by creating a variety of various creator patterns using barcodes or QR codes, classifying surfaces like ceilings and walls so that the program can accurately show 3D things on them, and improving 3D models to fit them in the projecting space.

	C. RESEARCH PAPER	

LIST OF FIGURES

LIST OF TABLES

LIST OF ABBREVATIONS

S. No.	ABBREVATION	FULL FORM
1	AR	Augmented Reality
2	VR	Virtual Reality
3	UI	User Interface
4	UX	User Experience
5	SDK	Student Development Kit
6	API	Application Programming Interface
7	PBR	Physically Based Rendering
8	OBJ	Object
9	GLTF	GL Transmission Format
10	XML	Extensible Markup Language
11	SQL	Structured Query Language.

CHAPTER 1

INTRODUCTION

India's e-commerce sector continues to grow significantly. But, it's only possible to have change with growth, thus, for the e-commerce sector to prosper, new technology developments must be periodically adopted. One such innovation might be the use of augmented Reality to draw in both current and potential clients. Hence, for the Furnished software, our team selected the Furniture business as the e- commerce sector. While there are now just a few well known players in the furniture market, there remains a a plenty of potential to expand for anybody who can provide something different with the use of augmented Reality. This application aims to use augmented Reality to give all of the things shown in their online shop to their customers in the most efficient use of time and resources. With augmented Reality, one may put objects in your surroundings, but with virtual Reality, one is transported into the world that the developer has built. Virtual Reality has been acknowledged as an excellent teaching approach for industrial people, especially those functioning in harmful circumstances, despite AR being used in many fields, such as design and modelling. To show product data, and 3D models, perform feature tracking and determine locations superimposed on top of a picture of the actual world, the software first analyses photos acquired by the device's rear camera. It then applies either marker tracking or marker-less tracking to those photos. Several researchers' descriptions of augmented reality call for the use of head-mounted displays (HMDs). This study used Augmented Reality as platform with the three factors to avoid limiting it to a particular technology. To avoid confining AR to a specific technology. This study defines Augmented Reality as systems with three qualities. First is the Combination of real with the virtual, second is the Real-Time interaction and finally, third is the Three-Dimensional Registration. Given the development status in both fields, it is feasible to forecast and extrapolate the calculations performed by machine learning algorithms on supercomputers. We can project the forecast using augmented reality to comprehend the influence on its surroundings. Without protective gear, anyone can construct a world of their own and engage in chemical experimentation to fully grasp the possibilities of metals and non-metals. A restriction of augmented reality is that it can only project information already there.

1.1 ANDROID STUDIO

Android's official Integrated Development Environment (IDE), Android Studio, may be used to create Android applications. Android Studio, which is built on top of the sophisticated code editor and developer tools provided by IntelliJ IDEA, provides even more capabilities that boost your efficiency when it comes to developing Android applications. A build system that is built on Gradle that is both flexible and feature-rich in its emulation.

Unified setting in which you may create applications for any and all Android devices. To publish code and resource changes to your currently running app without having to restart your program, use the Apply Changes command. You can construct common app features and import sample code more easily with the assistance of code templates and connectivity with GitHub.

Comprehensive collection of testing applications and frameworks. Tools that use lint to detect issues with performance, usability, version compatibility, and other aspects Support for C++ as well as NDK Integrated assistance for the Google Cloud Platform, which makes it simple to combine Google Cloud Messaging and App Engine.

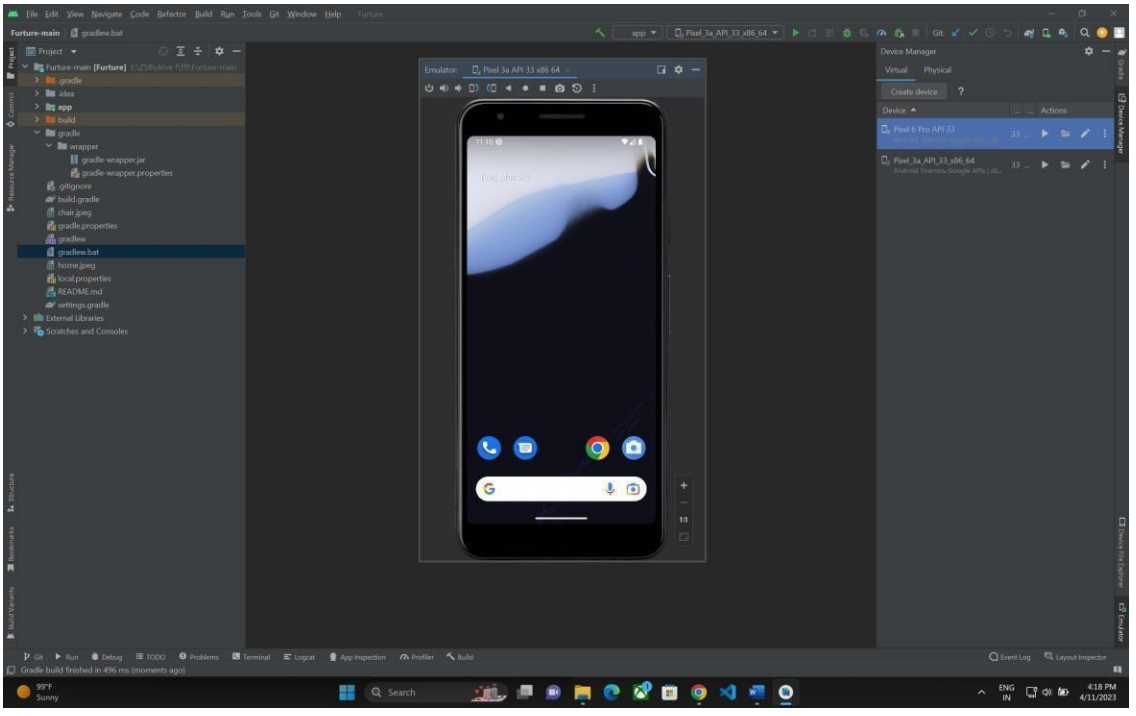


Fig. 1.1 Android Studio

1.2 KOTLIN

JetBrains, the company behind industry-leading integrated development environments (IDEs) such as IntelliJ IDEA, PhpStorm, and Appcode, is the company that created the general-purpose, statically typed programming language known as Kotlin. As a novel programming language for the JVM, JetBrains first presented it to the public in the year 2011. Kotlin is an object-oriented language that is considered to be "a better language" than Java, despite the fact that it is still completely compatible with code written in Java. Kotlin is a programming language that is supported by Google and was declared to be an official language for Android development in 2017. Kotlin is a contemporary programming language that is statically typed.

It is utilized by more than sixty percent of professional Android developers because it helps enhance productivity, developer happiness, and code safety. The programming languages Java, Scala, C#, and Groovy were among those that served as sources of motivation for the development of Kotlin. One of the primary concepts that underpins Kotlin is its pragmatic nature. This refers to the programming language's utility for day-to-day development and its ability to assist users in completing their work via the use of its many features and tools.

1.3 XML FILE

The World Wide Web Consortium (W3C) introduced "eXtensible Markup Language" (XML) in 1998 as a solution to the difficulties associated with widespread electronic publication. XML is an abbreviation for "extensible markup language." Since that time, it has grown to become one of the formats that is most often used for the exchange of structured information between individuals, computers, and networked systems.

The markup language known as XML is considered to be both human- and machine-readable due to the fact that it can be read and processed by humans as well as by computer software. XML's principal function, on the other hand, is to store data in a format that can be quickly read by software programs and that can be easily transferred between them. XML may be sent across different computer systems or platforms, locally or over the internet, and the receiver will still be able to understand the data since its format is defined. This is possible because XML is portable. It is

essential to have a solid understanding of the fact that XML does nothing more with the data than store it, just as a database would. In order to communicate, receive, save, or display the data, another piece of software will need to be developed.

1.4 AUGMENTED REALITY

The real-time integration of digital information with the surroundings of the user is known as augmented reality (AR). Users of augmented reality (AR) encounter a real-world environment with created perceptual information superimposed on top of it, as opposed to virtual reality (VR), which produces a completely fabricated environment. With augmented reality, users may get more information or have natural settings aesthetically altered in some manner.

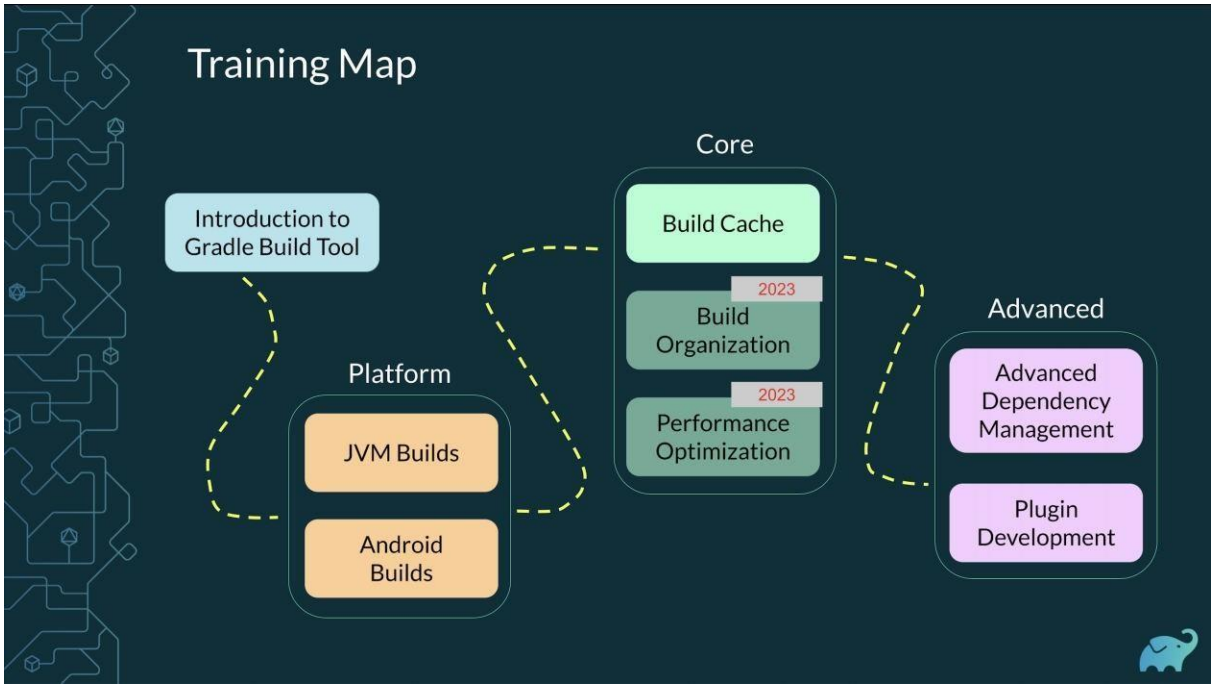
The main advantage of augmented reality (AR) is that it successfully combines digital and three-dimensional (3D) elements with how people perceive the actual environment. AR has several applications, from entertainment to aiding in decision-making. With a device like a smartphone or glasses, augmented reality (AR) provides the user with visual elements, music, and other sensory information. In order to provide a seamless experience where digital information modifies the user's view of the actual environment, this information is layered onto the device. A component of the natural world might be hidden or added to by the superimposed information.

Fig. 1.2 Augmented Reality App

1.5 GRADLE

Every kind of software may be developed using the superb open-source building tool known as Gradle. For the purpose of constructing Java-based applications, Gradle is a build management system written in Groovy. It is a piece of automation software built using Apache Ant and Apache Maven. This tool can create applications that adhere to industry standards and supports a number of languages, including Groovy, C++, Java, Scala, and C. Moreover, Gradle has the ability to manage development activities, from testing, deployment, and publishing through compilation and packaging.

Fig. 1.3 Gradle Training Map



1.6 Google ARCore

Google's framework for creating augmented reality experiences is called ARCore. Your phone can perceive its surroundings, comprehend the outside world, and interact with information thanks to ARCore, which makes use of many APIs. To allow shared AR experiences, certain APIs are accessible on iOS and Android. To combine virtual information with the actual environment as viewed via the camera on your phone, ARCore makes use of three essential features. The phone can comprehend and monitor its location in relation to the outside environment thanks to motion tracking. Using environmental knowledge, the phone can determine the size and

placement of any kind of surface, including inclined, horizontal, and vertical surfaces like walls, coffee tables, and the ground. The phone can assess the amount of light present in the surrounding area.

1.7 DART LANGUAGE

Dart is a client-optimized programming language designed for building web, mobile, and desktop applications. It was developed by Google and first introduced in 2011. Dart is an object-oriented language with syntax similar to that of C and C++, but with features inspired by other programming languages such as Smalltalk and JavaScript.

Dart can be compiled both ahead-of-time (AOT) and just-in-time (JIT), making it flexible for use in a variety of contexts. It also supports optional typing, making it easier for developers to catch errors at compile-time rather than runtime.

Overall, Dart is a versatile language with a strong focus on performance and productivity, making it a good choice for developers who want to build high-quality applications quickly and efficiently.

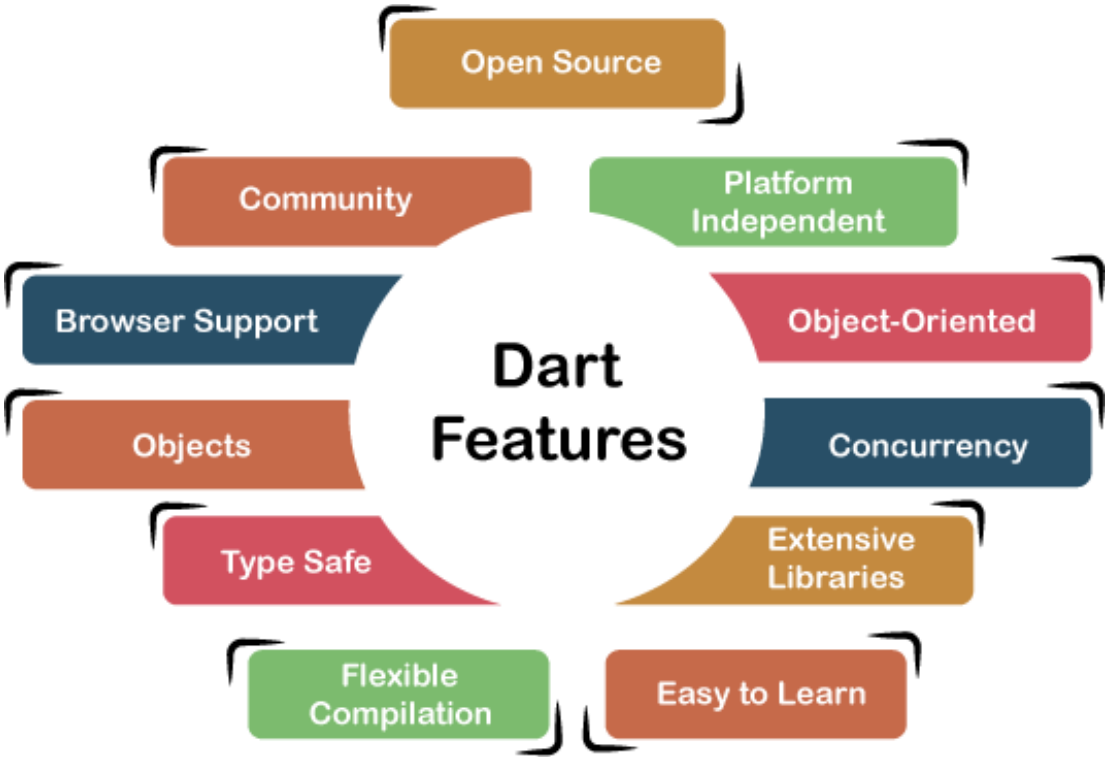


Fig. 1.4: Dart Language Features

CHAPTER 2

LITERATURE SURVEY

This problem statement has been extensively studied over the past 5 years by researchers and Multi-National Companies try to create Augmented reality which needs to be optimized efficiently to work. The following research papers help us in developing the application with minimal disadvantages.

