Lemmi

Use-Case Specification

Version 1.1

Revision History

| **Date** | **Version** | **Description** | **Author** |
| --- | --- | --- | --- |
| 08/11/2021 | 1.0 | Brief use-case model and use-case specification. | Tien Nguyen Van,  Khoa Ta Ngoc Minh,  Dat Thai Ngoc Thanh |
| 13/11/2021 | 1.1 | Complete all defined use-case specifications and use-case UML. | Lemmi team |
|  |  |  |  |
|  |  |  |  |

Table of Contents

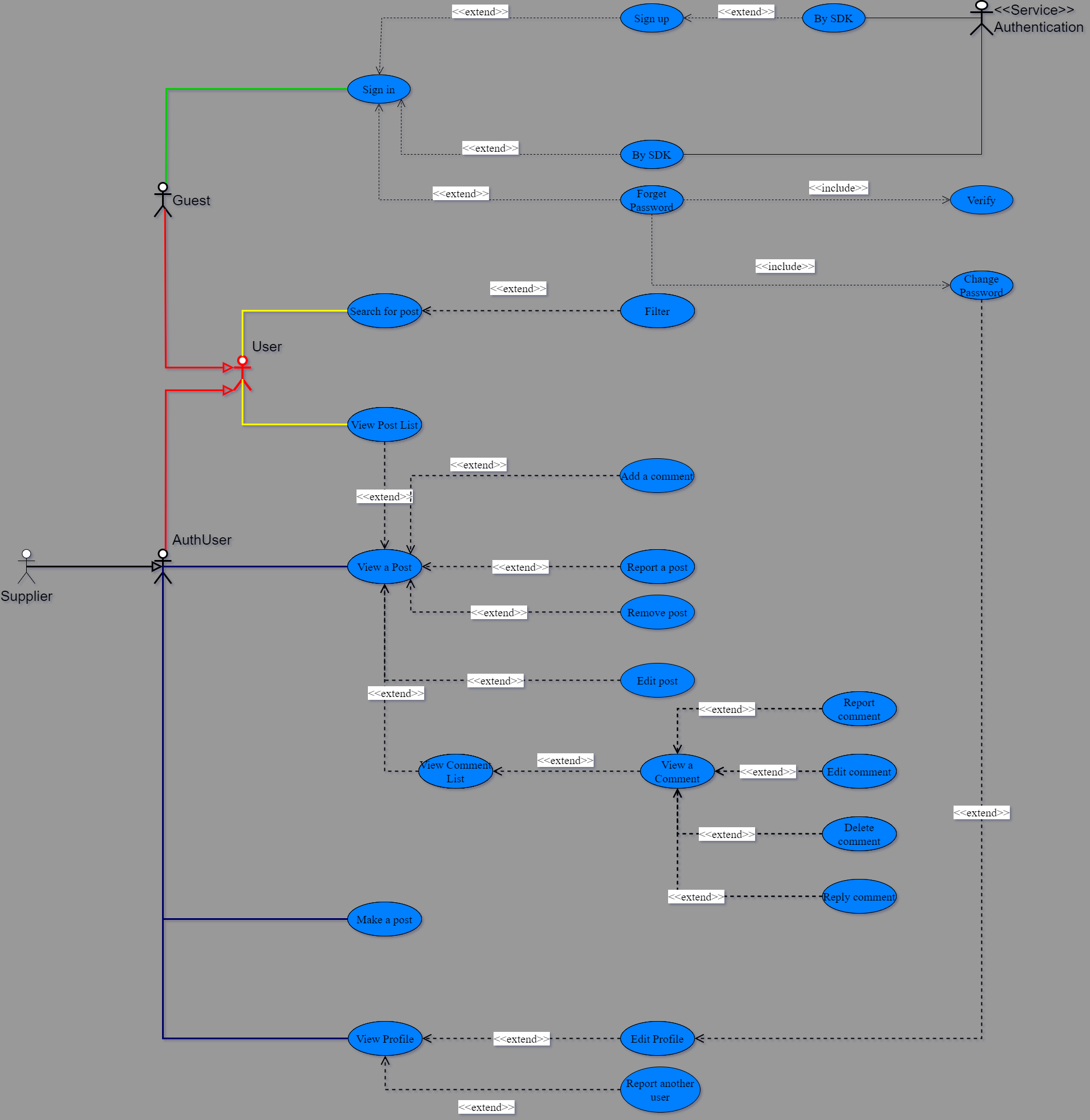
1. Use-case Model 4

2. Use-case Specifications 4

2.1 Use-case: Add a product to cart 4

2.2 Use-case: Create a new account 5

# Use-case Model



Link: [Use-case.drawio.png](https://drive.google.com/file/d/1xjyIkhckg9Hl7jgcWwHPiijlvMagoL8r/view?usp=sharing)

