



Amos Roland

Southern New Hampshire University

CS-499 Computer Science Capstone

Professor Akhil Gudivada

16 November 2025

CS 499 Module 3-2 Milestone Two: Enhancement One: Software Design and Engineering?

Artifact Description

The artifact I selected for my software design and engineering enhancement is my Travlr Getaways Full-Stack Web Application, which I originally created in CS 465: Full Stack Development I. The project started as a partially functional travel listing website that relied on server-side Handlebars templates. The earlier version did not fully implement RESTful communication, persistent data storage, or secure authentication.

For the CS-499 enhancement, I rebuilt the application as a true MEAN stack solution using MongoDB, Express, Angular, and Node.js. I redesigned the admin portal as a modern single-page application and added secure API endpoints, authentication workflows, and complete CRUD features.

Why This Artifact Is in My ePortfolio

I selected this artifact for my ePortfolio because it is the strongest representation of my growth as a full-stack engineer. This project demonstrates my ability to design, develop, and enhance a complete software system using industry-standard tools. The enhancements show my



skills in software architecture, API engineering, authentication systems, and user interface design.

Several components highlight my software engineering capabilities:

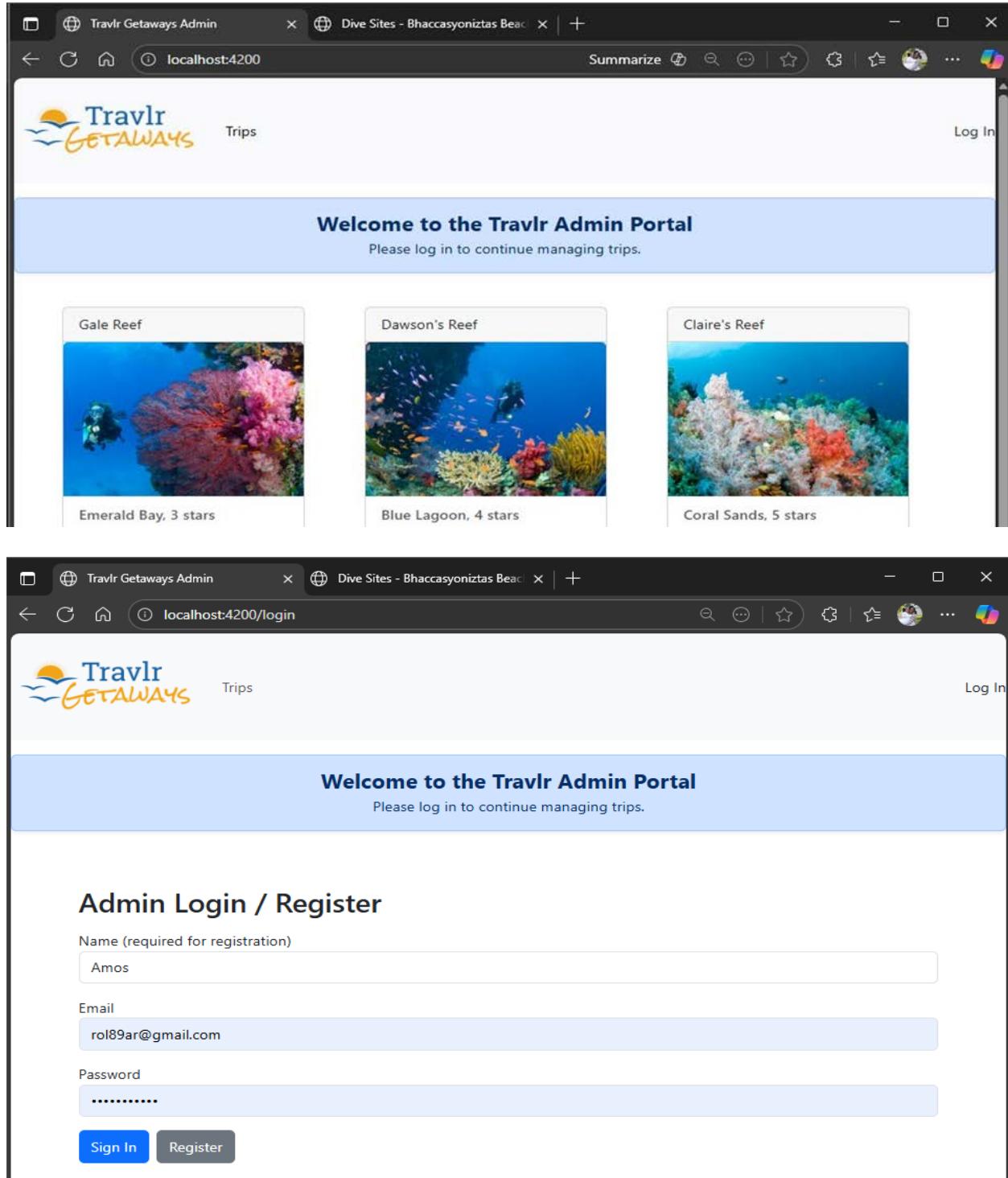
- I implemented a complete RESTful API with GET, POST, PUT, and DELETE routes.
- I developed secure JWT authentication using Passport and LocalStrategy.
- I redesigned the Angular admin portal using modular components, reactive forms, and service-oriented architecture.
- I added Angular interceptors to automatically attach authorization tokens.
- I improved the visual layout and added features like welcome messages and error handling.

