

AUSTIN HOCHHALTER

1301 N Olde Wagon Rd
Sioux Falls, SD 57110

(605) 251-4757
austinhochhalter@gmail.com

<https://github.com/ahochha>
<https://www.linkedin.com/in/ahochha/>

EDUCATION

South Dakota State University, Brookings, SD

Bachelor of Science in Computer Science, Minor in Software Engineering (Cumulative GPA: 3.558)

- **Honors:** Dean's List (5 semesters)
- **Activities:** Member and Vice President (1 semester) of SDSU's Association for Computing Machinery club, CS tutoring

SKILLS AND QUALIFICATIONS

- Languages: (*Proficient*) - **C/C++, C#, Javascript/Typescript, SQL, HTML/CSS** (*Familiar*) - **Java, Python, Assembly, Pascal, Ada**
- Knowledgeable in **Git** and **Azure DevOps** source control.
- Well-practiced with agile development processes, specifically **SAFe (Scaled Agile Framework)**.
- Adept at creating/modifying **SQL** scripts and stored procedures.
- Skilled with frameworks/technologies such as **ASP.NET Core, Angular, Ionic, Bootstrap, Xamarin, and Firebase**.
- Working knowledge on cloud computing services such as **Microsoft Azure**.
- Experienced with using **Unity** to code 3D games.

WORK EXPERIENCE

Daktronics, Brookings, SD

Software Engineer, September 2018 to May 2020

- Developed and maintained production level web applications as a student on IT Software Development teams.
- Worked with other developers to design user interfaces, architect APIs, and write stored procedures to improve HR/Personnel processes.
- Implemented the validation architecture for a time off management application that was in development. The architecture was designed to scale for future needs and included validation such as matching submitted hours to existing company rules.
- Collaborated with users by holding review sessions to ensure application expectations would match reality.
- Gained agile development experience by participating in bi-weekly sprint planning and review meetings.
- Leveraged knowledge in **ASP.NET Core, Javascript/Typescript, SQL, HTML, CSS, Visual Studio, SAFe, and Web Development**.

Rise United Media, Brookings, SD

Software Engineer, September 2019 to May 2020

- Helped Rise United develop their mobile focused social media application as a senior design capstone project.
- Designed and implemented a user interface that allows the user to listen to audio content from other users.
- Implemented save and update functionalities to push audio recordings and photos to Azure Blob Storage and other relevant data to a SQL server database.
- Leveraged knowledge in **ASP.NET Core, XAML, Visual Studio, Microsoft Azure, Xamarin, and Mobile Development**.

South Dakota State University, Brookings, SD

Teacher's Assistant, August 2019 to December 2019

- Helped computer science freshmen learn how to code in SDSU's Intro to Computer Science Lab using Unity to implement a basic game. Involved helping students with their code, but also general college and computer science questions as well.
- Leveraged knowledge in **C#, Unity, and Visual Studio**.

PROJECTS

Visit <https://ahochha.github.io/portfolio> for more projects.

Java Compiler, January 2020 to May 2020

- The compiler implements common grammar used in Java such as classes, functions, simple expressions, and I/O.
- It parses one character at a time from a source Java program. Detailed error messages are output for incorrect Java syntax.
- Three address code is generated from the Java code and output to a TAC file. Once TAC generation is complete, it is parsed to generate an Assembly file containing Intel x86 instructions. The Assembly file can then be run in an x86 MSDOS environment.
- Technologies Used: **C#, Java, Assembly**

Monte Carlo Method, December 2019

- Program estimates the value of PI using the monte carlo method.
- It uses OpenMP to calculate PI with multiple parallel threads. The number of threads and samples are provided by the user on the command line. The amount of time to complete the process is tracked and the results are displayed.
- Technologies Used: **C, OpenMP**

Assembler, September 2018 to November 2018

- Involved understanding assembly language and low-level programming.
- The assembler consists of two passes. Pass one parses symbols from an assembly file. After validation, the symbols are loaded into a symbol table and used to track the program counter. Pass two generates object code and outputs an object file.
- Technologies Used: **C++, Assembly**