# Use-case Specifications

## Use-case: Login by Lemmi account

| Use case Name | Login by Lemmi account |
| --- | --- |
| Brief description | This use case allows users to log into the system to perform some specific tasks |
| Actors | Guest, <<Service>> Authentication |
| Basic Flow | 1. Fill in signed-in account:    1. Fill username:       1. Click on the “username” edit box.       2. Fill in the “username” edit box.    2. Fill password:       1. Click on the “password” edit box.       2. Fill in the “password” edit box. 2. Click on the “Log In” button to save the change. 3. User is directed to the homepage. |
| Alternative Flows | **Alternative flow 1: Username and Password are not matched.**   1. The user will be notified about this. 2. The user is allowed to re-type their account.   **Alternative flow 2: Username has not existed in the system.**   1. The user will be notified about this and recommended to create a new 2. The user is allowed to re-type their account. |
| Extension Flows | **Extension Flow 1:** **Login by a third-party account**   1. Choose the platform by which the user wants to sign in. 2. User signs in their third-party account. 3. Users click “Accept” to allocate Lemmi’s access to their account. 4. User is directed to the homepage.   **Extension Flow 2: The connection to a third-party platform has failed.**   1. The user will be notified about this. 2. The user is allowed to login again.   **Extension flow 3: The third-party account has not existed in Lemmi’s system.**   1. The user will be notified about this and recommended to sign up a new account. 2. The user is allowed to login again. |
| Pre-conditions | * User accessed home page. |
| Post-conditions | * The user successfully logged into Lemmi. |

## Use-case: Register (Create a Lemmi account)

| Use case Name | Create an new account |
| --- | --- |
| Brief description | This use-case describes how a user can create an account. |
| Actors | Guest, <<Service>> Authentication |
| Basic Flow | 1. Fill in username 2. Fill in password 3. Fill in First name 4. Fill in Last name 5. Choose gender 6. Fill in birthday 7. Fill in address 8. Fill in email 9. Fill in phone number 10. Click on the “Save” button to save the change. 11. The Edit profile page is reloaded with new updates. |
| Alternative Flows | **Alternative flow 1: Invalid content for at least one field.**   1. The user is notified about this. 2. The user fixes their content. 3. Continue step #10 of basic flow.   **Alternative flow 2: There is an existing account with the same username in the system.**   1. The user is notified about this. 2. The user chooses another username. 3. Continue step #10 of basic flow. |
| Extension Flows | **Extension Flow 1: Sign up by a third-party account**   1. Choose the platform by which the user wants to sign up. 2. User signs in their third-party account. 3. Users click “Accept” to allocate Lemmi’s access to their account. 4. User is directed to the homepage.   **Extension Flow 2: The connection to a third-party platform has failed.**   1. The user will be notified about this. 2. The user is allowed to sign up again. |
| Pre-conditions | * User accessed home page. |
| Post-conditions | * The user successfully created an account with their information saved in the system. |

## Use-case: Edit user’s profile

| Use case Name | Edit user’s profile |
| --- | --- |
| Brief description | This use-case describes how a user can edit his/her profile. |
| Actors | AuthUser |
| Basic Flow | 1. Click on the edit button to edit the profile. 2. Edit any fields in profile (First name, Last name, Date of Birth, Email, Phone, Address) 3. Save the change. |
| Alternative Flows | **Alternative flow 1: User cancels the change.**   1. Click on the cancel button to delete the change. 2. The profile remains old data. 3. Continue #1 of basic flow if user want to edit again. |
| Pre-conditions | * The user has already logged in. * The user is on the profile page. |
| Post-conditions | * The user successfully edited the profile. * New user’s information, that its attributes can be empty if permissible, is saved and shown to the user. |

