

Charles Campbell

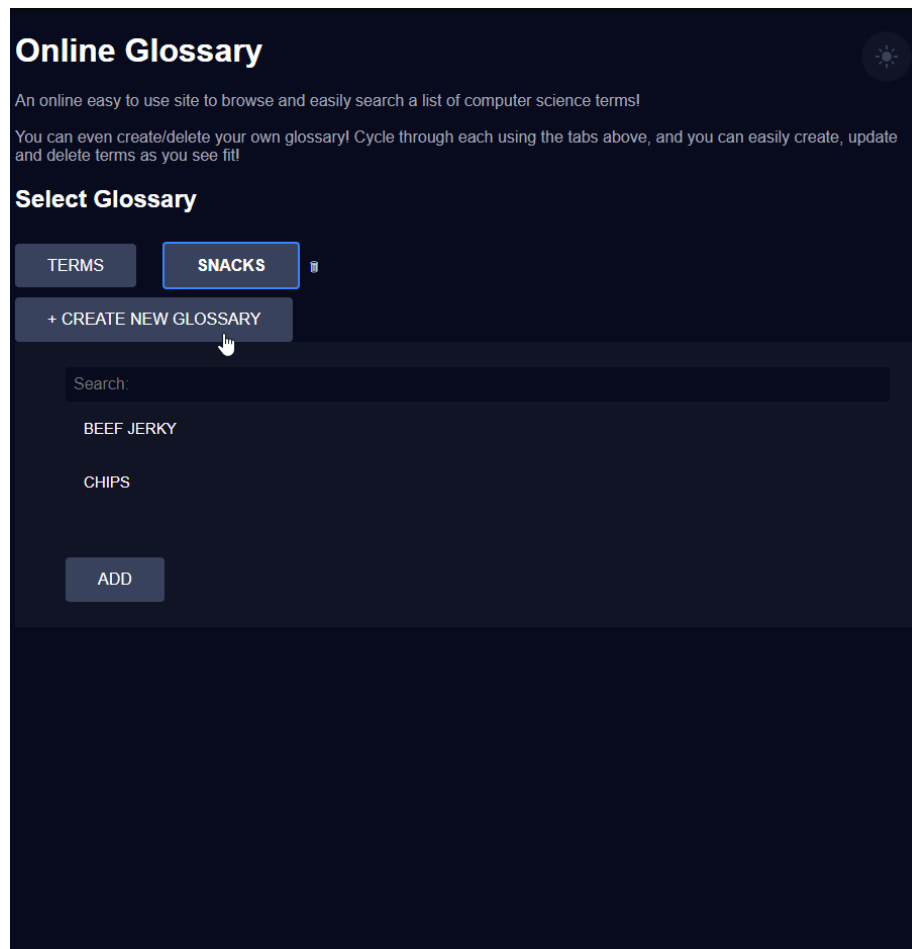
Enhancement Three

2 August 2025

Databases Narrative

The original artifact I made back in February – April of 2025 of a client/server project would connect to the MongoDB. The MongoDB itself was pre-defined, and the CRUD operations that the user could utilize were only for reading the items in the database. The user could only see the information stored in the database, and the front-end operations would only be able to filter that data. However, many operations of SPAs or other mutli-layerd web applications involve more than just reading data from a database. The ability to correctly make changes, insert new data into the collections, or even create entirely new collections are highly technical and purposeful actions that modern sites use. That's what I wanted to make in my revision of the artifact; to showcase I can make full use of the MongoDB in development.

The enhancements, thanks to the Node & Express backend frameworks, allowed for a seamless enhancement of utilizing the MongoDB. I was able to fully implement operations for users to easily create their own glossary list (which would become a new collection in the MongoDB), and with familiar CRUD capabilities to add new items to their newly created list. The users could even delete the glossaries they created. However, I saw a potential data security issue with this, in that users could potentially delete the original 'terms' collection that the database was seeded with. So, to enhance the security, I created failsafes by ensuring in the code that the delete button (the little trashcan next to each glossary tab) would not show up for the original 'terms' collection. While users could delete new glossaries they made, they wouldn't be able to remove the core terms list from the site, and more importantly from the MongoDB.



Designing the back-end functions for creating/deleting the extra glossaries was simple, as they followed the CRUD operations easily. The main issue came with the front-end, and I had to complete a bunch of re-factoring to make sure that not only the new feature worked well, but that the original list functions (create/delete/add) also still worked as intended. But it did allow me to fully modularize the front end functions, and create a seamless user interface that made the functions immediately responsive and easy to use for users on the SPA. It's my hope with these new database features, I've been able to meet the desired outcomes of showcasing my abilities to use well-founded techniques, skills, and tools in designing computer solutions that deliver value; while showcasing my understanding of developing with a security mindset and anticipating exploits in software architecture with enhancing security of data and resources.