Kevin Lane

Full Stack Software Engineer

**Contact Info**

Cell: 208-251-1087

kevinclane21@gmail.com

[linkedin.com/in/KevinChristopherLane](http://www.linkedin.com/in/KevinChristopherLane)

[kevinclane.com](http://www.kevinclane.com)

[github.com/Kevinclane](https://github.com/Kevinclane)

**Skills**

Java, Spring Boot, Maven, Gradle, C#, .NET, Entity Framework, Angular, React, Vue, JavaScript, TypeScript, CSS, SCSS, HTML5, Node.js, MySQL, MSSQL, Docker, Azure Cloud Services, Git, Firebase, MongoDB, Figma, CI/CD pipelines

**Employment**

**Allata LLC | Senior Consultant**

**June 2022 - Apr 2024**

* Architected system to track batched data through multiple Azure Service Bus Queues, enabling ingestion status transparency to the business team
* Extracted and refactored file download processes into its own generic, highly extensible microservice, saving computing resources from the main servers and increasing speed
* Integrated Web Sockets for real time data updates, reducing api server traffic, reducing response time, and increasing business team’s productivity
* Built various pages and API endpoints with CRUD operations for SPA
* Wrote documentation/diagrams of complicated data flows for future training/onboarding
* Included extensive unit testing for functional integrity, ensuring the application works as intended

**In Time Tec | Software Engineer**

**Aug 2021 - June 2022**

* Updated MySQL Server database, removing the need for manual editing to generate sales reports and increasing sales team’s performance
* Created Splunk dashboards for monitoring application metrics, allowing detailed insight into the application’s health
* Built and Deployed web API, secured with limited authentication access
* Implemented continuous deployment via CI/CD pipeline to multiple environments

**Mesh Software LLC | Software Developer | Contract**

**Oct 2020 - July 2023**

**Flatout MotorSports (**[**www.flatout motorcycles.com**](http://www.flatoutmotorcycles.com)**)**

* Improved data security by creating and rerouting API calls to a limited data server
* Added/updated MySQL tables and server functions to provide additional data to users
* Built powerful data modals to assist the sales team, boosting data availability through faster updates, ultimately increasing sales performance
* Increased performance, speed, and security by updating outdated, legacy code

**Fourcaster LLC | Software Developer | Contract**

**Mar 2021 - Apr 2021**

[4casters.io](http://4casters.io)

* Refactored Node(Koa) server into MVC pattern
* Built test pages to check data changes and server strength

**Education**

**Codeworks | Boise, Id**

**Apr 2020 - July 2020**

* Field: Full-Stack Web Development
* Certificate: Completion
* Coursework/Technologies: Vue.js, Node.js, and .NET frameworks. MySQL, MongoDB, Bootstrap, Axios, JQuery, Auth0, and RESTful API’s
* Applications: Built multiple apps using (front-end) vanilla JavaScript or Vue.js with bootstrap library and (back-end) Node.js and .NET with MongoDB data storage.

**Personal Projects**

[**TimeTrackers**](https://www.timetrackers.net/#/)- Time tracking application learning project

* Vue frontend, Node backend, MongoDB database
* PayPal Integration for secure payment handling
* Leverages Docx.js and FileSaver.js to generate/download invoices
* Supports a variety of pay frequencies and pay types (Salary, Hourly, ect)

[**TexasHold ‘em**](https://github.com/Kevinclane/HackathonPoker)- 1 week Hackathon project

* Vue.js frontend, Node backend, MySQL database
* Utilizes Web Sockets for real-time game updates
* Multiplayer support

**Java/Angular Training** - Final Fantasy inspired ([Front End Repo](https://github.com/Kevinclane/mini-ff)) ([Back End Repo](https://github.com/Kevinclane/JavaGame))

* Angular frontend, Java backend, MongoDB database
* Traditional turn based combat with rudimentary AI enemies
* Smooth sprite animation

[**AZ204Training**](https://github.com/Kevinclane/az204quizmasterAPI) - Quiz application (In progress) - Intended workflow:

* Drop JSON file in Blob Storage
* Azure Function detects file and forwards contents to web API
* API validates/saves data into DB
* Front end requests quiz question(s) and returns user answer(s)
* Backend validates response