Taskforce

In-house Project Management Software

Sprint One

Version 1.0

Written by:
Brian Nge Jing Hong
Muhammad Ibrahim bin Mohd Yusni
Chua Xian Loong
Lucas Wee
Koe Rui En
Diana Wijaya

Taskforce

Contents

1. Sprint Planning

1.1 Important Links

Link to Trello board:
https://trello.com/invite/b/XmURix1g/ATTIcca68202ded99301ef54aa2fd5dc1e114A503979/project-1

Link to the meeting minutes: 🅦 Meeting 1 - 4/9/2023

Link to Refined User Stories: ⊞ User Stories

Link to Task Allocation: 🗐 Task Allocations

1.2 Product Backlog

The product backlog consists of all the user stories that the team has decided for the project. The product backlog is logged on Trello. The link to the Trello board is provided in section 1.1. To decide on the user stories as well as its story points, a meeting was held on 04 September 2023, Monday from 9:00 PM to 10:45 PM. The meeting minutes for the meeting can be accessible through the link provided in section 1.1. It is confirmed that the product backlog follows the DEEP criteria. Additionally, the user stories are refined according to the INVEST criteria before placing it into Trello. The refined user stories can also be accessible through the spreadsheet link in section 1.1.
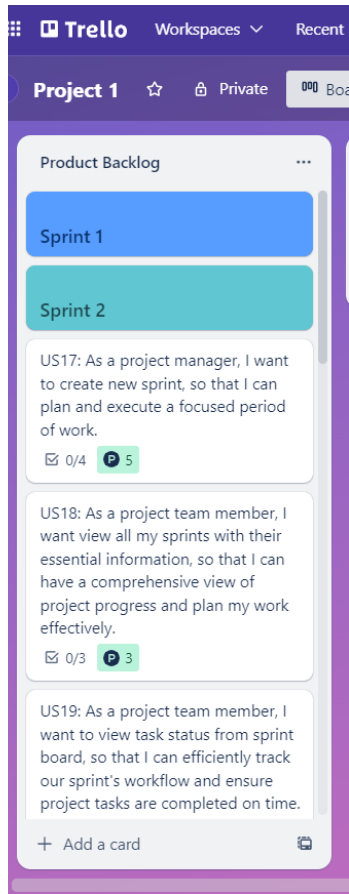
Several spike stories are also added in Trello, for team members who are not familiar with the programming languages to work on their skill sets while working on the project. Each spike story includes a link to a youtube video that is sufficient for team members who are not familiar with the programming language to gain necessary knowledge for the project.

