React To Spring

React To Spring Forums
Phân tích Use Case

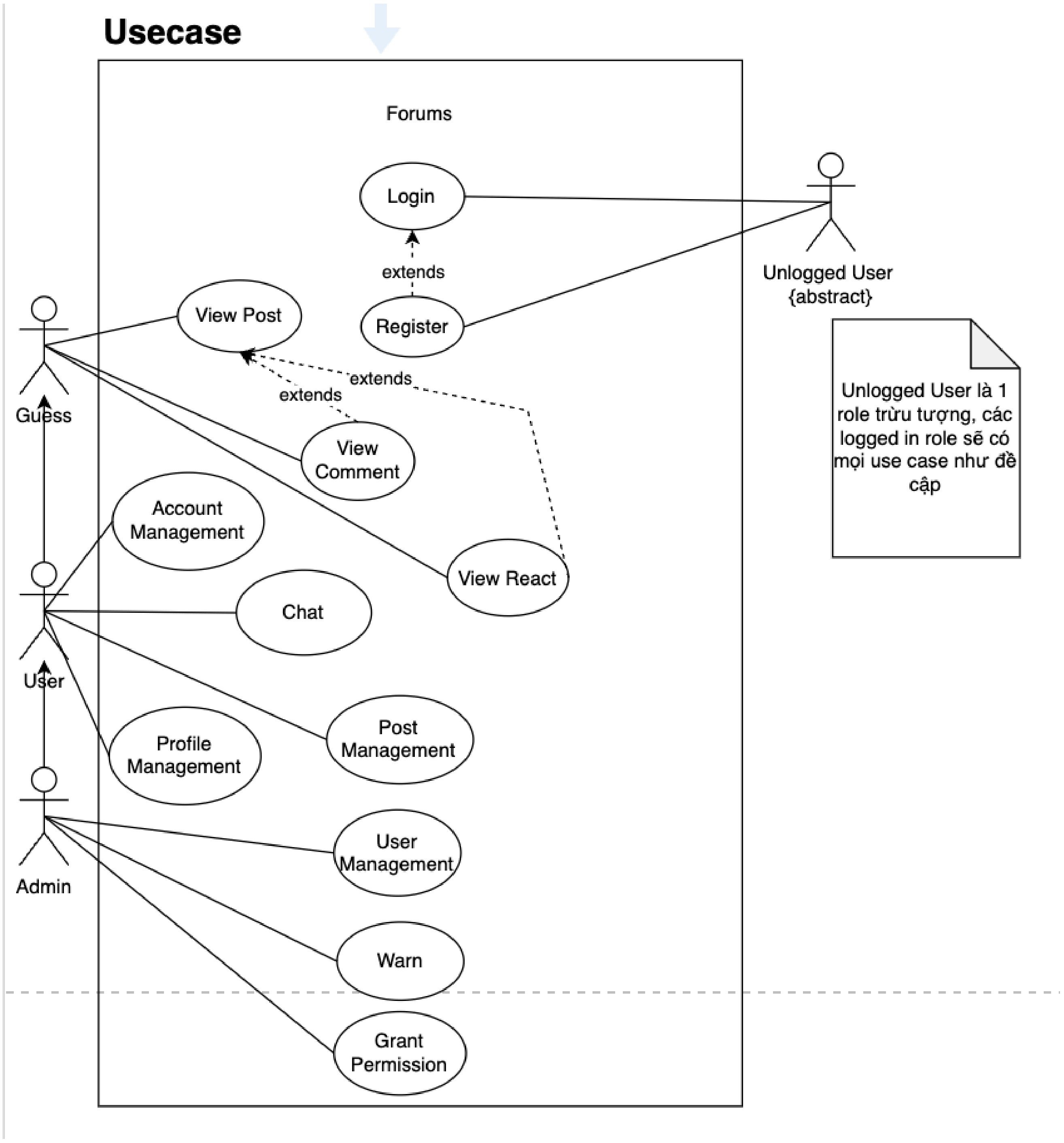
Version 2.0

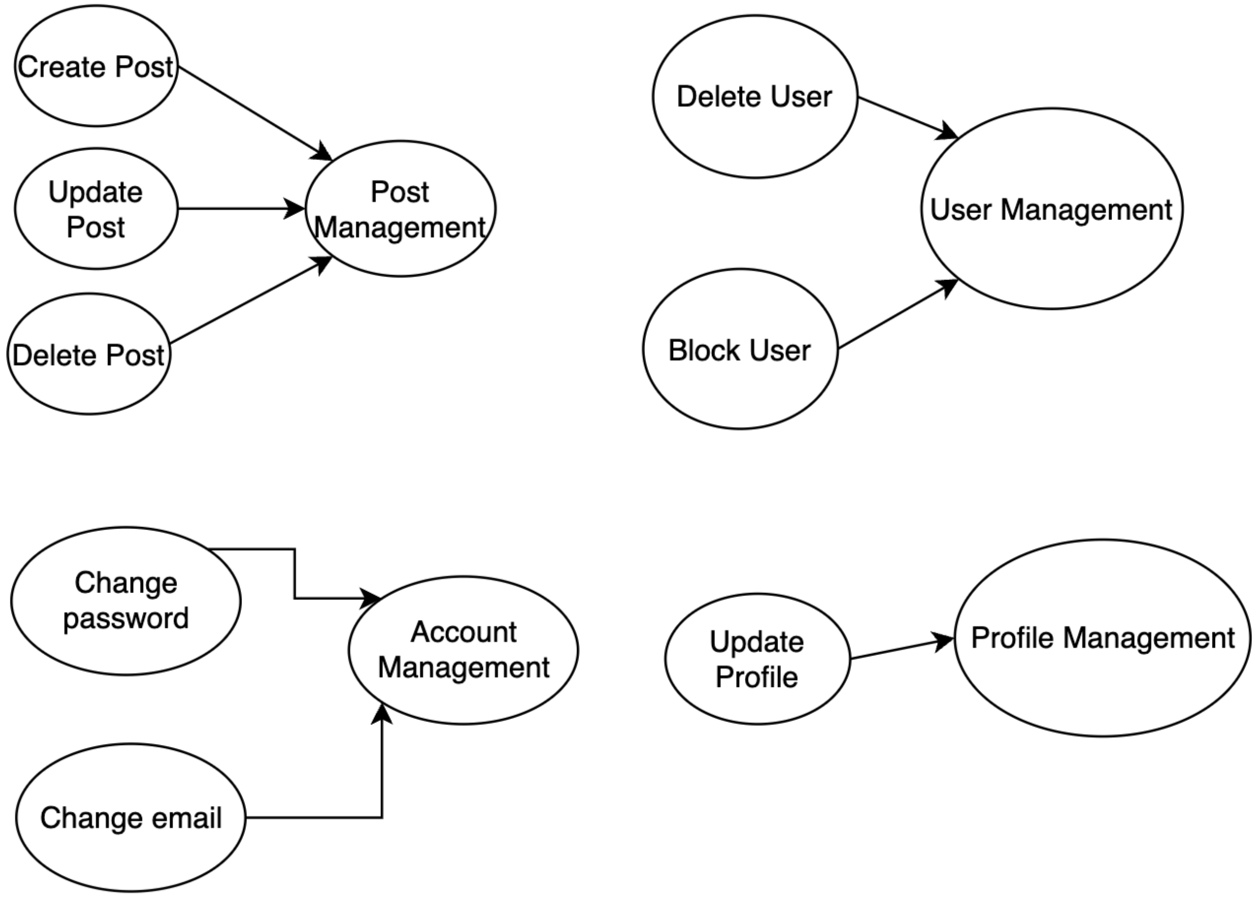
Revision History

Date	Version	Description	Author
03/11/2021	1.0	The initial version	Team
20/11/2021	2.0	Revised version of the initial one. 1. Change Alternative flow of Login use case: Forget password. (section 2.1 Login use case) Reason: To make it similar to other popular social media platforms and be simpler for user. 2. Change basic flow of change password use case (section 2.3 Change Password use case) 3. Change basic flow of change email use cases. (section 2.4 Change email use case)	Tuyen

Table of Contents

1. Use Case Model




Create Post
Update Post
Delete Post
Post Management
Delete User
Block User
User Management
Change password
Change email
Account Management
Update Profile
Profile Management

