CMPT 363
Group Project Part 2

Team 35

Team Information

Barun Gambhir – bsg10@sfu.ca – 301437428

Gursahib Badwal – gsb18@sfu.ca - 301444626

Part 2a: Context of Use and Requirements Established

Context Identification:

W**hen/Where:** The users can access the app from their smartphone at any point in time no matter where they are, if they have access to a stable internet connection.

Who: All SFU members, be they students, alumni, professors, or exchange students. The users of the application can be of any gender, age, ethnicity, or educational background as the SFU community is very diverse.

What: To add a review mechanism to the dining feature of the SFU Snap app that would allow the users to read and post reviews about any restaurant being shown within the dining feature.

How: First, by adding a review section to the home page of each restaurant, this review section acts as a button and directs the user to a review page that contains all the reviews posted by other users of the SFU Snap app. The review page also includes an add button that directs the user to an add-review page that allows the user to post a review about their personal experience. This add review page consists of two options in which the user can provide their input/review, the first one being the option to select the number of stars out of five and the second option will be to allow the user to write a comment about the restaurant in a text box.

User Identification:

Persona 1: Saumil Sharma is a second-year Computing Science student at SFU. He enjoys playing football and goes to the gym regularly. He is a full-time student taking five courses each semester due to which he is on campus throughout the weekdays. To keep up with his gym progress he grabs protein-rich meals from the campus dining areas whenever he is on campus. So, he usually uses the SFU Snap app to check out different dining options and to see what reviews those dining options have been getting from other people at SFU.

Persona 2: Amandeep Singh is a first-year international student majoring in psychology. He is currently in his first semester at SFU and lives in the campus residence. He loves to produce music and is also a Twitch streamer who majorly

streams Fortnite. He has a meal plan at the SFU Dining commons and is often hanging out with his colleagues there during dinner times but also likes to check out other dining options on campus for a change of taste. Moreover, being an international student, Amandeep has a tight budget so he usually goes through the reviews of the dining options through the SFU Snap app s he could get the best food for his money.

Persona 3: Kendalpreet Grewal is a third-year Criminology major at SFU, and she loves to post food vlogs on her YouTube channel. She enjoys roaming about all three campuses of SFU and checking out different restaurants to get as much content for her channel. Furthermore, Kendalpreet always mentions in her videos if the restaurant she visited was allergy-friendly to help her audience who might be allergic to food items like peanuts or tree nuts as she has a peanut allergy. Therefore, her food vlogs are both interesting and useful for SFU students. Also, she posts reviews about all the restaurants, on the SFU Snap app using the review feature of the SFU Snap application.

Three Functional Requirements:

1. (FR1) The home page of each restaurant in the dining feature should display the average star rating and the number of reviews on which the star rating is based.

2. (FR 2) Every restaurant page should have a reviews section which upon click, directs the user to a read-reviews page that consists of all the reviews about that restaurant.

3. (FR 3) Every read-reviews page should have a feature in the form of a button that the user can click to get directed to an add-review page which allows the user to add a review in the form of a star rating and/or write a review in the text box present on the add-review page. Additionally, there should be a submit button within the add-review page that allows the users to submit their reviews.

Three Non-Functional Requirements (with their relation to functional requirements):

1. (NFR 1) The average star rating and the number of ratings being shown on the home page of each restaurant should only be based upon the ratings/reviews

provided by the users in the previous 730 days (reviews received on the days before that should not be considered). This non-functional requirement relates to the first functional requirement mentioned above as this requirement determines what numbers (average rating and number of ratings) would be shown on the home page of each restaurant.

2. (NFR 2) The colour of the stars on the add-review page should change according to the user's selection of stars. For example, if the user plans to give a restaurant a rating of three out of five, then the user can select three stars which should change the colour of the first three stars out of the total five stars present on the page. This non-functional requirement relates to the third functional requirement mentioned above if the colour of the selected stars changes then it will provide the users with good feedback before they submit a review. Hence, making sure that they submit an accurate review.

3. (NFR 3) The reviews that will be shown on the read-reviews page should be ordered such that the most recently added review should appear on the top, then the less recent review and so on. This non-functional requirement relates to the second functional requirement as the order of the reviews in the read-reviews page will be determined by this non-functional requirement.