Access to essential links such as the UI/UX design decisions and the web application prototype are also attached in trello, allowing easier access to the important links for team members.
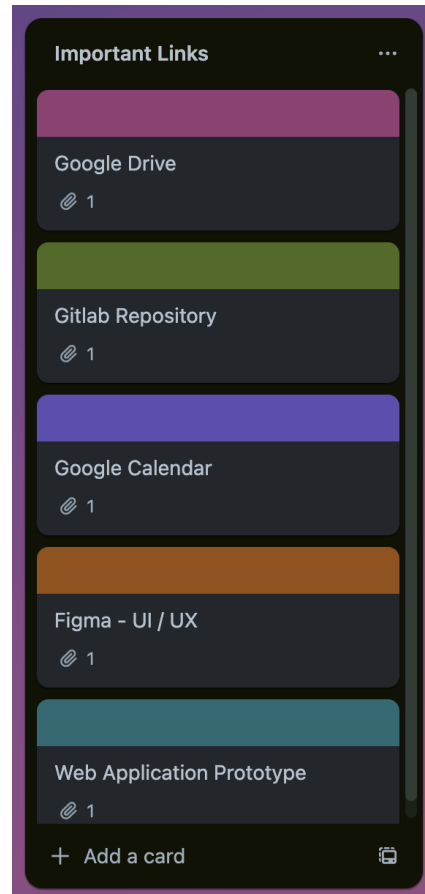
Some documentation examples for the product backlog and spike stories are attached below.
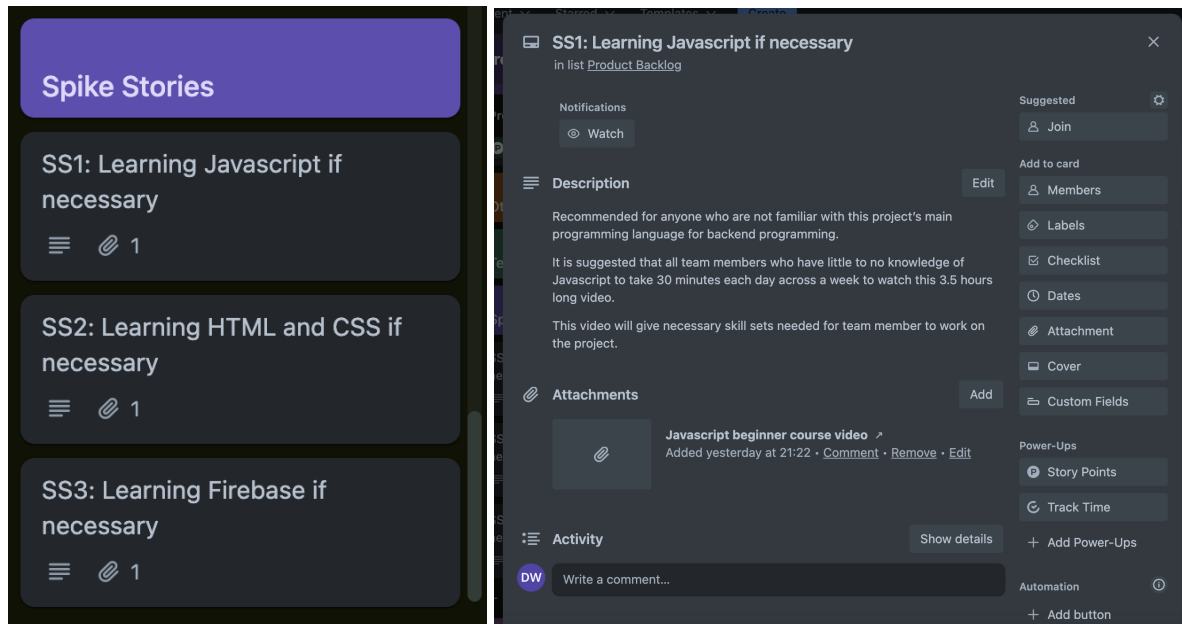
Taskforce

Product backlog:

Essential links:

Trello | Workspaces ∨ | Recent

Project 1 ☆ 🔒 Private ▯▯▯ Boa

Product Backlog ...

Sprint 1

Sprint 2

US17: As a project manager, I want to create new sprint, so that I can plan and execute a focused period of work.
☑ 0/4 Ⓟ 5

US18: As a project team member, I want view all my sprints with their essential information, so that I can have a comprehensive view of project progress and plan my work effectively.
☑ 0/3 Ⓟ 3

US19: As a project team member, I want to view task status from sprint board, so that I can efficiently track our sprint's workflow and ensure project tasks are completed on time.

+ Add a card

Important Links ...

Google Drive
📎 1

Gitlab Repository
📎 1

Google Calendar
📎 1

Figma - UI / UX
📎 1

Web Application Prototype
📎 1

+ Add a card

Spike stories:

Spike Stories

SS1: Learning Javascript if necessary
☰ 📎 1

SS2: Learning HTML and CSS if necessary
☰ 📎 1

SS3: Learning Firebase if necessary
☰ 📎 1

🖵 SS1: Learning Javascript if necessary   ✕
in list Product Backlog

Notifications   Suggested ⚙
👁 Watch   👤 Join

                                                       Add to card
☰ Description   Edit   👤 Members
Recommended for anyone who are not familiar with this project's main   🏷 Labels
programming language for backend programming.   ☑ Checklist
It is suggested that all team members who have little to no knowledge of   🕑 Dates
Javascript to take 30 minutes each day across a week to watch this 3.5 hours   📎 Attachment
long video.   🖵 Cover
This video will give necessary skill sets needed for team member to work on   ⛃ Custom Fields
the project.
                                                       Power-Ups
📎 Attachments   Add   Ⓟ Story Points
        📎   Javascript beginner course video ↗   🕑 Track Time
            Added yesterday at 21:22 · Comment · Remove · Edit   + Add Power-Ups

☰ Activity   Show details   Automation ⓘ
DW Write a comment...   + Add button

Taskforce

1.3 Sprint backlog

The sprint backlog is logged into Trello, provided link is in section 1.1. All the user stories for the first sprint are decided during the first meeting, as shown in the meeting minute 1. The story point for each user story was decided through planning poker, where each team member will decide on the story point individually, then discussions were raised when there were wide spreads of story point choices, ultimately resulting in one agreed story point per user story. The User Acceptance Criteria is included under each individual task, showing the expectations of each user story with clear definition and standard. The story point is clearly indicated under each user story in the sprint backlog. We have also discussed the task allocation and each team member has assigned themselves to their respective tasks. The link to the summary of task allocations is attached in section 1.1.

