LEARNING MANAGEMENT SYSTEM

GROUP 5

Esha	**Jijendran**	**Dhakshini**
Scrum Master	*Product Owner*	*Quality Analyst*
Yuhan	**Hui Yi**	**Ee Dhing**
Risk Manager	*User Experience Specialist*	*Minutes Taker*

Agile Iteration II-Team 5

Sprint 2 - Project Vision

According to the structure that we developed during Sprint 1, our LMS is moving towards a complete learning system which alters the way teachers and learners collaborate in structured learning spaces. Better course structure, classroom tools and better user experience are added as part of the better course structure in this second version.

Improved Platform Architecture.

Sprint 2 also extends our LMS above and beyond the mere organization of lessons and introduces a complete learning hierarchy. Courses are now the primary container that stores the numerous lessons and each lesson could have a classroom in which to live teach. This design is based on the way teachers do their work and it provides the teachers with the means of teaching the lessons sequentially.

Extensive Course and Classroom Management.

Teachers have complete control of what they teach using our platform. They are able to create entire courses, prepare elaborate lessons and schedule classrooms that commence and finish at a given date. We included pre-requisite rules whereby students have to complete previous lessons before proceeding to more difficult lessons.

The lesson building screen has large text boxes, allowing teachers to type longer descriptions and objectives. The lesson list is also more readable, which gives the teachers easy time managing the lesson list, and adding content without much difficulty.

Enhanced User Interface and browsing.

Sprint 2 is aimed at the role-specific and user-friendly design. Teachers would be able to access lesson plans and student data quickly because when they log in, they can immediately see their courses. The students access the courses they are taking directly and hence getting what they need is easy.

Our color palette is plain black, blue and white color, which appears professional all over. It is also user- friendly even to individuals with varying technology skills.

Student and Teacher Advanced Interactions.

It is now possible to view all that teachers need in a single place. They can view live lists of students to be able to manage class well, and publish courses to enable students to get them at the appropriate time. pupils possess their page of lessons, on which they can follow their course and access the materials without any difficulties.

Infrastructure Improvements on the technical side.

We have added a common database to make sure that data is used in the same way across devices and users. This prevents the issue of data stagnating in a single location and retaining the experience at any point. We also prepared complete test reports to be able to know that the system is working and complies with all the rules.

Principles of Future Development.

After the expiry of Sprint 2, our LMS ceases to be a lesson manager but an entire educational system that represents all the teaching and learning levels.

In the following sprint, we would be focussing on the finishing features of the software to make it full fledged. This includes but is not limited to hosting, admin features, grading features for students and more.

1. Sprint 2 - Process Model

In our team, the second academic sprint lasted 2 weeks where we continued with Scrum to provide the second significant release of our LMS during Sprint 2. Based on Sprint 1, Product Owner Jijendran refined the prioritisation of the backlog and established a clear acceptance rule that enhanced the management of courses and classrooms. Scrum Master Esha Veena, chaired all sprint meetings and addressed blockers that occurred. The user stories that we had promised were fulfilled by the developers team (Dhakshini, Esha, Jijendran, Yuhan, Hui Yi, Ee Dhing) with the same frontend (Dhakshini, Hui Yi) and backend (Ee Dhing, Yuhan) roles.

In Sprint Planning, the Product Owner presented eighteen user stories with the highest priority; they included making courses, classroom management, improving the UI, as well as enhancing system integration. These tales contained important elements such as instructor controlling courses, learner watching courses, prerequisite classroom scheduling, less complicated lesson development, user registration, and collective database. Planning Poker was applied in estimating the stories, and we reviewed how much work we could absorb what we learned in Sprint 1 and we pledged to accomplish the tasks that were within our objective of full course and classroom system.

We subdivided every user story into technical work: updating UI/UX, optimizing the database structure, creating APIs, creating frontend components, testing integrations, and conducting acceptance tests. We assigned the right individuals to the task and approximated the number of hours and hence each was allocated a reasonable amount of work both on frontend and backend.

We continued to have one stand-up per week, during which we exchanged progress, blockers and plans per day. We also had formal Sprint Planning and Retrospective meetings therefore we had Scrum rituals. The Scrum Master monitored blockers and assisted in their resolution so that the team did not lose its pace.

During the Sprint Review, we demonstrated to the Product Owner the working product, namely the new course structure, classroom tools, improved UI, and database features. The Product Owner accepted all the stories that fulfilled the rules of acceptance and provided feedback on the subsequent versions.

The Sprint Retrospective revealed the large-scale progress over Sprint 1: the coordination of tasks increased, we were more adherent to Git rules and the teams were able to communicate better.

