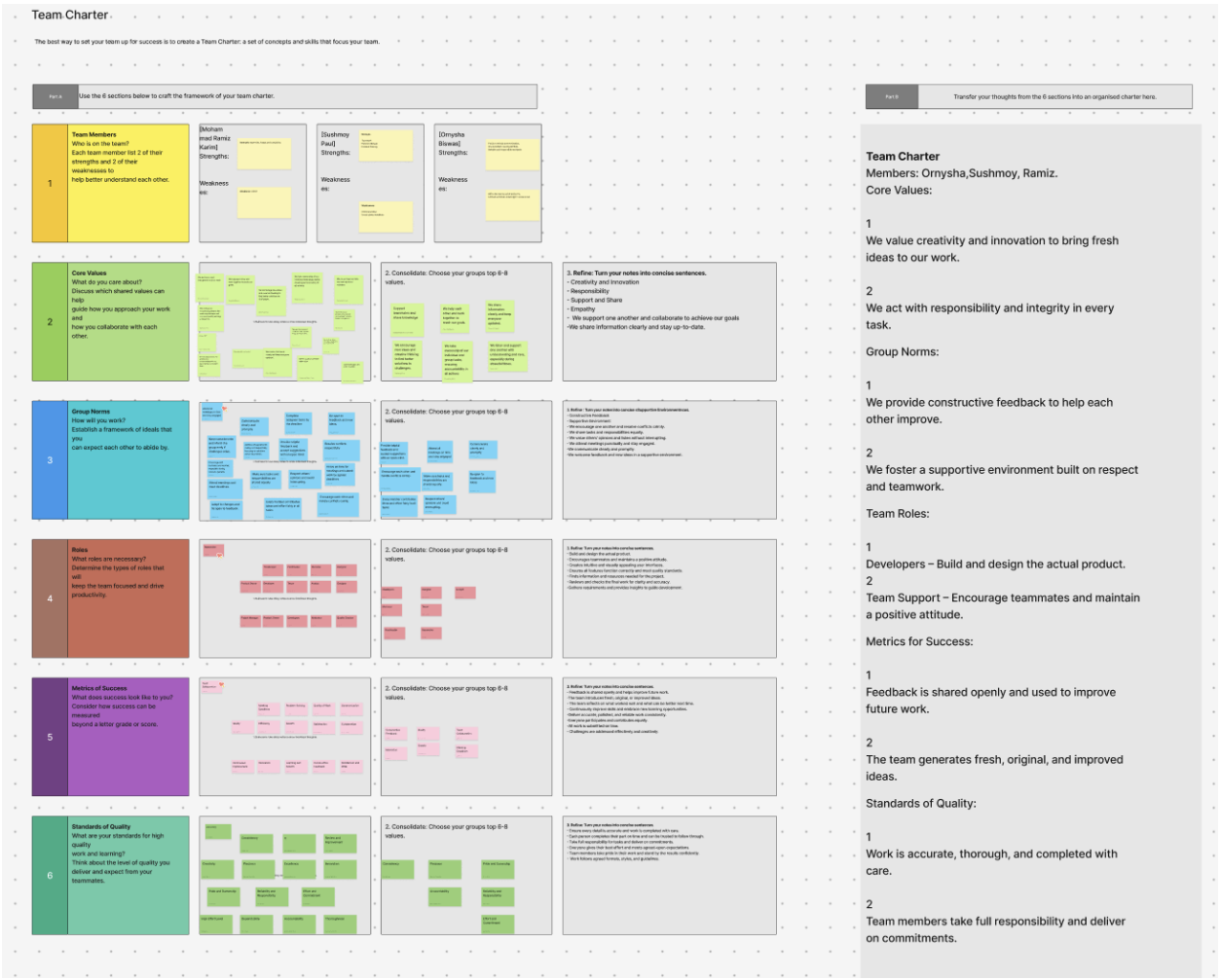
Townsville Team Six 2025 Scrum Project Report

Team member's name and emails:

Name	Email Address
Ornysha Biswas	ornysha.biswas@my.jcu.edu.au
Sushmoy Paul Rohit	sushmoypaul.rohit@my.jcu.edu.au
Mohammad Ramiz Karim	mohammadramiz.karim@my.jcu.edu.au

