

Team 4 Project Evaluation

Unit 1-11: Teamwork, Project Design and Delivery

Module 3 dealt with introducing students to security concepts related to information systems. This module introduced the ability for teamwork and collaboration to (1) deliver a system design report and (2) implement a Python-based system that adhered to the requirements identified. The first week revolved around allocation of teams, and I was tremendously pleased to be paired with the team members assigned, since they are two members on the course that exhibit a sense of engagement and interaction.

Team Contract

During the initial phase, we developed a Team Contract and revised the contents to ensure we captured the essence of *expectations* and *roles*. It was a collaborative effort, very amicable and open to ideas and considerations from all members. As part of the contract, we considered how each member would play specific roles, and even discussed who would enjoy fulfilling which roles. Based on the tutor's feedback, one point was to consider revolving roles to facilitate knowledge transfer cited as the major goal of role rotation, Santos et al. (2016).

Requirements Analysis

We chose to look at the International Space Station (ISS) project and read through the provided brief (NASA, 2007). We followed this process to identify requirements for our design document:

1. Read the brief to understand the domain.
2. Understand the threats to their systems.
3. Identify all the actors involved in their systems.

4. Pick an area of concern which we could focus on resolving. We settled on their use of Safety Database entries to help provide information to resolve future incidents.
5. Based on the chosen *area of concern*, we then formulated several requirements that we felt were necessary to resolve the ISS issue.
6. Discuss the requirements and area of concern as a team to settle on agreement of the design focus.

Leading on from the requirement identified, the architect role then set about developing multiple UML diagrams—class and activity—which we then discussed among the team. The architect incorporated the feedback and presented to the team for any changes.

Team Contributions

Please refer to the **Team Project Collaboration** artefact on the e-portfolio.

Project Implementation

Each member contributed to the successful delivery of Python code for the team project, and we ensured that each requirement identified in the design document was met and delivered. All necessary packages and database tools were installed in the Codio environment.

Differences between Design and Implementation

While we were careful to ensure that all requirements were implemented, it was not possible, due to time constraints, to deliver on all.

What was left out?

- REQ05 Crew Management was left out.

- GDPR data requirements were not implemented due to time constraints. However, the team is aware of GDPR data protection and considered the need to encrypt sensitive data in motion and at rest.
- We did not consider the “Contractor” or “Employee” roles as initially designed on the UML class diagrams; although, the system *does contain permissions* relevant to each user. However, the function of **Create User** merely creates a standard user.
- We did not implement passwords in a separate database as originally envisioned, rather we hashed the passwords using the Python library bcrypt to securely store a user’s password together with their user information.
- We did not leverage Git source control, opting instead for the Codio environment. Each team member has a separate copy of the project implementation.

What was added?

- The introduction of session tokens for the Web API portion was introduced because it provided a layer of security between a user using the API and the monolithic backend, itself protected by authentication and authorisation.
- The introduction of “Screen” objects to cleanly separate each screen of functionality as the user navigates between each option. This was a wonderful introduction because, despite the implementation following monolithic approach, we still maintained clean separation between “functions”.

Summary

Overall, I am very pleased with the delivery of the project and feel that, given sufficient time and budget, the team would excel beyond the requirements for the NASA project. What was delivered is very close to the original design. The team worked tremendously well together, and I consider the project successful.

References

NASA, (2017). Final Report of the International Space Station Independent Safety Task Force. Available from

https://www.nasa.gov/pdf/170368main_IIST_%20Final%20Report.pdf.

Santos, R.E., da Silva, F.Q. & de Magalhães, C.V. (2016). Benefits and limitations of job rotation in software organizations: a systematic literature review. *Proceedings of the 20th international conference on evaluation and assessment in software engineering*: 1-12.