Sprint Backlog:
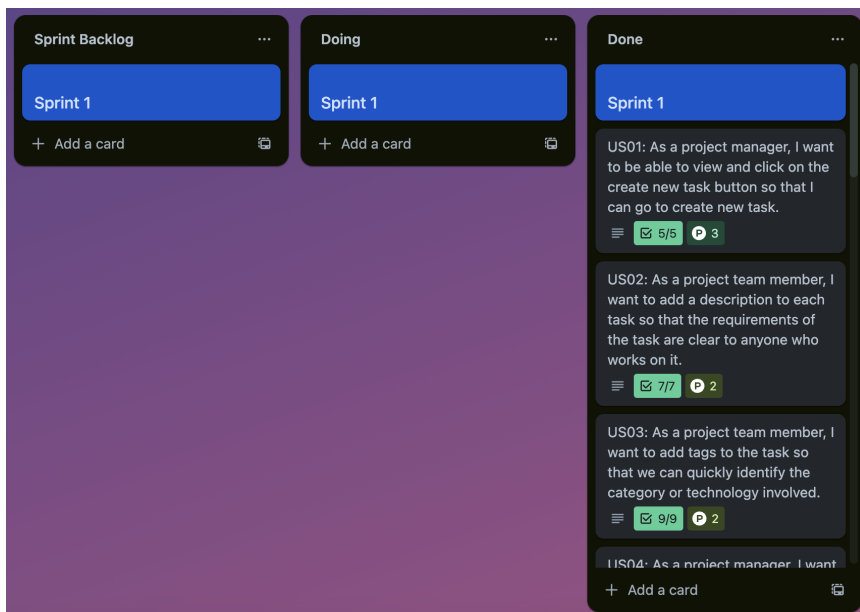
User Acceptance Criteria and Tasks:
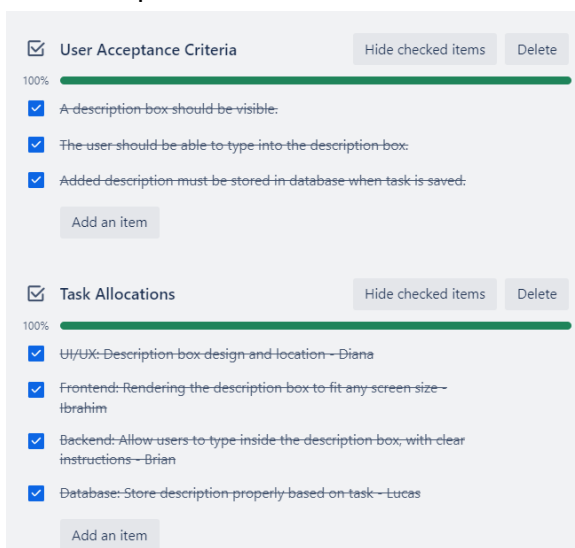
1.4 Sprint Goal

The sprint goal is to complete the product backlog page where tasks can be added with essential information, edited, deleted and filtered by priority or tags.

This sprint goal was decided in the meeting held on 04 September 2023, Monday from 9:00 PM to 10:45 PM. To achieve this sprint goal, we have come up with extensive planning and task allocation throughout the team. The details of the task allocation can be found on page 8 of the meeting minutes for 04 September 2023.