Section 1: Team Charter Page

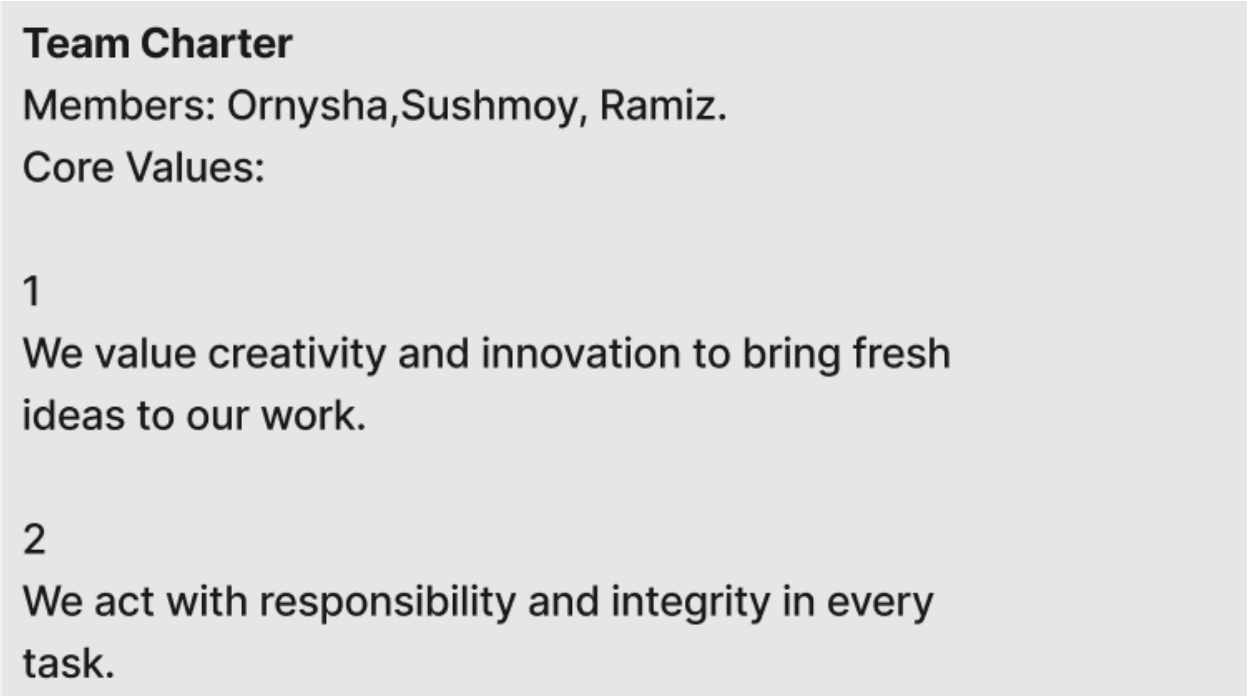


Team Charter - Improvements Made:

We revised our team charter multiple times during the preparation and sprint weeks. The final version, including all the updates and improvements, can be found on our “Team Charter FigJam Board” in Figma.

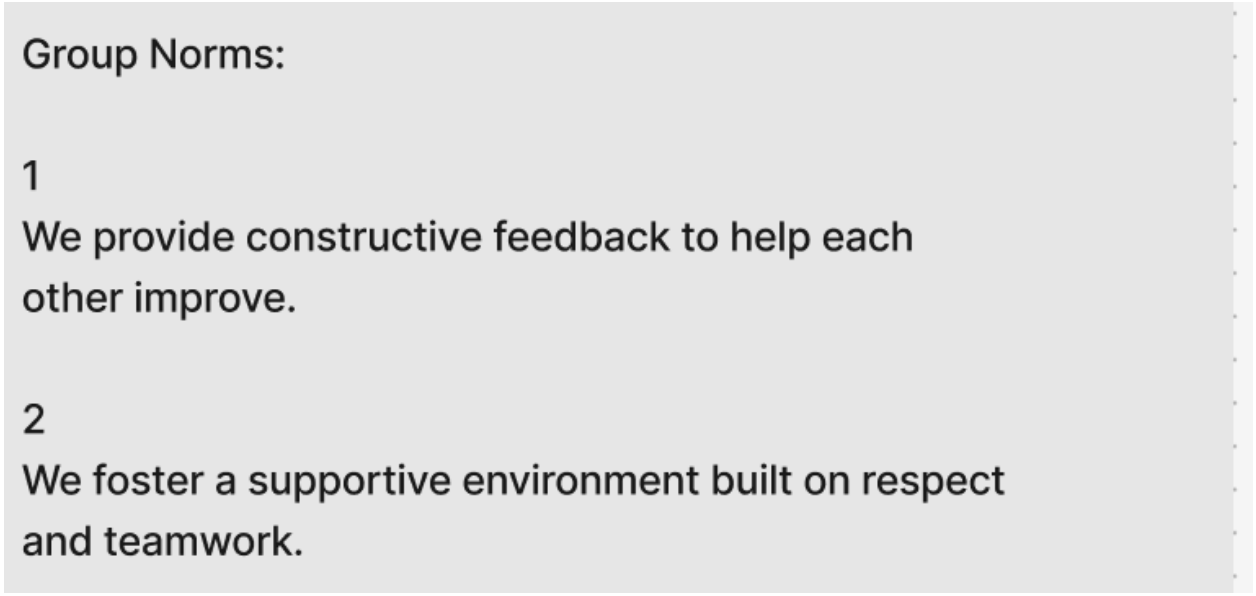
Core Values:

Our core values highlight how we work together as a team. We focus on being creative and bringing new ideas into our project, while also taking responsibility for our tasks and acting with honesty and integrity. These values guide the way we collaborate, make decisions, and support each other throughout the sprint.



Group Norms:

Our group norms focus on creating a positive and supportive team environment. We make sure to give each other helpful and constructive feedback so everyone can grow and improve. We also prioritise respect and teamwork, making sure every member feels supported and valued while working together toward our shared goals.



Team Roles:

Our team roles are divided to keep our work organised and effective. The developers focus on building and designing the actual product, turning ideas into real features. Meanwhile, the team support role ensures everyone stays motivated by encouraging teammates and helping maintain a positive and collaborative atmosphere. Together, these roles help our team work smoothly and stay productive.

Team Roles:

1
Developers – Build and design the actual product.
2
Team Support – Encourage teammates and maintain a positive attitude.

Metrics For Success:

Our team measures success by how well we learn and improve together. We make sure feedback is shared openly and used to strengthen future work. We also aim to consistently produce fresh, original, and improved ideas that show our growth and teamwork throughout the project.

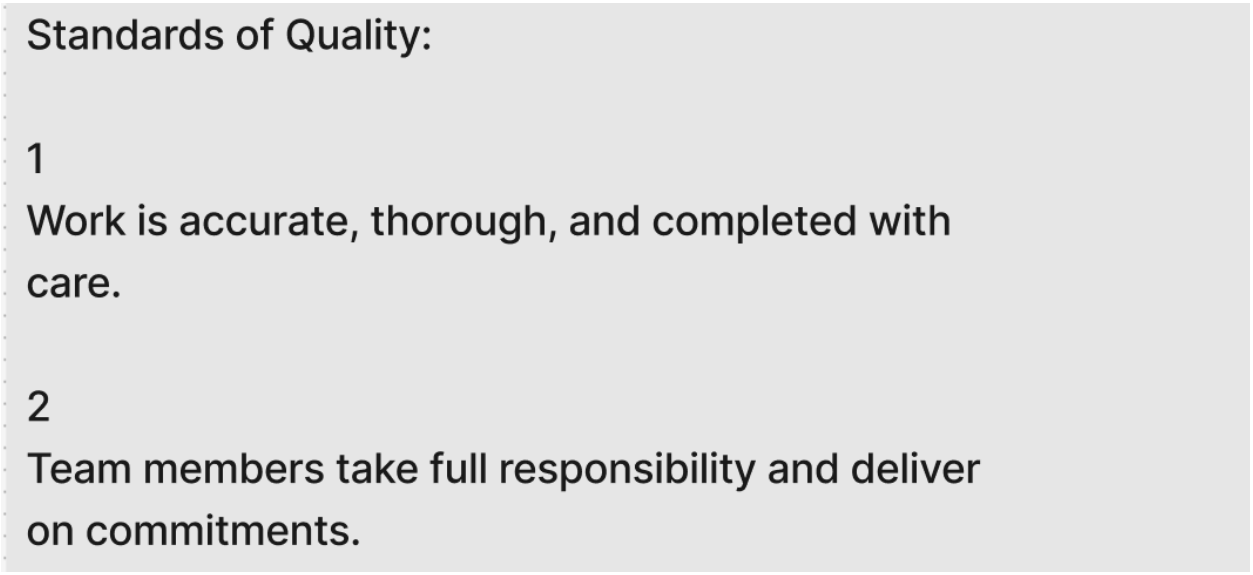
Metrics for Success:

1
Feedback is shared openly and used to improve future work.

2
The team generates fresh, original, and improved ideas.

Standards Of Quality:

Our team is committed to producing work that is accurate, thorough, and completed with care. We ensure that every task meets a high standard by double-checking details and paying attention to quality. Each team member takes full responsibility for their contributions and reliably follows through on their commitments, helping maintain consistency and professionalism in our work.



