Snapshot Week 10 of Group AttackFlow1

Building a dataset of real-world cyber-attacks with Attack Flow

Se Jin Yoon: a1706219
Ting-Wei Chin: a1782423
Faisal Hawsawi: a1822781
Lina Nehme: a1802697
Ran Qi: a1675122
Joseph Toubia: a1753547
Zemin Wong: a1780385
Jixing Ye: a1798631
Yu Zheng: a1739446

October 16, 2023

Product Backlog

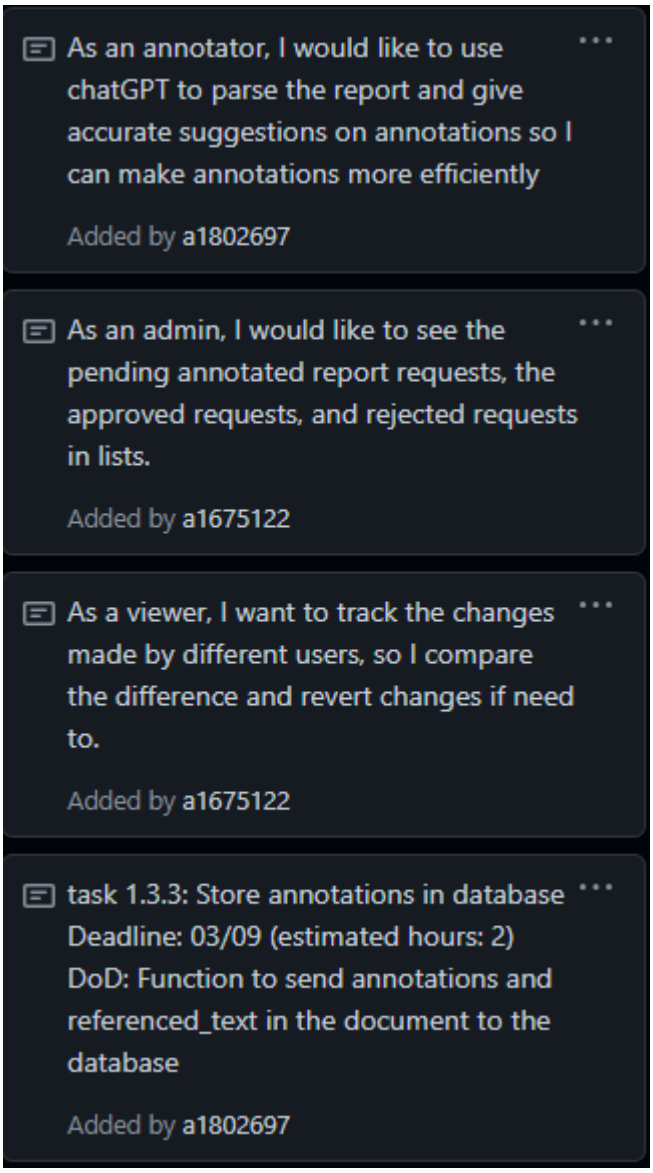

Figure 1: Product Backlog (Sprint 5) 1 of 5

