



*Mini Project Report On*

## **Insomnia:VR Horror Game**

*Submitted in partial fulfillment of the requirements for the  
award of the degree of*

**Bachelor of Technology**

*in*

**Computer Science & Engineering**

**By**

**Amith Kesav (U2103032)**

**Alfred Antu (U2103026)**

**Aravind Sivadas (U2103046)**

**Ashin Sunny (U2103051)**

**Under the guidance of**

**Mr Paul Augustine**

**Department of Computer Science & Engineering  
Rajagiri School of Engineering & Technology (Autonomous)  
(Affiliated to APJ Abdul Kalam Technological University)  
Rajagiri Valley, Kakkanad, Kochi, 682039  
May 2024**

# CERTIFICATE

*This is to certify that the mini project report entitled "**Insomnia**" is a bonafide record of the work done by **Amith Kesav (U2103032)**, **Alfred Antu (U2103026)**, **Aravind Sivadas (U2103046)**, **Ashin Sunny (U2103051)**, submitted to the APJ Abdul Kalam Technological University in partial fulfillment of the requirements for the award of the degree of Bachelor of Technology (B. Tech.) in Computer Science and Engineering during the academic year 2023-2024.*

Mr Paul Augustine  
Assistant Professor  
Dept. of CSE  
RSET

Mr. Harikrishnan M  
Assistant Professor  
Dept. of CSE  
RSET

Dr. Preetha K.G.  
Professor  
Dept. of CSE  
RSET

## **ACKNOWLEDGEMENTS**

We wish to express our sincere gratitude towards Dr P. S. Sreejith, Principal of RSET, and Dr. Preetha K.G., Head of the Department of Computer Science and Engineering for providing us with the opportunity to undertake our mini project, "Insomnia".

We are highly indebted to our project coordinators, **Mr. Harikrishnan M** and **Ms. Sherine Sebastian**, Assistant Professor, Department of Computer Science and Engineering for their valuable support.

It is indeed our pleasure and a moment of satisfaction for us to express our sincere gratitude to our project guide **Mr. Paul Augustine** for his patience and all the priceless advice and wisdom he has shared with us.

Last but not the least, we would like to express my sincere gratitude towards all other teachers and friends for their continuous support and constructive ideas.

**Amith Kesav**

**Alfred Antu**

**Aravind Sivadas**

**Ashin Sunny**

## Abstract

Insomnia is an immersive virtual reality (VR) horror game designed to plunge players into a realm of relentless terror and psychological suspense. Developed using Unity, C#, and Blender, this spine-chilling experience harnesses cutting-edge technology to deliver a truly visceral encounter.

Unity's key features such as cross-platform capability, Visual editor, Asset store, animation tools etc. provides great development options and potential for VR Game development. The C# is used to define game logic and for defining physics and actions in the game environment.

Blender's versatile toolkit is employed to craft meticulously detailed environments, from derelict corridors to haunted mansions, each designed to intensify the sense of dread and isolation. The combination of these technologies delivers an unparalleled horror experience, where players' senses are constantly on edge as they delve deeper into the darkness.

Utilizing the immersive capabilities of VR, players are fully immersed in a world where every shadow hides a lurking threat, and every sound sends shivers down their spine. The game leverages Unity's powerful engine to create stunning visuals and dynamic lighting effects, while C# scripting ensures seamless gameplay mechanics and interactions.

# Contents

<b>Acknowledgements</b>	<b>i</b>
<b>Abstract</b>	<b>ii</b>
<b>List of Figures</b>	<b>vi</b>
<b>List of Abbreviations</b>	<b>vii</b>
<b>1 Introduction</b>	<b>1</b>
1.1 Background . . . . .	1
1.2 Problem Definition . . . . .	2
1.3 Scope and Motivation . . . . .	2
1.4 Objectives . . . . .	3
1.5 Challenges . . . . .	3
1.6 Assumptions . . . . .	4
1.7 Societal / Industrial Relevance . . . . .	4
1.8 Organization of the Report . . . . .	4
<b>2 Software Requirements Specification</b>	<b>6</b>
2.1 Introduction . . . . .	6
2.1.1 Purpose . . . . .	6
2.1.2 Product Scope . . . . .	6
2.2 Overall Description . . . . .	6
2.2.1 Product Perspective . . . . .	6
2.2.2 Product Functions . . . . .	7
2.2.3 Operating Environment . . . . .	7
2.2.4 Design and Implementation Constraints . . . . .	7
2.2.5 Dependencies . . . . .	7
2.3 External Interface Requirements . . . . .	8

2.3.1	User Interfaces . . . . .	8
2.3.2	Hardware Interfaces . . . . .	8
2.3.3	Software Interfaces . . . . .	8
2.3.4	Communications Interfaces . . . . .	9
2.4	System Features . . . . .	9
2.4.1	Immersive Environment Creation . . . . .	9
2.5	Other Nonfunctional Requirements . . . . .	10
2.5.1	Performance Requirements . . . . .	10
2.5.2	Safety Requirements . . . . .	11
2.5.3	Security Requirements . . . . .	11
2.5.4	Software Quality Attributes . . . . .	12
<b>3</b>	<b>System Architecture and Design</b>	<b>13</b>
3.1	System Overview . . . . .	13
3.1.1	Frontend (VR Interface) . . . . .	13
3.1.2	Backend Services . . . . .	13
3.1.3	Database . . . . .	14
3.2	Architectural Design . . . . .	14
3.3	User Interface Design . . . . .	15
3.4	Database Design . . . . .	17
3.5	Description of Implementation Strategies . . . . .	17
3.5.1	Virtual Reality Development Framework . . . . .	17
3.5.2	Game Logic Implementation . . . . .	18
3.5.3	Environmental Design . . . . .	18
3.5.4	Audio Implementation . . . . .	18
3.5.5	User Interaction and Input Handling . . . . .	18
3.5.6	Performance Optimization . . . . .	18
3.6	Module Division . . . . .	19
3.6.1	Player Movement and Interaction . . . . .	19
3.6.2	Environmental Design and Asset Integration . . . . .	19
3.6.3	Game Logic and Event Triggering . . . . .	19
3.6.4	Audio Design and Implementation . . . . .	19

3.6.5	User Interface and Menu Systems . . . . .	19
3.6.6	Performance Optimization and Testing . . . . .	19
3.7	Work Schedule - Gantt Chart . . . . .	20
<b>4</b>	<b>Results and Discussions</b>	<b>21</b>
4.1	Overview . . . . .	21
4.2	Testing . . . . .	22
4.3	Discussion . . . . .	25
<b>5</b>	<b>Conclusion</b>	<b>26</b>
5.1	Conclusion . . . . .	26
5.2	Future Scope . . . . .	27
5.2.1	Multi-Platform Support . . . . .	27
5.2.2	Innovative Interaction Mechanics . . . . .	27
5.2.3	Advanced Audiovisual Realism . . . . .	27
<b>Appendix A: Presentation</b>	<b>29</b>	
<b>Appendix B: Vision, Mission, Programme Outcomes and Course Outcomes</b>	<b>43</b>	
<b>Vision, Mission, POs, PSOs and COs</b>	<b>44</b>	
<b>Appendix C: CO-PO-PSO Mapping</b>	<b>48</b>	

## List of Figures

3.1	System Architecture . . . . .	14
3.2	Use Case Diagram . . . . .	15
3.3	Start Screen . . . . .	15
3.4	Pause Screen . . . . .	16
3.5	End Screen . . . . .	16
3.6	ER Diagram . . . . .	17
3.7	Work schedule Gantt chart . . . . .	20
4.1	Start Screen . . . . .	22
4.2	Pause Screen . . . . .	22
4.3	Game play 1 . . . . .	23
4.4	Game play 2 . . . . .	23
4.5	Game play 3 . . . . .	24
4.6	End Screen . . . . .	24

## **List of Abbreviations**

- VR - Virtual Reality
- SDK - Software Development Kit
- API - Application programming interface
- NPC - Non-player character
- AI - Artificial intelligence
- HUD - Head Up Display
- UI - User Interface
- XR - Extended Reality

# Chapter 1

## Introduction

### 1.1 Background

In recent years, Virtual Reality (VR) technology has rapidly evolved, captivating audiences with its immersive experiences. One captivating genre within VR is horror games, where players are transported into terrifying worlds, experiencing fear like never before. This document outlines the background and importance of developing a VR-based horror game project. Despite the growing interest, there remains a scarcity of high-quality, immersive VR horror experiences in the market.

#### **Importance of the Project:**

1. Immersive Gameplay: VR technology offers unparalleled immersion, allowing players to feel as if they are truly present in the game world. In a horror game, this heightened sense of presence intensifies fear, delivering a more thrilling and memorable experience.
2. Innovation in Storytelling: VR opens up new possibilities for storytelling, enabling developers to craft narratives that deeply resonate with players.
3. Market Opportunity: The demand for VR content continues to grow, presenting a lucrative opportunity for developers. A well-executed VR horror game has the potential to attract a wide audience of gamers eager for spine-tingling experiences, thus driving sales and garnering critical acclaim.
4. Community Engagement: Horror games have a dedicated fan base eager to explore new titles and share their experiences with others. By creating a compelling VR horror game, developers can foster a vibrant community of players who eagerly anticipate future releases and actively participate in discussions and content creation.

## **1.2 Problem Definition**

Meeting the rising demand for innovative and immersive gaming experiences in the VR market. Providing users with an escape from reality through richly detailed virtual environments and compelling gameplay mechanics. Contributing to the advancement of VR technology and its integration into mainstream gaming culture.

## **1.3 Scope and Motivation**

Our project centers around the development of a VR Horror game, employing the Unity Engine, C#, and Blender to craft an unforgettable experience for players. With these powerful tools at our disposal, our primary objective is to immerse players in spine-chilling virtual environments, where every shadow and sound contributes to the sense of dread and terror. By harnessing the capabilities of Unity Engine for realistic graphics and interactive gameplay, coupled with Blender's versatility in creating intricate 3D models and environments, we aim to deliver an unparalleled horror gaming experience. Through careful attention to detail and a commitment to atmospheric storytelling, our game seeks to evoke genuine fear and suspense, ensuring that players are left trembling long after they've removed their VR headset.