2. Use Case Specifications

2.1. Login Use Case

Use Case ID	U001
Name	Login
Brief description	This use case describes how a user logs into the Forum Website.
Actors	Unlogged User
Preconditions	User has registered an account in the system.
Basic flow	1. User access login page 2. System displays the login form with fields for username, password, along with a “Login” button, “Remember me” to save password, “Forget password” 3. User enters their username, password 4. User selects "Remember Me" if they wish to stay logged in. 5. User click the “Login” button to submit the form 6. The system inform success and the user is redirected to main page.
Postconditions	The user is logged in and has access to the main dashboard with session saved if "Remember Me" was chosen.
Alternative flow	1. Forgot password flow : 1.1. User choose the ‘Foret Password’ option at step 2. 1.2. System prompts for an email and user fill in the email that they have registered before and click “Send verification code” button. 1.3. System sends a password reset code to the provided emails. 1.4. System displays a form with fields for new password and verification code. 1.5. User enters required information and clicks “Change password” button 1.6. System validates information and updates database. 1.7. System display ‘Reset Password successfuly’ message and redirect user to login page. 2. Incorrect username or password flow: 2.1. At step 6 of basic flow, if user enters wrong credentials, system will displays error message.

	2.2. User can try again until they log in successfully.
Postconditions	1. New password is saved in database and user can login with new password. 2. If user enter wrong credential they can not log in the system and remains on the login page until log in successfully and redirected to main page.
Special requirements	1. Use password encryption, regex validation to ensure they meet security standard (minimum length, complexity)
Extension points	Password recovery : Enables password reset if forgotten.

2.2. Register Use Case

Use Case ID	U002
Name	Login
Brief description	This use case describes how a new user create account.
Actors	User
Preconditions	There is no precondition for this use case
Basic flow	1. User accesses registration page. 2. System displays a form with fields for username, password, email, province. 3. User enters the required information and submits the form 4. System sends a verification link to provided email. 5. User clicks on the link sent to their emails for verification. 6. System creates new account, inform success message and redirect user to main page.
Postconditions	A new user account is created in the system and user is redirected to main page.
Alternative flow	1. Username or email already taken 1.1. If the username entered by the user is already taken, the system displays an error message. 1.2. The user is prompted to choose a different username. 2. Week password 2.1. If the password entered does not meet the system's security requirements (minimum length, complexity), the system displays a message with the password criteria. 2.2. The user is prompted to enter a stronger password. 3. Expired verification link 3.1. If the user does not verify within a specified time, the link will be expired

	3.2. System displays error message 3.3. User clicks on “Resend” button 3.4. System sends new verification link to user’s email 3.5. User click on the link to reverify.
Postconditions	1. Username or email already taken : user enter different information until successful and a new account is created. 2. Weak password : user enter different password until successful and a new account is created. 3. Expired verification link : new verification link sent to user’s email and user can verify successfully, then they are redirected to successful page.
Special requirements	Use password encryption, regex validation to ensure they meet security standard (minimum length, complexity)
Extension points	Email verification : 1. The system sends a verification email to the user. 2. The user must click the verification link to activate their account fully. This step ensures the user's email is valid and active.

2.3. Change Password Use Case

Use Case ID	**U003**
Name	Change Password
Brief description	This use case describes how a user change their account’s password.
Actors	User, Admin
Preconditions	User has registered an account in the system and needs to sign into the system.
Basic flow	1. User navigates to the "Change Password" page. 2. User enters the current password and new password. 3. User click “Send code” button. 4. System sends a verification code to the user's email. 5. User enters the verification code. 6. System validates the verification code, old password and new password. 7. System updates the password. 8. System confirms the password change.
Postconditions	User's password is changed.
Alternative flow	1. Validation is fail 1.1. If the entered information is incorrect, the system displays an error message. 1.2. User enter information again until successful.

