VR HORROR GAME

Prof. Punam Bagul
Guide
KCCEMSR, Thane East
punam.bagul@kccemsr.edu.in

Ritvik Babre
Student
KCCEMSR, Thane East
ritvikbabre@kccemsr.edu.in

Hitesh Behera
Student
KCCEMSR, Thane East
hiteshbehera@kccemsr.edu.in

Shruti Sabbani
Student
KCCEMSR, Thane East
shrutisabbani@kccemsr.edu.in

Swapnil Yadav
Student
KCCEMSR, Thane East
swapnilyadav@kccemsr.edu.in

ABSTRACT

In this study, we proposed an analysis framework that examines participants' immersive experience processes from three dimensions: (1) the state of enjoyment, (2) the transformation of the awareness of time, and (3) a sense of spatial integration. We found that the balance of skills and challenges is threshold factor affecting participants' overall enjoyment state and experience of transforming the time awareness during a VR gaming process.

With cutting-edge VR technology, you will be transported to a place where your every sense is assaulted by the unknown, and your survival instincts are pushed to the limit. Confront your deepest fears, solve intricate puzzles, and face grotesque monstrosities as you journey deeper into the darkness.

General Terms

Virtual Reality Horror Gaming Experience

Keywords :

virtual reality, VR, horror game, immersive experience, first-person perspective, android, gyroscope, immersive, jump scares, atmospheric environment, narrative storytelling, motion controllers, hand-tracking, google-cardboard, unity game engine, HMD.

INTRODUCTION

Virtual reality (VR) horror games offer an unparalleled level of immersion by leveraging first-person perspective, realistic graphics, and interactive gameplay. Players experience the game world through the eyes of the protagonist, enhancing the feeling of presence and intensifying the horror experience. Detailed environments, coupled with atmospheric lighting, create a sense of dread, while interactivity allows players to manipulate objects and solve puzzles. Jump scares are effectively utilized to startle players, taking advantage of the heightened sense of immersion in VR. Compelling narratives further engage players, drawing them deeper into the terrifying world of VR horror gaming.

Aim:

To promote virtual reality (VR) in India, a multi-faceted approach is essential. This involves raising awareness through targeted campaigns, fostering collaboration between government, industry, and educational institutions, and providing support to startups and developers. Encouraging the creation of high-quality VR content, integrating VR into education and training programs, and investing in infrastructure are also crucial. By establishing a conducive ecosystem, India can harness the potential of VR to drive innovation, economic growth, and social development, while positioning itself as a leader in immersive technology on the global stage.

Importance of VR:

Virtual reality (VR) is of paramount importance due to its ability to offer immersive and interactive experiences that transcend traditional boundaries. It holds significant potential across various domains, including education, training, healthcare, entertainment, and beyond. VR enables users to explore virtual environments, practice real-life scenarios, and engage with content in ways that are not possible through conventional mediums. By providing realistic simulations, VR enhances learning outcomes, improves skill retention, and fosters creativity and innovation. Moreover, VR has therapeutic applications, aiding in rehabilitation, pain management, and mental health treatments. In entertainment, VR offers unparalleled immersion, allowing users to immerse themselves in virtual worlds and experiences. Overall, the importance of VR lies in its capacity to revolutionize how we learn, work, play, and connect with others, shaping the future of technology and human interaction.

Why are we using Horror Game to promote VR ?

Using VR horror games to promote VR in India can be a strategic approach due to several factors. Firstly, horror games in VR offer an unparalleled level of immersion and

engagement, evoking strong emotional responses from players. This heightened sense of fear and suspense effectively showcases the capabilities of VR technology, allowing users to experience a thrilling and realistic environment unlike any other medium. Additionally, horror games have a universal appeal and a strong following in Indian culture, tapping into the country's fascination with ghost stories, folklore, and supernatural themes. By capitalizing on this cultural interest, VR horror games can attract attention and generate excitement among Indian audiences, particularly younger demographics who are avid consumers of horror content. Moreover, the use of VR horror games provides a unique selling point for VR technology, offering experiences that are exclusive to the platform and differentiating it from traditional forms of entertainment. Ultimately, by leveraging the immersive and emotional power of horror games in VR, promoters can effectively showcase the potential of VR technology and drive adoption and interest among Indian consumers.

