Professional Self-Assessment

Zane Russell Brown

Southern New Hampshire University

CS-499 Computer Science Capstone

GitHub Pages: https://github.com/BeardedArtist

Computer Science Program & ePortfolio

Hello, my name is Zane Brown, and this is my self-assessment on my computer science program and ePortfolio. The coursework that I have completed throughout my program has allowed me to experience a wide range of fields in computer science. These fields include programming, development lifecycles, mathematics, data mining, client-server development, software reverse engineering, applied statistics, and much more. Each of these courses provided me with essential skills that has prepared me for a career in computer science.

In this self-assessment, we will be discussing my experience and skills in collaboration, communications with stakeholders, data structures and algorithms, software engineering and databases and security. Each of these sections were covered in my program through various courses in programming, development lifecycles and development theories. In these courses, I had the chance to experience real world scenarios and create real-world projects that will help me create even more complex programs in my career.

Collaborating in a team environment

Collaboration in a team environment is vital for any position in computer science. Without collaboration, a development team cannot effectively and efficiently create a program that meets quality standards. Collaboration provides many benefits such as organization, peer learning, peer review, standardized coding, reduced vulnerabilities etc. Effective collaboration can ensure that a program is created with the best quality, least number of bugs and vulnerabilities and strengthen everyone on the development team.

The courses I took for my computer science degree has given me a lot of experience in collaboration in a team environment. Courses such as client-server development, collaboration

and team project, development lifecycle and mobile architecture and programming allowed us to work as a team to develop projects and documentation. These collaborations gradually gave me better experiences and allowed me to learn how to best communicate with my team and leader to produce best results in the program or project.

The collaboration and team project course gave me experience using version control tools when developing projects. Here, we were placed in a group and were tasked to create a project together and use version control tools to update the project using branches. We learned that only using version control tools in a group is not enough to develop a project. Communication was also vital and became commonplace for our group. Every week we would have two team meetings. One on Monday and one on Friday. Here we would discuss what changes were made to the program and what new features were added. Once we completed our meetings, we would update our project using version control tools.

Other courses such as mobile architecture and programming, and client-server development did not require us to work in a team environment. However, due to past experiences with team collaboration, many of us, including myself, ended up working with each other to produce top quality projects. One project that is included in this ePortfolio was my Inventory App application. During its development, I worked with several other students. Although we were not working on the same project, collaboration helped each of us become better developers. If we had any issue, we would communicate with each other to find a solution. This not only helped me complete my program earlier than expected, but it allowed me to gain more skills and experience than if I did the project alone. This communication helped me, and my fellow peers become much better developers.

Communicating to stakeholders

Communication is a vital part of any development team. However, communication allows spans more than just the development team. In many cases, development is needed for both clients and stakeholders. Client and stakeholder collaboration and communication is a necessary part of the development process. This communication helps development teams understand what the client wants in a program as well as allows the client to make any necessary changes if needed. A member of the development team is usually in charge of this communication and informs both parties of any changes or updates.

In my courses, we had a number of chances to experience this type of communication with stakeholders and clients. In many cases, the course created situations where we needed to read documentation provided from the client and create projects from those documents. In other cases, we were given constant communication with fictitious stakeholders that would constantly make changes to the program they wanted. This provided us with real-world situation and required us to stay alert and make the necessary changes quickly.

Two courses that provided us with the best real-world example was the development lifecycle and data mining. The development lifecycle course was broken up into two different sections that allowed us to experience different types of development methodologies. However, in both sections we were given client documentation that informed us what project to create. During development of these mini projects, the client would constantly update us of changes or new features that they wanted. These changes would come weekly, and we would need to quickly implement any features or changes to the program. This course gave the best real-world experience with this type of communication and gave me skills in quickly adapting to any situation.

Finally, the data mining course gave a similar experience with client/stakeholder communication. In this course, we were tasked to solve an issue that a fictitious company was experiencing in online sales. At the beginning of the course, we were given a database warehouse to analyze in order to find a possible solution to the issue. As the course progressed, the client gave us additional data to analyze and run against previous data to find additional patterns and correlations. This course, similarly, to the previous mentioned course, gave me a lot of experience with a constantly changing environment. These skills will be vital for any future career in computer science.