2. Daily Scrum

2.1 Important Links

Link to Trello board:
https://trello.com/invite/b/XmURix1g/ATTIcca68202ded99301ef54aa2fd5dc1e114A503979/project-1

Link to the meeting minutes: Ⓦ Meeting 2- 7/9/2023

Link to the Web Application Prototype: https://fit2101-taskforce.web.app/login.html
Credentials to Login:
- Email: admin@gmail.com
- Password: 123456

Link to GitLab repository:
https://git.infotech.monash.edu/FIT2101-S2-2023/MA_Thursday11am_Team1/project

2.2 Weekly Stand-Up Meeting

The team had the first stand-up meeting on 07 September 2023, from 12:00 PM - 12:30 PM. The link to this meeting minutes can be accessible via the link in section 2.1. During the meeting, each team member spoke on the following points:
- What has been accomplished since the last meeting?
- What will be done before the next meeting?
- What obstacles are in the way?

With this, each team member is able to understand what tasks each team member is working on.

We have also utilised the group chats to keep each other updated on current task status. Besides, we have also iterated the importance of the git policy that we have outlined in your project plan and made sure to only push to the "Firebase" branch and not the main branch.This is to prevent confusion of the code and ensure that everyone is able to work on

their end without much merge conflicts. It is also a good industry practice to work on a branch rather than on the main repository.

Throughout the standup meetings, the team was able to complete the following user stories. Further details on the user stories can be found in the trello link provided in section 2.1.

1. US01: As a project manager, I want to be able to view and click on the create new task button so that I can go to create new tasks.
2. US02: As a project team member, I want to add a description to each task so that the requirements of the task are clear to anyone who works on it.
3. US03: As a project team member, I want to add tags to the task so that we can quickly identify the category or technology involved.
4. US04: As a project manager, I want to set a priority to each task, so that the team knows what to focus on first.
5. US05: As a project team member, I want to assign story points to a task, so that we can estimate the effort needed to complete it.
6. US06: As a project manager, I want to assign each task to a maximum of one team member so that responsibilities are clear.
7. US08: As a project team member, I want to see all the essential information of each task, so that I can see all the critical information at once.
8. US09: As a project manager, I want to delete tasks in the product backlog so that we can remove items that are no longer relevant or needed.
9. US10: As a project team member, I want to filter tasks by tags so that I can focus on specific categories of work.
10. US11: As a project team member, I want to sort tasks by priority so that I can easily identify what should be worked on first.
11. US12: As a project team member, I want to switch between different views so that I can choose the layout that helps me manage tasks most efficiently.
12. US13: As a project team member, I want to use a Barbie-themed colour scheme, so that it's more visually appealing.
13. US14: As a project team member, I want to record past responses and activities so that I can keep track of what was being recorded.
14. US15: As a project team member, I want to assign a status for my task in the project backlog, so that I can figure out what has not been started, done and completed.
15. US16: As a project team member, I want to be able to track my time spent on completing a task, so that I can see how much time I dedicated on each task and improve my skills.
16. US26: As a project team member, I want to be able to view credentials so that I know which account is logged in.
17. US29: As a project team member, I want to log in into the project management software, so that I can access project resources.

The time spent per week is also logged and kept track of by each team member. This is the time spent on the first week for each team member:

1. Brian Nge Jing Hong - 9 hours
2. Muhammad Ibrahim bin Mohd Yusni - 18 hours
3. Chua Xian Loong - 11 hours
4. Lucas Wee - 10 hours

5. Koe Rui En - 9 hours
6. Diana Wijaya - 10 hours

3. Sprint Review

3.1 Important Links

Link to Trello board:
https://trello.com/invite/b/XmURix1g/ATTIcca68202ded99301ef54aa2fd5dc1e114A503979/project-1

Link to the meeting minutes: 🅦 Meeting 3 - 11/9/2023

Link to the Web Application Prototype: https://fit2101-taskforce.web.app/login.html
Credentials to Login:
● Email: admin@gmail.com
● Password: 123456

Link to the Demo Video: https://youtu.be/CwDdp7zNTnU

Link to Full Live and Updated Risk Register: 🗒 Risk Register Updates

Link to GitLab repository:
https://git.infotech.monash.edu/FIT2101-S2-2023/MA_Thursday11am_Team1/project

3.2 Burndown Chart

A burndown chart is a visual tool used in project management, specifically in Agile and Scrum methodologies, to track the progress of work over time. It helps teams and stakeholders monitor how tasks or user stories are being completed and whether the project is on track to meet its goals and deadlines. The team has utilised GitLab for its burndown chart feature. The user stories from Trello were placed into GitLab and issues were closed when we implemented that feature in the web application. The burndown chart and burnup chart can be found here:

Link to GitLab Milestone:
https://git.infotech.monash.edu/FIT2101-S2-2023/MA_Thursday11am_Team1/project/-/milestones/2#tab-issues
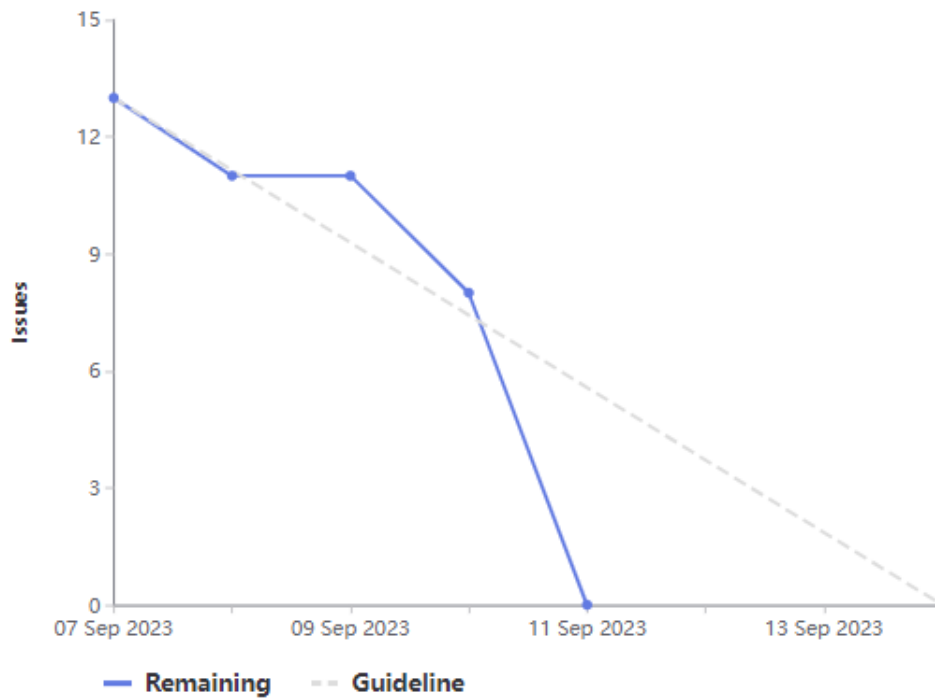
An overview of the burndown chart and burnup chart by the end of the sprint is shown below. As shown in the charts, all the tasks are successfully completed on time with no issue.
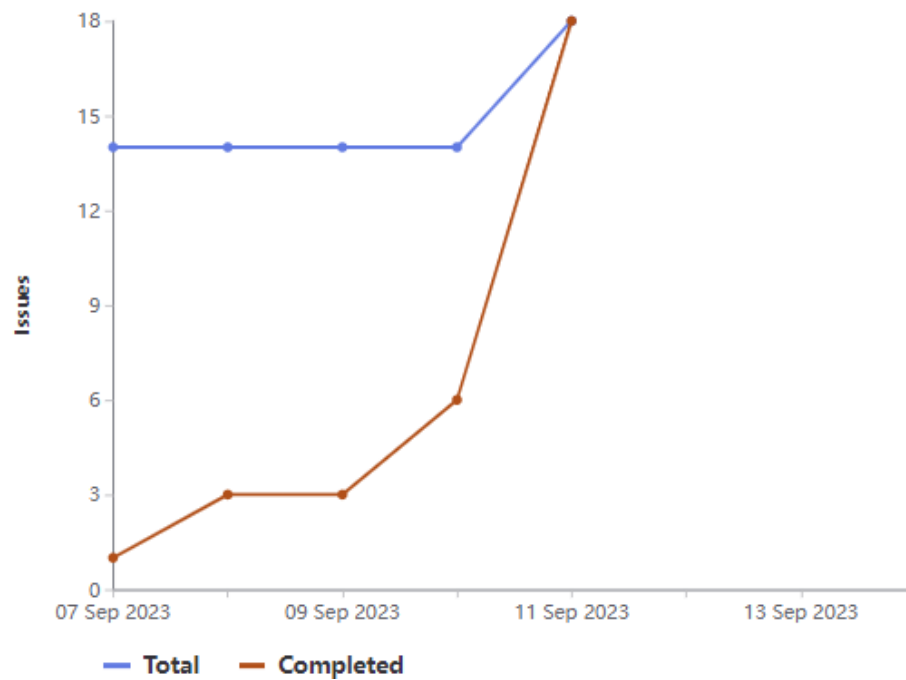
Taskforce