Part 2b: Medium-Fidelity Prototype

The review mechanism features designing process started by first creating a sketch for our functional and non-functional requirements and then combining them to form some LFPs on Balsamiq.

After, generating some LFPs for the review mechanism (sketches attached in the appendix), the next step was to generate MFPs on Figma. The most important decision in generating the overall MFP was to choose the LFPs which included the required features, generated from the functional and non-functional requirements, incorporated. This included combining the LFPs containing the required feature together and implementing the resulting design as an MFP on Figma.

As mentioned in the project part 2 description, we started designing our horizontal prototype, H-MFP first.

Horizontal MFP

The horizontal prototype of our review mechanism contains all the requirements as listed in part 2a.

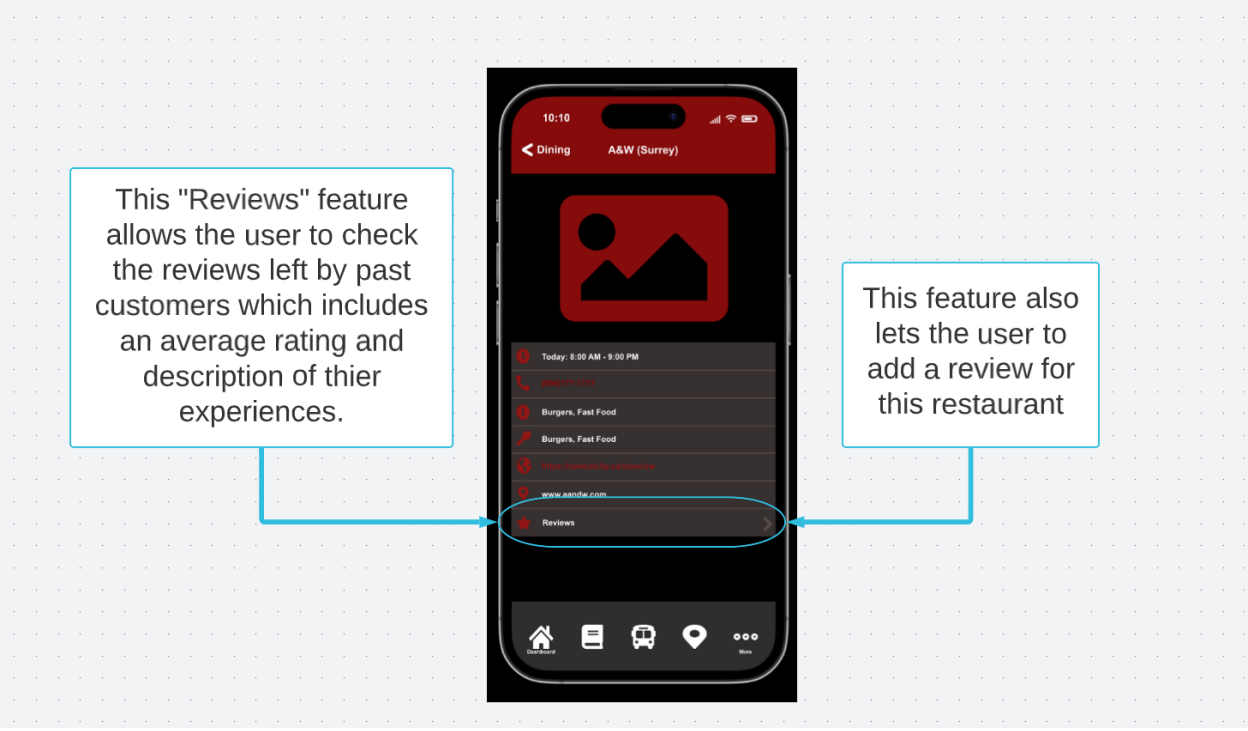

IMAGE 1.1

Image 1.1 shows a random restaurant (A&W, surrey) page containing the newly added review mechanism feature at the very bottom of the already existing information features of the restaurant.