Section 2:

1. Restate the project challenge

Goal:

To create a simple, fast, and effective method that helps business employees recognise, verify, and report suspicious emails or messages so they can avoid cyber-attacks during their everyday work.

Design Challenge:

Our design challenge focuses on improving workplace security by building a system that guides employees at the exact moment they face a suspicious request.Many employees still fall for fake emails, urgent payment scams, or unsafe links because current security training is too long, boring, and not relevant to their real job tasks.They often rely on guesswork, feel unsure about how to verify information safely, or don't know the correct way to report something suspicious.

The challenge is to design clear tools such as role-based scenarios, a simple pause–Verify–Proceed checklist, and just-in-time prompts (e.g., "New sender detected") that help people act safely in under 60 seconds while doing their normal work.

Targeted Goal:

To build a behaviour-support system that helps employees safely handle risky messages and reduce cyber-security incidents.
The solution should allow employees to:

- Quickly recognise high-risk situations
- Follow a simple, fast verification process
- Use clear prompts to guide decision-making
- Report suspicious messages easily
- Feel confident, not stressed, when something unusual appears
- Improve overall safety without slowing down their workflow

This aims to make workplace security faster, clearer, and more reliable, leading to fewer risky clicks, better reporting, and higher confidence across the organisation.

2. Pros (and why they are pros)

1. User Research

The team spoke directly with employees, helping them understand real problems.
Why it's a pro: It ensures the solution is based on real needs, making it more accurate and useful.

2. Engaging Training

Departments received relatable examples that made learning interesting.
Why it's a pro: Participants stay engaged and learn better when examples match their real work.

3. Supportive Culture

People felt safe to report issues without fear.
Why it's a pro: A safe environment encourages more reporting and improves overall security awareness.

4. Safer Decisions

Participants correctly identified fake messages during prompts.
Why it's a pro: Shows the system effectively improves decision-making in real situations.

5. Encourages Engagement

Short micro-lessons kept users active instead of bored.
Why it's a pro: Higher engagement increases training success.

6. Measurable Success

The system tracks clicks, reports, and outcomes.
Why it's a pro: Measurable data allows the team to evaluate effectiveness.

7. Simple & Quick

Lessons are bite-sized and easy to engage with.
Why it's a pro: Less time-consuming, encouraging more consistent participation.

8. Easy to Expand

The design can be quickly scaled or updated.
Why it's a pro: Future improvements and updates become easier to implement.

9. Improves Awareness

Helps employees identify scams faster.
Why it's a pro: Directly strengthens security within the organisation.

10. Real-World Skills

Boosts user confidence in handling real-life scam situations.
Why it's a pro: Reduces mistakes during real security threats.

Cons (and why they are cons)

1. Only Focuses on Awareness

Training mostly relies on users detecting threats manually.
Why it's a con: Without automation, users may still miss complex scams.

2. No AI Automation

The system lacks automatic scam detection or smart filtering.
Why it's a con: It creates more work for employees and reduces accuracy in fast-moving situations.

3. Depends on Voluntary Use

Users must remember to check prompts under pressure.
Why it's a con: Human error or stress may cause users to skip important actions.

4. Doesn't Cover All Scenarios

Some attacks, like deepfakes or voice scams, aren't included.
Why it's a con: Leaves gaps in training, making employees vulnerable in those areas.

5. Low Confidence

Some staff members hesitate to trust suspicious messages.
Why it's a con: Uncertainty slows decision-making and may increase errors.

6. Old Habits

Even after training, users still fall back into bad patterns.
Why it's a con: Training may not be strong enough to create long-term behaviour change.

7. Missed Details

Users sometimes miss small clues in examples.
Why it's a con: Mistakes can still occur even with training, reducing effectiveness.

8. Workflow Disruption

Pause prompts interrupt daily tasks.
Why it's a con: Training becomes annoying during busy work hours.

9. Setup Time Too Long

It requires effort to design and maintain the training prompts.
Why it's a con: Increases workload for the IT/security team.

