Richard Zhang

Waterloo, Ontario
r29zhang@uwaterloo.ca in/rz2004 @notzree rzhang.vercel.app

Education

University of Waterloo **Sep. 2022 – Present**
Bachelors of Applied Science in Systems Design Engineering- ***3.85 GPA*** *Waterloo, Ontario*

- Digital Computation (C++) — Linear Algebra — Introduction to Design — Graphics (Solidworks)

Technical Skills

Languages: Javascript, Typescript, Python, C#, Java, HTML/CSS, C++, MATLAB
Frameworks and libraries: Next.JS, React.JS, Prisma, PostgreSQL, SQLServer, Supabase, Tailwind, ASP.NET, Pytorch, NumPy
Tools: Git, SQL, Vercel, Solidworks, Confluence, Excel, SQL Server Mangement Studio

Experience

EnergyIntell **January 2023 – Present**
Fullstack Software Engineer *Richmond Hill, Ontario*

- Designed a reverse ETL pipeline enabling clients to access **50+ million** rows of archived data from **SQLSever** database using **.NET**, **MediatR**, and **Entity Framework**. Lowered job time from **2 days** to under **1 minute**
- Implemented a file storage page to enable clients to securely access their yearly data using **C#**, **.NET**, **JQuery Ajax**, and **JQWidgets**

University of Waterloo Alternative Fuels Team **October 2022 – Present**
Front-end-Developer *Waterloo, Ontario*

- Utilized **React.js** to develop University of Waterloo Alternative Fuel Team's custom user friendly website.
- Researched differences between multiple lateral controls models using **SIMULINK** for the Advanced Driver Assistance System (ADAS).
- Utilized **MATLAB** and **SIMULINK** models to develop constant radius maneuver algorithm for the 2023 Cadillac Lyriq

Projects

Zeno - Journalling Web App

- Developed a fullstack journalling app with user authentication and CRUD features using **Next.JS**
- Implemented **Supabase Auth** with multiple login providers with **Postgres** row-level security policies for security

write:here (Hackville 2023 winner)

- Designed and deployed a fullstack web application that process hand written letters into emails.
- Spearheaded development of the backend with **Next.JS**, using HTTP fetch requests that implement Google Vision API as well as **co:here's** NLP to spellcheck and format parsed text.

Bigram Language Model

- Utilized **Pytorch** to create a **Multilayer Perceptron** to implement a Bigram Language Model, which generated unique names
- Trained the model using **Stochastic Gradient Descent** with Pytorch's autograd library on a training data-set of 25K names, implementing **minibatching** to expedite the training process.

Home - Pet adoption hub

- Designed a full-stack website that allows users to adopt, buy/sell pets using **Prisma** for data modelling, **Supabase** for the database, and **Next.JS**
- Created the backend API's to allow users to post and mange listings on the website.
- Integrated **Next-Auth** in database stored sessions to protect backend API routes and secure CRUD features.

Student Design teams / Clubs

- University of Waterloo Alternative Fuels team
- University of Waterloo Muay Thai