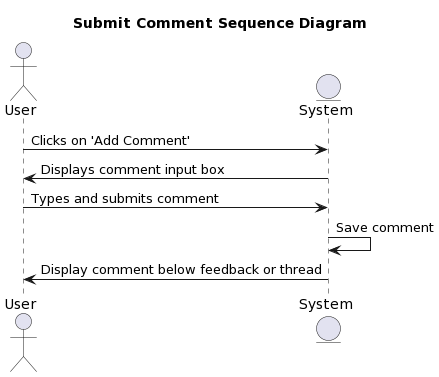
### **Diagram 1: Submit a Comment**

| Step | Actor | System | Description |
| --- | --- | --- | --- |
| 1 | User |  | Decides to comment on a feedback |
| 2 | User |  | Clicks on 'Add Comment' button below feedback or specific thread |
| 3 |  | System | Displays comment input box |
| 4 | User |  | Types in their comment |
| 5 | User |  | Clicks 'Submit Comment' |
| 6 |  | System | Saves the comment |
| 7 |  | System | Displays the comment below the feedback or specific thread |



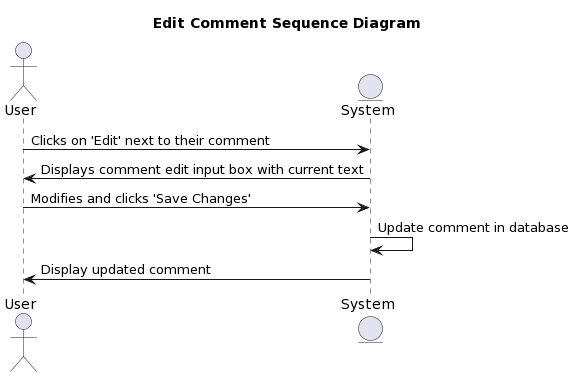
| @startuml  title Submit Comment Sequence Diagram  actor User as U  entity System as S  U -> S: Clicks on 'Add Comment'  S -> U: Displays comment input box  U -> S: Types and submits comment  S -> S: Save comment  S -> U: Display comment below feedback or thread  @enduml |
| --- |

### 

### 

### **Diagram 2: Edit Comment**

| Step | Actor | System | Description |
| --- | --- | --- | --- |
| 1 | User |  | Decides to edit their comment |
| 2 | User |  | Clicks on 'Edit' button next to their comment |
| 3 |  | System | Displays comment edit input box with current comment text |
| 4 | User |  | Modifies the comment text |
| 5 | User |  | Clicks 'Save Changes' |
| 6 |  | System | Updates the comment with the new text |
| 7 |  | System | Displays the updated comment in place of the old one |



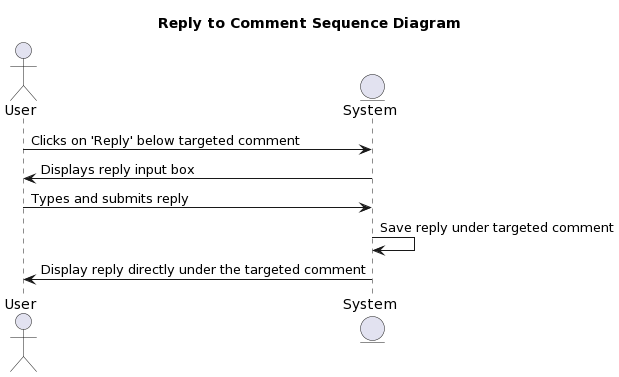
| @startuml  title Edit Comment Sequence Diagram  actor User as U  entity System as S  U -> S: Clicks on 'Edit' next to their comment  S -> U: Displays comment edit input box with current text  U -> S: Modifies and clicks 'Save Changes'  S -> S: Update comment in database  S -> U: Display updated comment  @enduml |
| --- |

### 

### 

### **Diagram 3: Reply to a Comment**

| Step | Actor | System | Description |
| --- | --- | --- | --- |
| 1 | User |  | Decides to reply to a specific comment |
| 2 | User |  | Clicks on 'Reply' button below the targeted comment |
| 3 |  | System | Displays reply input box |
| 4 | User |  | Types in their reply |
| 5 | User |  | Clicks 'Submit Reply' |
| 6 |  | System | Saves the reply under the targeted comment |
| 7 |  | System | Displays the reply directly under the targeted comment |