How We are using Animations:

In a VR horror game implemented on Unity, incorporating animation adds a crucial layer of immersion and tension to the gameplay experience. This involves creating or obtaining animated character models that are rigged and animated with various actions such as walking, running, and attacking. Unity's animation tools and scripting capabilities enable developers to program animation states and transitions, triggering animations based on player interactions and game events. Additionally, blend trees and layering techniques are employed to seamlessly blend between different animations, ensuring smooth and responsive character movements. Furthermore, animation is not limited to characters alone; it extends to environmental animations and visual effects, such as eerie lighting, atmospheric particle effects, and dynamic object interactions. By carefully crafting and integrating anime into the VR horror game, developers can evoke a heightened sense of fear and suspense, immersing players in a truly terrifying virtual world where every animation contributes to the overall sense of dread and anticipation.

Hardware and Software Requirements:

Table: Hardware and Software Requirements

Hardware	**Software**
Android device	Character Animations - Mixamo
Gyroscope	Asset Store - Unity
PlayStation Controller	Scripting Language – C#
Procus 1 (VR Headset)	Game Engine - Unity

IMPLEMENTATION STEPS:

The proposed the steps used in developing our VR horror game:

1. **Initialization**: Set up game engine, VR hardware, and load assets.
2. **Main Game Loop**: Continuously update game world and player interactions.
3. **Player Interaction**: Track head movements, implement movement controls, and enable object interaction.
4. **Game Environment**: Design eerie settings, implement lighting and sound effects, and trigger horror events.
5. **AI Entities**: Initiate AI-controlled entities with behaviour patterns and interactions.
6. **Game Logic**: Introduce puzzles, define conditions for progression, and control game flow.
7. **Audio Management**: Play background music, trigger audio cues, and use 3D audio.
8. **Inventory System**: Design item management system for players.
9. **Save and Load**: Implement save system for player progress.
10. **User Interface**: Create VR-friendly UI elements.
11. **End Game**: Handle game completion options.
12. **Cleanup**: Release resources and close connections.

SYSTEM FLOWCHART:

FEATURES:

GAME MANAGER:

In this game, the Game Manager serves as the central controller for managing game state, user interface, resource handling, input, events, scene transitions, game logic, and potentially save/load functionality. It orchestrates these elements to ensure a cohesive and enjoyable player experience.

It manages states of the game; includes Win Condition, Lose Condition and Restart

This defines Game Manager class in Unity, responsible for managing game state transitions and UI elements. It includes methods to display game over and game won screens, as well as restarting the game. The Awake method ensures there's only one GameManager instance using the Singleton pattern. The Restart method reloads the current scene to restart the game. Overall, it's a basic implementation of a central game management system.

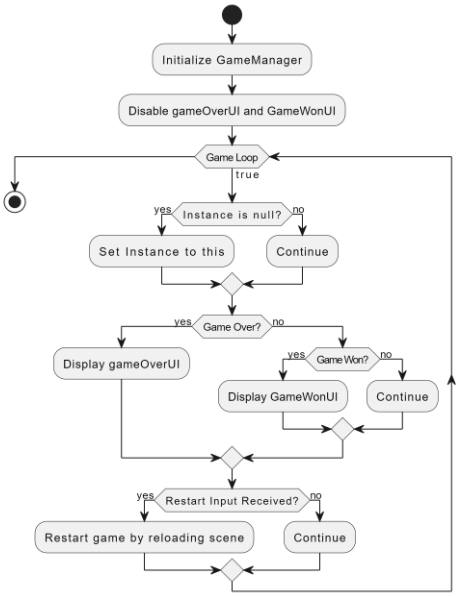

Fig: Game Manager Flowchart

HEAD TRACKING:

Head tracking in a VR horror game on Android devices involves using the device's built-in sensors, such as gyroscopes and accelerometers, to track the player's head movements. Unity's Android VR integration allows developers to access this tracking data and update the virtual camera's position and orientation accordingly. This ensures that the player's view in the game world aligns with their real-world head movements, enhancing immersion and fear-inducing experiences in virtual reality.

This enables head tracking using the gyroscope sensor on an Android device within a Unity environment. Upon starting, it creates a new GameObject called "Camera Container" to hold the main camera, positioning it at the same location as the script's GameObject. It then sets the script's GameObject as a child of the "Camera Container" to ensure that any rotations applied to the container also affect the camera's orientation.

