Michael Joyce

Email: mike.b.joyce@gmail.com
Phone: (514) 222-2102
Location: Montreal, Quebec
GitHub: @mikebjoyce

Professional Summary

Performance-focused software engineer with over a decade of experience building real-time gameplay systems and runtime architectures in Unity and C#. Co-founder and Lead Gameplay Systems Developer at Pansimula, where I architect high-performance gameplay systems and lead technical optimization across shipped titles. Profiling-driven by practice: I measure before optimizing and know when optimization actually matters. Recently expanded into JavaScript backend and operational tooling through a suite of MIT-licensed plugins for the SquadJS framework, deployed on live 100-player competitive multiplayer servers. Proven educator with experience teaching C#, algorithms, and data structures in classroom settings.

Education

Bachelor of Arts — Philosophy
Concordia University, Montreal, QC
Completed 2025

Additional Coursework — Software Engineering & Computer Science
Concordia University, Montreal, QC
2012 – 2018

DEC — Liberal Arts
CEGEP, Quebec
September 2009 – May 2012

High School Diploma
Quebec
Graduated June 2009

Core Specializations

- Real-time gameplay systems and runtime architecture
- Profiling-driven optimization and performance engineering
- Physics-heavy simulation systems
- Gameplay AI systems and pathfinding
- Runtime tooling and procedural systems
- Compute shader workflows (HLSL)
- Procedural and quasi-procedural gameplay systems
- Event-driven tooling and operational systems
- Technical problem-solving under production constraints

Work Experience

Pansimula Indie Game Studio

Co-Founder | Lead Gameplay Systems Developer
March 2017 – Present

- Co-founded studio and led design and development of *Galactic Thunderdome*, a physics-based brawler released on Steam (October 23rd, 2024).
- Contracted and delivered 4 educational titles to Legends of Learning under royalty-based agreements, with later contracts negotiated to retain studio IP rights.
- Architected and implemented core gameplay and runtime systems under heavy real-time simulation constraints, including gravity systems, projectile systems, adhesion mechanics, environmental hazards, AI systems, and procedural interaction systems.
- Served as primary performance-focused developer across production, profiling and resolving runtime bottlenecks, GC spikes, memory issues, and frame-time instability.
- Optimized physics-heavy gameplay systems to sustain above 60 FPS in extreme scenarios involving multiple players, AI entities, destructible terrain, and interacting environmental systems.
- Designed extensive pooling and staged initialization systems to prevent frame hitching and maintain stable runtime performance in simulation-heavy gameplay scenarios.

- Designed and implemented character systems including ragdolls, hybrid keyframe and procedural animation systems, gun handling, physics interaction, gravity transitions, and zero-gravity gameplay behaviors.
- Developed heuristic state-machine AI systems integrated with dynamic A* pathfinding operating on destructible and reconstructible environments with environmental hazard awareness.
- Implemented compute shader workflows to offload CPU bottlenecks in destructible terrain generation pipelines involving runtime Voronoi geometry processing.
- Built tooling and runtime systems improving iteration speed and usability for designers and content creators.
- Mentored junior developers through technical problem-solving, debugging, and systems implementation support.
- Assisted with Steam deployment workflows, production coordination, release preparation, post-launch bug diagnosis, and rapid hotfix deployment.
- Produced promotional video content and advertisements for *Galactic Thunderdome* using Adobe Premiere and Illustrator.
- Designed and implemented gameplay audio systems and original sound assets using Adobe Audition, including dynamic spatial audio approximation systems for large-scale entity groups and environmental effects.
- Currently prototyping and iterating on smaller-scale gameplay concepts in preparation for the next production cycle.

ROAM

Freelance UI Developer
March 2025 – April 2025

- Built runtime UI components and layouts using Unity UI Toolkit for an AI-driven game creation platform.
- Developed a procedural UI generation system using reflection to scrape object data and types at runtime, dynamically constructing interface panels from modular configurable elements.
- Developed lightweight animation systems for in-app Unity UI panels.