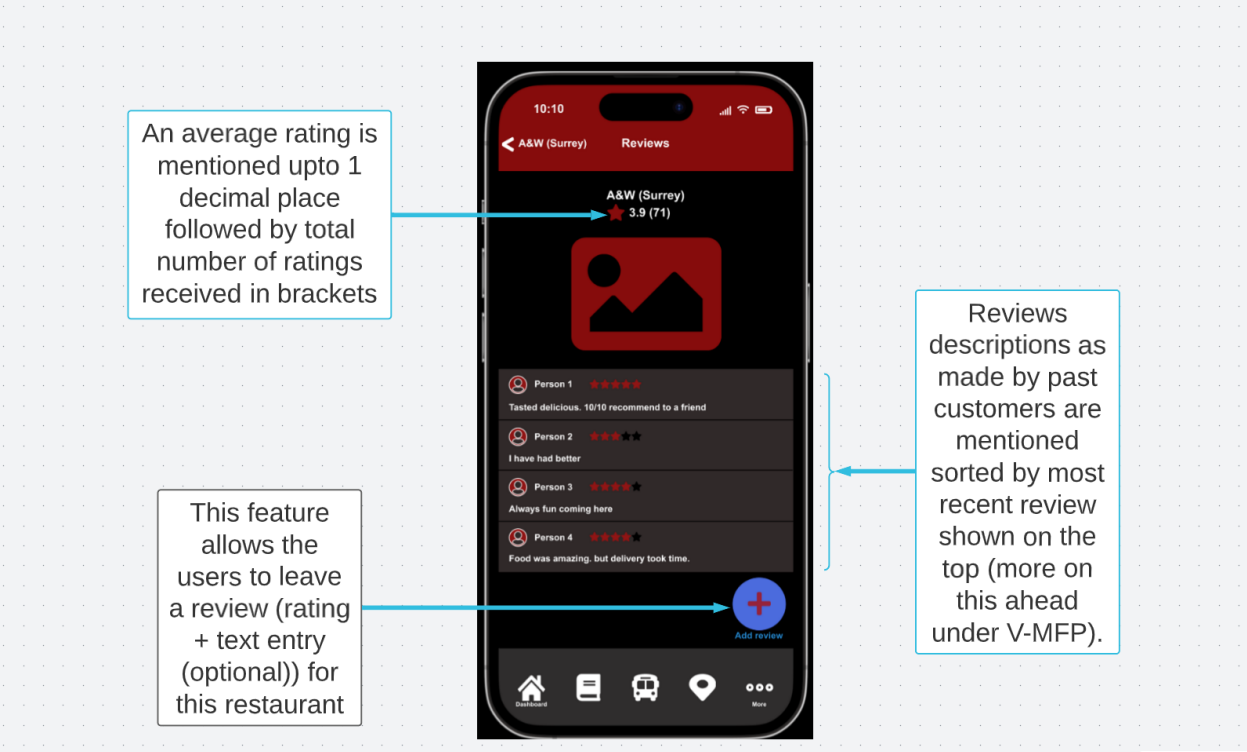

IMAGE 1.2

Image 1.2 shows the Reviews page of the A&W (Surrey) restaurant containing most of our requirements.

Functional Requirements Implementation: The “Reviews” page as shown in Image 1.2 contains the average rating (out of 5) with the total number of reviews received in the brackets as received by the restaurant displayed on the top (FR 1). In image 1.1, the review feature button present at the bottom of the page allows the user to travel to the reviews page to read all the reviews for the restaurant (FR 2). Further, Image 1.2 also contains the add review icon (+) on the bottom left which directs user to a new page where they may add a review for the restaurant.

Non-functional Requirements Implemented: The average rating mentioned in Image 1.2 just below the restaurant name is the average of all ratings of reviews made in the last 2 years (~730 days) (NFR 1). After clicking the + icon to add a review, while providing a rating out of 5 stars, only those stars will get bold (in red) which are selected. We will se more about this in the next section, V-MFP, Image 1.3 (NFR 2). Finally, after adding a review, since it will be the most recent review, it will be shown on the top of the reviews list, since the reviews are sorted by time with the most recent review shown on the top of the list. This can be seen in the Image 1.4 in the V-MFP section (NFR 3).

Vertical MFP

The next step was to create the Vertical MFP for our review mechanism.

