

Group 1 – Sprint 3 Review Report

(Riku Kaartoaho, Victoria Vavulina, Riku Toivanen, Monami Kirjavainen)

What has been done in this Sprint:

- Added a notification system to inform customers about due dates.
- Enable customers to delete their account.
- Enabled customers to update their email.
- Allowed customers to extend and return their books.
- Enabled librarians to edit books.
- Enabled librarians to delete books.
- Allowed librarians to handle book extensions and returns for customer.

Highlights:

- Integrated GitHub Actions for CI/CD.
- Automated unit and coverage testing.
- Implement a functional penalty system for overdue books (restricted future reservations).
- Developed and tested the Docker image.

User stories that were covered in Sprint 3:

1. User story: As a librarian, I'm able to view & manage customer reservations from the Librarian dashboard.
2. User story: As a customer, I can return a reserved book from customer dashboard.
3. User story: As a customer, I'm penalized for not returning a book on time (Restricted for future reservations).

Features that were covered in Sprint 3:

1. Feat: Create, update and delete book records (librarian).
2. Feat: Manage reservation status as customer (return book and extend reservation date).
3. Feat: Implement notification feature to notify customer to due dates.

User stories postponed for Sprint 4:

None

Features postponed for Sprint 4:

1. Feat: Integrate email notification system (Supabase).

2. Feat: Enhance notification feature to update when the “Extend” button is clicked and refactored datafetching in the notification component.

Tasks postponed for Sprint 4:

1. Task: Implement unit tests for book extend/return functionality on the librarian side.
2. Task: Implement unit tests for the penalty system.

Sprint Success Measurement

- **Obstacle Identification:**
 - o Unexpected Supabase version update caused compatibility issues in our project.
 - o This created a bug that prevented us from implementing the newly created feature.
 - o The team worked on fixing it and finding a solution to resolve the issue.
- **Task Completion:** Ensure all Jira board tasks are categorized correctly (To Do, In Progress, Done)
- **Sprint Backlog Status:** Monitor progress of planned user stories (Completed user stories, In-progress user stories, Postponed user stories)

The screenshot displays the Jira project board for 'SEP1-LibraryBookingSystem'. The top navigation bar includes tabs for Summary, Timeline, Backlog, Board, Calendar, List, Forms, Goals, All work, Code, Archived work items, Pages, and Shortcuts. Below the navigation bar is a search bar and a filter icon. The main content area shows a sprint backlog for 'SCRUM Sprint 3' (16 Sep - 7 Oct) with 18 work items. Each item is a checkbox followed by a description, a status indicator (green for 'DONE', blue for 'TO DO'), and a priority icon. The first 15 items are marked 'DONE', and the last three items (SCRUM-55, SCRUM-56, SCRUM-57) are marked 'TO DO'. Below the sprint backlog is a section for the 'Backlog' (0 work items), which is currently empty.

| Item ID | Description | Status | Priority |
|----------|--|--------|----------|
| SCRUM-30 | Create UI notifications design element to alert users about due dates (customer) (navbar) | DONE | M5 |
| SCRUM-32 | Create notification function that checking database due dates and send notification to customer page + Unit tests | DONE | M6 |
| SCRUM-34 | Implement deleting account and update user email address(customer side) and UI (customer) + Unit tests | DONE | M6 |
| SCRUM-46 | Implement deleting book records as a librarian in the UI (librarian) + Unit tests | DONE | B1 |
| SCRUM-48 | Design and implement the UI for the book information edit page (librarian) + Unit tests | DONE | B1 |
| SCRUM-49 | Enhance the Filtering system (Navbar, Hero) && Available book section to display only available books + new availableBooks function? + Unit tests | DONE | B1 |
| SCRUM-47 | Integrate email notification system (supabase) | TO DO | B2 |
| SCRUM-44 | Develop and test the Docker image in the local machine | DONE | B6 |
| SCRUM-31 | Implement the functionality (return/extend) and UI (section/table or page) to view and manage customer reservations from the librarian dashboard (librarian) | DONE | B7 |
| SCRUM-45 | Implement a functional penalty system for not returning a book on time (Restricted for future reservations) + UI | DONE | B2 |
| SCRUM-42 | Implement the functionality (extend/return) and enhance the UI to manage the return or extend the reservation date (customer) + Unit tests | DONE | B7 |
| SCRUM-43 | Implement CI/CD pipeline (GitHub Actions) | DONE | B7 |
| SCRUM-52 | Fix ESLint errors and warnings | DONE | B2 |
| SCRUM-53 | Create Navbar section tests (Jest) | DONE | B7 |
| SCRUM-54 | Implement Overdue Book Management system for librarians | DONE | B8 |
| SCRUM-55 | Enhance notification feature to update when extend button is clicked and refactor the data fetch at a notification component | TO DO | M6 |
| SCRUM-56 | Implement unit tests for the penalty system | TO DO | B7 |
| SCRUM-57 | Implement unit tests for book extend/return functionality on the librarian side | TO DO | B7 |

Links:

[Project Jira](#)

<https://rikukaartoaho-1755679798806.atlassian.net/jira/software/projects/SCRUM/boards/1/backlog>

[Project Plan](#)

<https://github.com/rikudi/SEP-R1/blob/main/documents/design/SEP1-ProjectPlanRyh%C3%A4.pdf>

[Project Vision](#)

<https://github.com/rikudi/SEP-R1/blob/main/documents/design/SEP1-ProjectVisionRyh%C3%A4.pdf>

[Figma UI Prototyping](https://www.figma.com/design/pzKWLDwl0qDaUedtmmVdAp/Library-System-Design?nodeid=0-1&p=f&m=draw) <https://www.figma.com/design/pzKWLDwl0qDaUedtmmVdAp/Library-System-Design?nodeid=0-1&p=f&m=draw>

[Project repository](https://github.com/rikudi/SEP-R1) <https://github.com/rikudi/SEP-R1>