Southern New Hampshire University

CS-499 Module One Assignment
Nik Myers
CS-499-19736
Oct. 30, 2025
Phillips

I. Self-Introduction: Address all of the following questions to introduce yourself.

A. How long have you been in the Computer Science program?

My first term at SNHU started in May 2023, so by the end of this term it will be two and a half years.

B. What have you learned while in the program? List three of the most important concepts or skills you have learned.

While I entered the program with almost 20 years of freelancing and recreational programming experience, the more structured academic environment helped me formalize and deepen my understanding of software development, and introduced me to the foundational topics in Computer Science that I would likely never attempt to learn on my own (i.e., the underlying mathematics, data structures and algorithms, etc.).

Regarding specific concepts or skills, I believe the most important ones I've developed are: Security-minded software engineering, system-level analysis and reverse engineering, and embedded computing.

C. Discuss the specific skills you aim to demonstrate through your enhancements to reach each of the course outcomes.

Through the enhancements planned for my capstone project, I intend to demonstrate a blend of technical proficiency and design maturity that reflects my growth throughout this program as well as my existing experience with web development. By re-engineering my original project into a modern web application, I will show my ability to design scalable software architectures, employ secure development practices, and integrate algorithmic reasoning into practical applications. I plan to demonstrate skills such as modular software design, efficient data handling, and type-safe programming. The project will also show that I can evaluate design trade-offs and make decisions that balance security, maintainability, and performance. Just as importantly, the documentation and testing I produce will highlight my ability to communicate my technical choices clearly and professionally, tying together conceptual design and implementation.

D. *How do the specific skills you will demonstrate align with your career plans related to your degree?*

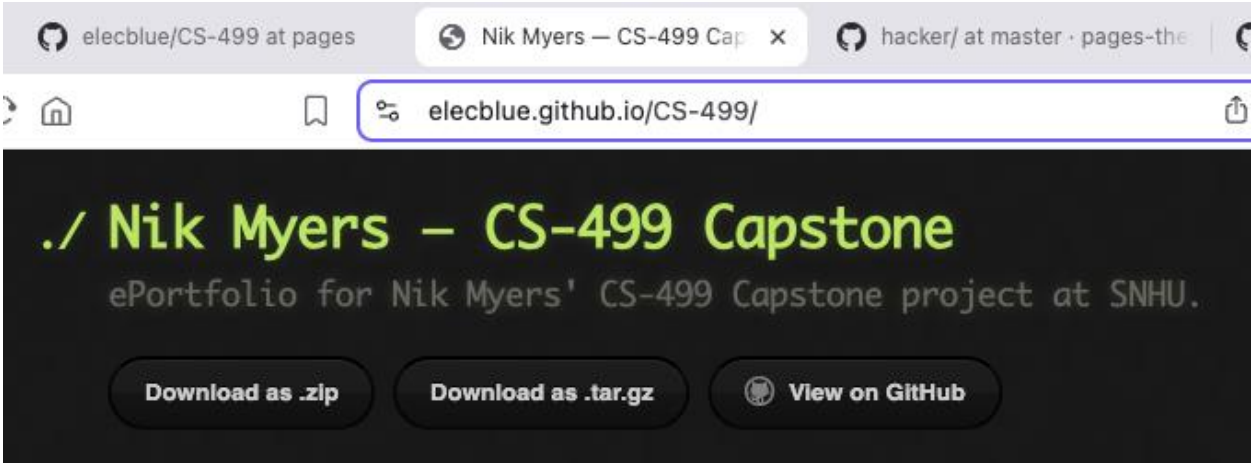
The skills I plan to demonstrate in my capstone align closely with the kind of career I hope to pursue after graduation, though I haven't yet decided exactly what that will be. My interests

have always centered on creative problem-solving through technology, whether in the context of game development, machine learning, or reverse engineering, and each of these paths demands an understanding of secure, efficient software design. By enhancing my project with a modern stack and a focus on data integrity and performance, I am refining the same core abilities that drive those fields: strong architectural thinking, algorithmic precision, and a disciplined approach to building reliable systems. Even outside of professional settings, I enjoy designing small projects and tools for fun, and the technical skills I'm honing here will directly transfer to both personal and professional endeavors. To me, this project isn't just about proving what I've learned: it's about reinforcing the habits and technical mindset I'll continue applying long after graduation.