	2. Expired verification code 2.1. If the user does not verify within a specified time, the link will be expired 2.2. System displays error message 2.3. User clicks on “Resend” button 2.4. System sends new verification code to user’s email 2.5. User uses new code to validate. 3. Sending mail is fail 3.1. If system can not send email, user is redirected to sending email fail page. 3.2. User enter information again until successful.
Postconditions	1. Validation is fail : The system displays an error message, and the user's information remains unchanged. 2. Expired verification link : new verification code sent to user’s email and user can verify successfully, then they can change their password.
Special requirements	1. Verification code is valid for a limited time. 2. New password must meet security requirements.
Extension points	There is no extension points for this use case

2.4. Change email Use Case

Use Case ID	**U004**
Name	Change Password
Brief description	This use case describes how a user change their account’s email.
Actors	User, Admin
Preconditions	User has registered an account in the system and needs to sign into the system.
Basic flow	1. User navigates to the "Change Email" page. 2. System displays a form with field for password, new email and verification code. 3. User enters password, new email address and click on “Send code” button. 4. System sends a verification code to the new email address. 5. User enters the verification code. 6. System validates the verification code. 7. System updates the email address.

	8. System confirms the email address change.
Postconditions	User's email address is changed. The old one will be deleted and user can not use it later.
Alternative flow	1. Sending mail is fail 1.1. If system can not send email, user is redirected to sending email fail page. 1.2. User enter information again until successful. 2. Expired verification code 2.1. If the user does not verify within a specified time, the link will be expired 2.2. System displays error message 2.3. User clicks on “Resend” button 2.4. System sends new verification code to user’s email 2.5. User uses new code to validate.
Postconditions	1. Sending email is fail : 1.1. The user's email address remains unchanged. 1.2. The system prompts the user to try again or contact support. 2. Expired verification link : new verification code sent to user’s email and user can verify successfully, then they can change their password.
Special requirements	1. Verification code is valid for a limited time.
Extension points	There is no extension points for this use case

2.5. Notification Use Case

Use Case ID	U005
Name	Notification
Brief description	This use case describes how the notification system alerts users and their friends about new activities on the forum.
Actors	User, Admin
Preconditions	User and their friends have registered an account in the system and need to sign into the system.
Basic flow	1. User creates a new post on their personal page, react, comment to their friends’s posts, send a message to their friends. 2. System sends notifications to the user and their friends based on the trigger event via in-app notification.

	2.1. If user create new post, system will send notifications to user's friends and admin to check whether it is valid or not. 2.2. If user react, comment to their friends's posts, send a message to their friends, system will send notifications to those friends 3. User and their friends receive and view the notifications.
Postconditions	1. User and their friends are informed about new activities related to posts, comments, reactions, and messages. 2. Admin are informed about new posts. 3. Notifications are marked as read once viewed.
Alternative flow	1. Sending notification is fail 1.1. The notification cannot be delivered due to a system error, network issue. 1.2. System will attempt to resend the notification either after 5 minutes or as soon as the network becomes available.
Postconditions	1. If the notification is successfully resent: The notification is delivered to the intended recipient(s).
Special requirements	1. Timely delivery of notifications should be less than 5 minutes.
Extension points	1. Admin Review: System sends a notification to the admin for review. The admin reviews and approves or rejects the post.

2.6. Update User Profile Use Case

Use Case ID	U006
Name	Update User Profile
Brief description	This use case describes how users can update their profile
Actors	User, Admin
Special requirements	- Users have a verified account
Preconditions	- Users have logged int
Basic flow	1. Users access profile page 2. Fill in the field that they want to update, includes: (firstName, lastName, address, profileImage) 3. Click Update button to update their information
Postconditions	- User's profile is updated in database and shown on screen

Alternative flow	- If the updated user's firstName or lastName is empty, an error snackbar with a message "First name or Last name must not be empty"
Extension points	- There is no extension points for this use case

2.7. Chatting Use Case

Use Case ID	U007
Name	Chatting
Brief description	This use case describes how users can chat with each other
Actors	User, Admin
Special requirements	- Users have verified account - Users have friend
Preconditions	- Users have logged in
Basic flow	1. Users access chatting page 2. Choose another user or a group on sidebar to chat with 3. Type message or choose an image to send
Postconditions	- Sent message is saved in database - Sent message is shown on all user's screen in the chat box
Alternative flow	- If users don't have any friend, a snackbar with message "You don't have any friend to chat with" will appear
Extension points	- There is no extension points for this use case

2.8. Add Friends Use Case

Use Case ID	U008
Name	Add friends
Summary	User make friend with others
Actors	User, Admin
Prerequisites	Registered users
Basic flow	1. The user logs into the forum. 2. The user searches for or views other users' profiles. 3. The user sends a friend request to another user. 4. The friend request is either accepted or rejected by the recipient. 5. If accepted, the users are added as friends and can now interact with each other.

