Attack Flow

Retrospective Sprint 2 of Group AttackFlow 10

Jie Shen Beh (a1834032)
Jian Zhe Chan (a1813851)
Gia Bao Hoang (a1814824)
Marcus Hoang (a1814303)
Guan Chern Liew (a1837053)
Vinh Diem Nguyen (a1838114)
Hoang Nam Trinh (a1807377)
Hung Yee Wong (a1815836)
Jiajun Yu (a1806320)

What went well in the sprint?

This sprint exemplified our team's cohesion and commitment. Every team member consistently showed up on time and was deeply involved in discussions, translating into collective clarity on our roles and the subsequent steps. Strategically, we split our workforce into two focused groups. The first group centered its energies on foundational aspects, paving the way for the web application's core structure. Meanwhile, the second group immersed itself in the crucial tasks of annotation and visualization.

We made a significant decision concerning the potential use of AI for the Attackflow generation. Instead of embracing AI unquestioningly, we prioritize our product's integrity. By choosing manual user inputs, we showcased our team's adaptability within the Scrum framework, making decisions based on quality over novelty.

Additionally, our approach to document collaboration further reflected our team's agile thinking. We ensured everyone could contribute smoothly and without overlap. Receiving positive feedback from our client was a highlight. It not only bolstered our confidence but also affirmed that our Scrum processes align seamlessly with software development objectives. A crowning achievement was the intuitive login system, underscoring our focus on both technical robustness and user experience.

What could be improved?

Receiving feedback from the client, we realized there were missed opportunities this sprint. They had hoped for the completion of a primary feature in conjunction with the milestones we had already reached. The implication of this lag in our software development process was evident, as the team had to recalibrate its goals, affecting overall morale and pacing.

Stepping into the scrum master's shoes for the first time, I encountered a steep learning curve. While this role has broadened my perspective, it also highlighted areas, particularly in client communication, needing refinement. Misunderstandings or misalignments with client expectations can lead to inefficiencies, impacting the team's momentum and focus. Enhancing this communication is crucial for the team to ensure that our development aligns with client needs seamlessly.

Several team members grappling with their academic commitments faced challenges in adhering to our project timelines. This imbalance not only puts strain on individual members but also has ripple effects across the team, possibly slowing down progress and reducing collaborative efficiency. To foster a healthier team dynamic and smoother software development process, we're emphasizing improved time management, equitable task distribution, and setting more grounded performance benchmarks in upcoming sprints.

What will the group commit to improve in the next sprint?

In the upcoming sprint, our foremost goal is to effectively introduce the annotation tool while offering a preliminary glimpse into the visualization aspect. This advancement will address our previous delay in feature implementation, ensuring that our software meets both the user's and client's expectations.

Recognizing the gaps in our time management, we commit to introducing structured daily stand-ups, allowing us to regularly sync, prioritize tasks, and address any roadblocks immediately. Such regular checks will ensure efficient task distribution and offer clarity, enabling us to set and achieve realistic milestones.

To nurture a more cohesive team environment, we'll institute weekly feedback sessions. These sessions will act as platforms where team members can voice opinions, discuss potential challenges, and brainstorm solutions. A proactive feedback loop can fast-track issue resolution, driving efficiency in our software development process.

Lastly, building on our 'what could be improved' insights, we'll roll out an enhanced progress monitoring system. By utilizing tools like burndown charts and sprint backlogs, we aim to gain a visual representation of our progress and pace. This visual aid will help the team preemptively identify lags and ensure timely intervention, keeping our software development process agile and on target.

Comment on your progress this sprint

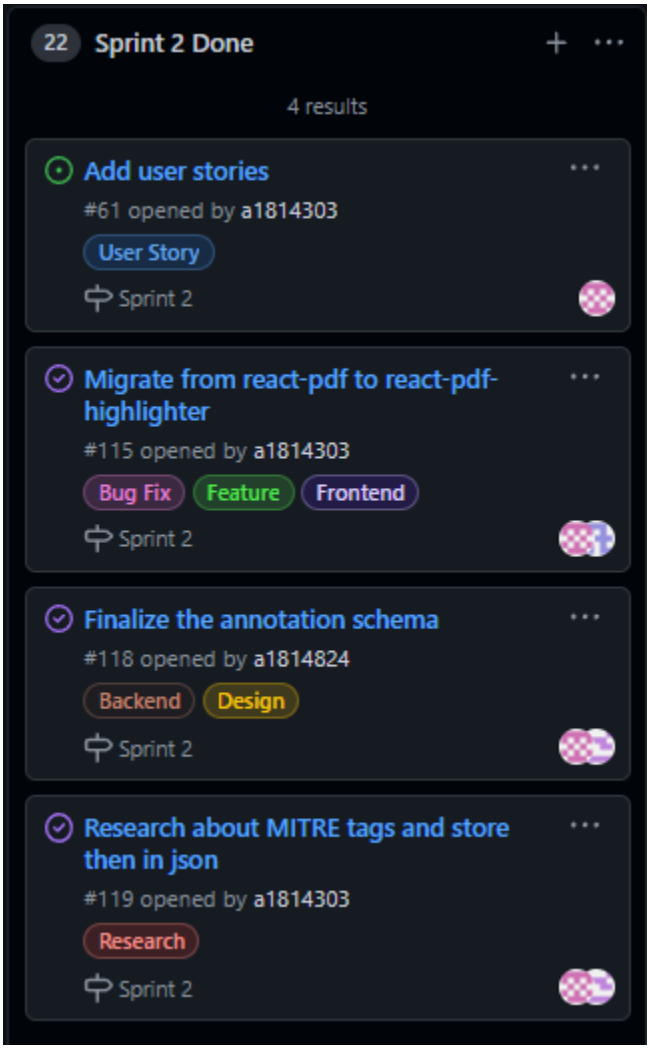


Image 1: Issues for Sprint 2

This sprint was a rich tapestry of personal growth, challenges, and collaborative solutions. As I transitioned into the Scrum Master role for the first time, I was introduced to the intricate balance between leading a team and diving deep into the project's technicalities.