The updated version clearly demonstrates growth in my software engineering skills and aligns with the program outcomes I planned to meet.

The figure consists of two screenshots of a web application. The top screenshot shows the homepage of 'Travlr Getaways'. It features a header with the logo and a 'Trips' link, followed by three cards: 'Gale Reef' (with an image of a coral reef), 'Dawson's Reef' (with an image of a reef), and 'Claire's Reef' (with an image of a reef). The bottom screenshot shows the 'localhost:4200/login' page. It has a header with the logo and a 'Trips' link, followed by a 'Login' form. The form includes fields for 'Name' (with placeholder 'Enter Name'), 'Email Address' (with value 'rol89ar@gmail.com'), 'Password' (with a masked input field), and a 'Sign In!' button.

Figure 1: Your original CS-465 homepage or code snippet showing the original structure.

Purpose: Demonstrates where the artifact began.



The image displays two screenshots of a web application interface, likely an Angular application, running on a local host at port 4200.

Screenshot 1: Dive Site Listing (Logged Out)

The top screenshot shows the "Welcome to the Travlr Admin Portal" page. At the top, there is a navigation bar with tabs for "Travlr Getaways Admin" and "Dive Sites - Bhaccasyonitzas Beach". The URL in the address bar is "localhost:4200". Below the navigation is the Travlr Getaways logo and a "Log In" button. A "Trips" link is visible in the top right corner. The main content area features three cards, each representing a dive site:

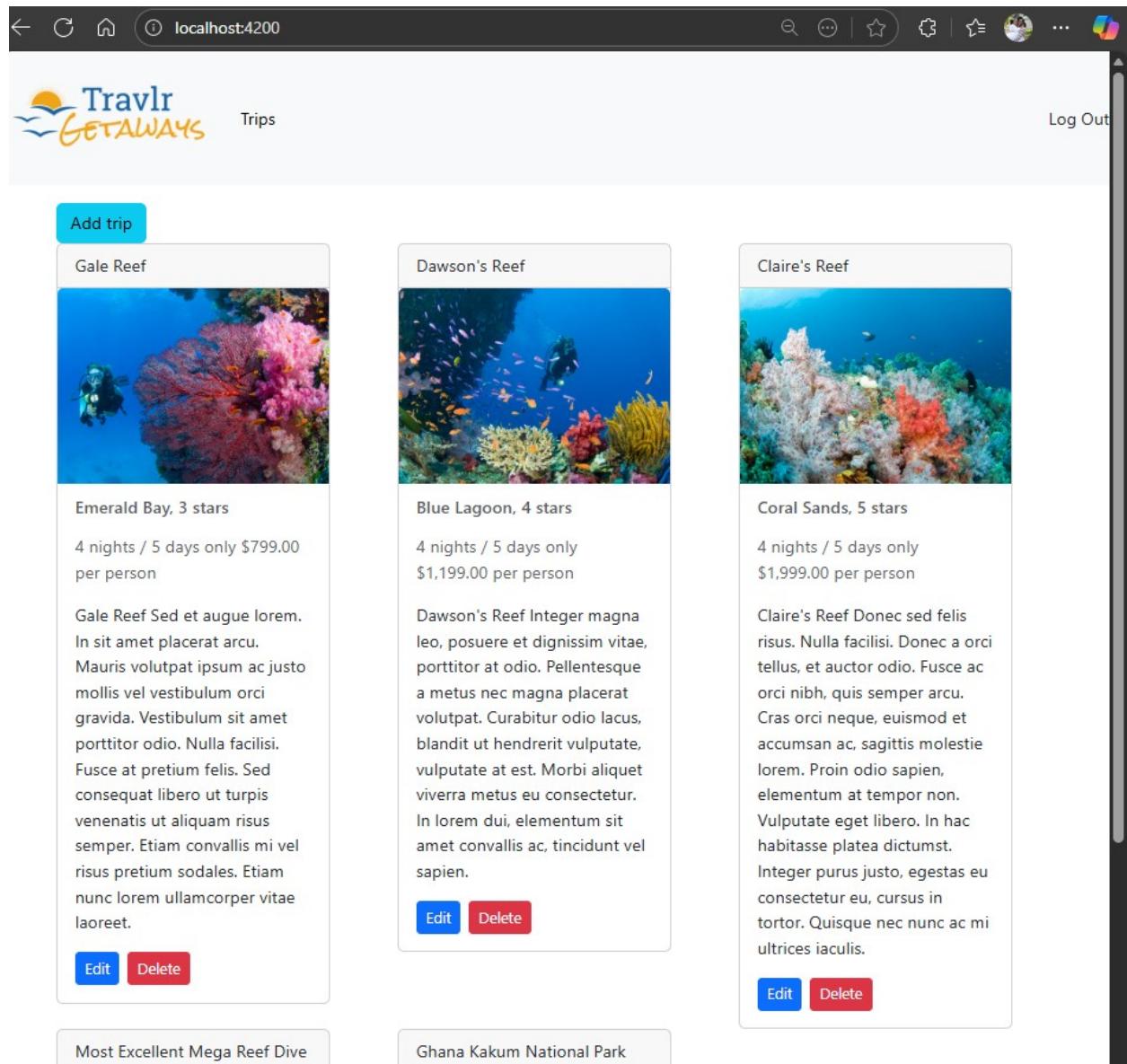
- Gale Reef**: An image of a scuba diver swimming near a large, pink coral reef. Below the image is the text "Emerald Bay, 3 stars".
- Dawson's Reef**: An image of a vibrant coral reef with many small fish. Below the image is the text "Blue Lagoon, 4 stars".
- Claire's Reef**: An image of a coral reef with various colors. Below the image is the text "Coral Sands, 5 stars".

Screenshot 2: Admin Login / Register Page (Logged Out)

The bottom screenshot shows the "Welcome to the Travlr Admin Portal" page with a "Please log in to continue managing trips." message. The URL in the address bar is "localhost:4200/login". The layout is identical to the first screenshot, with the Travlr Getaways logo, "Log In" button, and "Trips" link. The main content area is titled "Admin Login / Register" and contains fields for Name, Email, and Password, along with "Sign In" and "Register" buttons.

Figure 2: Angular Trip Listing page (logged out) with the welcome message.

Purpose: Shows improved UI and front-end design.



Gale Reef

Emerald Bay, 3 stars
4 nights / 5 days only \$799.00 per person

Gale Reef Sed et augue lorem. In sit amet placerat arcu. Mauris volutpat ipsum ac justo mollis vel vestibulum orci gravida. Vestibulum sit amet porttitor odio. Nulla facilisi. Fusce at pretium felis. Sed consequat libero ut turpis venenatis ut aliquam risus semper. Etiam convallis mi vel risus pretium sodales. Etiam nunc lorem ullamcorper vitae laoreet.

