|  |
| --- |
| **Use Case: Sign Up User** |
| ID: 1 |
| Brief Description:  A new user signs up for an account. |
| Primary Actors:  New User |
| Secondary Actors:  None |
| Preconditions:   1. The new user wants to sign up for an account. 2. The new user is not logged on to the system. |
| Main flow:   1. The use case begins when the new user clicks “Sign Up” button. 2. The system asks the user to enter phone number, email address, username, and password. 3. While the username and email does not exist in the system,    1. The system asks the user to enter username and password.    2. The user enters username and password in the system.    3. The system validates the information provided by the user. 4. The system records the information provided by the user. |
| Postconditions:   1. The system accepted and saved the user details. 2. The user gets a new username and password. 3. The system redirects the user to profile sign up page. |
| Alternative flows:  Invalid email address  Invalid password  Invalid username  Invalid phone number |

|  |
| --- |
| **Alternative flow : Sign Up User : Invalid Email Address** |
| ID: 1.1 |
| Actors:  New Users |
| Alternative flow:   1. The alternative flow begins after step 2 of the main flow. 2. If the user enters the email address which is already present in the system,    1. The system rejects the email address and informs user to enter another email address. 3. If the user does not enter email address,    1. The system displays the message and asks the user to enter email address. 4. If the email address does not contain ‘@’,    1. The system displays the message and informs user to include ‘@’ in the email address. 5. The system is unable to create a new account or sign up the user. |
| Postconditions:   1. The system does not save the user details. 2. The user sign up fails. |

|  |
| --- |
| **Alternative flow : Sign Up User : Invalid Username** |
| ID: 1.2 |
| Actors:  New Users |
| Alternative flow:   1. The alternative flow begins after step 2 of the main flow. 2. If the user enters the username which is already present in the system,    1. The system rejects the username and asks user to enter another username. 3. If the user does not enter the username,    1. The system asks the user to enter the username and displays the message. 4. If the user enters the username which is less than 5 characters long,    1. The system asks the user to enter the username which is at least 5 characters long. |
| Postconditions:   1. The system does not save the user details. 2. The user sign up fails. |

|  |
| --- |
| **Alternative flow : Sign Up User : Invalid Password** |
| ID: 1.3 |
| Actors:  New Users |
| Alternative flow:   1. The alternative flow begins after step 2 of the main flow. 2. If the user does not enter the password,    1. The system asks the user to enter the password. 3. If the user enters the password which is less than 8 characters long,    1. The system asks the user to enter the password which is at least 8 characters long. |
| Postconditions:   1. The system does not save the user details. 2. The user sign up fails. |

|  |
| --- |
| **Alternative flow : Sign Up User : Invalid Phone Number** |
| ID: 1.4 |
| Actors:  New Users |
| Alternative flow:   1. The alternative flow begins after step 2 of the main flow. 2. If the user does not enter the phone number,    1. The system asks the user to enter the phone number and displays a message. 3. If the user enters the phone number which is less than 10 characters long,    1. The system asks the user to enter phone number which is at least 10 characters long. |
| Postconditions:   1. The system does not save the user details. 2. The user sign up fails. |

|  |
| --- |
| **Use Case: Login User** |
| ID: 2 |
| Brief Description:  A registered user logs in to the system. |
| Primary Actors:  Registered Users |
| Secondary Actors:  None |
| Preconditions:  The user is in the login page. |
| Main flow:   1. The use case begins when the user enters the username and password. 2. While the login credentials are valid,    1. The system checks the role of the user who attempted to login.       1. If the role of the user is ‘user’,          1. The system redirects the user to the profile page.       2. If the role of the user is ‘admin’,          1. The system redirects the user to the admin page. 3. The user successfully logs into the system. |
| Postconditions:   1. The user successfully logs into the system. |
| Alternative flows:  1. a. The system redirects the user to “access denied page” because the login credentials are invalid.  1. b. The user is unable to login to the system. |