Planning poker and estimation

Technique Used:

Planning Poker was applied in Jira and was used to estimate story points of every backlog item. The members of each team were assigned their own deck of value story points and chosen a card individually. We scaled by the Fibonacci sequence (1, 2, 3, 5, 8, 13, 21, 34, 55, 89) to scale stories (refer to appendix Figure 1.4) since:

- The non-linear increase in story effort with increase in complexity is reflected in Fibonacci.
- Bigger distances between bigger figures facilitate greater emphasis on big/uncertain stories.
- It promotes segmentation of work once the numbers begin to soar too high.

How We Conducted It

1. The Product Owner introduced each user story with its acceptance criteria.
2. Team members asked clarifying questions.
3. Everyone dropped their estimated story points for each User story simultaneously in Zoom.
4. If estimates matched: that became the story point.
5. If estimates differed:
    - We invited the highest and lowest estimators to explain their reasoning.
    - We discussed assumptions, dependencies and complexity.
    - We re-voted until a consensus was reached.

Handling Conflicts / Variations

- When the team couldn’t reach agreement quickly, the Scrum Master facilitated and ensured everyone had a chance to speak.
- This process improved our shared understanding and made our estimates more accurate over time.

Comments & Review

After estimation, the Scrum Master summarised the agreed points in Jira. The team then reviewed the final sprint backlog to ensure the total story points aligned with our capacity.During the sprint review, we reflected on whether our estimates were realistic and noted lessons learned for future sprints.

sprint_planning.mp4

Produced in Sprint 2

Refined Product Backlog – PO (Jijendran) reprioritised stories for course, lesson and classroom management based on Sprint 1 feedback; new prerequisites and publishing features added.

Sprint Backlog – chosen Sprint 2 stories divided into tasks, with clear owners across frontend (Dhakshini, Hui Yi) and backend (Ee Dhing, Yuhan).

Definition of Done (DoD): Merged to master, feature tested, hand-verified against updated acceptance criteria, peer-reviewed, work-log updated in Jira.

Sprint 2 Objective: Deliver the second release of the LMS by enabling instructors to create/publish courses with lessons/classrooms, manage prerequisites, and allow students to view and navigate their enrolled lessons.

Product / Sprint Backlog (DEEP + INVEST)

Our Product Backlog in Sprint 2 continued to follow DEEP:

Detailed appropriately – high-priority items (classroom creation, prerequisites, lesson list UI) were fully specified with acceptance criteria; lower-priority items (theme/UX improvements) less detailed.

Estimated – top stories estimated in story points during Sprint Planning using Planning Poker.

Emergent – backlog evolved as new requirements emerged mid-sprint.

Prioritised/Ordered – PO kept Sprint 2 stories at the top for focus.

Our Sprint Backlog followed INVEST:

Independent – course creation, prerequisites and student-view features implemented independently.

Negotiable – requirements clarified with the PO during Sprint Planning.

Valuable – each story provided visible LMS improvements for instructors or students.

Estimable – all stories measured in Fibonacci story points.

Small – stories sized to complete within a single sprint; large stories split where needed.

Testable – acceptance criteria defined test cases for each story.

Burndown Chart

As seen in our burndown chart, the work seems to have been stagnant for the first few days and then has a gradual decrease after that, resulting in the chart's graph being above the guideline in grey. This is because, in the first couple of days, the team was clarifying requirements and working on filling up the knowledge gap as evidenced in our developer user stories that show our learning spikes. We spent this time working on deciding and negotiating the scope of this sprint and understanding the feedback to apply into this sprint.

However, as shown in the chart, our team has had a steady decrease in story points after the initial plateau, evidencing our improvement from the last sprint, showing consistent work throughout the sprint, which was one of our goals addressed in our last sprint retrospective.

Total User Story Points:192
User Story points completed:192
User Story points Incompleted: 0

2. Project Management

2.1 Standup Meetings

Meeting Details

- **Sprint week:** 2
- **Team:** Group 5

2.1.1 Scrum standup 1

- **Source:** stand_up_meeting1.mp4
- **Focus:** Every team member shared their progress on the sprint 2 tasks, mainly working on the corrections the client gave for learning management system.
- **Highlights:**
    - User stories have been rearranged as well as the backlog.
    - Implement Supabase database to link to all the information in this learning management system.
    - Most developers have started the instructor's user stories.
- **Meeting minutes:** refer to Appendix 1.1

2.1.2 Scrum standup 2

- **Source:** stand_up_meeting2.mp4
- **Focus:** Application of sprint tasks and cross-team collaboration
- **Highlights:**
    - Done most of the user stories for the course and classroom page creations.
    - Planning to continue working on the classroom and course page details and modify the current create lesson page.
- **Meeting minutes:** refer to Appendix 1.2

2.2 Sprint Planning

2.2.1 Sprint Planning Session

- **Source:** sprint_planning.mp4
- **Focus:** The team conducted a sprint planning session to finalize backlog items for the upcoming sprint.
- **Highlights:**
    - Reviewed the sprint backlog and prioritized the user stories.
    - Allocated tasks among team members.
    - Allocate the story points for each user story.
- **Meeting minutes:** Refer to Appendix 1.3

2.5 Sprint Retrospective

The work log can be found in Jira through Clockify by clicking the individual user stories and the tasks under them.