10. User Neglect Over Time

Users may start ignoring prompts.
Why it's a con: Engagement drops and reduces the overall impact.

Section 3:

Summary of Project Direction :

From the pros and cons we discussed, our team decided to build a simple phishing-awareness training system. The cons showed that employees often miss warning signs, forget steps, and get confused. But the pros showed that short lessons, quizzes, and tracking tools can help people learn faster.

So our direction is to make:

- a short phishing lesson,
- a small quiz to test understanding, and
- a simple dashboard for managers to see who completed the training.

This is realistic for our skills and our time, and it still solves an important problem.

Product Goal:

"Create a basic phishing-training prototype that teaches employees quickly and lets managers track completion."

Initial Product Backlog:

Our initial product backlog focused on tasks that we were unsure about or needed more research. First, we needed to check if Google Forms and Google Sheets were suitable for creating the quiz and tracking results. We also had to test which lesson format would work best, such as slides, a simple webpage, or a mobile layout. Another important backlog item was understanding privacy rules, because we were unsure whether to store employee names, IDs, or anonymous responses. We also needed to test how to make the

lesson work well on mobile devices. Building a basic dashboard for managers was another challenge, as we had to figure out how to show both completed and not-completed users. Lastly, we needed to explore what quiz questions would effectively test phishing awareness. These items helped guide the direction of our project and set up our Sprint planning.

Section 4:

Sprint Goal

To design and test an online phishing-awareness lesson and quiz system that employees can complete and managers can track.

Sprint Backlog

User story ID	Description in simple words	Tasks / Sub-tasks	Estimate (S/M/L)
US 1	Employee – Learn from short online lesson	Research short training formats (slides or videos) Design simple page layout Upload sample content and test view	S (1 day)M (2 days)M (2 days)
US 2	Security Officer – Create and send phishing quiz	Write 3–5 quiz questions Build email or webpage for quiz· Collect and store responses	S (1 day)M (2 days)M (2 days)
US 3	Manager – Track employee completion	Design dashboard mock-up Add completion log Test display of data	M (2 days)M (2 days)S (1 day)

Sprint Planning Event

During the Sprint Planning Event, our team (Scrum Master: Ornysha, Proxy Product Owner: Ramiz, Developer: Sushmoy) met to decide what we would complete in this sprint. We discussed the Product Goal again and selected the most important user stories that match the direction of our project. The Proxy Product Owner helped prioritise the stories, and the Scrum Master organised the tasks and made sure everyone understood the plan. Together, we agreed on the Sprint Goal, the backlog items, and the tasks needed to complete them. We also added estimates so we know how much work we can finish.

Research / Design Spikes

1. Tools for Lesson and Quiz:
The team compared Google Forms, Moodle, and an internal web app.
• Google Forms was chosen first because it's quick to set up, free, and easy to share through email or Teams.
• Real-life idea: In real offices, companies often start training through quick tools like Microsoft Forms or Kahoot before moving to bigger systems like Moodle or SAP SuccessFactors.

2. Privacy and Data Safety:
The team discussed how to handle quiz results safely.
• Only basic details (name or staff ID) will be collected.
• Results stored in a secure Google Workspace sheet with access for HR and the security officer only.
• Real-life idea: Similar to how real companies use company email logins to access training so data stays within the organisation's system.

3. Design Review:
The layout was tested for clear text and mobile use.
The final design is simple and easy for staff to finish in under 5 minutes.

Definition of Done (DoD)

Lesson and quiz work correctly and can be completed easily.

Results are visible to the manager or HR through the linked sheet.

Work has been tested and reviewed by the team.

Proxy Product Owner has approved it for the Sprint Review.

Sprint Backlog

Sprint Goal

To create a working phishing-awareness lesson, a short quiz, and a simple dashboard that shows who completed the training.

Selected and Refined Backlog Items:

For this sprint, our selected and refined backlog items focused on creating a complete phishing-awareness training prototype. We chose six small user stories that together support the Sprint Goal. These included letting employees open and view the phishing lesson, making sure the lesson works properly on mobile, allowing users to complete a short quiz, enabling the quiz to auto-submit and record results, and giving managers the ability to see both completed and not-completed users. Each of these user stories was broken into simple tasks such as designing the lesson layout, writing quiz questions, linking the quiz to Google Sheets, adjusting mobile formatting, and building a basic dashboard. These refined stories were selected because they directly match the Product Owner's priorities and are realistic for the team to complete within a single sprint.