Postconditions	- The users become friends and can now interact with each other
Alternative flow	- If the friend request is rejected
Postconditions	- No connection is established, and the user can send another request later if needed.
Special requirements	- There is no special requirements for this use case.
Extension points	- There is no extension point for this use case.

2.9. Report violating posts

Use Case ID	U009
Name	Report violating posts
Summary	Users report violating posts to admin for judging
Actors	User, Admin
Prerequisites	Registered users
Basic flow	1. The user logs into the forum 2. If the user sees a post that contains violating content, the user can report it to the admin by clicking the report button. 3. After a user sends a report request, the admin receives it and decides whether to delete the post or not.
Postconditions	- The report request was sent successfully
Alternative flow	- User report about a non-existent post: User report about the post that has been deleted by the admin or owner.
Postconditions	- User report about a non-existent post: Message "Something went wrong”
Special requirements	- There is no special requirements for this use case.
Extension points	- There is no extension point for this use case.

2.10. Create Post Use Case

Use Case ID	U010
Name	Create Post
Brief description	This use case describes how a user can upload a post to the Forum Website.
Actors	User, Admin
Preconditions	The user has registered an account and logged in to the Forum successfully.
Basic flow	1. The user logs into the forum. 2. On the home page, users can click on the “What’s New?” box or the “+” sign if they want to post. 3. After clicking, a dialog box will appear, asking to fill in post information such as title, content, and post image. 4. After entering the information and clicking “Post”, the post will be posted 5. After posting, it will automatically navigate to the main page with the post placed at the top. At the same time, it will notify friends.
Postconditions	- The post is successfully created and visible on the main page of the forum and the notifications are sent to the user's friends.
Alternative flow	1. The posting form is canceled: 1. 1 The user opens the posting form but decides not to complete the post and cancels the action. 2. Missing required information: 2. 1 The user fails to enter the required fields (missing required information (e.g, title or content))
Postconditions	1. The posting form is canceled: The dialog box will close without any post being created, and no changes will be saved. 2. Missing required information: No post is created, and the system prompts the user to fill in the missing information.
Special requirements	- A file quantity limit is applied for the post image (6 images maximum). - The post notification system must ensure delivery without delay to friends.
Extension points	- Real-time notification system for immediate friend updates.

2.11. Update Post Use Case

Use Case ID	U011
Name	Update Post
Brief description	The user can edit their previously posted posts.
Actors	User, Admin
Preconditions	The user has registered an account and logged in to the Forum successfully.
Basic flow	1. Users log in to the forum. 2. Search for their previously posted posts. 3. At each post, in the upper right corner there will be a ":" sign, when the user clicks on it, options will appear (edit, delete, ...). 4. After clicking edit, a dialog box will appear, including post information such as title, content and post image before updating 5. After updating the information and clicking "Edit", the post will be updated.
Postconditions	Post updated successfully.
Alternative flow	1. The edit form was canceled: 1. 1 The user opens the edit form but decides not to complete the edit and cancels the action. 2. Missing Required Information: 2. 1 The user does not enter required fields (missing required information (e.g. title or content)) 3. Edit Deleted Post 3.1 User chooses to edit a post previously deleted by admin
Postconditions	1. The edit form was canceled: The dialog will close without editing any posts and without saving any changes. 2. Missing Required Information: No posts are edited and the system prompts the user to fill in the missing information. 3. Edit Deleted Post: Message "Something went wrong”.
Special requirements	- A file quantity limit is applied for the post image (6 images maximum).
Extension points	- Real-time notification system for immediate friend updates.

