Matthew McAuley

(516)-946-0870 | mwm223@cornell.edu | Linkedin | GitHub

Education

Cornell University Aug. 2022 – Dec. 2025

Bachelor of Science in Computer Science, Minor in AI *Ithaca, NY*

GPA: 4.16/4.3

Coursework: OOP & Data Structures, Functional Programming, Discrete Structures, Algorithms, Machine Learning, Database Systems, Digital Logic, Linear Algebra, Embedded Systems, Language & Information, Probability & Statistics

Technical Skills

Languages: Python, Javascript, Typescript, C#, Java, OCaml, C, Assembly, HTML, CSS, SQL
Tools: Git, GitHub, Docker, AWS, OCI, Redis, Snowflake, PostgreSQL, SQLite, Postman, npm, conda, Verilog
Libraries/Frameworks: .NET, NUnit, React, Expo, Flask, SQLAlchemy, NumPy, Pandas, PyTorch, Selenium

Experience

Software Engineer Intern June 2024 – August 2024

Octus (Formerly Reorg) *Remote*

- Developed C# API for retrieving and organizing CLO market data into multi-level tables each with unique insights
- Optimized data retrieval with Redis caching, reducing page load times from ∼5s to under 75ms after initial request
- Created NUnit integration tests with complete coverage, ensuring proper functionality after codebase changes
- Accessed Snowflake loan DB with 3.2M+ rows, adding relevant entries to objects defined by Postgres relationships
- Launched website for **Credit Cloud** platform, used by over 30,000 legal and financial professionals worldwide

Software Developer August 2023 – May 2024

Cornell Cup Robotics *Ithaca, NY*

- Consulted major robotics company to design low-cost, educational robots for use across 2,000+ universities globally
- Designed website with Flask backend allowing for wireless connection and programming of onboard microcontroller
- Implemented React.js GUI for intuitive pairing, movement, blockly coding, script writing, and location graphing
- Built overhead vision system using April Tags to track robot ID, location, and orientation relative to other objects

Projects

Exercise Engine

- Full-stack **website** allowing ad-hoc or drop-down querying of exercises and returning the most similar results
- Created Flask API with SVD text mining and cosine similarity to measure relationships between descriptions
- Programmed HTML, CSS, and JS frontend allowing for API requests, applying filters, and neatly displaying results
- Used Selenium to web-scrape JSON database with exercise descriptions, muscle groups, and YouTube tutorials
- Containerized the application using Docker/Docker Hub for deployment on an AWS EC2 instance running Ubuntu

GymBuddy

- Fitness companion mobile app with rest timer, set counter, exercise display, and custom workout plan creation
- Developed UI using Expo, React Native, TypeScript, and Tailwind CSS, with Expo-Router for file-based navigation
- Implemented local storage of custom workout plans and exercises using SQLite database, enabling offline access
- Integrated Spotify Web API with OAuth2 Code Flow for user authentication, providing intuitive media control

To-Do Manager

- Web-app for managing projects and their respective 'todos', each containing a title, description, and due date
- Developed a React frontend with Typescript, utilizing emotion/styled CSS in TS for better code organization
- Built RESTful Flask API that interacts with a SQLAlchemy relational database for project/todo persistence

Caml Capital

- Terminal recreation of the board game 'Monopoly' done almost entirely with functional programming in OCaml
- Worked in a 3-person team with weekly sprints and multiple prototype stages that offered increasing client value
- Utilized compilation units, higher order functions, OUnit test suites, version control, and rigorous documentation