Idhant Dabral

Email: i.dabral26@gmail.com
Linkedin:https://bit.ly/4md4Ok4
Portfolio: https://bit.ly/4kUpyeJ

GAMEPLAY PROGRAMMER

I'm a Gameplay Programmer with 2 years of experience & proven track record for creating innovative **player facing experiences** across consoles and PC. I am excited to empower teams while facing technically demanding challenges. Specialised in **C++** with a comprehensive knowledge in **game engine architecture** and FPS mechanics.

SKILLS

Programming Skills	**C++**, C#, Lua
Commercial Game Engine	**Unreal Engine 5**, Unity 3D, RAGE (Deployment)
3D Graphics API	**OpenGL**, Vulkan, HDRP, URP
Gameplay Programming Patterns	Gameplay Ability System **(GAS)**, Entity-Component System **(ECS)**, Object Pooling, **Instancing**, State, Command
Version Control	Git, **Perforce**

PROFESSIONAL EXPERIENCE

Newcastle University, Newcastle Upon Tyne — *VR Gameplay Programmer*

MAY 2024 - NOVEMBER 2024

VR game in C#/Unity Engine, Targeting SteamVR and the Meta Quest.

- I developed a modular **Data Driven Game Event** system for in game responses to character movement and interactions. Allowed future development by both code and non-code teammates.
- Implemented full **VR 3D character movement**, with a both Discrete and Continuous movement mode, with the Meta controllers, allowing for a tight and intuitive player controls.
- Developed **Player Interaction** like clicking buttons, **interacting with physics objects in VR** allowing for dynamic gameplay events to trigger.
- Worked with HDRP to **optimize** Rendering on Meta Quest 3.
- Wrote and maintained Tech Documentation for building and code analysis.

Rockstar Games, India — *Dev Support QA (link)*

2022 - 2023

Worked as Dev QA building all rockstar games to PC, Playstation and Xbox Dev kits. Provided support for the gameplay Quality Assurance team as well.

- **Built all Rockstar titles** from source, to provide Dev Support QA on titles like, GTA-VI, GTA-V Online, RDO, including online character data tracking, **quest reward tracking**, and in-game sanity.
- Worked with building **PlayStation** DevKits, **Xbox** Dev kits, PC, and Phone.
- Provided **Dev support** by logging bugs at a near **97%** accuracy with the given tech documentation, providing new test cases, and **automated testing**.

- Followed up on Persistent bugs and helped with test suites on all Rockstar back-end online services, including The **Rockstar Games Launcher**, and other backend tools.
- Helped cross discipline teams with technical Challenges, providing support to the gameplay QA team, and bridged the gap between some QA and tech.

Jambox Games, Remote Global — *Gameplay Programmer (link)*

JANUARY 2022 - AUGUST 2023

Mobile Block Game — Mobile

Multiplayer asynchronous PvP Block based game for Android and IOS

- Developed a **JSON based Level Generation** system that allowed for a near 2x increase in content creation. This system was then extended to add **Powerups, High Score Stats,** and **Level Based Timer.**
- Worked on the **Asynchronous PvP SDK**, and the in-game **Player leaderboards**, and Player and account Currency Tracking**.**
- Implemented a Sound Manager solution for usage across all games using our PvP SDK.

Unannounced Title — Mobile

- **Bridged the gap** between the **Design Team,** and the **Tech team** by iterating, developing and working on the main new project's core concept and systems.
- Worked with the design team on multiple **POCs** including writing up tech documentation and plans for proper implementation with evidence based research.

Game Engine SDK, Tools Development

- Implemented **Dev Tools** to allow developers working with our SDK backend to run user analytics, to provide developer insights like **Playtime**, **Region,** and popularity.
- Worked with implementing our Asynchronous PvP SDK with an endless runner game.

PROJECTS *(link)*

Project Borne — *PC (link)*

Action RPG Combat system in Unreal Engine 5, with a focus on Gameplay Ability System, AI and Modular Weapon System.

- Implemented **Action RPG** Third person combat in Unreal Engine 5, featuring **4 different weapon archetypes**, Player and Weapon Stat systems, and a State based AI Boss fight.
- Programmed a **Modular and Robust** weapon system solutions that allowed for weapon data attributes to interact with the Unreal Engine's Attribute Set, allowing them to feel unique as they drain stamina and deal varying damage. Ease of adding new weapons and features to this system was also emphasized.
- Worked with EQS, Pawn Sensing, Custom AITasks, AIDecorators, AIServices within **Unreal Engine's Behavior tree** to create a robust, difficult and fun Boss fight.
- Made Use of Gameplay Tags, Gameplay Cues, Gameplay Effects and Notifies to further extend the player's abilities and actions. This allowed for easy Scalability and Rapid new feature prototyping with GAS.

Hellrunners — *PC (link)*

An Online Multiplayer First Person Speedrunning game developed in C++.

Team Lead, Core Tech Programmer

- As a Team Lead, I guided and helped a team of 8 developers finish and deliver a fun and technically adept, first person multiplayer speedrunning game.
- Built the **Core physics** and the **3Cs** for the game, resulting in a responsive first person controller, and a smooth ability system for momentum based gameplay.
- Spearheaded team management including task distribution (incl. JIRA), managing standups, and target based planning.
- Implemented the **main level loader**, making it data driven and **multi-threaded**, and accessible on multiple game engines. This allowed for our game's render and level loading to speed up **2 times**.
- Programmed the **Player input**, using a **Command design pattern**, and worked on many helper classes, including analytics for performance analysis and benchmarking.
- Worked on the scoring system, which resulted in level based timers and medals depending on when you finish your race.
- Implemented the kill plane, alongside similar player constraints. This also included physics constraint programming and bound checking.
- Worked with **GLSL**, and **OpenGL** to further add gameplay variance to the game. Implemented many shaders, both UI and in game.
- Wrote effect shaders - Speed lines, Lava, Fire, and the animated effects of the main menu with Lua.
- Helped with bug fixes, daily code standup, code reviews acting as a tech lead, setting up milestones.

Procedural Foliage System — *PC (link)*

Procedurally Generated Grass System in C++ and OpenGL with Compute Shaders.

- Developed a Grass system using Compute shaders, with features for wind, colorization and tool support.
- Optimized using **Compute shaders**, **GPU Optimization** techniques like GPU **Instancing**, and **Object Pooling**. This allowed for quadrupling the amount of objects instantiated, while keeping the framerate drop minimal.
- Added a **Wind** system, allowing multidirectional Grass sway, making each grass blade look rendered and individually animated. This was based on noise based generation, with GPU generated cellular noise.
- Implemented **Chunking**, **Camera culling**, **Grass clumping**, and camera based distance blending.
- Worked on height blending, allowing smooth and clean transitions based on height of the grass blade placed upon the terrain below.
- In-built **brightness, saturation, gamma correction, and contrast controls**. Also worked with ImGUI to provide a visual debugger.
- Performed Performance analysis, evaluating, and passing through all the benchmarks set by product criteria.

EDUCATION

Newcastle University, Newcastle Upon Tyne — *MSc Computer Game Engineering (Distinction)*

SEPTEMBER 2023 - SEPTEMBER 2024

Worked a year on C++, OpenGL, Engine Level programming graduating with a Distinction.

Manipal University Jaipur, India — *BTech Information Technology*

SEPTEMBER 2018 - SEPTEMBER 2022

Undergraduate degree with emphasis on practical programming language skills like JAVA, C and a specialization in AI/ML.