E. *How does this contribute to the specialization you are targeting for your career?*

While I haven't narrowed my focus to a particular specialization yet, my interests tend to converge around applied software engineering, a discipline that connects theoretical computer science with practical applications. Enhancing my original submission allows me to demonstrate this fusion in action by designing a complete software system that handles data securely, performs efficiently, and remains adaptable for future innovation. This kind of work reflects the versatility of my Computer Science degree and aligns with any future specialization I may pursue, whether it involves AI research, game development, or interactive systems. The project will serve as a clear example of how thoughtful design, secure coding, and algorithmic reasoning can combine to produce software that is both technically sound and purpose-driven.

II. ePortfolio Set Up:



III. Enhancement Plan:

A. **Category One:** Software Engineering and Design

i. ***Select an artifact*** *that is* ***aligned with the*** *software engineering and design* ***category*** *and explain its origin.*

CS-340: Client/Server Development
Grazioso Salvare (Animal Rescue) Dashboard

[Archive of Artifact (GitHub)]

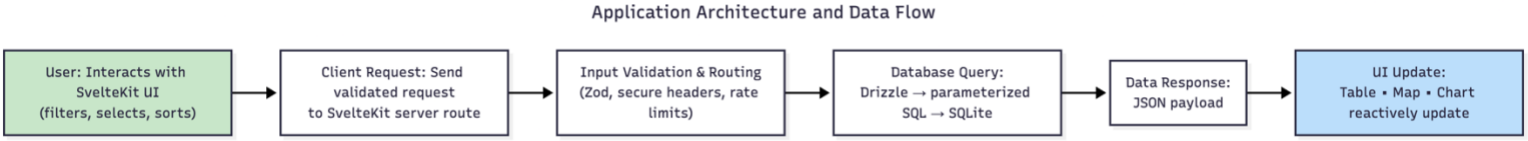
This artifact, which I will be enhancing for all three categories, was my submission for the final project in CS-340. This web application provided a dashboard for visualizing animal rescue data. The dashboard presented the database in a dynamic table view, and featured an interactive map that displayed a "pin" at the precise location a selected animal was found, as well as a pie chart that broke down the breed distribution of the animals within the active "rescue type" filter (i.e., the breeds considered suitable for mountain, water, or individual tracking rescue operations).

The original implementation as required by the course leveraged Python and the Dash framework for the front-end, while the backend used MongoDB with data imported from a provided CSV document. Although it functioned well for a classroom project, it was not a scalable or maintainable architecture for a real-world product.

ii. ***Describe*** *a practical, well-illustrated* ***plan*** *for enhancement in alignment with the category, including a pseudocode or flowchart that illustrates the planned enhancement.*

For the software engineering and design enhancement, I will rebuild the entire application using TypeScript with Svelte and SvelteKit running on the Bun runtime. This modern stack provides a clear separation of concerns and a more secure, maintainable foundation. The new design will feature a reactive, component-based UI that focuses on accessibility and performance. Client-side state will be handled using Svelte runes, allowing filters and selected views to remain in sync with the URL so users can share their results easily. The data table will use row virtualization and debounced filtering to maintain a smooth interface even with large datasets. Compared to the original Dash app, Svelte's compiler produces smaller bundles and faster rendering, while TypeScript ensures strong type safety and clearer documentation.

