Brett Wiseman

bwisema3@nd.edu | 708.927.0554 | 15630 Lorel Ave, Oak Forest, IL, 60452
linkedin.com/in/brett-wiseman77 | wiseman77.com

EDUCATION

University of Notre Dame, College of Engineering **Notre Dame, IN**
Bachelor of Computer Science in Engineering **|** GPA: 3.880 **May 2023**
Honors: College of Engineering Dean's List, Blackhawk Alumni Scholarship recipient and member
Relevant Courses: Compilers, Theory of Computing, Cryptography, Operating Systems, Computer Networks, Computer Architecture, Data Structures, Systems Programming, Logic Design, Embedded Systems, Circuits/Electrical Engineering Lab

SKILLS

Programing Languages: C, C++, Python, Bash/sh, ARM/X86, Matlab, HTML/CSS, Java, Javascript, Clojure, Solidworks
Frameworks: Linux Terminal, Git/Github, Vim, macOS, Windows, IBM Toolkit, Flex, Bison, Verilog, Arduino, WireShark, Django

EXPERIENCE

Visa Inc. **Austin, TX**
Software Engineer Intern - *Payment Product Development (PPD)* **Summer 2022**
- Successfully and quickly learned new software stack to complete beginner project of switch clean-ups
- Collaborated with interns and full-time employees at other office locations to perform weekly meetings virtually to stay updated on project progress
- Thrived in a hybrid work environment to complete the main project which was to rewrite and improve a JSON/XML to UMF parser using C++ inside IMB Toolkit, including designing and implementing a new algorithm
- Performed testing using VM3 interface to start up test systems and load in current complied project code to run regression tests

University of Notre Dame **Notre Dame, IN**
Undergraduate Teaching Assistant - *Data Structures, Systems Programming, Operating Systems* **Fall 2021 - Present**
- Help students develop a better understanding of relevant data structures as well as C++ classes, git / GitHub submissions, and memory management
- Assist students in learning concepts such as shell scripting, data processing with python, and system calls in C
- Manage a group of students as they developed software by hosting code reviews and giving feedback throughout the project

Undergraduate Research - *Cooperative Computing Lab (CCL)* **Fall 2021 -Fall 2022**
- Software Developer working on high-level python abstractions for parallel computing and distributed systems software to match results running on a local machine
- Contributed to Work Queue software used by thousands of users and at other universities such as the University of Wisconsin
- Attended weekly meetings to present progress, and discuss possible ways of improvement

PROJECTS

Bminor Compiler **Fall 2022**
- Constructed a compiler for a C-like language, starting from scanning tokens with flex, moving to parsing a grammar with bison, followed by building and type checking an AST and ending with generating X86 assembly
- Gained experience in software development practice by building distinct parts of compilers in stages and learning better code organization practices

Personal Website **Summer 2022**
- Built a personal website using HTML/CSS, using sass for CSS as well as some javascript
- Set up a Linux VPS to host a website and be an email server for my personal email while also learning about Linux OS

Multi-Player Competitive Worldle **Spring 2022**
- Write Server and Client applications to play a multiplayer game of wordle - have a server run the game with certain game options, then have clients join and have a round and point-based guessing game
- Complied using Thread and Socket programming to have game and clients run, as well as design API using well-defined JSON fields such that anyone could write their own version to join or host games
- Implemented rigorous error checking to make sure messages were valid, that the host wouldn't crash due to unexpected behavior, and keep the game fair

Personal Calendar Application **Spring 2022**
- Developed a Client and Server application that stores information about events on the server based on requests from the client
- Designed how messages between Client and Server are structured and how data is stored persistently on the Server-side
- Used multiple threads to handle multiple clients concurrently, as well as mutex locks for synchronization

Cisco Systems - Open MPI **Spring 2021**
- Contributed to open source project (Open MPI) by fixing issues in Github Action CI
- Implemented code to check cherry-picked commits to ensure merge is allowed only after merge for all parent commits
- Organized team meetings and led team discussions