Porter Matthew Laartz

plaartz@gmail.com

EDUCATION

University of Wisconsin, Madison
September 2021 - Present
Bachelor of Science in Computer Science & Data Science
3.8 GPA - 1490 SAT (800 Math / 690 Reading)

WORK EXPERIENCE

Freelance Programming
August 2021- Present

- Completed various projects for customers in languages such as Java and Python while keeping an open line of communication regarding the needs of the customer and necessary timeframes.

PROJECTS

Resume Site

- Created a resume website to showcase my skills as a software engineer outside of a resume
- Used React.js to create the dynamic site as well as Firebase to store project information and make it easily editable.
- plaartz.github.io

3D games

- Combined C++ skills with knowledge of Unity software to create playable games involving remaking classic 2D games in 3D that can be played both through VR and through the desktop computer.
- Used object-oriented programming to gain a deeper understanding of C++ and create larger-scale projects focused on being interactive for users.

Python

- Used APIs to create functional user interfaces for sites such as Spotify.
- Used raspberry pi to create a functional motion/proximity detection system, as well as network logging.

SKILLS

Languages: Java, Python, Javascript, HTML/CSS, R, Assembly
Frameworks: React.js, Node.js, JUnit, Firebase
Developer Tools: Git, Docker, Google Cloud Platform, Visual Studio Code, IntelliJ, Eclipse
Libraries: pandas, NumPy, ggplot2

COURSEWORK

Programming I-III - Learning and applying the fundamentals of Java. Working with different data types/ structures in collaborative settings to build programming projects and implement algorithms.

Data Science Modeling I - Learning reproducible data management, modeling, and analysis through hands-on approach using the R programming Language.

Discrete Mathematics - Basic concepts of logic, sets, partial order, and other relations. Basic concepts of mathematics with a focus on discrete structures: bits, strings, trees, and graphs. Learned mathematical induction and recursion, Invariants and algorithmic correctness, and asymptotic growth analysis.

Computer Engineering I - Logic components built with transistors, rudimentary Boolean algebra, basic combinational logic design, basic synchronous sequential logic design, basic computer organization and design, introductory machine- and assembly-language programming.