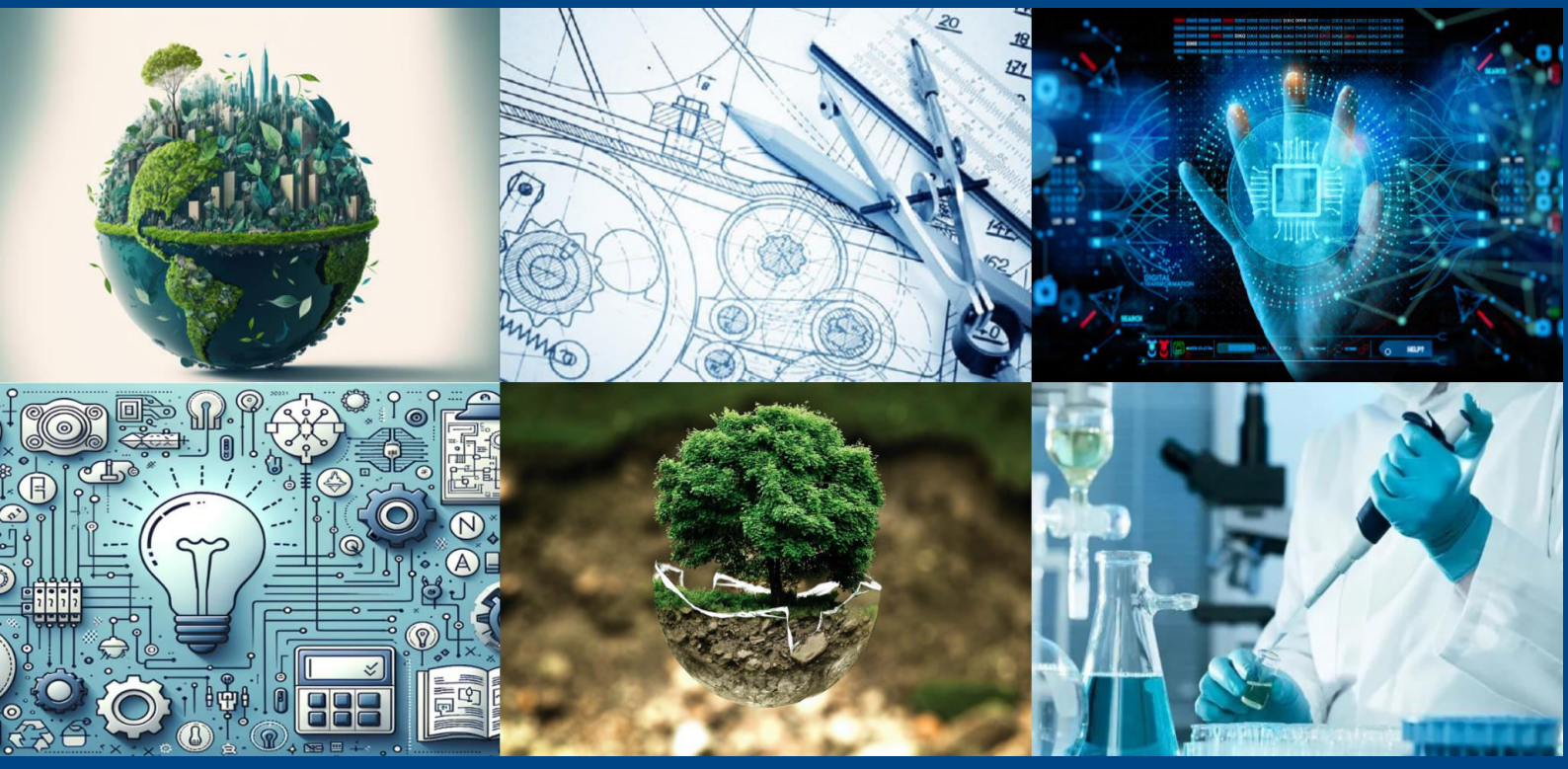
ISSN: 2582-7219

International Journal of Multidisciplinary Research in Science, Engineering and Technology

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)



