RAJALAKSHMI
ENGINEERING COLLEGE
An AUTONOMOUS Institution
Affiliated to ANNA UNIVERSITY, Chennai

AGUMENTED/VIRTUAL REALITY PROJECT REPORT

AR – AR 3D IPHONE MODEL

VR – VR DUNGEON EXPLORE

Submitted by

THEJESH N S 221701503

in partial fulfilment for the course

***C*S23B34 Immersive Experience in AR/VR**

for the degree of

BACHELOR OF ENGINEERING

In

COMPUTER SCIENCE AND DESIGN

RAJALAKSHMI ENGINEERING COLLEGE

RAJALAKSHMI NAGAR

THANDALAM

CHENNAI-602 105

APRIL 2025

RAJALAKSHMI ENGINEERING COLLEGE

CHENNAI – 602105

BONAFIDE CERTIFICATE

Certified that this project report **"AR – AR IPHONE MODEL ,VR - VR DUNGEON EXPLORE"** is the bonafide work of **THEJESH N S (221701503)**, who carried out the project work for the subject CS23B34 – Immersive Experience in AR/VR under my supervision.

SIGNATURE

Prof. Uma Maheshwar Rao

Head of the Department

Computer Science and Design

Rajalakshmi Engineering College

Chennai – 602105

SIGNATURE

Dr. L. Priya.,Ph.D.,

IIIE Coordinator

Rajalakshmi Engineering College

Chennai - 602105

Submitted to Project and Viva Voce Examination for the subject

CS23B34 – Immersive Experience in AR/VR held on_____.

Internal Examiner

External Examiner

ACKNOWLEDGEMENT

Initially we thank the Almighty for being with us through every walk of our life and showering his blessings through the endeavour to put forth this report. Our sincere thanks to our Chairman **Mr. S. Meganathan, B.E, F.I.E.,** our Vice Chairman **Mr. Abhay Shankar Meganathan,B.E.,M.S.,** and our respected Chairperson **Dr. (Mrs.) Thangam Meganathan, Ph.D.,** for providing us with the requisite infrastructure and sincere endeavouring in educating us in their premier institution.

Our sincere thanks to **Dr. S. N. Murugesan, M.E., Ph.D.,** our beloved Principal for his kind support and facilities provided to complete our work in time. We express our sincere thanks to our **Prof. Uma Maheshwar Rao** Associate Professor and Head of the Department of Computer Science and Design for his guidance and encouragement throughout the project work. We convey our sincere thanks to our internal guide and Project Coordinator, **Dr. L. Priya.,Ph.D.,** IIIE Coordinator, Rajalakshmi Engineering College for his valuable guidance throughout the course of the project.

THEJESH N S 221701503

TABLE OF CONTENT

CHAPTER No.	TITLE	PAGE No.
1	**Introduction**	**7**
1.1	Augmented Reality	8
1.2	Virtual Reality	9
1.3	Blender Software	9
1.4	Unity Engine	11
1.5	Vuforia Engine	12
2	**Augmented Reality Project – Virtual Watch Fitting**	16
2.1	Project Overview	16
2.2	Development Process	16
2.3	Step by step procedure to implement the project	17
2.4	Output and Discussion	19
2.5	Conclusion	20
3	**Virtual Reality Project - Pick and Place Object on the Plane Surface**	21

3.1	Project Overview	21
3.2	Module Descriptions	23
3.3	Step by step procedure to implement the project	24
3.4	Output and Discussion	24
3.5	Conclusion	25

LIST OF FIGURES

Fig. No.	TITLE	PAGE No.
1.1	Sample Screenshot of animation in Blender	9
1.2	Sample Output Screenshot of Texturing in Blender	10
1.3	Sample Output Screenshot of Rigging in Blender	10
1.4	Sample Output Screenshot of Rendering in Blender	11
2.1	Output Screenshot of Image Home Screen	19
3.1	Output Screenshot of Drag and Drop object	24

CHAPTER 1

INTRODUCTION

The evolution of technology has consistently reshaped how we interact with the world. Among the most transformative advancements are Augmented Reality (AR) and Virtual Reality (VR), technologies that merge the physical and digital realms to create immersive experiences. These innovations have opened up a multitude of possibilities in fields such as education, healthcare, gaming, architecture, and beyond. Coupled with powerful development tools like Blender and Unity Engine, AR and VR enable the creation of dynamic, interactive environments that captivate users and redefine engagement.