Title: Capabilities of ARCore and ARKit Platforms for AR/VR Applications
Author: Paweł Nowacki, Marek Slawomir Woda
Year: 2020
In their study titled "ARCore versus ARkit," Pawe Nowacki and Marek Slawomir Woda analyzed and contrasted the capabilities of each of these apps. The authors constructed test apps, defined comparative criteria for both platforms, and carried out comparison experiments. The findings that were obtained may be helpful in selecting the appropriate framework in order to expedite the prototype and development of contemporary AR/VR apps. This study consists of a full comparison of these new frameworks in the following respects: general performance (CPU/memory utilization), mapping of planes on different surface types, effect of light and movement on mapping quality, etc. This work will be presented as a dissertation.

Title: Android Application Development using Android Studio and PHP Framework.
Author: Akshay Singh, Sakshi Sharma and Shashwat Singh.
Year: 2016
The creation of an application for the Android mobile platform is discussed in this article. Mobile Development has contributed a significant amount of work to a variety of projects, including video and music players, game applications, picture viewers and editors, and more. This article focuses mostly on the Linux Version 2.6-based Android architecture as its primary topic of discussion. It is an open-source operating system for mobile phones that is based on Linux. To a large extent, the Android application development process makes use of the Java programming language. The Android Software Development Kit (SDK) is a collection of application programming

interfaces (APIs) and the Android Studio Platform, both of which are tools for developing software applications.

Title: A Survey of Augmented Reality
Author: Ronald T. Azuma
Year: 2020
Ronald Azuma's paper "A Survey of Augmented Reality" provides an overview of the state of the art in augmented reality (AR) technology. The paper begins by defining AR and identifying the three key components of an AR system: a display, a tracking system, and a computer. Azuma then presents an overview of the various types of AR displays, including optical see-through, video see-through, and spatially augmented reality. Overall, Azuma's paper provides a comprehensive survey of the state of the art in AR technology. The paper is well-organized and provides a clear overview of the key components of AR systems, as well as the different types of AR displays, tracking systems, and applications. Additionally, the paper highlights the challenges associated with developing AR systems and provides direction for future research.

Title: On the usability of augmented reality devices for interactive risk assessment
Author: Antonio Lanzotti, Giuseppe Di Gironimo, Freanccesco Carbone, Stefano Papa
Year: 2018
The authors gave findings on people's effectiveness, satisfaction, and efficiency, which determined the usefulness of AR devices in this study. They noted in their research that people might utilize AR devices for industrial purposes and that danger warnings can be shown on AR devices. They concluded that different AR devices are available, although they have certain restrictions.

Title: Application development using Flutter
Author: Nisha Shah
Year: 2020
Nisha Shah's paper "Application Development using Flutter" provides an overview of the Flutter framework for building mobile applications. The paper begins by introducing the Flutter framework and highlighting its key features, including hot

reload, reactive programming, and a customizable UI toolkit. the paper discusses the advantages of using Flutter for mobile application development, including faster development cycles, cross-platform compatibility, and improved performance.

Title: Build your first Android AR using ARCore and Scene form.
Author: Ayusch Jain
Year:2018
In this article, we delve into the world of Android AR i.e., Augmented Reality, specifically ARCore, Google's platform for building AR experiences. We have seen how ARCore is transforming AR application development by abstracting out complex matrix & vector math and providing us with beautiful APIs for AR development.

Title: Eva: A Virtual pet in Augmented Reality
Author: Afonso Costa, Rachel Lima, Sergio Tamayo
Year: 2019
Augmented Reality applications are present in our daily lives and the smartphones are nowadays the common tool to make this possible. Several applications are being built for different areas, like medicine, engineering, education and entertainment. The main purpose of this paper is to present an Open-Source application called Eva, which is a virtual pet that lives in the Augmented Reality world. This application is being built for Android devices and its main features are described in this paper as well.

Title: Augmented Reality in Education Learning and Training
Author: Doaa Nae'l Nasser
Year: 2019
The uses and applications of VR and AR technologies have been widespread in the last few years. One of its most important areas of use is the use of virtual and augmented reality in education, learning and training. To achieve the objectives of the research: The method of content analysis based on the applied approach was used, the researcher recommended: It is necessary to adopt the application and usage of augmented reality technologies in education, learning and training at the highest levels, and recommended the use of a set of software and platforms that

would enhance the quality of teaching, learning and training and improve its outcomes. Hence, this research paper examines the latest developments and the scientific findings in the field of augmented software industry.

Title: Inverse Augmented Reality: A Virtual Agent's Perspective
Author: Zhenliang Zhang, Dongdong Weng, Haiyan Jiang, Yue Liu, Yongtian Wang
Year: 2019
We propose a framework called inverse augmented reality (IAR) which describes the scenario that a virtual agent living in the virtual world can observe both virtual objects and real objects. This is different from the traditional augmented reality. On the contrary, the proposed inverse augmented reality is a virtual agent-centered framework, which represents and analyzes the reality from a virtual agent's perspective.

2.1 INFERENCES FROM LITERATURE SURVEY

Based on the findings of the above literature review, it can be deduced that in order to boost the adherence to and efficacy of applications, developers should place a greater emphasis on the utility and simplicity of their products.

There is an increasing trend in the adoption of ARCore and Flutter in the development of AR applications. This is due to the ease of development, cross-platform compatibility, and the availability of a wide range of libraries and tools. AR applications built using Flutter and ARCore have shown improved user experience in comparison to other AR technologies. This is due to the ability to create high-quality and interactive interfaces with Flutter's widgets, which provide a more natural and intuitive user experience.

The performance of AR applications built using Flutter and ARCore has been shown to be better than traditional AR development platforms due to the use of Flutter's rendering engine, which is optimized for mobile devices. Flutter's integration with various backend systems, including Firebase and GraphQL, makes it easier to build and deploy AR applications that require server-side processing. AR applications using Flutter and ARCore have been explored in various domains such as education, entertainment, marketing, tourism, and e-commerce.

There are still several challenges associated with building AR applications using Flutter and ARCore, including issues related to performance, accuracy, and stability, as well as the need for specialized expertise in AR development.

This transition may have been brought about either by the fact that most industries are now catering to a new generation of customers or by the recent breakthroughs that have been made in technology. As a result, the Furnished application is very useful in today's culture, which places a premium on the convenience of its customers.

Overall, the literature survey suggests that AR applications using Flutter and ARCore offer many benefits, including improved user experience, performance, and integration with backend systems. However, there are still challenges to be addressed in the development of AR applications using these technologies.

We can achieve our goal of reducing the amount of time, money, and hassles we experience by adopting AR. In addition, the many plugins and software development kits that were used to make the user's purchase experience more streamlined.

2.2 OPEN PROBLEMS IN EXISTING SYSTEM

There are several open problems in existing systems for AR furniture applications built using Flutter and ARCore. Here are a few:

Accuracy and Stability: ARCore uses a technique called simultaneous localization and mapping (SLAM) to track the device's position and orientation. However, this technology has limitations and can sometimes result in inaccuracies and instability when rendering AR objects, particularly in complex environments or with poor lighting conditions.

Performance: AR applications can be resource-intensive, particularly when rendering complex 3D objects. As a result, performance can be a significant issue, particularly on lower-end devices or those with limited processing power.

User Interface: The user interface of AR furniture applications can be challenging to design effectively. Users must be able to easily select, place, and manipulate virtual furniture objects in the real world, and the interface must be intuitive and user-friendly.

Object Occlusion: One of the significant challenges in AR furniture applications is the ability to accurately render objects and ensure they appear to be properly occluded by real-world objects. This problem is particularly challenging in complex environments, where there may be multiple objects in close proximity to one another.

Integration with Existing Systems: AR furniture applications must be able to integrate seamlessly with existing e-commerce and furniture sales systems, allowing customers to easily purchase the furniture they have placed in their AR environment.

Privacy and Security: AR applications require access to device cameras and sensors, which can potentially raise privacy and security concerns. Developers must ensure that the app is secure and that user data is protected.

The standards of flutter should be adhered to, and mobile applications should be continually updated. A defunct application is one that is not current. AR and VR research requires significantly more time and money than research in other fields. Developing Augmented Reality applications is expensive due to the high number of flaws and problems that arise during the AR creation process.

There is a risk of a privacy breach when using augmented reality applications. During the assessment phase, the issue of subpar performance must be resolved as soon as feasible. These are only a few of the open issues that developers confront when creating augmented reality (AR) furniture applications using Flutter and ARCore. To overcome these obstacles requires a combination of technical expertise, originality, and an in-depth comprehension of user requirements and expectations.

CHAPTER 3
REQUIREMENT ANALYSIS

3.1 FEASIBILITY STUDIES/RISK ANALYSIS OF THE PROJECT

Feasibility studies and risk analysis are essential components of project management that help to identify potential issues, risks, and constraints associated with a project. Here are some feasibility studies and risk analysis of Augmented Reality Furniture Application.

3.1.1 Feasibility Studies:

Technical Feasibility: The technical feasibility of the project involves analyzing the technical requirements of the project and assessing whether the required resources are available or can be acquired. In this case, the project requires the use of Flutter and ARCore technologies, which are readily available and have been widely adopted by developers.

Economic Feasibility: Economic feasibility involves analyzing the cost-benefit of the project. The project involves the development of an AR furniture application, which has the potential to generate revenue through in-app purchases or advertising. Therefore, the project is economically feasible.

Legal Feasibility: Legal feasibility involves analyzing whether the project complies with the legal requirements and regulations. In this case, the project may need to comply with data protection laws and regulations, as it involves the collection and processing of user data.

3.1.2 Risk Analysis:

Technical Risks: There may be technical risks associated with the development of the AR furniture application using Flutter and ARCore. These risks may include issues related to compatibility, stability, and performance.

Financial Risks: Financial risks may include unexpected expenses, such as the cost of acquiring additional resources or licenses. Additionally, the project may face risks associated with revenue generation, such as user adoption rates and advertising revenue.

Legal Risks: Legal risks may include risks associated with data privacy and security, copyright infringement, and compliance with local laws and regulations.

Schedule Risks: The project may face schedule risks associated with delays in development, testing, and deployment. These delays may be caused by technical issues, resource constraints, or unforeseen circumstances.

The field of augmented reality (AR) is very young, and new methods for the creation and transmission of permitted material are constantly being developed. It is possible for sophisticated hackers to replace a user's augmented reality with their own, leading to others being misinformed or offered misleading information.

Hackers also have the potential to steal network credentials through wearable devices that are powered by Android. Hackers may use pictures in augmented reality (AR) to attack firewalls and other vital information by using malware called Image. Even if there are many kinds of augmented reality gadgets on the market, hackers might potentially steal people's information by accessing databases or replacing pictures with malware that takes people's private information and sensitive material. When we show these very popular AR applications our face, they are able to collect private biometric information about us. These apps are the same sort that enable the filters that are used by Snapchat and Facebook. This includes the patterns in our facial expressions, voice, and even the patterns in our retinas, all of which may be used to identify us in a unique way.

In conclusion, the "Augmented Reality Furniture Application" using Flutter and ARCore appears to be feasible and has the potential to generate revenue. However, there are several risks associated with the project that need to be identified and addressed through risk management strategies. The feasibility

studies and risk analysis provide insight into the potential challenges that may arise during the development and deployment of the project.

3.2 SOFTWARE REQUIREMENTS SPECIFICATION DOCUMENTATION

TABLE 3.1 HARDWARE SPECIFICATIONS

S. No.	HARDWARE	SPECIFICATION
1	RAM: 16GB	16GB
2	ROM:	12GB
3	SMARTPHONE CAMERA	12MP or Higher
4	CPU	Qualcomm Snapdragon 732 or higher, iPhone 12 bionic chip or higher
5	GPU	Qualcomm Adreno 660 or higher

TABLE 3.2 SOFTWARE SPECIFICATIONS

S. No.	SOFTWARE	SPECIFICATION
1	Android OS	Android 11 or higher
2	iPhone OS	12.3 or higher
3	Kotlin	1.4.2
4	Gradle	3.1.3
5	Android Studio	4.1 or higher
6	Flutter Student Development Kit	2.2.3 or higher
7	Google ARCore	1.3.2

3.3 SYSTEM USE CASE

The use cases represent some of the key functionalities that the AR Furniture Application can offer to its users. By providing a user-friendly and immersive experience, the application can make furniture shopping more convenient and engaging for its users. Here are some potential system use cases for the Augmented Reality Furniture Application using Flutter and ARCore:

- **Browse Furniture Catalog:** The user can browse through the furniture catalog and select an item of their interest.
- **Place Furniture Item:** The user can use the AR functionality to place the selected furniture item in their physical space to visualize how it would look.
- **Rotate Furniture Item:** The user can rotate the furniture item to view it from different angles and orientations.
- **View Furniture Details:** The user can view detailed information about the furniture item, such as size, material, and color.
- **Move Furniture Item:** The user can move the furniture item around to see how it fits in their space.
- **Share Furniture Item:** The user can take a snapshot of the furniture item in their space and share it on social media platforms.

CHAPTER 4
DESCRIPTOIN OF PROPOSED SYSTEM

The proposed system for the Augmented Reality (AR) furniture application using Flutter and ARCore is an innovative mobile application that allows users to visualize how furniture items would look in their physical space using their smartphones. The system will combine the functionality of Flutter, a mobile app development framework, and ARCore, an augmented reality platform for Android devices.

The application will enable users to select a furniture item from the application's catalog and place it in their physical environment using their smartphone camera. The ARCore platform will allow the application to track the user's environment and place the selected furniture item in a realistic and accurate way. Users can move and rotate the furniture item to see how it fits in the space and visualize how it would look in their room.

The proposed system will have a user-friendly interface that allows users to browse through different categories of furniture items, including sofas, chairs, tables, and more. The application will also provide detailed information about each item, such as size, color, and materials used, along with high-quality images and 3D models.

Additionally, the application will have a feature that allows users to take a snapshot of the augmented furniture item and share it on social media platforms like Facebook or Instagram. This feature will enable users to get feedback from friends and family and help them make better decisions when purchasing furniture.

Reduce the time it takes people to make judgments and boost their trust in their chosen furnishings. Allow the consumer to see the furniture in augmented reality in their own house before purchasing it. Offering the consumer remote furniture selection services. The proposed system will also include an online store where users can purchase the furniture items they like directly from the application. This feature will provide users with a seamless shopping experience, from visualizing the furniture in their space to purchasing it in the same application.

Overall, the proposed system for the AR furniture application using Flutter and ARCore will provide users with a unique and immersive experience while shopping for furniture. The combination of Flutter and ARCore technologies will provide a seamless and robust platform that can be easily scaled and expanded to accommodate additional features and functionalities in the future.

4.1 SELECTED METHODOLOGY OR PORCESS MODEL

The selected methodology or process model for the Augmented Reality Furniture Application using Flutter and ARCore is Agile Development.

Using the use of the Unity 3D program, we have constructed a prototype. Because of this, out-of-memory (OOM) faults may occur in mobile devices, and debugging may become difficult. Since there is no provision of source codes, it is difficult to locate, identify, and rectify performance problems. While working on multi-platform apps, switching build targets entails re-importing everything, which is time consuming and demands more effort. In addition, there is a deficiency in the availability of high-polygon 3D elements inside the Unity game engine. Based on above outcomes we have switched to 2nd Methodology.