Burndown Chart:

Burndown chart

Burnup Chart:

Burnup chart

Taskforce

3.3 Risk Register

In Scrum, a risk register is a document or tool used to identify, assess, and manage risks that may impact the successful delivery of a project or a sprint. It is a proactive approach to risk management and it is beneficial in a project management setting. The team has updated the risk register accordingly. A full overview of the live and updated risk register, along with the existing risk register is included in a link provided in section 3.1. The new risks discovered are shown in table 3.3.1, while the live and updated risk register is shown in table 3.3.2.

ID	Date raised	Risk Description	Likelihood of the risk occurring	Impact if the risk occurs	Severity	Owner	Monitoring Strategy	Mitigation Plan
26	9/9/2023	Firebase Interruption	Low	High	Low	Scrum team	Ensure that each member hosts their own Firebase instance for testing purposes so that it will not interrupt the main Firebase instance for the software.	1. Utilise multiple Firebase projects to minimise the impact of interruptions on a project. 2. Regularly backup critical user data stored in Firestore to prevent data loss in case an interruption occurs.

Table 3.3.1: New Risks Discovered

ID	Occuration Date	Risk Description	Severity	Details	How the issue was resolved	Future Mitigation
22	4/9/2023	Data Loss Risk	Low	1. One team member accidentally pushed and committed the changes into the main branch. 2. Some coding data that was previously done by other members were lost. 3. Data lost was not severe, and was retrievable.	Team members reverted the changes and made sure the lost data were retrieved.	1. Avoid accidents like this from happening. 2. Make sure all changes made are informed to the group. 3. Report the accident immediately if any occurred.

19	5/9/2023	Laptop hardware breakdown risk	Low	1. One team member's laptop broke down during the middle of the sprint. 2. No data was lost, as everything was backed up in GitLab.	Laptop was fixed on the day, not a lot of time was wasted.	1. Ensure that backup copy is made and saved after every change. 2. Utilise gitlab for version control to save code to a repository. 3. Ensure that all team members have their own backup of the new code.
26	9/9/2023	Firebase Interruption	Low	1. Everyone working on the code was working on the same firebase instance. 2. There were lots of instances where team members were confused about the tasks added or deleted when testing.	Not a big issue, as it only happened during testing. Some team members decided to host their own Firebase for testing purposes. No important data was lost.	1. Utilise multiple Firebase projects to minimise the impact of interruptions on a project. 2. Regularly backup critical user data stored in Firestore to prevent data loss in case an interruption occurs.

Table 3.3.2: Live and Updated Risk

3.4 Sprint Review Planning

This is the sequence for the demo:
1. Brian, the Scrum Master and Assistant Programmer will briefly introduce the team, sprint goal and project information.
2. Ibrahim, the Product Owner and Front End Developer, will briefly show the sprint backlog and user stories.
3. Lucas, the Database Programmer will be introducing the login, registration user information page.
4. Diana, the UI/UX Designer will be introducing the list view, card view and filter tags functionality.
5. Brian, the Scrum Master and Assistant Programmer will be introducing the text box functionality in the "Add A Task" page.
6. Chua, the Head Programmer will be introducing the checkbox, dropdown and calendar functionality within the "Add A Task" page.
7. Rui En, the Technical Writer will be introducing the edit functionality in the software as well as the delete functionality in the software.

8. Ibrahim, the Product Owner and Front End Developer will be wrapping up the demo.

3.5 Product Examination

To ensure that the product is complete and shippable after the first sprint, the team has performed an intensive review of the product and cross-checked it with the user stories recorded in the sprint backlog of sprint one. The team has also reviewed the user acceptance criteria and tasks checklist for each user story to ensure that the user stories in the sprint backlog are actually completed. The designated user stories were confirmed as "done".

4. Sprint Retrospective

4.1 Main Criteria

Full discussions on the sprint retrospective can be found in the meeting agenda for meeting 3, attached in section 3.1.

Based on the template given, the team has performed sprint retrospective on sprint one. We then, evaluate the sprint based on the following points:
1. What went well?
2. What were the problems encountered?
3. What could have been done better?
4. What will we try next?
5. What questions do we have?