On the backend, I will implement SvelteKit server routes that act as well-defined API endpoints. These routes will include input validation, parameterized queries, and secure HTTP headers. The switch to the Bun runtime offers faster startup times, lower memory usage, and a more efficient dependency system, creating a much more responsive environment than running a Python notebook server. I also plan to add automated testing using Vitest and Playwright, linting through ESLint, and dependency checks with Socket's bun-security-scanner to enforce secure and consistent code quality.



iii. *Explain how the planned enhancement will **demonstrate** specific **skills** and align with course outcomes.*

a. *Identify and describe the specific skills you will demonstrate that align with the course outcome.*

The skills I will demonstrate by implementing the enhancements in this category touch on all five of the course outcomes. First is the ability to design a robust, secure architecture and communicate it to stakeholders. By ensuring a clear separation of concerns, the architecture of the app will be modular, reusable, maintainable, and readable. Thorough inline documentation and types enable effortless communication to developers and technical stakeholders.

Second is the ability to evaluate and manage design trade-offs. Building a modern web application requires a developer to make countless decisions, many of which would be incredibly costly to change late in the development lifecycle. The choice between server- and client-side rendering, virtualized ("infinite") and paginated data views, or immediate fetch and debounced search can each have substantial impacts on a user's experience, and can scale with the application.

Third is the ability to use modern techniques and tools to deliver a secure application and value to stakeholders. According to Stack Overflow (2025), JavaScript remains the most popular language used by web developers in their annual developer survey, and modern workflows have lately seen a fast and steady transition towards the type safety provided by TypeScript thanks to its interoperability with JavaScript. The popularity of JS/TS has produced an abundance of frameworks and tools that improve the developer experience and produce faster applications. For this category's enhancements, I will be using Svelte, the related metaframework SvelteKit, and the JS/TS runtime Bun (alongside other tools that will be outlined in the other categories).

- Svelte is an incredibly performant reactive web framework that's trusted by companies like Apple, StackOverflow, and The New York Times. Compared to more popular frameworks, the syntax is much simpler—a Svelte file contains simply CSS, JavaScript, and lightly-templated HTML—and the compiled application has a much smaller footprint with even less overhead.
- SvelteKit, which is maintained by the same community, extends Svelte to the server-side to provide a holistic framework for building very performant web apps. It supplies features like routing, rendering, loading, basic state management, and more, and is used in the same way as Next.js is for a React application.
- Behind all of my enhancements will be Bun, a runtime environment and alternative to Node.js that aims to to offer full compatability with its APIs. Bun is written in Zig and uses Apple's JavaScriptCore (as opposed to Google's V8 that other runtimes use), considerably reducing startup times and memory consumption.

Additionally, the inclusion of tests for every aspect of the application—as well as dependency checking during CI/CD—help form an architecture with security as its foundation. Together, these tools and techniques will deliver secure software and value to all stakeholders—from developers and designers to managers and users.

b. *Select one or more of the course outcomes below that your enhancement will align with*.

These enhancements demonstrate the ability to design a robust and secure software architecture, evaluate and manage trade-offs in design decisions, and apply innovative tools to deliver stakeholder value. As mentioned above, I believe they align with all five program outcomes: collaboration through modular design (outcome one), professional-quality documentation and communication (outcome two), thoughtful problem-solving using best practices (outcome three), application of modern tools and frameworks (outcome four), and a persistent security mindset built into every layer of the system (outcome five).

B. **Category Two:** Algorithms and Data Structures

i. ***Select an artifact*** *that is* ***aligned with the*** *algorithms and data structures* ***category*** *and explain its origin.*

The artifact I've selected is the same as in category one.

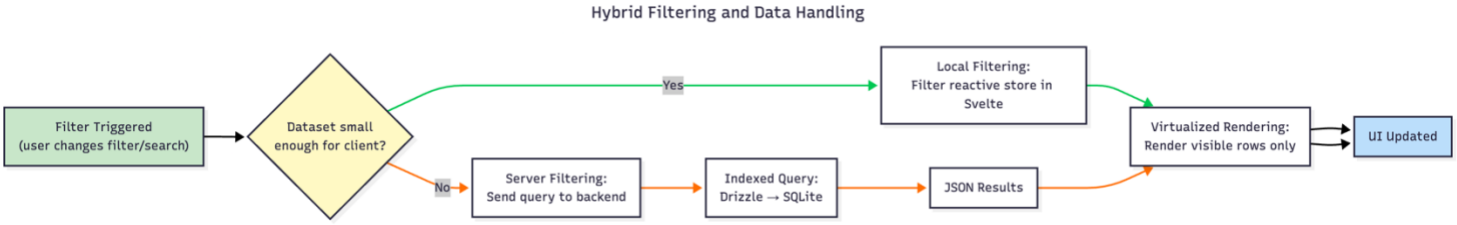
ii. ***Describe*** *a practical, well-illustrated* ***plan*** *for enhancement in alignment with the category, including a pseudocode or flowchart that illustrates the planned enhancement*.

