Ahmed Elzaria

905-519-7303 | elzariaahmed@gmail.com | linkedin.com/in/ahmed-elzaria | github.com/ahmed-elzaria

Education

McMaster University Hamilton, ON

Bachelor of Engineering, Software Engineering CO-OP *Sept. 2022 – Present*

Relevant Coursework: Software Design I, OOP in Java, Data Structures and Algorithms, Software Engineering Practice, Computer Architecture, Digital Systems & interfacing, Engineering Design II, and Discrete Mathematics I & II

Experience

Software Engineering Intern - AI/NLP for Mobile Applications May 2024 – Present

McMaster's Centre for Software Certification (McSCert) *Hamilton, ON*

- Enhancing RESO, a suicide prevention app, with **NLP models** to improve user support accuracy and timeliness
- Leveraging **advanced libraries and frameworks**, including Hugging Face Transformers, Tokenizers, Datasets, TensorFlow, and Python data science libraries (NumPy, pandas), for **model training and fine-tuning**
- Developing a **benchmarking tool** to measure the performance of AI models for on-device usage
- **Deploying on-device AI to prioritize data privacy**, addressing mobile platform challenges for **iOS and Android** compared to cloud/server-based AI solutions

Software Engineering Intern - Compiler Optimization and Visualization May 2023 – Aug. 2023

McMaster's Centre for Software Certification (McSCert) *Hamilton, ON*

- Analyzed interactions and dependencies of **LLVM** optimization passes during compilation
- **Utilized pandas and NumPy** for data manipulation and analysis, and **scikit-learn** for clustering programs
- Created **transition graphs** using **NetworkX and Matplotlib**, aiding code optimization understanding
- Implemented a **pass microscope** tool to analyze and draw conclusions on specific pass interactions
- **Downsized Angha Project benchmark** from **1 million to 3600 C programs** for practical analysis
- Presented research at **McMaster Undergraduate Research Fair**

Projects

Maze Runner | *Java, Maven, JUnit, JSON, Javadoc, UML, Git* Mar. 2024 – Apr. 2024

- **Prioritized Agile methodologies**, incorporating iterative and incremental approaches
- **Implemented key software engineering principles** and patterns including **SOLID, GRASP, and GoF**, along with effective **version control** and **project management via GitHub Projects**
- Developed algorithms for pathfinding and **shortest path discovery**, including **Tremaux, Righthand, DFS, and BFS**, with features for path verification and algorithm comparison
- Designed the application for **seamless algorithm integration** and provided performance comparisons

Rescue Mission | *Java, Maven, JUnit, JSON, Javadoc, UML, Git* Jan. 2024 – Mar. 2024

- Followed **Agile development lifecycle with iterative approaches over 2 months**, focusing on **requirements gathering, MVP development, feedback integration, testing, and deployment**
- Applied key software engineering techniques: **SOLID principles, GRASP and GoF design patterns, object oriented design, encapsulation, information hiding, and unit testing**
- Developed a control program for a rescue drone to locate stranded individuals, and identify rescue point
- Optimized commands for efficient battery use and control integrity, ensuring effective command execution

Ihsan App | *Swift, SwiftUI, Xcode, Figma* May 2023 – Present

- Developed **front-end for an iOS app** offering prayer times and tracking, directions, and supplications
- Designed user interface in **Figma** and implemented it using **SwiftUI**
- Created an intuitive user experience for accessing daily supplications and religious content
- Collaborated with a team to **integrate front-end designs with backend functionalities**

Technical Skills

Languages: Java, Python, JavaScript, C, Swift, HTML/CSS, Elm, Assembly, Matlab, Verilog, Bash
Developer Tools and Frameworks: Git, GitHub, Apache Maven, JUnit, Linux, Unix, SwiftUI, VS Code, IntelliJ, Xcode, Jupyter Notebook, Google Colab, plantUML, SonarQube, PMD, Quasar, Capacitor, Figma
Libraries: pandas, NumPy, Matplotlib, NetworkX, Transformers, Tokenizers, Datasets, TensorFlow, scikit-learn