The script checks if the device supports the gyroscope. If supported, it enables the gyroscope and adjusts the initial rotation of the "Camera Container" to align it correctly with the device's orientation. It then continuously updates the local rotation of the script's GameObject based on the gyroscope's attitude, simulating the player's head movements. Additionally, it ensures that the "Camera Container" follows any positional changes in the virtual environment by updating its position to match that of the CameraSlot Transform.

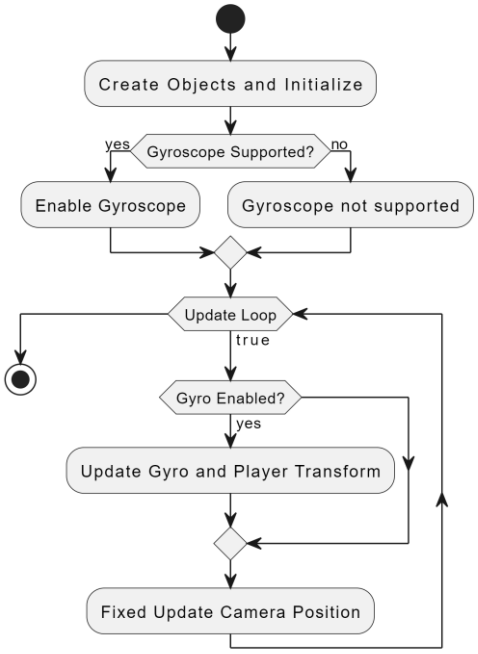

Fig: Gyroscopic head tracking flowchart

ITEM PICKUP:

In this game, the item pickup mechanism utilizes raycasting to allow players to interact with objects within the virtual environment. When players' cameras, typically representing their viewpoint in VR, look at an object, a raycast is projected from the camera's position in the direction it's facing. If the raycast intersects with an interactable object, indicating that the player is looking at it, they can trigger a pickup action. This action might involve pressing a button on their VR controller to pick up the object. Implementing item pickup via raycasting provides a natural and intuitive way for players to interact with objects in the game world, enhancing immersion and gameplay in the VR horror experience.

In a VR horror game developed in Unity, item pickup mechanics involve players using VR controllers to interact with objects in the virtual environment. Players can typically grab items by pressing buttons on their controllers, and once picked up, they can manipulate and use these items to progress through the game. Visual and auditory cues provide feedback to players when they successfully pick up an item, enhancing immersion. Item pickup mechanics often play a crucial role in solving

puzzles, unlocking doors, and uncovering secrets, adding depth to the gameplay and driving the narrative forward.

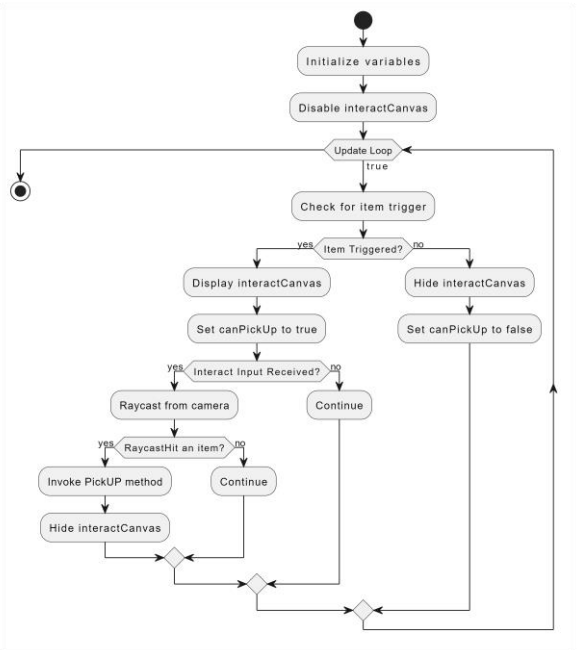

Fig: Mechanism to Pick Up items

ITEM SPAWNER:

In this game, implementing an Item Spawner is crucial for enhancing player immersion and progression. The flowchart begins with initialization, setting up variables and parameters related to item spawning. As the game starts, triggers activate the spawning process based on player progress and game state. The spawner evaluates conditions such as player proximity and visibility to determine when and where to spawn items. Once triggered, the spawner selects items and places them strategically within the game world, incorporating animations and audio cues to alert players. Players can then interact with these items using their VR controllers, aiding progression by solving puzzles or unlocking new areas. Throughout the game, the spawner continually evaluates player actions and adjusts spawning accordingly, ensuring a dynamic and immersive experience. By following this flowchart, developers can create a compelling VR horror game where item spawns contribute to the atmosphere, challenge, and narrative progression, keeping players engaged and immersed in the experience.