[Edit](#) [Delete](#)

Dawson's Reef

Blue Lagoon, 4 stars
4 nights / 5 days only \$1,199.00 per person

Dawson's Reef Integer magna leo, posuere et dignissim vitae, porttitor at odio. Pellentesque a metus nec magna placerat volutpat. Curabitur odio lacus, blandit ut hendrerit vulputate, vulputate at est. Morbi aliquet viverra metus eu consectetur. In lorem dui, elementum sit amet convallis ac, tincidunt vel sapien.

[Edit](#) [Delete](#)

Claire's Reef

Coral Sands, 5 stars
4 nights / 5 days only \$1,999.00 per person

Claire's Reef Donec sed felis risus. Nulla facilisi. Donec a orci tellus, et auctor odio. Fusce ac orci nibh, quis semper arcu. Cras orci neque, euismod et accumsan ac, sagittis molestie lorem. Proin odio sapien, elementum at tempor non. Vulputate eget libero. In hac habitasse platea dictumst. Integer purus justo, egestas eu consectetur eu, cursus in tortor. Quisque nec nunc ac ultrices iaculis.

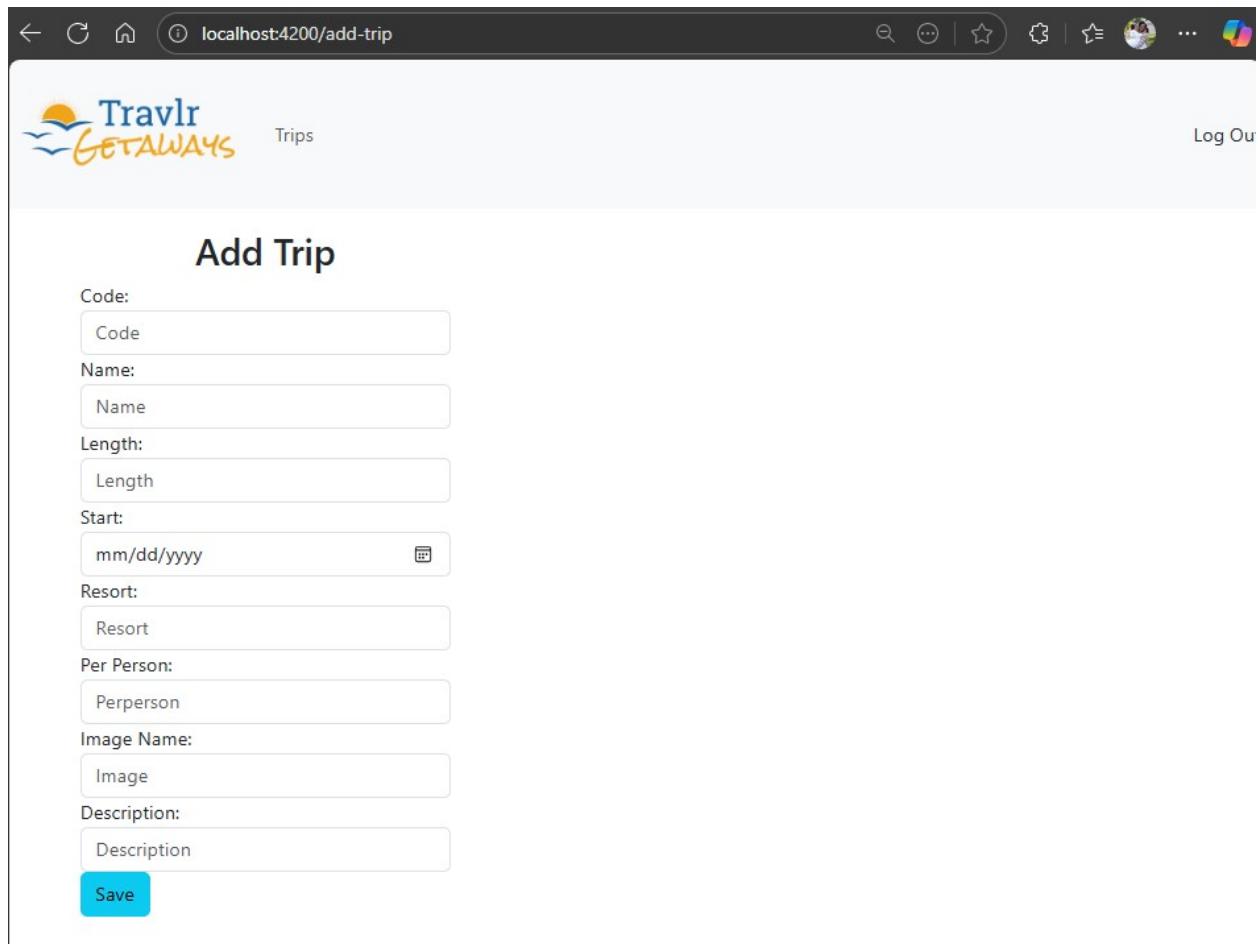
[Edit](#) [Delete](#)