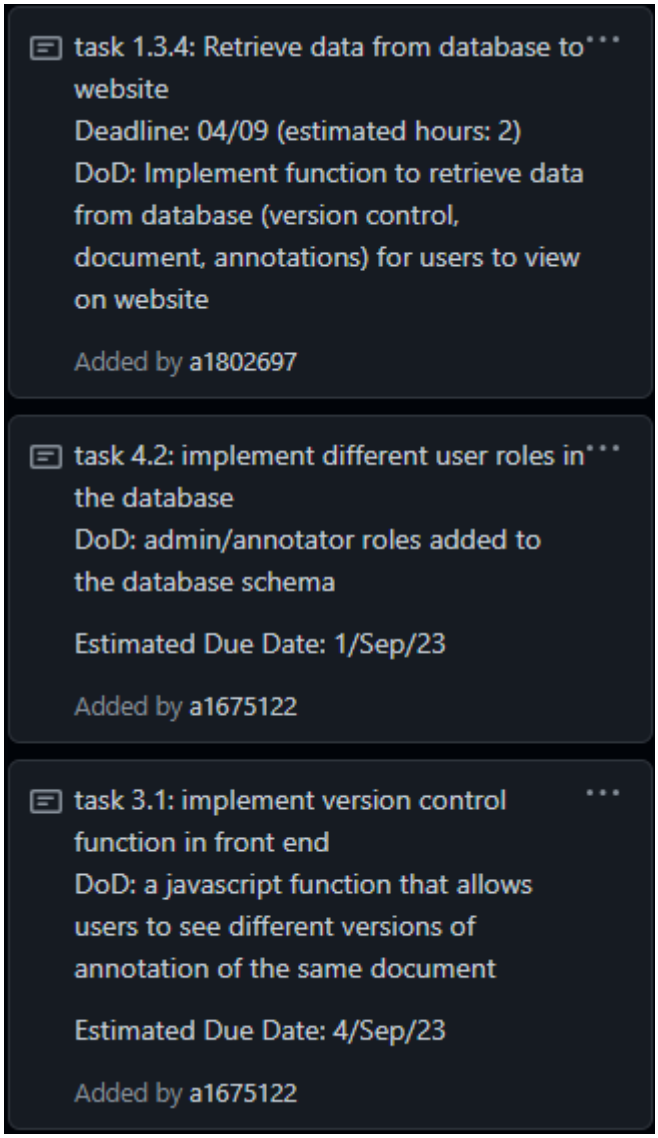

Figure 2: Product Backlog (Sprint 5) 2 of 5

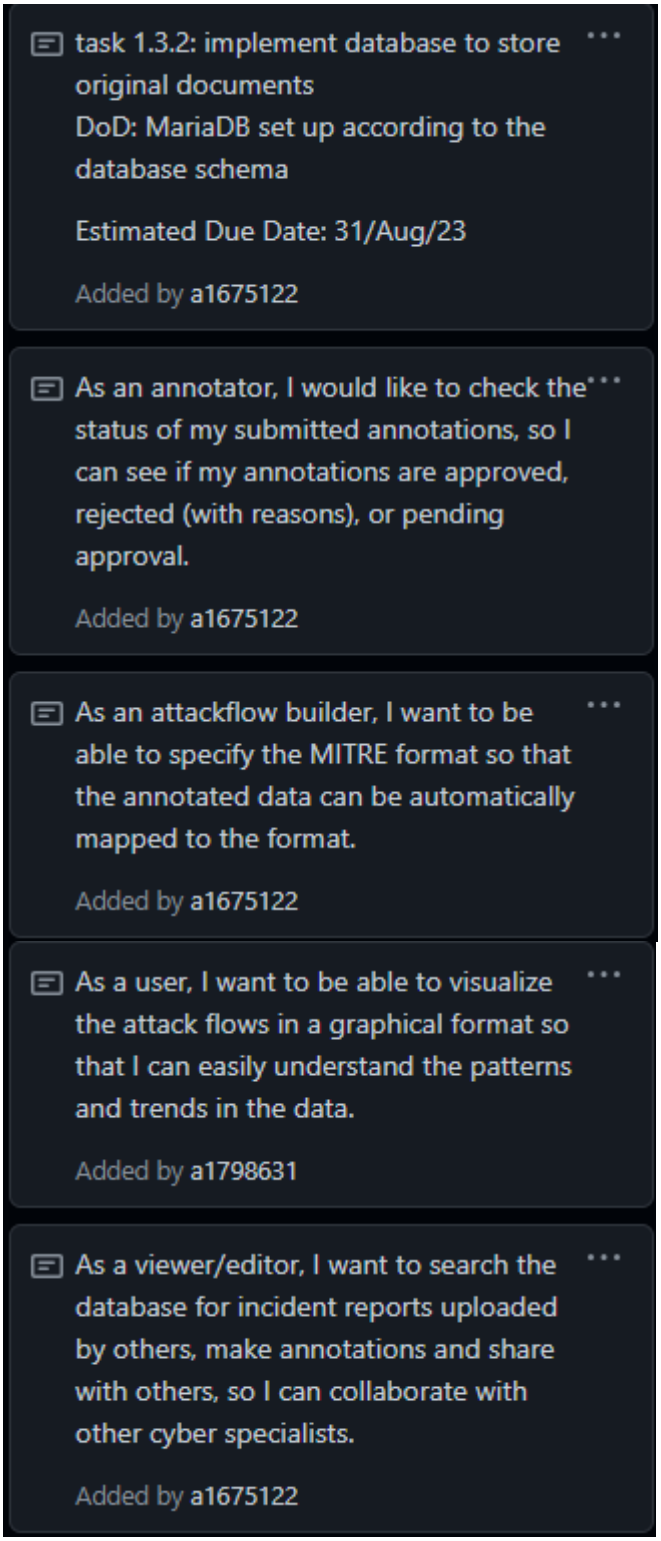

Figure 3: Product Backlog (Sprint 5) 3 of 5

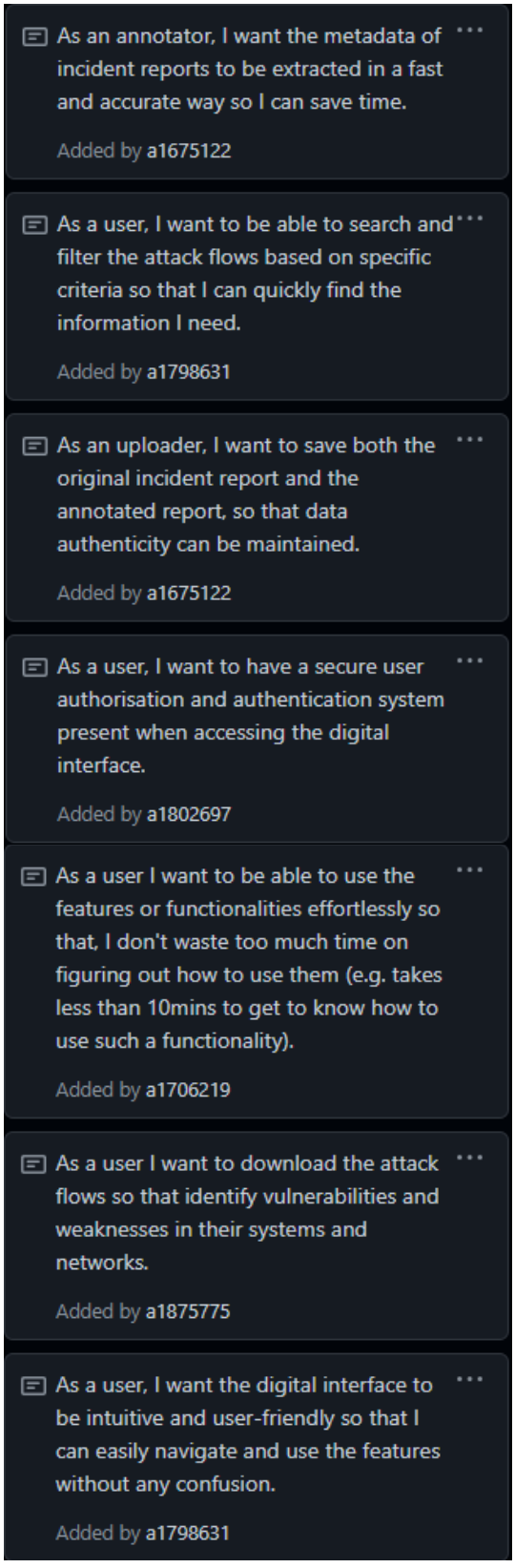

