High School Student Institute of Mathematics and Informatics

Summer Research School

Symposium

2022

NETPUNK

AN UNREAL ADVENTURE

Author:
Genadi Mladenov Kolev
NHSMS “Academic Luibomir Chakalov”
Sofia
genadimkolev@gmail.com

Scientific Advisors:
Ivan Dimitrov
ivan.miroslavov4@gmail.com
&
Tsvetoslav Mavrodiev

Table of Contents

Abstract

With the ever-increasing progress in game engines and tools, entry into A, AA or even AAA game-making is becoming more available to small indie teams and even sole developers. The aim of this project is to leverage the advantages of the newly released Unreal Engine 5 to create an adventurous multiplayer experience.

Introduction

The project aims to build a 3D Multiplayer high-fidelity game suited for all audiences alike. The initial goal is to lay the foundation for the project, specifically – basic mechanics for characters and host/join multiplayer and then set up more complex systems and build the game from there.

Unreal Engine compared to other engines

The gaming industry uses numerous engines to create various games of all genres, suited for the needs of audiences worldwide and market needs. Unfortunately, most of them are in-house exclusive to their respective studios, leaving the rest of the developers with little options. Despite this, developers have access to three powerful engines still.

GODOT

Excelling for 2D games, GODOT is great at making small performant games quickly. On top of that it is perhaps the most beginner friendly of the three. However, GODOT fails to meet expectations when making 3D games and high-end game development, making it unsuitable for this project.

Unity

Aimed towards smaller teams of developers and mainly beginners, Unity is one of the two major free to use game engines. However, when making games with emphasis on realistic graphics Unreal is able to deliver better visuals, especially with its newest iteration.

Why Unreal

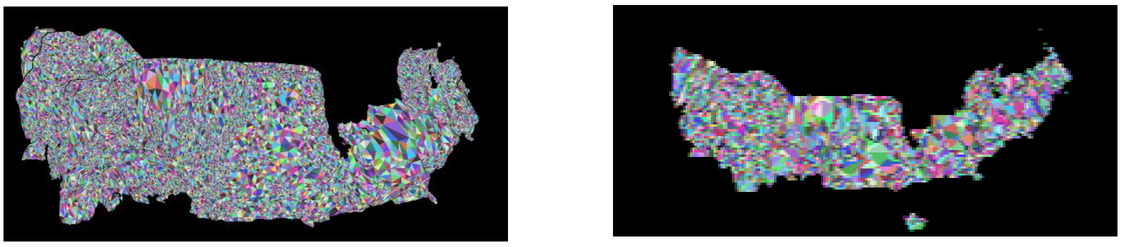
Unreal Engine can deliver large, beautiful, and highly performant games at the cost of development complexity. The engine excels at making high-fidelity 3D games and, being made with teams in mind, is best utilized by big studios worldwide. A compelling reason for picking Unreal is because of its major graphical and light improvements, introduced with its lates version.

- Lumen

Lumen is Unreal Engine 5's fully dynamic global illumination and reflections system that is designed for next-generation consoles, and it is the default global illumination and reflections system. It renders diffuse interreflection with infinite bounces and indirect specular reflections in large, detailed environments. Its biggest advantage is the fact that it reduces the process of baking lighting into the map from mandatory and only option to means for optimisation, saving tremendous amount of development time and resources.

- Nanite

Nanite is Unreal Engine 5's virtualized geometry system which uses a new internal mesh format and rendering technology to render pixel scale detail and high object counts. It intelligently does work on only the detail that can be perceived and no more, removing the need for manually created LODs (Level of Detail), which in turn again reduces development time significantly. Nanite's data format is also highly compressed, and supports fine-grained streaming with automatic level of detail as well as practically any mesh.