Estimation Method

We used **S, M, L** (Small, Medium, Large) because this method is simple and easy to understand for a small team.

- **S (Small)** = less than 1 day
- **M (Medium)** = 1–2 days
- **L (Large)** = more than 2 days (we had none)

This helped us plan the sprint without needing complex story point systems.

Why These Items Were Chosen

We chose these items because they match the Product Goal and because the Product Owner said the team must focus on the lesson, quiz, and manager tracking features. These were also the highest-priority items based on the pros and cons from the earlier section. The items are small enough for one sprint and help us create a complete working prototype.

Section 5:

Daily Scrum Summary

Sprint Day 1 – Key Activities & Issues

What Happened

- All members reported what they completed the previous day and what they planned to work on today.
- Team members focused on setting up layouts, researching tools (e.g., Google Forms, Moodle), and beginning quiz/lesson design.
- Several early blockers appeared, such as uncertainty about hosting tools, quiz result storage, and understanding where data would be kept.

Pivotal Issues Identified

Uncertainty about tools/platform for hosting lessons
Lack of clarity on quiz result storage and security

How It Changed the Sprint Backlog

- **New research tasks** were added to *To-Do* for Day 2 to compare hosting platforms and data security options.

- Some tasks originally planned for later (e.g., designing quiz layouts) were **delayed** to prioritise research.

- Backlog items were reorganised to ensure technical feasibility was confirmed before building.

Why These Changes Were Necessary

- The team needed a **stable technical direction** before designing lessons or quizzes.

- Without confirming lesson hosting and result storage, later work could have been wasted.

- Prioritising research ensured the future prototype would be functional and secure.

Sprint Day 2 – Key Activities & Issues

What Happened

- Team members progressed on quiz questions, design layouts, and manager dashboard ideas.

- Updated Sprint Backlog items reflected movement from "To-Do" to "In Progress."

- More technical blockers appeared, especially around **email automation** and **dummy progress data**.

Pivotal Issues Identified

Trouble formatting quiz submission and automation
Uncertainty about back-end logic for manager dashboards

How It Changed the Sprint Backlog

- New backlog items were added for:
    - Testing automated email reminders
    - Creating sample data for dashboards
    - Fixing formatting issues before finalising content
- Some design-based tasks moved to Day 3 to allow backend logic to catch up.

Why These Changes Were Necessary

- The dashboard and auto-submit features depended on correct data formatting.
- Without automation, the prototype would not meet the Sprint Goal requirements.
- Fixing these issues early prevented broken functionality later.

Sprint Day 3 – Key Activities & Issues

What Happened

- Team members began finalising quiz formatting, adding finishing touches to dashboards, and preparing lesson screens.
- Yesterday's blockers mostly shifted toward **small formatting and timing issues**, rather than major technical uncertainties.
- Final reviewing and refining tasks dominated the day.

Pivotal Issues Identified

Small errors in formatting and UI consistency
Some delay in completing lesson content and quiz structure

How It Changed the Sprint Backlog

- Minor polishing tasks were added to *To-Do* for the final day.
- Some items moved into *Done* earlier to allow the team to focus on remaining issues.
- “Ready for Sprint Review” tasks were updated to ensure all user stories matched the Definition of Done.

Why These Changes Were Necessary

- The team needed a polished prototype for Sprint Review.
- Ensuring UI consistency and finishing incomplete content kept the project aligned with the Sprint Goal.
- Fixing formatting issues early prevented larger presentation problems during review.

Sprint Day 4

On the final day, the team focused on polishing the prototype and preparing for the Sprint Review. Team members shared updates on content proofing, final formatting, dashboard improvements, and ensuring the auto-marking features worked correctly. The main blockers were:

- Finalising dashboard visuals in time
- Small errors in quiz logic
- Ensuring all screens met the Definition of Done

These issues led to last-minute adjustments in the Sprint Backlog, such as **adding quick-fix tasks**, including proofreading, layout corrections, and final testing tasks.

These changes were necessary to ensure everything met the Sprint Goal: delivering a complete phishing-training prototype with a lesson, a quiz, and a manager dashboard. Without these final adjustments, the product would not have satisfied the DOD and would have lacked polish for stakeholder demonstration.