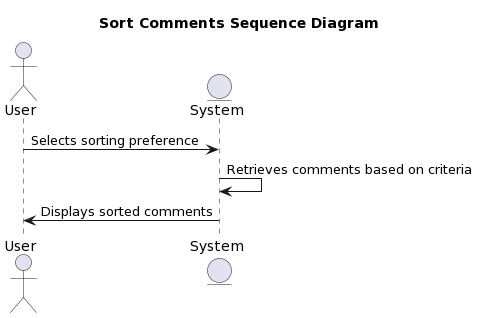
| @startuml  title Reply to Comment Sequence Diagram  actor User as U  entity System as S  U -> S: Clicks on 'Reply' below targeted comment  S -> U: Displays reply input box  U -> S: Types and submits reply  S -> S: Save reply under targeted comment  S -> U: Display reply directly under the targeted comment  @enduml |
| --- |

### 

### 

### **Diagram 4: Sort Comments**

| Step | Actor | System | Description |
| --- | --- | --- | --- |
| 1 | User |  | Decides to sort comments in a specific order |
| 2 | User |  | Selects sorting preference (e.g., newest, oldest, most upvoted) |
| 3 |  | System | Retrieves comments based on selected criteria |
| 4 |  | System | Displays comments in the chosen order |



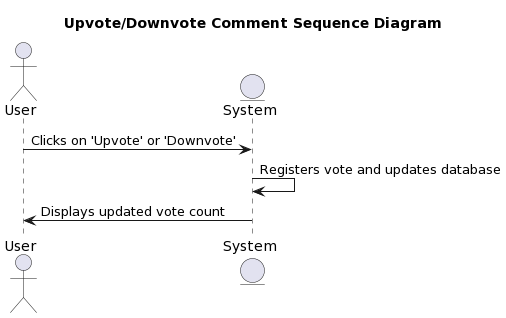
| @startuml  title Sort Comments Sequence Diagram  actor User as U  entity System as S  U -> S: Selects sorting preference  S -> S: Retrieves comments based on criteria  S -> U: Displays sorted comments  @enduml |
| --- |

### 

### 

### **Diagram 5: Upvote/Downvote Comment**

| Step | Actor | System | Description |
| --- | --- | --- | --- |
| 1 | User |  | Decides to rate a specific comment |
| 2 | User |  | Clicks on 'Upvote' or 'Downvote' button next to the comment |
| 3 |  | System | Registers the user's vote choice |
| 4 |  | System | Updates the comment's vote count in the database |
| 5 |  | System | Displays updated vote count next to the comment |



| @startuml  title Upvote/Downvote Comment Sequence Diagram  actor User as U  entity System as S  U -> S: Clicks on 'Upvote' or 'Downvote'  S -> S: Registers vote and updates database  S -> U: Displays updated vote count  @enduml |
| --- |

### 

### 

### 

### 

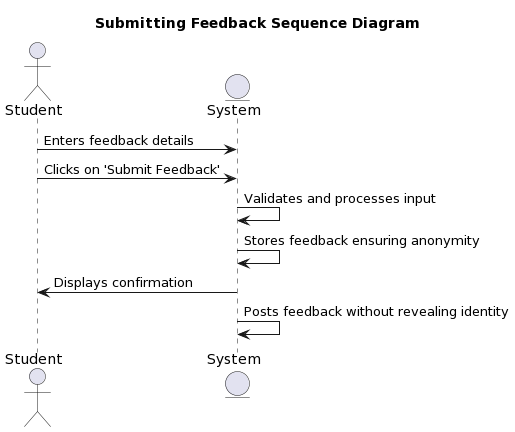
### 

### 

### 

### **Diagram 6: Submitting Feedback**

| Step | Actor | System | Description |
| --- | --- | --- | --- |
| 1 | Student |  | Decides to post feedback about a professor or course |
| 2 | Student |  | Enters feedback details in the provided input fields on "Submit Feedback" page |
| 3 | Student |  | Clicks on 'Submit Feedback' button |
| 4 |  | System | Validates and processes the input |
| 5 |  | System | Stores the feedback in the database ensuring the anonymity of the student |
| 6 |  | System | Displays a confirmation to the student |
| 7 |  | System | Posts the feedback on the platform without revealing the student's identity |