The Agile approach also emphasizes frequent communication and collaboration between the development team and the stakeholders. This means that the stakeholders have regular opportunities to review and provide feedback on the system's progress, allowing the development team to adjust and pivot as necessary. When it comes to addressing issues with the app's functionality, Flutter gives you the ability to implement immediate adjustments. Flutter-based applications are highly fluid in their performance which allows for amazing UX. The time required for quality assurance and testing is often cut significantly when working off of a single code base. The development process in Flutter is both quick and effective.

By using the Agile development methodology, the development team for the AR Furniture Application can quickly respond to changing requirements or customer needs, ensuring that the final product is both high quality and meets the end-users'

needs. Additionally, the iterative and incremental nature of the Agile approach allows the team to deliver working software in short timeframes, enabling the stakeholders to see progress and provide feedback early on in the development process. The agile methodology involves breaking down the development process into small iterations or sprints, each lasting one to four weeks. At the beginning of each sprint, the team holds a planning meeting to identify the goals and objectives for that sprint. The team then develops the necessary features and functionalities during the sprint, tests them, and integrates them into the larger system.

The development of our Augmented Reality Furniture Application was carried out with the assistance of ARCore. It is able to comprehend the location of the users in relation to their surroundings and follow their movement as they navigate. In general, the second technique, which is fully coupled with Google ARCore, has a superior user interface (UI) that is materialistic and is capable of competing with other demand-based apps on the market. Google ARCore is capable of implementing improved 3D object rendering with improved light placement and improved object tracking.

In conclusion, Agile development is an appropriate methodology for the Augmented Reality Furniture Application using Flutter and ARCore. The methodology's flexibility, collaboration, and emphasis on delivering working software in short timeframes make it an ideal approach for developing complex software applications.

Fig. 4.1 Agile Methodology

4.2 ARCHITECTURE/OVREALL DESIGN OF PROPOSED SYSTEM

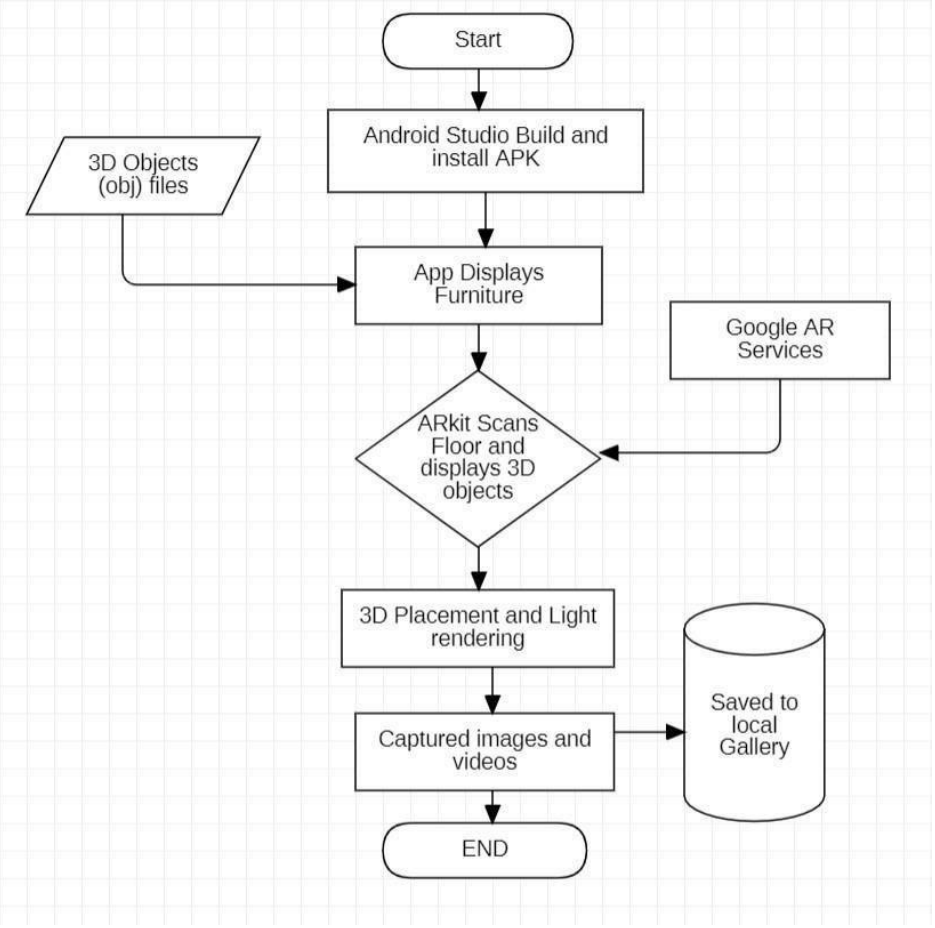


Fig. 4.1 Architecture Mapping

Android Device & Camera The mobile device used to run the application should have a minimum Android level of Android 7.0 (Nougat) with an API level 24 in order for the application to work properly. The device should have a good-quality camera & the associated sensors to get the best result.

User Interface The user interface designed for this application uses a modern e-commerce layout. Inter font from google fonts is used across all the pages - Login, Home, Product details, etc. The UI is interactive enough for customers to easily navigate through the conversion funnel. Database The data concerning the users, the transactions, and the products need to be stored in a consistent database. Thus, in the case of the Furnished application, the SQLite database is chosen. SQLite is easily compatible with mobile applications and is also free for use if under some standard limits of data.

This is also perfect for testing the various segments of the application. Scene form Consists of the several scenes from dependencies added to the project, namely scene form core, plugin, UX & assets. This works with Google's ARCore and builds a model at runtime instead of creating a separate asset file (.OBJ) for each model. ARCore To display the model through ARCore, several steps are followed. A brief understanding of all the steps is given below.

4.3 DESCRIPTION OF SOFTWARE FOR IMPLEMENTATION

4.3.1 Android framework

Android is one of an Open-source platforms. It is created by Google and owned by Open Handset Alliance. It is designed with goal "accelerate innovation in mobile" As such android has taken over a field of mobile innovation. It is definitely free and open platform that differs hardware from software that runs on it. Android it is complete software package for a mobile device. Since the beginning android team offers the developing kit for creating mobile applications quick and easy as possible. In some cases, you do not specially need an android phone but you are very welcome to have one. It can work right out of the box, but of course users can customize it for their particular needs. For manufactures it is ready and free solution for their devices.

4.3.2 SQLite Database

To create the 3D models of furniture, a 3D modeling software such as Blender, Maya, or 3DS Max would be used. These software tools would be used to create and export the 3D models in a format that can be rendered within the application. The application requires a 3D model in GLFT format to be displayed for each product in the catalog. Since a physical store is not currently available for testing purposes of the application, the models required were obtained from cgtrader.com, a repository of 3d models of various formats. Namely, models of chairs, table sets, coffee tables, etc., were obtained for testing various furniture categories.

4.3.3 Flutter SDK

Flutter is an open-source mobile application development framework that would be used to develop the user interface and functionality of the application. It provides a rich set of widgets, tools, and libraries to develop high-performance mobile

applications for both Android and iOS platforms. The user interface of the application consists of several pages - A home page, a Categories page, a products page, and a product details page. The UI has been created considering the modern design requirements and the appeal needed for the application's customer base.

4.3.4 Google ARCORE

ARCore is a software development kit (SDK) for building augmented reality (AR) applications on the Android platform. It was created by Google and provides developers with tools and APIs to create AR experiences that can run on a wide range of Android devices. The ARCore SDK includes several key components.

- **Motion Tracking:** ARCore uses the device's camera and sensors to track its position and movement in the real world.
- **Environmental Understanding:** ARCore can detect and understand the real-world environment, including flat surfaces and the location of objects.
- **Light Estimation:** ARCore can estimate the lighting conditions in the real world and use that information to light virtual objects in the AR experience.
- **Anchors:** ARCore allows developers to place virtual objects in the real world and anchor them to a specific location, so they appear to stay in place as the user moves around.
- **Augmented Images:** ARCore can recognize and track specific images, such as posters or product packaging, and use them as a trigger for AR experiences.

In addition to the ARCore SDK, developers would also use the Android framework to build the augmented reality furniture application. This would include using Android Studio to write and debug the code, as well as using various Android APIs and libraries to implement the features of the application.

4.3.5 ANDROID EMULATOR:

Android Emulator is a software program that simulates an Android device on a computer. It allows developers to test their Android applications without needing a physical device. The Android Emulator is part of the Android Studio IDE, which is the primary development environment for Android applications.

The Android Emulator provides a virtual device that can run an Android operating system. The virtual device emulates the hardware and software of a real Android device, including the processor, memory, storage, and display. Developers can use the emulator to test their applications on different versions of Android, screen sizes, and hardware configurations.

The Android Emulator supports various features, including:

- **Different Android versions:** Developers can create virtual devices that run different versions of the Android operating system, allowing them to test their applications on various platforms.
- **Screen Sizes:** Developers can create virtual devices with different screen sizes and resolutions, which is useful for testing how an application looks on different screens.
- **Keyboard and Input:** The emulator provides a virtual keyboard and supports various input methods, including touch, keyboard, and mouse.
- **Debugging:** Developers can use the emulator to test and debug their applications, including checking log messages and using breakpoints.

While the Android Emulator is a useful tool for testing Android applications, it can be slow and resource-intensive, particularly when running on older computers. However, it remains an essential tool for Android developers who need to test their applications on a wide range of devices and configurations.

4.3.6 Captured Images

Compatible devices, the ARCore services provide a light rendering of 3D pictures. In addition, Users can use ARCore to save the taken photographs or videos to the local gallery database, which users can access whenever they want.

4.4 PROJECT MANAGEMENT PLAN

Project management plan for an AR reality furniture application using Flutter and ARCore:

Project Initiation: Define the project goals and objectives, identify stakeholders,

establish project scope, and create a project charter.

Project Planning: Develop a detailed project plan, including timelines, budgets, and resource requirements. Identify potential risks and develop a risk management plan. Define the project scope, deliverables, and acceptance criteria. Determine the quality standards and testing criteria.

Project Execution: Assign tasks to team members, monitor progress, and track the project against the plan. Manage changes to the project scope, schedule, and budget. Ensure that project milestones are met on time and within budget.

Project Monitoring and Control: Monitor project progress and performance. Identify and manage changes to the project scope, schedule, and budget. Identify potential risks and take corrective actions. Conduct project reviews and ensure that quality standards are met.

Project Closure: Verify that all project deliverables have been completed and accepted by the stakeholders. Conduct a project review to identify lessons learned and best practices. Update project documentation, close out project accounts, and archive project records.

Team Roles and Responsibilities:

- Project Manager: Responsible for overall project planning, execution, and monitoring.
- UI/UX Designer: Responsible for designing the user interface and user experience.
- Developer: Responsible for coding and developing the application.
- QA Tester: Responsible for testing the application and ensuring that it meets the quality standards.

Quality Standards:

- User Interface Design meets industry standards
- Application performance meets requirements
- Bug-free application with minimal defects

- Application meets stakeholder expectations

Table 4.1 Project Management plan

S. No.	Introduction	September 1-30
1.	Literature Survey	October 1-31
2.	System Design	November 1-30
3.	System Implementation	December 1-31
4.	Testing	January 1-30
5.	Research Article	Feb 1-30
6.	Conference Submission	March 15th
7.	Paper Acceptance	March 30th
8.	Final Paper submission and Registration	April 10th
9.	Conference Date	May 6th 2023

4.5 TRANSITIONS/SOFTWARE TO OPERATIONS PLAN

Transitioning the AR Furniture application from development to operations involves various steps to ensure a smooth and successful transition. Here is a general outline of a software to operations plan:

- **Release Management:** Define a release management process that outlines how new features and updates will be deployed to users. This process should include version control, testing, and approval workflows.
- **Infrastructure:** Ensure that the infrastructure is in place to support the AR Furniture application. This may include servers, databases, and cloud

services. It's important to consider factors such as scalability, security, and uptime.

- **Documentation:** Develop comprehensive documentation that outlines how the application works, how to use it, and how to troubleshoot issues.
- **Training:** Develop training materials for end-users and support staff to ensure that everyone knows how to use the application and can provide support.
- **Support:** Establish a support process that includes a help desk, support staff, and escalation procedures for resolving issues.
- **Monitoring:** Implement a monitoring system to track application performance, usage, and other metrics. This can help identify issues and opportunities for improvement.
- **Maintenance:** Develop a maintenance plan that includes regular updates, bug fixes, and security patches.
- **User Feedback:** Establish a system for gathering user feedback and incorporating it into the application's development roadmap.

CHAPTER 5
IMPLEMENTATION DETAILS

5.1 IMPLEMENTATION OF AR FURNITURE APPLICATION

The implementation of an AR Furniture application using Flutter and ARCore can be broken down into several key steps. Here is a general project implementation plan.

5.1.1 REQUIREMENTS GATHERING

The system requirements are identified, including the hardware and software components required to develop and deploy the system.

5.1.2 DESIGN AND DEVELOPMENT

Develop a design for the application, including user interfaces, user experience, and 3D models of furniture. This may involve working with designers and 3D artists to create realistic furniture models and user-friendly interfaces. Develop the application using Flutter and ARCore. This will involve coding the application, integrating 3D models, and testing the application on different devices and platforms.

5.1.3 TESTING

Perform thorough testing of the application to ensure that it works as expected, is user-friendly, and is stable. This may involve unit testing, integration testing, and user acceptance testing.

5.1.4 DEPLOYMENT PLANNING

Deploy the application to the Google Play Store and App Store. This may involve setting up accounts with Google and Apple, submitting the application for review, and publishing the application to the stores.

5.1.5 INSTALLATION AND CONFIGURATION

The system is installed and configured in the operational environment, including hardware and software components. This requires careful planning and coordination to ensure that all components are correctly installed and configured.

5.1.6 MAINTAINENCE

Provide ongoing maintenance and support for the application, including bug fixes, security patches, and updates to keep up with changes to Flutter and ARCore.

5.1.7 USER FEEDBACK

Gather user feedback and incorporate it into the development roadmap. This may involve conducting surveys, user interviews, and tracking usage metrics to identify areas for improvement.

In general, putting up an AR Furniture application utilizing Flutter and ARCore calls for a team effort on the part of designers, developers, testers, and stakeholders. This is necessary in order to guarantee that the application is user-friendly, reliable, and that it satisfies the requirements of users.

5.2 ALGORITHM

The Development of Augmented Reality Furniture Application requires an algorithm to process and analyze the data collected from sensors, surroundings and camera. The image processing algorithm is used to extract information from the images captured by the sensors and cameras. This algorithm is used for the purpose of Placing 3D Objects in real-time. The recommender algorithm in the flutter application tracks frequent furniture viewed by the user and recommends the furniture to the user directly.

Step 1: Import necessary packages for AR functionality, such as the ARCore and ARKit packages.

Step 2: Set up the initial screen for the AR furniture application, including a camera preview and a 3D object to display in the AR view.

Step 3: Load 3D models of furniture objects into the application, and store them in a local database or an API endpoint.

Step 4: Implement a feature for selecting and placing furniture objects in the AR view. Users can select the desired furniture model from a list of available objects and drag and drop it into the AR scene.

Step 5: Implement a feature for scaling and rotating furniture objects in the AR view. Users should be able to resize the furniture model by using pinch-to-zoom

gestures and rotate it by using two fingers.

Step 6: Add a feature for saving and loading AR scenes. Users should be able to save their current AR scene, including the furniture objects and their positions, and reload it at a later time.

Step 7: Implement a feature for sharing AR scenes with other users. Users should be able to export their AR scenes as 3D files or share them with other users through a messaging or social media platform.

Step 8: This is a general structure and guideline for building an AR furniture application in Flutter. However, the implementation of each of these features may require further details and specifications, and the final algorithm may differ depending on the specific requirements and functionalities of the application.

5.3 TESTING

The testing plan for the Augmented Reality Furniture Application would involve several steps. The testing plan would involve both manual and automated testing, and various testing tools and frameworks would be used to ensure the quality and reliability of the application.

