

Jason Cabrera

(626) 541 – 1335 jrcabrer@ucsd.edu

Baldwin Park, CA jayrc7.github.io/personal-website

github.com/jayrc7

linkedin.com/in/jasonrcabrera

Education

September 2017 – June 2021 | Major GPA: 3.69

University of California, San Diego | B.S. Computer Science

January 2022 – PRESENT (Part Time) | Current GPA: 4.0

University of Texas at Austin | M.S. Computer Science - Machine Learning Specialization

Technical Skills

Advanced | Java, Spring Boot, C#, Python, JavaScript, NodeJS, ReactJS, AngularJS, SQL, Relational Databases, Express, Kubernetes, Docker, Azure DevOps, Azure Cloud, MongoDB

Familiar | C++, C, Kotlin, Machine Learning, Android, Pandas, Numpy, Android, JQuery, Amazon S3

Work Experience

General Motors | Full Stack Software Engineer | *July 2021 – PRESENT*

- Optimized SQL queries within microservices for better performance. Resolved concurrency issues using row level locking mechanisms provided by OracleDB.
- Implemented new frontend features such as form validation and duplicate account prevention.
- Completed frontend and backend for a feature which allows dealers to transfer a customer's existing vehicle's plans to their newly purchased vehicle. Created frontend pages to facilitate the transfer. Updated database schemas and created API routes in the backend.
- Worked on adding features to internal web applications such as being able to login using an employee's Microsoft account as well as adding middleware function to API routes to prevent unauthenticated users from accessing the different tools portals, also added the ability to lock and unlock users from a variety of internal applications that rely on the same user database.
- Created a new microservice to automate some of the manual work needed when updating a vehicle module's software over the air. Created a WebClient Spring Boot bean to call APIs, created DTOs to store API responses inside Java objects, coded the logic of the microservice according to the documented requirements, integrated Kafka using Spring Integration to receive and send messages to another microservice.
- Created Build and Release pipelines to deploy Docker images of microservices to Kubernetes using Azure DevOps build and release pipelines. Created Release Pipelines to deploy applications on Microsoft Azure.
- Lead a team workshop on how to .NET deploy applications to Kubernetes through Azure DevOps pipelines.
- Developed a new feature for the NCAA march madness in vehicle application that sends vehicle notifications to the driver for upcoming and current games played by their favorite teams. Implemented option for turning on/off notifications.

Projects

Kiwi Forum | Undergraduate Software Engineering Course, Project Manager and Full Stack Developer

- Led a team of ten developers in an Agile environment to develop a specialized forum web application for a software engineering course.
- Created the database schema, added navbar functionality that allowed employees to list/change the specialization by which their forum was filtered by, and created/tested RESTful API for CRUD operations on the database.
- Developed the authentication system using Express, Firebase, and cookies to allow employees and admin authenticated backend routes and pages.
- Created a secure employee invite system that emails a unique one-time-only invite link to a specified employee's email that would grant access to the admin's company forum.

Movie of the Week | Full Stack Software Development | <https://github.com/jayrc7/Movie-of-the-week>

- Developed a movie tool using React and NodeJS that allows friend groups to insert their movie suggestions and then randomly picks a movie from a user that hasn't been chosen yet.
- Created a recent updates section using merge sort by sorting on multiple date-valued fields in the database. Updates include someone recommending a new movie or a movie being marked as watched by a group.
- Implemented a user authentication system using server-side cookies and middleware functions as well as creating a react authentication component that checks if the user is authenticated before showing a page.