|  |
| --- |
| **Abstract Use Case: Register Profile** |
| ID: 3 |
| Brief Description:  A new user registers the profile in the system. |
| Primary Actors:  New Users. |
| Secondary Actors:  None |
| Preconditions:   1. The user is able to sign up an account. 2. The user has obtained a username and a password. |
| Main flow:   1. The main flow begins when a user registers a new profile in the system by providing address, education and personal details. 2. The registers the profile by providing following information:  * The address details include information about country, state, city, and zip code. * The personal details include information about first name, last name, gender, age, weight, occupation, salary, religion. * The education details include information about education, field of study, name of school, street address, state, and city.  1. While the user provides first name, last name, gender, country, state, and zip code,    1. The system asks the user to enter first name, last name, gender, country, state, and zip code. 2. The user uploads the image in the system to create a profile picture. 3. If the profile picture already exists,    1. The user replaces the existing profile picture by uploading another picture in the system. 4. The system saves the information that the user provided. 5. The user successfully registers a new profile that includes personal details, address details, and education details (optional) along a profile picture (optional). 6. The system sends email to the user along with an activation link to activate the account. |
| Postconditions:   1. The user activates the profile by opening the link in his email. 2. The user successfully registers an account. |
| Alternative flows:  2.a. The system displays the profile sign up page along with the message which informs the user that mandatory information which includes first name, last name, gender, country, state, and zip code to register the profile are missing.  2.b. The system does not save the details which the user provided.  2.c. The user is unable to register a new profile. |

|  |
| --- |
| **Use Case: Register Address Details** |
| ID: 4 |
| Brief Description:  *A new user registers the address details in the system.* |
| Primary Actors:  New Users. |
| Secondary Actors:  None |
| Preconditions:   1. The user is able to sign up an account. 2. The user has obtained a username and a password. |
| Main flow:   1. *The main flow begins when a user registers a new profile in the system by providing address details that includes the information about*  * *Country* * *State* * *City* * *Zip code*  1. While the user provides first name, last name, gender, country, state, and zip code,    1. The system asks the user to enter first name, last name, gender, country, state, and zip code. 2. The user uploads the image in the system to create a profile picture. 3. If the profile picture already exists,    1. The user replaces the existing profile picture by uploading another picture in the system. 4. The system saves the information that the user provided. 5. *The user successfully registers address details in the system.* |
| Postconditions:  *The user registers address details in the system.* |
| Alternative flows:  1.a. The system displays the profile sign up page along with the message which informs the user that mandatory information which includes first name, last name, gender, country, state, and zip code to register the profile are missing.  *1.b. The system does not save the address details which the user provided.*  1.c. The user is unable to register a new profile. |

|  |
| --- |
| **Use Case: Register Personal Details** |
| ID: 5 |
| Brief Description:  *A new user registers the personal details in the system.* |
| Primary Actors:  New Users. |
| Secondary Actors:  None |
| Preconditions:   1. The user is able to sign up an account. 2. The user has obtained a username and a password. |
| Main flow:   1. *The main flow begins when a user registers a new profile in the system by providing personal details that includes the information about:*  * *First name* * *Last name* * *Gender* * *Age* * *Weight* * *Occupation* * *Salary* * *Religion*  1. While the user provides first name, last name, gender, country, state, and zip code,    1. The system asks the user to enter first name, last name, gender, country, state, and zip code. 2. The user uploads the image in the system to create a profile picture. 3. If the profile picture already exists,    1. The user replaces the existing profile picture by uploading another picture in the system. 4. The system saves the information that the user provided. 5. *The user successfully registers personal details in the system.* |
| Postconditions:  *The user registers personal details in the system.* |
| Alternative flows:  1.a. The system displays the profile sign up page along with the message which informs the user that mandatory information which includes first name, last name, gender, country, state, and zip code to register the profile are missing.  *1.b The system does not save the personal details which the user provided.*  1.c The user is unable to register a new profile. |
| **Use Case: Register Education Details** |
| ID: 6 |
| Brief Description:  *A new user registers the education details in the system.* |
| Primary Actors:  New Users. |
| Secondary Actors:  None |
| Preconditions:   1. The user is able to sign up an account. 2. The user has obtained a username and a password. |
| Main flow:   1. *The main flow begins when a user registers a new profile in the system by providing education details that includes the information about:*  * *Education* * *Field of study* * *Name of school* * *Street address* * *State* * *City*  1. While the user provides first name, last name, gender, country, state, and zip code,    1. The system asks the user to enter first name, last name, gender, country, state, and zip code. 2. The user uploads the image in the system to create a profile picture. 3. If the profile picture already exists,    1. The user replaces the existing profile picture by uploading another picture in the system. 4. The system saves the information that the user provided. 5. *The user successfully registers education details in the system.* |
| Postconditions:  *The user registers education details in the system.* |
| Alternative flows:  None |