Crafting user stories wasn't merely about writing them; it involved dissecting the project requirements, understanding their nuances, and ensuring every story effectively communicated our roadmap and aligned with the client's vision. The challenge lay not just in capturing the essence but in the meticulous breakdown required to align our goals with client expectations.

My research into MITRE Tactics and Techniques was a complex undertaking. It wasn't about gathering data, but about translating vast amounts of information into a user-friendly JSON format that could seamlessly integrate with our system.

On the collaborative front, migrating to react-pdf-highlighter posed its challenges. My teammate and I navigated through the intricacies of integrating a new tool while ensuring we didn't lose any existing functionality. Likewise, finalizing the annotation schema was no straightforward feat. It required a blend of technical knowledge and foresight to create a framework that would stand the test of future requirements.

In conclusion, sprint 2 was as much about overcoming complexities as it was about achieving our objectives. The blend of individual perseverance and team collaboration ensured that we rose to every challenge presented.

Snapshots

I attended the sprint review/planning meeting on 22/08/2023 with the tutor.
I attended the sprint review/planning meeting on 05/09/2023 with the tutor.

Attack Flow

Snapshot Week 5 of Group
AttackFlow 10

Jie Shen Beh (a1834032)
Jian Zhe Chan (a1813851)
Gia Bao Hoang (a1814824)
Marcus Hoang (a1814303)
Guan Chern Liew (a1837053)
Vinh Diem Nguyen (a1838114)
Hoang Nam Trinh (a1807377)
Hung Yee Wong (a1815836)
Jiajun Yu (a1806320)