Institut Supérieur d'Informatique

Course Instructor
May 2020 – February 2022

- Designed and taught courses in C# and algorithms in both classroom and remote learning environments, with emphasis on boolean algebra, data structures, time/space complexity, and algorithm design.
- Created all course materials including lectures, assignments, and examinations.
- Delivered remote instruction through Microsoft Teams during the COVID-19 pandemic while maintaining strong student engagement and outcomes.

Freelance Technical Document Translator
February 2018 – October 2019

- Translated technical course materials from French to English across a wide range of subjects.
- Ensured accuracy, clarity, and terminological consistency in all translated material.

Projects

SquadJS — Server Management Plugin Suite

Open Source (MIT) — JavaScript
~8 months active development | Used in production on live 100-player multiplayer servers

- Developed a suite of three event-driven SquadJS plugins forming an integrated skill management system: TrueSkill ratings, skill-aware team balancing, and intelligent player assignment.
- Implemented a custom variant of Microsoft's TrueSkill ranking system tuned for 50v50 team sizes, validated against weeks of production game data before live deployment. Currently tracking over 12,000 players on a single server.
- Built a 4-phase exhaustive search scramble algorithm prioritizing squad cohesion, with ELO-weighted scoring when skill data is available.
- Implemented sub-2-second verified player assignment using force-polled player state to work around a 30-second RCON refresh limitation in the SquadJS framework.
- Implemented persistent systems resilient to hard resets and server crashes through SQLite-backed state management and reload-safe initialization.
- Built full Discord integration pipelines for operational feedback, administrative control, and runtime server management.

- Designed systems prioritizing operational usability and auditability for non-technical server administrators.

Galactic Thunderdome

Unity / C# — Released October 23rd, 2024 | Steam

Physics-based brawler emphasizing dynamic simulation, destructible environments, and real-time AI. Lead gameplay systems developer. See Pansimula entry above for full technical detail.

Educational Games for Legends of Learning

Unity / C#

- **Malfunction** (2019) — Math and science tower defense for grades 6-12. Players answer questions to defend a city from asteroid impacts.
- **Grow Hop** (2018) — 2D platformer for grades 2-4 teaching photosynthesis through environmental interaction systems.
- **Dance of Atoms** (2017) — Chemistry puzzle game for grades 7-10 involving strategic particle combination mechanics.
- **Bid for Life** (2017) — Biology strategy game for grades 9-11 exploring animal reproductive strategies through resource management scenarios.

Skills

Programming Languages

- C# (12+ years)
- JavaScript
- HLSL / Compute Shaders
- SQL (SQLite — schema design, WAL, concurrency patterns)

Software Development

- Real-time systems architecture
- Profiling-driven optimization and debugging

- Runtime and memory-conscious engineering
- Algorithmic complexity and data structures
- Object pooling and staged initialization systems
- Physics optimization for gameplay systems
- Gameplay AI and pathfinding systems
- Event-driven architecture
- Database-backed state management and persistence
- Open-source development (MIT-licensed public repositories)

Game Development

- Unity (12+ years)
- Unity Profiler (CPU, GPU, memory, physics)
- Unity UI Toolkit
- Gameplay systems engineering
- Simulation-heavy gameplay architecture
- Procedural and quasi-procedural systems
- AI behavior systems
- Runtime UI generation
- Technical prototyping and iteration workflows

AI-Assisted Development

- Cline (VSCode) as primary AI-assisted development workflow
- Hands-on evaluation of Claude and Gemini across real engineering tasks
- Model selection and cost-utility optimization based on task complexity and workflow efficiency
- Strong traditional programming background established prior to AI-assisted development workflows

Tools & Technologies

- Git / GitHub
- Visual Studio / VSCode
- Node.js / Yarn
- SQLite
- Discord API

- Jira
- Adobe Audition
- Adobe Premiere
- Adobe Illustrator
- Microsoft Teams

Languages

- English — fluent (spoken and written)
- French — fluent (spoken), basic (written)

Additional Strengths

- Technical mentoring and instruction
- Cross-disciplinary collaboration
- Production problem-solving
- Runtime debugging and issue diagnosis
- Rapid prototyping balanced with production-focused optimization
- Small-team technical leadership and coordination