Hamza Zaman

(226) 751-5104 | hzaman.tech@gmail.com | m-hamza-zaman | mhzaman-cs | www.mhzaman.com

TECHNICAL SKILLS

Languages: JavaScript, Java, TypeScript, Python, C, C++, C#, SQL
Technologies: Pandas, TensorFlow, MongoDB, Express, React, Angular, Node, Spring Boot, ASP.NET Core

EDUCATION

University of Waterloo <div align="right">Waterloo, CA</div>
Bachelors of Computer Science, Minor in Statistics <div align="right">*2021 - 2025(Expected)*</div>

EXPERIENCE

ATS Corporation — Jan. 2023 – Apr. 2023
Software Developer Intern — *Cambridge, ON (Remote)*
- Developed illuminate ↗, which has **350+** business users, using .NET with C#, Angular with JavaScript and MSSQL
- Optimized database calls by decomposing **MSSQL** queries resulting in a **16%** decrease in display table load time
- Cleaned up code and implemented best practices resulting in a **38%** decrease in code smell issues on **SonarQube**
- Fixed bugs related to security and permissions issues, resulting in a **12%** reduction in vulnerability alerts

Achievers Inc — May 2022 – Aug. 2022
Software Developer Intern — *Toronto, ON (Remote)*
- Developed a **React** component library with 35+ standardized components to ease the web development process
- Removed external dependencies (ex. Material-UI) by rewriting components which decreased load time by **8%**
- Established data fetching control and reduced api calls by **23%** through transitioning from REST APIs to **Graphql**
- Manually configured **Storybook** with **Webpack5** by modifying internals, in order to migrate away from Gatsby
- Achieved the **WCAG 2.1** Accessibility standards by adding ARIA labels with proper tab indexing to components

SPARK — July 2021 – Apr. 2022
Software Developer — *Fremont, CA (Remote)*
- Developed a multiple choice quiz app used in 800+ student quizzes, utilizing **ASP.NET Core** and **MSSQL**
- Implemented authentication on the quiz platform and set up automated email responses confirming scores, etc.
- Reduced load time by **17%** by utilizing CDNs, minifying code, removing unnecessary items and plugins
- Concurrently developed a front-end educational platform using **React** and **Bootstrap** to teach 500+ children
- Standardized the design language with a mobile-first approach which improved mobile user satisfaction by **29%**

CrowdDoing — May 2020 – Aug. 2020
Data Scientist Intern — *San Francisco, CA (Remote)*
- Collected data for 65+ herbs by scraping multiple sources through **Python** with **Scrapy** and **Beautiful Soup**
- Processed and cleaned 35+ data sources into a singular, coherent dataset using **Pandas** and **NumPy**
- Leveraged indexing and partitioning in **PostgreSQL** to improve query performance by **31%** on average
- Wrote a recommendation system for herbs and medicinal foods using the TF ranking algorithm in **TensorFlow**
- Applied cluster analysis techniques such as **K-means** to classify 100+ items into their nutrient categories

PROJECTS

Forex ↗ | *MongoDB, Express, React, Node, Firebase, Tailwind CSS, JWT*
- Used a **MongoDB** database to store login credentials with an **Express.js** server to handle user authentication
- Utilized Fixer.io's API to display exchange rates for **170+** currencies as recommendations on the dashboard
- Incorporated **JWT** signature verification to maintain authorized user's access to the API and chat
- **React**-based front-end is focused heavily on **UX/UI** by using frameworks like **Tailwind CSS** and **PostCSS**

Citadel Data Open ↗ | *Python, Plotly, Seaborn, Pandas*
- Co-authored a report which examined how investments in businesses and education affect traffic in major cities
- Cleaned, organized, and structured multiple provided and researched data sets with **1 Million+** entries
- Applied analysis techniques to derive statistically significant findings on traffic congestion patterns
- Created graphs using the **Python** libraries **Plotly** and **Seaborn**, while utilizing **Pandas** to organize the data

OpenCare ↗ | *Spring Boot, XAMPP, Maven, MySQL, REST API, React, Vite*
- Implemented the real-time Q&A using **STOMP** over a WebSocket protocol with **SockJS** on the client-side
- Built a back-end API using **Spring Boot** with **Java** and **Maven** to facilitate thread creation, voting on replies, etc.
- Implemented a database solution for storing and viewing previous Q&A threads using **MySQL** and **XAMPP**
- Collaborated on the development of a user-friendly front-end using **React** with **JavaScript**, and **Vite** (Demo ↗)