

CS-499-R4904 Computer Science Capstone 24EW4

CS499 Module 7-1: Computer Science Capstone Reflection

Instructor: Professor Brooke

Southern New Hampshire University

Stephen Owusu-Agyekum

April 20, 2024

Purpose

This paper reflects my final submission, the CS499 Computer Science Capstone course. It reflects on my journey throughout the computer science program. It also reflects on the artifacts I selected that meet the three categories in Software Engineering and Design, Algorithms and Data Structures, and Databases. I elaborated on how the original artifacts were enhanced and the changes made to the original artifacts. It also discusses all the five major computer science course outcomes and how I achieved them in each of the three categories. I discussed how including the artifacts in my ePortfolio helped me achieve the five-course goals. I created a professional ePortfolio showcasing my computer science program knowledge, skills, talents, and abilities.

Prompt

A professionally designed portfolio highlighting my strengths and abilities is one of the best ways to communicate visually and show prospective employers my value. The ePortfolio's content is a good foundation for concepts demonstrating my abilities in several important computer science domains. These artifacts demonstrate my technical proficiency and highlight my problem-solving capabilities and creative approach to challenges. Each project reflects my dedication to mastering industry-relevant skills and my commitment to delivering high-quality solutions. When you get into my ePortfolio, I've integrated artifacts from pivotal courses in the Computer Science program, illustrating my growth in essential areas such as databases, algorithms and data structures, and software design/engineering. Through this comprehensive portfolio, I aim to present myself as a versatile and resourceful computer scientist ready to make a meaningful impact in the field.

My ePortfolio is a culmination of the skills and expertise acquired during my Computer Science journey at Southern New Hampshire University (SNHU). Pages on my ePortfolio contain

a carefully constructed portfolio that showcases my skills through well-written and appealing content. To ensure consistency, technical accuracy, and relevance in my ePortfolio, I ensured that each part of the portfolio was customized to align with its intended audience and context. This portfolio stands as a testament to my dedication and passion for the field of computer science. It encapsulates countless hours of learning, experimentation, and refinement, resulting in a comprehensive showcase of my capabilities. As I continue to evolve and expand my skills, this ePortfolio will serve as a dynamic reflection of my ongoing journey in the ever-changing landscape of technology.

During my journey through the computer science program, I've experienced a transformative process of growth and learning. Each step has shaped my understanding and proficiency in the field, from the foundational courses in programming and data structures to the more advanced topics in algorithms, software engineering, and database management. Along the way, I've tackled challenging assignments, collaborated with peers on innovative projects, and delved into real-world applications of theoretical concepts.

As I progressed through the program, I encountered diverse opportunities to explore specialized areas of interest, such as artificial intelligence, cybersecurity, and mobile application development. These experiences not only deepened my knowledge but also broadened my perspective on the vast possibilities within computer science. Reaching the capstone course, CS499 marks a significant milestone in my academic journey. It represents the culmination of years of hard work, dedication, and perseverance. Here, I have the chance to apply all that I've learned in a comprehensive project that integrates various facets of computer science into a cohesive whole.

Before the enhancement of the artifact, I performed a code review, which serves as a systematic method for ensuring the quality, reliability, and maintainability of software artifacts. It provides a structured approach to identifying errors, weaknesses, and vulnerabilities in the codebase, facilitating their timely resolution. This process gave me valuable insights into best practices and strategies for enhancing software quality and promoting collaborative development environments. Some of the benefits of performing code review are that it improves the team's skills, identifies faults early on when they can be easily fixed, and gives the company or person a certain amount of fault tolerance, which helps them recover more quickly from unexpected changes.

Engaging in code reviews provided an opportunity to apply strategies for fostering collaborative environments within the team. Analyzing the selected artifacts for software design, algorithms, and databases gave me insights into enhancing existing functionalities and addressing potential weaknesses or vulnerabilities. The code review videos served as a valuable tool for communication, enabling me to articulate my observations and proposed enhancements effectively.