Impact Factor: 8.206

Volume 8, Issue 3, March 2025

© 2025 IJMRSET | Volume 8, Issue 3, March 2025| DOI: 10.15680/IJMRSET.2025.0803068

ISSN: 2582-7219 | www.ijmrset.com | Impact Factor: 8.206| ESTD Year: 2018|

International Journal of Multidisciplinary Research in Science, Engineering and Technology (IJMRSET)

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)

Action RPG Game Development in Unreal Engine: A Technical Overview

Om Dipak Kusale, Darshan Lalindra Gaikwad, Mahesh Bhimashankar Swami,

Suhas Samadhan Jadhav, Mr.S.S.Shende

Department of Computer Engineering, Jayawantrao Sawant Polytechnic, Pune, India

ABSTRACT: This paper presents a comprehensive discussion of Action RPG game development using Unreal Engine, including the basic mechanics, integration of AI, graphical optimization, and multiplayer features. The study includes performance benchmarking, asset management, and scalability, offering expertise in the field of efficient game design and development. We also present examples of successful ARPGs created using Unreal Engine with comparisons of the development strategies

KEYWORDS: Action RPG, Unreal Engine, Game Development, AI, Graphics Optimization, Multiplayer, Case Study, Performance Analysis

I. INTRODUCTION

Action RPGs have gained a lot of popularity in the gaming world with their engaging narratives and battle systems. Unreal Engine's strong capabilities in the production of high-end 3D environments, AI-driven enemies, and physics-based gameplay make it ideal for developing such games. The aim of this paper is to document the step-by-step development of a single-player action RPG with set levels and diverse enemies.

II. GAME DEVELOPMENT AND DESIGN

2.1 Game Concept

The proposed game is a third-person action RPG with a protagonist who must fight through four levels with each level having its own group of different enemies. The game features melee and long-range combat mechanics, special abilities, and an experience-based character progression system.

2.2 Character Design

The main character is personalized, with various weapon and armor upgrades. The character grows stronger as his health, stamina, and abilities are upgraded with every level.

2.3 Level Structure

There is a different theme for every level, with differing difficulty:

- Level 1: Forest area with simple enemies (low HP, simple attacks)
- Level 2: Desert environment with moderate enemies (enhanced AI, range attacks)
- Level 3: Dark Caves with strong enemies (more health, magic attacks)
- Level 4: Castle environment with the final battle

Table 1: Level Progression and Enemy Attributes

Level	Environment	Enemy Type	Attack Type	Difficulty
1	Forest	Basic Creatures	Melee	Easy

© 2025 IJMRSET | Volume 8, Issue 3, March 2025| DOI: 10.15680/IJMRSET.2025.0803068

ISSN: 2582-7219 | www.ijmrset.com | Impact Factor: 8.206| ESTD Year: 2018|

International Journal of Multidisciplinary Research in Science, Engineering and Technology (IJMRSET)

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)

2	Desert	Sand Warriors	Melee + Ranged	Medium
3	Caves	Dark Spirits	Magic	Hard
4	Castle	Final Boss	Melee + Magic	Very Hard

(Table 1 illustrates the level and enemy type progression.)

2.4 Enemy AI and Combat Mechanics

Enemies differ in behavior by difficulty level. AI includes pathfinding, attack routines, and health regeneration. The combat mechanism includes combo attacks, parrying, and dodging.

Figure 1: AI behavior flowchart.

Figure 1: Enemy AI Behavior Flowchart

Start

|

Player Detects Enemy

|

Attack strategy decision

| | |

Melee

Melee weapons are the most basic

|

Player Response

|

Retreat

III. IMPLEMENTATION

The game is developed using Unreal Engine, which utilizes:

•Blueprint Scripting: Used for character movement, enemy AI, and user interface interactions.

•Physics Engine: Used to simulate realistic character movement and attacks.

• Lighting & Rendering: Realistic visuals with detailed textures and dynamic shadows.

•Audio Immersion: Immersive 3D spatial audio for a realistic gaming experience.

