Concordia University	SOEN 287: Web Programming
Gina Cody School of Engineering and Computer Science	Fall 2025

Project

Campus Resource Booking & Management System

Notes

- *Project to be done in groups.*
- *Each group should have **4 students**. **This is very important.***
- *If you reuse material from other sources (internet, books, papers, websites…), you must provide appropriate citations.*

1. Project (quick) description

Universities and colleges often provide shared resources to students and faculty — such as study rooms, labs, sports facilities, specialized equipment, or even software seats. Managing the scheduling, usage, and availability of these resources is a common challenge. Students need a convenient way to see what's available and make bookings. Administrators need tools to manage resources, set availability, approve requests, and monitor usage patterns.

Your team has been asked to design and develop a web-based system to manage the booking of university resources. The system should cater to both end users (students/faculty) and administrators (resource managers).

2. Minimum Required Features

End Users (students/faculty) should be able to:

- Create and manage their account (profile info, contact email, etc.).
- Browse a list of available resources (rooms, labs, equipment…).
- View real-time availability calendars/schedules of resources.
- Make a booking/reservation for a resource (with date, time, purpose, etc.).
- Modify or cancel their bookings.
- View their past and upcoming bookings in one place.

Administrators (resource managers) should be able to:

- Create, edit, and remove resources (names, descriptions, locations, capacities, images).
- Set availability schedules (working hours, exceptions, blackout dates).
- Approve or reject booking requests (if necessary).
- Block or unblock resources temporarily (e.g., for maintenance).
- See statistics/reports about resource usage (popular resources, peak times).

3. Technical/Implementation Notes

- The system should be web-based, responsive, and accessible from modern browsers.
- It should have authentication and authorization, with different roles (User/Admin).
- Data must be persisted.
- Booking conflicts must be handled gracefully (no double-booking).
- The design should be modular enough to allow future additions (like payment for premium resources or integration with campus card systems).
- You can use any library you want.
- The backend is based on Node.js.

4. Bonus

- End Users (students/faculty) should be able to receive notifications (email or in-app) for booking confirmations, changes, or cancellations.
- Administrator should be able to send announcements or notifications to users (e.g., maintenance downtime).

Note: Please note that the description presented above is preliminary and other requirements/features might be needed.

5. Deliverables

There are two (2) deliverables for this project as follows:

- **Deliverable 1 – Frontend**: due Sunday, October 26, 2025. This deliverable is about what a user of the system can see and do. Please take note: a user here means any user of the website. This can be a student, faculty, or administrator. For this deliverable, data might be entered during execution or might be "hard-coded".

  Marking rubric for deliverable #1:
  - Completeness: 5 marks
  - Look and feel(style): 5 marks
  - Navigation between pages: 5 marks
  - Total marks:15

- **Deliverable 2 - Backend**: due Sunday, November 23, 2025. In this deliverable, data received from users, generated, or sent to users is stored in a data store residing behind a web server.
- There will be no hard-coded data as of deliverable #2 as data will go to/come from the data store. Also, you might need to revisit some of the features you implemented in deliverable #1.

  Marking rubric for deliverable #2:
  - Completeness: 3 marks
  - Look and feel(style): 1 mark
  - Navigation between pages: 1 mark
  - Data store: 10 marks
  - Tota marks:15

Bonus: 2 points

For each deliverable, you are going to demo your project. When each deliverable is due, you will be notified so you can pick a demo slot from Moodle scheduler. All demos will happen online through Zoom. More information on scheduling and demoing will be provided close to the deadline.

In addition to the demo, you need to submit the following with each deliverable:

- A list of features that have been implemented so far,
- A list of features to be implemented in the next deliverable,

- A completed and signed “Teamwork Discussion” sheet (see Moodle) showing the contribution(s) of each team member,
- A compressed file containing all your website’s files,
- An installation guide stating how to deploy your website, and
- A user guide showing how to use your website.

6. Submission:

You need to submit each deliverable before 11:59pm of its specified due date above through Moodle.

Please see the Project’s section in Moodle for the submission link.

A few points please:

- The project must be done in teams of 4 students each.
- You have the responsibility to make your own team. I will post in Moodle the list of all students enrolled in the course: names and emails. As soon as you have a team, please remove your name from that list.
- Make sure you meet, as a group, 2 or 3 times a week to check each other’s contributions and progress. **Statements like “we were surprised on the day of delivery that a colleague did not do their part” will not be accepted.**
- Make sure you submit each deliverable ahead of the deadline. That is, do not wait until the last minute.
- All submissions will be through Moodle. Do not send us your submission by email, as we cannot submit it for you. **Email submissions will not be accepted**.
- Only one member of the team can submit, do not submit more than once.
- However, it is understood that what is submitted is endorsed by all team members.

7. Marking considerations

For the marking of your deliverables, please keep in mind the following:

- Not all group members will necessarily get the same mark. There is a mark for the whole team and there are individual marks. The individual marks are derived from the contribution as stated in the "Teamwork discussion sheet" document and from the outcome of the demo.
- There is a mark for the work as a product of a team. If the product is not good, all team members will lose marks. A statement like "My part is working fine, the problem is in others' parts" is not accepted. After all, as engineers, you need to work as a team to deliver a professional product.
- After grading, if you send us emails such as "I did most of the work in deliverable #1 …",

please make sure to CC all of your team members. We will need confirmation from the other members on this. In that case, team members might have their marks appropriately reviewed (up or down).

- Make sure you detail the contribution(s) of each team member in the discussion sheet. The golden rule here is: "**Be professional in reporting the contributions and do not cover up for a team member who did not contribute.**".