## Use-case: Create a post

| Use case Name | Create a post |
| --- | --- |
| Brief description | Create a post to appear in the post section. |
| Actors | AuthUser |
| Basic Flow | 1. User clicks on the post button. 2. The page shows the user a form. 3. User fill in required information in below field:  * Title * Tags * Content * Image * Rating * Visible scope(all, friend, only me)  1. User clicks on the “Post” button. 2. The page shows a confirmation popup. 3. User clicks on the confirm button. |
| Alternative Flows | **Alternative flow 1: 1 of required information in the form is blank.**   1. The page show up error in the field that is blank in #3   **Alternative flow 2: User clicks close on the confirmation popup.**   1. The confirmation pop up closes 2. Return to basic flow #3 with current information that user filled   **Alternative flow 3: User closes the form**   1. The page shows a confirmation pop up   2. 1. User clicks exit  2. 1. 1 The page close the form and the post will not be created  2. 2. User clicks return to post  2. 2. 1 Return to basic flow #2 and keep data that user filled before |
| Pre-conditions | User has successfully logged in. |
| Post-conditions | * The post appears in the post section. * The page navigates the user to the post page. |

## Use-case: Edit a post

| Use case Name | Edit a post’s content. |
| --- | --- |
| Brief description | This use case describes how a post’s content is edited by its owner. |
| Actors | AuthUser |
| Basic Flow | 1. Click on the edit button. 2. Change any fields of the post (visible scope, title, content, image, tag, rating). 3. Save the change. |
| Alternative Flows | **Alternative flow 1: User cancels the changes.**   1. Click on the Cancel button. 2. The changes is deleted, get back to the post’s page with old data. 3. Continue #1 of basic flow if use wants to edit the post again.   **Alternative flow 2: User does not fill in all fields of the post.**   1. Pop up a warning message to notify user which fields are not completed. 2. Continue #2 of basic flow. |
| Pre-conditions | * The user has already logged in. * Users have already been on the post’s page. |
| Post-conditions | The user successfully edits their own post and the data is saved in the database. |

## Use-case: Comment on a post

| Use case Name | Comment on a post |
| --- | --- |
| Brief description | Compose a comment on a post |
| Actors | AuthUser |
| Basic Flow | 1. User clicks the text area in comment section of the post 2. Users fill in their comments. 3. User clicks the post button or hits enter. |
| Alternative Flows | **Alternative flow 1: User fill in nothing**   1. The page shows the user warning that the field is empty. 2. Post button will not work   **Alternative flow 2: User clicks on another non-purpose region in the website**   1. Basic flow #3 won’t have hit enter functionality |
| Pre-conditions | * User has successfully logged in. * User must be inside (viewing) a post. |
| Post-conditions | The user’s comment appears in comment section of the viewing post |

* 1. **Use-case: Report a post**

| Use case Name | Report a post |
| --- | --- |
| Brief description | Report a post that is violating or having false information |
| Actors | AuthUser |
| Basic Flow | 1. Inside the post, user clicks report button 2. The page show up confirmation pop up with reasons drop list 3. User choose reason then click confirm |
| Alternative Flows | **Alternative flow 1: User don’t choose a reason**   1. Prompt the user to choose a reason   **Alternative flow 2: User click the close button on confirmation popup**   1. User return to the viewing post |
| Pre-conditions | User has successfully logged in  User must be inside (viewing) a post |
| Post-conditions | A report for the viewing post is created in the system and waiting to be processed |

## Use-case: Search

| Use case Name | Search for a post |
| --- | --- |
| Brief description | Users search for posts that have a related or similar name as the search term. |
| Actors | User |
| Basic Flow | 1. In the homepage, user clicks on the search bar. 2. User fills in the search term. 3. User hits enter. 4. The page show to user post that have similar title in post section |
| Alternative Flows | **Alternative flow 1: No search result for the search terms in the result page**   1. The post section will not change   **Alternative flow 2: No search term was filled**   1. The post section will not change   **Alternative flow 3: User clicks on another non-purpose region in the website**   1. User can continue searching starting basic flow #1 and may skip #2 |
| Pre-conditions | User must be on the website. |
| Post-conditions | The post with a similar name appears in the results section. |