Overall Summary

Across the three Daily Scrums:

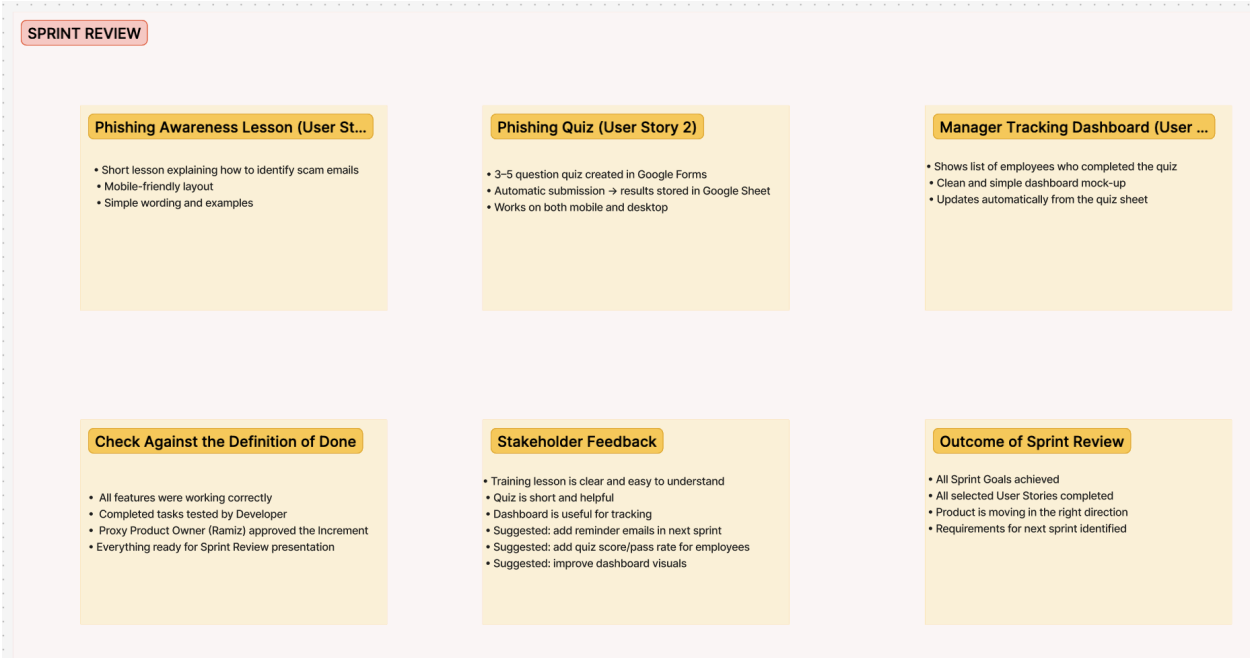
- The team continually **refined priorities**, ensuring work aligned with the Sprint Goal.
- Issues around **technical uncertainty, data handling, and formatting** caused multiple Sprint Backlog updates.
- These changes were essential to:
    - Maintain project feasibility
    - Ensure the prototype worked
    - Present a polished and functional product
    - Fulfil all user stories and DoD requirements