With the same artifact serving as the basis for my algorithms and data structures enhancement, this phase focuses on how the application handles and transforms data efficiently. I will implement a hybrid data-filtering strategy that dynamically chooses between client-side and server-side filtering based on dataset size and context. For large datasets, filtering and pagination will occur server-side to reduce bandwidth and client load; for smaller datasets, results will be loaded once and filtered in-memory for maximum responsiveness. To prevent unnecessary rendering overhead, the data table will employ virtualization so that only visible rows are rendered at a time.

The switch to a statically typed environment in TypeScript and Drizzle inherently improves data structure efficiency, minimizing serialization overhead and reducing type-related bugs. Breed-distribution data will be preprocessed as lightweight cached views within the application layer rather than as separate database views, allowing the front end to receive compact, well-structured arrays of typed objects.

Rendering performance will improve further through Svelte's compile-time reactivity model, which eliminates the need for a virtual DOM and enables near-instant updates. Once the initial data load is complete, subsequent updates will modify a single reactive data object rather than triggering repeated API calls, reducing latency and improving

efficiency.



*iii. Explain how the planned enhancement will **demonstrate** specific **skills** and align with course outcomes.*

a. *Identify and describe the specific skills you will demonstrate to align with the course outcome.*

The skills I intend on demonstrating in this category are comparable to the skills I will demonstrate in the previous category. My first skill combines both thinking algorithmically and choosing appropriate data structures, which is the ability to anticipate the most efficient way to handle, store, and perform operations on data. My second skill is leveraging modern, innovative tools and techniques to achieve maximum performance against stakeholder value, which represents the ability to incorporate the most effective features of my workflow and produce a performant application. My third skill is to build security-first software, which is my ability to write secure code on every level of an application and integrate with tools to mitigate the risks of vulnerabilities from dependencies.

b. *Select one or more of the course outcomes listed under Category One that your enhancement will align with.*

The enhancements in this category align with three distinct course outcomes. Firstly, I'll showcase how to think algorithmically and select suitable data structures to achieve the third course outcome. In my application, this involves planning queries (parameterized and prepared statements) and virtualization, as well as employing more efficient sorting algorithms when presenting data to the client.

Secondly, I'll demonstrate the use of innovative techniques and tools to maximize performance and value, which directly relates to the fourth course outcome. This skill is exemplified by the implementation of hybrid rendering techniques, the technologies driving the application, asynchronous data transmission, and accessible compilation output (i.e., rendered HTML).

Lastly, I'll showcase the development of security-first software, which naturally aligns with the fifth course outcome. By prioritizing type safety throughout the application, validating all dynamic objects against predefined schemas, and

operating on data at the client-side, a security mindset will permeate every aspect of the data layer and business logic of the app.