Through these videos, I enhanced my abilities to deliver professional-quality oral, written, and visual communication tailored to the specific needs of diverse audiences and contexts. The code review experience has been instrumental in refining my skills and contributing to the project's overall success. Moreover, the code review process cultivates a culture of continuous improvement, promoting knowledge sharing and skill development. This fosters personal growth and proficiency, enhancing project outcomes and streamlined development practices. As a result, the codebase becomes more robust, reducing risks and ensuring the project's overall success and efficiency.

Category One: Software Engineering and Design

I selected the computer science course CS370 Current/Emerging Trends in CS for the Software Engineering and Design category. I showcased my proficiency in employing established and inventive methodologies, abilities, and tools within computing practices to develop computer solutions tailored to meet industry-specific objectives and deliver tangible value. By creating deep Q-learning algorithms to train an intelligent agent (Pirate) to hunt for a hidden treasure, I showcased my proficiency in applying advanced computational techniques to solve complex problems.

The development process involved meticulous problem-solving, system architecture design, and optimization to ensure the agent navigates the maze efficiently while adhering to industry standards and best practices. Additionally, I leveraged innovative approaches, such as dynamic adjustment of exploration factors and experience replay mechanisms, to enhance the effectiveness and robustness of the solution. Through this project, I demonstrated my technical skills and ability to deliver solutions that meet industry-specific requirements and contribute value to real-world applications.

Enhancing and modifying the provided artifact has been a comprehensive process encompassing various aspects of software development. Initially, I evaluated the existing code to pinpoint areas for improvement, considering functionality, structure, and performance. Implementing planned changes involved writing new code, modifying existing code, and refactoring to enhance organization and structure while adhering to best practices and coding standards. Throughout this process, I encountered challenges, such as installation errors with Python variables for Jupyter Notebook, which required troubleshooting and the installation of

Anaconda Navigator as a workaround. Additionally, debugging errors within the code post-update posed challenges but provided valuable learning experiences.

The enhancements to the artifact improved its functionality and deepened my understanding of reinforcement learning concepts. Working with the Q-learning algorithm provided insights into how agents navigate environments to maximize rewards. Furthermore, applying design patterns like experience replay memory underscored the importance of efficiency in coding practices. Engaging in this project facilitated hands-on experience in reinforcement learning, software design, testing, and collaboration. These experiences honed my technical skills and emphasized the iterative nature of software development and the significance of clear communication, modular design, and continuous improvement. Moreover, working with reinforcement learning algorithms strengthened algorithmic thinking, problem-solving abilities, and essential software design and engineering skills.

Through the enhancement, I could design and evaluate computing solutions that solve a given problem using algorithmic principles, computer science practices, and standards appropriate to its solution while managing the trade-offs involved in design choices. This is demonstrated by improving the application of the Q-learning algorithm in the `qtrain` function, which complies with accepted algorithmic norms and principles in the field of reinforcement learning. This shows how well I can create computer solutions using strict algorithmic principles.

I employed strategies for creating cooperative settings that let many audiences contribute to organizational decision-making in the field of computer science. This goal is accomplished by using descriptive variable names like `win_rate`, `max_memory`, and `n_epoch`, combined with thorough documentation that encourages teamwork by making the codebase easily understandable

and customizable by various engineers. This improves code readability and lessens ambiguity, which encourages cooperation.

Additionally, the code is organized into distinct modules with well-defined roles, such as "TreasureMaze.py," "GameExperience.py," and "qtrain Function," which allow several developers to work on different components concurrently and independently of one another. This encourages teamwork and makes codebase creation and maintenance more efficient.

I showcased my ability to apply creative and well-founded methods, abilities, and resources in computer practices to implement computer solutions that meet industry-specific objectives and add value. Using the GameExperience class and the experience replay mechanism, I obtained this result through code improvement, which shows how to use a reliable reinforcement learning technique. Using tried-and-true methods with skill improves model learning efficiency.