Data structures and Algorithms

Understanding data structures and algorithms is a key aspect to any future in the computer science career. Algorithms are now used around the world in almost anything you can think of. Algorithms are used in big data, crime tracking programs, phones, computers, and ATM's. The list goes on and on, but the main point is that algorithms are what make our modern world work. Thus, understanding these algorithms and the data structures that make them work is vital information for any individual.

Data structures and algorithms has been a main focus during my computer science degree journey. Programming courses main focus usually fall under data structures and algorithms and therefore I have had quite a bit of experience in this field. In fact, I had a course named data structures and algorithms at the start of my journey that helped introduce me to this topic. This course covered a number of topics such as vectors, hash tables, tree structures, searching for data, sorting data, chaining, etc. This starting course gave me a great understanding a base for future classes that I would take.

As my time in computer science progressed, I gained additional knowledge in data structures and algorithms. Courses like mobile architecture & programming, emerging system architecture & technology, OpenGL, secure coding, software reverse engineering, etc. gave me many chances to experience and work with these various data structures and algorithms. Secure coding and mobile architecture allowed me to gain experience with c++ and java and I was able to create several projects in these fields. Secure coding and reverse engineering gave me a chance to not only experience algorithms and data structures, but also gave me insight to the inner workings of a program and how these algorithms work from the inside out.

As I stated previously, understanding how algorithms and data structures work and how they are created is a vital skill in computer science careers. The courses that I have taken have helped me gain the skills necessary to not only understand these concepts, but to also create them. These skills will be vital in my future careers.

Software Engineering and Databases

Software engineering and databases is another important aspect to computer science careers. An individual involved in computer science will eventually deal with software engineering and databases at some point and therefore should have a good understanding of how they work. Databases are also used in nearly every modern program, so an understanding of how to create databases is also vital.

Software engineering encompasses a number of topics in computer science. During my time in computer science courses, I have learned a lot about what software engineering is and why it is important. One of the first courses that I took was development lifecycles. This class taught us the different types of software engineering methodologies, such as waterfall and agile

models. This introduction to software engineering gave me an understanding of the different

types of development models and what model should be used in certain situations. I used this

knowledge throughout my time at SNHU to help me develop projects in an organized and timely

manner. In most cases, I preferred to use the agile model. This was because I liked the sprints

that the agile model offers. Constant development and testing cycles allowed for better

organization and improved quality.

Additionally, courses such as application development, client-server development and

emerging system architecture & technology allowed me to practice these software engineering

theories that I have learned. Through these courses, I have learned what theories and

methodologies work for me and what don't. This understanding of software engineering will

help me become a much more productive, organized and skilled programmer for future careers.

Databases was also another important part of my computer science journey. Databases

became especially important in the secure coding, and mobile architecture courses. The secure

coding course gave us many chances to experience different types of vulnerabilities. One specific

vulnerability that comes to mind was SQL injections. This vulnerability was especially present in

username/password databases and we were tasked to fix and ensure that this issue did not happen

again. The mobile architecture course also delt with databases. The Inventory App that is found

in this ePortfolio uses two different databases. One for usernames and passwords and one to hold

user input items. The creation of these two databases gave me a lot of insight into how complex

and important databases are.

Security

Security in computer science is an ever-growing field of study. Every year security in programs is becoming more and more vital and this field is also becoming the most sought-after job. Security in computer science aims to help protect programs, users and companies from malicious users or viruses. However, this becomes more difficult due to the accelerated rate at which technology is advancing. Therefore, security is one of the most important fields in computer science.

Security was one of the ever-present topics in my computer science journey. Professors always stressed how important security is and therefore we were always tasked to test our programs and check for any errors or vulnerabilities. I gained a lot of experience and knowledge of security in programs from my secure coding course. In this course we learned about numeric overflows, buffer overflows, SQL injections, exceptions, unit testing, encryption, etc. Throughout the course we worked on programs that contained vulnerabilities. We then implemented security measures to help protect these programs from specific attacks. We also created a security policy for a fictitious company in order to help the development team follow a strict development policy. This course gave me the skills and understanding about secure coding and these skills will be one of the most important for my future careers.