Figure 4: Product Backlog (Sprint 5) 4 of 5

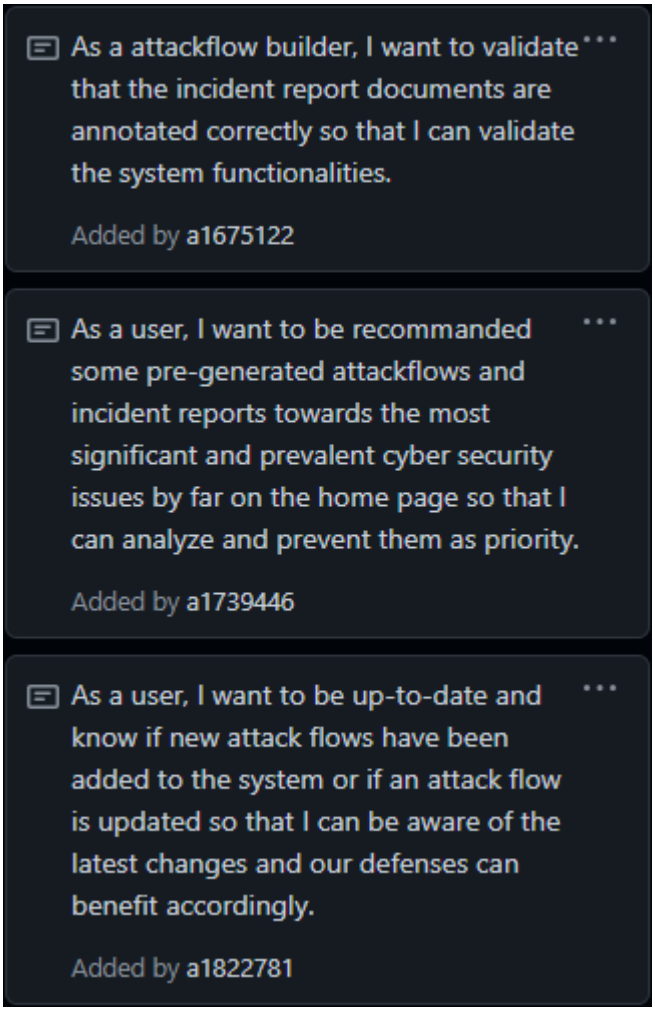

Figure 5: Product Backlog (Sprint 5) 5 of 5

Task Board

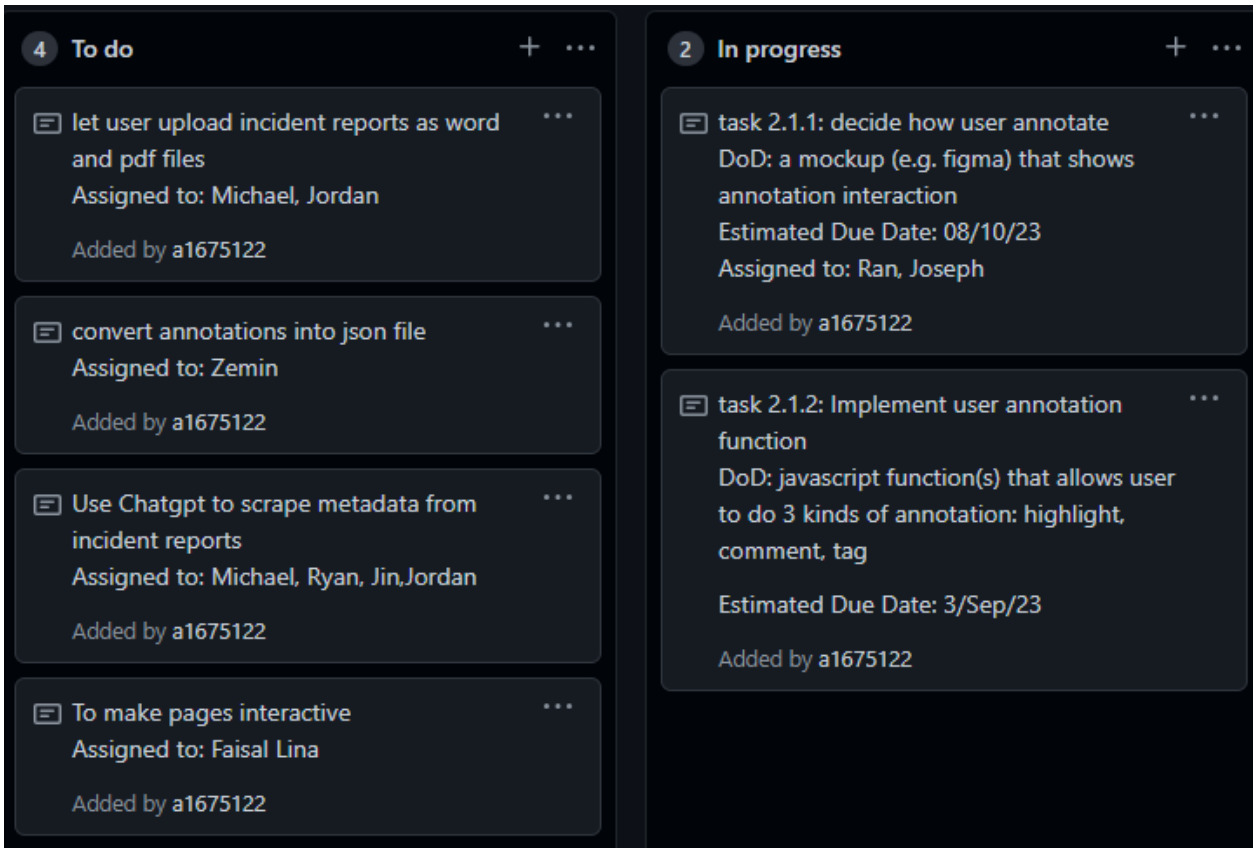

Figure 6: Task board (Sprint 5)

