

Mario Torres

Chicago, IL • email: martorres@protonmail.com • GitHub: <https://github.com/mariotee>

PROFILE

Well-rounded software engineering student eager to get into the field and build robust web-apps. I enjoy both front-end and back-end development. Aside from programming, I am an avid lover of most genres of music as well as music production. I am also fluent in Spanish.

SKILLS

JavaScript • GraphQL • HTML5/CSS3 • C# • Python • SQL • NodeJS • Amazon Web Services • Azure • .NET Entity Framework • .NET Web API 2 • MongoDB • Mocha • React • Jest • Babel • Webpack • npm • git • Test-Driven Development • Mobile-First Web Design

EDUCATION

BSE Computer Science & Engineering, *Minor in Mathematics*

The University of Iowa

GPA: 3.23

05/2018 - 05/2019

WORK EXPERIENCE

Software Engineering Intern

IDx Technologies, Inc.

05/2018 - present

- Developed class libraries and Windows service for back-end of internal document automation application
- Drafted a demo web application, complete with RESTful API
- Designed a mock database and scripted possible migrations in SQL

Audio/Visual Assistant

University of Iowa, Carver College of Medicine

09/2016 - 05/2019

- Prepared equipment for high-profile medical lectures
- Performed routine checkups and troubleshooted issues
- Trained new hires for excellent customer service

PERSONAL PROJECTS

(source code made public)

Enigma Machine Web

React, Object-Oriented JavaScript

Redesign of a C++ console application; the classes are modelled after the Enigma M3 encryption machine that was used by the Germans during WWII

Temperature Monitor

GUI: React, axios, AWS Lambda/API Gateway, Twilio

Hardware/Firmware: Raspberry Pi Zero W, DS18B20 Temperature Probe, asynchronous I/O, node.js, Google Firebase

School project that involved reading temperature data from a probe and displaying it on a web app GUI; also alerted (via text) when temperatures passed the set min and max thresholds

My Dash

React, Redux, Leaflet Maps, axios, Express, Mongoose/MongoDB

Practice making an admin dashboard app with React/Redux on the client and Express on the backend. Data is stored in MongoDB and validated with the mongoose ORM. Styled with Material UI