Module Reflections

The secure software development module has been thought provoking and provided challenges, at times. Overall, the aims of the module have been met, as detailed below:

Identify and manage security risks as part of a software development project.

The module has promoted an understanding of the main threats and security risks associated with secure software development. A touchstone is the Open Web Application Security Project (OWASP) which raises awareness of the types of risks and provides associated strategies to mitigate these risks.

In addition to a basic understanding of risks and mitigation, this knowledge has been considered alongside the approach to the software development lifecycle (SDLC). For example, the 'modern' agile approach has been gaining popularity over traditional waterfall approaches. While the Agile Manifesto has been in place for over 20 years, it is still referred to as a new development thoughout the learning in this module as it has gained more prominence as an approach in industry today. An important reflection would be to acknowledge the differences between the quicker approach to software development when comparing agile with waterfall approaches. An agile approach is more flexible and places more emphasis on working solutions rather than comprehensive documentation. This methodology brings associated risks which can be acknowledged and mitigated by the adoption of secure scrum methodologies.

Critically analyse development problems and determine appropriate methodologies, tools and techniques (including program design and development) to solve them.

Unified Modelling Language (UML) models can be used effectively to communicate effectively. These documents can quickly and easily convey working structures and sub-systems as well as identify and illustrate use and mis-use cases to efficiently document and design a system. The UML documents allow a consistent approach for a team of developers, sharing the important relationships within the systema and defining objects, functions and relationships. These can be enhanced with ontologies and schematic webs.

Systematically develop and implement the skills required to be effective member of a development team in a virtual professional environment, adopting real-life perspectives on team roles and organisation.

The team task provided a good opportunity to replicate a real-world scenario. This included, ensuring all team members had an opportunity to contribute and lead on certain aspects of the task. This also presented the real-world challenges on completing deadlines when, perhaps there was not always an equality of contribution or commitment. The global nature of the team was welcome and reflective of working in a large multi-national organisation. This meant that the team had to compromise on scheduling meetings across time zones to work towards the best outcomes. It was also important to understand the varied technical background of each to of the team members, and to assign tasks strategically to ensure quality. The team were able to successfully navigate these barriers and ensure that the design brief was completed to a high standard.

Design and develop/adapt computer programs and to produce a solution that meets the design brief and critically evaluate solutions that are produced.

e-Portfolio is available from: E Portfolio (karljackson10.github.io)

The programming tasks have provided an opportunity to put the theory into practice and acquire new skills and knowledge. In the previous modules the Codio tasks were well-matched to the coding element providing a scaffold for learning. As this was the third module, there was a clear expectation that the resources would only provide basic techniques. In order to consolidate this knowledge and put it into practice, extensive wider reading and research was necessary. For example, the module introduces and progresses the knowledge and procedural skills regarding the use of Flask in particular. While it was much slower to gain and synthesise the working knowledge and techniques necessary to complete the coding, it was satisfied to have learnt all of the Flask notation completely independently.

Similarly, the module promoted an insight into the development of API and the resources promoted further questioning and research to begin in order to complete the coding task. This would be an area to develop and explore further in the future.

Alongside the significant development and application of coding skills, this unit provided the opportunity to consider the importance of annotation for reusability and debugging. This was combined alongside the use of linters and understanding the importance of following PEP8 style guides. Readable code is more secure code. The project also presented opportunities to consider the implementation of other secure coding techniques such as permission management and the use of secure passwords alongside regex patterns. The secure coding techniques have been brought together in the project in a meaningful and purposeful way. An area to develop would be the use of file-handling and the ability to launch the application from a developer environment to a live web-site.

Alongside the development of coding skills, the project allowed an opportunity to bring prior knowledge to fruition. For example, in the introduction to computing module there was a very basic reference to command line interfaces, SQL queries and HTML coding. In the object-oriented programming module, there was a more structured approach to coding, notation and the use of objects themselves. This coding project brought these units together and it was very rewarding to be able to link the separate domains to study in one project. These skills could be sharpened and developed further, for example, the code could be more efficient and elegant reducing cyclomatic complexity and the interface, itself could be more engaging with the development more sophisticated HTML or JavaScript. Finally, the data-base used was very basic but has prompted a thirst for more knowledge in the data-science sphere, perhaps exploring MqSQL databases further with the use of PHP scripts to enhance security.

The e-Portfolio has been produced to accompany this module and can be found at: https://karljackson10.github.io/Secure-Software-Development/index.html