Sprint 2 Report

Team 3:

Domenic Amelio

Ella Khozin

Sally Busch

Julien Chagnon

Michael Whitwam

Alexandra Kaine

11/01/2025

1. Project Overview

This project is an academic scheduling tool developed in C++ with a Qt GUI, designed for Queen's University students. It imports .ics course schedules from SOLUS and combines them with user-entered tasks to automatically generate a personalized, conflict-free study calendar. Unlike existing static tools such as Google Calendar or MyStudyLife, this system adapts in real time to new deadlines and priorities using a weighted scheduling algorithm. Its success is defined by being able to generate schedules within 3 seconds and achieving a user satisfaction rating of at least 8/10.

2. Sprint Backlog

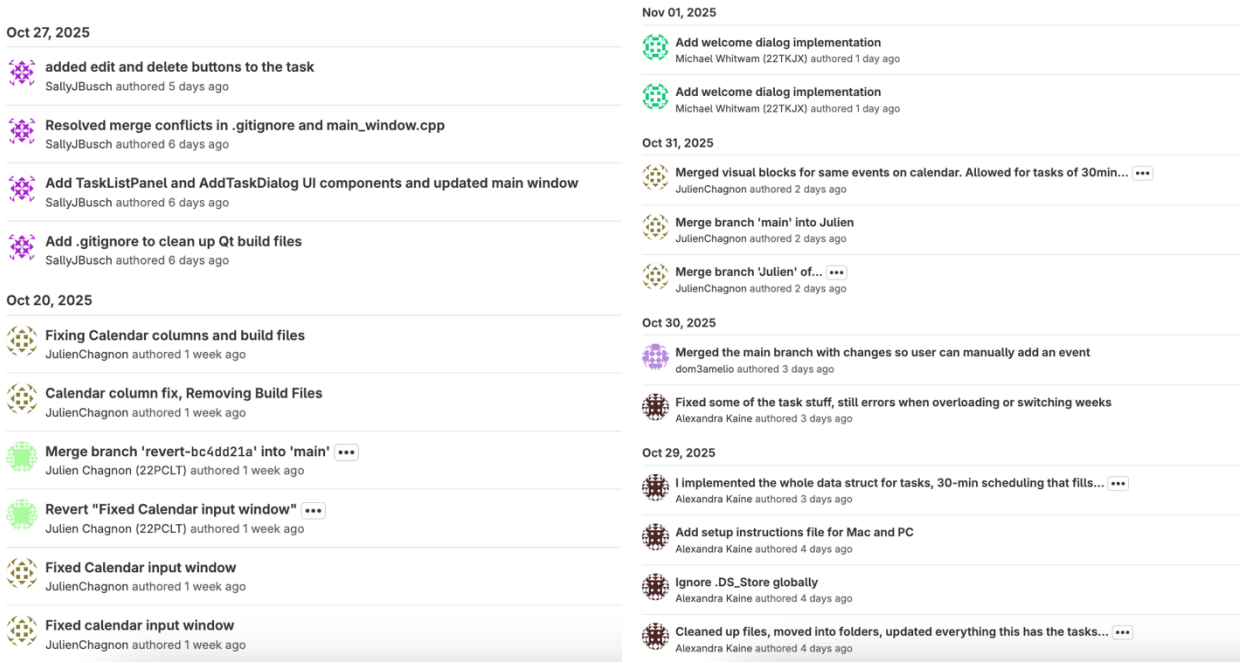
The sprint backlog is outlined below. The automated scheduling feature remains in progress and will be fully implemented in Sprint 3 under Alexandra and Julien.