Product Backlog and Task Board

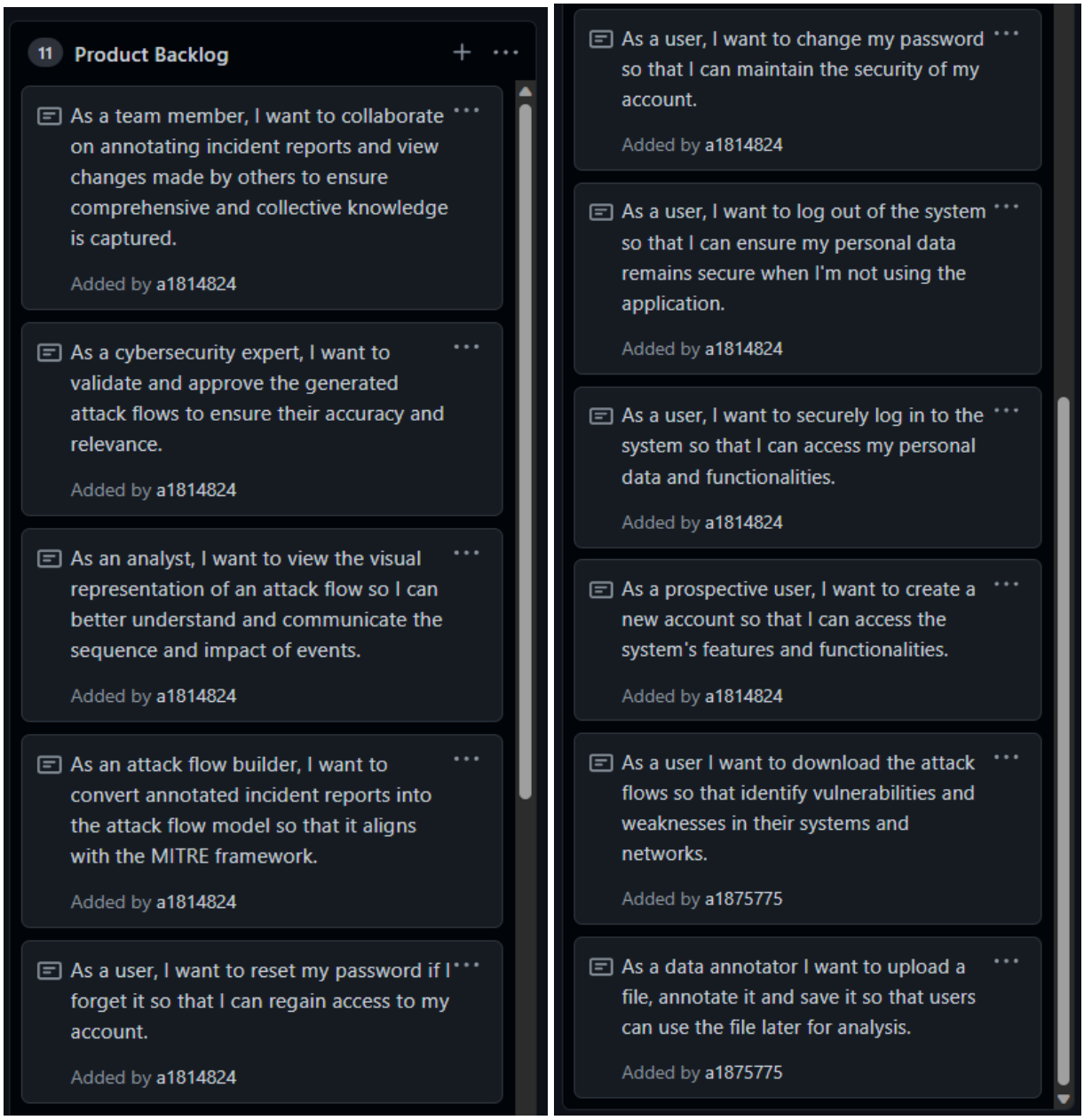


Image 1: Snapshot 2.1 Product Backlog

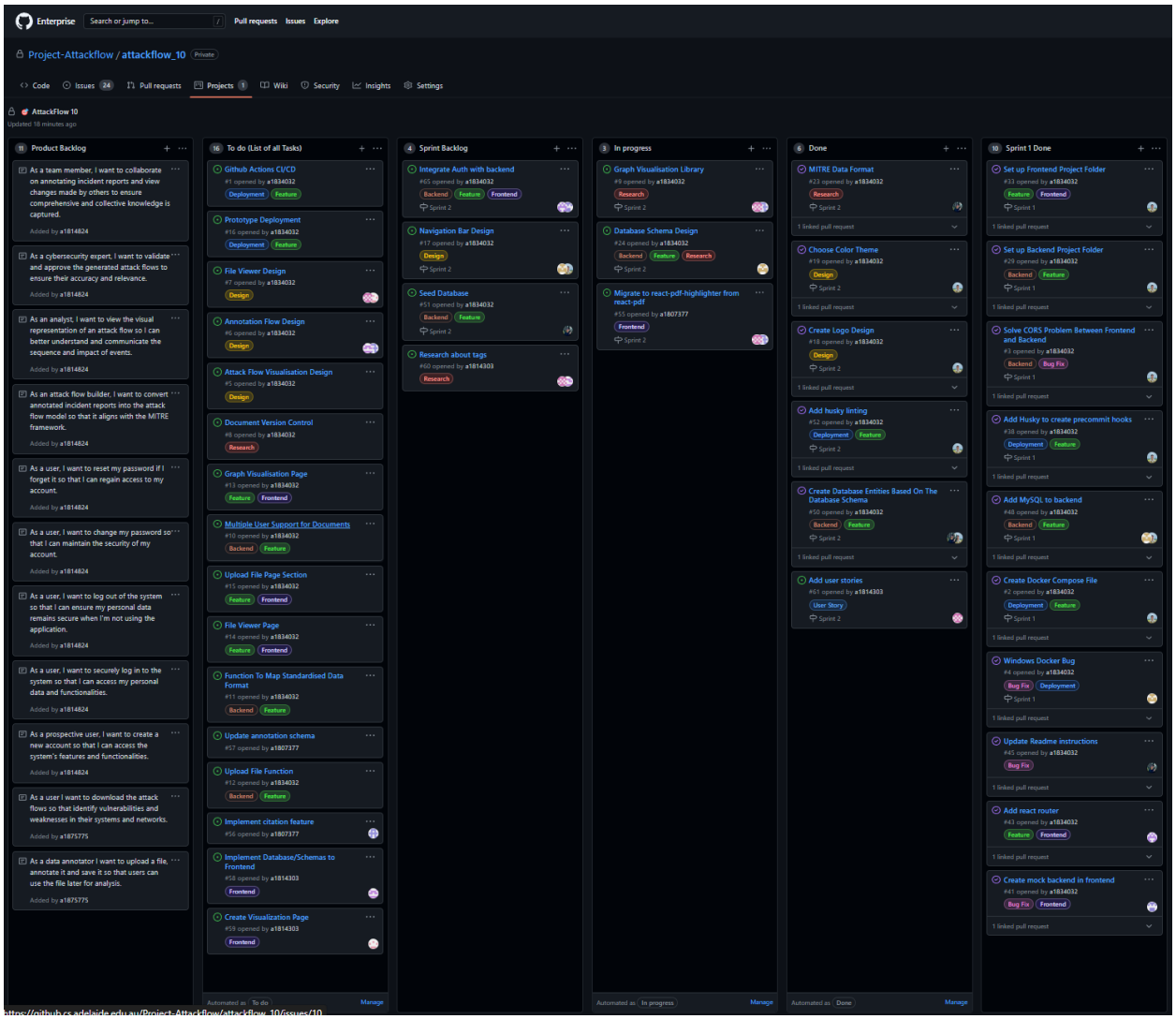


Image 2: Snapshot 2.1 Task Board

Sprint Backlog and User Stories

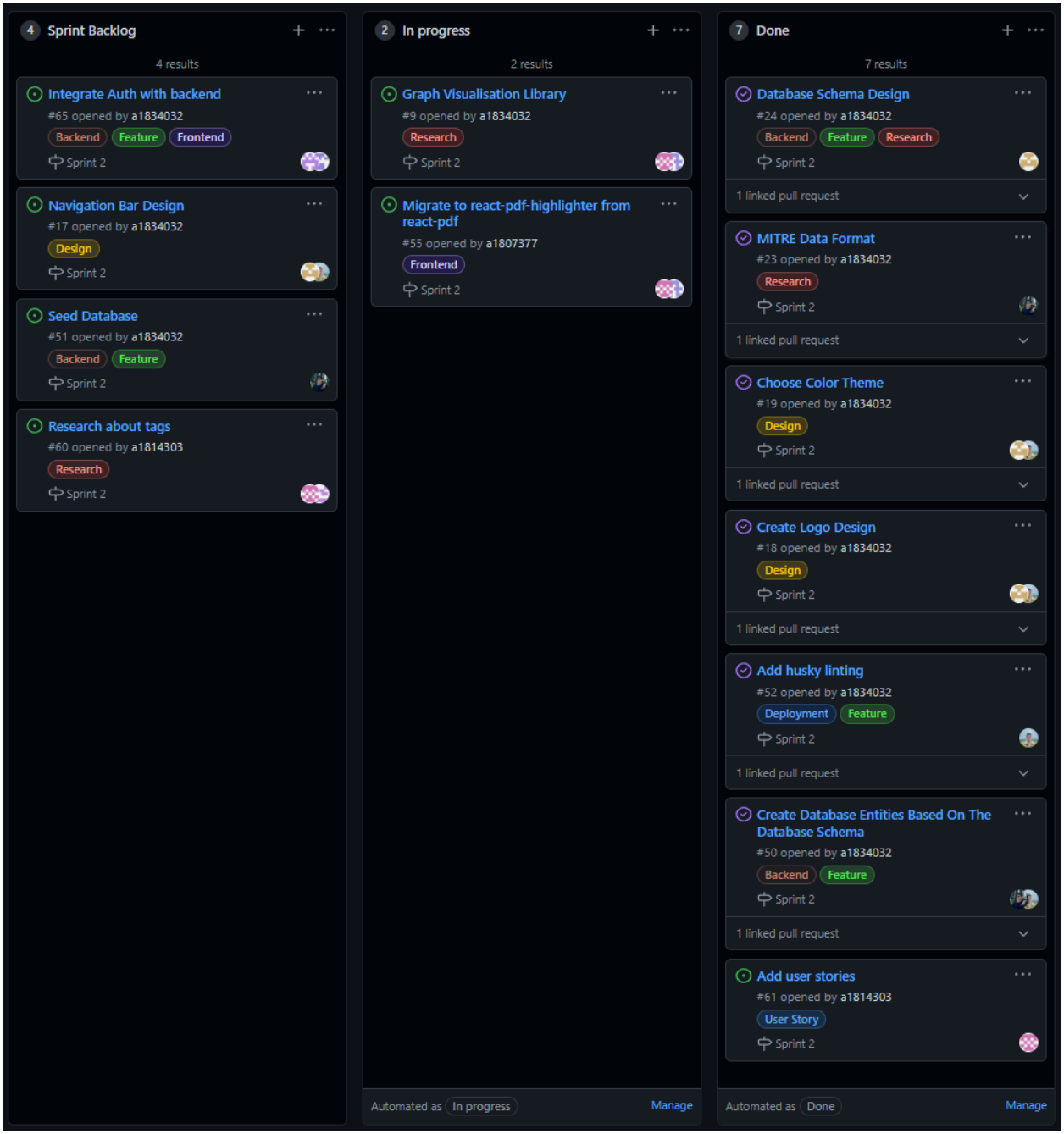


Image 3: Snapshot 2.1 Sprint Backlog

This screenshot depicts the three distinct stages of our sprint backlog: tasks queued in the backlog (Sprint Backlog), tasks currently underway (In progress), and tasks that have been successfully completed (Done).

User Stories

1. As a user I want to download the attack flows so that I can identify vulnerabilities and weaknesses in their systems and networks.
    a. Description: The user will be able to download the attackflows after it has been published. This feature enables them to simulate potential attack scenarios and strengthen their defenses for prevention.
    b. Related tasks include researching on MITRE Data Format and backend function to map standardized data format.
2. As a data annotator I want to upload a file, annotate it and save it so that users can use the file later for analysis.
    a. Description: The data annotator will be able to create annotations using their cybersecurity knowledge on an uploaded incident report. This feature will create a standardized data format for visualization. Then, users can analyze the visualization and prevent future cyberattacks in their systems.
    b. Related tasks include setting up the frontend project folder, designing the file viewer after a user uploads the file and researching on the document version control.
3. **As a prospective user, I want to create a new account so that I can access the system's features and functionalities.**
    a. Description: The user should be presented with a registration form that captures all necessary details for account creation. Upon successful registration, they should receive a confirmation email or notification.
    b. Related Tasks:
        - Design the registration UI. (Done)
        - Implement backend logic for account creation.
        - Research and integrate email notification systems for account confirmations.

4. **As a user, I want to securely log in to the system so that I can access my personal data and functionalities.**
    a. Description: The user should be presented with a simple and intuitive login form that asks for their credentials. Any unsuccessful login attempts should provide feedback to guide the user.
    b. Related Tasks:
        - Design the login UI. (Done)
        - Implement backend authentication logic.
        - Integrate error handling and feedback mechanisms.