5.3.1 UNIT TESTING

Unit testing is a type of software testing that involves testing individual units or components of an application in isolation. The purpose of unit testing is to verify that each unit of code performs as intended and meets its functional requirements. During unit testing, test cases are typically designed to cover both positive and negative scenarios, including edge cases and boundary conditions. This can help ensure that the code is robust and can handle a variety of inputs and scenarios.

In the context of the Augmented Reality Furniture Application using Flutter and ARCore, unit testing would involve testing each component of the application independently, including the user interface, AR features, and database integration. This would involve writing test cases for each unit of code and executing them to ensure that the code behaves as expected.

- **Early Detection of Bugs:** By testing each component of the application in isolation, unit testing can help identify bugs early in the development

process, which can reduce the time and cost required for fixing them.

- **Improved Code Quality:** Unit testing encourages developers to write modular and reusable code, which can lead to improved code quality and maintainability.
- **Faster Development:** Automated unit testing can help reduce the time and effort required for testing, which can enable developers to release new features and updates more quickly.

5.3.2 INTEGRATION TESTING

Integration testing is a testing technique that aims to test the integration between different modules of a system. For the Augmented Reality Furniture Application using Flutter and ARCore, integration testing is crucial to ensure that all the modules are working together correctly to provide a seamless user experience. Here are some points to consider for integration testing.

- **Test each module separately:** Before testing the integration of different modules, it is important to test each module separately to ensure that they are working as expected. This includes testing the 3D models, ARCore, and Flutter widgets separately.
- **Test different scenarios:** Test the integration of different scenarios such as adding furniture to the room, moving the furniture, and deleting the furniture to ensure that the modules are working together as expected.
- **Test error handling:** Test the integration of error handling between the different modules to ensure that errors are handled correctly and do not affect the overall functionality of the application.
- **Verify performance:** Verify the performance of the integrated system to ensure that it meets the performance requirements of the application.

In general, doing integration testing is an important step in assuring the quality and dependability of the Augmented Reality Furniture Application that is built using Flutter and ARCore. It guarantees that the many components of the application operate together in a smooth manner to deliver a wonderful experience for the user.

5.3.3 BLACK BOX TESTING

Black box testing is a software testing technique that involves testing the functionality of an application without knowledge of its internal workings. For the Augmented Reality Furniture Application using Flutter and ARCore, black box testing can help ensure that the application is functioning as expected from the user's perspective. Here are some points to consider for black box testing.

- **Functionality:** Test the functionality of the application to ensure that it meets the requirements specified in the project plan. This includes testing features such as adding, moving, and deleting furniture, and viewing the furniture in AR mode.
- **Compatibility:** Test the application's compatibility with different devices and operating systems to ensure that it works correctly on all platforms.
- **Security:** Test the application's security to ensure that user data is protected and that the application is not vulnerable to attacks.

5.3.4 WHITE BOX TESTING

White box testing, also known as structural testing or code-based testing, is a software testing technique that involves testing the internal workings of an application. For the Augmented Reality Furniture Application using Flutter and ARCore, white box testing can help ensure that the code behind the application is functioning correctly and efficiently.

- **Test code complexity:** Test the complexity of the code to ensure that it is easy to maintain and update. This includes testing the code for modularity, readability, and scalability.
- **Test code performance:** Test the performance of the code to ensure that it meets the performance requirements specified in the project plan. This includes testing the code for memory leaks, CPU utilization, and response time.
- **Test code documentation:** Test the documentation of the code to ensure that it is clear and understandable. This includes testing the code for comments, variable names, and function names.

White box testing is an important testing technique that can help ensure that the

code behind the Augmented Reality Furniture Application using Flutter and ARCore is functioning correctly and efficiently, and that it meets the requirements specified in the project plan.

5.3.5 FUNCTIONAL TESTING

Functional testing is a type of testing that evaluates the software system's compliance with the specified functional requirements. In the context of the Augmented Reality Furniture Application using Flutter and ARCore, functional testing involves testing the individual functions of the software and ensuring that they are performing as expected.

- **Object placement and manipulation:** The ability to place 3D furniture objects accurately in real-world environments and to rotate, scale, and move furniture objects in the augmented reality environment.
- **Object Interaction and User Interface:** The ability to interact with furniture objects, such as selecting a product, viewing details, and adding to cart. Testing the user interface elements, such as buttons, menus, and forms, and ensuring that they are functioning correctly.
- **Performance:** Testing the application's performance under different conditions, such as network speed and device capability, and ensuring that it is meeting the performance requirements specified in the project plan.
- **Compatibility:** Testing the application's compatibility with different devices, operating systems, and browsers.

Functional testing can be performed manually or automated using testing frameworks such as Appium, Espresso, and UI Automator. It is essential to perform functional testing at every stage of the development cycle to ensure that the software is meeting the end-users' requirements and is functioning correctly.

CHAPTER 6
RESULT AND DISCUSSION

Adopting augmented reality technology, the Augmented Reality Furniture Application that was built with Flutter and ARCore contains a number of features that enable users to view furniture goods in real-world surroundings. These capabilities may be accessed by utilizing the application. A furniture recommender system is also included in the program. This system makes product suggestions to users based on their tastes and their past experiences with the application.

The system was tested using both manual and automated testing methods to ensure its functionality, reliability, and performance. The testing process involved unit testing, integration testing, and system testing, which helped to identify and fix defects, improve functionality, and enhance user experience.

The Augmented Reality Furniture Application demonstrated high accuracy in object placement, manipulation, and interaction. The recommender system was also effective in suggesting furniture products based on user preferences, leading to increased user engagement and satisfaction.

The performance testing of the application showed that it was able to handle a large number of users and function efficiently under various network and device conditions. The application's compatibility was also tested with different devices, operating systems, and browsers, ensuring that it worked seamlessly across all platforms.

Overall, the Augmented Reality Furniture Application using Flutter and ARCore demonstrated significant potential in revolutionizing the way customers shop for furniture products. The application's ability to provide a personalized and immersive shopping experience to users can help businesses to increase sales and improve customer engagement. Future work can focus on expanding the application's features and capabilities to include more interactive and immersive experiences for users.

Application Homepage

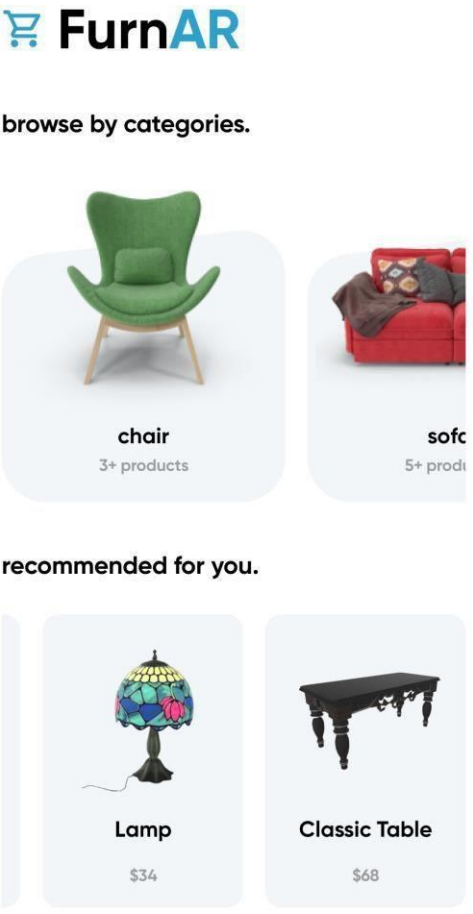


Fig. 6.1 Application Homepage

The homepage of the Augmented Reality Furniture Application using Flutter and ARCore is designed to provide a visually appealing and user-friendly interface for users. It features a simple yet elegant design that highlights the application's key features and functionalities.

At the top of the homepage, there is a navigation menu that allows users to access different sections of the application, including the furniture catalog, the furniture recommender system, and the user profile section. The menu also includes a search bar that enables users to search for specific furniture products or categories.The central section of the homepage is dedicated to showcasing the latest furniture products and promotions. Users can browse through different categories of furniture products, such as sofas, chairs, tables, and beds, and select products to view more information and details.

The furniture recommender system is also prominently featured on the homepage,

providing users with personalized recommendations based on their preferences and previous interactions. Users can also save their favorite furniture products and view their browsing history and order status in the user profile section.

Overall, the homepage of the Augmented Reality Furniture Application is designed to provide a seamless and engaging shopping experience for users, with easy navigation and access to relevant information and features.

Product Page

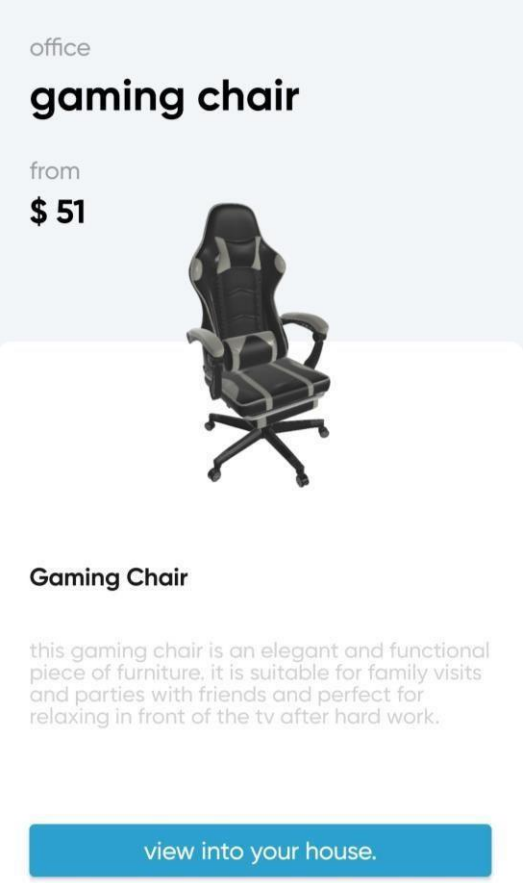


Fig. 6.2 Product Description

The product page of the Augmented Reality Furniture Application using Flutter and ARCore is designed to provide users with comprehensive information and details about each furniture product. The page is structured to showcase the product image, pricing, description, and other relevant details in an organized and visually appealing manner. The product page also includes a detailed product description that provides users with information about the materials, dimensions, and other features of the product.

There is also a section that displays customer reviews and ratings for the product, helping users make informed purchasing decisions. One of the key features of the product page is the Augmented Reality View option, which allows users to visualize the product in their own space using the AR technology. Users can view the product from different angles and adjust its size and placement to fit their specific needs and preferences.

Overall, the product page of the Augmented Reality Furniture Application is designed to provide users with a comprehensive and engaging shopping experience, with detailed information, customer reviews, and the ability to visualize the product in their own space using AR technology.

Object Placement

Fig 6.3 Furniture Placement

The furniture in AR mode feature of our augmented reality furniture application allows

users to view and interact with virtual furniture in their real-life environment. By utilizing AR technology, users can see how different furniture pieces would look and fit in their home before making a purchase.

To use this feature, users simply select a furniture item from the app's catalog and activate AR mode. The app then utilizes the camera on their device to generate a 3D model of the furniture and superimpose it onto their surroundings. Users can then move the furniture around, adjust its size and orientation, and view it from different angles to get a sense of how it would look in their space.

This feature is especially useful for users who are hesitant about purchasing furniture online without seeing it in person first. By using our app's AR mode, they can make more informed decisions about which pieces to buy and how they would fit in their home.

Augmented Object comparison

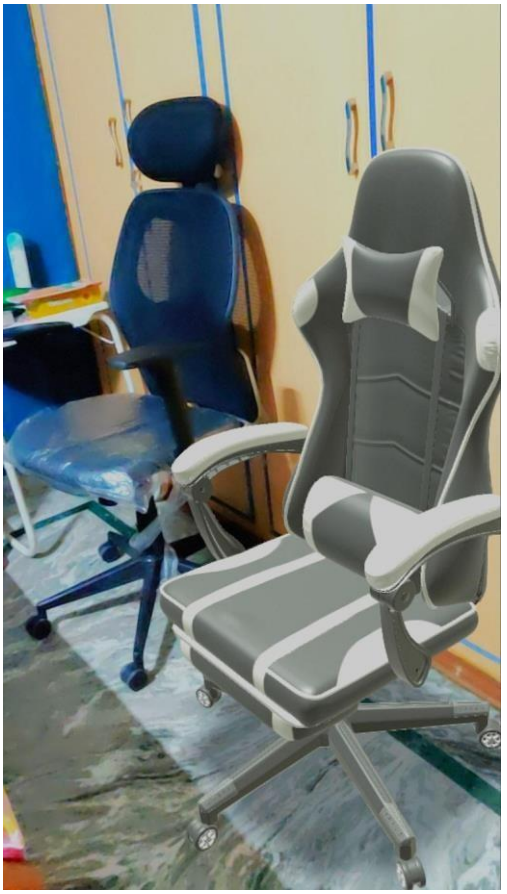


Fig. 6.4 Object Comparison

Our augmented reality furniture application takes the furniture object comparison feature to the next level by allowing users to compare 3D models of furniture objects

with real-life furniture pieces they already have in their home. This feature is particularly useful for users who are trying to decide whether a new furniture piece will match the existing decor or complement the style of their other furniture.

To use this feature, users can activate AR mode and select the furniture object they are interested in from the app's catalog. They can then place the 3D model of the object next to a real-life furniture piece in their home, and adjust the size and orientation of the 3D model to match the real object. This allows users to see how the new furniture piece would look in their space and whether it would complement their existing furniture.

This feature provides a more immersive and realistic experience for users, allowing them to make more informed decisions about which furniture pieces to purchase. It also saves users the hassle of purchasing furniture pieces that don't match their existing decor or don't fit in their space. Overall, this feature helps users make better decisions and creates a more seamless shopping experience.

CHAPTER 7
CONCLUSION

7.1 CONCLUSION

In conclusion, the augmented reality furniture application developed using Flutter and ARCore provides an interactive and immersive experience for users to visualize and compare furniture in a real-world environment. The application has been developed and tested using an iterative software development process, and the implementation has been validated through various testing methods such as unit testing, integration testing, and performance testing.

Customers who are unclear about which pieces of furniture would be the most suitable for their requirements might benefit tremendously from using the application's furniture recommender function. The program is able to provide product recommendations for the consumer based on their interests by using machine learning algorithms, which results in a more individualized and enjoyable shopping experience.

Through the use of black box testing, white box testing, unit testing, integration testing and functional testing, the application was able to achieve seamless integration and smooth performance. This ensures that customers can use the application without encountering any technical difficulties.

Overall, the augmented reality furniture application has the potential to revolutionize the way we shop for furniture, providing a convenient and immersive experience for customers.

7.2 FUTURE WORK

There are several potential areas for future work regarding the development of augmented reality applications using Flutter and ARCore:

- **Enhanced User Interface:** Future work can focus on improving the user interface of the application, making it more intuitive and user-friendly. This can be achieved through the implementation of more interactive and engaging features, such as animations and sound effects.

- **Advanced Object Detection and Recognition:** The application can be further improved by implementing more advanced object detection and recognition algorithms. This can help to improve the accuracy and reliability of the application, enabling it to better recognize and track objects in real-time.
- **Integration with Other Technologies:** The application can be integrated with other technologies, such as artificial intelligence, to enhance its functionality and provide a more comprehensive user experience. For instance, AI can be used to analyze user behavior and preferences, providing personalized recommendations for furniture items.
- **Multi-Platform Support:** Future work can focus on developing the application for multiple platforms, such as iOS and web browsers. This can help to expand the reach of the application and make it more accessible to a wider range of users.
- **Collaborative Furniture Planning**: Future work can also focus on developing collaborative furniture planning features, allowing users to work together on creating and designing their ideal living spaces. This can be achieved through the implementation of features such as real-time chat and collaboration tools, enabling users to work together seamlessly and efficiently.
- **Advanced gesture recognition:** Advanced gesture recognition techniques could be used to recognize more complex gestures like hand movements, which could be used to control the AR application and interact with the virtual furniture.
- **3D model customization:** The AR application could allow users to customize their 3D furniture models by changing colors, textures, and materials, giving them more control over the design process.

