# Exploring Third-Person Action Game Development with Melee Combat Using Unreal Engine 5

Nagarjuna Suyaji Mote
210431715
Pengwei Hao
Computer Games

***Abstract*—**The need for exciting and immersive gaming experiences has increased due to the rapid growth of digital entertainment. This dissertation explores third-person action game development, focusing on integrating melee combat mechanics within Unreal Engine 5. This research aimed to develop an engaging and visually appealing game that provides players with an exciting linear-level design and dynamic combat. The project explored gameplay mechanics, game systems, animation, and environmental design using Unreal Engine 5's sophisticated features. The methodologies employed encompassed a blend of iterative design, asset selection, and scripting within Unreal Engine's Blueprints system. Intricate melee combat interactions, fluid character animations, and carefully constructed level environments can all be seen in the final game. Notably, the responsiveness and choice of the combat system enhance the player experience, and the linear level design leads players through a well-paced narrative.

***Keywords—game development, third-person action, melee combat***

## I. Introduction

The demand for fascinating and immersive gameplay experiences has increased in the ever-changing world of video game design. Third-person action games, where players participate in visceral combat and traverse through elaborately planned worlds, are one popular genre still evolving. Incorporating melee combat mechanics within this genre adds a layer of physicality and strategy, enriching the player's engagement and immersion.

This dissertation explores third-person action game development, emphasising the incorporation of melee combat systems in Unreal Engine 5. The selection of Unreal Engine 5 serves as a crucial platform for showing the potential of cutting-edge game engines in creating aesthetically appealing and immersive gameplay in an era distinguished by extraordinary technological developments.

The primary objective of this study is to show the successful amalgamation of intriguing melee combat mechanics with the broad range of capabilities of unreal engine 5. This project aims to give players and game developers insights into the intricate process of creating immersive gaming experiences by meticulously designing and developing a third-person action game that emphasises fluid combat animations, responsive controls, and strategically designed linear levels. This study was motivated by the rising need for captivating, innovative game mechanics that push the frontiers of interactive entertainment. This project aims to add to the ongoing conversation about game design and technology by harnessing the power of Unreal Engine 5.

The subsequent chapters of this dissertation discuss the methodology used, the intricacies of game development, the difficulties encountered, and the results obtained. Readers will have a better knowledge of the creative and technical factors that support the development of a third-person action game with melee combat features utilising Unreal Engine 5 after reading this thorough investigation.

## II. Background

A new era of fascinating and immersive gameplay experiences has emerged due to the continual combination of technology innovation and creative design in the field of video game creation. The context for the investigation of third-person action game production with melee combat using Unreal Engine 5 is provided in this section, which covers essential aspects, including user perspective, game frameworks, scripting, and engine capabilities.

### *A. Player perspective*

In digital games, player perspective significantly impacts immersion and engagement. Research by Denisova and Cairns (2015) delves into the dynamics between first-person and third-person perspectives, highlighting how player preferences influence the sense of immersion and embodiment [3]. A first-person view allows the player to be more immersed in the world as the main character, whereas A third-person view allows players to observe the main character in action. Our project's decision to use a third-person perspective attempts to provide a well-balanced mix of player agency and participation.

### *B. Engine Choice*

Frameworks and game engines are crucial in determining how a game is developed and what it can do. The landscape of tools available for the development of serious games is illuminated by Cowan and Kapralos thorough study of frameworks and game engines [2]. This survey informs our decision to leverage the capabilities of Unreal Engine 5, a robust and versatile platform known for its graphical prowess and interactive potential.

With its real-time rendering methods, dynamic lighting systems, and cutting-edge graphics capabilities, Unreal Engine 5 is hailed as a technological marvel. The pedagogical relevance of Unreal Engine in game development education is highlighted by research by Dickson et al. (2017), attesting to its significance in influencing the learning experiences of aspiring developers [9]. Some examples of games using Unreal Engine 5 are Fortnite, Layers of Fear, Lords of the Fallen and Remnant 2.

### *C. Blueprint Scripting*

Unreal Engine 5's blueprint scripting is a powerful tool that enables a visual and natural method of game development.

According to Sewell (2015), Blueprints offer a potent way to create and alter gameplay components without the requirement for advanced coding knowledge. By allowing designers and artists to actively participate in the creation of mechanics, this methodology democratises game development [4]. A flexible visual scripting interface with a broad range of features is provided by blueprints. Blueprint nodes and functions can be used to coordinate game mechanics, character behaviours, environment interactions, and even sophisticated AI-driven systems. In order to achieve the intended user experience, developers can quickly prototype ideas, iterate on gameplay concepts, and polish mechanisms [4][1].