1.1 Augmented Reality (AR)

Augmented Reality enhances the real world by overlaying digital information such as text, images, and 3D objects onto the physical environment. This blending of virtual and physical spaces is achieved through devices like smartphones, tablets, and AR glasses, which utilize cameras, sensors, and software to detect and interact with the surroundings.

Key Features of AR

Real-Time Interaction: AR allows for real-time engagement with digital elements superimposed on the real world.

Device Versatility: From mobile apps to advanced AR headsets like Microsoft HoloLens, AR technology is accessible across various platforms.

Practical Applications: AR is used for navigation, retail, education, and entertainment, offering users a new dimension of interaction.

1.2 Virtual Reality (VR)

Virtual Reality immerses users in a fully digital environment, creating simulated worlds that may mirror reality or depart entirely from it. VR is experienced through headsets such as Oculus Quest or HTC Vive, which use visual, auditory, and sometimes tactile feedback to transport users into virtual spaces.

Key Features of VR

Immersive Environments: VR offers users a 360-degree experience, isolating them from the physical world.

Interactive Simulations: Virtual worlds respond to user input, enabling interaction with the environment.

Applications Across Industries: From gaming and entertainment to professional training and therapy, VR's potential is vast.

1.3 Blender Software

Blender is an open-source, all-in-one software tool for creating 3D content, including modeling, animation, rendering, and simulation. It is widely used in AR and VR projects due to its versatility and compatibility with other platforms. Blender is essential for creating the 3D assets required in AR/VR environments. Once assets are designed, they can be exported into game engines like Unity for further development. Blender's integration with AR/VR workflows makes it a cornerstone tool for immersive content creation. Unity provides specialized frameworks like AR Foundation and XR Interaction Toolkit to streamline the development of AR and VR applications. These frameworks offer built-in functionalities for environment mapping, object interaction, and multi-device support, significantly reducing development time and complexity.

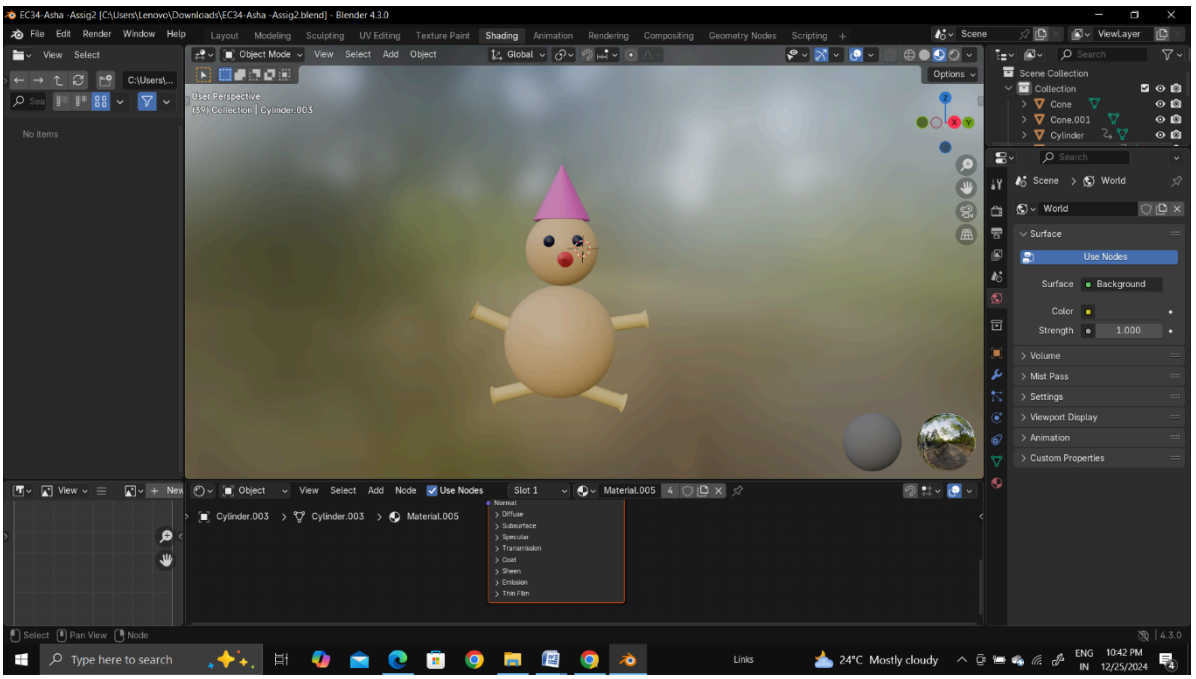

Fig 1.1 Sample Screenshot of animation in Blender

Key Features of Blender:

1. **Modeling**:
    - Blender offers a comprehensive set of modeling tools for creating both low and high-poly models.
    - It supports polygonal modeling, sculpting, and procedural modeling using modifiers.
2. **Sculpting**:
    - Blender provides advanced sculpting tools that are often used for detailed character creation or organic modeling.
    - Dynamic topology and multi-resolution sculpting allow for high-detail work.
3. **Texturing and Shading**:
    - You can create and apply textures using Blender's node-based material system (Cycles and Eevee engines).

- It also supports PBR (Physically Based Rendering) workflows, allowing realistic material and texture creation.

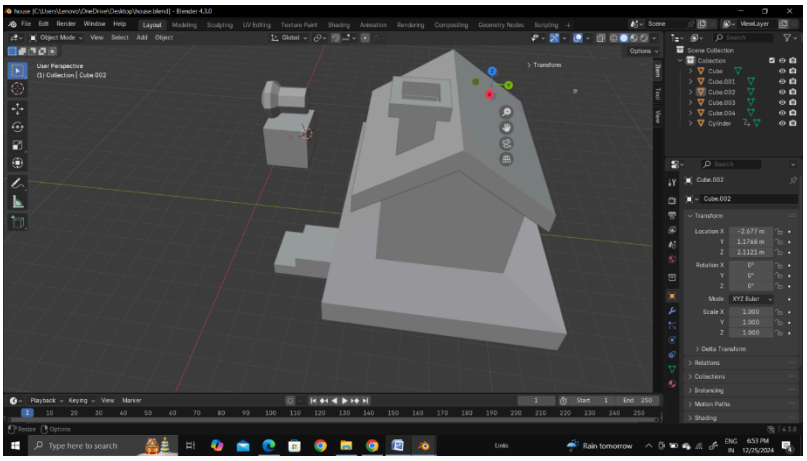

Fig 1.2 Sample Output Screenshot of Texturing in Blender

4. **Rigging and Animation**:
   - Blender has powerful rigging tools that allow for the creation of skeletal structures, inverse kinematics (IK), and character animation.
   - Non-linear animation (NLA) editor and action editor give control over complex animation tasks.
   - Blender also supports shape keys and physics simulations (cloth, smoke, etc.).

Fig 1.3 Sample Output Screenshot of Rigging in Blender

5. **Rendering**:
    - **Cycles**: A path-tracing renderer that produces high-quality images but requires more computational power.
    - **Eevee**: A real-time renderer that provides high-quality results with faster rendering times, ideal for real-time projects and previews.

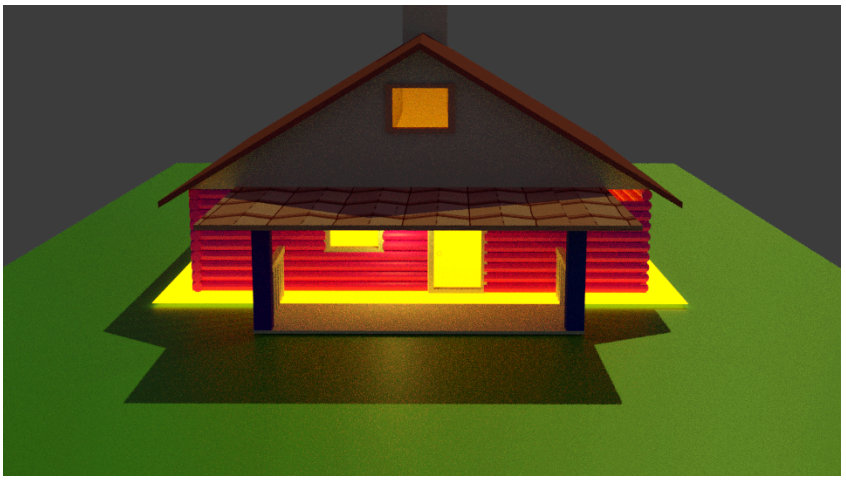

Fig 1.4 Sample Output Screenshot of Rendering in Blender

1.4 Unity Engine

Unity is a robust game engine and development platform that is widely used to create interactive applications across multiple platforms, including AR and VR. Its extensive features, ease of use, and strong community support make it an industry favorite.