|  |
| --- |
| **Use Case: Send Text** |
| ID: 7 |
| Brief Description:  One registered user who is not in the friend list of another registered user sends a text message to communicate. |
| Primary Actors:  Registered Users. |
| Secondary Actors:  SMS Providers. |
| Preconditions:   1. The user is logged in to the system. 2. The user searches another registered user in the system. 3. The searched user is not present in the friend list of the user. 4. The user clicks “Send Text” button. |
| Main flow:   1. The main flow starts when the user sends text message to the searched user. 2. The user sends the phone number (optional) and message to the searched user. 3. The system delivers the message to the searched user. 4. The system provides the confirmation message to the user that the text message has been sent. |
| Postconditions:   1. The message has been sent to the searched user. 2. The user is able to see a confirmation message that informs the user that message has been sent. |
| Alternative flows:  1.a. The system displays the message to the user that says the text message cannot be sent. |

|  |
| --- |
| **Use Case: Contact Admin** |
| ID: 8 |
| Brief Description:  Any user can contact admin to ask questions or provide suggestions by sending an email which involves submitting “Contact Us” form located in the main page. |
| Primary Actors:  Users |
| Secondary Actors:  Email Providers |
| Preconditions:   1. The user is currently in the main page. 2. The user fills “Contact Us” form by entering name, email, and message and clicks “Submit” button. |
| Main flow:   1. The main flow starts when the user sends a message to the admin by submitting “Contact Us” form. 2. While the user enters name, email address, and message,    1. The system asks the user to enter name, email, address, and message 3. The user is able to send message to the admin. |
| Postconditions:   1. The user is able to send message to the admin. 2. The system displays confirmation page that informs the user that the email has been sent. |
| Alternative flows:  1.a. The system is unable to send email to admin because the user did not enter all the required information which includes name, email, address, and message.  2.a. The system cannot send the email to admin because there was problem with the email providers.  3. The system informs the user by displaying error message that says the message could not be delivered. |

|  |
| --- |
| **Use Case: Manage Profile Picture** |
| ID: 9 |
| Brief Description:  The user can manage profile picture by uploading a new profile picture, replacing the existing picture with the new one, and deleting the existing picture. |
| Primary Actors:  Registered User |
| Secondary Actors:  None. |
| Preconditions:  The user must be logged in to the system. |
| Main flow:   1. The main flow starts when the user decides to upload, edit, or delete the profile picture. 2. If the user does not have a profile picture,    1. The user can click “Add” button to upload a new profile picture 3. If the user already has a profile picture and is present in the profile page,    1. If the user wants to replace it with new profile picture, the user clicks “Edit” button and uploads a new profile picture    2. If the user wants to delete the picture, the user clicks “Delete” button. 4. The system completes the user request. |
| Postconditions:  The system adds, replaces, or removes the profile picture from the user profile page. |
| Alternative flows:  4.a. The user is unable to manage profile picture because there was a problem in the system. |

|  |
| --- |
| **Use Case: Post Status** |
| ID: 9 |
| Brief Description:  The user can post status in his profile page, or he can post status on his friend’s profile page. |
| Primary Actors:  Registered User |
| Secondary Actors:  Friend |
| Preconditions:  The user must be logged in to the system. |
| Main flow:   1. The main flow starts when the user writes status on status field and click “Submit” button. 2. If the user posts status in his profile picture,    1. The status is visible to the user in his profile page.    2. The status is visible to the user’s friends when they visit his profile page.    3. The status is not visible to the registered users who are not present in the user’s friends list. 3. If the user visits his friend’s profile page and posts status on the wall,    1. The status is visible to the particular friend along with his other friends, and the user.    2. The status is not visible to other registered users who are not present in his friend’s friends list. 4. The system posts the status on the profile page. |
| Postconditions:  The system posts the status on the profile page. |
| Alternative flows:  4. a The user is unable to post the status because there was a problem in the system. |