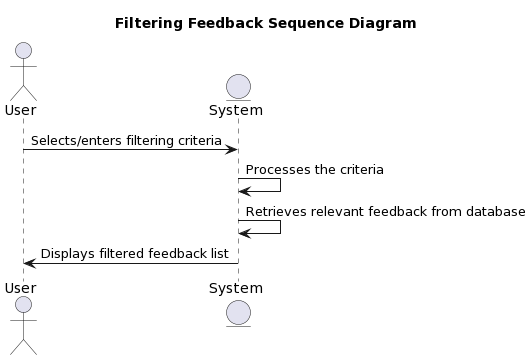


| @startuml  title Submitting Feedback Sequence Diagram  actor Student as S  entity System as Sys  S -> Sys: Enters feedback details  S -> Sys: Clicks on 'Submit Feedback'  Sys -> Sys: Validates and processes input  Sys -> Sys: Stores feedback ensuring anonymity  Sys -> S: Displays confirmation  Sys -> Sys: Posts feedback without revealing identity  @enduml |
| --- |

### 

### **Diagram 7: Filtering Feedback**

| Step | Actor | System | Description |
| --- | --- | --- | --- |
| 1 | User |  | Decides to filter feedback entries |
| 2 | User |  | Selects or enters filtering criteria (e.g., professor name, course code, date) |
| 3 |  | System | Processes the filtering criteria |
| 4 |  | System | Retrieves relevant feedback from the database based on criteria |
| 5 |  | System | Updates the displayed feedback list to show only matching entries |



| @startuml  title Filtering Feedback Sequence Diagram  actor User as U  entity System as S  U -> S: Selects/enters filtering criteria  S -> S: Processes the criteria  S -> S: Retrieves relevant feedback from database  S -> U: Displays filtered feedback list  @enduml |
| --- |

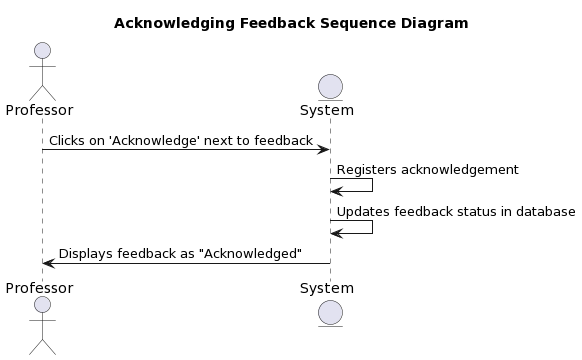
### 

### 

### 

### **Diagram 8: Acknowledging Feedback**

| Step | Actor | System | Description |
| --- | --- | --- | --- |
| 1 | Professor |  | Decides to acknowledge a specific piece of feedback |
| 2 | Professor |  | Clicks on 'Acknowledge' button next to the feedback entry |
| 3 |  | System | Registers the acknowledgement |
| 4 |  | System | Updates the feedback status in the database to "Acknowledged" |
| 5 |  | System | Displays the feedback as "Acknowledged" on the platform for all users |



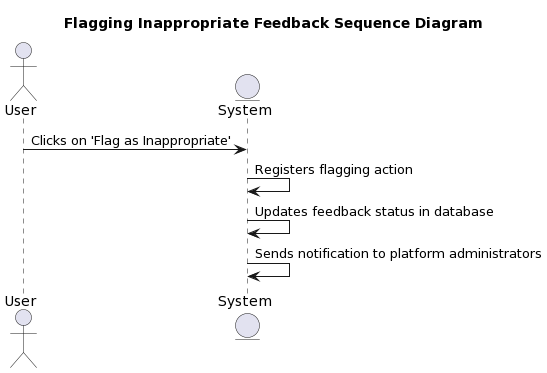
| @startuml  title Acknowledging Feedback Sequence Diagram  actor Professor as P  entity System as S  P -> S: Clicks on 'Acknowledge' next to feedback  S -> S: Registers acknowledgement  S -> S: Updates feedback status in database  S -> P: Displays feedback as "Acknowledged"  @enduml |
| --- |

### 

### 

### **Diagram 9: Flagging Inappropriate Feedback**

| Step | Actor | System | Description |
| --- | --- | --- | --- |
| 1 | User |  | Decides to flag a specific piece of feedback as inappropriate |
| 2 | User |  | Clicks on 'Flag as Inappropriate' button next to the feedback entry |
| 3 |  | System | Registers the flagging action |
| 4 |  | System | Updates the feedback status in the database to "Flagged for Review" |
| 5 |  | System | Sends a notification/alert to platform administrators about the flagged feedback for their review |



| @startuml  title Flagging Inappropriate Feedback Sequence Diagram  actor User as U  entity System as S  U -> S: Clicks on 'Flag as Inappropriate'  S -> S: Registers flagging action  S -> S: Updates feedback status in database  S -> S: Sends notification to platform administrators  @enduml |
| --- |