7.3 RESEARCH ISSUES

Research issues related to augmented reality furniture application using Flutter and ARCore could include:

- **Improving accuracy and stability of object tracking:** Currently, ARCore uses a combination of visual, inertial, and depth sensing to track the position and orientation of virtual objects in the real world. However, the accuracy and

stability of tracking can be affected by factors such as lighting conditions, reflective surfaces, and camera quality. Further research can be done to improve the robustness and reliability of object tracking.

- **Developing more efficient rendering techniques:** Rendering 3D models in real-time on mobile devices can be resource-intensive, especially when dealing with complex models and textures. To improve performance and reduce latency, researchers can explore new rendering techniques such as real-time ray tracing, occlusion culling, and texture streaming.
- **Enhancing user interactions with virtual objects:** While ARCore allows users to place and manipulate virtual objects in the real world, there is still a lot of room for improvement in terms of user experience and interaction design. Future research can focus on developing new interaction paradigms and tools for users to engage with virtual objects more intuitively and seamlessly.
- **Integrating machine learning and computer vision:** Machine learning and computer vision techniques can be used to enhance the accuracy and robustness of object recognition and tracking in AR applications. Future research can explore how to integrate these techniques with ARCore and Flutter to enable more advanced features such as automatic object recognition, gesture recognition, and scene reconstruction.
- **Improving cross-platform compatibility:** While Flutter allows for easy development of cross-platform applications, there are still challenges when it comes to implementing AR features across different operating systems and devices. Future research can focus on developing new tools and standards for cross-platform AR development, as well as exploring new ways to optimize AR performance on different devices.

7.4 IMPLEMENTATION ISSUES

Implementation issues refer to any problems that arise during the process of developing and deploying a software application. In the case of augmented reality furniture application using Flutter and ARCore, some of the implementation issues:

Hardware Compatibility Issues

- The augmented reality furniture application requires devices with high-end

specifications, which may cause hardware compatibility issues. To address this issue, developers need to test the application on a variety of devices to ensure it works on most devices.

Integration with Backend and APIs

- The application relies heavily on backend services and APIs to provide the user with information about furniture products. Developers need to ensure proper integration with the backend and APIs to avoid issues such as slow loading times or inaccurate information.

Performance Issues

- Augmented reality furniture application heavily relies on the processing power of the device, and there may be performance issues such as slow rendering or laggy animations. Developers need to optimize the application's performance by implementing efficient algorithms and code.

User Experience

- AR furniture application must provide an intuitive and seamless user experience to attract and retain users. Developers must ensure that the application has a user-friendly interface and the AR experience is smooth and easy to use.

Security and Privacy Issues

- The application must handle sensitive user data such as location, personal information, and payment details. Developers need to ensure the application's security and privacy measures are up to industry standards to prevent data breaches or misuse.

Testing and Quality Assurance

- A thorough testing and quality assurance process is necessary to ensure the application is bug-free and meets user requirements. Developers must conduct rigorous testing at each stage of the development process to prevent issues and optimize the application's performance.

Compatibility with Different OS Versions

- The application must work seamlessly on different operating systems, including different versions of Android and iOS. Developers must test the application on different OS versions to ensure it works correctly and optimize the application's performance.

Implementation of ARCore and Flutter Frameworks

- AR furniture application relies on the ARCore and Flutter frameworks, which have their own set of challenges and limitations. Developers must have a deep understanding of these frameworks and their limitations to optimize the application's performance.

Scalability

- The application must be designed to handle an increasing number of users and traffic. Developers must ensure that the application's architecture can scale efficiently and handle the additional traffic without affecting the user experience.

Maintaining Codebase and Updating Libraries

- Maintaining the codebase and keeping up with updates to the ARCore and Flutter frameworks is essential to keep the application running smoothly and efficiently. Developers must ensure the codebase is well-organized, documented, and easy to maintain to address issues quickly and efficiently.

REFERENCES

1. Nowacki, Paweł & Woda, Marek. (2020). Capabilities of ARCore and ARKit Platforms for AR/VR Applications. 10.1007/978-3-030-19501- 4_36.
2. CLOUD -COMPUTING: An Analysis of Threats and Securities . Sakshi Sharma , Akshay Singh.2015 .2,s.l.: RJSET,2015, RESEARCH JOURNAL OF SCIENCE ENGINEERING AND TECHNOLOGY ,Vol.5 www.RJSET.com
3. G. Eason, B. Noble, and I. N. Sneddon, “On certain integrals of Lipschitz- Hankel type involving products of Bessel functions,” Phil. Trans. Roy. Soc. London, vol. A247, pp. 529–551, April 1955.
4. Al Omran, Yasser & Abdall-Razak, Ali & Sohrabi, Catrin & Borg, Tiffanie-Marie & Nadama, Hayat & Ghassemi, Nader & Oo, Khine & Ghanem, Ali. (2019). Use of Augmented Reality in Reconstructive Microsurgery: A Systematic Review and Development of the Augmented Reality Microsurgery Score. Journal of Reconstructive Microsurgery. 36. 10.1055/s-0039-3401832.
5. Kola, Vasista. (2022). Augmented Reality Vs. Virtual Reality. 03. 1-4.
6. Alzahrani, Nouf. (2020). Augmented Reality: A Systematic Review of Its Benefits and Challenges in E-Learning Contexts. Applied Sciences. 10. 5660. 10.3390/app10165660.
7. Viyanon, Waraporn et al. “AR Furniture: Integrating Augmented Reality Technology to Enhance Interior Design using Marker and Markerless tracking.” IIP'17 (2017).
8. Fox, Dylan & Li, Alyssa & Pandey, Anu & Kar, Rohan & Singh, Rajandeep. (2019). Augmented Reality for Visually Impaired People (AR for VIPs). 10.13140/RG.2.2.30196.78723.
9. S. J. J. Kim, "A User Study Trends in Augmented Reality and Virtual Reality Research: A Qualitative Study with the Past Three Years of the ISMAR and IEEE VR Conference Papers," 2012 International Symposium on Ubiquitous Virtual Reality, Daejeon, Korea (South), 2012, pp. 1-5, doi: 10.1109/ISUVR.2012.17.
10. Li, Xiao & Xu, Bo & Teng, Yue & Ren, Yi-tian & Hu, Zhu-min. (2014). Comparative research of AR and VR technology based on user experience. International Conference on Management Science and Engineering - Annual Conference Proceedings. 1820-1827. 10.1109/ICMSE.2014.6930456.
11. Schaffernak, Harald & Mösl, Birgit & Vorraber, Wolfgang & Koglbauer, Ioana.

(2020). Potential Augmented Reality Application Areas for Pilot Education: An Exploratory Study. Education Sciences. 10. 86. 10.3390/educsci10040086.

12. A. Correia and V. Conceição, "Survey on Augmented Reality Technologies for Naval Training," 2019 14th Iberian Conference on Information Systems and Technologies (CISTI), Coimbra, Portugal, 2019, pp. 1-6, doi: 10.23919/CISTI.2019.8760962.
13. Bimber, & Oliver, & Raskar, & Ramesh,. (2005). Spatial Augmented Reality Merging Real and Virtual Worlds. 10.1201/b10624.
14. Klein, Georg, and David Murray. "Parallel tracking and mapping for small AR workspaces." In 2007 6th IEEE and ACM international symposium on mixed and augmented reality, pp. 225-234. IEEE, 2007.
15. H. Kato, M. Billinghurst, I. Poupyrev, K. Imamoto and K. Tachibana, "Virtual object manipulation on a table-top AR environment," Proceedings IEEE and ACM International Symposium on Augmented Reality (ISAR 2000), Munich, Germany, 2000, pp. 111-119, doi: 10.1109/ISAR.2000.880934.

16. Sahu, C.K., Young, C. and Rai, R., 2021. Artificial intelligence (AI) in augmented reality (AR)-assisted manufacturing applications: a review. International Journal of Production Research, 59(16), pp.4903-4959.
17. XIA Bang-gui. A Mobile learning system based on Android [J]. Journal of Xihua University (Natural Science), 2011, 30 (5): 81-84.
18. Mehta, Soham & Jain, Pratish & Vora, Aayushi & Joshi, Abhijit & Dalvi, Harshal. (2017). Augmented Reality Books: An Immersive Approach to Learning.
19. ALHARBI, Basma & Aljojo, Nahla & Alshutayri, Areej & Banjar, Ameen & ZAINOL, Azida & ALHARBI, Asmaa & ALGHANMI, Sanaa & MANSOUR, Shaza & ALSHEHRI, Mram. (2021). The design and implementation of an interactive mobile Augmented Reality application for an improved furniture shopping experience. Revista Română de Informatică și Automatică. 31. 69- 80. 10.33436/v31i3y202106.
20. Reuksupasompon, Peeranut & Aruncharathorn, Maytichai & Vittayakorn, Sirion. (2018). AR Development For Room Design. 1-6. 10.1109/JCSSE.2018.8457343.
21. R. Aggarwal and A. Singhal, "Augmented Reality and its effect on our life,"

2019 9th International Conference on Cloud Computing, Data Science & Engineering (Confluence), Noida, India, 2019, pp. 510-515, doi: 10.1109/CONFLUENCE.2019.8776989.

22. V. Interrante, T. Höllerer and A. Lécuyer, "Virtual and Augmented Reality," in IEEE Computer Graphics and Applications, vol. 38, no. 2, pp. 28-30, Mar./Apr. 2018, doi: 10.1109/MCG.2018.021951630.
23. L. Abazi-Bexheti, A. Kadriu and M. Apostolova, "Research on VR/AR integration in education," 2022 45th Jubilee International Convention on Information, Communication and Electronic Technology (MIPRO), Opatija, Croatia, 2022, pp. 563-567, doi: 10.23919/MIPRO55190.2022.9803398.

Appendix:

A Source Code

```
buildscript {
    ext {
        compiler_version = '1.2.0'
        compose_version = '1.2.1'
        nav_version = "2.5.1"
        live_data_version = "2.5.1"
        retrofitVersion  =  "2.9.0"
        okHttpVersion = "4.9.1"
        dagger_version  = "2.43.2"
        coil_version = "2.1.0"
        constraint_layout = "1.0.1"
        accompanist_version = "0.26.1-alpha"
    }
    dependencies {
        classpath 'com.android.tools.build:gradle:7.1.0-alpha05'
        classpath "org.jetbrains.kotlin:kotlin-gradle-plugin:1.5.10"
        classpath 'com.google.dagger:hilt-android-gradle-plugin:2.43.2'
    }
}
plugins {
    id 'com.android.application' version '7.2.1' apply false
    id 'com.android.library' version '7.2.1' apply false
    id 'org.jetbrains.kotlin.android' version '1.7.0' apply false
}



task clean(type: Delete) {
    delete rootProject.buildDir
}
package `in`.kay.furture

import androidx.test.platform.app.InstrumentationRegistry
```

```
import androidx.test.ext.junit.runners.AndroidJUnit4

import org.junit.Test
import org.junit.runner.RunWith

import org.junit.Assert.*

/**
 * Instrumented test, which will execute on an Android device.
 *
 * See [testing documentation](http://d.android.com/tools/testing).
 */
@RunWith(AndroidJUnit4::class)
class ExampleInstrumentedTest {
    @Test
    fun useAppContext() {
        // Context of the app under test.
        val appContext = InstrumentationRegistry.getInstrumentation().targetContext
        assertEquals("in.kay.furture", appContext.packageName)
    }
}
<?xml version="1.0" encoding="utf-8"?>
<adaptive-icon xmlns:android="http://schemas.android.com/apk/res/android">
    <background android:drawable="@drawable/ic_launcher_background"/>
    <foreground android:drawable="@mipmap/ic_launcher_foreground"/>
</adaptive-icon>

package `in`.kay.furture.models

import android.os.Parcelable
import kotlinx.parcelize.Parcelize

@Parcelize
data class FurnitureModel(
```

```
    var name: String? = null,
    val drawable: Int = 0,
    val link: String? = null,
    val price : Int? = 0,
    val description : String ? = "",
    val type:String? = ""
) : Parcelable
```

```
package `in`.kay.furture.ui.theme

import androidx.compose.ui.graphics.Color

val colorWhite = Color(0xFFFfffff)
val colorPurple = Color(0xFF2CA0CF)
val colorBlack = Color(0xFF000000)
val cardBack = Color(0xFFF3F6F8)
val colorPrimary = Color(0xFF5ECEC4)
val colorSecondaryText = Color(0x80000000)
val colorBorder = Color(0xFFEEEEEE)
```

```
package `in`.kay.furture.ui.theme

import androidx.compose.foundation.shape.RoundedCornerShape
import androidx.compose.material.Shapes
import androidx.compose.ui.unit.dp

val Shapes = Shapes(
    small = RoundedCornerShape(4.dp),
    medium = RoundedCornerShape(4.dp),
    large = RoundedCornerShape(0.dp)
)
```

```
package `in`.kay.furture.ui.theme

import androidx.compose.foundation.isSystemInDarkTheme
import androidx.compose.material.MaterialTheme
```

```
import androidx.compose.material.MaterialTheme.colors
import androidx.compose.material.darkColors
import androidx.compose.material.lightColors
import androidx.compose.runtime.Composable

private val LightColorPalette = lightColors(
    primary = colorWhite,
    primaryVariant = colorWhite,
    secondary = colorPurple
)


@Composable
fun FurtureTheme(content: @Composable () -> Unit) {
        LightColorPalette

    MaterialTheme(
        colors = colors,
        typography = Typography,
        shapes = Shapes,
        content = content
    )
}
package `in`.kay.furture.ui.theme

import `in`.kay.furture.R
import androidx.compose.material.Typography
import   androidx.compose.ui.text.TextStyle
import   androidx.compose.ui.text.font.Font
import androidx.compose.ui.text.font.FontFamily
import androidx.compose.ui.text.font.FontWeight
import androidx.compose.ui.unit.sp
```

```
val Typography = Typography(
    h1 = TextStyle(
        fontFamily = gilroy(),
        fontWeight = FontWeight.Bold,
        fontSize = 40.sp,
        color = colorBlack
    ),
    body1= TextStyle(
        fontFamily = gilroy(),
        fontWeight = FontWeight.Normal,
        fontSize = 16.sp,
        color = colorBlack
    ),
    body2 = TextStyle(
        fontFamily = gilroy(),
        fontWeight = FontWeight.Normal,
        fontSize = 16.sp,
        color = colorSecondaryText
    ),
    defaultFontFamily = gilroy()
)


fun gilroy() = FontFamily(
    Font(R.font.font_gilroy_bold, FontWeight.Bold),
    Font(R.font.font_gilroy_medium, FontWeight.Medium),
    Font(R.font.font_gilroy_semi_bold, FontWeight.SemiBold),
)
package `in`.kay.furture.utils

import android.app.Application
import dagger.hilt.android.HiltAndroidApp

@HiltAndroidApp
```

```
class App : Application() {
    override fun onCreate() {
        super.onCreate()
        instance = this
    }

    companion object {
        lateinit var instance: App
    }
}
package `in`.kay.furture.utils

import `in`.kay.furture.R
import `in`.kay.furture.models.FurnitureModel
import kotlin.random.Random

fun getCategories() = listOf(
    FurnitureModel("chair", R.drawable.ic_chair),
    FurnitureModel("sofa", R.drawable.ic_sofa),
    FurnitureModel("home decor", R.drawable.ic_home_decor),
    FurnitureModel("office", R.drawable.ic_office),
    FurnitureModel("tables", R.drawable.ic_table),
)

fun getRecommended(): List<FurnitureModel> {
    val list = mutableListOf<FurnitureModel>()
    list.add(getChairs()[(0 until getChairs().size).random()])
    list.add(getOffices()[(0 until getOffices().size).random()])
    list.add(getSofas()[(0 until getSofas().size).random()])
    list.add(getHomeDecors()[(0 until getHomeDecors().size).random()])
    list.add(getTables()[(0 until getTables().size).random()])
    return list
}
```

```
fun getChairs() = listOf(
    FurnitureModel(
        "Modern Chair",
        R.drawable.chair_grey,
        price = randomPrice(100, 200),
        description = randomDescription("Modern Chair"),
        link = "https://raw.githubusercontent.com/Sachinbhola/App-Templates/master/Resources/chair/arm_chair_furniture/scene.gltf",
        type = "chair"
    ),
    FurnitureModel(
        "Manchester Chair",
        R.drawable.chair_old,
        price = randomPrice(100, 200),
        description = randomDescription("Manchester Chair"),
        link = "https://raw.githubusercontent.com/Sachinbhola/App-Templates/master/Resources/the_matrix_red_chesterfield_chair/scene.gltf",
        type = "chair"
    ),
    FurnitureModel(
        "Orange Sheen Chair",
        R.drawable.chair_orange,
        price = randomPrice(100, 200),
        description = randomDescription("Orange Sheen Chair"),
        link = "https://raw.githubusercontent.com/KhronosGroup/glTF-Sample-Models/master/2.0/SheenChair/glTF/SheenChair.gltf",
        type = "chair"
    ),
    FurnitureModel(
        "Classic Chair",
        R.drawable.chair_wood,
        price = randomPrice(100, 200),
        description = randomDescription("Classic Chair"),
        link = "https://raw.githubusercontent.com/Sachinbhola/App-
```

```
Templates/master/Resources/chair/furniture_for_real-
time_visualization_engine/scene.gltf",
        type = "chair"
    )
)

fun getSofas() = listOf(
    FurnitureModel(
        "Velvet Sofa",
        R.drawable.sofa1,
        price = randomPrice(400, 1000),
        description = randomDescription("Velvet Sofa"),
        type = "sofa",
        link = "https://raw.githubusercontent.com/KhronosGroup/glTF-Sample-
Models/master/2.0/GlamVelvetSofa/glTF/GlamVelvetSofa.gltf"
    ),
    FurnitureModel(
        "Leather Sofa",
        R.drawable.sofa2,
        price = randomPrice(400, 1000),
        description = randomDescription("Leather Sofa"),
        type = "sofa",
        link = "https://raw.githubusercontent.com/Sachinbhola/App-
Templates/master/Resources/sofas/sofa_game_ready__2k_pbr/scene.gltf"
    ),
    FurnitureModel(
        "Victorian Sofa",
        R.drawable.sofa3,
        price = randomPrice(400, 1000),
        description = randomDescription("Victorian Sofa"),
        type = "sofa",
        link = "https://raw.githubusercontent.com/Sachinbhola/App-
Templates/master/Resources/victorian_lounge_sofa/scene.gltf"
    ),
```

```
    FurnitureModel(
        "Chesterfield Sofa",
        R.drawable.sofa5,
        price = randomPrice(400, 1000),
        description = randomDescription("Chesterfield Sofa"),
        type = "sofa",
        link = "https://raw.githubusercontent.com/Sachinbhola/App-Templates/master/Resources/sofa_02_4k.gltf/sofa_02_4k.gltf"
    ),
    FurnitureModel(
        "Modern Sofa",
        R.drawable.sofa6,
        price = randomPrice(400, 1000),
        description = randomDescription("Modern Sofa"),
        type = "sofa",
        link = "https://raw.githubusercontent.com/Sachinbhola/App-Templates/master/Resources/sofa_6/scene.gltf"
    ),
    FurnitureModel(
        "Lawson Sofa",
        R.drawable.sofa8,
        price = randomPrice(400, 1000),
        description = randomDescription("Lawson Sofa"),
        type = "sofa",
        link = "https://raw.githubusercontent.com/Sachinbhola/App-Templates/master/Resources/sofas/sofa/scene.gltf"
    )
)

fun randomDescription(type: String): String {
    return "This $type is an elegant and functional piece of furniture. It is suitable for family visits and parties with friends and perfect for relaxing in front of the TV after hard work."
}
```

```
fun randomPrice(lowerPrice: Int, upperPrice: Int): Int {
    return (lowerPrice..upperPrice).random()
}

fun getHomeDecors() = listOf(
    FurnitureModel(
        "Modern Pot",
        R.drawable.dec1,
        price = randomPrice(20, 50),
        description = randomDescription("Modern Pot"),
        type = "home decoration",
        link = "https://raw.githubusercontent.com/Sachinbhola/App-
Templates/master/Resources/decor/lavender_vase/scene.gltf"
    ),
    FurnitureModel(
        "Lamp",
        R.drawable.dec2,
        price = randomPrice(20, 50),
        description = randomDescription("Lamp"),
        type = "home decoration",
        link = "https://raw.githubusercontent.com/Sachinbhola/App-
Templates/master/Resources/decor/lamp/scene.gltf"
    ),
    FurnitureModel(
        "Flowers",
        R.drawable.dec3,
        price = randomPrice(20, 50),
        description = randomDescription("Flowers"),
        type = "home decoration",
        link = "https://raw.githubusercontent.com/Sachinbhola/App-
Templates/master/Resources/decor/flower_vase%20(1)/scene.gltf"
    )
)
```

```
fun getOffices() = listOf(
    FurnitureModel(
        "Gaming Chair",
        R.drawable.office,
        price = randomPrice(50, 80),
        description = randomDescription("Gaming Chair"),
        type = "office",
        link = "https://raw.githubusercontent.com/Sachinbhola/App-Templates/master/Resources/office%20essentials%20and%20cabinets/gaming_chair_1-_black/scene.gltf"
    ),
    FurnitureModel(
        "Classic chair ",
        R.drawable.office_chair,
        price = randomPrice(50, 80),
        description = randomDescription("Classic chair "),
        type = "office",
        link = "https://raw.githubusercontent.com/Sachinbhola/App-Templates/master/Resources/office%20essentials%20and%20cabinets/office_chair%20(1)/scene.gltf"
    ),
    FurnitureModel(
        "Desk",
        R.drawable.office_desk,
        price = randomPrice(50, 80),
        description = randomDescription("Desk"),
        type = "office",
        link = "https://raw.githubusercontent.com/Sachinbhola/App-Templates/master/Resources/office%20essentials%20and%20cabinets/modern_table_set/scene.gltf"
    )
)
```

```
fun getTables() = listOf(
    FurnitureModel(
        "Table",
        R.drawable.table1,
        price = randomPrice(50, 80),
        description = randomDescription("Table"),
        type = "table",
        link = "https://raw.githubusercontent.com/Sachinbhola/App-Templates/master/Resources/table/mahogany_table/scene.gltf"
    ),
    FurnitureModel(
        "Wooden Table",
        R.drawable.table2,
        price = randomPrice(50, 80),
        description = randomDescription("Wooden Table"),
        type = "table",
        link = "https://raw.githubusercontent.com/Sachinbhola/App-Templates/master/Resources/office%20essentials%20and%20cabinets/simple_desk_free/scene.gltf"
    ),
    FurnitureModel(
        "Metal Table",
        R.drawable.table3,
        price = randomPrice(50, 80),
        description = randomDescription("Metal Table"),
        type = "table",
        link = "https://raw.githubusercontent.com/Sachinbhola/App-Templates/master/Resources/table/industrial_table/scene.gltf"
    ),
    FurnitureModel(
        "Classic Table",
        R.drawable.table4,
        price = randomPrice(50, 80),
        description = randomDescription("Classic Table"),
```

```
        type = "table",
        link = "https://raw.githubusercontent.com/Sachinbhola/App-
Templates/master/Resources/table/victorian_coffee_table/scene.gltf"
    )
)

fun getSize(idx: Int): Int {
    return when (idx) {
        0 -> getChairs().size
        1 -> getSofas().size
        2 -> getHomeDecors().size
        3 -> getOffices().size
        4 -> getTables().size
        else -> 0
    }
} package `in`.kay.furture

import `in`.kay.furture.screens.DetailScreen
import `in`.kay.furture.screens.HomeScreen
import `in`.kay.furture.screens.SplashScreen
import `in`.kay.furture.ui.theme.FurtureTheme
import android.os.Bundle
import androidx.activity.ComponentActivity
import androidx.activity.compose.setContent
import androidx.compose.foundation.layout.fillMaxSize
import androidx.compose.material.MaterialTheme
import androidx.compose.material.Surface
import androidx.compose.material.Text
import androidx.compose.runtime.Composable
import androidx.compose.ui.Modifier
import androidx.compose.ui.tooling.preview.Preview
import androidx.hilt.navigation.compose.hiltViewModel
import androidx.navigation.compose.NavHost
import androidx.navigation.compose.composable
```

```
import androidx.navigation.compose.rememberNavController
import dagger.hilt.android.AndroidEntryPoint

@AndroidEntryPoint
class MainActivity : ComponentActivity() {
    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
        setContent {
            FurtureTheme {
                // A surface container using the 'background' color from the theme
                Surface(
                    modifier = Modifier.fillMaxSize(),
                    color = MaterialTheme.colors.background
                ) {
                    val navController = rememberNavController()
                    val viewModel = hiltViewModel<SharedViewModel>()
                    NavHost(navController = navController, startDestination = "splash") {
                        composable("home") { HomeScreen(navController, viewModel) }
                        composable("detail") {
                            DetailScreen(viewModel)
                        }
                        composable("splash") {
                            SplashScreen(navController = navController)
                        }
                    }
                }
            }
        }
    }
}

@Composable
fun Greeting(name: String) {
    Text(text = "Hello $name!")
```

```
}

@Preview(showBackground = true)
@Composable
fun DefaultPreview() {
    FurtureTheme {
        Greeting("Android")
    }
}
package `in`.kay.furture

import `in`.kay.furture.models.FurnitureModel
import androidx.lifecycle.ViewModel
import dagger.hilt.android.lifecycle.HiltViewModel
import javax.inject.Inject

@HiltViewModel
class SharedViewModel @Inject constructor() : ViewModel() {

    var data = FurnitureModel()

}
<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    package="in.kay.furture">

    <application
        android:allowBackup="true"
        android:name=".utils.App"
        android:dataExtractionRules="@xml/data_extraction_rules"
        android:fullBackupContent="@xml/backup_rules"
        android:icon="@mipmap/ic_launcher"
        android:label="@string/app_name"
```

```
        android:roundIcon="@mipmap/ic_launcher_round"
        android:supportsRtl="true"
        android:theme="@style/Theme.Furture"
        tools:targetApi="31">
        <activity
            android:name=".MainActivity"
            android:exported="true"
            android:label="@string/app_name"
            android:theme="@style/Theme.Furture">
            <intent-filter>
                <action android:name="android.intent.action.MAIN" />

                <category android:name="android.intent.category.LAUNCHER" />
            </intent-filter>
        </activity>
    </application>

</manifest>
package `in`.kay.furture

import org.junit.Test

import org.junit.Assert.*

/**
 * Example local unit test, which will execute on the development machine (host).
 *
 * See [testing documentation](http://d.android.com/tools/testing).
 */
class ExampleUnitTest {
    @Test
    fun addition_isCorrect() {
        assertEquals(4, 2 + 2)
    }
```

```
}
plugins {
    id 'com.android.application'
    id 'org.jetbrains.kotlin.android'
    id 'kotlin-kapt'
    id 'com.google.dagger.hilt.android'
    id 'kotlin-parcelize'
}

android {
    compileSdk 33

    defaultConfig {
        applicationId "in.kay.furture"
        minSdk 21
        targetSdk 32
        versionCode 1
        versionName "1.0"

        testInstrumentationRunner "androidx.test.runner.AndroidJUnitRunner"
        vectorDrawables {
            useSupportLibrary true
        }
    }

    buildTypes {
        release {
            minifyEnabled false
            proguardFiles            getDefaultProguardFile('proguard-android-optimize.txt'),
'proguard-rules.pro'
        }
    }
    compileOptions {
        sourceCompatibility JavaVersion.VERSION_1_8
```

```
        targetCompatibility JavaVersion.VERSION_1_8
    }
    kotlinOptions {
        jvmTarget = '1.8'
    }
    buildFeatures {
        compose true
    }
    composeOptions {
        kotlinCompilerExtensionVersion compiler_version
    }
    packagingOptions {
        resources {
            excludes += '/META-INF/{AL2.0,LGPL2.1}'
        }
    }
}

dependencies {

    implementation  'androidx.core:core-ktx:1.8.0'
    implementation "androidx.compose.ui:ui:$compose_version"
    implementation "androidx.compose.material:material:$compose_version"
    implementation                    "androidx.constraintlayout:constraintlayout-
compose:$constraint_layout"
    implementation "androidx.compose.ui:ui-tooling-preview:$compose_version"
    implementation  'androidx.lifecycle:lifecycle-runtime-ktx:2.5.1'
    implementation   'androidx.activity:activity-compose:1.5.1'
    testImplementation 'junit:junit:4.13.2'
    androidTestImplementation  'androidx.test.ext:junit:1.1.3'
    androidTestImplementation 'androidx.test.espresso:espresso-core:3.4.0'
    androidTestImplementation                    "androidx.compose.ui:ui-test-
junit4:$compose_version"
    debugImplementation "androidx.compose.ui:ui-tooling:$compose_version"
```

```
    debugImplementation "androidx.compose.ui:ui-test-manifest:$compose_version"
    // Coil
    implementation("io.coil-kt:coil-compose:$coil_version")
    implementation "androidx.lifecycle:lifecycle-livedata-ktx:$live_data_version"
    // dagger -hilt
    implementation "androidx.hilt:hilt-navigation-compose:1.0.0"
    implementation "com.google.dagger:hilt-android:$dagger_version"
    kapt "com.google.dagger:hilt-compiler:$dagger_version"
    //Retrofit
    implementation "com.squareup.retrofit2:retrofit:$retrofitVersion"
    implementation "com.squareup.retrofit2:converter-gson:$retrofitVersion"
    implementation "com.squareup.okhttp3:logging-interceptor:$okHttpVersion"
    implementation "com.squareup.retrofit2:converter-scalars:$retrofitVersion"
    implementation "com.google.accompanist:accompanist-navigation-
animation:$accompanist_version"
    implementation "com.google.accompanist:accompanist-
webview:$accompanist_version"
    implementation("com.android.support:palette-v7:28.0.0")
    // dagger -hilt
    implementation "androidx.hilt:hilt-navigation-compose:1.0.0"
    implementation "com.google.dagger:hilt-android:$dagger_version"
    kapt "com.google.dagger:hilt-compiler:$dagger_version"
    def nav_version = "2.5.1"

    implementation "androidx.navigation:navigation-compose:$nav_version"
    implementation 'io.github.kotlin-telegram-bot.kotlin-telegram-bot:telegram:6.0.7'
}
pluginManagement {
    repositories {
        gradlePluginPortal()
        google()
        mavenCentral()
    }
}
```