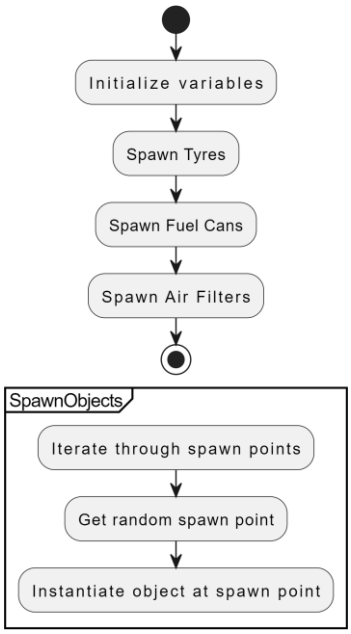

Fig: Item Spawner Flowchart

AI ENEMY:

The enemy movement in a VR horror game is governed by several states that dictate the behavior of the enemy AI. These states include "Idle," "Patrol," "Chase," and "Attack." During the "Idle" state, the enemy remains stationary or follows predefined patrol routes within the game world. Upon detecting the player within the "Chase" range, the enemy transitions into the "Chase" state, where it actively pursues the player's last known position. In the "Patrol" state, the enemy roams the environment according to a set patrol radius, periodically reducing its patrol area over time. If the player enters the enemy's "Close Range" zone, the enemy switches to the "Attack" state, triggering an attack animation to grab the player. The script utilizes a NavMeshAgent component to navigate the enemy AI, allowing it to move smoothly within the game environment. Through these states, the "EnemyController" script creates dynamic and immersive encounters, enhancing the player's sense of tension and fear in the VR horror experience.

In a VR horror game developed in Unity, managing enemy movement involves defining various states that dictate how enemies behave and interact with the player. These states include idle, patrol, alerted, chase, attack, and return to patrol. During the idle state, enemies remain stationary or follow predetermined patrol routes, establishing a baseline behavior when they are not actively engaged with the player. When alerted to the player's presence, enemies transition into an alerted state, prompting them to investigate or search for the player. Upon detecting the player, enemies enter a chase state, where they actively pursue the player, increasing tension and urgency. In the attack state, enemies engage the player with offensive actions such as melee or ranged attacks, posing a direct threat. Once the player evades detection or escapes combat, enemies may return to their patrol routes or idle behavior, creating dynamic and unpredictable encounters. Implementing these enemy states in a VR horror game adds depth to gameplay, enhancing immersion and providing players with challenging and suspenseful experiences.

