Reflection of SSD

In the first collaborative discussion, I have learnt the UML flow chart can help everyone to understand the potential cyber security threat, especially the management level, who are not familiar with the cyber security threat so deeply.

I found that it’s not easy to create a UML diagram, which makes everyone to understand. I need to understand the following, so I can create a clear, easy to understand UML:

1. Understand the flow: every component of the project, people who are involved, and the responsibility of each role.
2. Need to structure my charts clearly. For example, if I need to build a UML chart for a team’s workflow, and I need to reflect one of the team members need to pass things to another, I need to structure and display the action clearly, so that everyone can trace the flowing of the object. Also, I need to make is as simple as possible.
3. The UML chart should be able to display and identify issues. A UML chart, especially when the UML is used to identify the cyber security threat: the objects, flowing and the result (Setapp,2021).

In the Design document, I was responsible to the technical challenges. I have redone this section for two times.

In the first draft, I have listed different security threat of the project:

1. Threats from ransomware or crypto viruses
2. Sharing information without any caution
3. Single point of failure: Different areas of the law enforcement organisation will also contain different threats:   
    i. Physical security

ii. Resource security

iii. Device, network, and appliance security

But my team members said they were not “technical challenges”. Technical challenges such the difficulties of coding, how to overcome the potential security threat, in technical ways, such as the use of modules and coding.

After re-done, I have raised certain technical issue, including:

1. Data mapping between the app and database
2. Create a memorable URL
3. Create a user-friendly UI

I have learnt these technical challenges which are related to the interface, the ease of use, the communication between client and server, the connection and alignment between the database and the front end, a secured and easy to do administrative and hiding unnecessary information. But not only the potential cyber security threat (Django, 2021).

I also have learnt to simplify the security challenges, especially there are limitations on number of wordings. I need to use the simplest sentences and terms to describe the challenges and what are the solutions that Django has provided.

About the development team project, I was responsible to the fire sharing. As I was not good at coding, I need to research how to deploy the service with Python.   
  
The first thing I have learnt from Python was the modules. There are different modules for me to build the application, so that it can be functional easily. I also have learnt to setup a HTTP web server through the modules in Python: **HTTPServer** and **socketserver. These modules able to listen the HTTP protocol and write files to the network servers.**

**I also have learnt I can use python and create the front end easily, by using the web browser module. It able to display the web-based document; and PyPNG can read and write image files which is purely using Python (GeeksforGeeks, 2021).**

**But after have discussion with the team, they need to use the same application to run the whole thing, which was Django. I have learnt I need to co-operate with the whole team, to make the work done.**

**I have re-submitted the code again, which could be run on Django. As my work was file sharing, I have learnt how to use the filestorage function of Django; I also have discovered the base.html can create the background easily. Also, I have learnt we need to sacrifice the user-friendly functionality: admins need to access the files through the windows explorer, rather than a web page, to reduce any possible security threat.**

**In the project, I also have learnt we need to create an architecture and from for detecting security vulnerabilities, this need to be done before the application can be published. One of the tools is OWASP. According to IEEE (2017), We did test against the major attacks such as SQL injection, URL injection, Cross Site Scripting (XSS), and the HTTP Server (DDoS). Also, I have learnt how to protect our system from these attacks. For example, binding variables to control the SQL injection attack, using tags to the code, for example, “<” and “>” to prevent the XSS attacks. We also cannot trust the user input as they are always the source of the potential security threats.**

**Also, I have learnt a new CMS system, Wagtail, which can be integrated in Django project. But it’s quite complicated to modify the urls.py and setting.py. They’re difficult to integrate, including some modules are missing, invalid syntax, and confusions of the grammar between the old and new versions of Django. I need to visit different forums, the Python and Django web site, searching for documentations.**

**Besides Python scripts, I also have learnt some HTML in the project, to redirect the file upload web page from the ticket request form. A simple link with encryption can reduce the data loss.**

**References:**

**1. SETAPP (2021) How to create a UML diagram on Mac, Available at: https://setapp.com/how-to/uml-diagram-guide (Accessed: 31st August 2021).**

**2. Django (2021) Django Documentation. Available at: https://docs.djangoproject.com/en/3.2/ [Accessed 27 August 2021].**

**3. S. Kumar, R. Mahajan, N. Kumar and S. K. Khatri (2017), "A study on web application security and detecting security vulnerabilities," 2017 6th International Conference on Reliability, Infocom Technologies and Optimization (Trends and Future Directions) (ICRITO), pp. 451-455, doi: 10.1109/ICRITO.2017.8342469.**