Our project addresses the growing demand for immersive virtual reality experiences, tapping into a market hungry for innovative gaming content. By creating a VR game that combines the power of Blender's versatile 3D modeling, Unity's robust game development platform, and C#'s flexible scripting capabilities, we aim to deliver an unparalleled gaming experience that captivates players and pushes the boundaries of VR technology. Our innovative approach allows us to craft rich, interactive worlds with stunning visuals and dynamic gameplay mechanics, setting our game apart in a crowded market. This project aligns with our vision of revolutionizing the gaming industry by leveraging cutting-edge technology to create memorable, engaging experiences for players of all ages. While challenges may arise in the development process, we're committed to overcoming them with creativity, collaboration, and a dedication to delivering a VR game that excites and inspires.

## **1.4 Objectives**

- Intuitive Controls: Implement intuitive and responsive controls tailored for VR, utilizing hand presence and gestures.
- Immersive Visuals: Utilize Blender for high-quality 3D models and textures to create visually stunning and terrifying environments.
- Dynamic Lighting: Use Unity's lighting system to create dynamic and realistic lighting that enhances the horror atmosphere.
- Interactive Objects: Implement interactive objects that players can manipulate or use to progress through the game.
- Multi-Path Storyline: Introduce branching storylines based on player choices, adding replay value and unpredictability.
- Progressive Horror Experience: Design the game to intensify the horror gradually, building tension throughout the gameplay.

## **1.5 Challenges**

Developing VR games presents challenges such as optimizing performance for smooth gameplay, designing intuitive interfaces to prevent motion sickness, and creating immersive content tailored for VR environments. Additionally, ensuring compatibility across various VR platforms and devices, and addressing the high cost of entry for players are important considerations for developers. Balancing player comfort, technical constraints, and engaging gameplay mechanics are critical aspects of successful VR game development.

## **1.6 Assumptions**

Developing VR games presents challenges such as optimizing performance for smooth gameplay, designing intuitive interfaces to prevent motion sickness, and creating immersive content tailored for VR environments. Additionally, ensuring compatibility across various VR platforms and devices, and addressing the high cost of entry for players are important considerations for developers. Balancing player comfort, technical constraints, and engaging gameplay mechanics are critical aspects of successful VR game development.

## **1.7 Societal / Industrial Relevance**

The societal and industrial relevance of VR games lies in their potential to revolutionize entertainment, education, and various industries. VR games offer immersive experiences that can enhance learning, training, and therapeutic applications, impacting fields such as healthcare, education, and professional training. Furthermore, the growing VR market presents opportunities for innovation, economic growth, and job creation within the gaming industry and related sectors. As VR technology becomes more accessible and advanced, its societal and industrial relevance will continue to expand, shaping the way we interact with digital content and virtual environments.

## **1.8 Organization of the Report**

- **Chapter 1-Introduction:** This chapter introduces the background of the project, defines the problem statement, discusses the scope and motivation, outlines the objectives, highlights the societal and industrial relevance, addresses assumptions, and identifies challenges faced during the project development.
- **Chapter 2-Software Requirements Specification:** This chapter delineates the functional and nonfunctional requirements of the VR game. It defines the overall description of the software, external interface requirements, system features, and other nonfunctional requirements necessary for the development and deployment of the game.
- **Chapter 3-System Architecture and Design:** The system architecture and design chapter provides an overview of the technical framework of the VR game. It

includes discussions on the system overview, architectural design, identified datasets, proposed algorithms, implementation strategies, module division, and a work schedule presented as a Gantt chart for project planning and management.

- **Chapter 4 - Results and Discussions:** This chapter presents the results obtained from the project. Discussions on the implications of these results, any deviations from expected outcomes, and possible reasons for these deviations are also included.
- **Chapter 5 - Conclusion:** This chapter provides a summary of the project, reiterates the main findings, and reflects on the achievements and limitations of the project. It offers concluding remarks on the overall success of the game and its potential impact.

# **Chapter 2**

## **Software Requirements Specification**

### **2.1 Introduction**

#### **2.1.1 Purpose**

This Software Requirements Specification (SRS) document outlines the requirements for the development of a VR game. To create an immersive game for entertainment purpose. The software requirements specified in this document pertain to the development of a VR Horror game using Unity Engine, C#, and Blender

#### **2.1.2 Product Scope**

The software being specified is a VR Horror game developed using Unity Engine, C#, and Blender. Its purpose is to deliver immersive and terrifying experiences to players within virtual reality environments.

### **2.2 Overall Description**

#### **2.2.1 Product Perspective**

The VR Horror game developed using Unity Engine, C#, and Blender is a self-contained product aimed at delivering immersive and terrifying experiences to players in virtual reality. It is not a follow-on member of a product family but rather a standalone creation intended to provide unique gameplay and storytelling within the horror genre. This game serves as a replacement for traditional horror gaming experiences, offering a heightened level of immersion and interactivity through virtual reality technology. Its primary function is to deliver a spine-chilling atmosphere, intense gameplay mechanics, and compelling narratives tailored specifically for the VR medium.

### **2.2.2 Product Functions**

Create immersive horror environments. Implement interactive gameplay mechanics. Support VR headset integration. Provide compelling storytelling. Incorporate 3D models and animations. Enable user interaction with virtual objects. Implement audio and visual effects for atmosphere. Support multiple levels or scenarios. Allow for player movement and exploration.

### **2.2.3 Operating Environment**

Hardware Platform: VR-compatible devices such as Oculus Rift, HTC Vive, or PlayStation VR. Operating System: Compatibility with major operating systems, including Windows, macOS, and potentially Linux distributions supported by Unity Engine. Software Components: Unity Engine (version 2021.3.37f1 or later), Blender for 3D asset creation, and C# scripting language for game logic. Additionally, it may interact with VR platform specific SDKs for optimal performance and features. Other Applications: Peaceful coexistence with system-level software and drivers necessary for VR hardware operation, as well as any necessary third-party software libraries or tools used in the development pipeline.

### **2.2.4 Design and Implementation Constraints**

The game's performance and features may be limited by the processing power, memory, and graphical capabilities of the target VR hardware platforms. The use of Unity Engine, C#, and Blender for development imposes constraints on the choice of programming languages, 3D modeling tools, and game engine features. The game must adhere to the specifications and APIs provided by VR platform-specific SDKs to ensure compatibility and optimal performance on devices like Oculus Rift, PlayStation VR. Compliance with industry standards and regulations related to content rating, privacy, and safety may impose constraints on the game's design and content.

### **2.2.5 Dependencies**

Unity Engine: The project depends on the stable functioning and timely updates of Unity Engine for development purposes. Blender: Dependency on Blender for 3D modeling

and asset creation. VR platform SDKs: Dependency on VR platform-specific SDKs for integration with VR hardware devices and optimal performance.

## **2.3 External Interface Requirements**

### **2.3.1 User Interfaces**

- GUI standards will adhere to Unity's UI guidelines for consistency and usability.
- Standard buttons such as "Start Game," "Options," "Quit," and "Continue" will appear on appropriate screens.
- Keyboard shortcuts will be implemented for common actions, like "ESC" for pausing the game or accessing the menu.
- Error messages will be displayed clearly and concisely, avoiding technical jargon to maintain immersion.

### **2.3.2 Hardware Interfaces**

The game will support PC hardware components such as keyboard, mouse, and standard controllers. VR hardware compatibility will be integrated, including headsets like Oculus Rift and HTC Vive. Physical characteristics of interactions will vary based on the hardware, with VR interactions leveraging hand movements and gestures.

### **2.3.3 Software Interfaces**

- Unity Engine: The game will utilize Unity for development and runtime environment.
- Blender: Blender will be used for 3D modeling and animation.
- C#: Game logic and scripting will be implemented using C#.
- Database: No direct interaction with databases is required for this game.
- Operating System: The game will be compatible with Windows 8.1 and potentially other operating systems based on Unity's platform support.

- Libraries and Tools: Standard Unity libraries and tools will be utilized for development.
- Data Sharing: Limited data sharing between software components will occur, primarily related to game state and player input.

#### **2.3.4 Communications Interfaces**

No direct communication functions like email or web browser are required for the game.

Network Server: No network server communication is needed for single-player gameplay.

Message Formatting: Internal messaging within the game will adhere to standard protocols within Unity's scripting environment.

Security: Encryption is not necessary for this offline single-player game.

Data Transfer Rates: Data transfer rates are not a critical concern for local gameplay.

Synchronization: Game state synchronization will be handled internally by Unity's scripting and game engine components

### **2.4 System Features**

#### **2.4.1 Immersive Environment Creation**

##### **Description and Priority**

This feature involves creating immersive and terrifying environments for the VR Horror game. It is of high priority as it forms the foundation of the player's experience.

##### **Stimulus/Response Sequences**

- User selects "Start Game" from the main menu.
- System loads the selected game level.
- User enters the virtual environment.
- System renders detailed 3D environments with appropriate lighting and atmospheric effects.
- User explores the environment, triggering scripted events and encounters.

## **Functional Requirements**

- REQ-1: The system shall allow designers to import 3D models and textures for environmental elements.
- REQ-2: The system shall support dynamic lighting and particle effects to enhance atmosphere.
- REQ-3: The system shall provide tools for level designers to script environmental events and interactions.
- REQ-4: The system shall render environments in real-time, maintaining a smooth frame rate for optimal VR experience.
- REQ-5: The system shall ensure compatibility with various VR hardware platforms to deliver consistent experiences across devices.
- REQ-6: The system shall incorporate audio cues and environmental sounds to enhance immersion.
- REQ-7: The system shall optimize environmental assets and textures for efficient memory usage and loading times.
- REQ-8: The system shall provide error handling for cases where environmental assets fail to load or render properly.

## **2.5 Other Nonfunctional Requirements**

### **2.5.1 Performance Requirements**

CPU: Intel Core i5-4590/AMD FX 8350 equivalent or better.

GPU: NVIDIA GTX 970 / AMD Radeon R9 290 equivalent or better.

RAM: 8GB or more.

Operating System: Windows 8.1, or Windows 10.

Storage: At least 10GB of free space.

VR Headset compatible with SteamVR.

### **2.5.2 Safety Requirements**

Comfort Settings: Offer options like adjustable movement speeds to reduce motion sickness.

User Warnings: Clearly inform players about potential risks like motion sickness before they start.

Break Reminders: Prompt players to take breaks to avoid eye strain and fatigue.

Environment Awareness: Ensure players have safe space to move around in reality.

Emergency Procedures: Provide instructions to exit VR safely in case of discomfort.

Age Restrictions: Consider limiting access for young children to age-appropriate content.

Content Sensitivity: Ensure game content is suitable and not distressing.

Hardware Safety: Regularly check VR equipment for safety.

### **2.5.3 Security Requirements**

Data Encryption: Protect sensitive user data with encryption to prevent unauthorized access.

