**eric webb**

|  |  |
| --- | --- |
|  |  |
| Phone: (316)-706-4752 | LinkedIn: [linkedin.com/in/eric-webb-dev](https://linkedin.com/in/eric-webb-dev) |
| Address: 2100 E. Frontenac St, Park City, KS, 67219 | GitHub: [github.com/Leaderfirestar](https://github.com/Leaderfirestar) |
| Email: [EricAWebb2000@yahoo.com](mailto:EricAWebb2000@yahoo.com) | Portfolio: [portfolio.eric-webb.dev](https://portfolio.eric-webb.dev) |
|  |  |

**EDUCATION** Kansas State Polytechnic University, Salina, KS

**Bachelor of Science in Engineering Technology**  Graduation: May 7, 2022

*Computer Systems* GPA: 3.9

**Associate of Technology in Engineering Technology**  May 2020

*Web Development Technology*

**SKILLS** Programming: Typescript, JavaScript, Nodejs, Express, NextJs, ReactJs, C#, Python, SQL, DexieJs, Firebase, MongoDB, AWS Suite, Socket.IO, Strapi, GraphQL

**WORK** FactorEarth, Lakewood, CO May 2021-Current

**EXPERIENCE Software Engineer**

* Set up FactorEarth *Record* as a PWA with the browser’s IndexedDB (using Dexie.js) to allow users to work offline for extended periods time, and the bi-directional syncing to allow the users to sync to the cloud database while utilizing the timestamp their content was last updated at in combination with a revision history to merge local – remote without merge conflicts
* Inventor on the “Document Form Export System” and “Customized Field Action Template System of A Document Form Export System” patents
* Wrote CLI utility with the AWS-SDK to facilitate the creation of Cognito users and groups, as well as adding users to Cognito groups to speed up the process of *Record* user setup
* Collaborated on a Data Access Layer (DAL) library that allows our lambda functions to CRUD documents in our database, without needing to write the code over and over, and allowing for a potential database shift in the future that would only require changing the code in the library and updating the version the lambdas are using
* Optimized FactorEarth *Explore’s* performance with respect to Google’s *Lighthouse* metrics, including reducing the app’s bundle size from 87mb to 5mb, implementing lazy loading for components deeper in the app, converting all images uploaded through FactorEarth *Curate* to .webp format and compressing them to tremendously decrease the file size/load time, and migrating the app from Firebase 8 to Firebase 9 (and eventually, AWS Amplify with DynamoDB) to take advantage of tree shaking to minimize unused code within the site bundle
* Set up FactorEarth *Forms* function that would generate test data and insert them into all forms we had built out, and go through the pdf/word exports to ensure all data was inserted correctly into the documents, outputting a text file with fields within the forms that did not get filled in correctly, all for the purpose of ensuring we didn’t break any existing forms when bug fixing
* Handled setting up Ec2 instances to run the apps, including the Nginx configuration (serving all assets compressed with the brotli compression algorithm, automatically redirecting http to https, etc.) and cron jobs to automatically renew ssl certificates

**PROJECTS Root Access,** Senior capstone projectMay 2022

* Configured a raspberry pi to read output from a moisture sensor, light sensor, float sensor, thermometer, and humidity sensor into a python script to autonomously grow Red Rubin Basil

**Soul Arena,** Turn-based strategy game Oct 2022-Current

* Online game where 2 players bring 3 characters into a battle, last team standing wins. Created with Socket.IO, Express, and Unity