C. **Category Three: Databases**

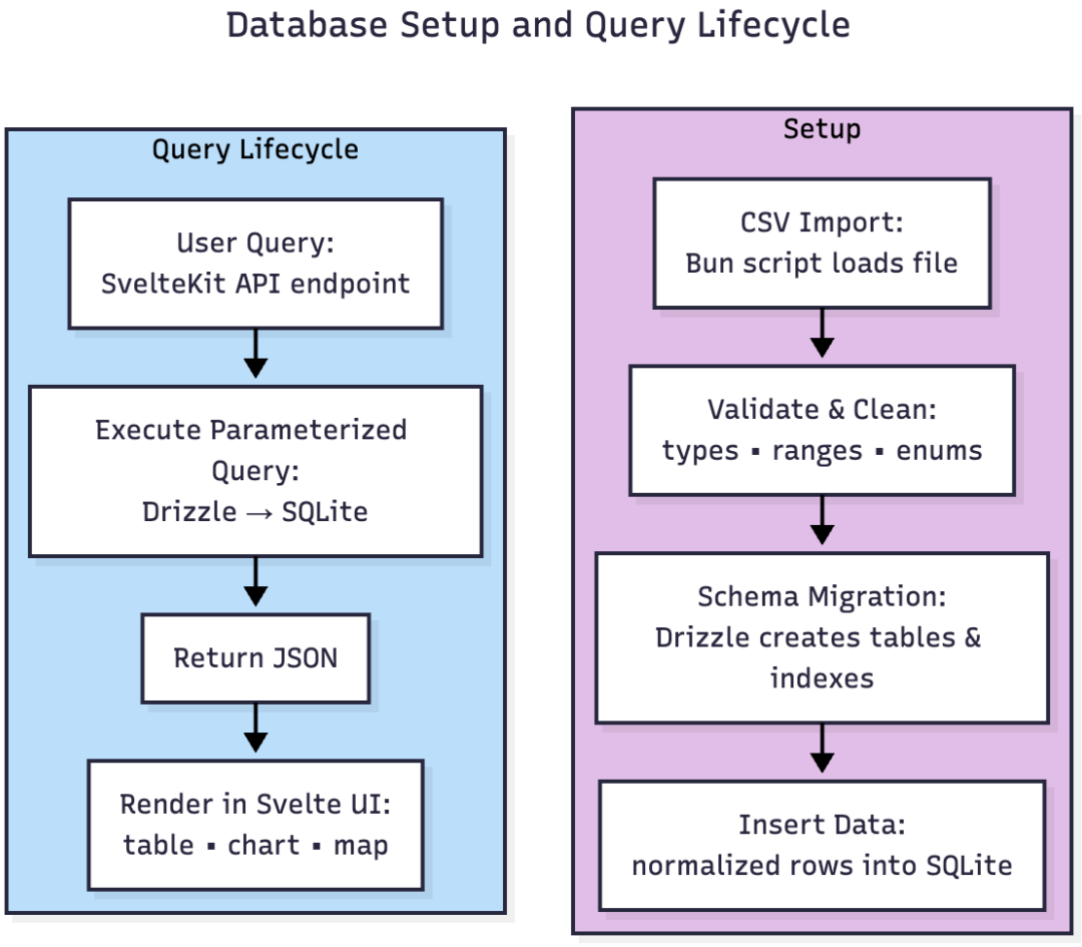
i. ***Select an artifact*** *that is* ***aligned with the*** *databases* ***category*** *and explain its origin.*

The artifact I've selected is the same as in category one.

ii. ***Describe*** *a practical, well-illustrated* ***plan*** *for enhancement in alignment with the category, including a pseudocode or flowchart that illustrates the planned enhancement.*

For the database category, I will continue using the same artifact but focus on refactoring its data layer to improve performance, portability, and security. The original dashboard used MongoDB running separately from the application, which added unnecessary complexity for such a simple dataset. For this enhancement, I will replace MongoDB with SQLite, using Drizzle ORM to define and manage the schema in code. SQLite's self-contained nature and its tight integration with the Bun runtime make it a high-performance, low-overhead choice for an application of this scale. SQLite, despite being an unconventional choice, has made significant strides in recent years with the development of drivers and the project itself. This has made it a viable solution for serving backend data on websites of this scale, and using an ORM like Drizzle facilitates porting to a different database should a project need to scale.

The database will be defined through a typed schema in Drizzle that enforces data integrity and validation automatically. The schema will include primary keys, foreign keys, and appropriate indexes for commonly filtered fields such as rescue type, breed, and status. Because the dataset originates from a CSV file, the import process will validate and normalize the data before inserting it into the tables. Each table (animals, breeds, rescue_types, locations) will include basic constraints such as non-null columns and type-checked enumerations—when feasible—to ensure the data remains consistent.



iii. *Explain how the planned enhancement will **demonstrate** specific **skills** and align with course outcomes.*

a. *Identify and describe the specific skills you will demonstrate that align with the course outcome.*

The planned database enhancement demonstrates multiple technical and professional skills, each closely tied to the program's outcomes. The first is the ability to design and implement an efficient, normalized database structure that promotes data integrity and consistency by using predefined responses for some queries and leveraging the benefits of the ORM.