Secure Authentication: Use secure methods like two-factor authentication to verify user identity.

Secure Communication: Encrypt communication between the game client and servers to prevent tampering.

Input Validation: Validate user input to prevent common security vulnerabilities.

Anti-Cheating Measures: Implement tools to detect and prevent cheating. Content Protection: Safeguard game assets from unauthorized copying or modification.

Privacy Compliance: Follow privacy regulations and obtain user consent for data processing.

Regular Security Audits: Conduct frequent security checks to identify and fix vulnerabilities.

#### **2.5.4 Software Quality Attributes**

Performance: Ensure smooth gameplay without lag by optimizing rendering and scripting.

Reliability: Minimize crashes and glitches through thorough testing and debugging.

Scalability: Make the game adaptable to various VR devices without compromising quality.

Usability: Create intuitive interfaces and clear instructions to enhance player experience.

Maintainability: Keep the code organized for easy updates and bug fixes in the future.

Security: Protect user data and prevent cheating with encryption and authentication.

Portability: Make the game compatible across different platforms for a wider audience.

Accessibility: Support diverse needs with alternative controls and adjustable settings.

# Chapter 3

## System Architecture and Design

### 3.1 System Overview

The project aims to develop an immersive Virtual Reality (VR) based game that provides players with an engaging and interactive experience. Leveraging the latest VR technologies, the game will transport players to fantastical worlds where they can explore, interact with objects and characters, solve puzzles, and embark on thrilling adventures. The system architecture is designed to be modular and scalable, allowing for flexibility in accommodating future expansions and changes. Here's an overview of the architecture:

#### 3.1.1 Frontend (VR Interface)

The frontend of the game will be the VR interface itself. It will be developed using game engine Unreal Engine, which provide robust support for VR development. The VR interface will render the game environment, characters, and objects in 3D, and it will handle player input from VR controllers or other input devices.

#### 3.1.2 Backend Services

The backend of the game system will consist of various services that support different aspects of the game:

- **Game Logic Service:** This service will manage the core game logic, including player interactions, NPC behaviors, game events, and progression.
- **Asset Management Service:** To handle the vast amount of 3D assets required for the game, an asset management service will be responsible for loading, unloading, and streaming assets based on player location and interactions.

- **Analytics Service:** This service will collect and analyze gameplay data to provide insights into player behavior, identify performance bottlenecks, and inform game design decisions.

### 3.1.3 Database

A database system will store various game data, including player profiles, game states, assets metadata, and analytics data.

