Promise Eskor Ononokpono

promiseono@gmail.com | 1 (506)-429-2882 | Toronto, ON | linkedin.com/in/promise-eskor | github.com/pononokp

Portfolio: promiseono.vercel.app

EDUCATION

Bachelor of Science in Software Engineering, GPA: 4.0 01/2021 – 12/2025 | Fredericton, NB

University of New Brunswick

Relevant Coursework: Data structures and Algorithms, UI/UX Design Principles, Agile methodologies, Software Development

SKILLS

Programming Languages: JavaScript, TypeScript, Python, HTML, CSS, SQL, Java, C#

Developer Tools/ OS: Git, Docker, Azure, MS SQL, Firebase, Selenium, Linux, Windows, PostgreSQL, Jenkins

Software/Frameworks: Node, React.JS, Java Spring Boot, NestJS, ASP.NET, Flask, TailwindCSS

PROFESSIONAL EXPERIENCE

Software Engineer Intern 05/2024 – 12/2024 | Fredericton, NB

UNB Alloy Design Research Laboratory

- Rapidly adapted by independently learning and applying new technology to migrate a MATLAB proof of concept data analytics application into **Python**.
- Developed and integrated integrating **10+** material simulation and constitutive models (including **machine learning** models), improving alloy design research efficiency by reducing analysis time by over **30%**.
- Designed and implemented a user-friendly GUI with PySide and Plotly, enhancing data visualization and eliminating manual graph creation, saving researchers quantify time saved, **10+ hours per week**.
- Applied object-oriented programming for scalability and provided comprehensive documentation, including a detailed user guide, facilitating seamless adoption and updates by the research team.
- Packaged and deployed the Python application using PyInstaller, ensuring seamless installation and distribution to **20+** end-users.

Quality Assurance Intern 09/2022 – 04/2023 | Remote

Global Vision (GVD)

- Spearheaded the development of a system health checker application using **PowerShell** scripting, optimizing **20,000+** client environments and reducing validation bugs by **75%**.
- Led a key phase of in-house software validation for the **GVD 5.12.0** release, ensuring software reliability and a smooth release cycle.
- Designed, implemented, and maintained **150+** daily automated UI testing scripts (including integration and unit tests) using Java/Selenium and Jenkins, improving software reliability by reducing critical bugs by **90%**.
- Collaborated cross-functionally in an **Agile** remote environment, participating in Scrum and providing regular status reports.

PROJECTS

Flow

Python, FastAPI, WebSockets, PostgreSQL, Docker, React, TypeScript, asyncio

- Developing a real-time audio translation system (Python/FastAPI, WebSockets) streaming audio for near-simultaneous interpretation using **cloud APIs**, with a React web application for user interaction.
- Implemented a scalable architecture that allows for custom **machine learning** model integration and secure user **API** key management.
- Optimized audio chunking and streaming using **Python's asyncio**, reducing translation latency by **40%** and enhancing user experience.

Fun2Learn

React.JS, JavaScript, MS SQL, Node, Azure, HTML, CSS, REST, Docker, Firebase

- Collaborated to develop a gamified time management system using React.JS, Node, and **REST APIs**, enhancing student productivity for university students.
- Integrated **Firebase** authentication with Node and **Azure MS SQL**, enabling secure user sign-ups and adhering to COPPA guidelines.
- Implemented AI-generated positive reinforcement messages using **LLM API** endpoints based on performace statistics, enhancing user motivation.
- Containerized the application with Docker for scalable deployment on **Azure** Container Apps, supporting **1,000+** concurrent users.

SenZy

Unity, C#/.Net/.Net Core, Java/Spring Boot, REST, GitLab

- Collaborated in a team to develop a sensor management web application with a Unity front-end and Java/Spring Boot back-end, enabling real-time visualization and monitoring for **500** active sensors implementing role-based access control.
- Engineered a communication system using Java and **REST APIs** (backend) and **C#/.NET** (Unity front-end), allowing for quick real-time data streaming.
- Implemented complex logic for CRUD operations on sensors and enabled the creation of virtual sensors through data aggregation.
- Developed and optimized complex **MS SQL** queries and stored procedures for efficient sensor data management reducing latency for large-scale data retrival by almost **50%**.