Category Two: Algorithms and Data Structures

I selected the computer science course CS360 Mobile Architecture and Programming for the Algorithm and Data Structures category. The chosen artifact revolves around Algorithms and Data Structures, encompassing a comprehensive algorithm design and implementation process. It involves thinking about how users will interact with the program's different algorithms and data structures. The focus is ensuring the algorithms, and their features are used intuitively, utilizing industry-standard iconography and symbols to provide a smooth user experience. The artifact, which focuses on algorithms and Data Structures, entails putting engineering and design techniques into practice for data input validation, planning for security, and taking automatic refuse access. Additionally, it involves designing algorithms and data structures to maintain functionality

and linkages between various classes, functions, and database interactions inside the app's layouts and code base.

I developed a security mindset that anticipates adversarial exploits in software architecture and designs to expose potential vulnerabilities, mitigate design flaws, and ensure privacy and enhanced security of data and resources. This is accomplished by enhancing the artifact by adding strong user authentication procedures that confirm users' identities before allowing access to sensitive features. This includes adding login and account creation processes to the app layouts and features like password validation and secure session management to stop unwanted access to user accounts. Passwords and other sensitive user data are safely saved in the SQLite database of the upgraded code artifact. To mitigate potential vulnerabilities in the software design, SQL injection attacks are prevented by using prepared statements and parameterized queries in database interactions.

I employed strategies for building collaborative environments that enable diverse audiences to support organizational decision-making in computer science. This is demonstrated by fostering open communication and collaboration among team members from diverse backgrounds, including deep online research and feedback from my instructor. This allowed everyone to share their perspectives, ideas, and expertise, contributing to more informed decision-making processes. I sought feedback from stakeholders and users throughout the development process to ensure our decisions aligned with organizational goals and user needs.

I demonstrated optimization, time complexity (BigO notation), and efficiency of the algorithmic logic in the code. The BigO notation for this implemented algorithm is $O(1)$ for operations like `createUser()` and `deleteItem()` because they involve direct database access and manipulation of individual user records, resulting in constant time complexity regardless of the

size of the database. The class AD_SMSNotification provides a method to create an AlertDialog with two buttons for enabling or disabling SMS notifications. The static method efficiently constructs the dialog with a time complexity of $O(1)$. It utilizes string and drawable resources for localization and consistency across the application.

It is optimized, as shown by the careful handling of database transactions in methods like `createUser()` and `updateUser()`, which minimize the number of database operations required to perform user-related tasks, leading to improved efficiency and reduced resource consumption. The implemented algorithm is optimized, as shown by the efficient database operations, and streamlined logic in the code.

I designed and evaluated computing solutions that solve a given problem using algorithmic principles and computer science practices. For instance, I applied the user-centered approach. I followed the principles of identifying problems, designing algorithms tailored to those problems, implementing them effectively into code, and rigorously testing and refining the solutions. The code implements algorithms for database operations such as creating, reading, updating, and deleting user records. For example, methods like `createUser()` and `updateUser()` employ algorithms to manipulate user data efficiently within the SQLite database. These algorithms are designed to optimize database transactions and ensure the smooth operation of the app.

Category Three: Database

I selected the computer science course CS465 Full Stack Development I for the Databases category. The artifact selected for the databases is the **Traveler Website**. This traveler website was created during my computer science course. The website was created with the JavaScript application, which uses the Node.js runtime environment, the Express back-end framework, and the Angular front-end framework. Additionally, I organized the content on the trip website pages

using HTML and CSS. An application that uses Angular for the front end and Node.js for the back end is known as a MEAN (MongoDB, Express.js, Angular, Node.js) stack application. Mongoose is used for database operations, and it also integrates with MongoDB.

