7-1 FINAL

7-1 Final Submission: Professional Self-Assessment

CS-499-12473-M01 Computer Science Capstone 2024 C-5 (Sept-Oct)

Michael Zietz

Southern New Hampshire University

October 20, 2024

This paper reflects my journey through the CS499 Computer Science Capstone course assignments. I explore how I achieved the five-course outcomes by enhancing the artifacts included in my ePortfolio, culminating in the creation of a professional portfolio that showcases my knowledge, skills, and abilities from the computer science program. Crafting a professional portfolio that highlights my unique talents and skills is one of the most effective visual communication tools to demonstrate our value to potential employers. The content of the ePortfolio serves as a solid starting point to clearly showcase my skills across key fields in computer science. The included artifacts from various courses in this program as well as a personal project’s inclusion illustrate my growth in areas such as software design and engineering, algorithms and data structures, and databases.

My ePortfolio integrates the knowledge and skills I have gained and developed throughout my years in the Computer Science program at SNHU, representing my academic progress and honor rolls from high-quality outcomes.

Despite the artifacts in this ePortfolio being developed and enhanced solely by myself. It is important to perceive these projects through the lenses of a professional in the computer science field. Here, a greater emphasis is placed on not just coding and troubleshooting skills but also communication with fellow coworkers and even stakeholders in a professional setting. This has been emulated in my ePortfolio through the deliberate creation of code reviews and descriptive narratives for each of the artifact enhancements. In the work force, I will be required to communicate key features to stakeholders like users or clients to incentivize the necessary features I am able to develop as well as critically reviewing a code base without bias and create robust arguments where possibly some areas are lacking and in need of enhancement without coming across as hostile or deconstructive. Ultimately, the main priority is to produce the most effective, secure, and efficient software of which I am capable which means realizing and following through when tough problems are placed in the backlog.

In addition to the soft skills mentioned, hard skills requiring critical thinking and problem solving are also necessary to compose a well-rounded developer. Of these hard skills, particularly when discussing software development, knowledge of data structures, algorithms, software engineering, databases, and security are all vital areas to focus on. Through my studies, I have learned that data structures comprise the knowledge and proper utilization of the various data types, both on a primitive and nonprimitive, when to use each structure properly, and how to format them given the language and development environment selected for a product. Algorithms are the logic that binds these data structures together to perform data-manipulating tasks that serve a function and require knowledge on how to organize them, so they can be understood and debugged. Software engineering is the marrying of these two previous ideas to create fully realized programs that serve a designed purpose for an intended audience. Databases allow for long-term data storage that can be created, read, updated, or deleted by a program, and it persists when the software application is not running. Lastly and most importantly, security considers all the previous ideas and locks them down so only specified users or tools have access to specified data or features. This requires a consistent effort to learn and implement current industry standards, how they can be implemented or maintained in a code base, and what specific security features are critical to emphasize depending on the program and circumstances. Constant threats of attacks from external and internal malicious parties are necessary for consideration. (Secure Development, 2017) For example, a software application that utilizes Internet of Things in congruence with a SQL database will need to implement security measures to defend against online attacks like SQL injection to protect sensitive company or user data and prevent unauthorized parties from gaining access to it. (W3Schools, 2024)

Fully realized, my ePortfolio combines both the necessary soft and hard skills that I have gained or cultivated through my studies at SNHU and are ultimately depicted by the artifacts that I have selected and enhanced. Through my ePortfolio, I have designed, developed, and delivered a professional-quality demonstration of my capabilities and abilities that are coherent, technically sound, and appropriately adapted to a specific audience and context.

The first of these artifacts is a project I have developed, I created in Python independent of my courses at SNHU to enhance. This artifact served as my first dive into studying Python before returning to higher education and completing my bachelor’s in computer science. The project spans three files with a main class and two subclasses. The current work can replicate a simplified version of classic table-top gaming combat within a Python-based terminal. It records user input to create one or more characters and weapons with unique properties and simulates the combination of character and weapon damage during their turn using rule systems inspired by Pathfinder by Paizo. Through my enhancements I have recreated this Python project as a Java mobile app using Android Studio to improve the artifact’s base functionality through implementing new features, correcting known issues, and improving both the error handling and comments. New feature additions include opponents, with their own sets of properties, for the characters to simulate combat, and a rounds system to challenge the user’s character with procedurally more difficult enemies to face. New screens for each activity needed to be created as the original artifact solely ran through a python terminal. During combat, the user can fight or flee, and once the user is defeated in combat or flees their first time the simulation ends. The user is then directed to a new screen that displays a permanent record of the user’s characters, how many rounds they were victorious, and how much damage they were able to deal. This final improvement I added required me to implement and maintain a local SQLite database stored on the phone creating a permanent record outside the application’s run instance that saves the user’s recorded attempts. I utilized this database to create a separate collection for recording and authenticating locally created and stored users that first have to sign into the app before accessing features.