The following google docs is the file with the answers for every team member regarding what went well, what could have been done better, what will we try next and what questions do we have.

Retrospective.mp4

3. Risk Register And Management

3.1 Risk Matrix and Live Risk Register

As part of the project inception, a comprehensive Risk Register and Management plan has been established. This includes the development of a Risk Matrix to assess the likelihood and impact of potential risks, as well as a Live Risk Register to record, monitor, and update risks throughout the project lifecycle. These tools ensure that risks are identified early, managed proactively, and regularly reviewed to support effective decision-making and project success.

3.2 Resolution Notes (Sprint 2)

R4 (Dependency on another task or team member that has fallen behind):
This risk was identified on 21 Sept 2025. When most of the tasks aren't done and the user stories are all still in to do. This risk is solved by reallocating effort to unblock critical dependencies and communicate the project impact and we also ensure communication between members.

R7 (Version control conflicts in Git):
This risk was identified on 24 Sept 2025. There are conflicts in the git merge requests when Hui Yi tried to merge her work on the frontend. We resolve this conflict by using code review sessions and working on the code collaboratively with good communication.

R9 (Technical skill gaps within the team):
This risk was identified on 20 Sept 2025. When some of the team members are unable to code really critical tasks due to inexperience of the language for the codes. The risk is resolved by reassigning critical tasks to experienced members and extending individual deadlines for beginner members.

4. Contribution/Work Log

For this sprint, the team used **Clockify** to record individual contributions, with all members consistently logging their hours. These work logs were linked to the corresponding **user stories in Jira**, providing clear traceability between effort and deliverables. Compared to the last sprint, the team significantly improved in maintaining up-to-date records in Jira, resulting in better visibility of progress and workload distribution.

5. Sprint Retrospective Reflection -Sprint 2

Tan Ee Dhing

In this sprint, I started my tasks and corrections late because I procrastinate. Although I did finish my tasks on time, I was rushing through it because of the upcoming deadline. However, my coding skills have improved a lot since the last sprint because I've been learning how to code JavaScript throughout this sprint also. My git commits, push, and pull is also very consistent and efficient so it's better than the previous sprint. I did my work on time and contributed the back end logic which is JavaScript to the classroom page, course page and also register page. I am also more familiar with Jira and how to use clockify to clock my work as well as how to set to do, in progress, in review, and done in Jira.

Ooi Hui Yi

For this sprint, I immediately started the correction from the sprint 1(increase the text box sizes for the lesson description and objective as well as changing the layout of the lesson page, moving the create lesson button to the top right corner under the header), which reduces some work loads for sprint 2. However, it is a bit more challenging to complete all the user stories within one week. But we are still able to complete it all as all of us have been working on them together. Compared to the previous sprint, I'm more familiar with html and css as well as using the git (pushing, pulling and commit) which allows me to finish the tasks on time also. Overall, I feel a bit more confident in coding with html and css as compared to before.

Chen Yuhan

During this sprint, I completed the workload assigned to me, mainly focusing on back-end development related to creating courses and lessons, as well as the navigation between them. Besides that, I was introduced to Supabase, learned the basic setup and how to connect it with the html and js code. Compared with the previous sprint, this sprint was more structured, which is clearly reflected in our burndown chart. It also indicated that our team members worked collaboratively and aligned toward the same goal.

However, there are still some areas that need to be improved and avoided in the future. I should start reviewing user stories early, ensuring I fully understand what needs to be completed within the current sprint, and if there are any problems, I should communicate with team members and the client promptly, providing any delays due to unclear information. Overall, this sprint has strengthened my understanding of software management and gave me greater confidence to perform better in the next sprint.

Dhakshini Subramanian

For this sprint 2, The burntdown chart has really improvised from the previous sprint 1. All the coding tasks i.e. frontend and backend were started right away once sprint 1 was completed.The git commits,branch push and pull were all much used to. Also I have strongly committed to the front - end development areas. HTML and CSS languages were more familiar to work with this sprint 2. And as a Quality tester I have taken in the feedback from sprint 1 and implemented feature testing to fulfil testing scopes. So by implementing testing in sprint 2 our DOD has also attained its expectation.In further sprints I will try to improvise more on my coding skills and meet the standards.With relevance to project management tools I am confident in navigating through different tools offered in jira and clockify..Through these I have recorded the time I have worked on for the tasks/user stories assigned to me.With the previous feedback from our sprint 1 model, we managed to improvise how we did the git commit messages.When working as a team it helped us what particular change or block of code the developers made

Jijendran

I feel like as a PO this sprint went better than the previous sprint since we were all very familiar with what to do and how to do it. The only thing we struggled a bit on was the communication aspect since we had a lot of user stories and the group was almost always active and full of messages. Due to this it is possible that some work assigned would have been missed and thus work would have slowed down. We will work to address this issue in the next sprint. We improved a lot with the usage of git compared to last time making it easier to organise code and know what one person is doing at any given time. This also made adding new features really easy and effective letting us complete a huge amount of work in a span of 2 weeks.