Section 6: Sprint Review and Sprint Retrospective

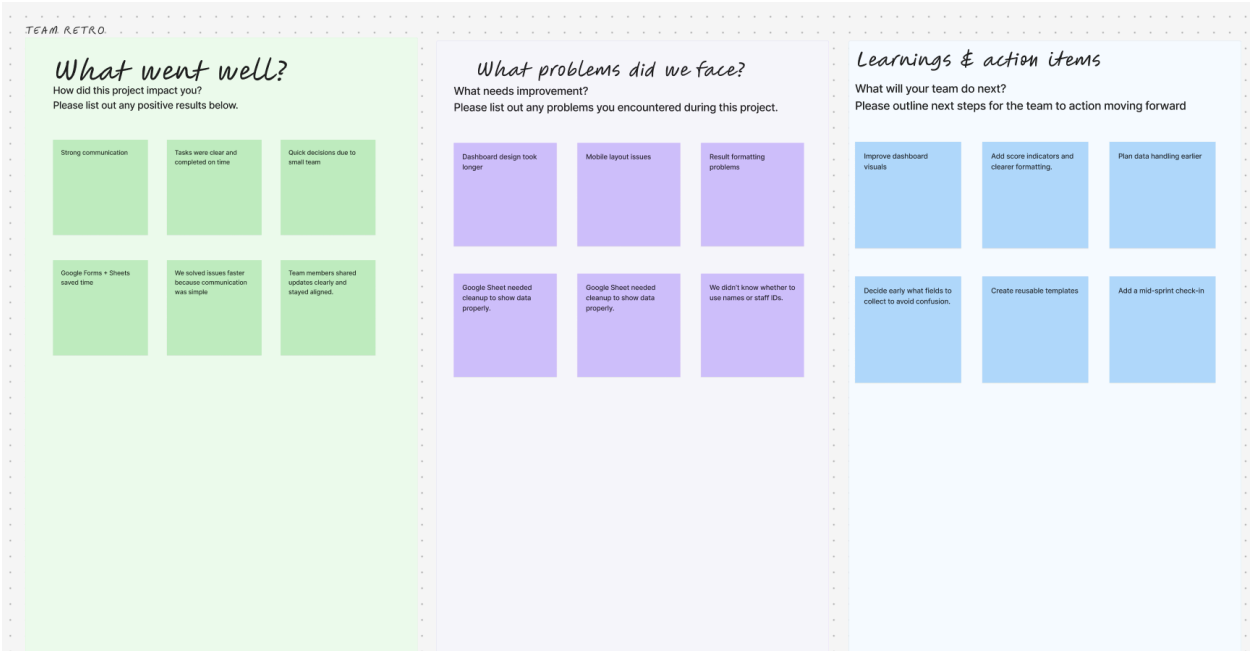
Sprint Review Summary

During the Sprint Review, our team demonstrated the phishing lesson, the quiz, and the manager dashboard we completed in the sprint. All parts worked as expected and met

the Sprint Goal. The Product Owner was happy with the progress and said the prototype was clear and easy to follow. He also suggested improving the dashboard design, adding quiz scores, and including reminder emails in the future so managers can follow up more easily. Overall, the Increment was accepted and seen as a good direction for the project.



Sprint Retrospective Summary



In the Retrospective, we discussed what went well and what could be improved. We agreed that communication was strong and that using simple tools like Google Forms and Google Sheets helped us finish tasks on time. However, we faced some problems, such as dashboard formatting taking longer than expected, mobile layout issues, and early confusion about what data to collect for privacy. These challenges helped us see where we can improve.

Improvements for Next Sprint

For the next sprint, the team wants to plan data handling earlier, improve the dashboard visuals, and create simple templates to speed up design work. We also decided to add a small mid-sprint check-in so we can solve blockers faster. These improvements should help us work more smoothly and deliver a better prototype in the next sprint.

Section 7: Reflection on scrum, Technical Feasibility and Team collaboration

Based on the outcome of our pilot study, our team believes that **Scrum was an effective and appropriate choice** for managing this project. The nature of our work—building a prototype that required constant feedback, iteration, and refinement—benefited greatly from Scrum's emphasis on short cycles and continuous improvement. The Daily Scrum meetings helped us track progress closely, identify blockers early, and adjust our Sprint Backlog according to evolving needs. This flexibility was essential because key challenges, such as choosing the right tools, managing data verification, and finalising UI layouts, emerged at different stages of development. Without Scrum's iterative structure, it would have been difficult to stay aligned with our Sprint Goal or respond quickly to technical issues.

In terms of technical feasibility, our team believes the project is achievable with the right tools and planning. During the Sprint, we successfully created lesson screens, quizzes, automated elements, and dashboard concepts using accessible platforms such as Google Forms, Figma, and simple automation logic. Although we encountered early uncertainties—especially around data handling and automation—our research showed that these challenges could be addressed with existing technologies such as APIs, email automation tools, and learning management platforms. The successful completion of our working prototype indicates that the project can be expanded into a fully functional system with moderate development effort. Overall, the technical components required are realistic, and no blockers appeared that would prevent the system from becoming a complete product.

Regarding **team collaboration**, we believe our group worked together very effectively throughout the Sprint. Each team member consistently contributed tasks, shared ideas, and supported others when blockers appeared. Our team charter helped guide our behaviour, ensuring clear communication, mutual respect, and responsibility. During the Daily Scrums, everyone openly discussed their progress and obstacles, allowing the team to redistribute tasks when needed and maintain momentum. Each person completed their assigned work while also helping others when challenges arose. By the end of the Sprint, we met our Sprint Goal, demonstrated a complete prototype, and reflected on improvements through a structured team retrospective. This shows that our collaboration was strong, well-coordinated, and productive.

Overall, the pilot study demonstrated that Scrum fits the project well, the technical requirements are achievable, and the team functioned efficiently and cohesively throughout the development process.