Combat mechanics that are compelling and interactive are made possible via blueprint scripting. According to Milam and El Nasr, design patterns direct player movement and activities in 3D games. These patterns can be converted into actual gameplay components using blueprints, such as fluid fighting animations, sophisticated combos, and adaptable opponent behaviour. The immersive potential of the game is increased by this integration, which increases player agency [1].

### *D. Liner Level Design*

Linear level design is a deliberate method of level building that leads players down a set path while gradually revealing story beats and gameplay difficulties. This design philosophy can be especially effective in telling a gripping story, keeping the action moving, and offering a carefully crafted player experience. Because game levels are linearly organised, developers can create a compelling narrative arc. To deliberately advance the plot, game designers might carefully insert puzzles, battle situations, and gameplay mechanisms. Our research uses linear-level design to forge a seamless player experience that heightens immersion and encourages participation [4].

The seamless integration of narrative components into gameplay is made easier by linear levels. Players experience story-driven events, character interactions, and environmental narratives as they advance through the level. Furthermore, the precise pacing of gameplay experiences is made possible by linear-level design. A dynamic and exciting player experience is made possible by interspersing moments of tension, exploration, and conflict within precisely designed portions. This strategy promotes players' emotional investment in the difficulties and events as they develop, which improves the overall gaming experience [4][5].

The introduction of properly crafted gameplay difficulties that progressively get harder on a linear level's canvas is excellent. Sewell (2015) talks about the value of giving players a sense of growth and accomplishment as they go through stages. Linear-level design can inspire a sense of accomplishment and reward players for their skill development by deftly organising combat encounters, platforming sequences, and puzzles [4].

## III. METHODOLOGY

This section describes the methodical process of creating a third-person action game with melee combat elements utilising Unreal Engine 5. Each element of the project contributed to the development of a streamlined and engaging gameplay experience. This section goes into detail about how Blueprint scripting was used to coordinate the development process, character movement, combat system, statistics system, enemy design, enemy AI, and linear level design.

### *A. Character Movement*

The project commenced with designing and implementing the main character's movement system. The character movement system was methodically designed to provide the right blend of fluidity, responsiveness, and player agency. Blueprint scripting enabled the creation of intricate animation blend spaces that seamlessly transitioned between various movement states, such as walking, running, and jumping. By utilising animation blending techniques, the game world could be navigated more easily, improving the visual quality of the character's movements. Additionally, responsive controls that directly translated player inputs to on-screen actions were implemented using Blueprint scripting. The character's movement was made to feel natural and intuitive by using computations for acceleration, deceleration, and momentum [4], which increased the player's sense of agency and immersion.

### *B. Melee Combat System*

The gameplay experience's focal point was the combat system, which was created to convey a sense of physical involvement and mastery. The orchestration of intricate combat animations that seamlessly transitioned between assaults and dodges was made possible via blueprint programming. Players were able to manoeuvre through combat situations with ease while executing precise strikes because of the integration of these animations with character movement.

Combo mechanics implementation was also made easier by blueprint scripting [1]. Different attacks that each had their unique timing and impact could be chained together by players, creating intricate battle scenes that rewarded skilful execution. A lock-on system is also implemented, allowing players to focus on one enemy and making aiming the attacks easier. The synchronisation of animations, sound effects, and haptic feedback enhanced the visceral nature of combat interactions, creating a satisfying and immersive gameplay loop.

A takedown system is also employed using blueprint interfaces which allow players to take down and kill enemies that implement this interface in one hit from certain angles and positions.

### *C. Character Statistics System*

A statistics system was created to enhance player progression and engagement. Implementing blood which acts as the player’s health, was made easier using blueprint scripting. This enabled managing player traits and creating a sense of character development during the games.

### *D. Enemies and Enemy AI*

The project went on to include the construction of enemy characters, each with unique traits and fighting styles. The development and integration of enemy animations, attack patterns, and healing systems were made possible through blueprint programming. Additionally, using Blueprint

scripting, artificial intelligence (AI) was used to give opponents sophisticated behaviours so they could dynamically interact with the player during combat.

