Secure Software Development – Reflection

https://tbrays.github.io/e-Portfolio/

This reflection looks at the module as a whole and the evaluation of the coding project. It follows the What, So What, What Now format suggested by Rolfe et al.’s (2001).

What?

I enjoyed the opportunity to learn new skills during this module especially in Python and using linters, this experience has broadened my understanding of software development. Despite this I found the module structure confusing and the explanations of tasks unclear. The flow of the module seems random, and the chosen unit topics appeared poorly selected with no clear connection to the module's learning aims.

During the development project I worked in a group of four. The project was not explained well and the advice we received was often contradictory. We collaborated during the design phase while the development and testing were done individually. My team met weekly and worked well together, assigning new tasks based on individual strengths. Our team had two members with a background in development and two with a background in education. One big debate was whether to use a command-line interface or a web interface which was difficult because two team members had little experience with large projects. We decided to develop a web interface thinking it would look better but I later realised the original brief specified a command-line interface which we overlooked. In our final meeting we checked the reference list, spending a lot of time to make sure they were accurate. The Harvard referencing system seems out of date and convoluted.

We were told to use references no more than five years old which I thought was reasonable. However, as the module progressed many references we used were older including the core texts for the course. A lot of the subject matter has been established for many years leading to older documentation. Some references in the brief were hard to find and one could not even be provided by the tutor.

I enjoyed learning Python as it differs from other languages I know, the way Python enforces layout and how the linter checks PEP-8 style were useful. Tollervey, N (2015) suggest that it is a good first language due to its simplicity and readability, however I found Python's loose variable typing made it harder to understand data types compared to strongly typed languages. Since the project required Python I faced the challenge of learning a new language. Starting on this module felt like a mistake because it seemed to combine several units however, this presented an interesting challenge and an opportunity to learn a new language. A team member mentioned the CS50 Introduction to Programming with Python course, the videos for this were online and I made a point to watch all of them, they help to fill a lot of gaps.

I used the Flask framework to develop the website with an HTML/CSS interface. During development I spent a lot of time learning Python which limited my ability to finish the system. As the deadline approached I had to submit the application without completing all requirements, it worked but some features were missing. The project could have benefited from refactoring, as I needed to rebuild the database from scratch because it had become unnormalised and may have contained redundant or incorrect data.

So What?

Reflecting on my experience I see how unclear guidance affected our project. The contradictory advice and lack of clear direction made it hard for my team to succeed. Choosing to develop a web interface seemed right at first but it led to a misalignment with project requirements. This taught me the importance of clear communication and understanding project briefs and methods I can use to better overcome this in the future.

I learned that while picking up a new programming language can be beneficial it can also distract from completing the project. My experience with Python highlighted the need for a structured approach and using tools like linters to maintain code quality. The challenges I faced with the database emphasised the need for a solid understanding of data management allowing for the effective design of the database.

I believe that the final application met the original specification:

- It was structured for admin, teacher and student users.
- It allowed adding new users and updating or deleting current ones.
- It provided the ability to view timetables for both students and teachers.
- The admin could view delete or add new timetable entries.
- When creating a user, the system enforced a strong password and provided clear error messages if a password did not meet standards.
- The login system tracked failed logins and locked an account after three failed login attempts.

Throughout this module we were encouraged to programme defensively adding error handling and writing clean code (Anaya, 2018), assuming users might make mistakes due to this the requirement to turn off the security was challenging because it was integrated into the code.

The tutor suggested that there should be evidence of attacks on the system such as emulating and testing a brute-force attack. However, testing for DoS attacks and API injection attacks proved difficult. The database entries were managed with the Flask framework which provides ORM through the SQLAlchemy library.

What Now?

I am looking forward to my next module Launching into Computer Science, as I believe this module should have been the first one taken. I expect it will help me learn the structure of the modules to follow and clarify what is expected of me.

In the future I plan to use Python and the Flask framework more while also exploring Django extra tools such as its added security features. In my next module I will read all unit briefs carefully at the start and plan out deliverables so I can ask questions early. This reflection has motivated me to take a more proactive approach in my learning and teamwork in future projects.

References

Anaya, M. (2018) *Clean Code in Python.* 1st ed. Birmingham: Packt Publishing. Available from: https://learning.oreilly.com/library/view/clean-code-in/9781788835831/9d4711f1-586a-45e1-8040-dd472f7c41d3.xhtml [Accessed 19 October 2024].

Rolfe, G. (2011) *A reflective model for nursing and health care professionals.* Available from: https://my.cumbria.ac.uk/media/MyCumbria/Documents/ReflectiveModelRolfe.pdf [Accessed 19 October 2024].

Tollervey, N. (2015) *Python in Education.* 1st ed. Sebastopol: O'Reilly Media, Inc. Available from: https://learning.oreilly.com/library/view/python-in-education/9781492037880/preface01.html [Accessed 19 October 2024].