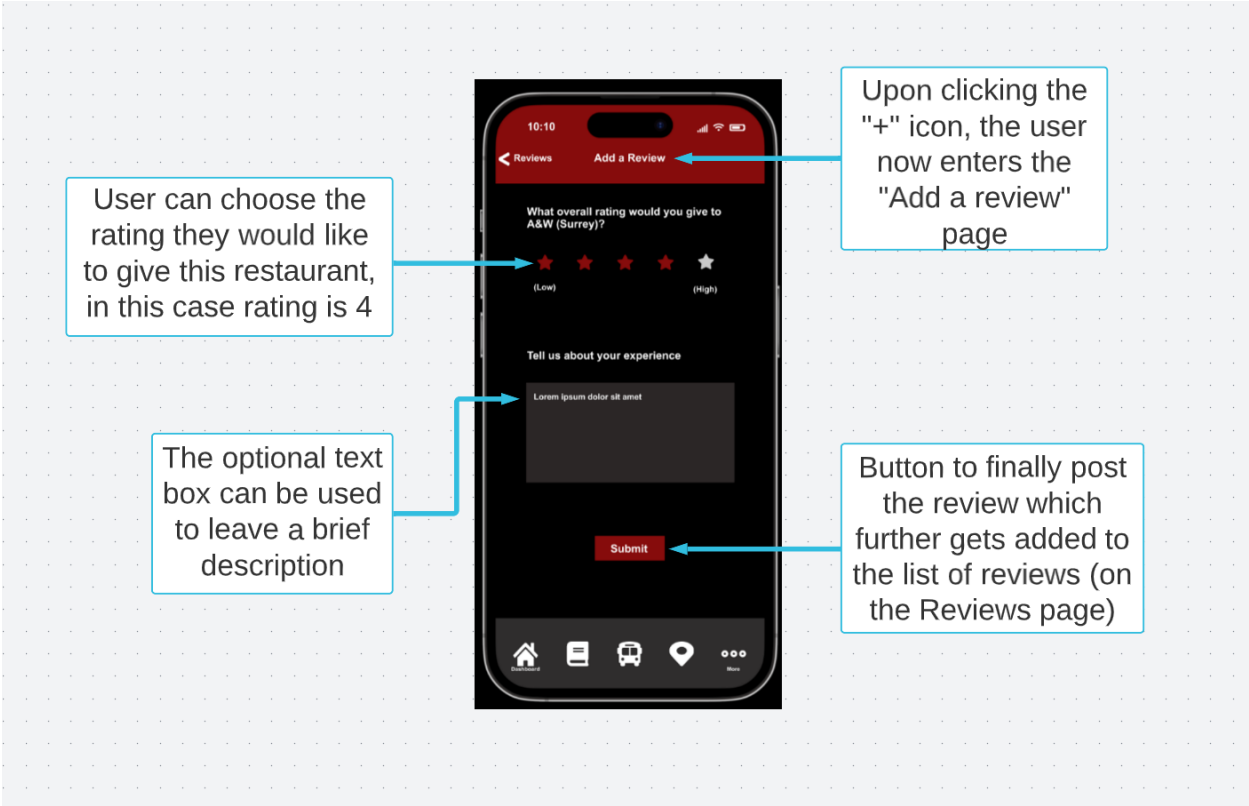

Image 1.3

After combining our functional and non-functional requirements as mentioned in part 2a, we now introduce some features to our review page

Feature I: The “add review” feature allows the user to give a restaurant a rating of their choice and add a brief description of how their experience has been. This satisfies the third functional requirement and ahead also satisfies the second non-functional requirement as well (as shown in image 1.3).

Feature II: Further, the “submit” button feature allows the users to finally post their reviews which can then be publicly read in the SFU snap app by the students and faculty. This satisfies the third non-functional requirement since the most recent review gets pushed to the top place in the list of reviews (as seen in Image 1.4, next page).

Now, we will combine both these features into one single feature to be tested.

Follow the given steps to successfully add a review for the A&W (Surrey) restaurant including the rating (out of 5) and a description.

Upon opening the V-MFP, the user is taken to the SFU Snap App dashboard page.