|  |
| --- |
| **Use Case: Search User** |
| ID: 10 |
| Brief Description:  One registered user can search another registered user in the system by entering his username. |
| Primary Actors:  Registered User |
| Secondary Actors:  None |
| Preconditions:   1. The user must be logged in to the system. 2. The person that the user is searching must be registered to the system. |
| Main flow:   1. The main flow starts when one registered user tries to find another registered user by searching the username in the system. 2. The user enters user name in the search bar which is located at the top of profile page and click “GO” 3. If the searched user is not present in his friend list,    1. The user is able to see only “Add Friend” and “Send Text” button. 4. The system displays the profile page of user with that particular name. |
| Postconditions:  The user is able to see the profile page of the interested user. |
| Alternative flows:  2. a The system displays a message to the user that says the user with the particular username is not present in the system. |

|  |
| --- |
| **Use Case: Like Profile Picture** |
| ID: 11 |
| Brief Description:  The registered user can like the profile picture of his friends by clicking like symbol in their profile page. |
| Primary Actors:  Registered User |
| Secondary Actors:  Friends |
| Preconditions:  The user must be logged in to the system. |
| Main flow:  include (Search User)  The user is able to see the profile page of the interested user.  The user likes the profile page by clicking the like symbol below the profile picture. |
| Postconditions:  The number of likes increases by one.  The color of like symbol changes to deep blue. |
| Alternative flows:  The user is unable to see any changes when he likes his profile picture. |

|  |
| --- |
| **Use Case: View Likes** |
| ID: 12 |
| Brief Description:  The user can see who liked his profile picture or his friend’s profile picture by visiting his wall along with the total number of likes. |
| Primary Actors:  Registered User |
| Secondary Actors:  Friends |
| Preconditions:   1. The user must be logged in to the system. 2. The friend of user must be registered to the system and present in the user’s friends list. |
| Main flow:   1. If the user wants to view likes in his friend’s profile picture,    1. include (Search User)    2. The user is able to see the profile page of the interested user. 2. The user clicks the total number of likes that is present below the profile picture. 3. The user is able to see the username of friends who liked the profile picture. |
| Postconditions:  The user is able to see the username of friends who liked the profile picture. |
| Alternative flows:  None |

|  |
| --- |
| **Use Case: Send Friend Request** |
| ID: 13 |
| Brief Description:  One registered user can send friend request to another registered user if he is not present in his friend list. |
| Primary Actors:  Registered User |
| Secondary Actors:  None |
| Preconditions:   1. The user must be logged in to the system. |
| Main flow:   1. Include (Search User) 2. The user is able to see the profile page of the interested user. 3. The user clicks “Add Friend” button to send friend request. 4. The system sends the friend request to the searched user. |
| Postconditions:  The user is able to send the friend request to another registered user who is not in his friend list. |
| Alternative flows:  None |

|  |
| --- |
| **Use Case: Cancel Friend Request** |
| ID: 14 |
| Brief Description:  The registered user can cancel the friend request that was sent to another registered user who is not present in his friend list. |
| Primary Actors:  Registered User |
| Secondary Actors:  None |
| Preconditions:   1. The user must be logged in to the system. 2. The interested user is registered to the system but is not present in the user’s friend list. 3. The user has already sent the friend request to the interested user. |
| Main flow:   1. include (Search User) 2. The user is able to see the profile page of the interested user. 3. The user clicks “Cancel Friend Request” button to cancel the friend request. 4. The system cancels the friend request that was sent before. |
| Postconditions:   1. The system cancels the friend request, and the interested user is unable to see the friend request sent by the user. 2. The “Add Friend” button replaces the “Cancel Friend Request” button. |
| Alternative flows:  None |

|  |
| --- |
| **Use Case: Accept Friend Request** |
| ID: 14 |
| Brief Description:  One registered user can accept the friend request which was sent by another registered user who is not present in his friend list. |
| Primary Actors:  Registered User |
| Secondary Actors:  None |
| Preconditions:   1. The user must be logged in to the system. 2. The user must receive the friend request from another user. |
| Main flow:   1. The user can see total number of friend requests near the red user icon in his profile page. 2. The user clicks the red user icon in the profile page 3. For each user who sent the friend request,    1. The system displays his username along with “accept” and “reject” button. 4. The user clicks “accept” button to accept particular friend request. |
| Postconditions:   1. The user accepts the friend request. 2. The user who sent the request is added to the user’s friend list. |
| Alternative flows:  None |