## 3.2 Architectural Design

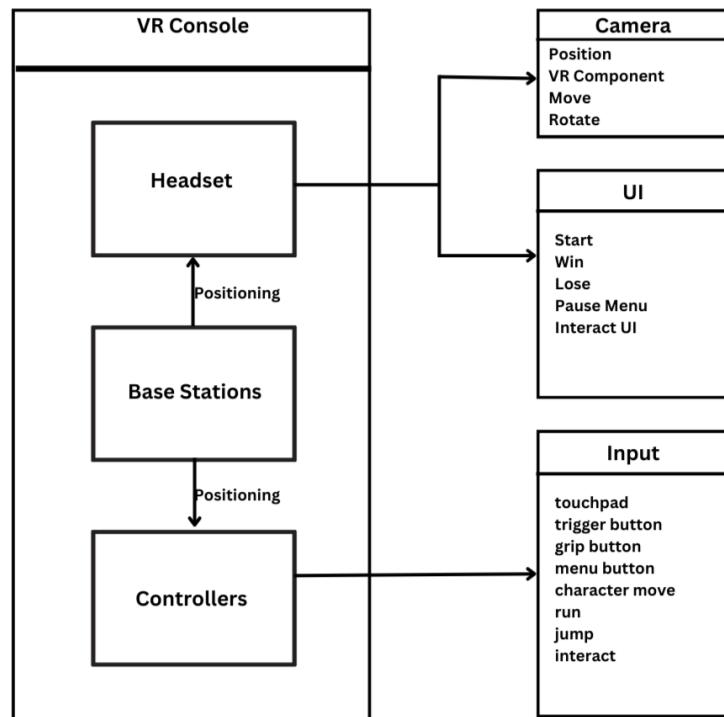


Figure 3.1: System Architecture

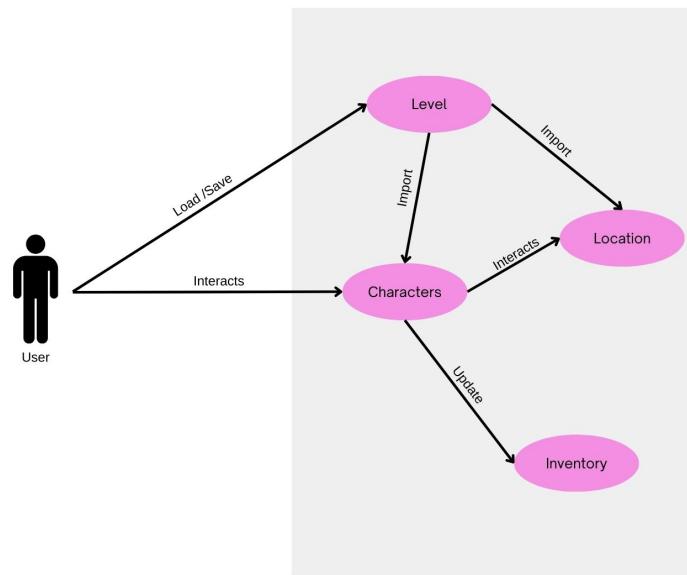


Figure 3.2: Use Case Diagram

### 3.3 User Interface Design

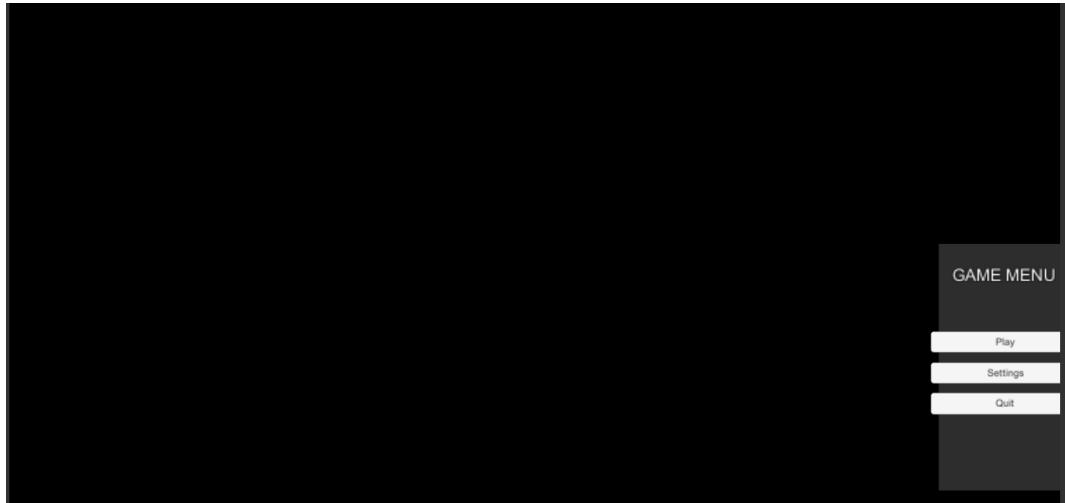


Figure 3.3: Start Screen

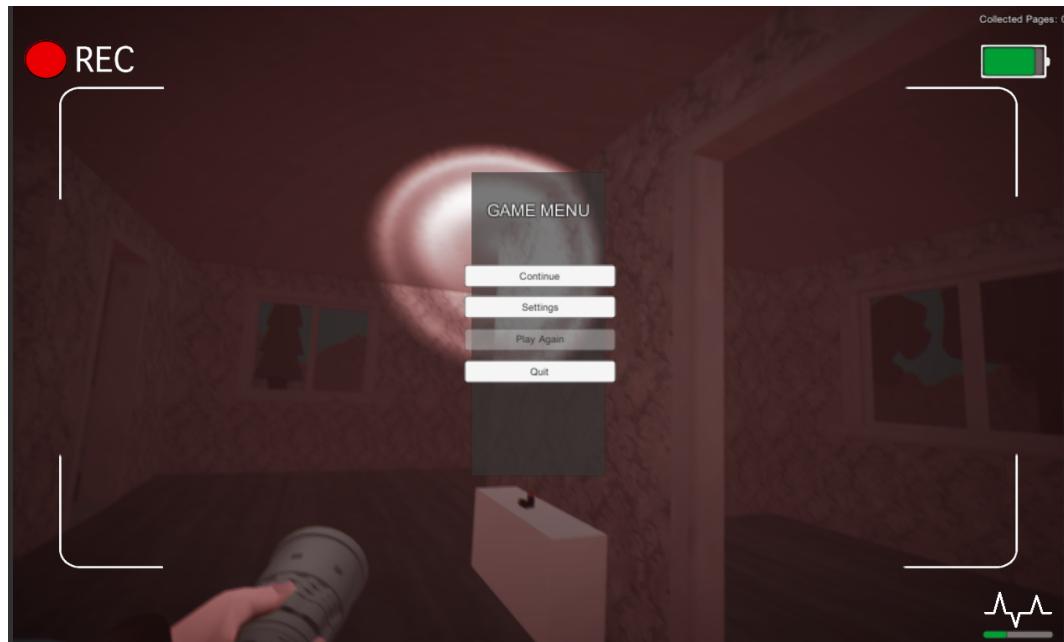


Figure 3.4: Pause Screen



Figure 3.5: End Screen

## 3.4 Database Design

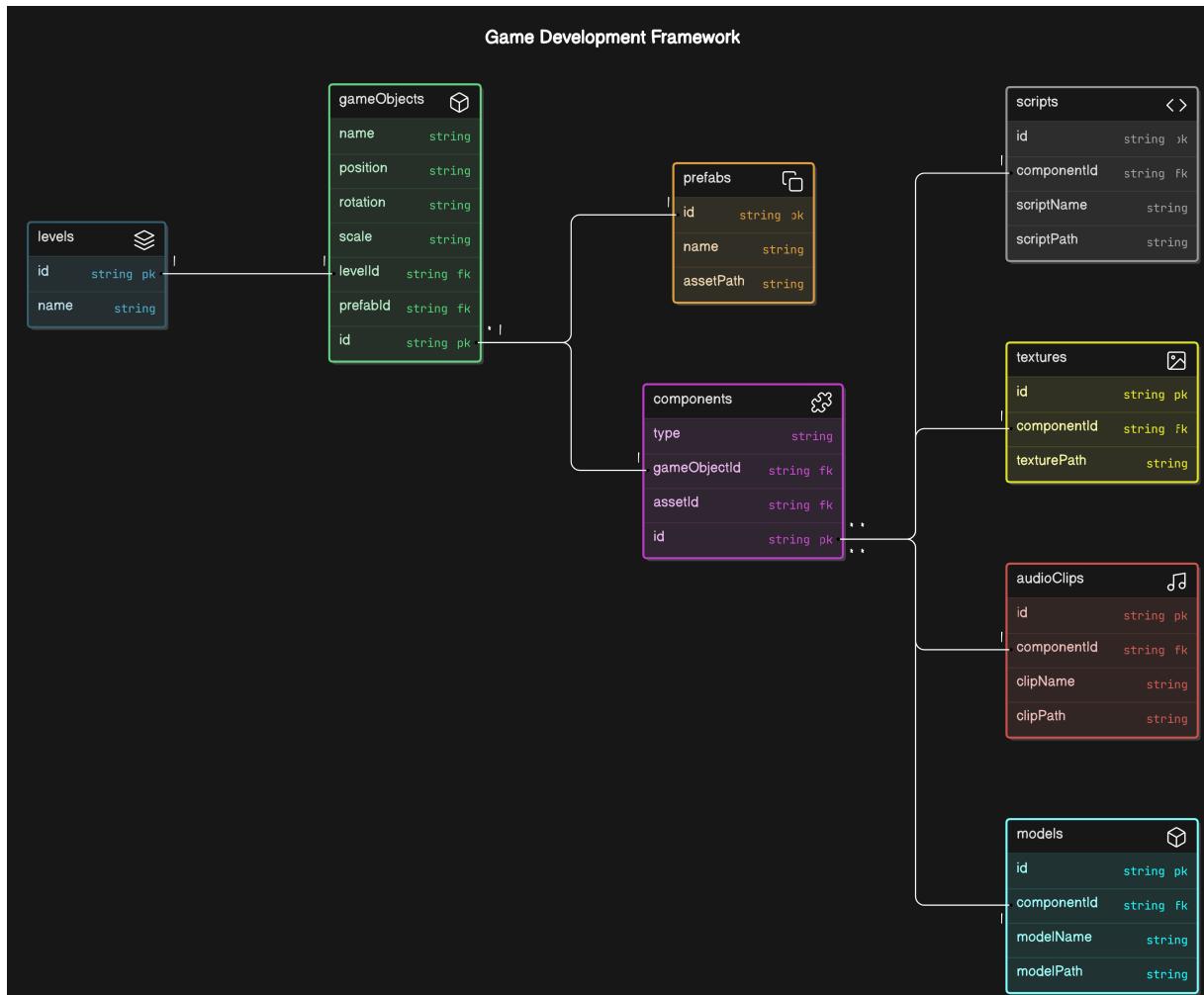


Figure 3.6: ER Diagram

## 3.5 Description of Implementation Strategies

### 3.5.1 Virtual Reality Development Framework

For developing the VR horror game, we utilized a robust virtual reality development framework such as Unity3D. These frameworks provide comprehensive support for VR development, including rendering, input handling, physics simulation, and audio spatialization.

### **3.5.2 Game Logic Implementation**

The game logic for the VR horror experience was implemented using C# in Unity3D. This involved coding various gameplay mechanics such as player movement, interaction with objects, AI behaviors for enemies, and triggering of scares and events.

### **3.5.3 Environmental Design**

The eerie and immersive environments in the VR horror game were designed using 3D modeling software such as Blender. These environments were then imported into Unity3D and optimized for performance in VR.

### **3.5.4 Audio Implementation**

To create a chilling auditory experience, we utilized audio libraries such as FMOD or Wwise for spatial audio and dynamic sound mixing. These libraries allow for precise control over sound positioning, attenuation, and effects, enhancing the immersive horror atmosphere.

### **3.5.5 User Interaction and Input Handling**

Input from VR controllers or other input devices was managed using built-in input handling systems provided by Unity3D. This involved mapping controller inputs to in-game actions such as movement, interaction, and inventory management.

### **3.5.6 Performance Optimization**

Given the resource-intensive nature of VR, performance optimization techniques such as level of detail (LOD) management, occlusion culling, and texture compression were employed to ensure smooth and immersive gameplay across a variety of VR hardware platforms.

## **3.6 Module Division**

### **3.6.1 Player Movement and Interaction**

This module handles player movement within the virtual environment and interaction with objects. It includes functionalities such as walking, running, grabbing objects, and interacting with interactive elements in the game world.

### **3.6.2 Environmental Design and Asset Integration**

This module focuses on designing the eerie and immersive environments of the horror game. It involves creating 3D models, textures, and animations for environmental assets and integrating them into the game engine.

### **3.6.3 Game Logic and Event Triggering**

The game logic module is responsible for implementing various gameplay mechanics and triggering events based on player actions or predefined triggers. It includes functionalities such as enemy AI behavior, scare events, puzzle mechanics, and level progression.

### **3.6.4 Audio Design and Implementation**

This module deals with the creation and implementation of audio assets to enhance the horror atmosphere of the game. It includes sound effects, ambient sounds, music, and voice acting. Spatial audio techniques are utilized to create an immersive auditory experience.

### **3.6.5 User Interface and Menu Systems**

The user interface module focuses on designing and implementing the in-game HUD, menu systems, and user interaction interfaces. It includes functionalities such as health indicators, inventory management, settings menu, and in-game prompts.

### **3.6.6 Performance Optimization and Testing**

This module is responsible for optimizing the performance of the game to ensure smooth gameplay and immersive experience across different VR platforms. It involves techniques

such as level optimization, texture compression, and performance profiling. Testing activities include debugging, playtesting, and performance testing.

### 3.7 Work Schedule - Gantt Chart

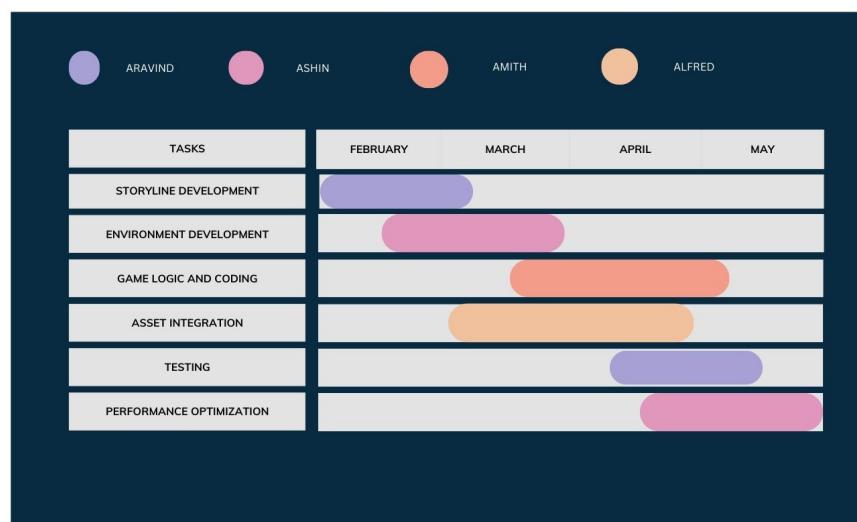


Figure 3.7: Work schedule Gantt chart

# **Chapter 4**

## **Results and Discussions**

### **4.1 Overview**

Insomnia VR Game, developed with Unity and C#, offers an immersive horror experience in virtual reality. Players explore a haunted house, solve puzzles, and evade terrifying enemies. The game features realistic graphics and sound effects that create a spine-chilling atmosphere. Players can interact with objects, unlock doors, and solve puzzles, making the gameplay engaging and immersive. Detailed 3D models of the environment and characters, created using Blender, enhance the eerie setting. Smart AI-driven enemies provide unpredictable and challenging encounters, adding to the suspense. Objects in the game react believably to player actions, thanks to Unity's physics engine, ensuring realistic interactions. Dynamic sound and visual effects react to player actions and enemy presence, heightening the sense of fear.

## 4.2 Testing

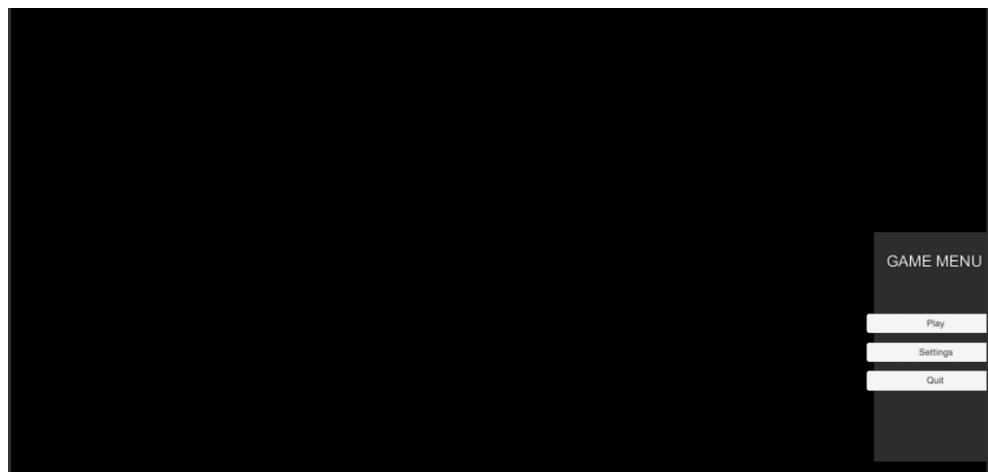


Figure 4.1: Start Screen

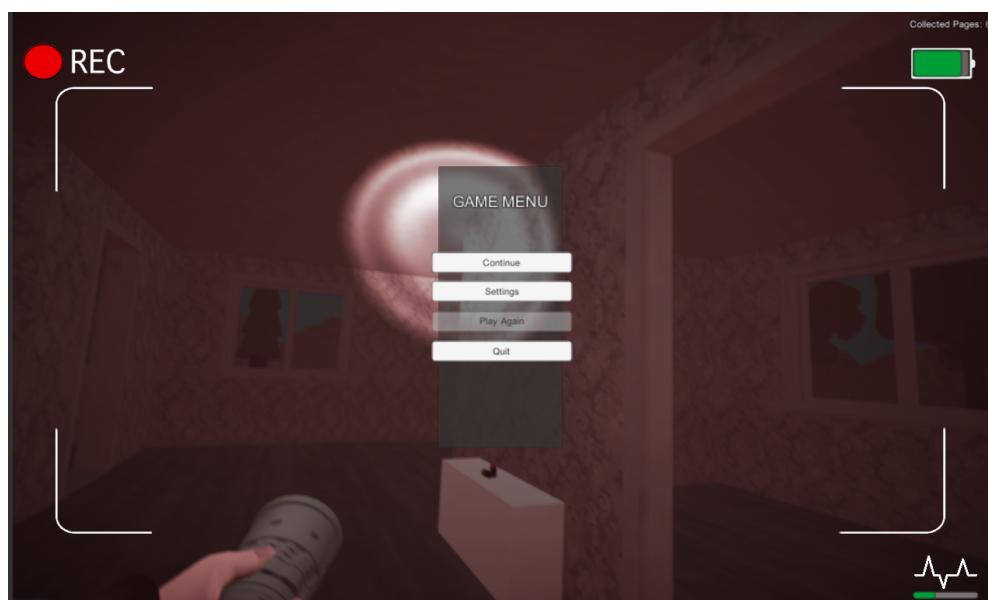


Figure 4.2: Pause Screen



Figure 4.3: Game play 1



Figure 4.4: Game play 2



Figure 4.5: Game play 3



Figure 4.6: End Screen

### **4.3 Discussion**

The development and testing of the horror game "Insomnia" using Unity, C#, and Blender yielded several key findings. The primary objective was to create an immersive horror experience that effectively uses advanced graphics, sound design, and gameplay mechanics to evoke fear and tension.

