Oluwaniifemi "NF" Emmanuel

+1 (415) 792-8813 | oluwaniifemi.emmanuel@uni.minerva.edu | github.com/nfemmanuel |
linkedin.com/in/oluwaniifemi-emmanuel

Education

Minerva University San Francisco, CA

Bachelor of Science, Computer Science & Artificial Intelligence *Sep 2023 – May 2027*

- Relevant Coursework: Data Structures & Algorithms, Formal Analysis, Complex Systems, Linear Algebra

Technical Projects

Tic-Tac-Toe with Minimax AI | *Kotlin, Android SDK, Coroutines, Game Theory*

- Engineered Android game implementing minimax algorithm with alpha-beta pruning for optimal AI gameplay
- Optimized performance using Kotlin coroutines for asynchronous calculations and non-blocking UI operations
- Architected modular game engine with separate logic/presentation layers and Intent-based navigation system

Static Site Generator | *Python, Jinja2, File I/O, GitHub Actions*

- Developed Python-based generator with Jinja2 templating engine and real-time file system monitoring
- Implemented two-repository architecture with CI/CD pipeline using GitHub Actions for automated deployments
- Designed template inheritance system using recursive directory traversal and dynamic content injection

UNO Card Game REST API | *Python, Flask, REST, OOP, JSON*

- Built full-stack UNO implementation with Flask REST API supporting complete game rules and state management
- Designed object-oriented architecture with Card, Deck, and GameState classes using OOP design patterns
- Implemented AI opponent system with configurable heuristic algorithms and JSON-based state serialization

Game Night - Progressive Web App | *React, JavaScript, Service Workers, Vercel*

- Engineered installable PWA with service worker implementation for offline functionality and native app experience
- Implemented drag-and-drop player ordering with React state management and game state persistence
- Deployed responsive web application to Vercel with automated builds and cross-platform compatibility

ProofScript - Turing-Complete DSL | *Python, Lark Parser, Language Design*

- Built Turing-complete DSL from scratch using Lark parser library with lexer, parser, and interpreter implementation
- Designed context-free grammar supporting variables, expressions, control flow, and algorithm specifications
- Implemented interpreter with input-based execution limits preventing infinite loops via step counting

Experience

Process Optimization Intern, UL Solutions May 2025 – Aug 2025

Fremont, CA

- Collaborated with engineering team to integrate Python automation scripts, reducing test data errors by 50% across product lines
- Analyzed workflow data using statistical methods and data visualization libraries to identify bottlenecks and optimize processes
- Redesigned operational workflows that reduced project timelines by 3 weeks, saving clients $XM+ through process optimization
- Implemented systematic validation protocols ensuring regulatory compliance across 200+ SAR technical test procedures

Product Builder, Google Academic Collaboration Sep 2023 – May 2024

San Francisco, CA

- Developed 4 product features for Google Photos platform, collaborating with 5-person cross-functional engineering team
- Built interactive prototypes using Figma and conducted quantitative analysis on user behavior from 50+ participant studies
- Improved user engagement by 40% through data-driven iteration cycles and A/B testing frameworks
- Analyzed engagement metrics and implemented testing methodologies to validate feature performance improvements

Cybersecurity Risk Assessment Analyst, Proficient Dynamics LLC Jun 2024 – Jul 2024

Cranston, RI

- Developed comprehensive risk assessment frameworks and technical documentation for enterprise security systems
- Created cybersecurity roadmaps across 20 security domains through systematic analysis and technical specifications
- Designed training materials translating complex security concepts into actionable technical recommendations

Technical Skills

Languages: Python, Kotlin, JavaScript, Java, C++, HTML/CSS, SQL
Frameworks & Tools: Flask, Jinja2, React, Android SDK, Git/GitHub, REST APIs, Coroutines, Service Workers
CS Fundamentals: Data Structures & Algorithms, OOP Design Patterns, System Architecture, Parser Design