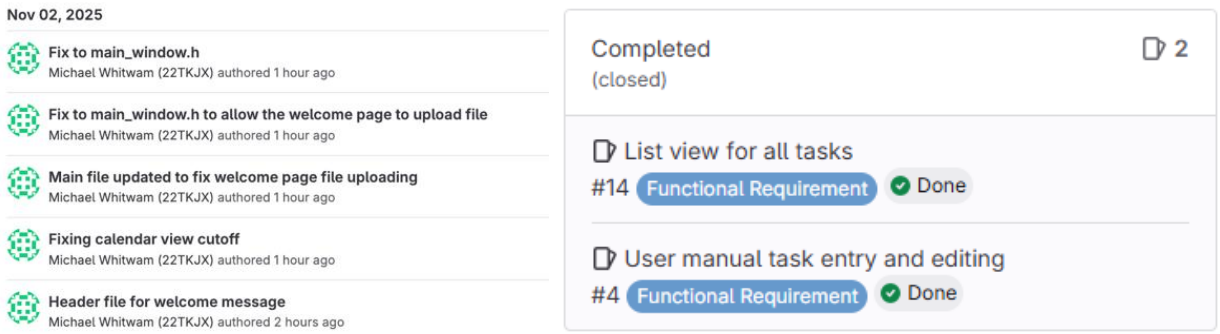
User story (one liner + estimate)	Priority	Assigned to	Status (completed/not)
As a student, I want to view my tasks in a list view so that I can clearly see what is coming up. (4 hours)	High	Sally	Completed
As a student, I want to be able to add, edit and delete tasks so that I can adjust for changing tasks. (4 hours each)	High	Sally (Front end), Alexandra (Back end)	Completed
As a student, I want to be able to add events to my schedule so that I can account for scheduled activities. (4 hours)	High	Domenic	Completed
As a student, I want my tasks to be added to my calendar in a free slot so that I know how to split up my time. (8 hours)	High	Alexandra	Completed
As a student, I want a welcome page that clearly explains the setup and the first step I need to take. (4 hours)	Medium	Michael	Completed

3. GitLab Activity

The team used GitLab for Sprint 2 to coordinate work and integrate new features. Each member contributed through regular commits and merges tied to an assigned task from the user stories. Branches were used for components such as task management, event creation, and the UI layout, which were then merged into the main branch once testing was complete. The group collaborated by pulling each other's changes, resolving merge conflicts together, and verifying builds on both Mac and Windows. The commit and merge history shows steady contributions across the sprint, demonstrating effective teamwork and continuous progress.

Each member contributed actively, as shown in the screenshots below. Sally focused on the task list and UI polish, Alexandra implemented the backend task logic, Domenic handled event creation, Julien supported task-calendar integration, Michael worked on interface layout and testing, and Ella verified features and documented changes.





4. Team Meetings

The team met twice weekly throughout Sprint 2, during tutorials and outside class up to the deadline. The meetings below describe what was done during each, and the last two meetings October 27 and 30 were used to write the report.

4.1 Meeting Logs

Meeting 1 – Mon Oct 6th (10:30 AM Tutorial)
Attendees: Domenic Amelio, Sally Busch, Julien Chagnon, Ella Khozin, Alexandra Kaine, Michael Whitwam
Topics discussed:
-Reviewed sprint 1 progress and defined Sprint 2 goals
-Planned implementation of manual task entry and list view display
-Discussed integration of task data with calendar model for future functional requirement.
Decisions made:
-Assign Sally to task creation and list view implementation
-Assign Domenic to add event creation logic
Next steps:
-Create task card UI templates and define data structures for Task and Event classes

Meeting 2 – Thu Oct 9th (10:30 AM Outside Class)
Attendees: Domenic Amelio, Sally Busch, Julien Chagnon, Ella Khozin, Alexandra Kaine, Michael Whitwam
Topics discussed:
-Progress check on task add/edit features
-Bug reports with Qt layouts
Decisions made:
-Switch to nested vertical layouts for task cards
-Adopt consistent style for buttons and task labels

-Continue testing on both Windows and Mac to catch build issues

Next steps:

-Michael and Ella to finalize UI alignment and styling

Meeting 3 – Mon Oct 20th (10:30 AM Tutorial)

Attendees: Domenic Amelio, Sally Busch, Julien Chagnon, Ella Khozin, Alexandra Kaine, Michael Whitwam

