University of Calgary

SENG 513 Project

Web-Based Systems

Rate My Course

GROUP 20

Haris Ahmad (30088192)

Yiannis Hontzias (30066528)

Mohammed Allam (30106564)

Kai Ho Cha (30147119)

Haider Tawfik (30097912)

Anish Pokhrel (30115576)

Schedule

The following is our original project timeline, created as a part of the project proposal:

Phase 1 (Oct 31): Static UI Design

- Create static user interface design and wireframes.

Phase 2 (Nov 10): Interactive Mockup/Prototype

- Develop an interactive mockup.
- Choose development technologies.

Phase 3 (Nov 15): Core Feature Implementation

- Implement user account management, university requests, courses, and reviews.
- Maintain a functional version in the repository.

Phase 4 (Nov 21): Optional and Challenging Features

- Add optional features like professor creation.
- Tackle challenging features like account registration and email verification.

Phase 5 (Nov 28): UI Polishing and Testing

- Polish the UI and design.
- Debug, test, and finalize the website.

Here's how our project timeline was modified at the end of the project:

Phase 1 (Oct 31): Static UI Design → Completed on Time

- Create static user interface design and wireframes.

Phase 2 (Nov 10): Interactive Mockup/Prototype → Completed on Time

- Develop an interactive mockup.
- Choose development technologies.

Phase 3 (Dec 03): Core Feature Implementation → Extended

- Implement user account management, university requests, courses, and reviews.
- Maintain a functional version in the repository.

Phase 4 (Dec 06): Optional and Challenging Features → Only account registration was completed

- ~~Add optional features like professor creation.~~ **(Removed)**
- Tackle challenging features like account registration and email verification.

Phase 5 (Dec 06): UI Polishing and Testing → Extended

- Polish the UI and design.
- Debug, test, and finalize the website.

Phase 1 and **Phase 2** were completed on schedule however, deviations from the project timeline began between **Phase 3** and **Phase 4**. This occurred during the "Core Feature Implementation", we started seeing issues trying to implement the prototypes from **Phase 2** (Interactive Mockup/Prototype). Due to this, the overall project timeline had to be extended to provide more time for the team to complete **Phase 3** and **Phase 4**.
However, our project timeline was designed to give us extra time in case of any difficulties. Therefore, this issue was easily adapted as the team utilized the extra available time, which allowed us to push deadlines to December 6.

The following were factors that led to delays and changes in the project timeline:

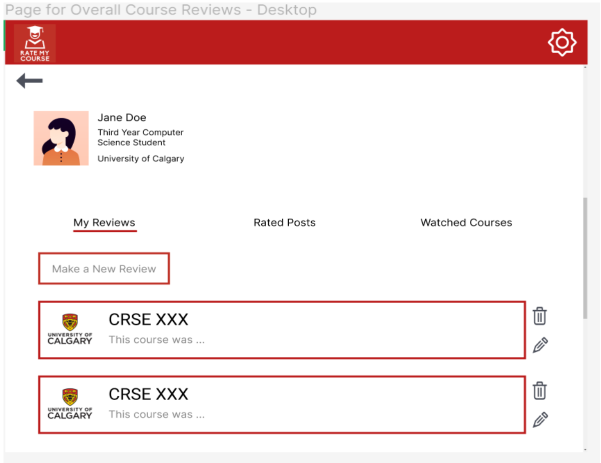
- Limited time we had to deploy this project (2 weeks after the results of the proposal were released)
- Courseload from other courses affecting this project
- Knowledge gaps for using certain tools, such as React, Docker, Django, etc.

GUI Changes

When comparing our original designs with the final implementation we noticed some inconsistent styling across pages that required addressing.

Here in the original “My Reviews” tab on the users’ profile, users are displayed the reviews that they have written, with the ability to edit them, and or delete them if desired. However, this review form did not match the one the user originally filled out.

On the left, we have the original “My Reviews” tab, and on the right, we can see the updated one.