|  |
| --- |
| **Use Case: Reject Friend Request** |
| ID: 14 |
| Brief Description:  One registered user can reject the friend request which was sent by another registered user who is not present in his friend list. |
| Primary Actors:  Registered User |
| Secondary Actors:  None |
| Preconditions:   1. The user must be logged in to the system. 2. The user must receive the friend request from another user. |
| Main flow:   1. The user can see total number of friend requests near the red user icon in his profile page. 2. The user clicks the red user icon in the profile page 3. For each user who sent the friend request,    1. The system displays his username along with “accept” and “reject” buttons. 4. The user clicks “reject” button to reject particular friend request. |
| Postconditions:   1. The user rejects the friend request. 2. The user who sent the request is not added to the user’s friend list. |
| Alternative flows:  None |

|  |
| --- |
| **Use Case: Remove Friend** |
| ID: 15 |
| Brief Description:  One registered user can remove another registered user who is present in his friend list. |
| Primary Actors:  Registered User |
| Secondary Actors:  None |
| Preconditions:  The user must be logged in to the system. |
| Main flow:   1. For each friend in the user’s friend list,    1. The system displays his username along with “unfriend” button 2. The user clicks the “unfriend” button. 3. The system removes the selected user from the user’s friend list. |
| Postconditions:  The system removes the selected user from the user’s friends list. |
| Alternative flows:  None |

|  |
| --- |
| **Use Case: Update User Roles** |
| ID: 16 |
| Brief Description:  An admin can manage the role of registered users including his role. |
| Primary Actors:  Admin |
| Secondary Actors:  Registered Users |
| Preconditions:  The user must be logged in to the system as admin. |
| Main flow:   1. The admin is able to see all the registered users in the system along with their roles. 2. For each registered user in the system,    1. The system displays username, role, update button, and a delete button. 3. The admin assigns either “admin” or “user” role for a particular user. 4. The admin selects “UPDATE” button to update the role. 5. The system updates the role of a particular user. 6. The system displays the message as soon as the request is complete. |
| Postconditions:  The system displays the message to the admin as soon as the role of the selected user changes. |
| Alternative flows:  None |

|  |
| --- |
| **Use Case: Delete Users** |
| ID: 17 |
| Brief Description:  An admin can delete the any registered users present in the system including him. |
| Primary Actors:  Admin |
| Secondary Actors:  Registered Users |
| Preconditions:  The user must be logged in to the system as admin. |
| Main flow:   1. The admin is able to see all the registered users in the system along with their roles. 2. For each registered user in the system,    1. The system displays username, role, update button, and a delete button. 3. The admin selects “DELETE” button to delete a particular user. 4. The system removes the user from the system. 5. The system displays the message as soon as the request is complete. |
| Postconditions:  The system displays the message to the admin as soon as the user gets removed from the system. |
| Alternative flows:  None |

|  |
| --- |
| **Use Case: Browse Email** |
| ID: 18 |
| Brief Description:  An admin browses emails that are sent by the users regarding comments or suggestions. |
| Primary Actors:  Admin |
| Secondary Actors:  Any User |
| Preconditions:  The user must be logged in to the system as admin. |
| Main flow:   1. The admin is able to see email notifications in the form of red envelope with total number of emails near it. 2. The admin clicks the envelope symbol. 3. The admin is able to see all the incoming emails along with old emails (if they exist).   extension point: readEmail  extension point : deleteEmail |
| Postconditions:  The admin is able to see all the emails in the system. |
| Alternative flows:  None |

|  |
| --- |
| **Extension Use Case: Read Email** |
| ID: 18 |
| Brief Description:  An admin reads the emails which are sent by the users regarding comments or suggestions. |
| Primary Actors:  Admin |
| Secondary Actors:  Any User |
| Preconditions:   1. The user must be logged in to the system as admin. 2. The admin must click envelope symbol that is displayed in his page. |
| Main flow:   1. The admin must click any email at a time to read the particular email. 2. The user is able to read the email which is sent by a user.   extension point: replyEmail |
| Postconditions:  The system displays email to the admin that is sent by a user. |
| Alternative flows:  None |