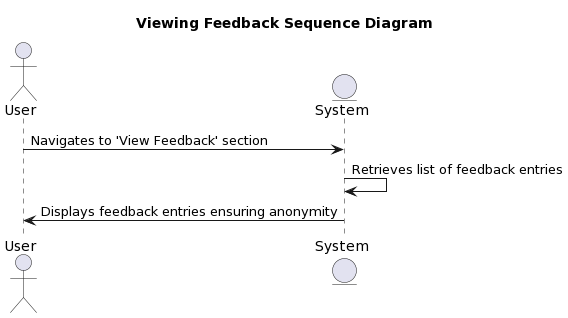
### 

### 

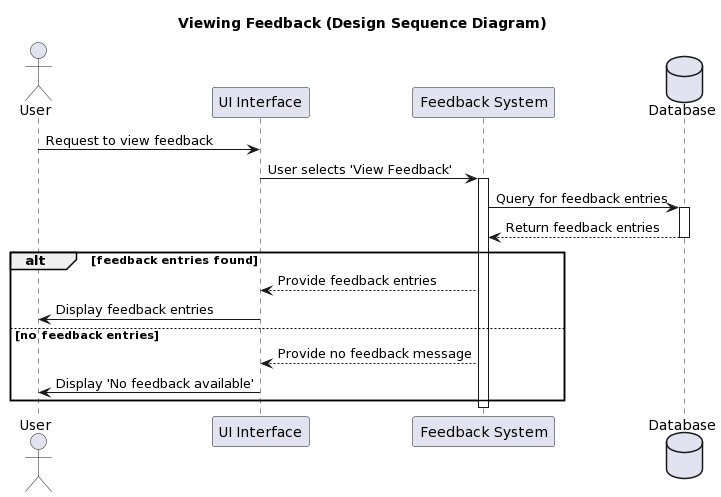
### 

### **Diagram 10: Viewing Feedback**

| Step | Actor | System | Description |
| --- | --- | --- | --- |
| 1 | User |  | Decides to view feedback entries on the platform |
| 2 | User |  | Navigates to the 'View Feedback' section/page |
| 3 |  | System | Retrieves the list of feedback entries from the database |
| 4 |  | System | Displays the feedback entries for the user, ensuring anonymity of posters |



| @startuml  title Viewing Feedback Sequence Diagram  actor User as U  entity System as S  U -> S: Navigates to 'View Feedback' section  S -> S: Retrieves list of feedback entries  S -> U: Displays feedback entries ensuring anonymity  @enduml |
| --- |



Spoiler Alert:

After creating a few of the Design Sequence Diagrams by hand to get the feel for them and a better understanding of how they’re developed, the team used “Generative AI” to create a few extra DSD’s outside of the scope of the requirements of the assignment.

Using the paid version of ChatGPT which allows for the uploading and analysis of images, the basic sequence tables and sequence diagrams were uploaded, with the following prompt:

take the following image, and create a design sequence diagram from it, in PlantUML code.

Be very detailed with the design sequence diagram steps and actors.

Ensure the title is the name of the diagram only, with Design Sequence Diagram, in parentheses

Use a structure with alternative paths and multiple components interacting with each other, as is indicated in the second image provided

the second image, the "edit comments" diagram, is the diagram to use as an example for styling and format purposes, the other image is the diagram to construct in plantUML code.

The resulting code was put into PlantUML, checked for errors, inconsistencies, and formatting issues, fed back into ChatGPT as required, until a finalized diagram proved viable.

Pros:

* quick creation of a diagram in PlantUML, a language many are unfamiliar with
* very fast prototyping of different diagrams, to try out different ideas or pathways for each diagram
* creation of multiple diagrams very quickly

Cons:

* less thought put into the diagrams than if created manually
* easy to overlook mistakes by the generative AI
* less time spent on learning PlantUML code, and UML diagramming in general

Specifically for the sequence diagrams, learning their purpose, and how they can be used throughout the design process, is the bulk of their importance in this class. The group project work is a very realistic and thoughtful way to both teach the material, and for students to learn it, but there are still significant downsides to the “group work” methodology.

In this class, specifically for the diagrams (both sequence and architectural), they don’t feel very important to the overall design process, or for the project itself. The diagrams come across as a box to check, something we may encounter in the industry, but not vital enough to spend the time actually doing.

By using generative AI, we’re able to spend more time on what we feel is important, the areas in which we want to learn more about, and less time on diagrams which are not being used to move this project forward.

Below is a sample interaction between the human, and the Generative AI.