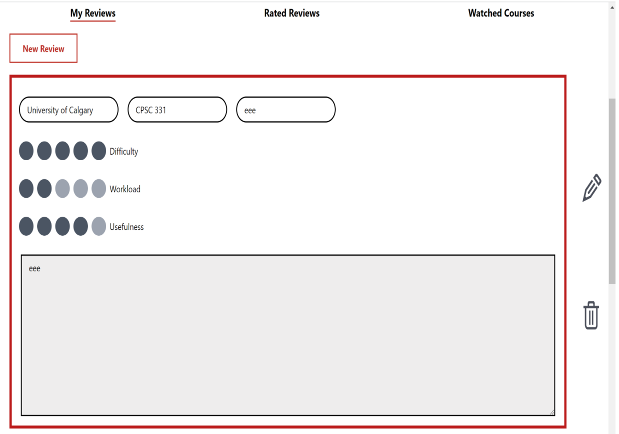


Page for Overall Course Reviews - Desktop

Jane Doe
Third Year Computer Science Student
University of Calgary

CRSE XXX
This course was ...

CRSE XXX
This course was ...



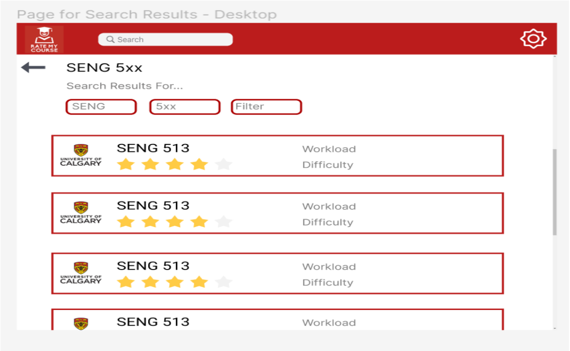
University of Calgary | CPSC 331 | eee

Difficulty ●●●●●
Workload ●●○○●
Usefulness ●●●●○

eee

This change was done to keep styles consistent as in the mock-up the information about their reviews did not match the forum they filled out, in the updated version it more closely aligns with the review forum the user filled out, and the fields are also grayed out to indicate that users cannot modify their previous review until the edit button is clicked.

Another change we made is to the search result page, here we changed the search bar style, and information displayed to more closely match the landing page. On the left, we have the previous search results page, on the right we have the updated one.



Page for Search Results - Desktop

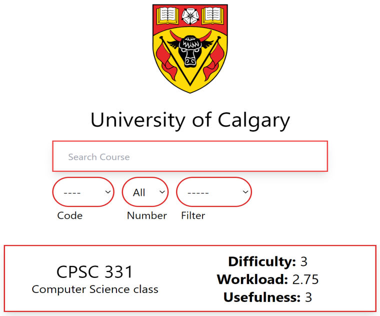
SENG 5xx
Search Results For...

SENG 513 ★★★★☆ Workload Difficulty

SENG 513 ★★★★☆ Workload Difficulty

SENG 513 ★★★★☆ Workload Difficulty

SENG 513 Workload



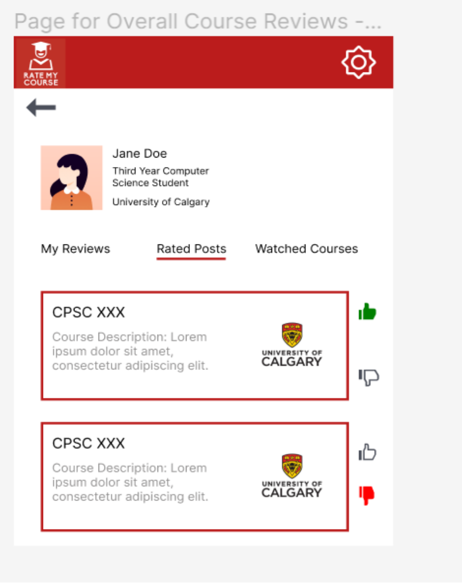
University of Calgary

CPSC 331
Computer Science class

Difficulty: 3
Workload: 2.75
Usefulness: 3

From our changes, one can see what university they have clicked on, as well as a quick overview of how others perceived the difficulty, workload, and usefulness on a scale of 1 - 5. Users can also see the name of the class. We also removed the university logo being displayed for each class as we felt putting it at the top would help reduce redundancy.

