REVISION NO.:	00
REVISION DATE:	May 10, 2016

DAILY JOURNAL

IMPORTANT INFORMATION

- INCLUDE TASK ASSIGNMENTS OR MOVEMENTS, REFLECTION ON THE DAY'S NEW LEARNING, ACCOMPLISHMENT, CHALLENGES FACED AND HOW YOU RESPONDED, OBSERVATIONS AND RECOMMENDATIONS ON THE IMPROVEMENT OF SYSTEMS / OPERATION / MANAGEMENT, ETC.
- SCANNED COPIES OF THIS FORM SHALL BE SUBMITTED ON A WEEKLY BASIS THROUGH APPROVED LMS.
- HARD COPIES OF THIS FORM SHOULD BE COMPILED AS PART OF THE STUDENT'S PORTFOLIO.

DATE	May 30 - June 3 2022	AREA ASSIGNMENT	ATA
TASK	Improve Your Application Security Testing Skills	SHIFT/TIME	8:00 AM - 5:00 PM

May 30 - May 31 The first part of this module is basically a review of risk assessment, vulnerability assessment and compliance assessment. The next part is about web security where we check if a user has a permission to access resources or if they have a role or belong to a group that can access these resources. I was able to learn more about penetration testing and the different tools and techniques that I can use to make sure that the application that I am building is secured from attacks. For javascript codes, we can use Retire.js to determine which imported library that are in use has known vulnerabilities or if we are developing using node.js we can use the built in function npm audit to look and search for vulnerabilities in the imported libraries in our project. The next part after this introduced me to tools and applications that I can use to perform penetration testing on my website such as Fiddler 2 to analyze and inspect incoming and outgoing HTTP traffic, Samurai Web Testing Framework, Burp Suite, and OWASP Zed Attack Proxy (ZAP) which can easily scan my web app for known vulnerabilities like XSS, SQL injection, URL manipulation and this will also show detailed information about which area of our application is vulnerable to certain attacks. I also learned about efficient penetration testing by splitting our test in multiple builds or running tests on parallel build to expedite the process since some tests take long time to complete. The last part of this module is about different tools that we can use to perform automated security testing like setting up testing environment using docker, and using Gauntlt to automate Cross Site Scripting tests, SQL injection tests, etc. Overall, this module taught me how to properly write secure applications by implementing user authentication, authorization, performing security tests and performing automated test and I will definitely use this because prior to this, the only test that I do when I'm making a website is unit testing, performance testing and basic security testing for authentication and authorization, now I will be able to properly do various security tests and even automate it using the tools that I learned in this module.

June 1 - June 3 Now that I finished all my learning module, I will be using the remaining of my time to complete the requirements for the assessments, final report and my portfolio.

TRAINEE'S SIGNATURE