A set of Nanite meshes from up close and afar

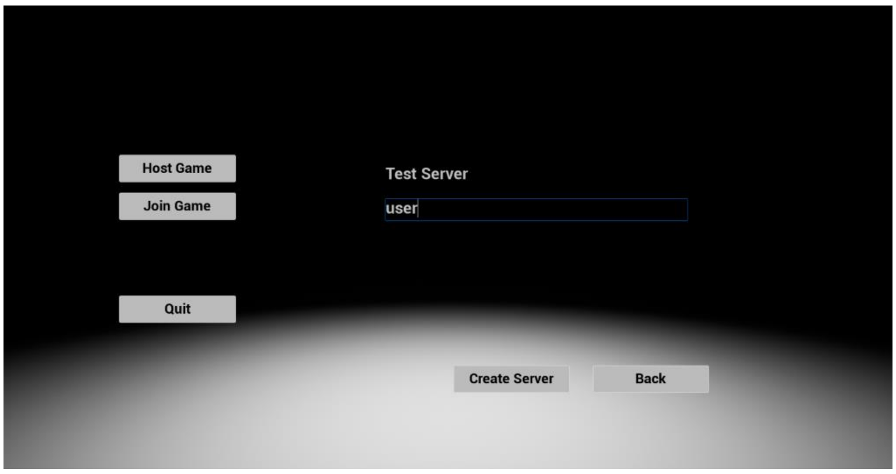
- Megascans library

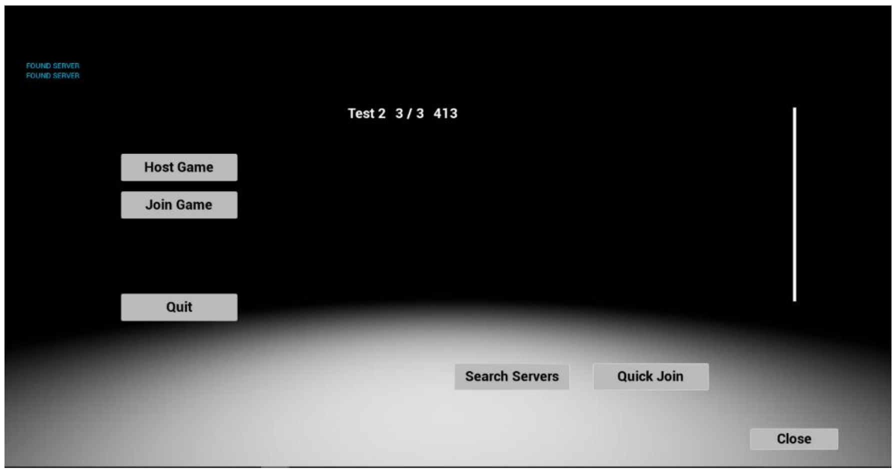
Consisting of about 16 thousand Nanite-ready assets, the library is built into the engine. It has been comprised of real-life scanned high-fidelity objects from rock formations to simple household objects. This makes it perfect for open worlds and, combined with Unreal’s graphics, is ideal for building the world of the game.

Game Flow

Upon starting the game, the player is greeted with the main menu. Currently it consists of the two main buttons – host and join a game.

- Host menu -> opens the host menu. The hosting player enters the Session Name and Host Name and creates the session
- Join menu -> opens the join menu. Upon opening it, the game searches for open sessions and if any are found they are displayed and available to join





Upon creation, gameplay begins in the tutorial cave where players are introduced to the key mechanics of the game. There they will also face their first enemy. After completion of the cave, players proceed to the hub of the game from where the game unrolls.

While the game is oriented towards 2-player gameplay, it can still be played solo, functioning as its own server host and client.