User feedback revealed that the game achieved a high level of immersion and fear induction. Most players reported feeling genuinely scared and engaged, noting the effectiveness of the game's environment and sound design. The realistic environments, designed and textured in Blender, and the dynamic lighting and shadows programmed in Unity, contributed significantly to the game's atmosphere. The carefully crafted soundscapes, including ambient noises, sudden audio cues, and eerie background music, were highlighted as key elements that increased the sense of dread and anticipation.

Some players reported performance issues, such as frame rate drops and occasional glitches, particularly on lower-end systems, indicating a need for further optimization. A minority of players became desensitized to the horror elements after prolonged exposure, suggesting a need for more varied and unpredictable scare tactics. A small number of players experienced motion sickness, indicating the necessity of implementing additional comfort options, such as adjustable camera movements or reduced motion effects.

The positive results can largely be attributed to the effective use of advanced game development technologies, which enhance immersion and presence. Unity's powerful game engine capabilities and Blender's advanced modeling tools allowed for the creation of a highly detailed and interactive game world. Meticulous attention to detail in both visual and audio design ensured that players were consistently engaged and on edge. Continuous testing and incorporation of player feedback throughout the development process allowed for refinements that enhanced the game's overall quality and effectiveness.

Overall, the development of "Insomnia" successfully demonstrated the potential of advanced game development techniques in creating immersive and effective horror experiences. While there were some challenges and areas for improvement, the positive feedback on immersion and fear induction highlights the strengths of the game's design and implementation. Future iterations should focus on optimizing performance, increasing scare variety, and enhancing player comfort to further improve the experience.

# Chapter 5

## Conclusion

### 5.1 Conclusion

In conclusion, our VR-based Slenderman game promises an unparalleled horror experience that will captivate and terrify players. By leveraging cutting-edge virtual reality technology, we've crafted a game that immerses players in a world where every shadow holds a secret and every sound sends shivers down their spine.

From the moment players start the game, they are transported into a chilling, dark forest where the haunting presence of Slenderman lurks around every corner. The advanced VR technology allows for a level of immersion that traditional gaming platforms simply cannot match. Players can explore the environment in a way that feels incredibly realistic, making them feel as though they are truly in the game's world.

With its stunning visuals, realistic environments, and spine-chilling encounters with the iconic Slenderman, our game pushes the boundaries of immersive entertainment. The attention to detail in the game's design is evident in the lifelike textures of the forest, the eerie lighting that creates a sense of dread, and the subtle audio cues that make every rustle of leaves or distant footsteps a potential threat. The game's atmosphere is meticulously crafted to ensure that players are constantly on edge, never knowing when Slenderman might appear.

The encounters with Slenderman are designed to be unpredictable and frightening. Players must navigate the environment, solve puzzles, and avoid the ever-present danger of Slenderman. The VR technology enhances these encounters, making them more intense as players feel the looming presence of Slenderman closing in on them.

Whether you're a fan of horror games or simply seeking an adrenaline-fueled adventure, our VR-based Slenderman game offers an unforgettable experience that will leave you breathless.

## **5.2 Future Scope**

### **5.2.1 Multi-Platform Support**

Insomnia is a game designed for integration with Meta's Oculus Quest VR headsets. The game control mechanics can be adjusted for other VR Headsets such as HTC Vive and Valve Index, using Unity's Open XR Plugin.

### **5.2.2 Innovative Interaction Mechanics**

The existing gameplay can be enhanced by creating more complex game logic and event trigger mechanisms within the C# script for various game objects.

### **5.2.3 Advanced Audiovisual Realism**

Improved Audiovisual elements in the game opens up wide possibilities for an immersive horror atmosphere. Unity's FMOD sound effects engine offers a platform SFX creation and manipulation.

## Bibliography

- [1] Gabajová, G., Krajčovič, M., Matys, M., Furmannová, B., Burganová, N. Designing Virtual Workplace Using Unity 3D Game Engine.2021
- [2] Cruz, C., Neira, N., Fernandez, M.,& Portales, C. Virtual Reality and Games,2018.
- [3] Shneiderman, B.; Plaisant, C.; Cohen, M.S.; Jacobs, S.; Elmquist, N.; Diakopoulos, N. Designing the User Interface: Strategies for Effective Human-Computer Interaction; Pearson: London, UK, 2016.
- [4] Sherman, W.R.; Craig, A.B. Understanding Virtual Reality: Interface, Application, and Design; Morgan Kaufmann Publishers Inc.: San Francisco, CA, USA, 2002
- [5] Conroy, R.A. Spatial Navigation in Immersive Virtual Environments. 2001.

## **Appendix A: Presentation**

---

# **VR HORROR GAME**

## **FINAL PRESENTATION**

Name Of the Guide :  
Mr.Paul Augustine

Team Members  
Alfred Antu  
Amith Kesav M  
Aravind Sivadas  
Ashin Sunny

---

---

## **Table Of Contents**

1. Introduction
  2. Problem Definition
  3. Objectives
  4. System Features
  5. Module Division
  6. Design Models
  7. UI Design
  8. Database Design
  9. Work Schedule
  10. Functional Requirements
  11. Software/Hardware Requirements
  12. Conclusion
  13. References
-

# Introduction

## Relevance



## Need

- Rapid growth of VR game development sectors offering highly immersive forms of entertainment
- Unity engine, C# and Blender constitute a powerful trio for creating immersive VR experiences, enabling developers to craft engaging worlds and gameplay mechanics.
- Our project focuses on developing a captivating VR game using Unity engine, C# and Blender.

# Problem Definition

Meeting the rising demand for innovative and immersive gaming experiences in the VR market.

Providing users with an escape from reality through richly detailed virtual environments and compelling gameplay mechanics.

Contributing to the advancement of VR technology and its integration into mainstream gaming culture.

# Objectives

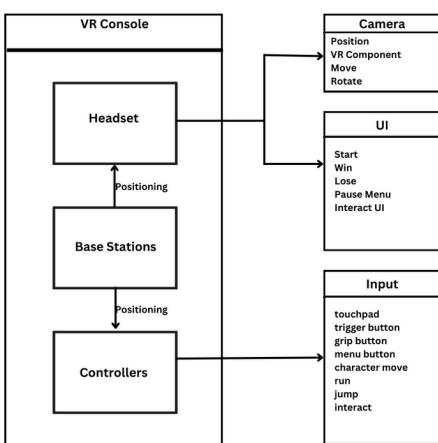
- **Intuitive Controls:** Implement intuitive and responsive controls tailored for VR, utilizing hand presence and gestures.
- **Immersive Visuals:** Utilize Blender for high-quality 3D models and textures to create visually stunning and terrifying environments.
- **Dynamic Lighting:** Use Unity's lighting system to create dynamic and realistic lighting that enhances the horror atmosphere.
- **Interactive Objects:** Implement interactive objects that players can manipulate or use to progress through the game.
- **Progressive Horror Experience:** Design the game to intensify the horror gradually, building tension throughout the gameplay.

# System Overview

The project aims to develop an immersive Virtual Reality (VR) based game that provides players with an engaging and interactive experience. Leveraging the latest VR technologies, the game will transport players to fantastical worlds where they can explore, interact with objects and characters, solve puzzles, and embark on thrilling adventures.

**Front End: VR Interface(Unity Engine)**

**Back End: Game Logic(C#), Asset Management(Unity and Blender), Data management**



---

# Module Division

## Player Movement and Interaction

This module handles player movement within the virtual environment and interaction with objects. It includes functionalities such as walking, running, grabbing objects, and interacting with interactive elements in the game world.

## Environmental Design and Asset Integration

This module focuses on designing the eerie and immersive environments of the horror game. It involves creating 3D models, textures, and animations for environmental assets and integrating them into the game engine.

---

---

# Module Division

## Game Logic and Event Triggering

The game logic module is responsible for implementing various gameplay mechanics and triggering events based on player actions or predefined triggers. It includes functionalities such as enemy AI behavior, scare events, puzzle mechanics, and level progression.

## Audio Design and Implementation

This module deals with the creation and implementation of audio assets to enhance the horror atmosphere of the game. It includes sound effects, ambient sounds, music, and voice acting. Spatial audio techniques are utilized to create an immersive auditory experience.

---

# Module Division

## User Interface and Menu Systems

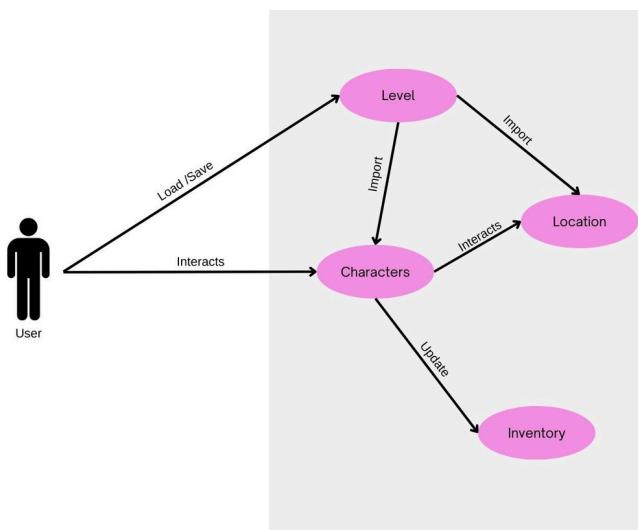
The user interface module focuses on designing and implementing the in-game HUD, menu systems, and user interaction interfaces. It includes functionalities such as health indicators, inventory management, settings menu, and in-game prompts.