|  |
| --- |
| **Extension Use Case: Reply Email** |
| ID: 19 |
| Brief Description:  An admin replies an email back to the user. |
| Primary Actors:  Admin |
| Secondary Actors:   1. Any User 2. Email Providers |
| Preconditions:   1. The user must be logged in to the system as admin. 2. The admin must open the email. |
| Main flow:   1. The main flow starts when the admin clicks “Reply” link in the page where he read the email. 2. The admin writes the message in the text area. 3. The admin clicks “Send” button to reply the email 4. The system replies the email back to the client. |
| Postconditions:  The system replies the email back to the user. |
| Alternative flows:  None |

|  |
| --- |
| **Extension Use Case: Delete Email** |
| ID: 19 |
| Brief Description:  An admin deletes the email which is sent by the user. |
| Primary Actors:  Admin |
| Secondary Actors:   1. Any User 2. Email Providers |
| Preconditions:  The user must be logged in to the system as admin. |
| Main flow:   1. include (Browse Email) 2. The admin is able to see all the emails in the system. 3. For each email that the admin received,    1. There is a checkbox near to select a particular email 4. The admin selects one or more checkboxes to delete the emails. 5. The admin selects “Trash” button to remove emails from the system. 6. The system deletes the selected email. |
| Postconditions:  The system deletes the emails which are selected by the admin. |
| Alternative flows:  None |

|  |
| --- |
| **Use Case: Post Comments** |
| ID: 20 |
| Brief Description:  The user can post comments in his profile picture, or he can post comments on his friend’s profile picture. |
| Primary Actors:  Registered User |
| Secondary Actors:  Friend |
| Preconditions:  The user must be logged in to the system. |
| Main flow:   1. The main flow starts when the user writes comments on the profile picture. 2. If the user posts comments in his profile picture,    1. The comment is visible to the user in his profile page below his profile picture.    2. The comment is visible to the user’s friends when they visit his profile page.    3. The comment is not visible to the registered users who are not present in the user’s friends list. 3. If the user visits his friend’s profile page and posts comments on his profile picture,    1. The comment is visible to the particular friend along with his other friends, and the user.    2. The comment is not visible to other registered users who are not present in his friend’s friends list. 4. The system posts the comments on the profile page. |
| Postconditions:  The system posts the comments on the profile page. |
| Alternative flows:  None |

|  |
| --- |
| **Use Case: Get YouTube Recommended Videos** |
| ID: 21 |
| Brief Description:  A registered user gets recommended videos from the Youtube in his profile page. |
| Primary Actors:  Registered Users |
| Secondary Actors:  Youtube Provider |
| Preconditions:  The user must be logged in to the system. |
| Main flow:   1. The main flow starts when the user navigates to “Media” section in the navigation bar and clicks “Videos” 2. The user is able to see many recommended videos on the right hand side. 3. The use case ends. |
| Postconditions:  The system displays many YouTube recommended videos to the user. |
| Alternative flows:  None |

|  |
| --- |
| **Use Case: Search Youtube Videos** |
| ID: 21 |
| Brief Description:  A registered user searches Youtube videos from his profile page. |
| Primary Actors:  Registered Users |
| Secondary Actors:  YouTube Provider |
| Preconditions:   1. The user must be logged in to the system. 2. The user must navigate to the video page by clicking “Media” and then “Videos” in the navigation bar from his profile page. |
| Main flow:   1. The main flow starts when the user searches the YouTube videos from his profile page. 2. The user enters the search criteria in the search field. 3. The user clicks “Search” button to search the item. 4. The system displays videos that match the search criteria.   extension point: addToPlaylist |
| Postconditions:  The user can see the searched YouTube videos in his profile page. |
| Alternative flows:  None |

|  |
| --- |
| **Extension Use Case: Add To Playlist** |
| ID: 22 |
| Brief Description:  A registered user can add videos to an existing playlist. |
| Primary Actors:  Registered Users |
| Secondary Actors:  YouTube Provider |
| Preconditions:   1. The user must be logged in to the system. 2. The user must navigate to the video page by clicking “Media” and then “Videos” in the navigation bar from his profile page. |
| Main flow:   1. include (Search YouTube Videos) 2. The user can see the searched YouTube videos in his profile page. 3. For each video that the system displays,    1. The user can see a checkbox, a dropdown that includes all of his playlists, and a “Add To Playlist” button. 4. The user selects a video by checking the checkbox. 5. The user selects a playlist from the dropdown to add the checked video to that playlist. 6. The user selects “Add to Playlist” button. 7. The system adds the checked video to the selected playlist. |
| Postconditions:  The user is able to add searched videos to the selected playlist. |
| Alternative flows:  None |

