|  |  |  |
| --- | --- | --- |
| **Use Case** | Delete Workspace | |
| **Scenario** | Delete existing workspace | |
| **Triggering Event** | A workspace admin wants to delete a workspace | |
| **Brief Description** | An admin in a workspace wants to delete the workspace. If there exists other admins, then those admins will be notified. If all admins aggreed to delete the workspace than all boards in the workspace will be closed. | |
| **Actors** | User | |
| **Related Use Case** | Create Workspace, Close Board | |
| **Stakeholders** | User | |
| **Postconditions** | User must be the workspace’s admin. | |
| **Flow of Activities** | **Actor** | **System** |
| 1. User clicks on delete workspace button.  2. Other admins agree to delete workspace. | 1.1 Check database for other admins in the workspace and notify them.  2.1 Close all boards in the workspace. And delete workspace from the database. |
| **Exception Conditions** | * 1. There are no other admins immediately delete the workspace.   2. One or more admin do not aggree to delete the workspace. | |

|  |  |  |
| --- | --- | --- |
| **Use Case** | Close Board | |
| **Scenario** | Close a board | |
| **Triggering Event** | A board admin decides to close the board. | |
| **Brief Description** | An admin in a board decides to close the board. A closed board can still be seen by the board admin and can be reopened in any existing workspace. | |
| **Actors** | User | |
| **Related Use Case** | Delete Workspace, Create Board | |
| **Stakeholders** | User | |
| **Postconditions** | User must be the board’s admin. | |
| **Flow of Activities** | **Actor** | **System** |
| 1. User clicks on the close board menu. | * 1. System Checks if User is admin.   2. System checks if board is already closed.   1.3. Close the board in the database.  1.4 System adds database into the Closed Board List. |
| **Exception Conditions** | 1.1 User is not admin  1.2 Board is already closed. | |

|  |  |  |
| --- | --- | --- |
| **Use Case** | Change Workspace Visibility | |
| **Scenario** | Change visibility for workspace | |
| **Triggering Event** | An admin in the workspace changes the visibility for the workspace. | |
| **Brief Description** | An admin in a workspace decides to change the visibility for that workspace. All boards in the workspace will adjust accordingly. | |
| **Actors** | User | |
| **Related Use Case** | Create Workspace, Change Board Visibility | |
| **Stakeholders** | User | |
| **Postconditions** | User must be the workspace’s admin. | |
| **Flow of Activities** | **Actor** | **System** |
| 1. User clicks on the change visibility menu, and chooses a visibility. | 1.1 The workspace will change its visibility in the database.  1.2 All boards in the database will have it’s visibility changed accordingly in the database.  1.3 All user will have their access changed according to the visibility. |
| **Exception Conditions** | 1. User chooses the same visibility. | |

|  |  |  |
| --- | --- | --- |
| **Use Case** | Specify Due Date and Time | |
| **Scenario** | Specify due date and time for a card. | |
| **Triggering Event** | Member in the board specified a due date for a card in a list. | |
| **Brief Description** | A member in the board decides to specified a certain due date and time for a card in the list. All watchers for the card will be notified if due date is close according to the reminder specified by them. | |
| **Actors** | User | |
| **Related Use Case** | Create Card, | |
| **Stakeholders** | User | |
| **Postconditions** | User must be a member in the board. | |
| **Flow of Activities** | **Actor** | **System** |
| 1. User chooses to specific date and time for a card.  2. User specifies reminder time for the due date.  3. User receives reminder notification. | 1.1 Due Date and Time is added in the database for the card.  2.1 Sets the reminder date and time.  2.2 If date time is equal to reminder date time then notify all watchers for the card.  C |
| **Exception Conditions** | - | |

|  |  |  |
| --- | --- | --- |
| **Use Case** | Add Card Comment | |
| **Scenario** | Add new comment into the card. | |
| **Triggering Event** | Member in the board adds a new comment to a card. | |
| **Brief Description** | A member commented a card in a list. | |
| **Actors** | User | |
| **Related Use Case** | Create Card | |
| **Stakeholders** | User | |
| **Postconditions** | User must be a member in the board. Card must exist. | |
| **Flow of Activities** | **Actor** | **System** |
| 1. User clicks on a card on a list.  2. User input card comment | 1.1 Verify if user has access.  2.1 Add comment to the database.  2.2 Notify all watchers for the card. |
| **Exception Conditions** | 2. If user mentions another user in the comment, system will check if mentioned user has acces, if has access then notify. | |