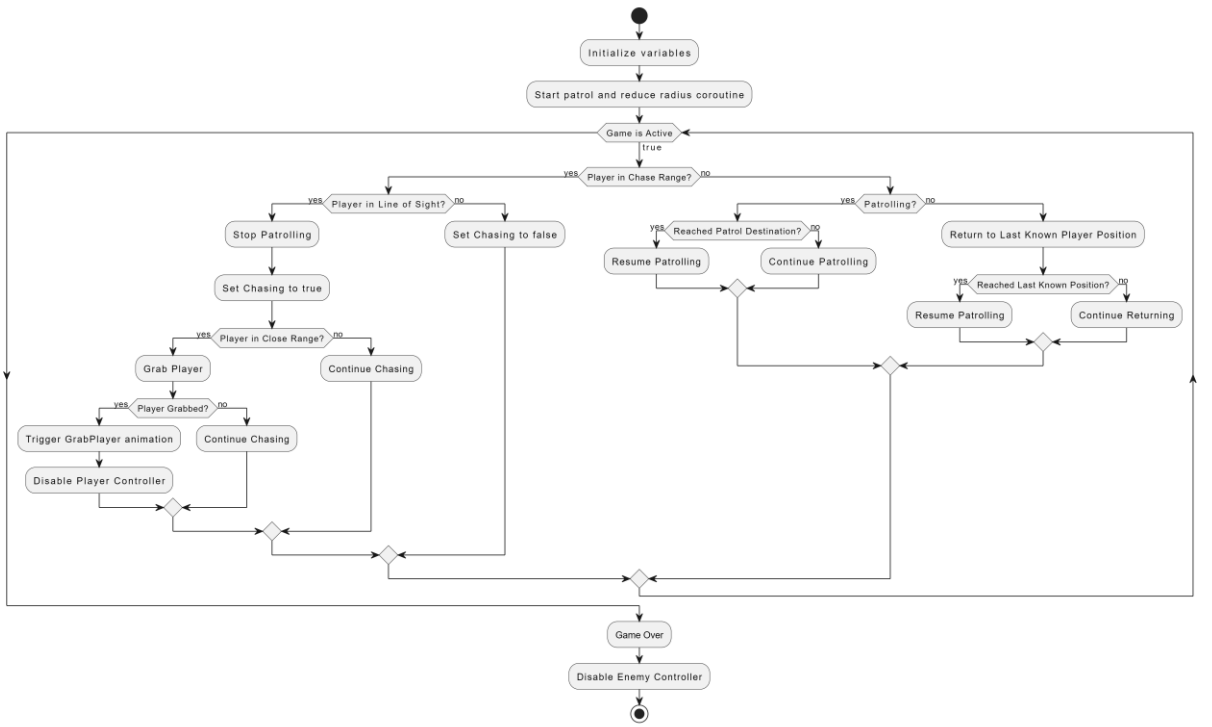

Fig : Enemy movement flowchart

In a VR horror game, implementing enemy collider functionality is essential for enabling interactions between the player and enemy entities. The enemy collider implementation involves attaching colliders to the enemy AI's game objects, typically encompassing their physical boundaries. These colliders are configured to detect collisions with other game objects, including the player character's collider. When the player enters the detection range of the enemy collider, it triggers specific actions or state transitions within the enemy AI's behavior. For example, when the player comes into contact with the enemy collider, it may initiate a chase sequence, causing the enemy to pursue the player. Additionally, the enemy collider can detect when the player is within attack range, prompting the enemy to perform attack animations or actions. Through careful tuning of collider properties such as size, shape, and detection range, developers can create immersive and responsive enemy interactions that contribute to the tension and suspense of the VR horror experience. Moreover, efficient collision detection algorithms and optimizations are crucial to maintaining smooth gameplay performance, especially in dynamic and fast-paced scenarios typical of VR horror games.

