

IDDO ADAR

Senior Software Engineer · Unity / C# · Client Architecture · Technical Leadership

Nieuwegein, Netherlands · iddocodes@gmail.com · +31 6 17591444 · [LinkedIn](#) · Portfolio: [iddo.codes](#)

SUMMARY

Senior Unity developer with 15+ years in game development, including mobile casual and hyper-casual games and long-running live-service titles. I own features from design through release, and as Client Technical Lead on Bubble Busters (a live 1v1 competitive mobile game) I led client architecture, mentored developers, and ran major refactors without pausing content releases. I work closely with design, art, and production, and I'm happiest where a game feels great to play: animation, game feel, and the tools that let designers move fast.

SKILLS

Languages: C# (primary) – working knowledge of GDScript, Lua, JavaScript, Python, ActionScript and others across past roles and projects.

Game & Interactive Development: Unity · Gameplay Systems · Animation Systems · Unity UI (uGUI) · 2.5D / 3D Character Integration · UI Architecture · Deterministic Multiplayer · Game Architecture

Engineering: Client Architecture · State Machines · Modular / Pluggable Features · Factories · Dependency Injection · Asynchronous Programming · Performance · Refactoring · Tool Development

Collaboration & Leadership: Technical Leadership · Mentoring · Code Review · Agile / Iterative Delivery · Architecture & Design · Cross-functional Collaboration · Jira · Git

AI-Assisted Development: AI-assisted code research · Code exploration · Test generation · Workflow experimentation

EXPERIENCE

Client Technical Lead — PixelPie

Feb 2023 – Aug 2026

- Took technical ownership of the client for Bubble Busters, a 1v1 competitive mobile game with 3D character models in 2D gameplay (iOS/Android, 500K+ Android downloads, 4.9★, live 3.5 years) built with deterministic multiplayer on Photon Quantum. Led technical direction for a ~5-person client team, working closely with the CTO and server lead.
- Designed and built a scriptable sequencing and synchronization system for animations and gameplay logic, executing consistently across multiplayer clients. It became the team standard: it made previously impossible sequences possible, helped keep the client architecture viable for a small team, and cut implementation of a new playable character (Buster) from 1-2 months to 1-3 weeks.
- Developed core gameplay systems and mechanics including bubbles, Busters (playable characters with ultimates), game modes and supporting gameplay infrastructure. Led major refactors to streamline content creation incrementally, allowing releases of new content to continue while underlying architectures are being improved.
- Designed the core state machine architecture in collaboration with the server lead, along with a plug-in/plug-out feature architecture allowing major gameplay and meta features to be independently enabled or removed. This was supported by a queuing mechanism so each feature could add its animations without colliding with others.
- Led development of major meta-game features, including monetization events and player progression systems, delivered through complex UI flows built in Unity UI (uGUI), coordinated animations and dedicated tooling for designers and technical artists.
- Contributed to online and on-device performance work, including a move to a 30 Hz simulation with interpolation: worked closely with the developer who implemented it and reviewed the code, with my sequencing system making animations straightforward to adapt. Also developed allocation-free coding patterns for Photon Quantum, reaching zero runtime allocations in an internal engine prototype.
- Mentored junior and senior developers through regular 1:1s and code review, and contributed to code quality, architectural decisions and development practices across the client team.
- Worked in a fixed release-cadence model (biweekly, later monthly), planning scope with design and production to ship reliably and keep content flowing.

Independent Game Developer

Sep 2021 – Feb 2023

- Developed independent game and software projects while working across design, programming and production, continuing to build expertise in Unity, C# and game architecture.

Unity/Flash Developer — Scientific Games / SciPlay

Jun 2017 – Dec 2021

- Worked with the company's architect to migrate Quick Hit, one of the company's two flagship titles (10M+ Android downloads, ~1M iOS ratings at 4.8★), onto the Jackpot Party codebase, keeping its branding and slots and adding new features. Live to this day on iOS and Android.
- Transitioned professionally from Flash development to Unity and C#, becoming a core gameplay developer within a small three-person hyper-casual team.
- Developed gameplay systems and contributed to a visually editable rules/scripting framework within the Unity Editor.
- Developed an early sequence-based animation system for scripted UI animations, establishing ideas that later evolved into the more sophisticated animation sequencing system built at PixelPie.
- Developed and maintained Flash-based casino games and supporting meta-game features in a large production environment.
- Built an internal tool to automate the import of animations and layouts into the Flash project; developed the initial working version in a single day, and it remained in active use for about 2 years, right up until the Flash project was phased out.

Earlier Career

2009 – 2016

- **Developer, Diwip / Tangelo Games (2014–2016):** Progressed from junior to independent developer on commercial Flash casino games. Built a new slot product from scratch, including a script interpreter/import pipeline for design-to-engine workflows, enabling a production pace of roughly one new machine per week, a significant improvement over prior products. Later led the company's transition from SVN to Git.
- **Independent / Freelance Game Developer (2011–2013):** Developed independent games, prototypes and commissioned interactive projects across programming, design and animation.
- **Client Developer, WDIFA (2009–2010):** Sole client-side developer for an Arabic-learning website, building Flash games and a meta-application integrated with server-side systems over HTTPS.

SELECTED EARLY WORK

- Created interactive games, animations and prototypes from an early age, including commissioned and sponsored projects.
- Independently published a Flash animation that reached ~100,000 views on Newgrounds, ranking #1 for the week and #2 for the day with ~500 comments.
- Designed and developed *Swordsman Steve*, an original game made with a US-based collaborator (animation, publishing); handled game design, levels, music and all programming. Reached ~50,000 views on Newgrounds with community Let's Plays on YouTube.

EDUCATION

Self-taught developer — building interactive projects since childhood and coding since age 16, well before formal study.

Game Design & Development — 3 semesters, Beit Berl College.

LANGUAGES

English (fluent) · Hebrew (native)