

The Plan of Sprint2

Date

21/08/23 – 22/09/23(5 weeks)

Team Members

- [Murong Gu](#)
- [Xunhai Wang](#)
- [Youcheng ZHANG](#)
- [Xintao Chen](#)
- [Haonan TAO](#)

Sprint 3 goal

- Finish all sprint 2 checklist
- Finish module, profile pages develop
- After getting the feedback from Sprint1 and doing well in Sprint2
- Communicate with team members well
- Backend APIs and database schema
- Optimize all documentation with a clear structure
- Communicate with the Redback team with database
- Development code standard and code peer-to-peer review

Sprint Backlog Items:

ID	Task Name	Task Description	Task Dependency	Task Workload	Additional information
1	Confluence Management	Keep the confluence consistent with the Trello board	Task2	3 weeks	Contents are available and updated on Confluence (meeting minutes, scope of the project, diagrams, technologies used in the project, user stories, and test cases). Contents are consistent with Trello (or GitHub project) and with their code repositories.
2	Trello board Management	Keep the Trello board consistent with the Confluence and sprint backlog divided into user stories	Task1 Task 3	3 weeks	We need to organize a product backlog and a lower-level sprint backlog. Tasks in the sprint backlog were estimated, had an appropriate due date, and had a sufficiently low level of granularity. A link to our Trello
3	User Story Management	if needed, perform user story segmentation into sprint backlogs	Previous User Story	3 days	Tasks are also clear, linked to their user stories (Confluence), and offer additional descriptions when necessary.
4	Ethical consideration	Ethical consideration	None	3 days	Make sure that your ethical considerations are documented on Confluence: Clear indication that students discussed/reflected on project ethical issues.
5	Cyber security consideration	Cyber security consideration	None	3 days	Make sure that your cyber security considerations are documented on Confluence: Clear indication that students discussed/reflected on project cyber security issues.
6	Sprint 3 Plan	Sprint 3 Plan documentation and meeting	None	3 days	A clear indication that sprint planning was followed this sprint and a clear, consistent, updated sprint planning for Sprint 3. A clear indication that sprint review was followed this sprint. The team organized a meeting for this, documented discussions, reflections, and the next steps to be taken in Sprint 3.
7	Sprint2 review	Sprint2 review documentation and meeting	None	3 days	Client Review Meeting and get feedback on our project
8	Sprint2 Retrospective	Sprint2 Retrospective documentation and meeting	None	3 days	For members playing in sprint 2 review what we did well and what they didn't do well. Maintain the good parts and improve the bad parts in the next sprint.

9	Product	Sprint 2 product	None	14 days	<p>Make sure that your product satisfies the following criteria:</p> <p>Product is deployed and a URL is available on Confluence and GitHub README so the client can access the current version of this software.</p> <p>FOR PROJECTS YOU CAN'T DEPLOY NOW: Can you emulate the project and demonstrate current progress for us in a short recorded video? we need to be able to measure your development progress in Sprint 2, that's all.</p>
10	Github Management	Github Management	None	14 days	<p>Folders are structured.</p> <p>Sprint 2 documents were exported from Confluence and added to the repository (and are updated)</p> <p>README file is updated and explains the team's repository and new release</p> <p>A baseline tag was generated for this Sprint</p>
11	Frontend needed components	Frontend needed components(re-use on another page)	None	3 days	Basic React components develop
12	Frontend profile page	Frontend profile page	None	3 days	The basic front page is developed according to the UI diagram
13	Frontend Module page	Frontend Module page	None	3 days	The basic front page is developed according to the UI diagram
14	Frontend file upload	Frontend file upload(firebase)	None	2 days	Firebase file upload and delete
15	database schema	database schema	None	3 days	database schema version update
16	Backend needed APIs	Backend needed APIs	None	7 days	Backend needed APIs(CRUD)
17	Fullstack interaction	Frontend and backend interactions	None	5 days	Frontend and backend interactions
18	Meeting	Keep a record of the meeting minutes	None	14 days	Keep the weekly meeting and documentation
19	Code review	Code review	None	3 days	Basic peer-to-peer review according to the provided document
20	Code standard	Code standard	None	3 days	Code standard when development

Scrum meeting

- Weekly Meeting with supervisor 5:30 pm, Monday
- Internal StandUp Meeting: 7:00 pm, Tuesday. 8:30 pm, Friday
- Client Review Meeting: TBD
- Team Meeting with RedBack: TBD

Review and Retrospective

Sprint Review and Retrospective: During 16/09/23–22/09/23

Testing and Validation

To ensure the quality and stability of our front-end and back-end code, the team will implement the following testing strategies:

Unit Testing: Unit tests will be executed before each code commit to ensure that newly added or modified code does not affect existing functionality.

