

SUMMARY

Gameplay systems developer specialising in scalable, data-driven mechanics and backend-integrated gameplay. Experienced in Unreal Engine, Unity, and web-based game frameworks, with a focus on combat systems, progression design, and runtime architecture.

Strong track record of delivering production features, building reusable systems, and designing frameworks that enable emergent gameplay and rapid iteration.

TECHNICAL SKILLS

Programming Languages - C++, C#, TypeScript, JavaScript, Java, Python, SQL

Engines & Frameworks - Unreal Engine 5, Unity, PIXI.js-based frameworks

Backend & Infrastructure - Amazon Web Services (AWS), Supabase, Docker

Core Competencies - System Design, Gameplay Architecture, Prototyping, Scripting, ECS Patterns

Professional Skills - Agile & iterative development, cross-team collaboration, working to production deadlines, live service development, debugging & optimisation

EXPERIENCE

Junior Developer At Light & Wonder - iGaming

May 2025 – April 2026

- Developed gameplay systems and game logic using TypeScript within a proprietary PIXI.js-based framework
- Built reusable gameplay components adopted across multiple shipped titles
- Collaborated with design, art, and QA to deliver gameplay features, UI behaviour, and animation integration
- Diagnosed and resolved gameplay issues across live production builds
- Contributed to multiple live releases under tight deadlines in an Agile environment

Lead Gameplay Programmer & Designer At MNNT / Andrometa

Indie Mobile Project (8-person team)

Aug 2023 – May 2025

- Led gameplay development for a **published 3D roguelike looter-shooter** (iOS & Android) using Unreal Engine 5
- Designed and implemented core systems including combat (melee/ranged), progression, and stat scaling
- Built an online trading system backed by a custom API and persistent backend
- Integrated gameplay systems with VFX and animation pipelines alongside artists and designers
- Implemented monetisation systems balancing ads with free-to-play progression
- Developed a player feedback pipeline, improving bug resolution turnaround by ~40%

PROJECTS

Arcanum – Domain-Specific Gameplay Language & Virtual Machine (*Flagship Project*)

- Designed and implemented a **domain-specific language (DSL)** for gameplay, enabling players to author spells via a node-based programming model
- Built a custom **virtual machine (SpellVM)** to compile and execute spell graphs with deterministic runtime behaviour
- Developed an **ECS-based execution pipeline** supporting asynchronous effects, deferred events, and complex system interactions
- Engineered the system for **composability and scalability**, shifting gameplay power from static abilities to programmable logic
- Enabled **emergent gameplay patterns** through modular composition, mirroring principles seen in systems like programmable magic frameworks and scripting runtimes

Wave Function Collapse (WFC) – Procedural Generation System

- Designed and implemented a modified Wave Function Collapse system in Unity for generating traversal-friendly parkour environments
- Developed a modular constraint framework using socket rules, rotation handling, and multi-level adjacency validation
- Tagged environment modules with traversal actions such as jump, climb, and wall-run to guide generation toward playable movement routes
- Built custom tooling to bake authored prefabs into reusable procedural modules for faster iteration and content scaling

Don't Look - Shader-Based Echolocation

- Designed and implemented a shader-based echolocation system that reveals the environment through expanding visual pulses, turning navigation into the core gameplay mechanic
- Built custom URP rendering passes to highlight enemies and interactable objects, using layered colour and object ID buffers to support clear gameplay readability in low-visibility scenes
- Developed the experience in Unity 6 with C#, integrating the Input System and XR tooling to support immersive first-person interaction
- Created an atmospheric horror prototype set in an abandoned hospital, using limited visibility, audio cues, and spatial exploration to drive tension
- Implemented environmental interaction systems such as doors and object affordances to reinforce exploration and player decision-making under pressure

EDUCATION

BSc Computer Science - University of Nottingham

International Baccalaureate (IB) - Dartford Grammar School (Sixth Form)

11 GCSE qualifications including Mathematics, Physics, Computer Science - Dartford Grammar School