**Drake Lopez**

**C# Software Developer**

drakelopez.dev@gmail.com | (256) 701-7759

[Drake](https://www.linkedin.com/in/drake-lopez-5a407b261/) [Lopez](https://www.linkedin.com/in/drake-lopez-5a407b261/) [|](https://www.linkedin.com/in/drake-lopez-5a407b261/) [LinkedIn](https://www.linkedin.com/in/drake-lopez-5a407b261/) | [DrakeLopez27](https://github.com/DrakeLopez27) [(github.com)](https://github.com/DrakeLopez27)

# Summary

I am an experienced software developer with expertise in C#, HTML, SQL, JavaScript, JSON, and various other languages. I have a passion for learning how to create innovative and efficient solutions. I strive to build a strong background in software development and a comprehensive understanding of various programming languages and technologies.

# Skills

C# ASP.NET MVC Architecture .NET Framework NuGet Package Manager

Web Services API’s MySQL SQL Server Dapper

XUnit Git GitHub HTML Razor Pages

CSS JavaScript Node.JS Visual Studio Code Visual Studio 2022

Zoom

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

# Education

**TrueCoders Academy**

Graduated June 2023

Certification in Software Development, Web Development, .Net Full Stack

**IBM Intro to Hardware and Operating Systems**

Completed through Coursera January 2023

**IBM Intro to Technical Support**

Completed through Coursera January 2023

# Technical Projects

**MovieQs – Movie database site for information regarding movies (On-going project)**

**C# | .Net Framework |MVC (.NET Core) | JSON |API |Nuget Packages | Github**

* Uses TMDB API to call for movie lists by search or by popularity/trending and includes an Id, popularity rating, and a brief overview of each movie.
* Efficient updating to get latest results for the popularity or trending movies page.
* Allows users to click for a random movie to watch, or allows users to search by title.
* This application is UNFINISHED – anything has the possibility to change

**CRUD ASP.Net MVC Inventory Management Application**

**ASP.Net MVC | .Net Framework | MySQL | C# | HTML | CSS | JavaScript | jQuery | Razor**

**View | JSON | Dapper | MySQL.Data |**

* Used model, view, controller architecture to build a dynamic website for database management
* Created 2 controllers, one for the home page, and one for the product page to take input from the user from the view
* Created 6 views to display information gather from the models and database, chosen by the decision of the user of how to manipulate the data
* Created 5 models, each with their own custom methods of how they connect to the database, and the queries of those databases based off the input from the user
* Implemented CRUD methods with C# to create, read, update, and delete products from the database, as well as input specific information about a product when creating or updating a product
* Implemented a sleek, modern theme using HTML, CSS (Bootstrap Library), and JavaScript for the website for ease-of-use and understanding of displayed information

**Geolocation Distance Compiler (TacoParser)**

**C# | .Net Framework |Console APP (.NET Core) | GeoCordinate library | XUnit | CSV |**

* Implemented the GeoCordinate package to read and organize coordinate data according to distance.
* Implemented an Interface to apply attributes to a child class that eventually becomes the main object that holds all the attributes for every Taco Bell location
* Created a custom value type using a structure named “Point” that holds a double value type for latitude and longitude values to pass into the ITrackable interface
* Created nested for loops to enumerate over each location points in a parsed array while comparing them to all other arrays’ locations, then replaces the value in a variable that holds the location of the furthest distance from the origin location within the first loop
* Used string interpolation to present the two furthest locations including city name, longitude, and latitude