Another reason for some changes was due to issues concerning the mobile interface where some buttons may be accidentally clicked due to being too close together. For example on “My Reviews” and “Liked Posts”, on the mobile interface, the clickable buttons were close together and hard to click since they were so far on the right side of the screen.



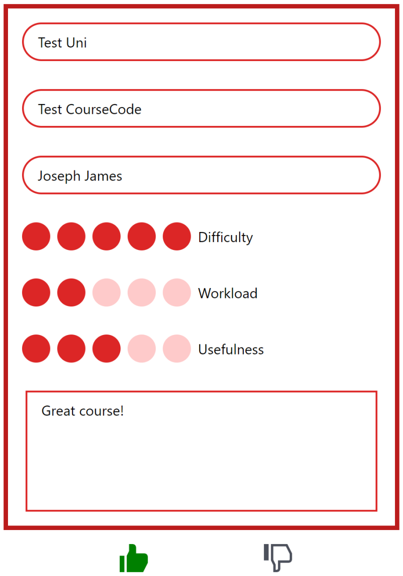
Page for Overall Course Reviews -...

Jane Doe
Third Year Computer Science Student
University of Calgary

My Reviews | Rated Posts | Watched Courses

CPSC XXX
Course Description: Lorem ipsum dolor sit amet, consectetur adipiscing elit.

CPSC XXX
Course Description: Lorem ipsum dolor sit amet, consectetur adipiscing elit.



Test Uni

Test CourseCode

Joseph James

●●●●● Difficulty

●●○○○ Workload

●●●○○ Usefulness

Great course!

Here we can see that on the mobile view, the buttons for liking and disliking are moved to the bottom. This was to prevent issues where the user would accidentally click the wrong button, due to spacing issues. The same is done for editing and deleting a review on the “My Reviews” tab.

Unexpected Challenges

Challenges

The following are some of the unexpected challenges the team faced:

- **Complexity of seemingly “simple tasks”**
    - Initially underestimated the intricacies of implementing basic functionalities, such dynamically rendering data to user, keeping track of user’s state when user is logged in.
    - Required additional time and effort to ensure a good user experience.
- **Knowledge gaps in technologies, such as React, Django, Postgres**
    - Took more time and effort to learn and utilize these technologies
    - Made certain functionalities too difficult to implement considering the limited timeline
- **Multi-User Support and Roles:**
    - Introduction of user complexities not fully considered in the early planning stages.
    - For instance, tracking the state of the user for all pages upon logging in was more difficult than initially perceived, which caused challenges when wanting to fetch and send data for each unique user.
    - Required a more comprehensive framework for user management.
- **Data Persistence:**
    - Challenges in ensuring proficient data storage and retrieval within project time constraints.
    - Required a reduction of user data to be stored in the database, such as removing low-priority client attributes to user on storing more essential user information.

Overcoming Challenges

The following describes how the team tackled the challenges presented above:

- **Resource Reallocation and Timeline Extension:**
    - Team was able to reallocate resources and skills to address unforeseen challenges,
    - This was done by managing members to focus on high-priority core functionalities such as rating reviews, posting reviews, and account registration, rather than low-priority account settings.
    - Project timeline was extended to accommodate necessary refinements and optimizations.

- Overall, this provided the team with more time and resources to complete all the core functionalities

- **Excellent collaborative effort**
    - Team commitment to overcoming challenges through collaboration.
    - Successful navigation of complexities to deliver a robust rating website.
    - Weekly meetings were increased to daily meetings to ensure there was more time to complete the project
- **Online resources**
    - Online resources such as StackOverflow, ChatGPT, and Google were the primary tools to tackle challenges related to project development
    - Team members possessed excellent research skills to tackle any challenges quickly and effectively
    - For instance, ChatGPT was used to determine why a certain functionality was not working. The proposed solution was first analyzed by the team and then implemented once it was understood and considered functionally correct.

