Chorsie Martin III

SNHU

CS-499 Computer Science Capstone

Enhancement Three: Databases

The artifact I used for this milestone is my CS 465 full stack web application that I originally built earlier in the program while learning server-side development with Node.js and Express. When I first created the project, it mainly focused on routing and rendering pages, and it did not include a real database. For this enhancement, I added MongoDB with Mongoose so the application now stores and retrieves room data from a database instead of using hard-coded data. This made the application feel more like a real system where data is persistent and structured rather than temporary.

I chose this artifact because it gave me a good opportunity to show database skills inside a working application. I created a Room schema that includes validation rules to help make sure the data stored is correct and safe. I also added database indexes to improve performance when users sort or search for rooms. For example, price and rating indexes help with filtering and sorting, and a text index allows users to search by keywords. Earlier in the project, filtering and sorting were done in memory using arrays, which works for small datasets. By moving this logic into MongoDB queries and using indexes, the application is more scalable and closer to how real-world systems handle larger amounts of data.

This enhancement helped me meet the course outcomes I planned earlier. I demonstrated the use of industry tools like MongoDB, Mongoose, schemas, and environment variables, which

shows my ability to implement real database solutions. I also improved the design of the application by thinking about performance and scalability, not just whether it works. Adding schema validation supports a security mindset by reducing the risk of bad or unsafe data entering the system. These updates strengthened my understanding of how databases support full stack applications and connect to overall system design.

Working on this enhancement taught me that adding a database changes how an application is built and organized. I learned how schemas, indexes, and queries work together to manage data efficiently. One challenge I faced was making sure the application could still run safely even if the database was not connected, which required adding checks and error handling. Another challenge was understanding how search and sorting behave differently when handled by the database instead of in memory. Overall, this milestone helped me move from thinking about small program logic to thinking more about how real systems manage data and scale.