Over the past three weeks I have found that the subjects covered in these modules to be challenging and

intellectually rewarding. After some reflection, I have come to realize the following:

I have learned techniques that systems and application programmers can use to write new code securely,

as well as to find and mitigate vulnerabilities in existing code. In addition to covering these threats,

we discussed various tools and techniques that can be used to write secure software.

The discussion and feedback on assignment help me reinforce my understanding and usage of UML

diagrams as a tool for software development. The history of programming languages where both interesting

and informative.

I learned that Building secure software begins with defining what is the security requirements that one

needs to take into account. Just as business requirements help us shape the product, security requirements

help us take into account security from the beginning. I found that the OWASP list of the top 10 critical

security risks to web applications is a great starting point on some of the challenges I may face as a

software developer. I especially learn the importance of SQL injection detection and prevention. I

learned that web applications should never run with administrative privilege at the server level or at the

database level. I founded that running applications with the low-privileges are an affected way of minimizing

trouble and still maintain the function of the application, also setting databases permissions to only the essential resources offer some level of protection.

References:

OWASP (2018) OWASP Top 10 Proactive Controls 2018. Available at: https://owasp.org/www-project-proactive-controls/ [Accessed 25 May 2021]

 PTSecurity.(2019)How to prevent SQL injection attacks. Available from: https://www.ptsecurity.com/ww-en/analytics/knowledge-base/how-to-prevent-sql-injection-attacks/ [Accessed 25 May 2021].