5. **As a user, I want to log out of the system so that I can ensure my personal data remains secure when I'm not using the application.**
    a. Description: The user should easily find and use the logout functionality. After logging out, their session should be terminated.
    b. Related Tasks:

- Add the logout button to the UI. (Done)
- Remove JWT Token after logout.

6. **As a user, I want to change my password so that I can maintain the security of my account.**
    a. Description: The user should be able to easily locate the change password option, be prompted for their current password, and then specify a new password.
    b. Related Tasks:
        - Design update password page. (Done)
        - Implement backend logic for password updates.
        - Return message and status of request for successful or unsuccessful password changes.

7. **As a user, I want to reset my password if I forget it so that I can regain access to my account.**
    a. Description: The user should find a "Forgot Password" option in the login screen, which guides them through the process of resetting their password via their registered email.
    b. Related Tasks:
        - Design UI for password reset. (Done)
        - Implement backend logic to send reset password emails.
        - Ensure secure token generation for password reset.

8. **As an attack flow builder, I want to convert annotated incident reports into the attack flow model so that it aligns with the MITRE framework.**
    a. Description: The system should interpret annotations and use them to create a standardized attack flow model.
    b. Related Tasks:
        - Understand and implement the MITRE framework format.
        - Design algorithms or tools to convert annotations into attack flow models.

9. **As an analyst, I want to view the visual representation of an attack flow so I can better understand and communicate the sequence and impact of events.**
    a. Description: Once the attack flow model is generated, the system should provide a visualization tool or integrate with the MITRE visualization tool.
    b. Related Tasks:
        - Integrate with the MITRE visualization tool or develop a custom visualization module.
        - Ensure visual clarity and interactive features for better analysis.

10. **As a cybersecurity expert, I want to validate and approve the generated attack flows to ensure their accuracy and relevance.**
    a. Description: Users should be able to review, modify, and approve attack flows to ensure they accurately represent the incident report.
    b. Related Tasks:
        - Design a review and validation interface.
        - Implement feedback feature for continuous improvement.

11. **As a team member, I want to collaborate on annotating incident reports and view changes made by others to ensure comprehensive and collective knowledge is captured.**
    a. Description: The platform should support multiple users annotating a document and provide version control to track changes.
    b. Related Tasks:
        - Integrate collaboration tools or features.
        - Implement a version control system for documents.

Note: The user stories are highlighted in **bold** because they are first introduced in this sprint.

For detailed insights into the Task Board, Product Backlog, and Sprint Backlog, please click here.

Definition of Done

For our second snapshot, our "definition of done" remains divided into two sections: general goals and specific goals. While these largely build upon the foundations set in our first snapshot, they have been expanded to incorporate fresh objectives and other pivotal functionalities that emerged from our recent discussions and user stories.

General Goals:

- **Testing**: All code has undergone rigorous testing and passes all unit tests.
- **Deployment**: The code is successfully deployed to a staging environment, accessible via a public URL.
- **Code Review**: The code has been reviewed, critiqued, and approved by at least one other developer.
- **Feedback Integration**: All feedback from reviewers has been addressed and incorporated.
- **User Experience**: The website provides a seamless experience on both mobile and web platforms, ensuring easy navigation for users.

Specific Goals:

- **User Registration**: A new user can smoothly register using an unused email address, receiving a confirmation notification after successful registration.
- **User Authentication**: Users can log in securely using their registered credentials. Additionally, the system provides clear feedback for any unsuccessful login attempts.
- **Password Management**: Users can change their passwords without any hassle. Moreover, if they forget their passwords, they can initiate a clear and secure recovery process.
- **User Logout**: Users can easily log out, ensuring their session is terminated and their personal data remains secure.
- **File Upload and Annotation**: Data annotators can upload files and annotate specific sections within them. The annotations should be clear and easily modifiable.
- **Attack Flow Integration**: The system can interpret annotations from uploaded files to generate attack flow models aligned with the MITRE framework.
- **Visualization**: Analysts and other users can view a clear visual representation of any attack flow, providing an easy understanding of sequences and impact.
- **Validation and Collaboration**: Cybersecurity experts can review and validate generated attack flows. Team members can collaborate on annotations and utilize a version control system to track any changes to incident reports.
- **Download Capability**: Users can download the attack flows for offline analysis or sharing. The downloaded format should be in line with the MITRE Data Format.

Summary of Changes

Since our last update, we've been making significant progress. We've built a strong foundation for the system by carefully creating a solid database structure. Our new authentication system is now operational, enhancing security for user interactions. The real achievement was seamlessly connecting the backend with the frontend, resulting in a user-friendly experience that flows smoothly. To give our project a unique identity, we've crafted an eye-catching logo and selected colors that truly reflect its essence. And that's not all – we've also successfully developed two key interface elements: the file viewer page and attack flow nodes list. Additionally, we've integrated a side menu to enhance navigation. These improvements showcase our dedication to both functionality and aesthetics, marking a clear advancement since the last snapshot.