Artifacts in ePortfolio

In order to help showcase my full range of computer science talents and abilities, three artifacts have been chosen for this ePortfolio. Each of these artifacts have been chosen because they represent the three key concepts in computer science, software engineering & design, algorithms and data structures, and databases. The artifacts found in this ePortfolio include the Inventory App application created in the mobile architecture & programming course, the data analysis report created in the data mining course and the reverse engineering project created in the software reverse engineering course.

Mobile Architecture & Programming

The first artifact in the ePortfolio is the Inventory App. This artifact is one of the best additions to this ePortfolio. This artifact fits into all three categories. Software engineering/design, algorithms and data structures and databases. During the development of this artifact, we were tasked to create one of three applications for Android smartphones. The application that was chosen for this artifact was the Inventory App. This app's main purpose was to allow users to insert items of their choosing into a database and display these items on the main screen. The user can also edit the information for these items or delete it at any time.

This artifact fits the first category, software engineering/design, because during the development of this app, software engineering best practices were implemented as well as design elements such as designing the UI and implementing on the UI for the entirety of the development process. An agile development methodology was used to create this app through several sprints. This artifact also fits the second and third categories, algorithms and data structures and databases because a very large portion of this artifact is made up of these

categories. Throughout the development process, the programs data structures and algorithms were constantly iterated upon in order to effectively organize and manage its data as well as reach the desired functionality.

This artifact showcases my skills and abilities in creating effective data structures and algorithms, using coding best practices, and creating databases. The design and code created for this application went through many iterations. These iterations were completed through a number of sprints. Once the program was tested at the end of the sprint, necessary changes were documented and implemented in the next sprint.

Software Reverse Engineering

The first artifact in the ePortfolio is the software reverse engineering project. This artifact is made up of several programs and two projects that revolve around the practice and theory of reverse software engineering. This project allowed me to learn how reverse engineering works, how to read assembly and hex code and then how to recreate those programs.

This artifact fits into all three categories, Software engineering/design, algorithms and data structures and databases because the practice of software reverse engineering involves each of these. When a developer needs to go into a legacy system and discover what needs to be updated or what bug has occurred in the system, they need to read and understand the program completely. They do this by reading and analyzing the assembly and hex code. After doing this, they can then recreate the program and redesign it. This entire process deals with software engineering principles and practices, algorithms, data structures and databases. Thus, we have a correlation between this artifact and all three categories.

The skills that are illustrated through this include understanding how to read assembly code and how to re-create a program from assembly code. Additionally skills include developing a security mindset that anticipates adversarial exploits in software, demonstrate an ability to use well-founded and innovative techniques to implement computer solutions, and design and evaluate computing solutions that solve a given problem.

Data Mining

The second artifact in the ePortfolio is the data analysis report. The first two artifacts contain projects that using coding and therefore, I wanted to include an artifact that focused on other aspects of computer science. This artifact is made up of a database warehouse and one large 40-page document that analyzes a fictitious company and its data. The documentation and analysis's goal was to discover why the company's sales have started to decline. The company provided several different pieces of data such as a data warehouse, online sales, customer data, etc. This artifact fits into the Software engineering/design category.

This artifact fits into the software engineering/design category because it focused primarily on the design and analysis of the program itself. Using tools to create algorithms to analyze the data can help developers create apps, websites or solutions for a client to help the business grow.

The skills that are illustrated through this artifact include understanding how to analyze and read data from a client such as a database warehouse, customer purchase data, etc. and use that analysis to help create patterns and correlations in the data. Once specific patterns and correlations have been found, additional skills in creating visual representations of the data have been learned. Additionally other skills from this artifact include employing strategies for

building collaborative environments, designing, developing and delivering professional-quality

oral, written and visual communications and demonstrating an ability to use well-founded and

innovative techniques, skills and tools for implementing computer science solutions.

ePortfolio Artifacts

The artifacts that are included in this ePortfolio were thought about and carefully

selected. The artifacts were chosen due to their connections in the three key computer science

concepts. I included two coding projects because I am most interested in the programming side

to computer science. These two artifacts help showcase my skills in databases, data structures

and algorithms. I also included an artifact that deals with data analysis to help showcase my

skills in other fields other than programming.

Each artifact that has been included are strong examples that showcase my skills in a

wide range of fields of computer science. I am proud of each artifact that has been included in

this ePortfolio. I will continue to add additional projects to this ePortfolio and also update this

document as I add more projects.