Core Features of Unity

- **Cross-Platform Compatibility**: Unity supports deployment to mobile devices, desktops, consoles, and AR/VR devices.
- **Real-Time Development**: Unity's tools allow developers to visualize and test changes instantly.
- **Scripting with C#**: Developers can create complex behaviors and interactions using Unity's scripting environment.
- **Asset Store**: A vast library of pre-made assets, scripts, and tools simplifies the development process.

Augmented Reality, Virtual Reality, Blender, and Unity Engine collectively represent the foundation of modern immersive technology development. Together, they enable the creation of engaging, interactive, and impactful experiences that push the boundaries of what is possible in the digital world. Mastering these technologies equips developers to innovate across industries and deliver transformative applications.

1.5 Vuforia Engine

In Vuforia Engine, the **Image Target** option is used to detect and track images in the real world to display augmented reality (AR) content. Here's how to set up and use Image Targets in Vuforia:

Step-by-Step Guide to Using Image Target in Vuforia

1. Create a Vuforia Developer Account

- Go to the Vuforia Developer Portal and create a free account or log in if you already have one.

2. Create a Vuforia License Key

- Once logged in, go to the **License Manager** section.
- Create a new license key for your project.
- Make a note of the license key because you'll need it later in your project.

3. Download and Set Up Unity (if you haven’t already)

- Download and install Unity, which is the primary development environment for Vuforia.
- Open Unity and create a new project or open an existing one.

4. **Import Vuforia Engine into Unity**

- In Unity, go to the **Unity Asset Store** and search for **Vuforia Engine**.
- Download and import the Vuforia Engine package into your Unity project.

5. Enable Vuforia in Unity

- Go to **Edit > Project Settings > Player**.
- In the **Player Settings** window, click on the **XR Settings** tab.
- Check the **Vuforia Augmented Reality** box to enable Vuforia.

6. Set Up Image Targets in Vuforia

- Open the **Vuforia Configuration** by going to **Window > Vuforia Configuration**.
- In the Vuforia Configuration window, paste your license key (the one you generated earlier).

7. Create an Image Target

- In Unity, create an **Image Target** by right-clicking in the **Hierarchy** window and selecting **Vuforia > Image Target**.
- You should now see an **Image Target** object in your scene.

8. Upload Your Target Image to Vuforia

- Go to the Vuforia Developer Portal and navigate to the **Target Manager**.
- Create a new database and upload the image you want to use as the target.
- Make sure the image is of high quality and has good contrast for better tracking.
- After uploading, download the database and import it into Unity.

9. Configure the Image Target

- In Unity, select the **Image Target** object in your scene.
- In the **Inspector** window, under the **Image Target Behaviour** component, select the image target from the **Database** drop-down.
- Choose the specific image you uploaded in the **Image Target** section.

10. Add AR Content to the Image Target

- You can now add any 3D model, text, or other AR content as children to the Image Target object in the hierarchy.
- For example, to display a 3D model when the image target is detected, drag and drop a 3D model under the Image Target.

11. Test the Scene

- Click **Play** in Unity to test the scene. If everything is set up correctly, the AR content will appear when the image target is detected by the camera.

12. Build and Deploy the Project

- Once you're satisfied with the results in the Unity Editor, you can build and deploy the project to a mobile device.
- For mobile deployment, go to **File > Build Settings**, select your platform (Android or iOS), and then click **Build**.

CHAPTER 2

Augmented Reality Project – Virtual Iphone Model

2.1 Project Overview

AR 3D iPhone Viewer is an augmented reality (AR) application developed using Unity and Vuforia, designed to place a realistic 3D model of an iPhone into real-world space using mid-air tracking. Instead of requiring a predefined image, the application uses spatial understanding to anchor the 3D model on flat surfaces detected by the camera. This demonstrates AR's potential for virtual product demonstration, marketing, and user-interactive experiences, especially in consumer electronics visualization.