## Performance Optimization and Testing

This module is responsible for optimizing the performance of the game to ensure smooth gameplay and immersive experience across different VR platforms. It involves techniques such as level optimization, texture compression, and performance profiling. Testing activities include debugging, playtesting, and performance testing.

---

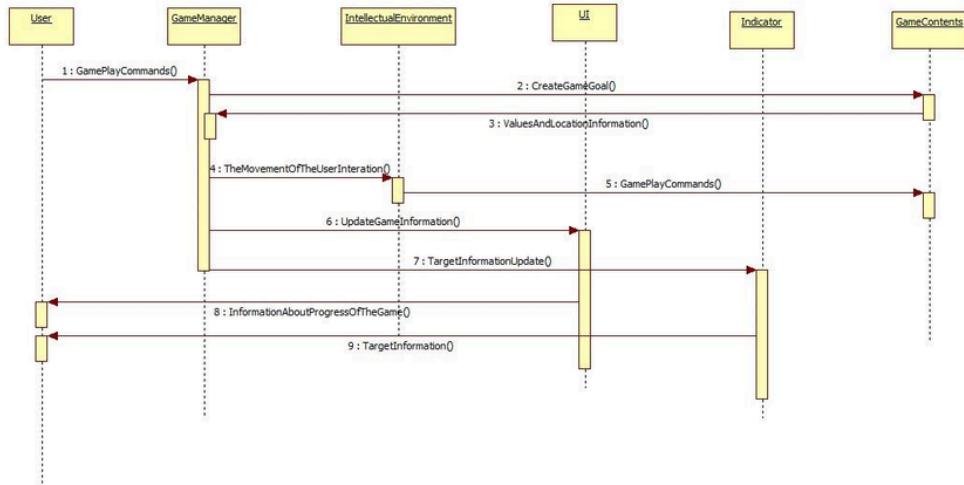
# Design Models



**Use Case Diagram**

---

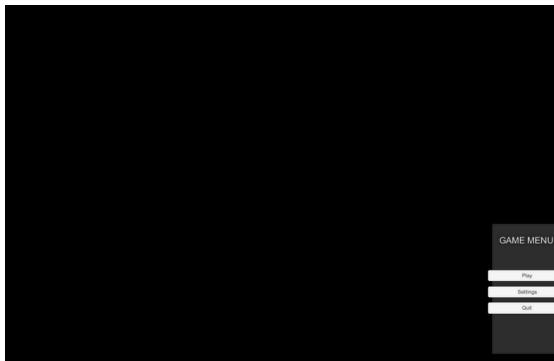
# Design Models



**Sequence Diagram**

---

# UI Design



**Main Menu**



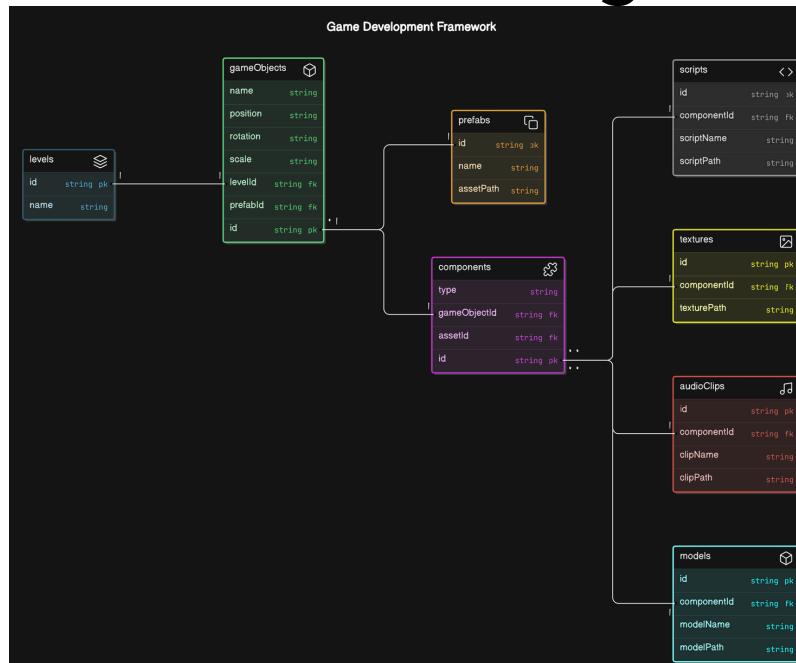
**Pause Menu**

# UI Design



End Menu

# Database Design



# Work Schedule



# Functional Requirements



## Player Controls

Define how the player will navigate and interact within the virtual environment using VR controllers or other input devices.

## Game Mechanics

Specify the core gameplay mechanics, such as combat, puzzles, exploration, or any unique features your game will have.

## User Interface (UI)

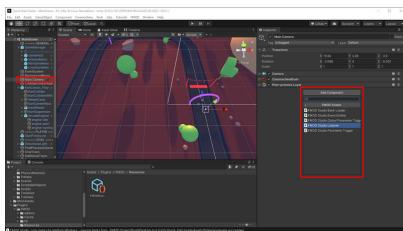
Describe the UI elements, including menus, HUD, and interactive UI components within the VR environment.



## Character Interaction

Determine how characters within the game will interact with the player and each other, including dialogue, gestures, and behaviors.

# Functional Requirements



## Audio

Specify sound effects, background music, and voiceovers to enhance immersion and gameplay experience using FMOD

## Performance Optimization

Ensure a smooth gameplay experience by optimizing performance, frame rate, asset loading, and memory usage.



## Physics

Define how physics will be implemented in the game, including collision detection, gravity, and object interactions.

## Non-Playable Characters (NPCs)

If your game includes NPCs or entities, describe their behaviors, decision-making processes, and interactions with the player

# Software & Hardware Requirements



## Software Requirements

- Unity 3D 2022.3 LTS
- Visual Studio or Visual Studio Code
- Blender
- SteamVR Plugin for Unity
- C# Programming Language

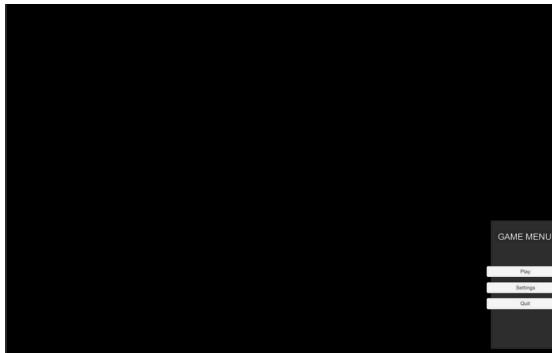
## Hardware Requirements

- CPU: Intel Core i5-4590/AMD FX 8350 equivalent or better.
- GPU: NVIDIA GTX 970 / AMD Radeon R9 290 equivalent or better.
- RAM: 8GB or more.
- Operating System: Windows 8.1, or Windows 10.
- Storage: At least 10GB of free space.
- VR Headset compatible with SteamVR

---

# Results

## Game Menus



Main Menu



Pause Menu

---

# Results

## Gameplay



House Exterior



House Interior

---

# Results

## Interactables



Batteries



Pages on Map

# Results

## Exit Cases



Mission Success



Mission Failed

---

# Conclusion

In conclusion, our VR-based Slenderman game promises an unparalleled horror experience that will captivate and terrify players. By leveraging cutting-edge virtual reality technology, we've crafted a game that immerses players in a world where every shadow holds a secret and every sound sends shivers down their spine.

With its stunning visuals, realistic environments, and spine-chilling encounters with the iconic Slenderman, our game pushes the boundaries of immersive entertainment. Whether you're a fan of horror games or simply seeking an adrenaline-fueled adventure, our VR-based Slenderman game offers an unforgettable experience that will leave you breathless.

---

---

# Future Enhancement

## Multi-Platform Support

Insomnia is a game designed for integration with Meta's Oculus Quest VR headsets. The game control mechanics can be adjusted for other VR Headsets such as HTC Vive and Valve Index, using Unity's Open XR Plugin.

## Innovative Interaction Mechanics

The existing gameplay can be enhanced by creating more complex game logic and event trigger mechanisms within the C# script for various game objects.