I added the traveler website artifact to my ePortfolio because it demonstrates several valuable skills and competencies I have in databases for potential employers, recruiters, or educational institutions. Also, I added this artifact to my ePortfolio because I would like to effectively showcase my technical skills, problem-solving abilities, and creativity to potential employers to increase my chances of hiring.

Users are prompted to open their browsers and navigate to <http://localhost:4200/> when the travel website loads or runs. The Angular Live Development Server starts listening on localhost:4200. The server generates the files required to operate the web application upon successful compilation properly. In the meantime, nodemon begins watching changes in the terminal and restarts the server based on those changes. To obtain trip information, the application sends HTTP GET requests as it retrieves data from the MongoDB database when the server is operating. After each request is processed, the associated trips are retrieved and put on the website.

I designed and evaluated computing solutions through the enhancement by developing the CRUD module for handling database records. This required strategic decision-making and adherence to computer science practices and standards, demonstrating my ability to make informed choices and manage design trade-offs. I achieved this objective by carefully designing the Create, Read, Update, and Delete (CRUD) module to efficiently handle database operations such as creating, reading, updating, and deleting records.

I designed, developed, and presented oral, written, and visual communications of the highest caliber that are technically sound, logical, and well-tailored to the target audiences and

situations. This was achieved by efficiently documenting the travel website's development process, which included brief explanations of the system architecture, codebase, and implementation specifics. For example, the README.md file in the traveler website's repository provides comprehensive documentation outlining the project's purpose, features, setup instructions, and usage guidelines. This documentation is written clearly and concisely, making it easy for stakeholders, collaborators, or future developers to understand the project and its components. Furthermore, comments in the source code act as documentation and outline the capabilities and uses of particular classes, functions, and modules.

The enhancement of the artifact aligns closely with the course objective of developing a security mindset in software development, as it encompasses various security measures and considerations aimed at safeguarding data and resources against potential threats and exploits. For example, I improved data and resource security, enhanced data validation, secured database access, encouraged code reuse, and modified security protocols within the travel website. Also, the website protects user credentials that are kept in the database using secure password storage methods like salting and hashing. The website ensures that the actual passwords are safe and difficult to decrypt even if the database is stolen by employing these encryption techniques. As I improved the artifact, I gained a deeper understanding of web development technologies such as Angular, Node.js, and MongoDB by working with them extensively during the project.

I demonstrated my ability to use well-founded and innovative techniques, skills, and tools in computing practices to implement computer solutions that deliver value and accomplish industry-specific goals. This is achieved by leveraging specific modern web development tools and frameworks, such as Angular for the front-end and Node.js with Express for the backend. By incorporating industry-standard practices and innovative techniques, I created a functional and

user-friendly travel website that meets the specific goals of the travel industry, showcasing my mastery of these modern technologies. Additionally, I learned the importance of thorough documentation and clear code commenting to facilitate future development and troubleshooting. I also honed my skills in debugging and problem-solving as I encountered and resolved various issues during the development process.

In summary, my CS499 Computer Science Capstone course demonstrates the culmination of my journey through the computer science program. It showcases my growth and expertise across software engineering and design, algorithms and data structures, and databases while highlighting my ability to apply innovative techniques to solve industry-specific problems. Through the enhancement of artifacts and the rigorous evaluation of computing solutions, I've honed my skills and created a professional ePortfolio that reflects my readiness to make a meaningful impact in computer science.

References:

MongoDB. (n.d.). *MongoDB Atlas: Cloud Document Database*.

<https://www.mongodb.com/lp/cloud/atlas/try4?utm>

CSS Design Awards. (n.d.). *CSS Design Awards - Website Awards - Best Web Design*

Inspiration - CSS Awards. <https://www.cssdesignawards.com/>

Southern New Hampshire University. (2024, April 20). Module 7-1 Final Project Submission.

Retrieved from Module Seven: Professionalization:

<https://learn.snhu.edu/d2l/le/content/1535837/viewContent/31021136/View>