## Use-case: Filter

| Use case Name | Search for a post with filter support |
| --- | --- |
| Brief description | This use case describes how the Customer can search post(s) with filter support. |
| Actors | User |
| Basic Flow | 1. Click on filter in the search bar. 2. Choose filters that will be applied (District, Tag, Sort order) 3. Click search for applying the filter. |
| Alternative Flows | **Alternative flow 1: User clear existing filter.**   1. Click on the clear button in the filter box. 2. Continue #2 of basic flow to choose a new filter.   **Alternative flow 2: No result matches the chosen filter.**   1. The list of post view return “No result” 2. Continue #2 of basic flow to choose a new filter. |
| Pre-conditions | Users are at the homepage. |
| Post-conditions | Search is successful and the main view shows the search results. |

## 

## Use-case: Delete comment

| Use case Name | Delete a comment |
| --- | --- |
| Brief description | This use-case describes how a user can delete his/her/their comment in a post. |
| Actors | AuthUser |
| Basic Flow | 1. Click on the comment options. 2. Choose to delete the comment. 3. Agree to delete in confirmation box popped up. |
| Alternative Flows | **Alternative flow 1: User cancels action delete comment.**   1. Click the cancel button in the confirmation box. |
| Pre-conditions | The user has already logged in.  The user is on the post’s page containing the comment.  The user responds to the comment. |
| Post-conditions | Delete the comment successfully in the database.  The post’s page does not contain the comment any longer. |

## 

## Use-case: Edit comment

| Use case Name | Edit a comment |
| --- | --- |
| Brief description | This use-case describes how a user can edit his/her/their comment in a post. |
| Actors | AuthUser |
| Basic Flow | 1. Click on the comment options. 2. Choose to edit comment. 3. Modify comment content. 4. Save the change. |
| Alternative Flows | **Alternative flow 1: User cancels the changes.**   1. Click the cancel button. 2. The change is deleted. The post’s page still shows the old comment. |
| Pre-conditions | The user has already logged in.  The user is on the post’s page containing the comment.  The user responds to the comment. |
| Post-conditions | Edit the comment successfully.  New data of the comment is saved in the database.  The post’s page replaces the comment with the new one. |

## 

## Use-case: Report comment

| Use case Name | Report a comment |
| --- | --- |
| Brief description | This use-case describes how user can report a comment. |
| Actors | AuthUser |
| Basic Flow | 1. Click on the comment options. 2. Choose to report the comment. 3. The system pops up a report box. 4. Choose the reason for the report. 5. Choose confirm the report to send it. |
| Alternative Flows | **Alternative flow 1: The user closes the report box to cancel the report.**   1. Click on the close report box button. |
| Pre-conditions | The user has already logged in.  The user is on the post’s page containing the comment. |
| Post-conditions | The report is sent to the administrator. |

## 

## Use-case: Reply comment

| Use case Name | Reply a comment |
| --- | --- |
| Brief description | This use-case describes how a user can reply to a comment which can belong to this user or other users. |
| Actors | AuthUser |
| Basic Flow | 1. Click on the reply comment button of the comment that the user wants to reply to. 2. Write comment. 3. Post the comment. |
| Alternative Flows | **Alternative flow 1: The user cancels the reply comment action.**   1. Click on the cancel button. 2. The reply comment is not saved and the post’s page shows the same content as before the action. |
| Pre-conditions | The user has already logged in.  The user is on the post’s page containing the comment. |
| Post-conditions | A new comment is added to the database and shown on the post’s page below the comment it replies to. |

## 

## Use-case: Report another user

| Use case Name | Report another user |
| --- | --- |
| Brief description | This use-case describes how a user reports another user. |
| Actors | AuthUser |
| Basic Flow | 1. Click on the target user that will be reported. 2. Click on the report button. 3. The system pops up a confirmation box. 4. Choose the reason for the report. 5. Agree to report user. |
| Alternative Flows | **Alternative flow 1: User cancels the report user action.**   1. Click the close report box button. |
| Pre-conditions | The user has already logged in.  The reported user and the user reporting are not the same. |
| Post-conditions | The report is sent to the administrator. |