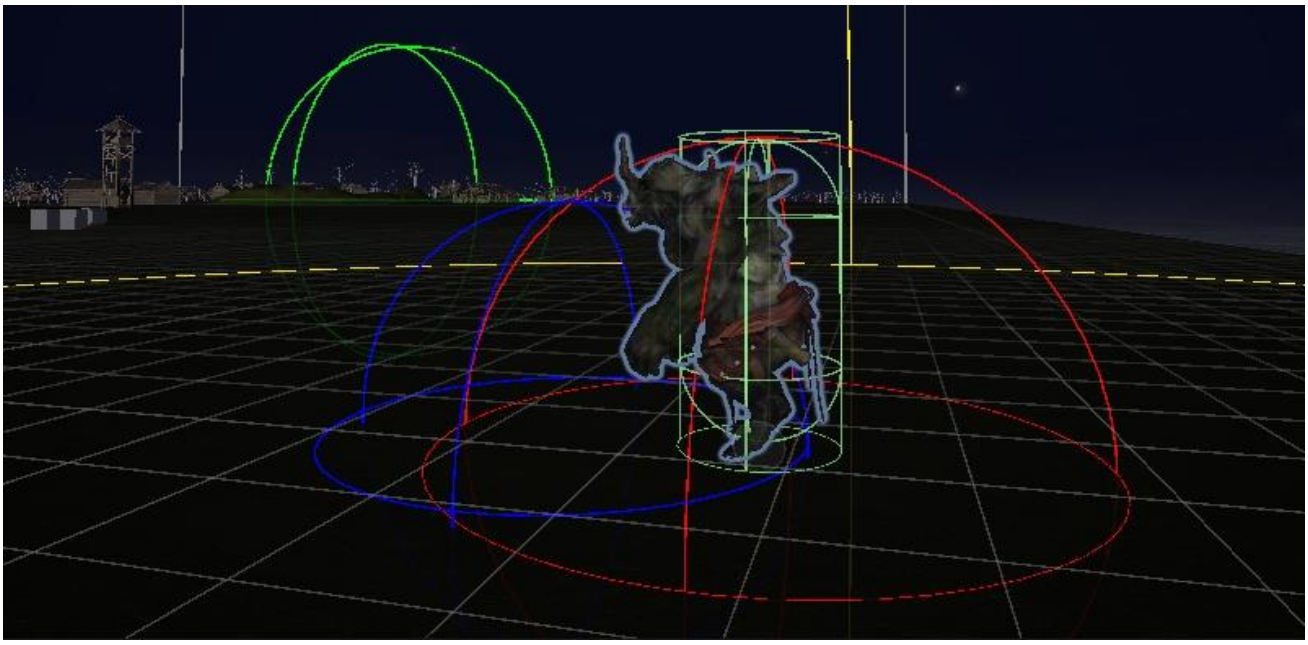

Fig: Enemy Collider Implementation

ARCHITECTURE:

The architecture of a VR horror game developed in ***Unity*** encompasses a sophisticated integration of various components to deliver a spine-chilling and immersive experience. At its core lies the game engine, Unity, providing a comprehensive development environment for crafting the virtual world and implementing gameplay mechanics. This engine serves as the foundation for integrating essential VR-specific features such as head tracking, motion controllers, rendering, and audio.

The ***VR headset*** acts as the player's gateway to the virtual horror experience, housing display screens, lenses, and sensors to track head movements accurately. Unity interfaces with the headset's SDK to access head tracking data, enabling real-time updates to the in-game camera perspective based on the player's movements. This seamless integration ensures that players can explore the eerie environments from immersive first-person viewpoints.

Motion controllers are pivotal for enabling player interaction within the virtual horror world. Equipped with buttons, triggers, and motion sensors, these controllers facilitate intuitive gestures and movements, allowing players to manipulate objects, interact with the environment, and navigate treacherous scenarios. Unity's support for motion controller integration enables developers to implement immersive and responsive interactions that heighten the horror experience.

Rendering plays a critical role in bringing the terrifying world to life, with Unity's rendering pipeline delivering high-quality graphics, lighting, and visual effects tailored for VR environments. From eerie shadows and atmospheric lighting to intricate textures and immersive environments, Unity's rendering capabilities enable developers to create chilling and visually captivating horror scenes.

Audio serves as the backbone of immersion, enhancing the atmosphere and tension of the VR horror experience. Unity's audio engine allows developers to integrate spatial audio, ambient sounds, and unsettling sound effects that dynamically respond to player actions and environmental cues. This spatial audio simulation ensures that sounds emanate realistically from specific directions, intensifying the sense of dread and suspense.

Through the collaborative integration of VR headset functionality, game engine features, head tracking, motion controllers, rendering, and audio, developers can craft a truly terrifying VR horror game that plunges players into a chilling and unforgettable nightmare. This architectural synergy ensures that every element of the experience works harmoniously to immerse players in a world of terror and suspense.