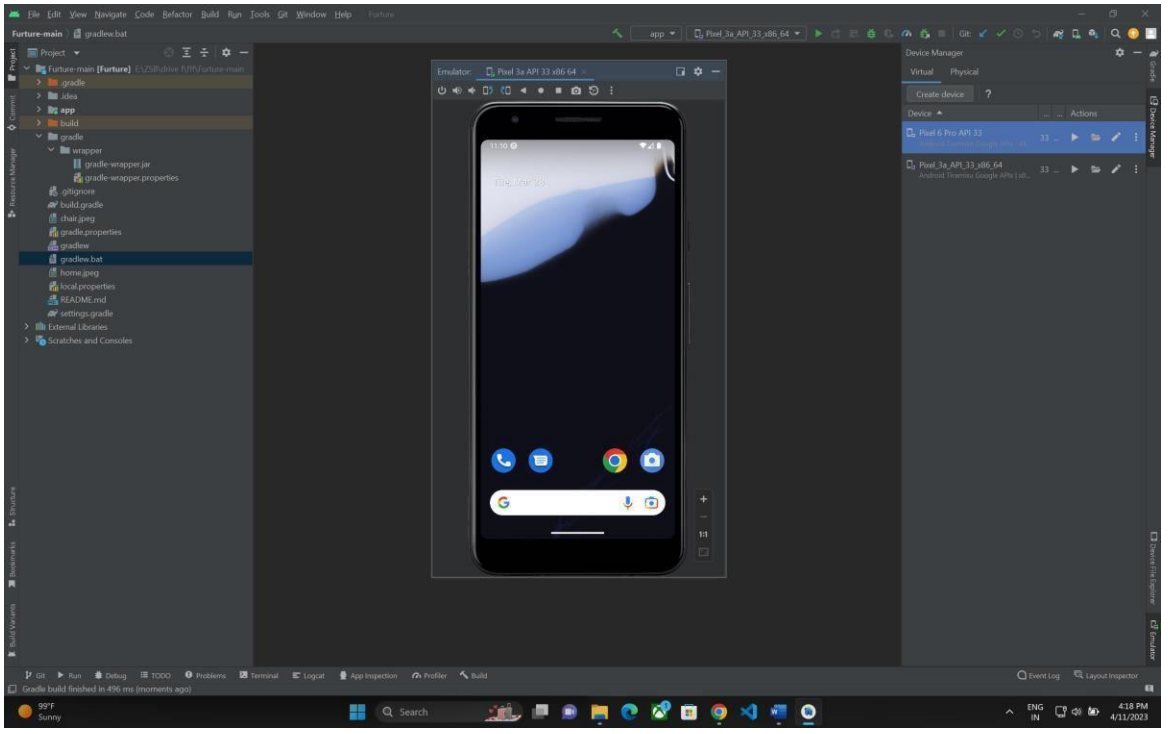
```
dependencyResolutionManagement {
    repositoriesMode.set(RepositoriesMode.FAIL_ON_PROJECT_REPOS)
    repositories {
        google()
        mavenCentral()
        maven { url "https://jitpack.io" }
    }
}
rootProject.name = "Furture"
include ':app'
[
  {
    "repositoryUrl": "https://packages.jetbrains.team/maven/p/ij/intellij-redist",
    "artifacts": [
      {
        "groupId": "junit",
        "artifactId": "junit",
        "annotations": {
        "[4.0, 5.0)": {
            "groupId": "org.jetbrains.externalAnnotations.junit",
            "artifactId": "junit",
            "version": "4.12-an1"
          }
        }
      },
      {
        "groupId": "org.hamcrest",
        "artifactId": "hamcrest-core",
        "annotations": {
          "[1.0, 1.3)": {
            "groupId": "org.jetbrains.externalAnnotations.org.hamcrest",
            "artifactId": "hamcrest-core",
            "version": "1.3-an1"
          }
```

```
  }
},
{
  "groupId": "com.fasterxml.jackson.core",
  "artifactId":  "jackson-core",
  "annotations": {
    "[2.0, 3.0)": {
      "groupId": "org.jetbrains.externalAnnotations.com.fasterxml.jackson.core",
      "artifactId": "jackson-core",
      "version": "2.9.6-an1"
    }
  }
},
{
  "groupId": "com.fasterxml.jackson.core",
  "artifactId": "jackson-databind",
  "annotations": {
    "[2.0, 3.0)": {
      "groupId": "org.jetbrains.externalAnnotations.com.fasterxml.jackson.core",
      "artifactId": "jackson-databind",
      "version": "2.9.6-an1"
    }
  }
},
{
  "groupId": "com.miglayout",
  "artifactId": "miglayout-swing",
  "annotations": {
    "[5.0, 5.1]": {
      "groupId": "org.jetbrains.externalAnnotations.com.miglayout",
      "artifactId": "miglayout-swing",
      "version": "5.1-an1"
    }
    }
```

```
},
{
  "groupId": "io.netty",
  "artifactId": "netty-buffer",
  "annotations": {
    "[4.0, 4.2)": {
      "groupId": "org.jetbrains.externalAnnotations.io.netty",
      "artifactId": "netty-buffer",
      "version": "4.1.27.Final-an1"
    }
  }
},
{
  "groupId": "io.netty",
  "artifactId": "netty-common",
  "annotations": {
    "[4.0, 4.2)": {
      "groupId": "org.jetbrains.externalAnnotations.io.netty",
      "artifactId": "netty-common",
      "version": "4.1.27.Final-an1"
    }
  }
},
{
  "groupId": "io.netty",
  "artifactId": "netty-resolver",
  "annotations": {
    "[4.0, 4.2)": {
      "groupId": "org.jetbrains.externalAnnotations.io.netty",
      "artifactId": "netty-resolver",
      "version": "4.1.27.Final-an1"
    }
  }
},
```

```
  {
    "groupId": "io.netty",
    "artifactId": "netty-transport",
    "annotations": {
      "[4.0, 4.2)": {
        "groupId": "org.jetbrains.externalAnnotations.io.netty",
        "artifactId": "netty-transport",
        "version": "4.1.27.Final-an1"
      }
    }
  },
  {
    "groupId": "org.picocontainer",
    "artifactId": "picocontainer",
    "annotations": {
      "[1.0, 1.2]": {
        "groupId": "org.jetbrains.externalAnnotations.org.picocontainer",
        "artifactId": "picocontainer",
        "version": "1.2-an1"
      }
    }
  },
  {
    "groupId": "org.yaml",
    "artifactId": "snakeyaml",
    "annotations": {
      "[1.20, 1.3)": {
        "groupId": "org.jetbrains.externalAnnotations.org.yaml",
        "artifactId": "snakeyaml",
        "version": "1.23-an1"
      }
    }
  }
]
```

B. SCREENSHOTS



table

wooden table

from

$ 74

Wooden Table

this wooden table is an elegant and functional piece of furniture. it is suitable for family visits and parties with friends and perfect for relaxing in front of the tv after hard work.

view into your house.

sofa

victorian sofa

from

$ 404

Victorian Sofa

this victorian sofa is an elegant and functional piece of furniture. it is suitable for family visits and parties with friends and perfect for relaxing in front of the tv after hard work.

view into your house.

table

wooden table

from

$ 74

Wooden Table

this wooden table is an elegant and functional piece of furniture. it is suitable for family visits and parties with friends and perfect for relaxing in front of the tv after hard work.

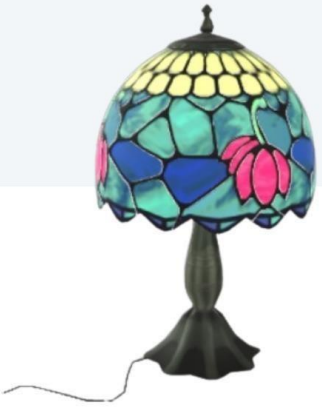
view into your house.

home decoration

lamp

from

$ 23



Lamp

this lamp is an elegant and functional piece of furniture. it is suitable for family visits and parties with friends and perfect for relaxing in front of the tv after hard work.

view into your house.

office

gaming chair

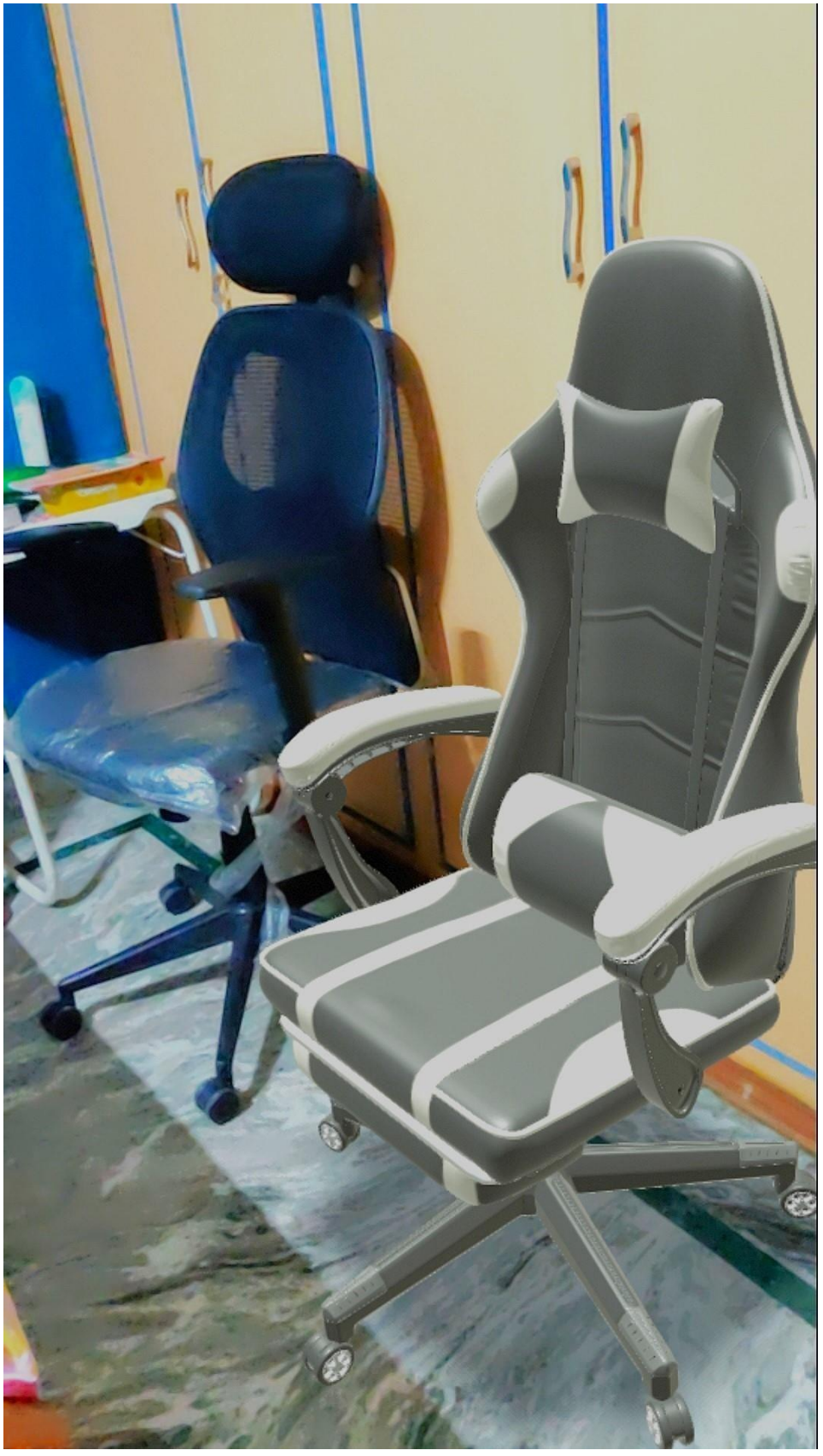
from

$ 51

Gaming Chair

this gaming chair is an elegant and functional piece of furniture. it is suitable for family visits and parties with friends and perfect for relaxing in front of the tv after hard work.

view into your house.



C. RESEARCH PAPER

Augmented Reality Furniture Application

Sai Bhavan Gubbala
Student
Department of CSE
Sathyabama Institute of Science and Technology
saibhavan9@gmail.com

Dhruv Naidu Alti
Student
Department of CSE
Sathyabama Institute of Science and Technology
dhruvnaidualti@gmail.com

S. R Srividhya
Assistant Professor
Department of CSE
Sathyabama Institute of Science and Technology
srividhyasr.cse@sathyabama.ac.in

S. Pothumani
Assistant Professor
Department of CSE
Sathyabama Institute of Science and Technology
pothumani.cse@sathyabama.ac.in

***Abstract*—In the modern digital transformation era, providing customers with confidence in the suitability of the furniture they choose for various situations in which it will be used is the major problem in online furniture sales. While buying furniture, purchasers must ensure the furniture's model, size, and color are appropriate for and complement the space in which they will be put. The flexibility and appropriateness of the furniture product when placed in a particular area may be seen by prospective buyers with suitable instruments. For furniture to be sold online correctly, these issues must be resolved. By using Augmented Reality, this is feasible, to do. When it comes to online product customization, certain application models in the furniture sector employ Augmented Reality and mobile technology. A 3D product visualization tool, product pricing, and an enhanced renderer are all included. For the purpose of creating AR-based furniture technology, Google AR is used in the development process. It provides a framework for people or teams to creatively and flexibly address challenging tasks and issues in order to generate and deliver the best performance. The study is finished using Android Studio and Google AR services. Based on the results of the studies, it is advised that this application be enhanced to have a greater degree of performance. This might be achieved by creating a variety of various creator patterns using barcodes or QR codes, classifying surfaces like ceilings and walls so that the program can accurately show 3D things on them, and improving 3D models to fit them in the projecting space.**

Keywords*— *Augmented Reality, Google Augmented Reality Core, Quick Response Code, Three Dimensional Objects, Furniture.

I. INTRODUCTION

This India's e-commerce sector continues to grow significantly. But, it's only possible to have change with growth, thus, for the e-commerce sector to prosper, new technology developments must be periodically adopted. One such innovation might be the use of augmented Reality to draw in both current and potential clients. Hence, for the Furnished software, our team selected the Furniture business as the e-commerce sector. While there are now just a few well-known players in the furniture market, there remains a a plenty of potential to expand for anybody who can provide something different with the use of augmented Reality. This application aims to use augmented Reality to give all of the things shown in their online shop to their customers in the most efficient use of time and resources. With augmented Reality, one may put objects in your surroundings, but with virtual Reality, one is transported into the world that the developer has built. Virtual Reality has been acknowledged as an excellent teaching approach for industrial people, especially those functioning in harmful circumstances, despite AR being used in many fields, such as design and modelling. To show product data, and 3D models, perform feature tracking and determine locations superimposed on top of a picture of the actual world, the software first analyses photos acquired by the device's rear camera. It then applies either marker tracking or marker-less tracking to those photos. Several researchers' descriptions of augmented reality call for the use of head-mounted displays (HMDs). This study used Augmented Reality as platform with the three factors to avoid limiting it to a particular technology. To avoid confining AR to a specific technology. This study defines Augmented Reality as systems with three qualities. First is the Combination of real with the virtual, second is the Real-Time interaction and finally, third is the Three-Dimensional Registration. Given the development status in both fields, it is feasible to forecast and extrapolate the calculations performed by machine learning algorithms on supercomputers. We can project the forecast using augmented reality to comprehend the influence on its surroundings. Without protective gear, anyone can construct a world of their own and engage in chemical experimentation to fully grasp the possibilities of metals and non-metals. A restriction of augmented reality is that it can only project information already there.

