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 Waterloo, CA
Bachelor of Computer Science, Minor in Statistics *2021 - 2025(Expected)*

Experience

Achievers Inc May 2022 – Aug. 2022
Software Developer Intern *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

SPARK July 2021 – April 2022
Software Developer *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 May 2020 – Aug. 2020
Data Scientist Intern *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

Citadel Data Open | *Python, Plotly, Seaborn, Pandas* March 2022

- Wrote a report in a team of 2 about how investments in businesses and education affect traffic in major American cities and provided recommendations on how congestion can be reduced through investments in these areas
- Cleaned and structured multiple provided and external data sets with **1000000+** entries and used that data to come to statically significant conclusions about congestions in New York, NY, Austin, TX, and Washington, DC
- Graphs are generated using the **Python** libraries **Plotly** and **Seaborn**, and the data is organized using **Pandas**

Stockify | *Python, Plotly, Seaborn, Pandas* May 2021

- Created a stock visualizer capable of displaying an interactive graph of any of the S&P 500 stocks with different comparisons such as Opening value, Daily High, Daily Low Price, Closing value, or Volume traded compared against the date from 2013-2018 based on a Kaggle data set
- The graphs are generated using the **Python** library **Plotly** and the data is organized using **Pandas**

Amazon Reviews Scraper | *Python, Scrapy* July 2021

- Leveraged **Python** and **Scrapy** to scrape **10k+** customer reviews from different products based on ASIN number
- Utilized **Scrapy**'s built-in boilerplate and implemented the scraper components such as the HTML parser for scraping content on pages and the initiator which loops through the different products using **OOP principles**
- Added cool down and opened tabs in-browser to prevent the program from getting caught in Amazon's CAPTCHA