

Muhammad Zaman

(226) 751-5104 | mhzaman.tech@gmail.com | www.linkedin.com/in/muhammad-hamza-zaman/ | github.com/mhzaman-cs

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

Bachelor of Computer Science, Minor in Statistics

Waterloo, CA

2021 - 2025 (Expected)

EXPERIENCE

Achievers Inc

Software Developer Intern

May 2022 – Aug. 2022

Toronto, ON (Remote)

- Developed a **React** component library with 25+ standardized components to ease web development process
- Removed external dependencies(ex. Material-UI) by rewriting components which decreased load time by **8%**
- Created independent components such as Icon and Checkbox Group, and had written documentation for them
- Migrated documentation from Gatsby to **Storybook.js** with **Webpack5**, by manually configuring SCSS, SVGs, aliases and storybook internals (docs, canvases, controls) to create an intuitive playground for testing components
- Achieved the **WCAG 2.1** Accessibility standards by adding ARIA labels with proper tab indexing to components
- Met and exceeded the **80%** coverage threshold of **JavaScript** unit testing by writing **50+** React unit tests

SPARK

Software Developer

July 2021 – April 2022

Fremont, CA (Remote)

- Created a multiple choice quiz section using **ASP.NET Core** as the back-end and an **SQL database** for storing the questions which were used in 800+ quizzes taken by students
- Authenticated 200+ student accounts to access quizzes using **ASP.NET Core** with **C#** to validate credentials
- Set up automated emails which sent 2000+ emails to student accounts confirming quiz results, registration, etc.
- Developed a front-end educational platform using **React** and **Bootstrap** to teach 500+ children
- Reduced load time by **17%** by code-splitting, utilizing CDNs, minifying code, and removing unnecessary plugins
- Standardized design outputs with a mobile-first approach which increased mobile user's satisfaction rates by **29%**

CrowdDoing


Data Scientist Intern

May 2020 – Aug. 2020

San Francisco, CA (Remote)


- Collected data for 65+ herbs from different sources including the National Library of Medicine using data crawling techniques through **Python** with libraries such as **Scrapy** and **Beautiful Soup**
- Processed 35+ unstructured data sets through libraries such as **Pandas** and **NumPy** for data standardization
- Constructed a recommender system for herbs and medicinal foods using **TensorFlow** which served with ScaNN for retrieval, ranked items with TF ranking and leveraged multitask learning to recommend the top 5 items for a user
- Applied cluster analysis techniques such as **K-means clustering** to classify 100+ items into nutrient categories

PROJECTS

Forex  | *MongoDB, Express, React, Node, Firebase, Tailwind CSS, JWT*

December 2021

- Full-stack **MERN** app featuring a discussion form based on **Firebase** that allows for P2P Currency Exchange
- Uses a **MongoDB** database to store login credentials with an **Express.js** server to handle user authentication
- Utilizes Fixer.io's API to display exchange rates for **170** currencies as recommendations on the dashboard
- Keeps user authenticated in order to retain access to the API and chat by using **JWT** signature verification
- React**-based front-end is focused heavily on **UX/UI** by using frameworks like **Tailwind CSS** and **PostCSS**

Employee Manager  | *Spring Boot, Angular, MySQL, REST API, Bootstrap*

June 2022

- Full-stack CRUD application which allows for managers to store and edit information about employees
- Utilizes **Spring Boot** in **Java** on the back end to handle the logic related to managing employee data which is stored in the **MySQL** database, along with **Angular** in **TypeScript** and **BootStrap** in the front end for the UI
- REST controller is to make a **REST API** that uses HTTP requests to access and use data from the server