**Use Cases**

UC1 Create a Profile

UC2 Sign In

UC3 View Profile

UC4 Accessing Another User's Profile

UC5 Add Friend

UC6 Comment on a Post

UC7 Create a Post

UC8 Edit the Visibility of a Post

UC9 Edit a Post

UC10 View Groups

UC11 Create a Group

UC12 Join a Group

UC13 View Schedule

UC14 Create Schedule

UC15 Edit Schedule

UC16 View Resume

UC17 Upload Resume

UC18 Delete Resume

UC19 Check Lost and Found

UC20 View My Lost and Found Items

UC21 Add Item to Lost and Found

UC22 Remove Item from Lost and Found

UC23 View Polls

UC24 Create Poll

UC25 Vote on Poll

**UC1 Flow of Events for the *Create a Profile* Use Case**

**1.1 Preconditions:**

None.

**1.2 Main Flow:**

The user clicks the “Sign-up” button from the main page and is presented with a form in which they must enter their username, password, email [E1], student ID, campus, and gender [E2]. The user will then be required to access their email and click the confirmation link from the confirmation email.

**1.3 Subflows:**

None.

**1.4 Alternative Flows:**

[E1] The email that is entered must be a valid @mun.ca email address. If the email does not end with @mun.ca an error is given.

[E2] All the fields must be filled out. If any fields are empty, an error is given.

**UC2 Flow of Events for the *Sign In* Use Case**

**2.1 Preconditions:**

None.

**2.2 Main Flow:**

After clicking the “Sign-In” button, the user is prompted to enter their account information. Once the user enters their account name and password and the “Submit” button is clicked [E1] [E2], the user will be brought to their profile.

**2.3 Subflows:**

None.

**2.4 Alternative Flows:**

[E1] All fields must be filled out. If any fields are left blank, an error is given.

[E2] Must be a valid username/password combination. If they information is not valid, an error is given.

**UC3 Flow of Events for the *View Profile* Use Case**

**3.1 Preconditions:**

1. The user is signed in [UC2].

**3.2 Main Flow:**

Upon creating an account [UC1] or signing in [UC2], the user is brought to their profile. The user may also click their name/profile picture from the top of the page to be brought to their profile [E1].

**3.3 Subflows:**

None.

**3.4 Alternative Flows:**

[E1] A user is automatically signed out after 20 minutes of inactivity. If the user attempts to view their profile after being logged out, they will be prompted to sign-in [UC1].

**UC4 Flow of Events for the *Accessing Another User's Profile* Use Case**

**4.1 Preconditions:**

The user is signed in [UC2].

**4.2 Main Flow:**

The user clicks another user's name to view their profile [S1][S2][S3][S4][S5].

**4.3 Subflows:**

[S1] The user enters a name in the search bar and clicks “Search” which brings up a list of users based on the name they entered.

[S2] After accessing their own profile [UC3], the user clicks on a friend from their suggested friend's list and is brought to their profile.

[S3] After accessing their own profile [UC3], the user clicks on a user in a friend's friend list and they are brought to the users profile.

[S4] The user clicks a friend request from their notification area and can click their name to view their profile.

[S5] After accessing their own profile [UC3] or another user's profile [UC4], the user clicks on the name of someone who has posted on their newsfeed and is brought to their profile.

**4.4 Alternative Flows:**

None.

**UC5 Flow of Events for the *Add Friend* Use Case**

**5.1 Preconditions:**

The user is signed in [UC2].

**5.2 Main Flow:**

After the user accesses another user's profile [UC4], they click the “Add Friend” button to send that user a friend request, or they can accept a friend request from their notification area [S1].

**5.3 Subflows:**

[S1] The user clicks on a friend request notification and clicks the “Accept” button.

**5.4 Alternative Flows:**

None.

**UC6 Flow of Events for the *Comment on a Post* Use Case**

**6.1 Preconditions:**

1. The user is signed in [UC2]

2. The user is viewing another user's profile [UC4]

**6.2 Main Flow:**

After the user accesses another user's profile [UC4], they click on the “Add Comment” button under a post on the other user's timeline[E1]. The user is prompted to enter text [E2], pictures [E3], hyperlinks, or other text-based information.

**6.3 Subflows:**

None.

**6.4 Alternative Flows:**

[E1] The post cannot be empty. If the post is empty an error given.

[E2] The post cannot be longer than 1000 characters. If the post exceeds the character limit an error is given.

[E3] Only .jpeg, .png, .gif files can be uploaded up to the size of 50MB. If the image is an invalid type or too large an error is given.

