Secure Software Development

Team Meetings for Team 4

Notes:

Date	3 Sep 2021	**Meeting ID**	5
Attendees	Andrey Smirnov; Taylor Edgell; Michael Justus;		

Agenda

- Review UML diagrams
  - Activity Diagram - Application Access V2
  - Activity Diagram - Application Access Swim lanes V1
  - Activity Diagram - Safety DB V2
  - Use Case V1
  - Class Diagram V3
- Discuss preparation for Technical Report
  - Layout, Paragraphs, Content
  - Integration of UML diagrams

Minutes of Meeting

Review UML Diagrams

The team walked through a review of the UML solution diagrams.

- Review Topic: **Tokens**.

  - The user is given a token by an admin.
  - The solution will have a list of viable tokens. We shall make an assumption that a list of valid tokens is preconfigured.

- When a token is “used” by a user, it will then be marked as inactive and cannot be reused.
- Tokens are created when we create a new user.

- Review Topic: **Authentication**.
  - The solution shall have a separate authentication API.
  - We use single-factor authentication.
  - Should 2FA be leveraged? The team agreed that it may not be wise to invest in it for a monolithic application.
  - OAuth 2.0 (and specifically, OpenID Connect) was referenced in consideration of populating claims in the user token.
  - Taking the example of OAuth2, the team feels we can emulate similar functionality for authorization.

- Review Topic: **Authorisation.**

  - Having an access level is a simple thing to implement. We will write a few sentences in the Technical Report to show the team is aware of more advanced authorisation concepts.
  - The Technical Report shall contain authorisation assumptions based on time limitations and in state that a real syste would implement authorisation in such-and-such manner.
  - The team agreed to separate authentication and authorisation.

- Review Topic: **Design.**

  - For Part 2 we shall ensure separation of concerns.
  - Possible introduction of new authorisation service.
  - Solution’s UML diagrams reflect the *ideal* solution, not real-world solution.
  - Addressed scaling concerns through use of multiple DB instances.
  - We discussed the class diagram v3.
  - We discussed the flow chart with swim lanes.
  - We did not discus the use cases.

- Review Topic: **Databases.**
  - The team considered use of two DBs: one to store passwords and another to store document content. The reason is because databases are structured data. Usually SQL, but can be Postgres, structured data, easy not complex. For Safety Entries feels like we are dealing with unstructured data.
  - Safety DB encapsulate Safety Entry and Users. From a safety point of view, if the Users DB was compromised, the solution’s access controls could be bypassed since the attacker could access other parts of the system.
  - Multiple DBs also complements the microservice approach to design and security.
  - User class doesn’t store the password but may have a link to the password DB
  - SQL Lite implementation is really light.
  - Using MongoDB adds complexity.

    - Mongo for SafetyDB. Separate class for Password DB. Tokens stored in dictionaries? Simulating a separate DB. In this example,

- Review Topic: **Changes.**
    - (Application Access V2) Activity “Run antibvirus software”. Make it a different colour.
    - Add new password DB.
    - Authorisation to move out of Authentication.
    - Add new InputValidator interface to allow implementation to have polymorphic behaviour for validation.

Technical Report Preparation

Of the requirements, at least half are a technical lead breakdown of the solution. For example, libraries, patterns. That’s something the Technical Lead can list, though not all, because we the diagrams will provide input too.

For the Technical Report, we will populate:

- Challenges to be addressed [Technical LEad]
- List of patterns [Technical Lead]
    - These are referring to system architecture such as use of “MVC”.
- List of tools and libraries [Technical Lead]
- List of assumptions [Collaborative]
    - We shall populate these into a table.
    - We listed original assumptions in our first meeting.
- List of system requirements [Collaborative]
    - Gathered from the client.

Other Points of Discussion

1. Who is going to submit the document?
2. From Wednesday in Week 5, we shall go into review mode.
3. The team feels we are able to deliver the Technical Report on time.
4. Caution was expressed to not overthink the Technical Report, as it is better to focus on implementation.
5. Use of ISO terms is encouraged in the report content.
6. If the report has academic references, it is important they are *relevant* given that the lecturers are familiar with many references.
7. There is no real limit on number of pages in the report (despite the brief stating a limit of 2 pages).
8. We will work collaboratively on the Technical Report using OneDrive integration through the University’s Microsoft Office account. The Architect will generate the first template, and QA role will take the lead on report content.