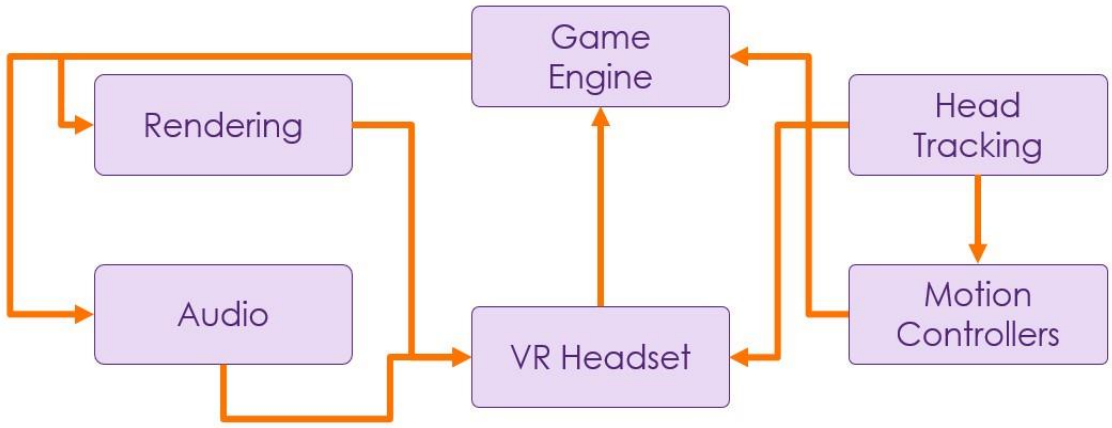

Fig: Data Flow between Components

3. CONCLUTION:

In conclusion, developing an immersive first-person view (FPV) virtual reality (VR) horror game is both challenging and rewarding. This genre has the potential to offer players a spine-tingling and unforgettable experience. Attention to every detail, from the chilling narrative and atmospheric environments to realistic character models and haunting audio, is crucial. By harnessing the capabilities of game engines, VR headsets, rendering techniques, and motion controllers, developers can craft a world where players feel fully immersed and utterly vulnerable. The success of a VR horror game hinges on its ability to terrify and captivate players, compelling them to explore, solve puzzles, and survive in a nightmarish realm. Iterative testing, user feedback, and continuous improvement are essential throughout the development process. Ultimately, a successful VR horror game creates an experience that lingers long after the headset is removed, leaving players with a sense of true immersion and the haunting thrill of a virtual world well-executed. Whether you're a seasoned horror enthusiast or a

newcomer to the genre, our VR Horror Game promises an unforgettable journey into the heart of fear.

4. ACKNOWLEDGMENTS

We would like to express special thanks of gratitude to our guide & co-guide *Prof. Punam Bagul* as well as our Project Coordinator *Dr. Kiran Bhandari* who gave us the golden opportunity to do this wonderful project on the topic of *IMMERSIVE VR HORROR GAME*, which also helped us in doing a lot of research and we came to know about so many new things. We are very grateful to our Head of the Department *Prof Amarja Adgaonkar* for extending her help directly and indirectly through various channels in our project work. We would also like to thank I/C *Principal Dr. Vilas N Nitnaware* for providing us the opportunity to implement our project. We are thankful to them.

5. REFERENCES

[1] Ntokos, Konstantinos. "Level of fear": Analysis of fear spectrum into a tool to support horror game design for immersion and fear." An International Journal (CGDEIJ) 1, no. 33-43 (2018).

[2] Årnell, Tobias, and Nikola Stojanovic. "Horror game design–what instills fear in the player?: A study on the effects of horror game design theories and level design patterns on player behaviour in a horror environment." (2020).

[3] Ziwen. "Analysis of the design aesthetics and player emotions of horror games: Take 'Little Nightmares' as a case." (2022).

[4] de Lima, Edirlei Soares, Bruno MC Silva, and Gabriel Teixeira Galam. "Adaptive virtual reality horror games based on Machine learning and player modeling." Entertainment Computing 43 (2022): 100515.

[5] Shaikh, Omar, Yilu Sun, and Andrea Stevenson Won. "Movement visualizer for networked virtual reality platforms." *2018 IEEE Conference on Virtual Reality and 3D User Interfaces (VR)*. IEEE, 2018.

[6] Lin, Jih-Hsuan Tammy, Dai-Yun Wu, and Chen-Chao Tao. "So scary, yet so fun: The role of self-efficacy in enjoyment of a virtual reality horror game." New Media & Society 20.9 (2018): 3223-3242.

[7] Lin, Tammy Jin-Hsuan. "Virtual Reality Horror Games and Fear in Gaming." *Oxford Research Encyclopedia of Communication*. 2023.

[8] Lin, Tammy Jin-Hsuan. "Virtual Reality Horror Games and Fear in Gaming." *Oxford Research Encyclopedia of Communication*. 2023.

[9] Bian, Shijie. "Research on the Application of VR in Games." *Highlights in Science, Engineering and Technology* 39 (2023): 389-394.

[10] Zhang, Ruiqi. "Research on the Progress of VR in Game." *Highlights in Science, Engineering and Technology* 39 (2023): 103-110.