Dashboard → More (bottom right on the navigation bar) → Dining → A&W (Surrey) → Reviews → “+” icon on the bottom right → Add a rating out of 5 (user preference) → click on the description box to add a text → Submit

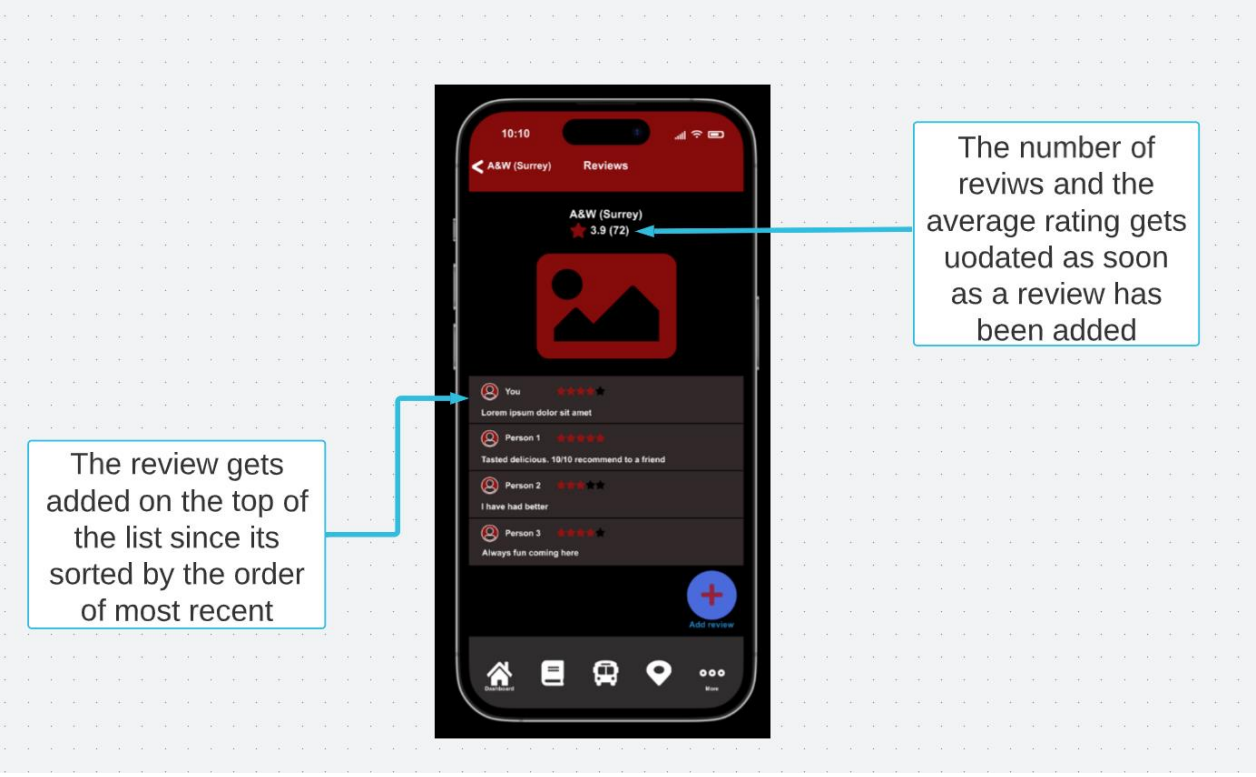

Image 1.4

Summarizing the Part 2b: MFPs

From creating sketches and LFPs on Balsamiq to creating multiple pages on Figma and combining them to make an MFP has been an important part of implementing a new feature – a review mechanism inside the Dining feature in the SFU Snap App.

Using the prototyping tools inside Figma, we have been successful in creating a look-alike of how the review mechanism should be. For instance, implementing the feature to include the list of reviews of past customers for a restaurant, or adding an “add review” feature to let users leave their experience, these features have been easily implemented. Although Figma has been a great help in generating our MFP, there have been some troubles in implementing some advanced features like getting a user input entered in the description box inside the add a review page. Or it has been our fault for not reading the advanced functionalities provided by Figma. Either way, in the end, we figured out a way to enter a text on our end as soon as the user clicks on the text box.

In conclusion, it has been a wonderful learning experience with Figma in generating both horizontal and vertical MFPs.

Part 2c: Analytical Evaluation & Reflection

Component 1: Cognitive Walkthrough

For this part, consider a scenario where SFU students love to go to the gym (potential targets based on our persona) and our conscious about their diet want to increase their protein intake, and want to try protein-rich meals from A&W (Surrey) restaurant.

