

Zachary Joubert

Minnesota | joubertzack@gmail.com

SUMMARY

Software developer and systems engineer with 7+ years of hands-on development experience, specializing in scripting, debugging, automation, modular architecture, developer tooling, and complex system design.

Experienced building software from the ground up, including gameplay systems, UI frameworks, procedural-generation tools, APIs, persistent data systems, agent architectures, execution systems, validation infrastructure, and autonomous software-engineering platforms.

My approach to engineering comes from a broader technical background spanning software development, diesel systems, industrial maintenance, robotics, electrical troubleshooting, and appliance repair. Across disciplines, I focus on understanding the underlying system, identifying root causes, and building solutions that are maintainable and reliable.

EDUCATION

Montgomery High School

High School Diploma

Montgomery, TX

2009–2012

Scoular Inc.

Forklift / Skid Steer Certification

2022

Diesel Mechanic Certification

Driver's License

EXPERIENCE

Lead Technical Designer & Scripter

ClixEnt

January 2019 – Present

Minnesota

Primary software developer responsible for the technical development of Roblox projects and supporting development infrastructure.

- Design and implement modular gameplay architecture using Lua, ModuleScripts, object-oriented patterns, and event-driven systems.
- Develop reusable UI frameworks for settings, audio, controls, graphics, and other interactive systems.
- Build procedural-generation systems, terrain tools, heightmap pipelines, progression systems, interaction systems, and custom developer tools.
- Design systems for long-term maintainability rather than one-off implementations.
- Debug and optimize existing systems while identifying underlying causes of failures.
- Manage the complete technical development lifecycle from prototype through implementation, optimization, testing, and live updates.
- Develop systems supporting PC, console, and mobile platforms.
- Maintain and extend large codebases while balancing new development with stability and compatibility.

Diesel Mechanic

Valley Bus

August 2024 – May 2025

Fargo, ND

- Diagnosed mechanical, electrical, fuel, and emissions-system faults using systematic troubleshooting and diagnostic equipment.
- Performed maintenance and repairs while prioritizing safety, reliability, and vehicle uptime.
- Maintained service documentation and repair records.
- Applied root-cause analysis to unfamiliar mechanical and electrical problems.

Industrial Maintenance / Robotics

The Scoular Company

January 2021 – March 2023

Mentor, MN

- Repaired conveyor systems and production equipment to minimize downtime.
- Performed robotics maintenance and programming.
- Prepared production equipment for startup, reducing reported startup time by approximately 15–20%.
- Diagnosed equipment failures and restored production under time constraints.

Appliance Repair Technician

MD Repair & Service

July 2012 – February 2017

Spring, TX

- Diagnosed and repaired residential and commercial appliances.
- Troubleshoot mechanical and electrical systems and identified root causes before performing repairs.
- Managed high-volume service calls across the Houston metropolitan area.
- Worked independently while maintaining customer communication and service quality.

PROJECTS

NEXIOS / NexiCode | Autonomous Software Engineering Platform

- Designed and developed an autonomous software-engineering platform intended to understand, modify, test, and maintain software projects through persistent project intelligence and controlled execution.
- Designed a multi-agent Mesh architecture for coordinating specialized engineering responsibilities.
- Developed persistent project intelligence for architecture discovery, code understanding, dependency relationships, and engineering context.
- Designed a governance model separating policy/intent, task execution, and system learning/evolution.
- Implemented runtime components for agent execution, task management, project observation, memory, and controlled filesystem interaction.
- Developed execution infrastructure capable of running tools, tests, code operations, and engineering workflows within controlled runtime boundaries.
- Implemented a unified Event Bus supporting structured event ownership, historical immutability, and event promotion rules.
- Integrated event streaming into agent execution and real-time reasoning interfaces.
- Implemented lifecycle management and standardized shutdown behavior for background observers and execution processes.
- Identified and resolved resource-leak issues involving orphaned background threads and subprocesses.
- Developed instrumentation and validation workflows to evaluate runtime performance and system behavior.
- Built a Mesh Validation Suite for testing the platform against multiple software project types.
- Developed verification workflows emphasizing verify-before-write behavior.
- Implemented testing and evidence-collection mechanisms designed to distinguish intended actions from verified results.
- Designed project discovery and code-intelligence systems for analyzing unfamiliar repositories.
- Developed architecture-memory and persistent-context concepts intended to allow an engineering system to retain knowledge of a project across sessions.
- Built local AI workflows using local LLM infrastructure.
- Integrated model inference with agent execution and engineering tools.
- Designed hybrid execution strategies separating fast operations from deeper reasoning workloads.
- Investigated inference, indexing, and runtime performance bottlenecks through direct instrumentation and benchmarking.

TECHNICAL SKILLS

Lua

C#

Python

Debugging

Modular Architecture

Object-Oriented Design
Event-Driven Systems
APIs
Automation
Software Testing
Runtime Systems
Developer Tools
System Architecture
Agent Architecture
Multi-Agent Systems
Local AI
Persistent Project Intelligence
Task Execution
Event-Driven Runtime Design
Validation Systems
Engineering Automation
Roblox Studio
Visual Studio
Git
WebSockets
Local AI/LLM Infrastructure
Filesystem Operations
Process Management
Persistent Storage
Event Bus Architecture
Runtime Lifecycle Management
Diagnostics
Technical Documentation
Root-cause analysis
Debugging unfamiliar systems
Rapid technical learning
Systems thinking
Modular software architecture
Independent development
Technical problem solving
Software validation
Documentation
Cross-disciplinary engineering
Translating real-world problems into technical solutions

CERTIFICATIONS

Diesel Mechanic Certification
Forklift / Skid Steer Certification
Driver's License

SOFTWARE & TECHNICAL PROJECT EXPERIENCE

Roblox / Game Systems: Modular gameplay architecture; Lua scripting; ModuleScript systems; DataStore workflows; Event-driven systems; UI frameworks; Procedural generation; Terrain and heightmap tooling; Developer utilities; Progression and interaction systems; Performance optimization.

Engineering & Automation: Developer tooling; Workflow automation; API integration; Runtime process management; Filesystem automation; Testing infrastructure; Technical diagnostics; Persistent data systems; Event-driven infrastructure.