As a developer I only struggled with the part of integrating the new database we use “Supabase” with the existing software as I was unsure on how to store the data and in what data format and once I had that down the coding aspect went well too. But in the aspect of logging my work I would keep forgetting to start the timer when I start my work leading to hours of work being unlogged when I actually did work.

Overall this sprint demonstrated the team’s growing maturity handling the developmental process. The issue of communication will be addressed and fixed for the next sprint. On a personal level I will work to ensure that I log my time appropriately so that the project metrics are accurate

Esha Veena

In this sprint, as a scrum master, I believe everything came naturally to me. This is because I overcame the learning spike of being a scrum master in the last sprint, so things in this sprint seemed more routine. My main takeaways from this sprint are regarding some refinement in our submission, based on what was given as feedback. This includes using Jira in the standup meetings and the proper way to do the sprint review and more. Additionally, I notice that we as a team have improved a lot more in this sprint. We fill in each other's gaps and work effectively to complete the goal, as demonstrated. Our workflow was definitely more smoother this time because everyone was a lot more used to agile processes.

As a developer, using databases in the databases in the backend was something that I did not have real experience with. In this sprint, I learnt how to use Supabase and fetch data and update data to the database. I realized that we as a team need to start our coding process earlier but we found out the root issue was actually our delay in negotiating and finalising the MVP for this sprint.

I believe that in this sprint, the team very visibly improved and learnt from the previous sprint and corrected mistakes in this one. We also learnt from the current sprint and will apply that knowledge to the next sprint. Most importantly, for the next sprint, I want to prioritize starting the sprint earlier and ensuring communication and task delegation is smoother in the team.

6. Final Product Demo

This is the sprint review where our scrum master presents the demo of our learning management system on student,instructor,classrooms,courses and lesson pages in detail.

demo.mp4

Appendix

1.0 Meeting minutes

1.1 Scrum Stand-Up 1 Meeting Minutes

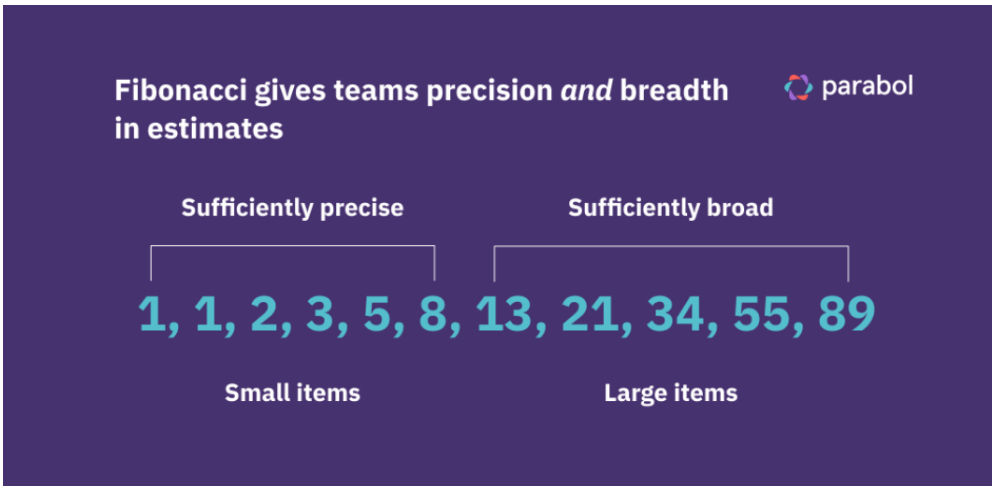
1.2 Scrum Stand-Up 2 Meeting Minutes

1.3 Sprint Planning Meeting Minutes

1.4 Sprint Retrospective Meeting Minutes

1.5 Sprint Review Meeting minutes

2.0 Planning poker done by using fibonacci sequence of numbers



Reference

Google Workspace. (2025). Create a spreadsheet.

https://developers.google.com/workspace/sheets/api/guides/create#javascript

W3Schoolds.(n.d.). *HTML forms*.

https://www.w3schools.com/html/html_forms.asp

W3Schoolds.(n.d.). *Window localStorage.*

https://www.w3schools.com/jsref/prop_win_localstorage.asp

W3Schoolds.(n.d.). *Json.parse().*

https://www.w3schools.com/js/js_json_parse.asp