|  |
| --- |
| **Use Case: Create Playlists** |
| ID: 23 |
| Brief Description:  A registered user can create a playlist with YouTube videos in his profile page. |
| Primary Actors:  Registered Users |
| Secondary Actors:  YouTube Provider |
| Preconditions:   1. The user must be logged in to the system. 2. The user must navigate to the video page by clicking “Media” and then “Videos” in the navigation bar from his profile page. |
| Main flow:   1. The user enters a playlist name, playlist details, and selects playlist privacy in the video page. 2. The user clicks “Create Playlist” button to create a new playlist. 3. The system creates a new playlist to the user. |
| Postconditions:  The user gets a new YouTube playlist where he can add YouTube videos. |
| Alternative flows:  None |

|  |
| --- |
| **Use Case: View Playlists** |
| ID: 24 |
| Brief Description:  A registered user can view all his YouTube playlists from his profile page. |
| Primary Actors:  Registered Users |
| Secondary Actors:  YouTube Provider |
| Preconditions:   1. The user must be logged in to the system. 2. The user must navigate to the video page by clicking “Media” and then “Videos” in the navigation bar from his profile page. |
| Main flow:   1. The user clicks “Show My Playlist” link to view all his YouTube playlists. 2. The system displays all the playlists to the user. |
| Postconditions:  The user is able to view his playlists from his profile page. |
| Alternative flows:  None |

|  |
| --- |
| **Use Case: Create Albums** |
| ID: 25 |
| Brief Description:  A registered user can create photo albums from his profile page. |
| Primary Actors:  Registered Users |
| Secondary Actors:  None |
| Preconditions:   1. The user must be logged in to the system. 2. The user must navigate to the album section by clicking “Media” and then “Pictures” in the navigation bar from his profile page. |
| Main flow:   1. The main flow starts when the user clicks “Create Album” button in the album section of his profile page. 2. The user creates a new album by giving an album name and uploading images to the particular album. 3. The system creates an album for the user. |
| Postconditions:  The user is able to see a new album in his album section. |
| Alternative flows:  None |

|  |
| --- |
| **Use Case: Open Album** |
| ID: 26 |
| Brief Description:  A registered user can view his all albums from his profile page. |
| Primary Actors:  Registered Users |
| Secondary Actors:  None |
| Preconditions:   1. The user must be logged in to the system. 2. The user must navigate to the album section by clicking “Media” and then “Pictures” in the navigation bar from his profile page. |
| Main flow:   1. The main flow starts when the user clicks “View Album” button in the album section of his profile page. 2. The system displays all the albums which belong to the particular user. |
| Postconditions:  The user is able to view all the albums from his profile page. |
| Alternative flows:  None |

|  |
| --- |
| **Use Case: View Album Pictures** |
| ID: 27 |
| Brief Description:  A registered user can view pictures from an album. |
| Primary Actors:  Registered Users |
| Secondary Actors:  None |
| Preconditions:   1. The user must be logged in to the system. 2. The user is in album section and clicks “View Album” button. |
| Main flow:   1. include (Open Album) 2. The user is able to view all the albums from his profile page. 3. The user clicks a particular album. 4. The system displays all the pictures inside the particular album to the user. |
| Postconditions:  The user is able to view all the pictures that are present inside the selected album. |
| Alternative flows:  None |

|  |
| --- |
| **Use Case: Update Information** |
| ID: 27 |
| Brief Description:  A registered user can update his information in the system. |
| Primary Actors:  Registered Users |
| Secondary Actors:  None |
| Preconditions:   1. The user must be logged in to the system. 2. The user navigates to the “Settings” and clicks “Update Personal Information” in the navigation bar. |
| Main flow:   1. If the user select “Update Address Details”    1. The system displays the user’s address details that are currently present in the system.    2. The user makes some changes to his address details. 2. If the user selects “Update User Details”    1. The system displays the user’s personal details that are currently present in the system.    2. The user makes some changes to his personal details 3. If the user selects “Update Education Details”,    1. The system displays the user’s education details that are currently present in the system.    2. The user makes some changes to his education details. 4. The system updates the information. 5. The user is able to update his information in the system. |
| Postconditions:  The user is able to update his information in the system. |
| Alternative flows:  None |