Knowledge Gaps

As we began to start our project many members encountered knowledge gaps in the required technologies that we would be using in our project despite the coverage of some of the technologies during lectures and tutorials. Learning how to use the technologies described in our proposal such as React, Django, Tailwind and Docker presented a steep learning curve for group members who have never used them before in a nontrivial manner.

To address the challenges posed by knowledge gaps in the technologies outlined in our project proposal we used the varying levels of knowledge in terms of the technologies to our advantage. This included things such as a guide for setting up React, Django, and Docker. Additionally, those who had greater knowledge were willing to spend their time teaching those others the fundamentals and some specific details with examples to give them a head start in completing their tasks as opposed to having to learn it from scratch. This expedited the development process as those who were still learning were free to ask the others for help and would have their queries answered. Even if no one knew the solution to the problem, many group members were often willing to meet up to try and work together in an attempt to find a solution. Once a solution to the problem was found, the other team members would benefit by learning how they came to the solution. This fostered a collaborative environment where everyone shared deep insights in how to solve challenges.

Another method group members used was effective self-learning techniques. By attempting something failing, trying it a different way, getting it wrong again, doing their own research using websites such as StackOverflow, ChatGPT, and Google to find the information they needed, and then finally getting it to work allowed group members to fully understand the problem and the solution. By being able to figure it out themselves they had a much greater grasp on the concept and were able to teach it to others who might've been struggling with the same issues.

Despite the challenges presented by the described knowledge and skills gaps, our groups' willingness to assist those in need, whether it be by helping install and run the required technologies, going over detailed examples at the beginning, working together to find solutions to problems, and each group members utilization of effective self-learning techniques allowed our group to successfully overcome a majority of the knowledge and skill gap.

Successes and Shortcomings

Successes

Communication

To foster effective communication our team decided to set up a Discord server, where members were frequently active. This server allowed members to ask questions regarding details, issues, and so on and the other team members were quick to respond with insightful answers or offer help if needed resulting in an effective workflow. In this server meetings would be hosted on Tuesdays at 5:00 pm with all members in attendance, during these meetings, we would discuss what tasks each member was able to complete, what tasks members needed more time to work on, and assign new tasks according to each member. This way each team member could ask for assistance if they are having trouble on a task, and team members that completed their task could move forward with the next steps in development allowing for effective workflow. During these meetings, those who finished their tasks would demonstrate their changes to the others, and if their work was deemed up to standard by the other group members, their branch would be merged into the "production" branch.

Splitting of Tasks

To split tasks, the team divided the members into roughly two halves: the "frontend team" and the "backend team". Each member assigned themselves to one of the two teams based on their interests and skills. Any member from either team was allowed to be part of the other team for flexibility and adaptability. This way situations, such as the frontend team having too many tasks while the backend experienced a lack of tasks, were mitigated as members from either team could switch roles and assist when needed.

Once each member assigned themselves to their respective team(s), the team gathered in weekly "general team meetings" to discuss tasks, concerns, and queries. In addition, the frontend and backend teams held their own weekly meetings in which they discussed tasks, concerns, and queries. Due to this, in the case that member had trouble with their task, the rest of the team was able to support them.

UI prototyping

It is said that most software projects fail, ranging from 50-80% rate of failure. One of the main reasons for this is insufficient planning. The team was aware of this, so we placed heavy emphasis on creating a well-structured plan. One of the successful aspects of our planning was the use of collaborative interface tools to help create user interface (UI) mockups before starting **Phase 4** (Core Feature Implementation). The team used Figma, a collaborative interface design tool, to design the initial UI prototype. Using Figma, the team generates high-level ideas for the project in the form of visual sketches. By doing this, it exposed flaws in our designs, allowing us to modify our ideas before implementing them in **Phase 4**. For instance, the team caught issues such as missing pages, missing use cases, and irrelevant functionality before implementing them. The use of Figma mainly assisted in the frontend development, as the frontend team had a reference point that included required images as well (such as edit button, delete button, logo, etc.). Overall, this was a success as it allowed us to have a clear vision for the project and detect flaws early in the development.