2.2 Development Process

1. **Requirement Analysis & Concept Design**
   - Identified the core objective: to allow users to view and interact with a virtual 3D iPhone placed in their environment.
   - Chose Vuforia for mid-air tracking and Unity as the development environment.
   - Decided to use Vuforia's Ground Plane feature for anchoring the iPhone in real space.

2. **Environment Setup**
   - Installed Unity (latest stable version) and integrated Vuforia Engine.
   - Created a new Unity 3D project and set up Vuforia Ground Plane support.
   - Registered and licensed the project on the Vuforia Developer Portal.

3. **Ground Plane Configuration (Mid-Air Target)**
   • Verified the device supports Ground Plane tracking using Vuforia's supported device list.
   • Added Vuforia's AR Camera and Plane Finder components into the Unity scene.
   • Configured the Mid-Air Stage to serve as the anchor point for the 3D iPhone.

4. **3D Model Integration**
   • Imported a detailed 3D model of an iPhone (.fbx or .obj format).
   • Made the iPhone a child of the Mid-Air Stage GameObject to ensure proper placement.
   • Applied realistic materials and textures for visual appeal.

5. **Testing & Debugging**
   • Ran tests in various lighting conditions to ensure the Ground Plane was accurately detected.
   • Adjusted the position and scale of the iPhone model for realism.
   • Resolved bugs related to surface detection, animation glitches, and model drift.

6. **Build & Deployment**
   • Configured build settings for Android platform.
   • Built and installed the APK on a compatible physical device.
   • Conducted real-world user testing and collected feedback for improvements.

2.3 Step-by-Step Procedure to Implement the Project

Step 1: Install and Set Up Tools

1. Install Unity Hub and a compatible version of Unity (preferably LTS version).
2. During installation, add support for:
   • Android Build Support
   • OpenJDK

• Android SDK & NDK

3. Create a Unity Project using the 3D template.

Step 2: Set Up Vuforia and Ground Plane

1. Sign up at developer.vuforia.com.
2. Create a license key and add it to Unity (Window > Vuforia Configuration).
3. Enable Vuforia AR support in Project Settings > XR Plug-in Management.
4. Import the Vuforia Engine package into Unity.

Step 3: Configure AR Scene with Mid-Air Target

1. In the Unity scene, delete the default Main Camera.
2. Add a Vuforia AR Camera to the scene.
3. Add the following to the scene:
   • **Plane Finder**: Detects flat surfaces.
   • **Mid-Air Stage**: Acts as an anchor point for placing virtual objects in the air or on a plane.
4. Assign the 3D iPhone model as a child of the Mid-Air Stage.

Step 4: Import and Set Up 3D iPhone Model

1. Import a high-quality 3D model (.fbx or .obj).
2. Make the model a child of the Mid-Air Stage.
3. Position and scale it so that it appears realistically when placed in the environment.

Step 5: Customize and Enhance

1. Apply textures and materials for realism.
2. Optionally add scripts for touch-based rotation, scaling, or animation triggers.
3. Add UI controls for user interaction (e.g., reset position, model variants).

Step 6: Build and Deploy

1. Go to File > Build Settings and switch platform to Android.
2. Set Player Settings (app name, orientation, etc.).
3. Connect a supported ARCore-enabled Android device.
4. Enable Developer Mode and USB Debugging.
5. Build and Run to deploy the APK.

Step 7: Test and Refine

1. Test in various environments (indoors, outdoors, different lighting).
2. Verify tracking stability and iPhone placement accuracy.
3. Apply finishing tweaks to enhance performance and UX.

2.4 Output and Discussion

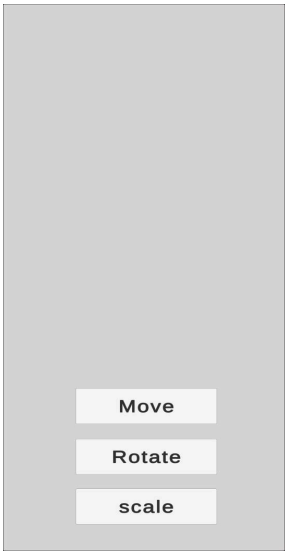

Fig 2.1 Output Screenshot of Image Home Screen

Fig 2.2 Output Screenshot of Image Target

The project successfully leverages Vuforia's Ground Plane to place a virtual 3D iPhone model in mid-air over detected surfaces. This provides an intuitive and interactive experience where users can walk around the virtual product and view it from multiple perspectives, mimicking real-world interaction.