## Use-case: Forgot Password

| Use case Name | Forgot Password |
| --- | --- |
| Brief description | This use-case describes when a user forgot password. |
| Actors | AuthUser |
| Basic Flow | 1. Click on the forgot password button on the. 2. The system moves to the forgot password page. 3. Fill in the username or phone number of the account you forgot. 4. Verify:    1. Verify by email:   System sends a code to your email  Fill in the code you had received   * 1. Verify by phone number   System will send an OTP code to your phone   1. Change password |
| Alternative Flows | **Alternative flow 1: Invalid username.**   1. The user is notified about this. 2. The user fixes their content. 3. Continue step #4 of basic flow. |
| Pre-conditions | * User accessed home page. * User have an account |
| Post-conditions | * The user successfully gets the password back. |

## Use-case: View profile

| Use case Name | Edit a post’s content. |
| --- | --- |
| Brief description | This use case describes how a post’s content is edited by its owner. |
| Actors | AuthUser |
| Basic Flow | 1. Click on the edit button. 2. Change any fields of the post (visible scope, title, content, image, tag, rating). 3. Save the change. |
| Alternative Flows | **Alternative flow 1: User cancels the changes.**   1. Click on the Cancel button. 2. The changes is deleted, get back to the post’s page with old data. 3. Continue #1 of basic flow if use wants to edit the post again.   **Alternative flow 2: User does not fill in all fields of the post.**   1. Pop up a warning message to notify user which fields are not completed. 2. Continue #2 of basic flow. |
| Pre-conditions | * The user has already logged in. * Users have already been on the post’s page. |
| Post-conditions | The user successfully edits their own post and the data is saved in the database. |

## Use-case: Edit user’s profile

| Use case Name | Edit user’s profile |
| --- | --- |
| Brief description | This use-case describes how a user can edit his/her profile. |
| Actors | AuthUser |
| Basic Flow | 1. Click on the edit button to edit the profile. 2. Edit any fields in profile (First name, Last name, Date of Birth, Email, Phone, Address) 3. Save the change. |
| Alternative Flows | **Alternative flow 1: User cancels the change.**   1. Click on the cancel button to delete the change. 2. The profile remains old data. 3. Continue #1 of basic flow if user want to edit again. |
| Pre-conditions | * The user has already logged in. * The user is on the profile page. |
| Post-conditions | * The user successfully edited the profile. * New user’s information, that its attributes can be empty if permissible, is saved and shown to the user. |

## Use-case: Delete a post

| Use case Name | Create a post |
| --- | --- |
| Brief description | Create a post to appear in the post section. |
| Actors | AuthUser |
| Basic Flow | 1. User clicks on the post button. 2. The page shows the user a form. 3. User fill in required information in below field:  * Title * Tags * Content * Image * Rating * Visible scope(all, friend, only me)  1. User clicks on the “Post” button. 2. The page shows a confirmation popup. 3. User clicks on the confirm button. |
| Alternative Flows | **Alternative flow 1: 1 of required information in the form is blank.**   1. The page show up error in the field that is blank in #3   **Alternative flow 2: User clicks close on the confirmation popup.**   1. The confirmation pop up closes 2. Return to basic flow #3 with current information that user filled   **Alternative flow 3: User closes the form**   1. The page shows a confirmation pop up   2. 1. User clicks exit  2. 1. 1 The page close the form and the post will not be created  2. 2. User clicks return to post  2. 2. 1 Return to basic flow #2 and keep data that user filled before |
| Pre-conditions | User has successfully logged in. |
| Post-conditions | * The post appears in the post section. * The page navigates the user to the post page. |

## Use-case: View a post

| Use case Name | Edit a post’s content. |
| --- | --- |
| Brief description | This use case describes how a post’s content is edited by its owner. |
| Actors | AuthUser |
| Basic Flow | 1. Click on the edit button. 2. Change any fields of the post (visible scope, title, content, image, tag, rating). 3. Save the change. |
| Alternative Flows | **Alternative flow 1: User cancels the changes.**   1. Click on the Cancel button. 2. The changes is deleted, get back to the post’s page with old data. 3. Continue #1 of basic flow if use wants to edit the post again.   **Alternative flow 2: User does not fill in all fields of the post.**   1. Pop up a warning message to notify user which fields are not completed. 2. Continue #2 of basic flow. |
| Pre-conditions | * User is at the list post page. |
| Post-conditions | The user successfully view content of the post. |

## 