The second artifact takes a web-based application, named Travlr Getaways, that I created for a previous course, CS 465 Full Stack Development, and hosts a local platform where vacation trips are packaged and published for travel agents to add and modify. Its original state was the final submission for the class, and it utilizes the MEAN stack, which includes MongoDB, Express JavaScript, Angular JavaScript, and Node JavaScript. Angular serves as the client-side framework written in JS, while Express is used as the framework for the backend, and MongoDB functions as the database. Node, also written in JavaScript, acts as the primary web server framework. In its current state, Travlr’s web application is capable of hosting these travel pack ages, allowing non existing users to view trip details, but requiring authenticated sign in to edit existing or create new trip packages. This artifact was selected with the intention to correct known functionality issues in addition to new features that create purpose for new users outside of the current one designed for a trip agent. I sought to improve the clarity and structure of the project as well as correcting some lingering issues left after its original submission. Due to deprecated use of login authorization in the project’s final version, this left an issue with users not being able to submit POST commands to create new or edit existing trip objects after signing in. I wanted to implement current tools and APIs used to perform this authentication and enable full functionality to the website developed. I implemented booking reservations and reconciling availabilities. In the previous state, the website was only able to take advantage of a small portion of its promised features and does not separate different types of users signing in. I sought to introduce user roles, those intended to edit trip and those intended to book them, and allow functionality related to them. The website will also need to have features implemented to allow booking once all conditions allow it.

The third and final artifact was developed for a previous course, CS-340 Client/Server Development, at SNHU. I have selected this artifact to enhance because the successful recreation of this application initially designed for Linux OS demonstrates my understanding of the Python language as well as Dash, skills deploying MongoDB and CRUD functionality, and ability to reproduce and recreate scripts inside Jupyter Notebook. The client, Grazioso Salvare, needed to interact with their existing data from animal shelters. This artifact intended to identify and categorize animals which have been entered into the system based on their criteria. Grazioso Salvare needed to use this software to identify animal profiles to train for various tasks, and they have requested this project be open source and accessible on GitHub to aid other, similar organizations. Rather than manually sorting through each database, which can take tedious and time-consuming effort, this software will reduce this human waste by requiring simple inputs to retrieve data from databases quickly. After the enhancements it can create new entries, read and search for existing ones, update entries, and delete them as well. I sought to enhance this application’s current GUI, as its pie chart often results in unappealing visuals that clutter the screen, migrate the project, as it was initially created for a virtual Linux-based lab environment, named Apporto, and recreate it in a Windows OS environment. This required me to install and coordinate compatible versions of Python and MongoDB. For this application, the PyMongo library needs to be utilized in order to successfully connect Dash’s capabilities with MongoDB. The recreated application includes authentication before allowing access to CRUD features as only specified individuals should be granted permissions to modify stored data.

The intention behind selecting each of these artifacts to enhance is to showcase my personal growth as a developer from my time spent studying in Southern New Hampshire University’s computer science program, that I have practiced and cultivated necessary soft and hard skills required by a developer, and depicted how I have completed the necessary course outcomes for this capstone to pave a path forward for me in the industry.

I maintain that my ePortfolio highlights my ability to apply well-established and innovative techniques. I created computer solutions that deliver value and meet industry-specific goals, illustrated through features such as importing CSV data files into MongoDB, integrating dependencies like the Python PyMongo driver, Python libraries, the Dash framework, and Python source code, and utilizing a CRUD module to manage data imported into MongoDB. I designed and assessed computing solutions to solve problems using algorithmic principles and computer science practices and standards suitable to the task, while balancing the trade-offs involved in design choices, considering relationships and functionality between different classes and methods by using arguments, parameters, and variables in scope in JAVA programming. All the enhanced artifacts showcase my focus on applying a security mindset during development to anticipate potential exploits in software architecture and design, address design flaws, and ensure the privacy and enhanced security of data and resources.

Resources

‌ *Secure Development | Software Engineering Institute*. (2017). Www.sei.cmu.edu. https://www.sei.cmu.edu/our-work/secure-development/

W3Schools. (2024). *SQL Injection*. W3schools.com. https://www.w3schools.com/sql/sql\_injection.asp

‌

‌