## Advanced Audiovisual Realism

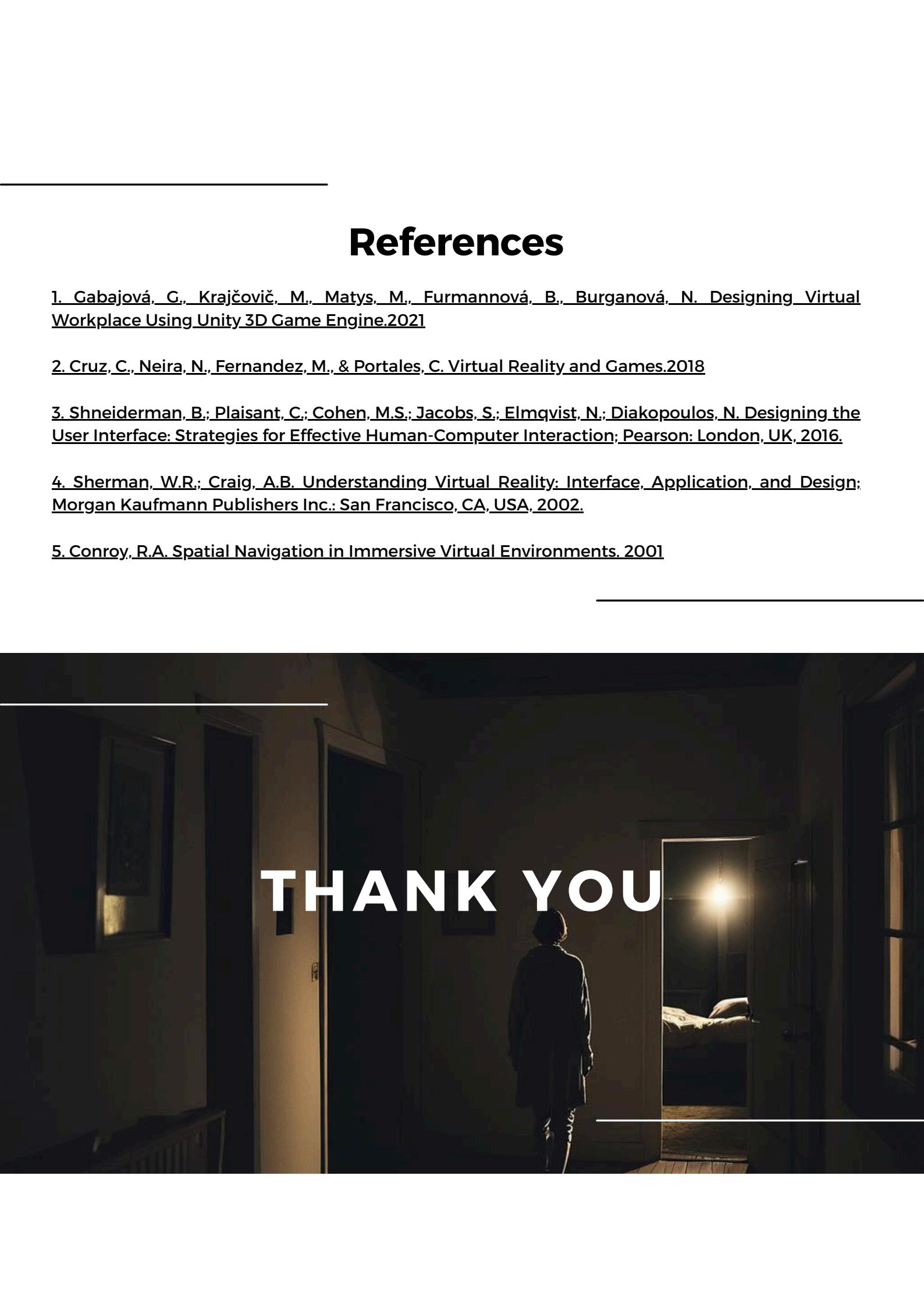
Improved Audiovisual elements in the game opens up wide possiblities for an immersive horror atmosphere. Unity's FMOD sound effects engine offers a plaform SFX creation and manipulation.

---

---

## References

1. Gabajová, G., Krajčovič, M., Matys, M., Furmannová, B., Burganová, N. Designing Virtual Workplace Using Unity 3D Game Engine.2021
  2. Cruz, C., Neira, N., Fernandez, M., & Portales, C. Virtual Reality and Games.2018
  3. Shneiderman, B.; Plaisant, C.; Cohen, M.S.; Jacobs, S.; Elmquist, N.; Diakopoulos, N. Designing the User Interface: Strategies for Effective Human-Computer Interaction; Pearson: London, UK, 2016.
  4. Sherman, W.R.; Craig, A.B. Understanding Virtual Reality: Interface, Application, and Design; Morgan Kaufmann Publishers Inc.: San Francisco, CA, USA, 2002.
  5. Conroy, R.A. Spatial Navigation in Immersive Virtual Environments. 2001
- 



THANK YOU

## **Appendix B: Vision, Mission, Programme Outcomes and Course Outcomes**

**DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING  
RAJAGIRI SCHOOL OF ENGINEERING & TECHNOLOGY (AUTONOMOUS)  
RAJAGIRI VALLEY, KAKKANAD, KOCHI, 682039  
(Affiliated to APJ Abdul Kalam Technological University)**



## **Vision, Mission, Programme Outcomes and Course Outcomes**

### **Institute Vision**

To evolve into a premier technological institution, moulding eminent professionals with creative minds, innovative ideas and sound practical skill, and to shape a future where technology works for the enrichment of mankind.

### **Institute Mission**

To impart state-of-the-art knowledge to individuals in various technological disciplines and to inculcate in them a high degree of social consciousness and human values, thereby enabling them to face the challenges of life with courage and conviction.

### **Department Vision**

To become a centre of excellence in Computer Science and Engineering, moulding professionals catering to the research and professional needs of national and international organizations.

### **Department Mission**

To inspire and nurture students, with up-to-date knowledge in Computer Science and Engineering, ethics, team spirit, leadership abilities, innovation and creativity to come out with solutions meeting societal needs.

## **Programme Outcomes (PO)**

Engineering Graduates will be able to:

- 1. Engineering Knowledge:** Apply the knowledge of mathematics, science, engineering fundamentals, and an engineering specialization to the solution of complex engineering problems.
- 2. Problem analysis:** Identify, formulate, review research literature, and analyze complex engineering problems reaching substantiated conclusions using first principles of mathematics, natural sciences, and engineering sciences.
- 3. Design/development of solutions:** Design solutions for complex engineering problems and design system components or processes that meet the specified needs with appropriate consideration for the public health and safety, and the cultural, societal, and environmental considerations.
- 4. Conduct investigations of complex problems:** Use research-based knowledge including design of experiments, analysis and interpretation of data, and synthesis of the information to provide valid conclusions.
- 5. Modern Tool Usage:** Create, select, and apply appropriate techniques, resources, and modern engineering and IT tools including prediction and modeling to complex engineering activities with an understanding of the limitations.
- 6. The engineer and society:** Apply reasoning informed by the contextual knowledge to assess societal, health, safety, legal, and cultural issues and the consequent responsibilities relevant to the professional engineering practice.
- 7. Environment and sustainability:** Understand the impact of the professional engineering solutions in societal and environmental contexts, and demonstrate the knowledge of, and need for sustainable development.
- 8. Ethics:** Apply ethical principles and commit to professional ethics and responsibilities and norms of the engineering practice.
- 9. Individual and Team work:** Function effectively as an individual, and as a member or leader in teams, and in multidisciplinary settings.

**10. Communication:** Communicate effectively with the engineering community and with society at large. Be able to comprehend and write effective reports documentation. Make effective presentations, and give and receive clear instructions.

**11. Project management and finance:** Demonstrate knowledge and understanding of engineering and management principles and apply these to one's own work, as a member and leader in a team. Manage projects in multidisciplinary environments.

**12. Life-long learning:** Recognize the need for, and have the preparation and ability to engage in independent and lifelong learning in the broadest context of technological change.

### **Programme Specific Outcomes (PSO)**

A graduate of the Computer Science and Engineering Program will demonstrate:

#### **PSO1: Computer Science Specific Skills**

The ability to identify, analyze and design solutions for complex engineering problems in multidisciplinary areas by understanding the core principles and concepts of computer science and thereby engage in national grand challenges.

#### **PSO2: Programming and Software Development Skills**

The ability to acquire programming efficiency by designing algorithms and applying standard practices in software project development to deliver quality software products meeting the demands of the industry.

#### **PSO3: Professional Skills**

The ability to apply the fundamentals of computer science in competitive research and to develop innovative products to meet the societal needs thereby evolving as an eminent researcher and entrepreneur.

### **Course Outcomes**

After the completion of the course the student will be able to:

#### **CO1:**

Identify technically and economically feasible problems (Cognitive Knowledge Level: Apply)

**CO2:**

Identify and survey the relevant literature for getting exposed to related solutions and get familiarized with software development processes (Cognitive Knowledge Level: Apply)

**CO3:**

Perform requirement analysis, identify design methodologies and develop adaptable & reusable solutions of minimal complexity by using modern tools & advanced programming techniques (Cognitive Knowledge Level: Apply)

**CO4:**

Prepare technical report and deliver presentation (Cognitive Knowledge Level: Apply)

**CO5:**

Apply engineering and management principles to achieve the goal of the project (Cognitive Knowledge Level: Apply)

## **Appendix C: CO-PO-PSO Mapping**

## COURSE OUTCOMES:

After completion of the course the student will be able to

<b>SL. NO</b>	<b>DESCRIPTION</b>	<b>Blooms' Taxonomy Level</b>
CO1	Identify technically and economically feasible problems (Cognitive Knowledge Level: Apply)	Level 3: Apply
CO2	Identify and survey the relevant literature for getting exposed to related solutions and get familiarized with software development processes (Cognitive Knowledge Level: Apply)	Level 3: Apply
CO3	Perform requirement analysis, identify design methodologies and develop adaptable & reusable solutions of minimal complexity by using modern tools & advanced programming techniques (Cognitive Knowledge Level: Apply)	Level 3: Apply
CO4	Prepare technical report and deliver presentation (Cognitive Knowledge Level: Apply)	Level 3: Apply
CO5	Apply engineering and management principles to achieve the goal of the project (Cognitive Knowledge Level: Apply)	Level 3: Apply

## CO-PO AND CO-PSO MAPPING

	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12	PSO 1	PSO 2	PS O3
C O1	3	3	3	3		2	2	3	2	2	2	3	2	2	2
C O2	3	3	3	3	3	2		3	2	3	2	3	2	2	2
C O3	3	3	3	3	3	2	2	3	2	2	2	3			2
C O4	2	3	2	2	2			3	3	3	2	3	2	2	2
C O5	3	3	3	2	2	2	2	3	2		2	3	2	2	2

3/2/1: high/medium/low

## JUSTIFICATIONS FOR CO-PO MAPPING

MAPPING	LOW/ MEDIUM/ HIGH	JUSTIFICATION
101003/CS6 22T.1-PO1	<b>HIGH</b>	Identify technically and economically feasible problems by applying the knowledge of mathematics, science, engineering fundamentals, and an engineering specialization to the solution of complex engineering problems.
101003/CS6 22T.1-PO2	<b>HIGH</b>	Identify technically and economically feasible problems by analysing complex engineering problems reaching substantiated conclusions using first principles of mathematics.
101003/CS6 22T.1-PO3	<b>HIGH</b>	Design solutions for complex engineering problems by identifying technically and economically feasible problems.
101003/CS6 22T.1-PO4	<b>HIGH</b>	Identify technically and economically feasible problems by analysis and interpretation of data.
101003/CS6 22T.1-PO6	<b>MEDIUM</b>	Responsibilities relevant to the professional engineering practice by identifying the problem.
101003/CS6 22T.1-PO7	<b>MEDIUM</b>	Identify technically and economically feasible problems by understanding the impact of the professional engineering solutions.
101003/CS6 22T.1-PO8	<b>HIGH</b>	Apply ethical principles and commit to professional ethics to identify technically and economically feasible problems.
101003/CS6 22T.1-PO9	<b>MEDIUM</b>	Identify technically and economically feasible problems by working as a team.
101003/CS6 22T.1-PO10	<b>MEDIUM</b>	Communicate effectively with the engineering community by identifying technically and economically feasible problems.
101003/CS6 22T.1-P011	<b>MEDIUM</b>	Demonstrate knowledge and understanding of engineering and management principles by selecting the technically and economically feasible problems.
101003/CS6 22T.1-PO12	<b>HIGH</b>	Identify technically and economically feasible problems for long term learning.
101003/CS6 22T.1-PSO1	<b>MEDIUM</b>	Ability to identify, analyze and design solutions to identify technically and economically feasible problems.
101003/CS6 22T.1-PSO2	<b>MEDIUM</b>	By designing algorithms and applying standard practices in software project development and Identifying technically and economically feasible problems.
101003/CS6 22T.1-PSO3	<b>MEDIUM</b>	Fundamentals of computer science in competitive research can be applied to Identify technically and economically feasible problems.
101003/CS6 22T.2-PO1	<b>HIGH</b>	Identify and survey the relevant by applying the knowledge of mathematics, science, engineering fundamentals.