The second demonstrated skill is the use of innovative tools and techniques to deliver a secure, high-value solution. Using Drizzle within the Bun environment not only modernizes the technology stack but also simplifies security and deployment workflows.

The third key skill is maintaining a security-first mindset throughout database

design and access, specifically through privilege enforcement, data validation, and using parameterized queries. The chosen technology stack facilitates this skill, as Drizzle's API is parameterized by default and works using only defined schemas.

Finally, the documentation and schema definitions themselves will demonstrate effective communication and collaboration skills by making the database design transparent and easy for other developers to understand and extend.

b. *Select one or more of the course outcomes listed under Category One that your enhancement will align with.*

By defining relationships, constraints, and indexes directly within the Drizzle schema, I will show an understanding of relational database principles and how they improve query performance and data reliability. This directly supports the third course outcome, which emphasizes designing and evaluating computing solutions using appropriate standards and principles while managing trade-offs. Parameterized queries, strong typing, and schema-defined validation ensure that no unsafe data is written or retrieved, aligning this work with the fourth outcome, which highlights the implementation of innovative techniques and tools to deliver industry-relevant computing solutions.

By enforcing least-privilege access (read-only roles for standard operations), validating all imported data, and ensuring parameterized queries, I will show a sound understanding of secure data handling practices. These design choices relate to the fifth outcome, which focuses on developing a proactive security mindset that anticipates vulnerabilities and mitigates risks.

Lastly, the clarity of the schema files, comments (inline and block-level), and the migration scripts to convert CSV to a SQLite database serve as professional-grade documentation that supports collaboration and aligns with the first and second course outcomes.

IV. ePortfolio Overall Skill Set

*A. Accurately describe the **skill set** to be illustrated by the **ePortfolio overall**.*

i. *Skills and outcomes planned to be illustrated in the code review*

The code review portion of my ePortfolio will highlight my ability to design secure, maintainable software using modern frameworks and practices. It will show my understanding of modular architecture, type-safe programming, and efficient database design through the reimplementation of the dashboard using TypeScript, Svelte/SvelteKit, and SQLite. In my review, I will emphasize clean and readable code, adherence to security principles, and thoughtful use of tools like Drizzle and the Bun runtime. This portion will primarily reflect my mastery of course outcomes three through five: designing strong computing solutions, applying innovative technologies, and maintaining a proactive security mindset.

ii. *Skills and outcomes planned to be illustrated in the narratives*

The narratives will demonstrate my ability to communicate complex technical concepts clearly and professionally by explaining the reasoning behind each major enhancement, the trade-offs considered, and the ways the new implementation improves upon the original artifact. These reflections will show my capacity to evaluate problems critically, articulate solutions for both technical and non-technical audiences, and connect design decisions to real-world value. Together, they align most closely with outcomes one, two, and three by illustrating collaborative communication, professional writing, and analytical thinking in software design.

iii. *Skills and outcomes planned to be illustrated in the professional self-assessment*

The professional self-assessment will focus on how this project represents the culmination of my learning throughout the Computer Science program. It will connect the technical growth I've demonstrated (secure coding, efficient data handling, and strong system design) to my personal and professional goals. The assessment will also discuss how the project reflects my readiness to pursue more advanced education and work, and how maintaining a security-first mindset will continue to shape my work going forward. This section will primarily address outcomes four and five, reinforcing my ability to apply innovative tools and uphold security as a fundamental aspect of every computing solution.

References

Stack Overflow. (2025). *Technology: Most popular technologies*. Stack Overflow 2025 Developer Survey. https://survey.stackoverflow.co/2025/technology#1-programming-scripting-and-markup-languages