Topics discussed:

-Demo of completed task list view

-Integration of events and tasks into data structure

-UI polish

Decisions made:

-Use centralized TaskManager object to store tasks and emit update signals to views

-Document code and update GitLab milestones for completed stories

Next steps:

-Begin implementing placement of tasks into calendar time slots

-Managing tasks of differing lengths (incorporating 30min increments)

Meeting 4 – Thu Oct 23rd (10:30 AM Outside Class)

Attendees: Domenic Amelio, Sally Busch, Julien Chagnon, Ella Khozin, Alexandra Kaine, Michael Whitwam

Topics discussed:

-Reviewed remaining bugs and incomplete user stories

-Discussed allocation of task study time

Decisions made:

-Opt for drop down menu for time selection rather than typed entry

Next steps:

-Implement study time allocation algorithm and add to visual interface

5. Completed Sprint Evidence

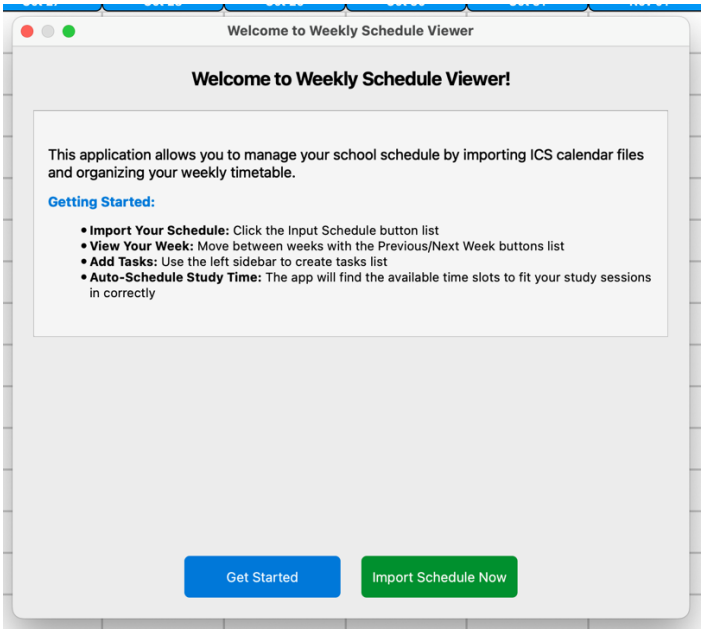


Figure 1: Entry window with initial prompts

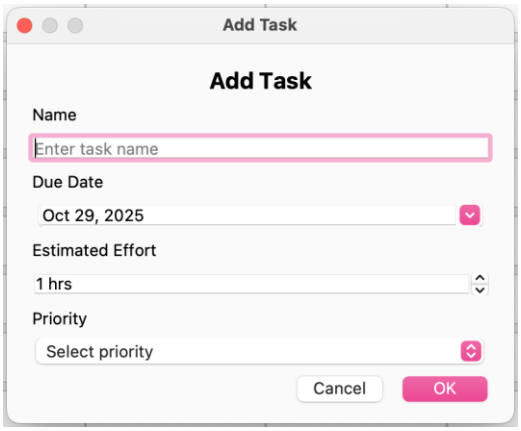


Figure 2: Task info entry window

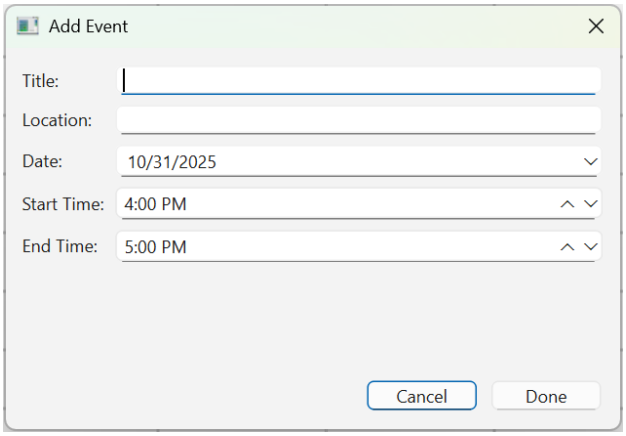


Figure 3: Event info entry window

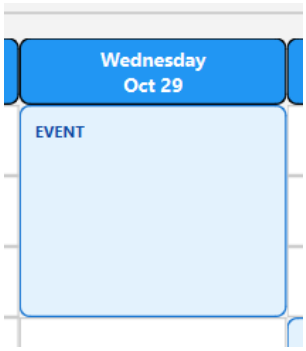


Figure 4: Custom event block in calendar with title "EVENT"

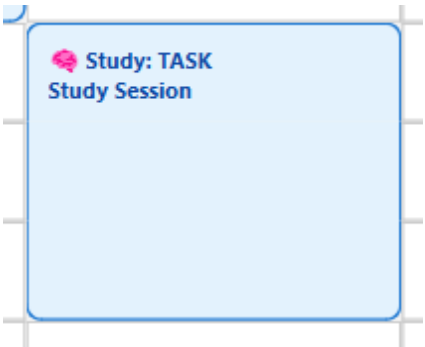


Figure 5: Allocated study/work time for sample task

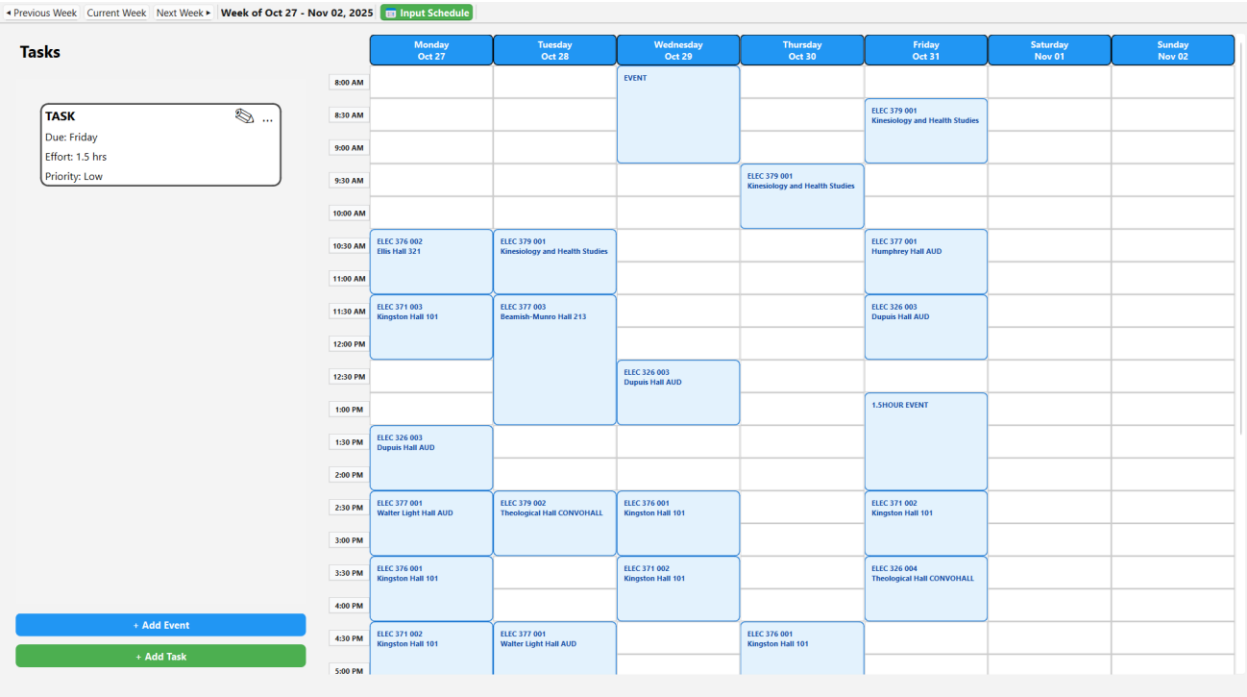


Figure 6: Current view of main calendar window, including vertical list view of tasks

6. Sprint Review

Our TA meeting is scheduled for Monday October 3rd at 6:30pm.