101003/CS6 22T.2-PO2	<b>HIGH</b>	Identify, formulate, review research literature, and analyze complex engineering problems get familiarized with software development processes.
101003/CS6 22T.2-PO3	<b>HIGH</b>	Design solutions for complex engineering problems and design based on the relevant literature.
101003/CS6 22T.2-PO4	<b>HIGH</b>	Use research-based knowledge including design of experiments based on relevant literature.
101003/CS6 22T.2-PO5	<b>HIGH</b>	Identify and survey the relevant literature for getting exposed to related solutions and get familiarized with software development processes by using modern tools.
101003/CS6 22T.2-PO6	<b>MEDIUM</b>	Create, select, and apply appropriate techniques, resources, by identifying and surveying the relevant literature.
101003/CS6 22T.2-PO8	<b>HIGH</b>	Apply ethical principles and commit to professional ethics based on the relevant literature.
101003/CS6 22T.2-PO9	<b>MEDIUM</b>	Identify and survey the relevant literature as a team.
101003/CS6 22T.2-PO10	<b>HIGH</b>	Identify and survey the relevant literature for a good communication to the engineering fraternity.
101003/CS6 22T.2-PO11	<b>MEDIUM</b>	Identify and survey the relevant literature to demonstrate knowledge and understanding of engineering and management principles.
101003/CS6 22T.2-PO12	<b>HIGH</b>	Identify and survey the relevant literature for independent and lifelong learning.
101003/CS6 22T.2-PSO1	<b>MEDIUM</b>	Design solutions for complex engineering problems by Identifying and survey the relevant literature.
101003/CS6 22T.2-PSO2	<b>MEDIUM</b>	Identify and survey the relevant literature for acquiring programming efficiency by designing algorithms and applying standard practices.
101003/CS6 22T.2-PSO3	<b>MEDIUM</b>	Identify and survey the relevant literature to apply the fundamentals of computer science in competitive research.
101003/CS6 22T.3-PO1	<b>HIGH</b>	Perform requirement analysis, identify design methodologies by using modern tools & advanced programming techniques and by applying the knowledge of mathematics, science, engineering fundamentals.
101003/CS6 22T.3-PO2	<b>HIGH</b>	Identify, formulate, review research literature for requirement analysis, identify design methodologies and develop adaptable & reusable solutions.

101003/CS6 22T.3-PO3	<b>HIGH</b>	Design solutions for complex engineering problems and perform requirement analysis, identify design methodologies.
101003/CS6 22T.3-PO4	<b>HIGH</b>	Use research-based knowledge including design of experiments, analysis and interpretation of data, and synthesis of the information to provide valid conclusions.
101003/CS6 22T.3-PO5	<b>HIGH</b>	Create, select, and apply appropriate techniques, resources, and modern engineering and IT tools.
101003/CS6 22T.3-PO6	<b>MEDIUM</b>	Perform requirement analysis, identify design methodologies and assess societal, health, safety, legal, and cultural issues.
101003/CS6 22T.3-PO7	<b>MEDIUM</b>	Understand the impact of the professional engineering solutions in societal and environmental contexts and Perform requirement analysis, identify design methodologies and develop adaptable & reusable solutions.
101003/CS6 22T.3-PO8	<b>HIGH</b>	Perform requirement analysis, identify design methodologies and develop adaptable & reusable solutions by applying ethical principles and commit to professional ethics.
101003/CS6 22T.3-PO9	<b>MEDIUM</b>	Function effectively as an individual, and as a member or leader in teams, and in multidisciplinary settings.
101003/CS6 22T.3-PO10	<b>MEDIUM</b>	Communicate effectively with the engineering community and with society at large to perform requirement analysis, identify design methodologies.
101003/CS6 22T.3-PO11	<b>MEDIUM</b>	Demonstrate knowledge and understanding of engineering requirement analysis by identifying design methodologies.
101003/CS6 22T.3-PO12	<b>HIGH</b>	Recognize the need for, and have the preparation and ability to engage in independent and lifelong learning in the broadest context of technological change by analysis, identify design methodologies and develop adaptable & reusable solutions.
101003/CS6 22T.3-PSO3	<b>MEDIUM</b>	The ability to apply the fundamentals of computer science in competitive research and prior to that perform requirement analysis, identify design methodologies.
101003/CS6 22T.4-PO1	<b>MEDIUM</b>	Prepare technical report and deliver presentation by applying the knowledge of mathematics, science, engineering fundamentals, and an engineering specialization to the solution of complex engineering problems.
101003/CS6 22T.4-PO2	<b>HIGH</b>	Identify, formulate, review research literature, and analyze complex engineering problems by preparing technical report and deliver presentation.

101003/CS6 22T.4-PO3	<b>MEDIUM</b>	Prepare Design solutions for complex engineering problems and create technical report and deliver presentation.
101003/CS6 22T.4-PO4	<b>MEDIUM</b>	Use research-based knowledge including design of experiments, analysis and interpretation of data, and synthesis of the information to provide valid conclusions and prepare technical report and deliver presentation.
101003/CS6 22T.4-PO5	<b>MEDIUM</b>	Create, select, and apply appropriate techniques, resources, and modern engineering and IT tools and Prepare technical report and deliver presentation.
101003/CS6 22T.4-PO8	<b>HIGH</b>	Prepare technical report and deliver presentation by applying ethical principles and commit to professional ethics and responsibilities and norms of the engineering practice.
101003/CS6 22T.4-PO9	<b>HIGH</b>	Prepare technical report and deliver presentation effectively as an individual, and as a member or leader in teams, and in multidisciplinary settings.
101003/CS6 22T.4-PO10	<b>HIGH</b>	Communicate effectively with the engineering community and with society at large by prepare technical report and deliver presentation.
101003/CS6 22T.4-PO11	<b>MEDIUM</b>	Demonstrate knowledge and understanding of engineering and management principles and apply these to one's own work by prepare technical report and deliver presentation.
101003/CS6 22T.4-PO12	<b>HIGH</b>	Recognize the need for, and have the preparation and ability to engage in independent and lifelong learning in the broadest context of technological change by prepare technical report and deliver presentation.
101003/CS6 22T.4-PSO1	<b>MEDIUM</b>	Prepare a technical report and deliver presentation to identify, analyze and design solutions for complex engineering problems in multidisciplinary areas.
101003/CS6 22T.4-PSO2	<b>MEDIUM</b>	To acquire programming efficiency by designing algorithms and applying standard practices in software project development and to prepare technical report and deliver presentation.
101003/CS6 22T.4-PSO3	<b>MEDIUM</b>	To apply the fundamentals of computer science in competitive research and to develop innovative products to meet the societal needs by preparing technical report and deliver presentation.
101003/CS6 22T.5-PO1	<b>HIGH</b>	Apply the knowledge of mathematics, science, engineering fundamentals, and an engineering specialization to the solution of complex engineering problems.
101003/CS6 22T.5-PO2	<b>HIGH</b>	Identify, formulate, review research literature, and analyze complex engineering problems by applying engineering and management principles to achieve the goal of the project.

101003/CS6 22T.5-PO3	<b>HIGH</b>	Apply engineering and management principles to achieve the goal of the project and to design solutions for complex engineering problems and design system components or processes that meet the specified needs.
101003/CS6 22T.5-PO4	<b>MEDIUM</b>	Apply engineering and management principles to achieve the goal of the project and use research-based knowledge including design of experiments, analysis and interpretation of data, and synthesis of the information to provide valid conclusions.
101003/CS6 22T.5-PO5	<b>MEDIUM</b>	Create, select, and apply appropriate techniques, resources, and modern engineering and IT tools and to apply engineering and management principles to achieve the goal of the project.
101003/CS6 22T.5-PO6	<b>MEDIUM</b>	Apply reasoning informed by the contextual knowledge to assess societal, health, safety, legal, and cultural issues and the consequent responsibilities by applying engineering and management principles to achieve the goal of the project.
101003/CS6 22T.5-PO7	<b>MEDIUM</b>	Understand the impact of the professional engineering solutions in societal and environmental contexts, and apply engineering and management principles to achieve the goal of the project.
101003/CS6 22T.5-PO8	<b>HIGH</b>	Apply ethical principles and commit to professional ethics and responsibilities and norms of the engineering practice and to use the engineering and management principles to achieve the goal of the project.
101003/CS6 22T.5-PO9	<b>MEDIUM</b>	Function effectively as an individual, and as a member or leader in teams, and in multidisciplinary settings and to apply engineering and management principles to achieve the goal of the project.
101003/CS6 22T.5-PO11	<b>MEDIUM</b>	Demonstrate knowledge and understanding of engineering and management principles and apply these to one's own work, as a member and leader in a team. Manage projects in multidisciplinary environments and to apply engineering and management principles to achieve the goal of the project.
101003/CS6 22T.5-PO12	<b>HIGH</b>	Recognize the need for, and have the preparation and ability to engage in independent and lifelong learning in the broadest context of technological change and to apply engineering and management principles to achieve the goal of the project.
101003/CS6 22T.5-PSO1	<b>MEDIUM</b>	The ability to identify, analyze and design solutions for complex engineering problems in multidisciplinary areas. Apply engineering and management principles to achieve the goal of the project.

101003/CS6 22T.5-PSO2	<b>MEDIUM</b>	The ability to acquire programming efficiency by designing algorithms and applying standard practices in software project development to deliver quality software products meeting the demands of the industry and to apply engineering and management principles to achieve the goal of the project.
101003/CS6 22T.5-PSO3	<b>MEDIUM</b>	The ability to apply the fundamentals of computer science in competitive research and to develop innovative products to meet the societal needs thereby evolving as an eminent researcher and entrepreneur and apply engineering and management principles to achieve the goal of the project.