2.5 Conclusion

The AR 3D iPhone model viewer demonstrates the capabilities of mid-air tracking using Vuforia Ground Plane, offering a dynamic method for product visualization. This approach eliminates the need for image markers and enhances immersion by letting users place and explore the virtual object directly in their environment. With Unity and Vuforia, the project delivers a highly visual and interactive experience applicable in tech product demos, education, and virtual retail. The project lays the groundwork for more advanced features such as gesture controls, real-time lighting adaptation, and multi-product AR showrooms.

CHAPTER 3

Virtual Reality Project – VR Dungeon Explore

3.1 Project Overview

The VR Dungeon Explorer is an immersive virtual reality experience designed to simulate the thrill of exploring a dark, mysterious dungeon filled with interactive elements, puzzles, and enemies. Built using Unity's XR Interaction Toolkit for the Oculus Quest, the project focuses on intuitive VR controls, environmental storytelling, and atmospheric immersion. The primary objective was to develop a VR experience that engages the player through exploration, interaction, and light combat or puzzle-solving mechanics.

3.2 Module Descriptions

3.2.1. Player Interaction Module

Purpose: Enables the player to navigate and interact with dungeon elements using VR controllers.

Key Features:

• Grabbing and using items (e.g., keys, torches, weapons).
• Opening doors, pulling levers, and activating mechanisms.
• Intuitive teleportation or smooth locomotion for movement.
• Haptic feedback during interactions and combat.

Tools Used: XR Interaction Toolkit, Unity Input System, C#

3.2.2. Object Physics & Item Handling Module

Purpose: Provides realistic interactions with dungeon objects and props.

Key Features:

• Physics-based object movement and collisions.
• Pickup and drop functionality for weapons and artifacts.
• Interactions like throwing rocks, breaking jars, or triggering traps.

Tools Used: Rigidbody, Collider, XR Grab Interactable, C#

3.2.3. Environment & Navigation Module

Purpose: Designs the dungeon environment and navigation mechanics.
Key Features:
• Modular dungeon layout with interconnected rooms and corridors.
• Hidden paths, doors, and locked areas.
• Lighting systems including dynamic torches and dark zones.
Tools Used: Unity 3D modeling, Lighting System, NavMesh (if AI included)

3.2.4. Puzzle & Interaction Module

Purpose: Adds puzzle elements to enhance gameplay variety.
Key Features:
• Levers, pressure plates, and pattern-based locks.
• Key-item dependencies (e.g., find a key to unlock a gate).
• Feedback mechanisms (audio/visual) for puzzle progress.
Tools Used: Custom C# scripts, Unity Events, Timeline/Animation

3.2.5. Enemy & Combat Module (Optional for basic version)

Purpose: Introduces simple enemies and combat elements.
Key Features:
• Basic AI for patrol and chase behaviors.
• Simple melee or projectile-based combat.
• Health tracking for both player and enemies.
Tools Used: NavMesh Agent (optional), Animations, C# AI scripts

3.2.6. UI & Feedback Module

Purpose: Provides user information and feedback in VR.
Key Features:
• In-world UI panels for objectives or inventory.
• Sound cues for successful/failed actions.
• Haptic feedback for actions and combat.
Tools Used: Unity UI Toolkit, AudioSource, XR haptic system

3.2.7. Game Management Module

Purpose: Manages overall gameplay flow and state transitions.
Key Features:
• Game start, pause, and reset functionality.
• Objective progression (e.g., find artifact, reach exit).
• Level restart or respawn mechanics.
Tools Used: GameManager script, Singleton pattern (optional), C#

3.3 Step-by-Step Procedure to Implement the Project

Step 1: Set Up Unity Project
• Create a new 3D Unity project.
• Install XR Interaction Toolkit and Oculus XR Plugin.
• Enable XR Plugin Management for Oculus Quest support.

Step 2: Build Dungeon Scene
• Create modular dungeon rooms, corridors, and props.
• Add atmospheric lighting and sound for immersion.

Step 3: Set Up XR Rig
• Add XR Origin (Action-Based) to the scene.
• Configure camera and controller input.
• Add teleportation or continuous locomotion systems.