Integration testing: Integration tests will be performed on the front-end and back-end interfaces to ensure that data flow and functional interactions are normal.

User Acceptance Testing: Communicate with the client and ask them to test the new features at the end of each Sprint to gather their feedback.

Performance testing: Before the project is deployed, we will test its performance under high traffic to make sure it can meet the actual usage requirements.

The tests will be conducted using automated testing tools and will be documented on Confluence and shared with team members.

User Experience and Design

To ensure that our products meet user expectations and are easy to use, we will:

Prototyping: Before development, we will create a prototype of the product so that both the team and the client have a clear idea of how the product will ultimately look and function.

User Feedback: We will collect and analyze user feedback to ensure that the design and functionality of the product meet their needs.

Design Iteration: Based on user and customer feedback, we will continue to iterate our design to ensure it achieves the best possible user experience.

Documentation Management

To ensure that the project is sustainable and easy to maintain, we will:

Code comments: When team members submit code, they must add clear comments describing the functionality and logic of the code.

Technical Documentation: In addition to the code, we will create technical documentation that describes the architecture of the system, the technologies used, and third-party libraries.

~~User manuals: To help users use our products, we create a user manual that describes the main features of the product and how to use it.~~

These documents will be stored on Confluence to ensure easy access and use by all team members and future maintainers.

Risk Management

Risks can be encountered at every stage of a project. The team has identified the following potential risks and prepared strategies to address them:

- **Changes in requirements:** Requirements may change during communication with the client. For this reason, we will maintain regular contact with the client and update the requirements document after each change.
- **Technical Challenges:** During the development process, technical problems may be encountered. Team members will regularly share and discuss technical issues and consider advice from external experts.
- **Time constraints:** The project schedule may be delayed due to various reasons. For this reason, we will regularly check the progress of the project and reassign tasks or increase the work hours if necessary.

Risk management documentation is recorded on Confluence to ensure that team members are aware of possible risks and how to deal with them.

Documentation Submission:

- Keep all documentation are order and structured
- Make sure the documentation with proper name on it
- Make sure the GitHub with release Tag and correct branch name (also the push, and commit message)
- The final submission needs to include all documentation (except the meeting minutes)
- Everything needs to be double check before submitting to all team members

Additional

- Continuous optimization of completed tasks
- Continued development of unfinished pages and back-end database interactions
- Break down tasks with team members to ensure that the project schedule is not delayed.
- Keep in touch with the supervisor and client to ensure that the project requirements do not change. Prioritise and deal with changes if they occur.
- Focus on documentation
- Keep good communication with team members

Summary

During the development of this Sprint2, the team will be focusing on a few key areas: firstly, task tracking to ensure that students put together a complete Product Backlog as well as a detailed Sprint Backlog, and tasks that need to be aligned with Trello and its codebase. Front-end and back-end development will be the focus of this Sprint, involving tasks that will ensure the interactivity and functionality of the website or application. Database design, optimization, and management are also considered to ensure data integrity, security, and efficient access. Code review is another core component that students will need to document in detail on GitHub and Confluence. Regarding Sprint planning and reviews, there must be clear plans and updates on Confluence, as well as clear minutes of team meetings. Ethics and cybersecurity are also a focus for this Sprint, and the team must discuss and document both on Confluence. Management of the GitHub repository is also a focus, ensuring that folders are organized, documentation is up to date, and baseline tags are created for this Sprint. This sprint clarified the team's responsibilities and expectations for front-end, back-end, database, and interactions, setting the project up for success. Each team member also works on their corresponding individual tasks in their roles, complementing each other to provide a deliverable end product