For representative task 1, consider these potential targets who want to check the reviews of the above restaurant before trying their meat.

Further, for representative task 2, consider that now they want to add a review of their own after trying their meals.

Representative Task 1

Action Sequence	**Does the user know what to do given the action?**	**Can the user find the right interface component to perform this action?**	**Can the user associate the feedback from the interface with the correct action they perform?**	**Does the user understand the feedback so that they know where they are in the task after performing the correct action?**
Action 1: Open more options from the dashboard screen	Yes, the user understood that he had to access more options that were similar to the ones that were present on the navigation bar.	Yes, the user was able to find the button that was named “more” and directed the user to a page containing more options.	Yes, the user was able to associate with the feedback as he was able to see the heading of the page that said “more”, hence confirming the user that he took the correct action.	Yes, the user understood the feedback as he was able to see a list of more options and he saw the heading that said “more” which confirmed that he was on the page that consisted of more options.
Action 2: Proceed to the dining page	Yes, the user understood that now he had to go to the dining page.	Yes, the user saw an option named “dining” that assisted him to find the correct interface component which he used to access the dining page.	Yes, the user was able to associate with the feedback as after clicking the dining option he was directed to a page that had the heading “Dining”.	Yes, the user understood the feedback as he was able to see a list of restaurants and the heading “Dining” so he was sure that now he was on the dining

				page of the application.
Action 3: Open the “A&W (Surrey)” restaurant page	Yes, the user understood that from the list of various restaurants, he had to open the restaurant page for the “A&W (Surrey)”.	Yes, the user was able to find an option in the list of restaurants that said “A&W (Surrey)” which he had to click to open the restaurant page of “A&W (Surrey)”.	Yes, the user was able to associate with the feedback as after clicking on the “A&W (Surrey)” option on the dining page, he was directed to a restaurant page that had the heading “A&W (Surrey)”.	Yes, the user understood the feedback as he was able to see a restaurant page with the heading “A&W (Surrey)” so now he knew that he was on one of the restaurant pages of the app
Action 4: Proceed to checkout the reviews about the “A&W (Surrey) restaurant posted by the other users	Yes, the user understood that he had to access the option on the restaurant page that will direct him to the page that will contain the reviews.	Yes, the user was able to find the right interface component as he was able to locate an option named “Reviews” which he clicked to access the page that contained the reviews.	Yes, the user was able to associate with the feedback as after clicking the “Reviews” option on the restaurant page the user was directed to a page with the heading “Reviews” and it contained the reviews provided by other users in the past.	Yes, the user understood the feedback as he was able to access the “Reviews” page that contained reviews posted by other users of the application and now he knew that he was on the review page of the “A&W (Surrey) restaurant.

Representative Task 2

Action Sequence	**Does the user know what to do given the action?**	**Can the user find the right interface component to perform this action?**	**Can the user associate the feedback from the interface with the correct action they perform?**	**Does the user understand the feedback so that they know where they are in the task after performing the correct action?**
Action 1: From the dashboard	Yes, the user knew of the presence of	Yes, the user quickly referred to the navigation	Yes, the user received the appropriate	Yes, the user had no trouble understanding the