**UC7 Flow of Events for the *Create Post* Use Case**

**7.1 Preconditions:**

1. The user is signed in [UC2].

2. The user is viewing their profile [UC3].

**7.2 Main Flow:**

After clicking “Create Post” on their newsfeed, the user is prompted to enter text [E2], pictures [E3], hyperlinks, or other text-based information. The user may select the visibility settings of their post [S1]. The user then clicks “Post” and their post is displayed on their newsfeed [E1].

**7.3 Subflows:**

[S1] The user can choose to set the visibility of their post to something other than their default setting. If they choose to select a list of friends, their friends list is displayed and they can check a box next to the friend that they wish to allow visibility.

**7.4 Alternative Flows:**

[E1] The post cannot be empty. If the post is empty an error given.

[E2] The post cannot be longer than 1000 characters. If the post exceeds the character limit an error is given.

[E3] Only .jpeg, .png, .gif files can be uploaded up to the size of 50MB. If the image is an invalid type or too large an error is given.

**UC8 Flow of Events for the *Edit Visibility of a Post* Use Case**

**8.1 Preconditions:**

1. The user is signed in [UC2].

2. The user is viewing their own profile [UC3].

**8.2 Main Flow:**

After finding a post they wish to edit the visibility of, the user clicks the “Edit Visibility” button and select how they want to change it [S1][S2][S3][S4].

**8.3 Subflows:**

[S1] The user selects “Everyone,” allowing anyone to see the post.

[S2] The user selects “Friends Only,” allowing only users on their friends list to view the post.

[S3] The user selects “User Only,” allowing only themselves to view the post.

[S4] The user selects “Only Certain Friends,” and is prompted to select friends from a list to allow to view the post.

**8.4 Alternative Flows:**

None.

**UC9 Flow of Events for the *Edit a Post* Use Case**

**9.1 Preconditions:**

1. The user is signed in [UC2].

2. The user is viewing their own profile [UC3].

**9.2 Main Flow:**

After finding a post they wish to edit, they click the “Edit Post” button and select how they want to change it [E1][E2].

**9.3 Subflows:**

None.

**9.4 Alternative Flows:**

[E1] The post must meet the requirements for a post [UC7]. If it does not an error is given.

[E2] The user may cancel the edit and leave the post the same.

**UC10 Flow of Events for the *View Groups* Use Case**

**10.1 Preconditions:**

1. The user is signed in [UC2].

**10.2 Main Flow:**

The user clicks the “Groups” button from the top of the page and is taken to the list of groups.

**10.3 Subflows:**

None.

**10.4 Alternative Flows:**

None.

**UC11 Flow of Events for the *Create a Group* Use Case**

**11.1 Preconditions:**

1. The user is signed in [UC2].

2. The user is on the Groups page [UC10].

**11.2 Main Flow:**

After navigating to the groups page, the user clicks “New Group.” The user enters the name and description of the group, sets the privacy settings, and clicks “Create Group.” [E1] The user then has the option to invite friends.

**11.3 Subflows:**

**11.4 Alternative Flows:**

[E1] The group must have a name and description. If either fields are blank, an error is given.

**UC12 Flow of Events for the *Join a Group* Use Case**

**12.1 Preconditions:**

1. The user is signed in [UC2].

**12.2 Main Flow:**

After finding a group they wish to join, they click “Join Group.” [S1][S2][S3]

**12.3 Subflows:**

[S1] The user finds a public group from the groups page [UC10][E1].

[S2] The user recieves an invite from a friend in their notifications tab and clicks “Accept Invite.”

[S3] The user requests an invite to a private group and a message is set to the group owner. Once approved by the owner, the user is placed in the group.

**12.4 Alternative Flows:**

[E1] If the user tries to join a private group without an invite, an error is given.

**UC13 Flow of Events for the *View Schedule* Use Case**

**13.1 Preconditions:**

1. The user is signed in [UC2].

**13.2 Main Flow:**

The user clicks “Schedule” on the top of the page and is brought to their schedule page.

**13.3 Subflows:**

None.

**13.4 Alternative Flows:**

None.

**UC14 Flow of Events for the *Create Schedule* Use Case**

**14.1 Preconditions:**

1. The user is signed in [UC2].

2. The user is on the Schedule page [UC13].

**14.2 Main Flow:**

After navigating to the Schedule page, the user clicks “Create Schedule.” [E1] The user is brought to a new screen which prompts them to add courses and set the times [E2].

**14.3 Subflows:**

None.

**14.4 Alternative Flows:**

[E1] Users can only have one schedule. If a schedule already exists, the “Create Schedule” button is replaced with the “Edit Schedule” button.

[E2] The user may not have two courses at the same time. If a conflict exists an error is given.