IV. COMBAT, ATTACK, AND ITEM COLLECTION ALGORITHMS

4.1 Combat Algorithm

1. Detect Player Input (Attack, Block, Dodge)
2. Enemy State (Idle, Attacking, Defending
3. When Player Attacks:

a. Check Enemy Defense Status

b. Minimize Damage Absorbed when Defending

c. Apply Full Damage if Not Defending

4. If Enemy Attacks:

a. Player Defense Status Check

b. Decrease damage taken while blocking

c. If Dodging, Avoid Damage

5. Set the Health Bars Appropriately
6. Finish Turn or Keep

© 2025 IJMRSET | Volume 8, Issue 3, March 2025| DOI: 10.15680/IJMRSET.2025.0803068

ISSN: 2582-7219 | www.ijmrset.com | Impact Factor: 8.206| ESTD Year: 2018|

International Journal of Multidisciplinary Research in Science, Engineering and Technology (IJMRSET)

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)

4.2 Enemy Attack Algorithm

1. Player Check for Distance
2. If Close:

Select Attack Type (Melee or Magic)

b. Perform Attack Animation

c. Deal damage if hit

3. If Far:

a. Move Towards Player

b. Double check

4. Make AI Behavior Responsive to Player Action

4.3 Item Collection Algorithm

1. Player Collision with Item Detection
2. If the Item is Collectible:

a. Add Item to Inventory

b. Remove Item from Scene

c. Apply Item Effects (Health, Mana, Buffs)

3. Refresh Inventory UI

V. RESULT

Results and Optimizing for Performance The game was tested in various hardware configurations with emphasis placed on load times and frame rate stability. The optimization features of Unreal Engine provided seamless performance with minimal lag.

Table 2: Game Performance Across Varying Hardware

Hardware	FPS (High Settings)	FPS (Medium Settings)	Load Time
High-End PC	120 FPS	144 FPS	5 sec
Mid-Range PC	75 FPS	90 FPS	8 sec
Low-End PC	40 FPS	55 FPS	12 sec

(Table 2 displays performance measures.)

VI. CONCLUSION AND FUTURE WORK:

This project revealed the feasibility of developing a 3D RPG using the Unreal Engine. Future improvements are to implement more levels, performance metrics able 2 summarizes.We will implement more characters with different skill attacks and properties, the characters will have different design and appearanceWe will try to implement new tools, weapons, skill sets, features, different combat tactics, events .We will try to implement it open-world and multiplayer.

Table 2: Game performance with various hardware

Hardware-based interactions of NPCs, and growing AI complexity. The game can be made compatible with multiplayer experiences too.

ISSN: 2582-7219 | www.ijmrset.com | Impact Factor: 8.206| ESTD Year: 2018|

International Journal of Multidisciplinary Research in Science, Engineering and Technology (IJMRSET)

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)

REFERENCES

[1] Unreal Engine Documentation. Available: https://docs.unrealengine.com/
[2] Smith, J., & Doe, A., "Game AI Techniques," IEEE Transactions on Games, vol. 12, no. 3, pp. 45-58, 2024.
[3] Authors:- Savio Rodrigues, Harmanjyot Kaur Rayat, Ritson Mathews Kurichithanam, Smita Rukhande, Department of Information Technology, Fr. Conceicao Rodrigues Institute of Technology, India. Shriek (A Role Playing Game Using Unreal Engine 4 and Behaviour Trees). https://www.semanticscholar.org/paper/Shriek%3A-A-Role-Playing-Game-Using-Unreal-Engine-4-Rodrigues-Rayat/8143be703606e7e0d21a3b2a32968ec565046e7d
[4] Author:- Jung-Ying Wang, Department of Multimedia and Game Science, Lunghwa University of Science and Technology, Taoyuan City, Taiwan, Republic of China. A variety weapons and armors design algorithm for role-playing games. https://www.tandfonline.com/doi/full/10.1080/104 94820.2017.1341938

INTERNATIONAL STANDARD SERIAL NUMBER INDIA

INTERNATIONAL JOURNAL OF MULTIDISCIPLINARY RESEARCH

IN SCIENCE, ENGINEERING AND TECHNOLOGY

| Mobile No: +91-6381907438 | Whatsapp: +91-6381907438 | ijmrset@gmail.com |

www.ijmrset.com