CameronGreen

Cameron@CameronGreens.com
(214) 642-2565

Professional Summary:

Seasoned Software engineer & Bachelor's Degree candidate studying Computer Science at Sam Houston State university. Eagerly pursuing an opportunity with an organization that fosters both professional and technical growth. Keen to learn new and robust technologies in front-end and responsive software development. Strong passion for creating robust software to solve complex problems. Experienced in architecting, designing and developing software from scratch for both corporate and personal projects

Technical Summary

Languages	Frameworks	Tools	Databases
JavaScript	React	GIT	SQLite
HTML5	Angular	NPM	MongoDB
CSS3	Ionic Framework	APT	Redis
Python	Node.js	SCSS	PostgreSQL
Java	Socket.io	Play Store	
C++	Gatsby.js	Babel	
	Android	ESLint	
	GraphQL	Webpack	

Education

Sam Houston State University Huntsville, TX
Bachelor's Degree Candidate in Software Engineering (08/2018)

Blinn College Bryan, TX
Associate's of in Applied Sciences in Computer Engineering (08/2015)

Detailed Summary

Americans for Responsible Insurance – Technical Director, Chief Developer (May 2017 – September 2017)
Responsible for building and developing an in-house cross-platform application for ARI from scratch using Angular 2, Ionic 2, Typescript, Node.js, Express, MongoDB, and Redis. This app is a full stack mobile application that allows ARI employees to gather and enter information to go door-to-door for gathering insurance data for their customers.
The employees would take their tablet in the morning and run their given route inputting the data as they go with a responsive prompt in response to what the customer responded. At the end of the day, they would connect the tablet to Wi-Fi and all data would sync to the server in the cloud.
The server would run at a scheduled time and email ARI's clients the data that they had ordered.

- Architected, Designed, and Developed from scratch using Angular 2, Ionic 2, Typescript, Node.js, Express, MongoDB, and Redis
- Used Ionic Cloud for application deployment without manual updates from the user.
- Used a REST API to send HTTP requests to a Node.js server
- Integrated Redis for caching of data and MongoDB for storage.
- Prototyped an admin dashboard for viewing data on our website

Chad Jones Law Firm – Systems Administrator (January 2015 – December 2016)
Responsible for complete integrity of all devices and systems at the firm. Projects and tasks included:

- On call 24/7
- Contributed to the design of new work spaces and completed the implementation for two new offices
- Worked on setup and placement of the primary on-site server
- In charge of all tiers of technical support on and off the campus
- Create documentation for proper procedures for all IT related projects
- In charge of technical purchases
- Primary advisor to the attorneys in matters of technology for their cases

CodeChat – Project Manager, Network Server Administrator

During my Fall semester in 2017, I worked with a 9-man team to design and build any project we desired. Some of my team mates had been working together previously and had tried sharing code between each other, However they found no optimal way.

We decided to build a chatting application that could run and compile code immediately. I was the lead programmer for all JavaScript modules within front-end and back-end. Our whole system utilizes websockets between the webapp, the chat server, and the compiling server, making it as network performant as possible.

Kaijuwu Site

My Overwatch Coach Approached me about working together to design and build a team site for his Overwatch Teams. This was my first time working with a UX/UI designer where I had to focus more on styling than I have in the past. I thoroughly enjoyed it and will plan on doing similar projects if possible.

My Pantry

To work on my SQL skills and to show off my JavaScript abilities, I decided to build the classic to-do app, but of course with a small twist.

The front end of the app is in pure JavaScript with the exception of using axios for http requests. The back end is written in node and uses PostgreSQL for the SQL database. There is a basic register/login system encrypted with crypto-js.

References Available Upon Request