**UC15 Flow of Events for the *Edit Schedule* Use Case**

**15.1 Preconditions:**

1. The user is signed in [UC2].

2. The user is on the Schedule page [UC13].

**15.2 Main Flow:**

After navigating to the Schedule page, the user clicks “Edit Schedule.” [E1] The user is brought to a new screen which prompts them to add or remove courses and set the times and clicks “Create Schedule [E2].

**15.3 Subflows:**

None.

**15.4 Alternative Flows:**

[E1] If the user does not have a schedule, the “Edit Schedule” button is replaced with the “Create Schedule” button.

[E2] The user may not have two courses at the same time. If a conflict exists, an error is given.

**UC16 Flow of Events for the *View Resume* Use Case**

**16.1 Preconditions:**

1. The user is signed in [UC2].

2. The user is viewing their profile [UC3].

**16.2 Main Flow:**

The user clicks “View Resume” from their profile page.

**16.3 Subflows:**

None.

**16.4 Alternative Flows:**

None.

**UC17 Flow of Events for the *Upload Resume* Use Case**

**17.1 Preconditions:**

1. The user is signed in [UC2].

2. The user is viewing their resume page [UC16].

**17.2 Main Flow:**

From their profile, the user clicks the “Upload Resume” button and is prompted to select a file to upload [E1].

**17.3 Subflows:**

None.

**17.4 Alternative Flows:**

[E1] Only files of the type .pdf, .docx, and .txt can be upload. If another file type is selected an error is given.

**UC18 Flow of Events for the *Delete Resume* Use Case**

**18.1 Preconditions:**

1. The user is signed in [UC2].

2. The user is viewing their resume page [UC16].

3. The user has uploaded a resume [UC17].

**18.2 Main Flow:**

The user clicks the “Delete Resume” below the file they wish to remove.

**18.3 Subflows:**

None.

**18.4 Alternative Flows:**

None.

**UC19 Flow of Events for the *Check Lost and Found* Use Case**

**19.1 Preconditions:**

None.

**19.2 Main Flow:**

The user clicks “Lost and Found” from the top of the page.

**19.3 Subflows:**

None.

**19.4 Alternative Flows:**

None.

**UC20 Flow of Events for the *View My Lost and Found Items* Use Case**

**20.1 Preconditions:**

1. The user is signed in [UC2].

2. The user is viewing the lost and found [UC19].

**20.2 Main Flow:**

The user clicks the “My Items” button on the top of the lost and found page.

**20.3 Subflows:**

None.

**20.4 Alternative Flows:**

**UC21 Flow of Events for the *Add Item to Lost and Found* Use Case**

**21.1 Preconditions:**

1. The user is signed in [UC2].

2. The user is viewing their lost and found items [UC20].

**21.2 Main Flow:**

The user clicks the “Add item” button and enters a description of the object, the area where it was found, and a photo (optional) [E1].

**21.3 Subflows:**

None.

**21.4 Alternative Flows:**

[E1] A description and location are required. If either field is blank, an error is given.

**UC22 Flow of Events for the *Remove Item from Lost and Found* Use Case**

**22.1 Preconditions:**

1. The user is signed in [UC2].

2. The user is viewing their lost and found items [UC20].

**22.2 Main Flow:**

The user clicks the “My Items” button on the top of the lost and found page and then clicks the “Remove Item” button next to the item they wish to remove.

**22.3 Subflows:**

None.

**22.4 Alternative Flows:**

None.

**UC23 Flow of Events for the *View Polls* Use Case**

**23.1 Preconditions:**

1. The user is signed in [UC2].

**23.2 Main Flow:**

The user clicks “Polls” from the sidebar and is brought to the list of polls.

**23.3 Subflows:**

None.

**23.4 Alternative Flows:**

None.

**UC24 Flow of Events for the *Create Poll* Use Case**

**24.1 Preconditions:**

1. The user is signed in [UC2].

2. The user is on the polls page [UC23].

**24.2 Main Flow:**

The user clicks “Create Poll” from the top of the page and enters the poll information (name, voting options, course the poll is about, and who is allowed to vote) [E1][E2].

**24.3 Subflows:**

None.

**24.4 Alternative Flows:**

[E1] The poll must have a name and at least one voting field. If field are blank, an error is given.

[E2] The user must have the course the poll is about on their schedule. If it is not an error is given.

**UC25 Flow of Events for the *Vote on a Poll* Use Case**

**25.1 Preconditions:**

1. The user is signed in [UC2].

2. The user is on the polls page [UC23].

**25.2 Main Flow:**

The user clicks the “Vote” button under the poll they wish to vote on [E2]. The user is brought to a new page that prompts them to enter answers to the poll questions [E2].

**25.3 Subflows:**

None.

**25.4 Alternative Flows:**

[E1] The user must have the course the poll is about on their schedule. If it is not, an error is given.

[E2] The user must fill out all the fields of the poll. If any are left blank, an error is given.