Charles Quigley

Address: 15418 Grass Cay Ct. Corpus Christi Texas, 78418
Phone Number: 361-251-7222
Email: cmq2st@gmail.com

About Me

Hello, I am a recent graduate with a great deal of experience working in teams. I have extensive knowledge of various programming languages, best practices, and methodologies. I am seeking to leverage my coding experience and enthusiasm towards achieving goals in team-based efforts and am eager to learn new skills as well as build upon the skills I already know.

Education

Texas A&M University-Corpus Christi

- **Bachelor of Science in Computer Science with a Systems Programming Concentration**
- **Overall GPA:** 3.83
- **Graduated:** December 2021

Relevant Courses

Data Structures Algorithms Operating Systems Internet Programming Computer Graphics

Skills

Languages

HTML, CSS, JavaScript, SQL, C#, C, C++, Java, Python, R

Environments

Visual Studio 2019, Visual Studio Code, Intellij IDEA, Code::Blocks, Spyder, RStudio, MATLAB

Honors

Placed on the Dean's List for Texas A&M University-Corpus Christi's College of Science and Engineering for the 2020 Spring and Fall semesters and the 2021 Spring semester.

Projects

Full Stack Web Development

In a team of 2, my partner and I created a website for a local construction company using the ASP.NET framework in Visual Studio. This website was built using HTML, CSS, JavaScript, and C#, and utilizes an MSSQL database to store information regarding the company's projects. This website uses an API that allows employees of the company to clock-in and clock-out of work and an API that displays a new Bible verse on the Home page every day to show the company's Christian values. Also, an interface is implemented that allows admins to add new projects as well as edit and delete old projects featured on the website. We are still finalizing hosting details with the company, but to view the website, please use this temporary link: http://charlieq361-001-site1.itempurl.com/

K-Nearest Neighbors Algorithm

An individually completed project coded in Python. This algorithm uses Euclidean distances between features and a Max-Heap data structure to predict class label information. For testing purposes, the class labels of roughly 15,000 testing instances were predicted based on a training set consisting of roughly 30,000 instances. The algorithm accurately predicted 82% of all class labels based on this data.