### *E. Levels*

Level design and the choice of linear progression were envisioned as key components in creating a unified and immersive player experience. The ability to create stages that fluidly directed players through a narrative-rich journey while fostering engagement and immersion was a key component in making this vision a reality. Linear levels where high-intensity combat sections were linked using low-intensity areas, giving players breathing space between intense combat rooms.

The level design included the creation of meticulously planned linear progressions using Blueprint scripting. Dynamic narrative events, human interactions, and environmental storytelling aspects were choreographed. Blueprint scripting enabled careful pacing management, ensuring that players faced important plot moments and gameplay obstacles in a predetermined order.

The presentation of immersive sequences, which are triggered at specified moments utilising triggers to heighten the narrative impact, was made easier by the integration of Blueprint scripting with level design. This strategy successfully included players and told the story as it developed, emphasising the link between player agency and the character's journey.

To offer a closely linked narrative experience, a linear progression was purposefully chosen. Linear levels allowed for the precise presentation of character development, plot twists, and emotional moments, creating a cohesive and impactful storytelling arc. Without being distracted by non-linear exploration, this method allowed the player to fully immerse themselves in the character's adventure [12].

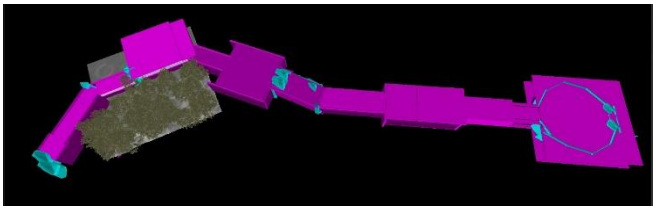

*Figure 1 Example Level Layout*

### *F. Testing and Iterative refinement*

Iterative refinement characterised the development period, with Blueprint scripting enabling quick iterations, playtesting, and fine-tuning. This iterative method meant that the final product resonated with players, offering the desired immersive and engaging experience.

## IV. Results and Evaluation

We examine the results of the game creation project in the "Results and Evaluation" section, recognising both the successes and the difficulties that were faced. This part seeks to give a thorough overview of the finished product while highlighting its strengths in the midst of its limitations and deviations.

The game's current version has several noteworthy achievements. Players can interact with the game environment and participate in combat encounters thanks to the successful integration of the game's fundamental features, such as character movement and melee combat. The linear level designs provide a structurally complete journey, guiding players through the narrative arc.


*Figure 2 In-Game Example Scene*

### *A. Implemented Core Mechanics*

The project's successful implementation of fundamental gameplay elements, such as responsive character movement and useful combat interactions, is noteworthy. Players can interact with the game world and enjoy the action-packed scenes despite the difficulties.

### *B. Deviation from Initial Design*

However, it is crucial to recognise that due to time restrictions, several aspects that were envisioned at the conceptualisation phase were not fully realised in the finished product. Although directional dodge was initially intended, a more straightforward backward dodge function was substituted. This aberration harmed the level of player tactics during combat encounters. Even while the directional dodge was not included in the simpler dodge technique, it still gives players a chance to avoid enemy assaults tactically. This gameplay element motivates players to modify their combat tactics, which makes for more exciting conflicts.

### *C. Incomplete Asset Integration*

The visual aspects of the game were not fully realised due to a lack of assets. While the levels are structurally complete, they lack the detailed decorations and environmental elements that were envisioned in the initial design. This departure from the desired visual ambience may affect both the level of overall immersion and the coherence of the narrative.

### *D. Limited Enemy Variance*

Due to resource constraints, the expected variety in enemy models was limited. As a result, the game only used a limited number of enemy model types. This potentially affects player involvement because different foes frequently add to the gameplay experience and present tactical difficulties.

### *E. Level Implementation and Bug Challenges*

Time constraints also impacted the number of implemented levels. While the game contains a subset of levels, the entire narrative progression may feel abbreviated due to the fewer stages. A few bugs and technical issues were identified during the development process, which could have an impact on the consistency of gameplay and player satisfaction.

### F. Impact on Player Experience

The changes made from the original design and the difficulties encountered have an effect on the overall player experience. The simplified dodge mechanic modifies the nature of combat encounters, potentially influencing how deeply strategic choices are considered. The lack of diverse enemy models and a lack of completed level decorations may affect the player's immersion and the game's overall aesthetic appeal.

### G. Future Directions and Lessons Learned

The difficulties encountered during development serve as a reminder of the value of effective time management, the accessibility of resources, and the necessity of contingency planning. Future updates could focus on adding the missing functionality, broadening the variety of enemies, and polishing the game's visual style and mechanics. The knowledge obtained from working on this project can be used to balance scope and execution within practical limits.

