Secure Software Development

Team Meetings for Team 4

Notes:

Date	10 Sep 2021	**Meeting ID**	6
Attendees	Andrey Smirnov; Taylor Edgell; Michael Justus;		

Agenda

- Review Technical Report

Minutes of Meeting

Review Technical Report

- The team will work on the report and have a large portion of the content ready next week. We will schedule a follow-up meeting for Tuesday, 14 Sept. 2021 to provide any further content.
- In terms of content, the “Overview” section is fine, but we may have more assumptions to fill out. Though, currently, our Assumptions do not consider tokens, authorisation, or fine-grained access control.
- The current template will add a separate header for security considerations as the team considered security to warrant a section on its own, without the need to mix it amongst the assumptions. This section will add reference to OWASP and we will consider relevant points as well as how we shall address the security.
  - “we’re using tokens because…”
  - “instead of tokens, we using XXX to implement YYY”
  - This is because it is analysed and we know that it is there
  - Cathryn still feels that what we’re doing is web oriented and not mobile or cloud so OWASP top ten fits nicely
  - Reference to OWASP will also be added into the REFERENCES section

Class Diagram

- Add description about Password DB will be added.
- And new Authentication class.
- Remove note from Authorisation class and add it to the new section about Security Considerations.
- The class diagram is not giving all the answers and is open to interpretation for how the developers can build the system.
- Authorisation is targeted at the access level of the user and is compared to the access level of the Safety Entry. For our description to Cathryn, we will show a simple application of the *ideai* where Authorisation is more along the lines of *actions*.
- Even though some members are marked as PRIVATE, accessibility modifiers will be ignored in Python code.

References

References to OAuth and OWASP, MongoDB, xxx Fowler (UML) as a cloud-based solution must be fit in somewhere.

Report Sections

Section 5.4 split into two diagrams. We will move the diagram with swim lanes into the Appendix and keep the diagram that is of primary importance.

Technical Overview

Technical challenges. We are not fully sure of what to achieve with “challenges”, however some pointers were raised, such as:

- Python **accessibility.**
- Notes about **encryption** as a software solution.
- CIA implementation is a challenge.
- We’re using default parts of a library for **encryption** method
    - We mitigate by using only the basics of the particular library. Like adopting the library with the most adoption by the community. We have not say over the suitability
- **Source control** and how to develop code because we have only worked in Codio so far. What tools are there to support the development of code?
    - Input required from Cathryn.
- Coding **standards** for the team members.

Patterns and Methodology

If not using MVC or whatnot, it leaves us with software patterns like singleton. The Authorisation and Authentication classes are singleton. To demonstrate polymorphism, we can show it via the User/Admin. We can introduce functionality to show polymorphism.
We can also make reference to use of th eRepository pattern.