Step 4: Interactive Elements
• Add grab interactions for keys, torches, and levers.
• Use physics to simulate breaking objects or environmental changes.

Step 5: Puzzle and Exploration Logic

- Design rooms with interactive puzzles (e.g., locked doors, levers).
- Create progression logic for unlocking new areas.

Step 6: Enemy and Combat Setup (if included)

- Add basic enemy AI for patrolling.
- Set up simple weapon mechanics like swinging swords.
- Implement health and damage systems.

Step 7: Feedback and UI

- Add in-world UI for current objective or messages.
- Integrate sounds, visual effects, and haptics.

Step 8: Build and Test

- Optimize performance for standalone Quest.
- Build and deploy APK to Oculus Quest.
- Playtest and refine dungeon flow and interactions.

3.4 Output and Discussion

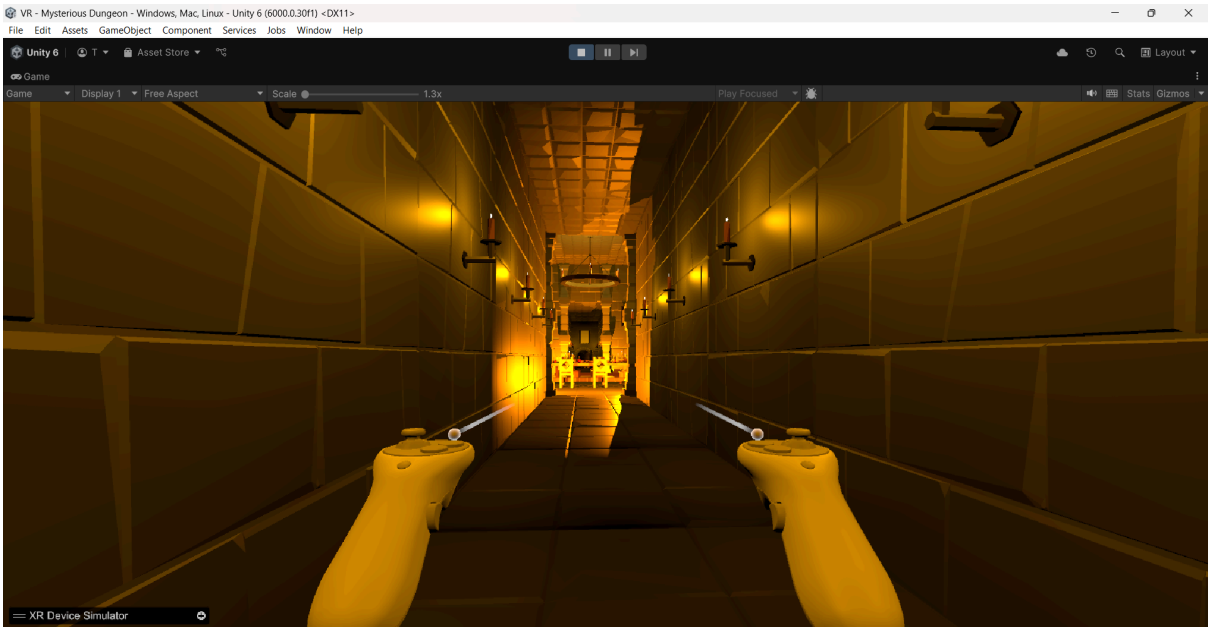

Fig 3.1 Output Screenshot of Dungeon

The final output is an immersive VR dungeon adventure where players can explore dark hallways, solve puzzles, and engage with interactive objects and enemies. The game effectively combines exploration, physics-based interaction, and atmospheric effects to create a rich virtual experience on the Oculus Quest.

3.5 Conclusion

The VR Dungeon Explorer project showcases how virtual reality can be used to craft interactive and immersive experiences in fantasy environments. Using Unity and the XR Interaction Toolkit, the project demonstrates key VR mechanics such as grabbing, locomotion, puzzle-solving, and basic combat. The dungeon theme allowed for creative level design, immersive lighting, and compelling interactivity. Despite challenges in optimizing performance and designing intuitive interactions, the final product is responsive, atmospheric, and highly engaging. This project lays the groundwork for future enhancements such as expanded levels, multiplayer co-op, or RPG-style elements.