Trace Bailey

Dr. Maciosek

CS-499-19502-M01

April 6 2025

Databases

The artifact I picked for the databases category is the backend for Swiftride Rentals, a fictional car rental app I’ve been working on. I started this project originally to learn React, and after learning React and the rest of web development, I thought it would be the perfect opportunity to come back and show what I have learned and build a full stack web application. I built a custom Express backend and hooked it up to a PostgreSQL database which is where most of the complexity came in. Before this, I only really had static data on the frontend but now the vehicles, users, and rentals are all stored and managed in a real database and everything works together.

I included this in my ePortfolio because I want to become a full stack developer. That’s the career path I’m aiming for so I wanted to show that I understand both sides of development, not just how to make things look good on the screen but also how the data flows, how it’s stored, and how to build out APIs that handle that data securely and efficiently. This artifact shows I can build a backend from scratch, connect it to a PostgreSQL database, protect it with JWT, and create routes that only certain users (like admins) can access. I think it reflects the kind of real-world work I hope to be doing once I graduate.

This artifact was improved a lot since the original idea. I designed a full schema for users, vehicles, and rentals and made sure to include things like foreign keys and indexing. The backend has full CRUD functionality now and every operation interacts with the live database. Admins can log in and update vehicles, and all of it is protected by token-based authentication. I also built out the logic for signup and login so that the user experience is fully integrated from the frontend to the backend.

I met the course outcomes I planned for in Module One. Specifically, I’ve demonstrated the ability to use current tools and techniques to create something useful and functional. And I made sure to think about security from the beginning with hashed passwords, role-based access, and limited exposure of sensitive data. I don’t have any big changes to my outcome plan—it stayed pretty much on track with what I intended.

During this process I ran into a lot of little things I hadn’t expected. At first, I had trouble hosting the database and ran into SSL certificate issues. I also learned how important it is to structure your database clearly from the beginning because one wrong field type or a missing constraint can cause a lot of headaches later. Figuring all that out taught me more than any tutorial I’ve done. I learned how to debug real problems, plan better, and think ahead.

This part of the project was a huge step forward for me and honestly it’s one of the things I’m most proud of. I know there’s more to do still, but I finally feel like I’m not just building pieces of an app, but I’m building the whole thing.