

Cody Wiggins

Location: Fort Worth, TX, 76109 | Mobile: (903) 238-1454 | Email: codywigginsdev@gmail.com

Portfolio: codywigginsdev.neocities.org/ | GitHub: github.com/cwiggins12

SUMMARY

Software engineer specializing in C/C++ real-time systems and digital signal processing. Experience building deterministic audio pipelines, low-latency DSP components, and real-time visualization tools. Strong background in FFT-based signal analysis, audio synthesis, and cross-platform systems development on Linux and Windows.

EDUCATION

THE UNIVERSITY OF TEXAS AT DALLAS

Bachelor of Science, Computer Science

Dec. 2025

GPA: 3.884

Relevant Coursework: C/C++ in Unix, Operating Systems Concepts, Computer Graphics, Computer Networks

TECHNICAL SKILLS

Programming Languages: C++, C, Python, Java, SQL

Frameworks, APIs, & Tools: JUCE, miniaudio, CMake/Make, SDL2/SDL3, Git/Git LFS, Unreal Engine 5

DSP & Audio Systems: Fast Fourier Transform Analysis, IIR Filtering, Signal Generation, Peak/RMS Metering, Spectral Visualization, Deterministic Processing, Low-Latency Systems, Noise Modeling

Systems Programming: Multithreading, Thread-Safe Data Exchange, Lock-Free Mechanisms, Memory Management, Low-Level Debugging, Performance Optimization, Linux & Windows Systems Programming

PROJECTS

12 Band Parametric Visual Equalizer - C++ / JUCE / DSP

- Implemented a real-time, thread-safe C++ application using lock-free audio thread design and RAII
- Built modular DSP components and reusable filter architectures supporting multiple configurations
- Developed real-time signal processing pipelines with strict latency and determinism requirements
- Implemented interactive visualization tools for real-time data inspection and FFT-based analysis
- Integrated parameter and user preferences persistence system for per instance state management
- Emphasized performance, determinism, and reliability suitable for production environments

Cross-Platform Stand-Alone Signal Generator - C / SDL2 / DSP

- Designed and implemented a real-time audio signal generator in pure C using SDL2
- Built deterministic per-sample DSP pipeline supporting sine, square, saw, triangle, and white & pink noise
- Implemented click-free parameter transitions using multiplicative frequency and linear amplitude ramps
- Enforced real-time safety in audio callback (no dynamic allocation, deterministic execution)
- Implemented thread-safe parameter exchange between UI and audio threads
- Produced cross-platform statically linked binaries with embedded assets and zero runtime dependencies

Team-Based Game Development Project - Unreal Engine

- Led a team of 5 engineers through the full software development lifecycle from design to delivery
- Implemented a modular state persistence system supporting serialization and restoration of runtime data
- Designed interface-driven systems architecture managing runtime object creation and lifecycle events
- Developed texture-based shaders, post-processing, and environmental lighting systems
- Defined Git workflows, coding standards, and reference implementations to assist collaboration

PROFESSIONAL EXPERIENCE

Freelance Audio Engineer

Mar. 2014 - Current

- Managed end-to-end live audio signal chains from microphone capture through FOH and monitoring
- Diagnosed real-time audio issues including signal integrity, RF interference, and routing faults
- Coordinated with performers and production crews to deliver reliable, high quality experiences
- Developed strong expertise in audio signal flow, acoustics, and real-time system behavior