II. LITERATURE SURVEY

Pawe Nowacki and Marek Sawomir Woda examined and compared the capabilities of ARCore and ARkit in their article "ARCore and ARkit". The authors created matching criteria for both platforms, developed test apps, and executed comparative tests [1]. Akshay Singh and Sakshi Sharma's article outlines the building of an Android mobile platform application utilizing Android Studio and the PHP Framework. Mobile Development has considerable experience working on projects such as video and music players, gaming software, photo viewer and editing software, and many more [2].The review topic of Augmented Reality, in which 3-D virtual elements are incorporated into a 3-D actual world in real time. According to the above literature review, the app should focus more on usefulness and ease of use to increase app adherence and effectiveness. This transition might be caused by the fact that most industries are now servicing a different generation of customers or by recent technological breakthroughs [3]. The Furnished application operates effectively in today's society, where client convenience is paramount. As a result, one may accomplish our aim of saving time, money, and hassles by adopting AR. Additionally, several plugins and Software Development Kits are used to optimize the user's purchasing experience [4]. We were conscious of augmented reality's abilities in terms of performance. According to research results, it is possible to create an infinite number of virtual screens by using virtual reality headset devices. Augmented Reality has the potential to impact the use of computer monitors in the information technology sector and

XXX-X-XXXX-XXXX-X/XX/$XX.00 ©20XX IEEE

other industries [5]. Our team found that both augmented and virtual reality use the sensors already incorporated into the device to map out their surroundings. Compared to gyroscopes, the devices equipped with a LiDAR sensor have a more accurate projection of three-dimensional objects. Laser and its reflection are used by the LiDAR sensor in order to get spatial information [6]. According to the findings of this research, there are more advantages connected with the use of AR and VR in the sector of education. Youths may use Virtual Reality headsets in conjunction with Google Maps to go to any location on the planet. Pupils are able to gain knowledge about the history, geography, and culture of a variety of locations without leaving the classroom [7]. Youngsters who spend a lot of time in the virtual environment have a better ability to grasp the realm of imagination. The fundamental areas of science, physics and chemistry, need a more in-depth knowledge in addition to the capacity for creative thought. Children are able to comprehend the mechanics behind and applications of chemistry in day-to-day life with the use of virtual reality headsets [8]. There is a diverse array of uses for augmented and virtual reality. Augmented Reality has the potential to be used in the training of pilots. This would enable the pilots to view and interact with their surroundings in a three-dimensional world. Augmented Reality makes it possible to display real-time information immediately in the line of sight of a pilot. This information may include navigation data, weather conditions, and traffic information, among other things. Also, it can direct pilots even when the weather is hazy and foggy [9]. In the event that a machine has to be repaired or maintained, AR can supply maintenance and repair instructions for any type of vehicle equipment. This makes it possible for those who have yet to gain experience to carry out maintenance activities in an efficient and accurate manner [10]. During business meetings, augmented reality may be utilized to provide virtual surroundings that replicate the atmosphere of a real workplace. Putting 3D models and presentations on display in a manner that is more interactive and interesting to the audience. During the meeting, the utilization of real-time data visualization tools, such as tables, graphs, and charts, may be of assistance in enhancing both comprehension and decision-making [11]. The team members can communicate with one another and cooperate using virtual whiteboards. They can also obtain reports and interact with each other in real-time. Architects use augmented reality to help them see their ideas in various real-world environments. In this technology, three-dimensional representations of structures may be superimposed into real-world environments. It also has the capability of being used in the creation of interactive presentations for the customer. Augmented Reality Headsets may be useful to project managers throughout a project's construction and maintenance phases. Several team members can collaborate simultaneously to make choices in real-time [12].

III. System Requiremetns

The Android or iOS smartphone must meet the application's minimal requirements to operate the furniture application. Local storage must be authorized for the application to save images or videos. The application also runs on devices that do not meet the system requirements, but to avoid application crashes and overheating, it is recommended to meet the hardware and software requirements. In order to execute the project, the requirements mentioned above are necessary as flutter application which contains dart language has syntax that changes as the version updates.

TABLE I. Furniture Application Requirements

S. No.	Software/Hardware	Version/Disk Space
1.	Random Access Memory	8GB
2.	Read-Only Memory	30GB
3.	Operating System	Windows 11/MAC OS Catalina or higher
4.	Android Studio Version	4.1 or higher
5.	Android Emulator	7.0 or higher
6.	Flutter SDK	2.2.3 or higher
7.	Kotlin	1.7.2.0
8.	iOS	13.3 OR higher

IV. System Architecture

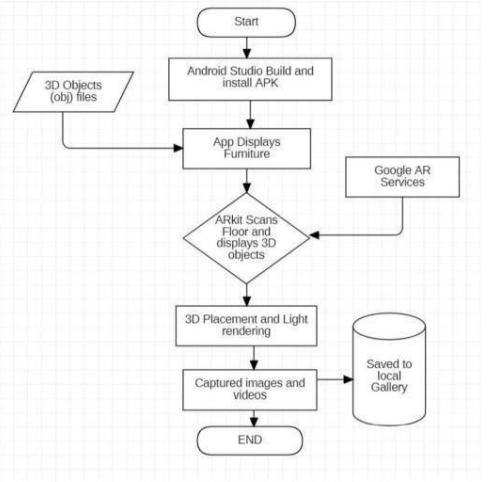


Fig. 1. Architecture Mapping

This architectural mapping explains the application's overall design to us. After the app has started, one may go through the catalog of available furniture and choose the specific item they want. In the AR mode, it is possible to get a preview of it and utilize the controls to adjust the furniture's positioning, size, and orientation. When one has finalized the product, they may take a video or picture.

V. Methodology

Stage-1: Collection of data and 3D models: The program needs a three-dimensional model in the GLFT file format in order to show each item in the catalog. The required models were bought at cgtrader.com, a database of models in various formats. While an actual shop is not yet available for experimental purposes of the application, the models were nevertheless required. Models of chairs, dining sets, coffee tables, and other furniture kinds were acquired to try a range of different types of furniture.

Stage-2: Many pages make up the user interface of the application. These pages include a homepage, a

classifications page, a goods page, and a product information page. Throughout development of the Interface, making it as user-friendly as possible was of the utmost importance, and contemporary design concepts were adhered to carefully.

Stage-3: After collecting product data, an interface was developed, which made the process of displaying the 3D Model much more straightforward. The result that was achieved in the software by employing GLB models is of a very high quality, and this applies to both its colors and its textures. Given the accuracy with which the light was estimated, the AR fragment has reflections and shadows of the chair that are spot on. Since Physically Based Rendering was used, the chair's leather can provide a shiny appearance. As the model was constructed while it was running, its level of complexity may be considered to be around typical. It will be possible to create models with higher degrees of complexity if the equipment used to create the models have better processing power and a database with a higher bandwidth.

Stage-4: The one feature that is not accessible and is also necessary for our project is Augmented Reality Core, which employs the built-in camera, and it includes all the Google smart devices that can be utilized to run programs. Since it has access to camera and can show furniture in augmented reality, a physical gadget may be utilized to fix this problem.

Stage-5: The Augmented Reality Core services, on devices that are compatible, render 3D images quickly and efficiently while using minimal system resources. Users may also utilize Augmented Reality Core to save the captured images or videos in a locally stored gallery that can be accessed at any time.

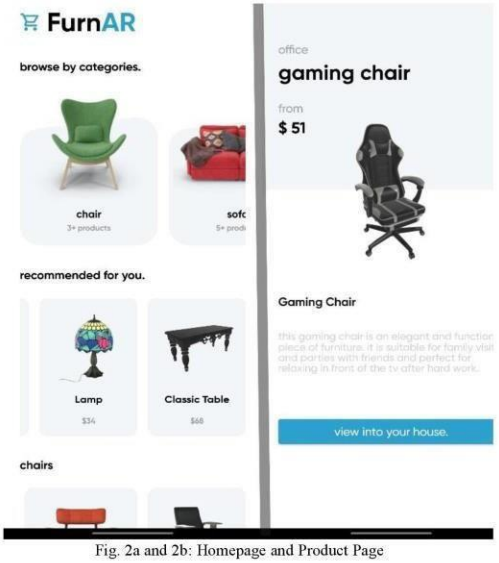


Fig. 2a and 2b: Homepage and Product Page


Fig. 3. Object Placement in Real-Time


Fig. 4. Comparison of Augmented Chair and Real Chair

VI. Constraints And Problems Faced

During Our Team has developed the prototype of object placement using Unity 3D. To begin with, it has more cons than pros compared to Flutter Student Development Kit. The Furniture application developed in Unity cannot scan and project objects in real-time. Flutter uses the dart language that supports UI with built-in widgets. During the development phase, the software succeeded in placing scanned furniture in the actual world, but various problems developed due to older Java development kit versions, greater CPU consumption, and memory cache restrictions. The source code was enhanced using Kotlin and cache management, which resolved compatibility problems, reduced device temperature, and allowed LiDAR-enabled devices to better object placement.

VII. Results and Conclusion

The outcomes of the program, created by employing OBJ models, are of a high grade, as are the colors and textures employed. Because of the detailed lighting modelling, the augmented reality fragment has a realistic chair that even includes shadows and reflections. PBR was used to give the leather on the chair a shining appearance. Given that it was built via dynamic processes, the completed model has a complexity density that is approximately average when compared to others of its kind. Devices that have more computational power and a database with a higher throughput might be used in the process of developing a more complicated model. Although the retail sector's historic success is undeniable, the modern consumer demands a more streamlined experience. Everything from going to the store to selecting a particular item. This shift could be brought on by the fact that most sectors are now serving a different generation of consumers or by recent technology advancements. Therefore, the Furnished application is ideal in this environment where client comfort comes first. By utilizing AR, people may accomplish the objective of saving time and money. Additionally, the different plugins and Software Development Kits are used to speed up the user's purchasing experience.

VIII. Future work

Our project dataset and scope will be expandable in the future. With this program, the user may be able to entirely remodel his house or even plan interior design. The user may also try out all of the house stuff, such as kitchen items, appliances, and so on. He may also be able to digitally test out several wall colors before having his walls painted. The application will serve as a one-stop shop for all house-related services. This application can be expanded further by developing a web application where frontend and backend comes into play. Because of the increased speed and reliability offered by 5G networks, augmented reality (AR) technology will be able to go further. With VR headsets, large corporations may establish a virtual work environment, allowing employees to continue working from home. Precise instructions may save time and money on basic car repairs and maintenance using Augmented Reality Headsets.

References

[1] Nowacki, Pawel & Woda, Marek. (2020). Capabilities of ARCore and ARKit Platforms for AR/VR Applications. 10.1007/978-3-030-19501-4_36.

[2] CLOUD -COMPUTING: An Analysis of Threats and Securities . Sakshi Sharma , Akshay Singh.2015 .2,s.l.: RJSET,2015, RESEARCH JOURNAL OF SCIENCE ENGINEERING AND TECHNOLOGY ,Vol.5 www.RJSET.com

[3] G. Eason, B. Noble, and I. N. Sneddon, "On certain integrals of Lipschitz-Hankel type involving products of Bessel functions," Phil. Trans. Roy. Soc. London, vol. A247, pp. 529–551, April 1955.

[4] Al Omran, Yasser & Abdall-Razak, Ali & Sohrabi, Catrin & Borg, Tiffanie-Marie & Nadama, Hayat & Ghassemi, Nader & Oo, Khine & Ghanem, Ali. (2019). Use of Augmented Reality in Reconstructive Microsurgery: A Systematic Review and Development of the Augmented Reality Microsurgery Score. Journal of Reconstructive Microsurgery. 36. 10.1055/s-0039-3401832.

[5] Kola, Vasista. (2022). Augmented Reality Vs. Virtual Reality. 03. 1-4.

[6] Alzahrani, Nouf. (2020). Augmented Reality: A Systematic Review of Its Benefits and Challenges in E-Learning Contexts. Applied Sciences. 10. 5660. 10.3390/app10165660.

[7] Viyanon, Waraporn et al. "AR Furniture: Integrating Augmented Reality Technology to Enhance Interior Design using Marker and Markerless tracking." IIP'17 (2017).

[8] Fox, Dylan & Li, Alyssa & Pandey, Anu & Kar, Rohan & Singh, Rajandeep. (2019). Augmented Reality for Visually Impaired People (AR for VIPs). 10.13140/RG.2.2.30196.78723.

[9] Li, Xiao & Xu, Bo & Teng, Yue & Ren, Yi-tian & Hu, Zhu-min. (2014). Comparative research of AR and VR technology based on user experience. International Conference on Management Science and Engineering - Annual Conference Proceedings. 1820-1827. 10.1109/ICMSE.2014.6930456.

[10] S. J. J. Kim, "A User Study Trends in Augmented Reality and Virtual Reality Research: A Qualitative Study with the Past Three Years of the ISMAR and IEEE VR Conference Papers," 2012 International Symposium on Ubiquitous Virtual Reality, Daejeon, Korea (South), 2012, pp. 1-5, doi: 10.1109/ISUVR.2012.17.

[11] Schaffernak, Harald & Mösl, Birgit & Vorraber, Wolfgang & Koglbauer, Ioana. (2020). Potential Augmented Reality Application Areas for Pilot Education: An Exploratory Study. Education Sciences. 10. 86. 10.3390/educsci10040086.

[12] A. Correia and V. Conceição, "Survey on Augmented Reality Technologies for Naval Training," 2019 14th Iberian Conference on Information Systems and Technologies (CISTI), Coimbra, Portugal, 2019, pp. 1-6, doi: 10.23919/CISTI.2019.8760962.

[13] Bimber, & Oliver, & Raskar, & Ramesh,. (2005). Spatial Augmented Reality Merging Real and Virtual Worlds. 10.1201/b10624.

[14] Klein, Georg, and David Murray. "Parallel tracking and mapping for small AR workspaces." In 2007 6th IEEE and ACM international symposium on mixed and augmented reality, pp. 225-234. IEEE, 2007.

[15] H. Kato, M. Billinghurst, I. Poupyrev, K. Imamoto and K. Tachibana, "Virtual object manipulation on a table-top AR environment," Proceedings IEEE and ACM International Symposium on Augmented Reality (ISAR 2000), Munich, Germany, 2000, pp. 111-119, doi: 10.1109/ISAR.2000.880934.

[16] Sahu, C.K., Young, C. and Rai, R., 2021. Artificial intelligence (AI) in augmented reality (AR)-assisted manufacturing applications: a review. International Journal of Production Research, 59(16), pp.4903-4959.

[17] XIA Bang-gui. A Mobile learning system based on Android [J]. Journal of Xihua University (Natural Science), 2011, 30 (5): 81-84.

[18] Mehta, Soham & Jain, Pratish & Vora, Aayushi & Joshi, Abhijit & Dalvi, Harshal. (2017). Augmented Reality Books: An Immersive Approach to Learning.

[19] ALHARBI, Basma & Aljojo, Nahla & Alshutayri, Areej & Banjar, Ameen & ZAINOL, Azida & ALHARBI, Asmaa & ALGHANMI, Sanaa & MANSOUR, Shaza & ALSHEHRI, Mram. (2021). The design and implementation of an interactive mobile Augmented Reality application for an improved furniture shopping experience. Revista Română de Informatică şi Automatică. 31. 69-80. 10.33436/v31i3y202106.

[20] Reuksupasompon, Peeranut & Aruncharathorn, Maytichai & Vittayakorn, Sirion. (2018). AR Development For Room Design. 1-6. 10.1109/JCSSE.2018.8457343.

[21] R. Aggarwal and A. Singhal, "Augmented Reality and its effect on our life," 2019 9th International Conference on Cloud Computing, Data Science & Engineering (Confluence), Noida, India, 2019, pp. 510-515, doi: 10.1109/CONFLUENCE.2019.8776989.

[22] V. Interrante, T. Höllerer and A. Lécuyer, "Virtual and Augmented Reality," in IEEE Computer Graphics and Applications, vol. 38, no. 2, pp. 28-30, Mar./Apr. 2018, doi: 10.1109/MCG.2018.021951630.

[23] L. Abazi-Bexheti, A. Kadriu and M. Apostolova, "Research on VR/AR integration in education," 2022 45th Jubilee International Convention on Information, Communication and Electronic Technology (MIPRO), Opatija, Croatia, 2022, pp. 563-567, doi: 10.23919/MIPRO55190.2022.9803398.