What went well?	1. Everyone is responsive and would typically reply to messages within a day typically. This has greatly improved the team productivity and allowed us to complete the tasks on time. 2. Every team member is accountable for their tasks and roles, leading to a proactive work ethic in the team where team members help others who are struggling to complete their task allocations. 3. Work was evenly distributed between all team members. This improved the team dynamic as there were no team members that were doing most of the work or any that did little to no work. 4. Everyone completes their tasks on time. This made our group's sprint go smoothly. We can also edit the tasks if any errors are found. 5. Every team member worked on the project. This made the sprint go smoothly as there were no conflicts between the members as everyone understood what each other was doing. 6. Every team member attended every meeting and every team member was quite attentive during the meetings and asked relevant questions to the sprint which made the meetings very efficient, contributing to the overall success of the project.
What were the problems encountered?	1. User stories were not done correctly initially, resulting in a confusion in the team and a lot of user acceptance criterias and tasks allocations were duplicated. 2. There was no proper task allocation in Trello initially, leading to a few

	bottlenecks of unclear responsibilities, static allocation and lack of feedback mechanisms. 3. Some functionality is hard to implement as we need domain knowledge on some languages such as Javascript. Team members need it to learn in a shorter time as the duration for sprint is short. 4. The creation of the burndown chart was started late and most of the user stories were already completed. Therefore, the burndown chart is not an accurate representation at the start. 5. Initially we have poorly defined acceptance criteria which can potentially lead to incomplete or incorrect implementations. 6. There was limited knowledge on Javascript and Database overall in the team which led to the team not being able to grasp the idea to code the software at the start.
What could have been done better?	1. The team can have more standup meetings during the sprint. This is because we can then keep each other accountable as we would be reporting our progress during the standup meeting. 2. The team can improve the process of user stories refinement, which we can achieve by clarifying user acceptance criteria, break stories into smaller task allocations, conduct regular refinement meetings and follow the INVEST criteria. 3. Reply group message on time. Each team member can be more active to reply to group messages so that other team members do not feel neglected. 4. More specific task allocation for each person. Because there are some tasks where some people do both the front end and back end. There is not much specialisation. 5. Have more organised meetings - In the first sprint, we did not exactly have an actual agenda to follow each meeting, this resulted in everyone being confused of what we have to do in the meetings, and we wasted some time on deciding the tasks we have to do during the meeting, which should be done before the meeting. 6. Being more attentive and direct during standup meetings as this helps each group member grasp the detail of each update making it easier to understand the team's current status.
What will we try next?	1. We can inform the group whenever any of the team members starts a task to ensure member accountability and provide regular updates to the group which would ensure increased productivity.. 2. Team members can try pair programming where two individuals can work together at one workstation and one member codes while the other reviews, switching up the roles frequently. 3. Team members can volunteer on some tasks, rather than all of them being assigned immediately - During the first sprint, when all the user stories and tasks are 4. Inform the group if any team members complete their task. This can prevent any confusion whether the task is done by the team member. 5. Skill based task allocation. This can ensure that the task can be specialised by the person who is more adept at it. 6. Having more organised meetings with specific agendas like having a specific agenda ensures that all discussion points are aligned with the project and team goals aiding in effective strategy execution.
What questions do	1. How can we make stand-up meetings more efficient and productive? This is

we have?	because stand-up meetings can sometimes be time consuming and the points mentioned during the meeting might get repetitive. Therefore, it would be beneficial to the team if our stand-up meetings are quick, efficient and productive. 2. How can we improve our engagement in our group communication? This is because engagement in group communication is more significant to strengthen the team bonding and determine whether the Sprint goes smoothly. 3. How can we improve our code review process to catch bugs earlier and reduce the time spent on debugging? 4. What are our goals for the next sprint? This is because setting goals can provide a clear direction for the team, together with being able to appropriately prioritise user stories, properly allocate tasks and to encourage a more proactive approach. 5. How can we ask more questions during meetings? This helps to clarify points that might be vague or complicated to certain team members and ensure that everyone is on the same page. 6. How do we allocate tasks to each team member more fairly? This is because, there are some instances where team members decided to do the tasks without informing the group, this results in the originally assigned team member to not have the chance to do the work.