screen, proceed to more options	the “more” options	bar and found the “more” options icon at the bottom right of the dashboard screen.	feedback as the interface signifies “More” on the top of the page.	feedback they received since they are now able to view the extended list of features present in the application.
Action 2: Within the more options, proceed to the page containing a list of restaurants to dine in	Yes, the user is sure of what to do, i.e., locate a feature that would have all the restaurant options.	Yes, the first choice of the users was the Dining feature which was correct as per the intended actions	Yes, after opening the Dining page, the user gets access to all the restaurants by campus which was the ultimate goal	Yes, the user confirms what they see was actually what was initially asked as the action, also they received correct feedback containing the entire list of the restaurants.
Action 3: Open the “A&W (Surrey)” restaurant page	Yes, the user was confident while performing this action as they knew they were assigned to choose a restaurant from the list	Yes, the user was able to locate the restaurant which was present at the top of the list	Yes, the user can easily link the action they performed to the feedback they received since the A&W (Surrey) page is now on display	Yes, since there is a top heading displaying the name of the restaurant which is the same as in the action, they are sure that they received the correct feedback
Action 4: Proceed to take a look at what reviews other customers have for this restaurant	Yes, by reviews the user understood that there should exist a similar feature and they continued looking for it.	Yes, the user was able to find the “Reviews” feature present in this restaurants interface	Yes, the user was able to quickly link the feedback they received with their action as they have successfully opened the “Reviews” page for this restaurant	Yes, the user has now access to the most recent reviews for this restaurant and they can understand the connection between the action and the interface.
Action 5: Further, proceed to add a review of your own.	Yes, the user was aware that now they have to post their review and started looking for a tool/ button to do so	No, not at first. The user was not able to process that the “+” icon is responsible for adding a review. Hence, the user was then guided	Yes, the user was taken to the “add a review” page which was the initial purpose and thus they were able to	Yes, the user understands that they are asked to submit a rating for the restaurant and give a brief description of their experience which

		to the “+” icon for adding a review	associate it with the action	comprises the overall review of the restaurant.
Action 6: Finally, post your review for it to be seen on the reviews page of this restaurant	Yes, after adding their rating and description, the user knew that the final step is to submit the review.	Yes, the user was able to find the submit button present at the bottom of the “add a review” page and understood that clicking this would submit their review once and for all	Yes, the user got the feedback from the interface where they were taken back to the “Reviews” page. However, there was slight confusion about the proof of submission at first	Yes, the user finally understood the feedback where their review was now present at the top of the list and thus, they were able to confirm the action they took provided fruitful feedback

Part 2c – Component 1 - Cognitive Walkthrough Summary

Getting reviews from real users helps the designers in implementing what the users want in a feature and or an application. After conducting a cognitive walkthrough for our vertical MFP, we received both positive and negative feedback for our prototype which would help us in the future while redesigning some functionalities of this feature or while adding a new feature. The following are the strengths, weaknesses, and suggestions for our design

Strengths – Our review feature design of the SFU snap app is comparable to that of other famous reviews mechanism containing applications, like google maps. For instance, our reviews page allows the user to select a rating out of 5 stars and lets them add a few lines about their experience. Further, the evaluator added (and I feel) that the visibility of the most recent reviews on top of the “reviews” page is an advantage for future customers since they would know about recent customer interactions with this restaurant.

Weaknesses – First, the non-obviousness of the “add a review” icon which is present on the “reviews” page. The evaluator was not able to find the “+” icon to add a review. Although, a text stating that this icon allows the users to add a review was present just below the icon, it just wasn’t readable. Secondly, there is no feedback received after submitting a review telling the user that the review has successfully been posted.

Suggestions – Both of our teammates and the evaluator worked on some improvement suggestions for the design at the end. One suggestion was to make your features visible to the users, as in the case of the “+” icon for adding a review. Adding a hovering effect on the icon, or just making the text below it larger are some suggestions. Secondly, letting the users add an image as a part of their review along with rating and text descriptions would give the review a personal touch and would help highlight the review (be it good or bad). This is another good suggestion.

Part 2c: Reflection

Suggestions/Lessons learnt about designing a socializing feature for an existing system:

1. It is the best practice to completely understand the functioning/working of the previously existing system to which the socializing feature is supposed to be added. This helps in the development of the socializing feature by making the designer understand how the newly added feature would go hand in hand or we can say smoothly with the existing system. Without having a complete understanding of the existing system, it would force the designer to make a new feature that would not fit efficiently with the existing system and therefore might affect the usability of the overall interface.

2. Another important suggestion is to make a road map of the process that you as a designer is going to follow while designing the socializing feature. Before starting the process of prototyping the functional and non-functional requirements should be very clear should be the final ones. For instance, while trying to add the review mechanism to the existing dining feature of the SFU Snap app, I and my group member did not finalize the non-functional requirements and started the process of prototyping. This was a huge mistake that we both did, as when we finalized the non-functional requirements and some other actions that the app might support, it was too late. It took us a very long time to modify the medium-fidelity prototype and that's why I think that no one else should make the same mistake.

