

# ERIC MAY

Software Engineer · Unity & C# · Backend & Cloud · Game & Simulation Systems

EricMay256@gmail.com · (951) 707-3011 · [github.com/EricMay256](https://github.com/EricMay256) · [ericmay256.github.io](https://ericmay256.github.io) · [ericmay256.medium.com](https://ericmay256.medium.com)

Blacksburg, VA · Open to Remote or DC/VA Contractor Corridor

US citizen, eligible for security clearance · Open to full-time, contract, or contract-to-hire

## PROFESSIONAL SUMMARY

---

Software engineer with 10 years of applied CS experience building Unity/C# game and simulation systems, backend services, and developer tooling. Shipped production software including a cloud-hosted leaderboard backend (FastAPI, PostgreSQL, Redis, Heroku) with a Unity C# SDK, a commercial game on Steam and Google Play, and a published Unity utility library. Comfortable across the stack from gameplay code to backend services to client integration. Multi-language programming depth from a decade of university teaching and project work in algorithms, data structures, memory management, and testing infrastructure.

## TECHNICAL SKILLS

---

**Languages:** C#, C++, Python, C, SQL, TypeScript, JavaScript

**Engines & Frameworks:** Unity (6, URP, Input System, UI Toolkit, WebGL builds), FastAPI, React, Vite

**Data & Backend:** PostgreSQL, Redis, Jinja2, REST APIs, SQL window functions

**Cloud & Infrastructure:** Heroku, AWS (EC2, RDS, S3, Elastic Beanstalk), Docker, Linux

**Testing & CI:** Pytest, GitHub Actions, Sentry observability, integration test harnesses

**Disciplines:** Game Development, Full-Stack Engineering, Simulation Systems, Procedural Generation, Curriculum Engineering, Technical Writing

## SELECTED PROJECTS

---

**High Score Server** — Full-Stack Cloud Backend with Unity SDK · [Live demo](#) · [Repo](#)

- FastAPI / Python REST backend for persistent leaderboards across Unity games; deployed on Heroku with PostgreSQL and a swappable cache backend (Redis opt-in, in-process TTL by default).
- Unity C# SDK with coroutine-based API calls, typed response models, and a result-wrapper error pattern; supports full auth lifecycle including silent guest login, token storage, and account claiming. Design writeup: [Schemas as Claims](#).
- Concurrent score submissions, authenticated writes, ranked reads via SQL window functions, and IP-based rate limiting; supports multiple games and game modes per deployment.
- Production observability via Sentry with sampled performance tracing; pytest integration suite against an ephemeral Postgres service on GitHub Actions on every push and PR; continuous deployment to Heroku from main.
- Companion live demo *Flick Fest* — Unity WebGL build hosted on GitHub Pages submitting real scores to the live HSS deployment ([ericmay256.github.io/Flick-Fest-Build](https://ericmay256.github.io/Flick-Fest-Build)).

**Minesweeper Madness** — Shipped Commercial Game · [Steam Store](#)

- Solo-developed and shipped on Steam and Google Play with full ownership of design, implementation, build pipeline, and store-side operations; Unity/C# across gameplay, UI, audio, save data, and platform integration.
- Designed and implemented many custom gameplay variants beyond the classic Minesweeper ruleset, with mode-aware scoring and progression.
- Wrote a [platform-lifecycle devlog](#) on Steam-vs-Play maintenance tradeoffs and the engineering case for platform consolidation.

**UBear** — Unity Utility Library, Published UPM Package · [github.com/EricMay256/UBear](https://github.com/EricMay256/UBear)

- Cross-project Unity library providing composable primitives—reusable systems, components, and tooling—including the C# SDK client for the High Score Server.
- Published as a versioned UPM package (com.ericmay.ubear) installable via Unity Package Manager from the repository's git URL; MIT-licensed, semver-tagged, with separated runtime and test assemblies.

**Intro Game Dev Course** — Unity Curriculum · [github.com/EricMay256/intro-gamedev-course](https://github.com/EricMay256/intro-gamedev-course)

- Solo-authored Unity 6 / URP project backbone for a university game development course; 12 progressive lessons covering physics, scripting patterns, tilemaps, audio, animation, procedural generation, debugging tools, shaders, and multiplayer.

- Used by 550+ students across multiple cohorts; published as a public MIT-licensed template repository for educators and self-learners.

**Portfolio site** · [ericmay256.github.io](https://ericmay256.github.io) — hand-built site (HTML/CSS/JS, GitHub Pages) presenting the above work with live data, architecture writeups, playable and video demos, and a downloadable resume.

## PROFESSIONAL EXPERIENCE

---

**Lecturer — Computer Science** · *California State University, Fullerton* *2016 – Present*

- Designed and delivered Unity/C#, C++, and Python coursework across game development, algorithms, systems, and introductory programming; applied engineering practices (version control, automated testing, CI) as first-class teaching content.
- Rebuilt the introductory Game Development course from theory-only to a hands-on Unity/C# project track—solo-authored 12 progressive Unity 6 lessons used by 550+ students across multiple cohorts; project repo is public, MIT-licensed, and continues to evolve.
- Built automated grading infrastructure in Python and C++ with GitHub Classroom integrations, cutting grading cycle time and improving feedback consistency across 3–5 sections per semester.
- Designed and deployed an AI academic integrity framework—syllabus policy, structured assignments, and evaluation rubrics for responsible AI use—adopted ahead of most peer institutions.

**Independent Developer / Contractor** · *Self* *2013 – Present*

- Re-engineered a client's legacy VB.NET line-of-business application into a modern C#/SQL stack; designed schemas and front-end software through direct consultation with management and end-users; deliverable served as the foundation for the production uplift the client subsequently adopted.
- Designed and implemented procedural generation systems in Unity/C#—Perlin/simplex noise-based heightmap terrain and dungeon generation with arbitrary room sizes, edge-aware placement, and hallway connection logic.
- Delivered playable game prototypes for external clients on contract, with full ownership of architecture, scope, and version control.

**Full-Stack Software Developer** · *Eddi* *May 2015 – Nov 2015*

- Contributed across back-end (Go, PostgreSQL), front-end (React, Redux), and cloud infrastructure (AWS: EC2, RDS, S3, Elastic Beanstalk; Docker) for a multi-platform marketplace application.
- Integrated Braintree payments and translated product requirements into technical implementations; administered company cloud accounts and managed AWS environment configuration.

## EDUCATION

---

**B.S., Computer Science** · *California State University, Fullerton* · 2015

*Admitted with Distinction; Dean's List*