Sprint Backlog

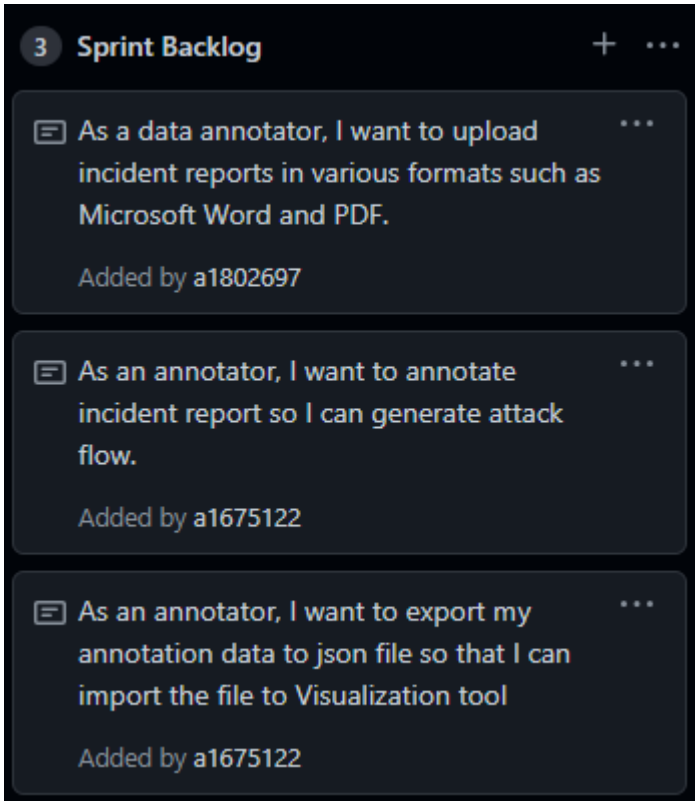

Figure 7: Sprint Backlog (Sprint 5)

User Stories

Nearing the end of the project and some user stories have become imperative to complete. One such task is to improve the annotation process with ChatGPT, which was ruled out last snapshot but we are trying to utilise it as much as possible given the short time left. We will be adding the pre-defined tags to the annotation page as well as a highlighting function to help the annotator. Finally, we need to implement the annotator functionality, which involves viewing and editing profile, and incident report management including pending requests, rejected requests and approved requests, and Administrator functionality including annotator management and incident report management similar to the annotator functionality.

- "As a data annotator, I want to upload an incident report and save it so that I can further edit the file."

  – Related tasks:
  task 1.2: upload function
  task 1.3.2: implement database to store original documents

- "As an annotator, I want to annotate incident report so I can generate attack flow."

  – Related tasks:
  task 2.1.1: decide how user annotate
  task 2.1.2: implement user annotation function
  task 2.1.3: integrate ChatGPT to parse report and give accurate annotation suggestions

- "As an admin, I would like to see the pending annotated report requests, the approved requests, and rejected requests in lists."

  – Related tasks:
  task 1.3.3: Store annotatations in database
  task 4.2: Implement different user roles in the database

Definition of Done

The individual DoD of each task can be seen on the screenshot of the task board.
In summary, the goals are:

- Testing the integrated backend functions works as we expected.
- Code is well-documented and adheres to our coding standards.
- Explain and ensure that all the scrum members understand the specifics of the tasks and how to construct the implementation.

Summary of Changes

This snapshot will still focus on the upload and annotate functions but the focus will now be on implementing the ‘keywords/tags’ function as well as the highlight function. ChatGPT, however, will be implemented as much as possible without compromising the end goal of the project which is a backflip on last snapshot. The annotator and administrator user roles are now also in the main focus.

- **Annotation function to be completed including the ability to view and edit profile.** We will continue with the annotate functionality including how to annotate? How to use keywords/tags? How to use predefined keywords/tags? How to incorporate keywords from Mitre Att&ck? Must identify tags and be able to manually search from drop down menu to define relevant tag as well as profile functionality.

- **This snapshot re-enstates the major role of ChatGPT in parsing the report and providing accurate annotation suggestions.** We have now decided to utilise the ChatGPT as much as possible in our web application to extract metadata and as much information as possible with regard to extracting techniques.

    - **Administrator functionality to be implemented.** The administrator functionality needs to be implemented including annotator management (delete annotator from the database and else) incident report management (pending request, approved request, rejected request).