3. Also, another suggestion is to spend some time thinking about the tasks that the socializing feature would allow the users to perform. The main goal of a socializing feature is to allow the users to socialize with one another. Therefore, a good amount of effort shall be dedicated while deciding the socializing tasks that the feature would offer. The designer should look from different perspectives as that would allow him to decide what all functionalities could be added to the socializing feature that would make that feature helpful for the users by allowing them various options to socialize. Furthermore, the layout of the socializing feature should be consistent with the layout of the existing system as not doing so will depreciate the usability of the entire system. Like, the review feature that our group added to the dining feature is completely consistent with the existing system.

Lessons learned all the way from Part 1 to the Cognitive Walkthrough:

1. **Heuristic Evaluation:** In Part 1 we learnt how to do the heuristic evaluation of a user interface. We did the heuristic evaluation of the dining feature of the SFU Snap app by checking which of Nielsen's ten heuristics were being supported and which ones were being violated. We had to find out two usability problems and two good examples of usability that the dining feature of the application depicted. This process of heuristic evaluation is crucial from the designing point of view as it allows the designers to know about what usability features and practices are lacking in their designed interface. This helps a designer get rid of these usability problems in the design. We also provided solutions/trade-offs that could help to get rid of the usability problems that we found. We learned how to determine the usability problems present in an interface.

2. **Requirements Gathering & Specification:** In Part 1, after finding out the usability problems in the existing interface of the dining feature, we started collecting design requirements that would act as improvements to the existing dining interface that we evaluated. The requirements that we gathered included context identification and user identification. Context identification consisted of factors like when/where/who/what/how the feature in the application was designed. User identification meant who was supposed to be the target audience that would use the feature of the application. We create two personas for the people that might use the dining feature of the SFU Snap app. The requirements also included a crucial thing which was to gather the functional and non-functional requirements of the dining feature. These requirements are very important as the whole process of prototyping and starting the design process of any feature relies on these requirements. This process of requirements gathering, and specification was also carried out in Part 2 of the project when we were designing the review mechanism for the dining feature of the SFU Snap app. We did the exact same steps of extracting the context identification, user identification and then gathering three functional and three non-functional requirements. This whole process of requirements gathering taught us to understand and helped us to learn how could a designer collect the requirements and specifications before he starts making the prototypes and adding functionalities to a system.

3. **Prototyping:** In Part 1 after gathering the requirements, we started creating Low-Fidelity prototypes using Balsamiq. These prototypes satisfied all the functional and non-functional requirements and therefore displayed the improvements that could be done to the previously existing dining feature. As these were low-fidelity prototypes, they were not very functional but still gave a very clear idea about how the final product would look like. Here, the final product meant the dining feature that was rid of all the usability problems that were found in Part 1 of the project using heuristic evaluation.

Prototyping was also used in part 2 of the project but this time we used Figma which provided us with the option to create medium-fidelity prototypes. These prototypes had more functionality and were a bit closer to the final version of the review mechanism in the dining feature. Also, in Part 2 of the project, we were supposed to add a review mechanism to the dining feature, so it was not like Part 1 prototypes where we tried to bring improvements to the dining feature of the SFU Snap app. By building these low-fidelity and medium-fidelity prototypes, we learnt how to go from writing requirements to satisfying them by developing a semi-functional version of the final interface. Without learning how to build prototypes, our requirements gathered in Part 1 and Part 2 would have stayed just requirements and would not have taken the form of a real feature in the user interface.

4. **Cognitive Walkthrough:** In Part 2 of the project, we did a cognitive walkthrough to evaluate our vertical-medium-fidelity prototype. We found a person from our class to simulate the UX Expert, and we informed him about the context and scenario in which our application can be used. We showed him the vertical MFP and asked him to perform two representative tasks, one at a time. We kept on telling him the actions to perform and then asked questions at each action, questions like if he understood what action he had to perform, if he was able to determine which interface component to click, whether he was able to associate with the feedback from the interface after performing the correct action or if he knew which stage of the task he was in, after performing the action. The feedback received by us from the evaluator was being filled in a table side by side. Two tables were made based on each representative task. By doing this cognitive walkthrough, we learnt that this allowed us to find the tweaks or problems that we as a designer might have ignored while designing the interface. It is very important for a designer to know about all these problems so that they could be sorted in future.