7. Sprint Review Details

Our sprint review was held during the tutorial on Monday, October 28, attended by Sally Busch, Domenic Amelio, Julien Chagnon, Michael Whitwam, and Alexandra Kaine. During the meeting, the team demonstrated the completed features including the task list view, task addition, editing and deletion functionality and the addition of events.

The team successfully met the sprint goal. All the high priority features planned for this sprint were completed. The “Add Task” button was created with the functionality to add tasks with a name, due date, time commitment and priority. The list view for tasks was implemented and tested. The edit and delete functionality for tasks was finished. Event addition was also completed and verified to display correctly. These new features will allow users to both visualize their weekly academic schedules and everything that they must get done.

One challenge was ensuring that the edit and delete buttons for each task were positioned and styled consistently within each task card. We experimented with different layout combinations (QHBoxLayout, QVBoxLayout, and nested wrappers) to prevent clipping and horizontal scrolling. We also tested multiple icon options and UI adjustments to make the task cards visually consistent, without relying on external image imports or dependencies (like QtAwesome) that introduced build errors.

After reviewing the outcomes of this sprint, the team updated the product backlog. The next sprint will focus on updating the schedule to account for new tasks being added and more study time being allocated. The user story, "As a student, I want the system to automatically update my schedule when new projects and tasks come up so that I don't have to constantly re-schedule" was added to the backlog.

During our first sprint review, our TA suggested exploring ways to incorporate assignment deadlines and due dates directly from OnQ into the schedule. While this would improve usability, the team determined that it is currently outside of the project's scope, as direct API access to OnQ would be required to import data automatically. Instead, during this sprint, we implemented manual task creation allowing students to input their own assignment deadlines. This approach maintains flexibility while avoiding the need for integration with external systems.

8. Sprint Retrospective

Date: October 30th, 2025
Time: 10:30 AM (Thursday meeting)
Attendees: Domenic Amelio, Ella Khozin, Sally Busch, Julien Chagnon, Michael Whitwam, Alexandra Kaine
Scrum Master: Michael Whitwam
Product Owner: Sally Busch

Sprint 2 focused on refining the UI and adding more manual entry features across the application. The team successfully completed additional UI features and general refinement. This included the ability to add, delete, edit, and view tasks, as well as add custom events to view on the calendar. The GitLab was well kept, the user stories for this sprint were well defined and proper subtasks and milestones were created. The team also became much more familiar with the software being used making it easier to implement changes. This helped with project progress when front end and backend were being completed by different team members and needed to be integrated.

The team faced challenges with the build and integration of the application across different devices. Team members lost time as a lot of problems arose when rebuilding the project after others pushed changes because of platform differences like windows vs mac. The UI was improved since sprint 1 as planned however the team will continue to update and polish the UI as there is more room for user friendliness and readability. The team also planned to implement the automated study schedule however it hasn't been fully completed and remains in the backlog for Sprint 3.

During team meetings, we reviewed progress, completed work, and defined goals for upcoming functionality and features we wanted to implement. The team accomplished this by assigning tasks for that week and discussing the general next steps for the project. The team updated GitLab milestones and user stories throughout the sprint to reflect their progress. The team also communicated regularly through Teams and message chats. Overall, the communication both in and outside of meetings was very effective throughout the sprint.

For Sprint 3, the focus will be on implementing the remaining functional and non-functional requirements outlined in the user stories. This includes developing the study schedule generator, which will automatically update as a user adds new tasks and events to the calendar. The team also plans to implement better security for accessing the application above just hosting the project locally, this will most likely be done through a sign in page. Because sprint 3 will be the final sprint of the project more emphasis will be put on testing the application and its corner cases before final submission.