Harry Gosse

46W 21st St Huntington Station, NY 11746 · gosseh@umich.edu · (646)-940-0945

Education

University of Michigan College of Engineering   Ann Arbor, MI
Computer Science *GPA: 3.31*   Sep 2018 - May 2021
Relevant Coursework: Web Systems (EECS 485) · Introduction to Operating Systems (EECS 482) ·
User Interface Development (EECS 493) · Introduction to Computer Security (EECS 388) ·
Foundations of Computer Science (EECS 376) · Data Structures and Algorithms (EECS 281) ·
Introduction to Computer Organization (EECS 370) · Discrete Math (EECS 203)

Bronx High School of Science   Bronx, New York
High School *GPA: 4.00*   Sep 2014 - Jun 2018

Work Experience

Amazon Web Services   Seattle, Washington
Software Engineer   August 2021 - Present

- CloudFormation Team

Juniper Networks   Sunnyvale, California
Software Engineer Intern   May 2021 - August 2021

- Connected a ledger system to record configuration changes to the packet forwarding engine. This enabled live updates to the configuration of the router to be displayed for PTX and MX router models.

- Implemented a tagging system for nodes in the Advanced Forwarding Toolkit sandbox.

MTA New York City Transit   Queens, New York
Database Developer Intern   June 2020 - Nov 2020

- Tasked with design, development, testing, implementation and optimization of a database which will be used to track and monitor collective bargaining, contract administration, employee availability and forming out of work initiatives within the MTA Office of Labor Relations.

- Implemented a Python web crawler which scraped news articles and extracted key information to generate a daily digest to judge the public perception of transit services.

Project Experience

Smart Rainwater Barrel *Python*
Designed a smart storm water irrigation barrel calculating optimal amounts of water to release from weather and sensor data as a solution to water scarcity.

Traveling Salesperson *C++*
Implemented a program that calculates the optimal tour for the TSP. The program can also calculate a close to optimal tour in significantly less time using MST approximation.

Little Computer 2000 *C*
Created a functioning compiler and linker to turn multiple assembly programs into executable files and a simulated environment in which those files were interpreted and run in a fully working pipeline with a simulated cache.

Client/Server Side Dynamic Website *Python, Javascript, SQLite, React*
Built a client application using Python, Flask, JavaScript, React and asynchronous programming that makes AJAX calls to REST API to emulate Instagram with attributes such as infinite scroll, dynamic like and comment, and session control for each user.

Threading Library *C++*
Created a threading library that balances workloads across multiple CPUs. Included implementation of basic multithreaded interfaces like threads, mutexes, and condition variables.

Sports App *Javascript, Firebase, Vue*
Created a website that allowed only University of Michigan students to find other students that want to participate in group activities. This was done by having the sign-up page only accept emails with umich.edu domain.

Skills

Technical:	C++, Python, Verilog, Go, HTML, Javascript, Flask, React, Vue, SQL, Jinja, Git
Environments:	Windows, Linux, Visual Studio