

Joshua Lollis

Fullerton, CA | (608) 397-7678

lollisjosh@csu.fullerton.edu

linkedin.com/in/lollisjosh

[Portfolio](#) | [GitHub](#)

C++-fluent nontraditional new-grad SWE with 10+ years of academic and personal work building modular, testable components and tooling, including real-time full-stack applications. Demonstrated meticulous attention to detail at Thales by supporting the content integration team, validating A/V metadata against contract specs, and producing detailed defect reports.

Education

B.S. Computer Science (GPA: 3.63)

California State University - Fullerton, January 2026

Compilers | Data Structures | OS | Assembly Lang.

Skills

Stack C++, Python, C#, Unity

Env | Linux, Windows, Android

Tools | Git, CMake, GitHub (Issues, PRs, Projects)

Methods | Agile/Scrum (standups), Kanban, roadmapping

Projects

Full-Stack Game Software – [Antillery](#) (Capstone, Solo) – [Unity, C#, UXML, USS, Addressables]

2D artillery game with async scene loading and ScriptableObject-driven state machines.

- Built an async loading framework using Addressables + ScriptableObject-based states.
- Implemented the match loop (turn alternation, timers, damage, win/loss).
- Structured unit behavior via composition and a factory-built state machine.

Embedded Firmware - [Pico Telemetry Node](#) (Personal, Solo) - [C/C++, Unity Test, Pico, CMake, Python]

Pico telemetry/command node with a framed serial protocol, streaming parser, and state machine.

- Implemented framed serial packets with streaming parse, CRC16 validation, and auto-resync on malformed frames.
- Built BOOT/IDLE/ARMED/FAULT state machine with fault handling and HAL abstraction for testability.
- Wrote Python host tool for command injection and telemetry logging; unit-tested state logic using Unity/CMake.

Full-Stack Game Software – [Snake](#) (Personal, Solo) – [C++, SFML, CMake, State/Command/Observer]

Complete desktop game with menus, gameplay states, pause, persistence, and new mechanics.

- Implemented menu + gameplay state system and decoupled input routing using State/Command patterns.
- Architected gameplay into discrete systems (movement, collision, scoring) to improve maintainability and testing.
- Persisted high scores using JSON serialization across sessions.

API-Driven Desktop App – [Card Collection](#) (Academic, Team Lead) – [Python, Qt/QML, JS, REST, Agile]

Desktop app for searching and collecting Pokémon TCG cards via a public API.

- Led a small team through sprints/roadmap and delivered working milestones.
- Implemented async API search over 50k+ cards using Python signals/slots for a responsive UI.
- Built a modular UI with 20+ QML components, using JavaScript for UI logic and Python signals/slots for app wiring.

Hardware Interface Utility – [XB2MIDI](#) (Personal, Solo) – [C#, .NET, WPF/MVVM, NAudio, XInput]

Windows tool that maps Xbox controller inputs to MIDI messages/macros for DAW/VST workflows.

- Built a low-latency controller input pipeline with debouncing and thread-safe controller to MIDI translation.
- Created an extensible JSON-backed mapping system for modes/profiles and custom bindings.
- Implemented MIDI device lifecycle handling (enumeration, hot-swap, reconnect, rate limiting).

Custom Language Compiler – [RAT21F](#) (Academic, Solo) – [C++, Lexer, Parser, Semantic Analysis]

Handwritten lexer/parser with semantic checks for a course-defined language.

- Built a modular compiler with lexical analysis, parsing, and semantic validation.
- Implemented an FSM-based lexer and CLI tooling for repeatable test runs.
- Validated each phase with multiple programs to verify correctness and spec compliance.

Select Employment

Content Integration Assistant – Thales Irvine, CA | Oct 2021 - Oct 2022

- Performed QA/QC validation of audio/video content against contract specifications and metadata requirements.
- Documented defects with clear reproduction steps and tracked issues through resolution.
- Partnered with technical teams to verify fixes and ensure deliverables met quality standards.

STEM Robotics Educator – Brain Builders STEM Ed Tustin/Santa Ana, CA | Aug 2022 - Jan 2023

- Taught introductory programming and robotics concepts (Scratch, basic engineering) through structured lessons.
- Managed classrooms (20–30 students) and communicated progress with staff/parents as needed.