Most Excellent Mega Reef Dive

Ghana Kakum National Park

Figure 3: Angular Trip Listing page (logged in) displaying Edit + Delete buttons.

Purpose: Shows authenticated state and admin actions.



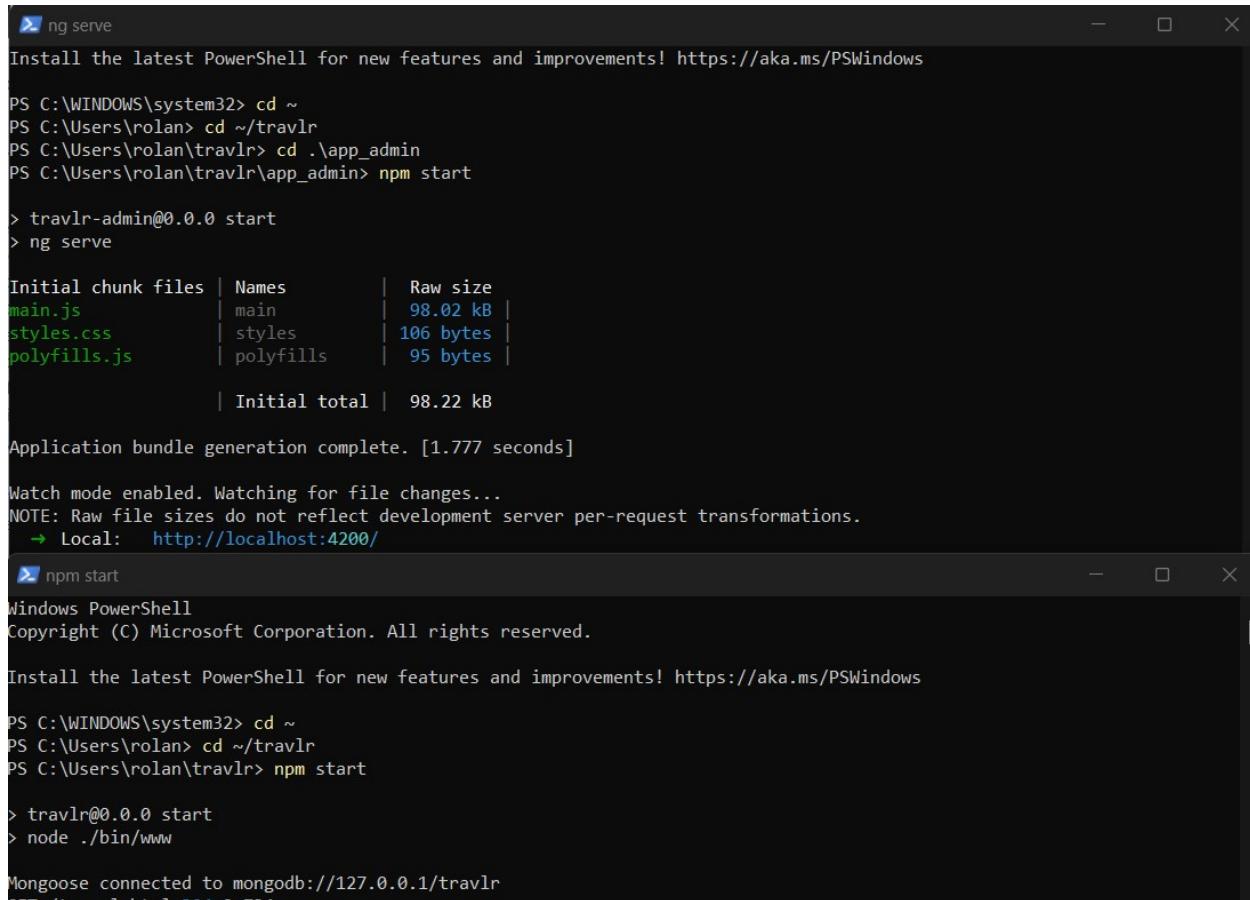
The screenshot shows a web browser window with the URL `localhost:4200/add-trip`. The page title is "Travlr Getaways". On the right side, there are navigation icons for search, refresh, and user profile. A "Log Out" link is also visible. The main content area is titled "Add Trip". It contains a form with the following fields:

- Code:
- Name:
- Length:
- Start:
- Resort:
- Per Person:
- Image Name:
- Description:

At the bottom left of the form is a blue "Save" button.

Figure 4: The Add Trip or Edit Trip form.

Purpose: Demonstrates reactive forms and user interaction.



```

PS C:\WINDOWS\system32> cd ~
PS C:\Users\rolan> cd ~/travlr
PS C:\Users\rolan\travlr> cd .\app_admin
PS C:\Users\rolan\travlr\app_admin> npm start

> travlr-admin@0.0.0 start
> ng serve

Initial chunk files | Names      | Raw size
main.js             | main       | 98.02 kB |
styles.css          | styles     | 106 bytes |
polyfills.js        | polyfills  | 95 bytes |
                           | Initial total | 98.22 kB

Application bundle generation complete. [1.777 seconds]

Watch mode enabled. Watching for file changes...
NOTE: Raw file sizes do not reflect development server per-request transformations.
→ Local:  http://localhost:4200/

```

```

PS C:\WINDOWS\system32> cd ~
PS C:\Users\rolan> cd ~/travlr
PS C:\Users\rolan\travlr> npm start

> travlr@0.0.0 start
> node ./bin/www

Mongoose connected to mongodb://127.0.0.1/travlr
GET /travlr.html 200 2.724 ms

```

Figure 5: Terminal running the Node API showing successful startup and MongoDB connection.

Alignment to Course Outcomes

In Module One, I planned to meet the outcomes related to software design and engineering by transforming this artifact into a fully functional, authenticated full-stack application. After completing the enhancement, I successfully met the outcomes I targeted. Specifically:

- I demonstrated the use of innovative coding techniques, including interceptors, structured services, and secure API workflows (Outcome: Software Engineering).



- I improved the system's security posture by implementing JWT authentication and protected API routes (Outcome: Security Mindset).
- I produced a modular and scalable solution aligned with industry practices in full-stack development (Outcome: Computing Practices and Standards).

At this stage, my next outcomes will focus on Algorithms and Data Structures, and Databases as I complete the remaining enhancements later in the modules.

Reflection on the Enhancement Process

Enhancing this artifact taught me how much work goes into transforming a partially functioning web application into a secure, production-ready system. One key lesson I learned was the importance of the separation of concerns. Splitting responsibilities between the Angular front-end, Express API routes, controllers, services, and database models made the application far easier to scale and debug.

I also learned more about authentication. Implementing JWT security forced me to understand how tokens work, how to protect routes, and how to safely store and retrieve user credentials. Adding full CRUD functionality also strengthened my understanding of REST design principles.

A major challenge I faced was ensuring that Angular components communicated properly with asynchronous authentication calls. Another challenge was debugging CORS issues and making sure the front-end and back-end were aligned. By working through these issues, I improved my debugging skills and developed a better understanding of full-stack workflows.



All in all, the enhancement process strengthened my confidence and showcased my ability to build full-stack applications that follow industry standards. This artifact now represents my capabilities in designing, securing, and deploying modern software solutions.

References:

Auth0. (n.d.). *JSON Web Tokens - jwt.io*. Auth0. Retrieved November 12, 2025, from

<https://www.jwt.io/>

OWASP Foundation, the Open Source Foundation for Application Security. (n.d.). OWASP Foundation. Retrieved November 13, 2025, from <https://owasp.org/>