Attack Flow

Snapshot Week 6 of Group
AttackFlow 10

Jie Shen Beh (a1834032)
Jian Zhe Chan (a1813851)
Gia Bao Hoang (a1814824)
Marcus Hoang (a1814303)
Guan Chern Liew (a1837053)
Vinh Diem Nguyen (a1838114)
Hoang Nam Trinh (a1807377)
Hung Yee Wong (a1815836)
Jiajun Yu (a1806320)

Product Backlog and Task Board

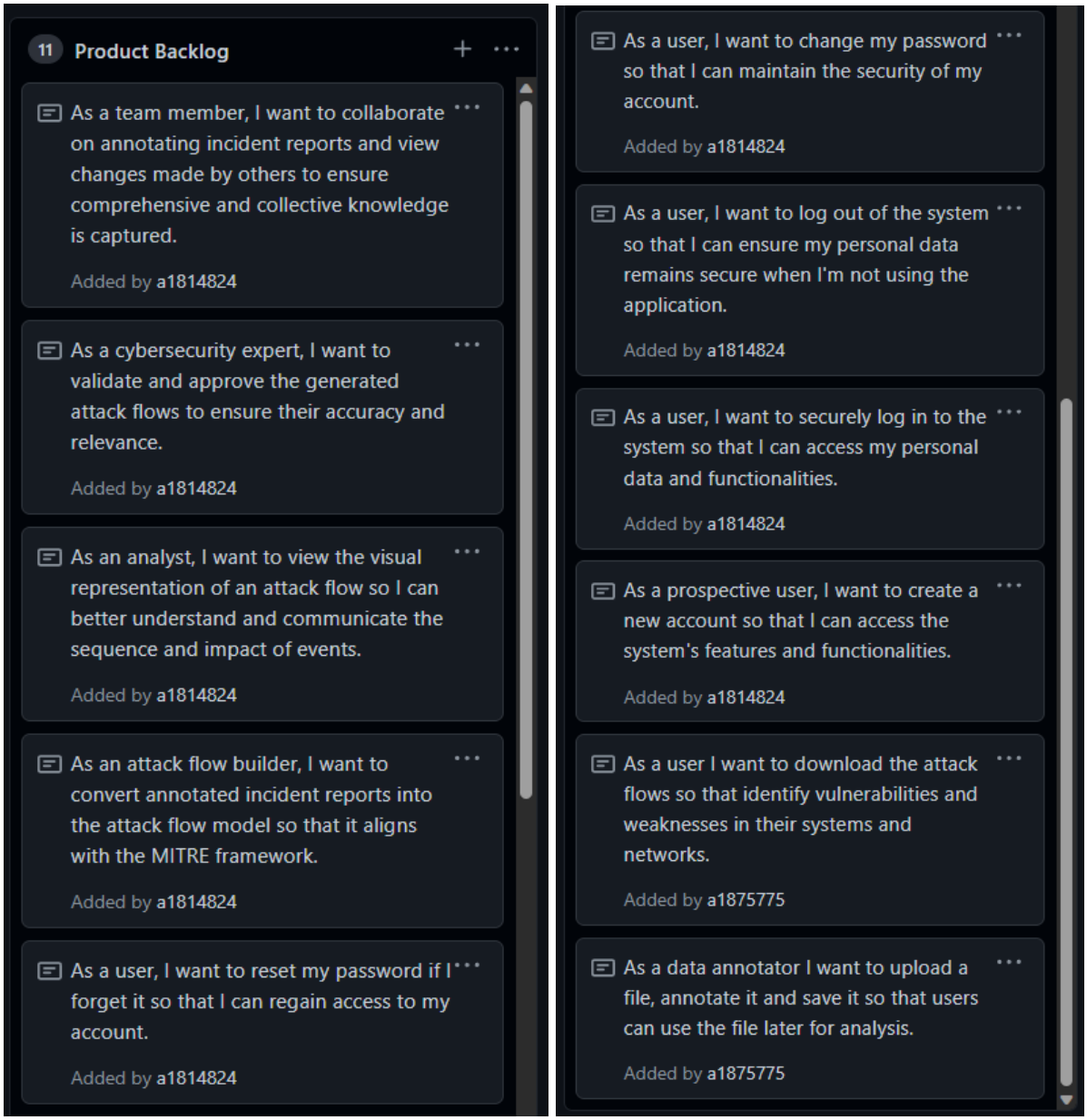


Image 1: Snapshot 2.2 Product Backlog

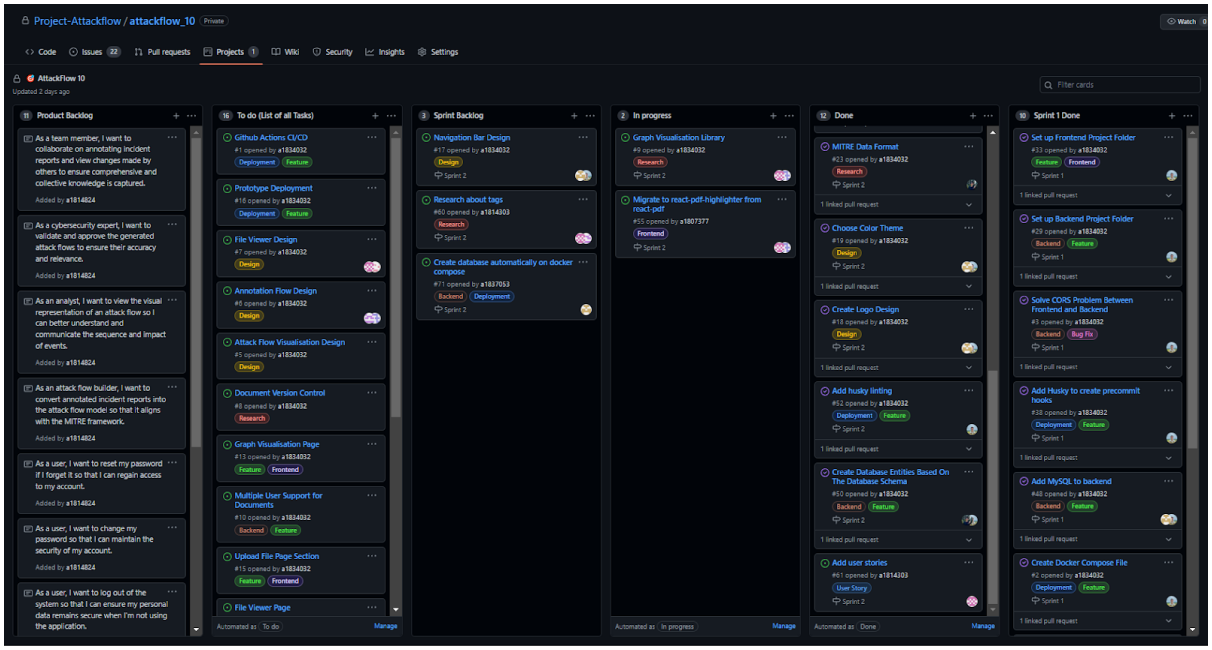
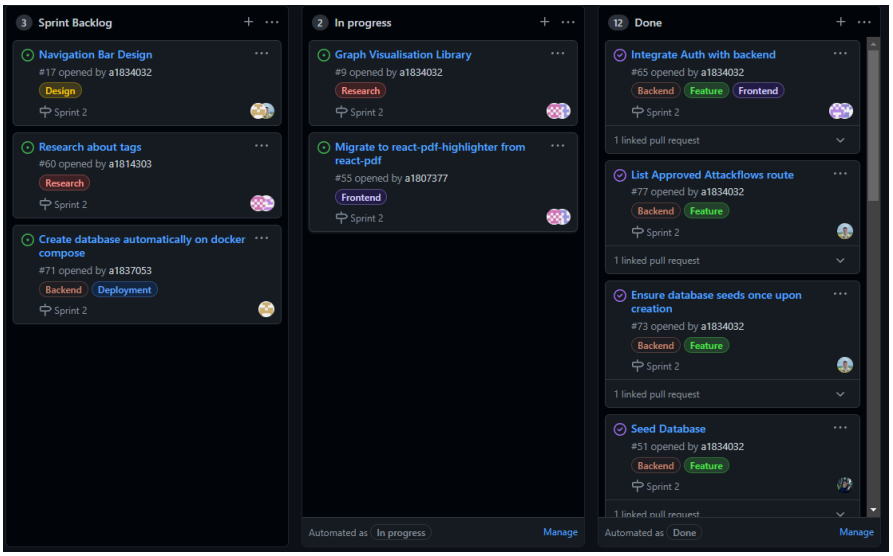
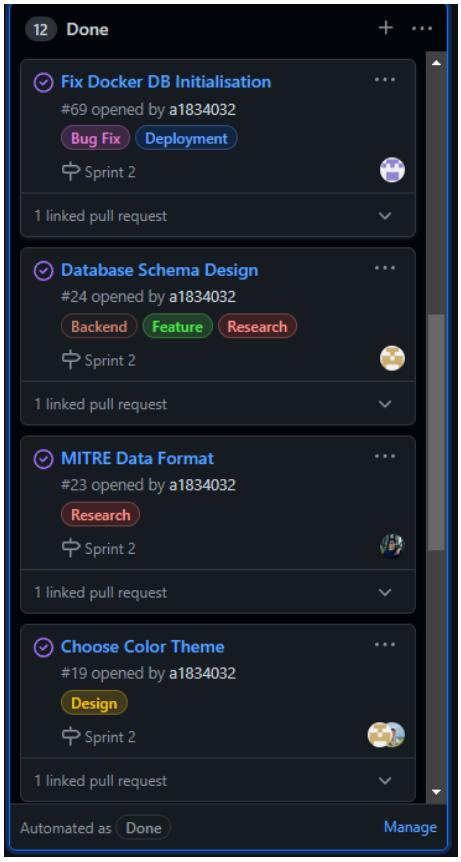