Shortcomings

Timeline Delay

In our project, one shortcoming we experienced was multiple delays due to us underestimating the scope of certain functionalities that we hoped to achieve. As a result, those who attempted to work on the functionalities were unable to work on anything else until they were completed, slowing down the implementation of that which was dependent on it. Another reason for the delay was to realize that certain functions were implemented, however as we got further into development we concluded that some things were irrelevant such as users having to input their year of study, university, and degree into the system on sign up, however, these were never used anywhere except for the initial sign up. As a result, those who developed it had to spend time changing the front and back end accordingly to accommodate this removal. Overall, these shortcomings resulted in delays in the project timeline

Overall project rating

Final quality

Considering the time given to develop this project we believe the project delivered is of high quality. In terms of UI, the layout of the website is simple, easy to use, and aesthetically pleasing. Similarly, in terms of functionality, the project is simple and easy to use, making it easy to use for university students. The final product seamlessly integrates into mobile screens, making it versatile for students to use on their desired platform. Furthermore, thorough testing of edge cases using use case stories ensured that issues of unexpected behavior did not occur. However, some functionality is missing due to time constraints, such as the use of a web-scraper. To enhance the user experience on Rate My Course, the team is planning to implement a web-scraper to retrieve universities and university information, such as course data.

Process of Working

In terms of the process of working on the project, we displayed effective collaboration, and communication, and worked together to solve problems. Our group held weekly meetings on Tuesdays at 5:00 pm to determine the next steps to take moving forward to ensure we are continually working and development is not coming to a halt due to certain issues and to assign new tasks as needed to members. During these meetings, issues with designs were also brought up, in which the group collectively brainstormed effective solutions. After the group meeting concluded members would go on to develop their assigned task in their branch on GitHub. Once the assigned task was completed members would create a pull request that would be tested and reviewed by other group members before being merged in the production/main branch.

Learning outcomes

All team members successfully demonstrated proficiency with Tailwind, React, Django, GitHub, and other tools that were used. Furthermore, each member was present and participated in every step of the development process. The learning outcomes were meaningful, as everyone gained valuable experience being exposed to new technologies, problem-solving, and working together towards a common goal. While there were minor hiccups along the way, the overall process was smooth, and the project's success is reflective of the dedication and skills demonstrated by the team.

Time and Effort Spent

The amount of time spent on this project can be described in three stages:

1. **Initial Stage**
2. **Intermediate Stage**, and
3. **Final Stage**

In the **Initial Stage,** the amount of time spent working on this project was low-medium. This was expected as at this stage the team is brainstorming ideas, and creating sketches for each promising idea, this is reflected as meetings only occurred on a weekly basis [1].

In the **Intermediate Stages**, the team began working on the implementation of the Rate My Prof design to create using Figma a mockup UI. From this mockup, the frontend and backend teams began working on their respective parts. At this stage, the time spent on developing the project gradually increased. Signs of successes and shortcomings began to emerge and the team was exposed to challenges.

Lastly, in the **Final Stages**, the effort and time spent working on the project increased considerably. At this stage, the frontend and backend teams have integrated their core functionalities, and the full-stack project is nearly finalized. Small design touch-ups and tests were executed. This can be seen as team meetings occurred more frequently from weekly to daily as seen in the meeting hours table [1].

Due to the factors, such as final quality, the process of working and difficulty, learning outcomes, and time and effort spent, the team graded the project “Rate My Course” an 8/10.

Appendix

Meeting minutes:
Everyone was in attendance for all meetings for the entire duration

Date	Time met (hours)
October 31	1
November 7	1
November 14	1.5
November 21	1
November 28	2
November 29	2
November 30	1.5
November 31	2
December 1	1
December 2	2
December 3	3
December 4	3
December 5	5
December 6	7