**8-2 Journal: Portfolio Reflection**

Austin Siegel

Southern New Hampshire University

CS-405 Secure Coding

Professor Trevor Hodde, M.S., Computer Science

8.27.2024

Secure coding standards and best practices are the new cornerstone of software development. Over the years, technology and programming capacity have increased, and with it the speed and dexterity of cyber-attacks. This course has shown me how important secure coding is and why it must be integrated from the base level of the software development lifecycle. In the long term, doing this minimizes the cost of security integration and makes it easier for developers to understand secure coding practices. Using secure coding and defense-in-depth may be a greater initial cost, but it will not cost nearly as much as a sever data leak that could be prevented by using these principles. This can be considered an opportunity cost and can be evidenced by the hundreds of successful cyber attacks across the world that have cost companies and their customers a lot of money and sensitive data.

It is for these reasons that no-trust policies are a valuable resource for protecting users and their data. This means that, by default, no one person or source is given express access to data. All pathways must be authenticated thoroughly and frequently. Authentication and authorization are just a few ways to ensure secure coding standards. Using tools like static code analysis, unit testing, and external vulnerability checks are also necessary to create a truly effective defense. There are many warnings that the compile of your IDE can miss, so having external tools to check for different errors in different scopes will help to prioritize which errors should be handled first or possess the highest risk.

Utilizing these tools alongside basic coding standards and best practices is an effective way to minimize inherent risk. Saving your code often and running it after each block of completed code will help the integration of defense-in-depth and secure coding standards.