Image 2: Snapshot 2.2 Task Board

Sprint Backlog and User Stories





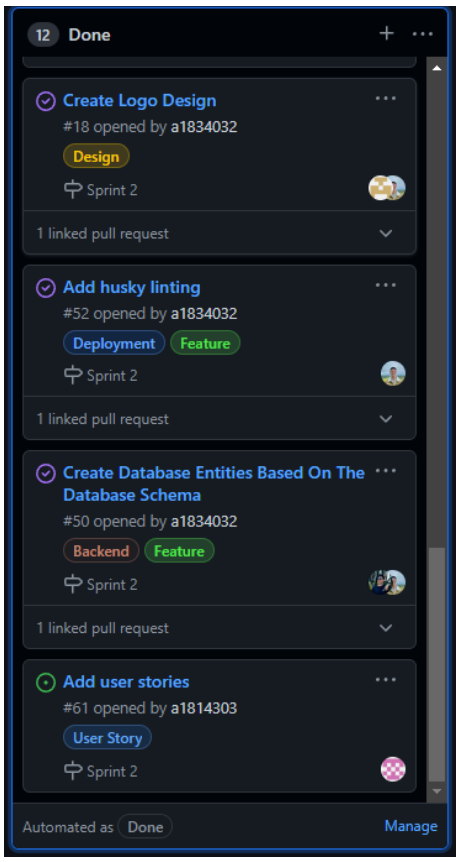


Image 3: Snapshot 2.2 Sprint Backlog

This screenshot depicts the three distinct stages of our sprint backlog: tasks queued in the backlog (Sprint Backlog), tasks currently underway (In progress), and tasks that have been successfully completed (Done).

User Stories of Current Sprint

1. As a prospective user, I want to create a new account so that I can access the system's features and functionalities.
    a. Description: The user should be presented with a registration form that captures all necessary details for account creation. Upon successful registration, they should receive a confirmation email or notification.
    b. Related Tasks:
        - Design the registration UI. (Done)
        - Implement backend logic for account creation. (Done)
        - Research and integrate email notification systems for account confirmations.

2. As a user, I want to securely log in to the system so that I can access my personal data and functionalities.
    a. Description: The user should be presented with a simple and intuitive login form that asks for their credentials. Any unsuccessful login attempts should provide feedback to guide the user.
    b. Related Tasks:
        - Design the login UI. (Done)
        - Implement backend authentication logic.
        - Integrate error handling and feedback mechanisms.

3. As a user, I want to log out of the system so that I can ensure my personal data remains secure when I'm not using the application.
    a. Description: The user should easily find and use the logout functionality. After logging out, their session should be terminated.
    b. Related Tasks:
        - Add the logout button to the UI. (Done)
        - Remove JWT Token after logout.

4. As a user, I want to change my password so that I can maintain the security of my account.
    a. Description: The user should be able to easily locate the change password option, be prompted for their current password, and then specify a new password.
    b. Related Tasks:
        - Design update password page. (Done)
        - Implement backend logic for password updates.
        - Return message and status of request for successful or unsuccessful password changes.

5. As a user, I want to reset my password if I forget it so that I can regain access to my account.

a. Description: The user should find a "Forgot Password" option in the login screen, which guides them through the process of resetting their password via their registered email.
b. Related Tasks:
    - Design UI for password reset. (Done)
    - Implement backend logic to send reset password emails.
    - Ensure secure token generation for password reset.

For detailed insights into the Task Board, Product Backlog, and Sprint Backlog, please click here.

Definition of Done

For our third snapshot, our "definition of done" remains divided into two sections: general goals and specific goals. While these largely build upon the foundations set in our first snapshot, they have been expanded to incorporate fresh objectives and other pivotal functionalities that emerged from our recent discussions and user stories.

General Goals:

- **Testing**: All code has undergone rigorous testing and passes all unit tests.
- **Deployment**: The code is successfully deployed to a staging environment, accessible via a public URL.
- **Code Review**: The code has been reviewed, critiqued, and approved by at least one other developer.
- **Feedback Integration**: All feedback from reviewers has been addressed and incorporated.
- **User Experience**: The website provides a seamless experience on both mobile and web platforms, ensuring easy navigation for users.
- **Documentation:** Maintain documentation of project requirements, changes, and decisions. Share this documentation with the client to ensure that both parties have a clear understanding of project scope and goals.

Specific Goals:

- **User Registration**: A new user can smoothly register using an unused email address, receiving a confirmation notification after successful registration.
- **User Authentication**: Users can log in securely using their registered credentials. Additionally, the system provides clear feedback for any unsuccessful login attempts.

- **Password Management**: Users can change their passwords without any hassle. Moreover, if they forget their passwords, they can initiate a clear and secure recovery process.
- **User Logout**: Users can easily log out, ensuring their session is terminated and their personal data remains secure.
- **File Upload and Annotation**: Data annotators can upload files and annotate specific sections within them. The annotations should be clear and easily modifiable.
- **Attack Flow Integration**: The system can interpret annotations from uploaded files to generate attack flow models aligned with the MITRE framework.
- **Visualization**: Analysts and other users can view a clear visual representation of any attack flow, providing an easy understanding of sequences and impact.
- **Validation and Collaboration**: Cybersecurity experts can review and validate generated attack flows. Team members can collaborate on annotations and utilize a version control system to track any changes to incident reports.
- **Download Capability**: Users can download the attack flows for offline analysis or sharing. The downloaded format should be in line with the MITRE Data Format.

Summary of Changes

After a highly productive week, we seamlessly implemented the database structure into the backend system and thoughtfully populated it with some seeding data. All data entities were meticulously initialized with some formatted value to facilitate fundamental testing. After thorough testing, we unearthed some significant bugs that were promptly fixed to ensure smooth development. Notably, we optimized our data seeding process and database creation process to execute only once upon container launch using Docker, eliminating redundancy.

In parallel, we also fortified our security framework with connecting the frontend authentication to the backend, establishing a robust system that grants access exclusively to authorized users for designated routes. This enhances the overall security and reliability of our application, contributing to its seamless functionality.