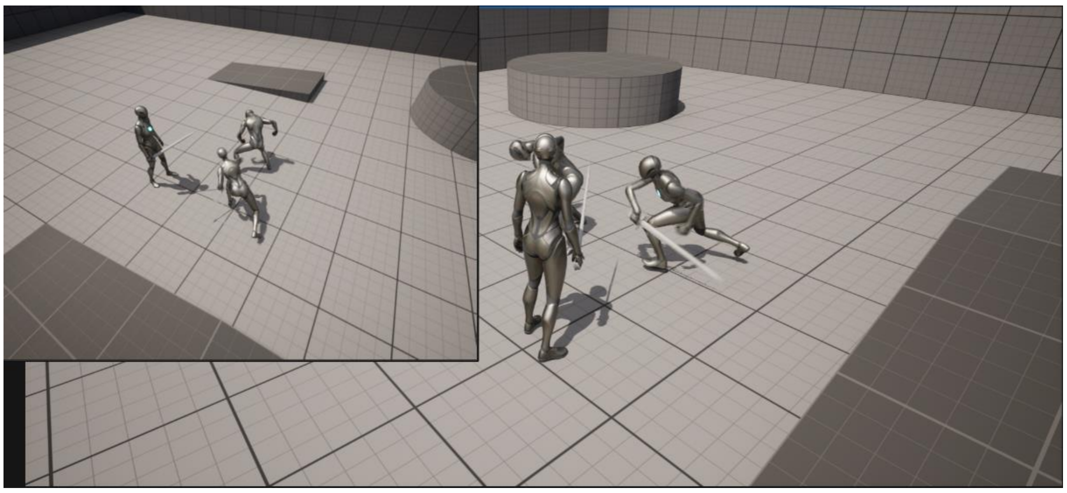
Mechanics

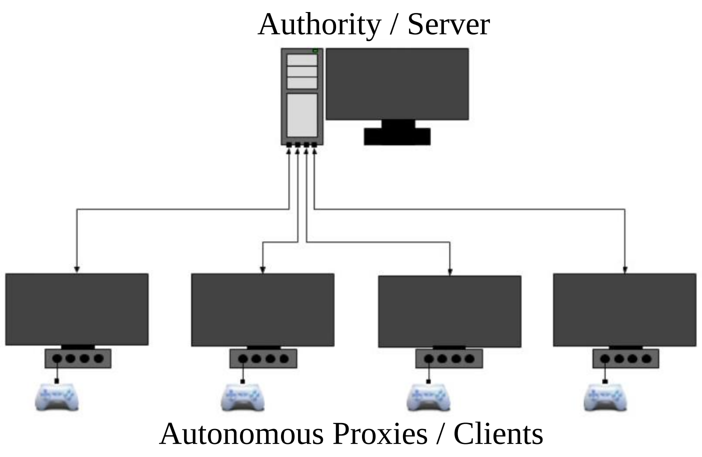
The main character is currently represented by the default Unreal mannequin.
Moveset:

- WASD & mouse – move along the four directions and look around
- Left Mouse Click – attack with sword
- Alt + W/A/S/D – side step/dodge in the respective direction

Multiplayer

The game uses Unreal's built-in Networking module and is built around a client-server model. One computer in the network acts as a server and hosts a session of a multiplayer game, while all of the other players' computers connect to the server as clients. The game itself always takes place on the server (be it a player's system or a dedicated one) and the server has the responsibility to communicate to all clients connected to it changes to the state of the game (e.g. animations, actor movement, changing levels, etc).





Conclusion

During the duration of SRS we have managed to successfully cover the initial layout of the project. We will carry on with the game into the following school year and continue development.

Used technologies

- Unreal Engine 5.0.3
- JetBrains Rider
- Mixamo Library

Third parties

The meshes used for the game are publicly available from the Quixel Megascans library and free to use. The animations were downloaded for free from the Adobbe Mixamo Library and converted to Unreal Engine format.

Future development

Having covered basic functionalities, we will start exploring more complex topics, add more mechanics and content to the game, namely:

- Improve combat mechanics with better movement and dodge mechanics
- Improve multiplayer functionality
- Improve menus
- Add save&load feature
- Add content to game
- Improve stability