## V. Conclusion

To sum up, the game development endeavour has proven to be a metamorphic expedition that has illuminated the intricate dance between imaginative vision, technical realisation, and player interaction. Despite minor digressions from the original plan, the successful integration of central mechanics highlights the potential of game development tools, particularly Blueprint scripting within Unreal Engine 5. As discussed in the "Results" section, the difficulties and restrictions that were experienced were not only setbacks but also opportunities for development. Technical difficulties and bugs teach us important lessons about the value of performance improvement and quality assurance. The lack of some functionality and the modifications to the initial concept highlight the necessity of strategic time and resource management in subsequent projects.

## VI. Future Work

This section of the dissertation looks at how the problems and potential areas for improvement that were discovered in the "Results and Conclusion" section can serve as catalysts for further development and improvement. It is possible to improve the gameplay experience by confronting these difficulties head-on and building on the lessons discovered.

Future work can concentrate on implementing the initially intended directional dodge mechanic, building on the difficulties described in the "Results" section. The fighting system can be expanded, and additional combat manoeuvres can be included so that players can engage foes in a more planned and intricate battle. Not only would this improve the gameplay, but it would also give the user more control over the outcome of battles.

Also, future work to improve the game's visual appeal has the potential to take advantage of the restrictions on asset availability and the unfinished level decorations, as stated in the "Results" section. The game's immersion can be considerably increased by purchasing a greater choice of items and devoting efforts to level decoration. Players could experience a world that is more engaging and immersive if it has detailed textures, realistic environmental impacts, improved UI elements and eye-catching visual components.

Narrative extension and the addition of new levels can be used to fill up the narrative gaps mentioned in the "Results" section. Future work may involve developing the plot and examining the characters' backstories. Cutscenes can be added to stories to enhance them.


## Acknowledgement

I would like to express my heartfelt gratitude to my supervisor Dr. Pengwei Hao, for his invaluable guidance and unwavering support throughout the course of this project. Dr. Hao's encouragement to explore innovative ideas allowed me to unleash my creativity and embrace unconventional approaches to game development.


## References


[1] Milam, D. and El Nasr, M.S., 2010, July. Design patterns to guide player movement in 3D games. In Proceedings of the 5th ACM SIGGRAPH Symposium on Video Games (pp. 37-42).

[2] Cowan, B., & Kapralos, B. (2019). A Survey of Frameworks and Game Engines for Serious Game Development. Serious Games and Edutainment Applications, 213-236

[3] ] Denisova, A. and Cairns, P., 2015, April. First person vs. third person perspective in digital games: do player preferences affect immersion?. In Proceedings of the 33rd annual ACM conference on human factors in computing systems (pp. 145-148).

[4] Sewell, B., 2015. Blueprints visual scripting for unreal engine. Packt Publishing Ltd.

[5] Nakamura, R., Lago, L.L., Carneiro, A.B., Cunha, A.J., Ortega, F.J., Bernardes Jr, J.L. and Tori, R., 2010, July. 3PI experiment: Immersion in third-person view. In Proceedings of the 5th ACM SIGGRAPH Symposium on Video Games (pp. 43-48).

[6] Valcasara, N., 2015. Unreal engine game development blueprints. Packt Publishing Ltd.

[7] Black, D., 2017. Why can I see my avatar? Embodied visual engagement in the third-person video game. Games and Culture, 12(2), pp.179-199.

[8] Schramm, J., 2013. Analysis of Third Person Cameras in Current Generation Action Games.

[9] Dickson, P.E., Block, J.E., Echevarria, G.N. and Keenan, K.C., 2017, June. An experience-based comparison of unity and unreal for a stand-alone 3D game development course. In Proceedings of the 2017 ACM Conference on Innovation and Technology in Computer Science Education (pp. 70-75).

[10] ] Dormans, J., 2010, June. Adventures in level design: generating missions and spaces for action adventure games. In Proceedings of the 2010 workshop on procedural content generation in games (pp. 1-8).

[11] Jenkins, H., 2004. Game design as narrative architecture. Computer, 44(3), pp.118-130.

[12] Huynh, E., Nyhout, A., Ganea, P. and Chevalier, F., 2020. Designing narrative-focused role-playing games for visualization literacy in young children. IEEE Transactions on Visualization and Computer Graphics, 27(2), pp.924-934.