# MANIGANDAN

## GAMEPLAY PROGRAMMER

+91 8870069369

manigandanrajan14@gmail.com

https://manigandanrajan.wixsite.com/portfolio

https://github.com/AuremMJS

## EXPERIENCE

### Unity Developer at Stratton Studios, Australia (June 2024 - Present)

- Designed the terrain for a Metaverse island using height maps and generated vegetations using Procedural Worlds Storm.
- Created NFT plots and developed Roads using EasyRoads.
- Created Editor tools for plot generation, asset integration and terrain custom painting.

### Freelance Slot Game Developer (February 2024 – June 2024)

- Worked as Freelance Slot Game Developer to create the probability math and feature generation in a Casual Slot Game using Unity 3D and C#.
- Used Unity Test Framework to test the Slot game features and spin orchestration.

### Unity Developer at Product Madness, Poland (March 2022 - January 2024)

- Worked as tech lead and managed project with two developers.
- Programmed and maintained the audio system in match-3 casual game.
- Worked on Firebase analytics and creating features and events for the game.
- Integrated Spine animations for characters.

### Software Engineer at Scopely (2021 – 2022)

- Developed user acquisition features and inbox messages using Zenject.
- Created mini games within GSN Casino app.

### Freelance Game Developer (2019 – 2021)

- Worked as freelancer taking up game development projects using Unity 3D and C#.

### Software Engineer at Light & Wonder (2017 – 2019)

- Ported existing slot machine games created using C++ to Unity and C# and developed slot games in C++.
- Designed and developed Old School Slot Games with multiplayer support.
- Created new mechanical orchestration of jackpot celebration.

### Associate Product Developer at DXC Technology (2015 –2017)

- Designed the UI in WPF/XAML for developers' knowledge sharing Visual studio plugin and got rewarded with Amazon vouchers for excellent UI.
- Developed and maintained features in healthcare application using ASP.NET and SQL queries.

## LANGUAGES & TECH

- C++
- C#
- Vulkan API
- OpenGL

## GAME ENGINES

- Unity3D
- Unreal Engine

## EDUCATION

- MS in High Performance Graphics and Games Engineering, University of Leeds – *September 2019 to November 2020*
- BE in Computer Science and Engineering, Saveetha Engineering College – *August 2011 to May 2015*

## KEY PROJECTS

- Physics Simulation of Bouncing Balls – C++ and OpenGL
- Cloth Simulation – C++
- Fur Rendering – C++
- Raytracing – C++ – Code
- Tree generation using L-System Trees – C++ – Code
- Simulating reflection in a mirror – C++ – Code
- Loop subdivision of triangle soup – C++ – Code
- Memory Manager – C++ – Source code
- Collision Detection using octree – C++ – Source code

## PERSONAL PROJECTS

### Bank Heist – Unreal Engine 5 and C++

Bank Heist is a game developed in Unreal Engine 5 using C++ and Blueprints. It is a third-person camera game, in which a simulation of secret Bank vault is achieved. Crossing across the end of the vault without getting hit by the dynamically moving laser rays is the goal of the game. Line trace is used to trace the player and reduce the health.

### Procedural Forest Generation using Level Streaming – Unreal Engine 5 and C++

Created a Procedural Forest Generation algorithm to load trees and vegetations in the forest to get generated based on a grid-based logic using Level Streaming technique.

### Fidget Spinner and Million Particles – Unity ECS and C#

Created a particle system with million particles moving randomly within a bounded box and with random velocity. Upon interaction with fidget spinner in the middle of the box, the particles get physically collided and move away from the fidget spinner. Used the Unity's Entity Component System to make this simulation. I used the job system and Burst compiler to efficiently calculate the movement of the particles and individual systems handle different movement and physics logic of the particles and the fidget spinner.

### Memory Manager

Created a memory manager with implementation of smart pointers and stack memory, double stack memory and pool memory data structures in C++. Source code

### Collision Detection using octree

Created a collision detection program using octree partitioning. Handled collision in three different phases and filter the colliders based on layers. Rendered using Vulkan API and programmed in C++.

### Stay Home – Unity3D and C#

A game developed using Unity3D for WebGL for creating awareness about social distancing. Used skeletal animations and ragdoll physics to animate the humans. Source code. Play this game!

### Color Snipers U – Unity3D and C#

A casual mobile game developed using Unity3D. Used scriptable objects for storing the level data. Shader graphs are used for special effects. Source

### Honey Collector – Unity3D and C#

Developed a game to render honey in a spoon and honey falling into the jar as a particle system. Source code

### Rendering Biological Iridescence – C++

Used Vulkan API to render iridescent colors of a butterfly by spectral computation method. This project is selected as featured student project by University of Leeds. Source code