2.12. Delete Post Use Case

Use Case ID	U011
Name	Delete Post
Brief description	User deletes previously posted post
Actors	User, Admin
Preconditions	The user has registered an account and logged in to the Forum successfully.
Basic flow	1. Users log in to the forum. 2. Search for their previously posted posts. 3. In each post, in the upper right corner there will be a ":", when the user clicks on it, options will appear (edit, delete, ...). 4. After clicking delete, a dialog box will appear asking again "are you sure you want to delete" will appear. 5. When the user chooses to confirm deletion, the post will be deleted..
Postconditions	- Post deleted successfully.
Alternative flow	1. User cancels delete confirmation 1.1 User clicks delete but decides not to click "confirm delete". 2. Delete non-existent post. 2.1 User chooses to delete post but post has been deleted by admin before
Postconditions	1. User cancels delete confirmation: The deletion confirmation form is gone and no posts are deleted. 2. Delete non-existent post: Message "Something went wrong”
Special requirements	There is no special requirements for this use case.
Extension points	There is no extension point for this use case.

2.13. Create Comment Use Case

Use Case ID	U011
Name	Create Comment
Brief description	Users comment on posts
Actors	User, Admin
Preconditions	The user has registered an account and logged in successfully.
Basic flow	1. The user logs in to the forum. 2. The user browses posts. 3. For a post, the user can comment on it. 4. When the user clicks on a comment, a dialog box appears with information about the post along with the user's comment (placed at the top if there are already comments) and other people's comments, along with a box to enter the comment. 5. When the user enters a comment and clicks the "Comment" button, the comment is posted. 6. Once the user adds a new comment, a notification is sent to their friends.
Postconditions	- Comment has been successfully created and notification will be sent to the poster.
Alternative flow	1. Missing required information: 1. 1 User does not enter any content. 2. Comment on non-existing post 2.1 User clicks to post comment but post has been deleted by admin or owner.
Postconditions	1. Missing required information: No comments are created and the system prompts the user to fill in the missing information. 2. Comment on non-existing post: Message "Something went wrong"
Special requirements	- Apply file count limit to comment images (max 3 images).
Extension points	- Real-time notification system to update information to the post owner immediately.

2.14. Update Comment Use Case

Use Case ID	U014
Name	Update Comment
Summary	User edit comment
Actors	User, Admin
Prerequisites	The user has registered an account and logged in successfully.
Basic flow	1. The user logs in to the forum. 2. The user browses posts. 3. For a post, the user can edit their comments on that post if they have commented before. 4. When the user clicks on a comment, a dialog box appears with information about the post along with the user's comment (placed at the top if there have been comments) and comments from others, along with a box to enter a comment. 5. For each previously commented comment. The user selects edit. 6. A dialog box appears with the previous comment. 7. The user edits this comment and clicks the "edit" button. 8. After clicking edit, the comment is updated
Postconditions	- Comment edited successfully.
Alternative flow	1. Canceled Comment Edit Form: 1. 1 User opens the form to edit the comment but decides not to complete the edit and cancels the action. 2. Missing required information: 2. 1 User does not enter any content when finished editing. 3. Edit comment on non-existent. 3.1 User edits comment but comment was previously deleted by admin or owner or post containing comment was deleted. 3.2 Message "Something went wrong".
Postconditions	1. Canceled Comment Edit Form: Back to post and no comments edited 2. Missing required information: Unable to complete editing and system requires full input 3. Edit comment on non-existent: Message "Something went wrong"
Special requirements	- Apply file count limit to comment images (max 3 images).
Extension points	- None

2.15. Delete Comment Use Case

Use Case ID	U015
Name	Delete Comment
Summary	User delete comment
Actors	User, Admin
Prerequisites	The user has registered an account and logged in successfully.
Basic flow	1. The user logs in to the forum. 2. The user browses posts. 3. For a post, the user can delete his/her comments on that post if he/she has commented on it before. 4. When the user clicks on a comment, a dialog box appears with information about the post along with the user's comment (placed at the top if there have been comments) and comments from others, along with a box to enter a comment. 5. For each previously commented comment. The user selects delete. 6. A dialog box appears to confirm that "Sure want to delete" 7. After confirming, the comment will be deleted.
Postconditions	- Comment deleted successfully.
Alternative flow	1. 1User cancels delete confirmation: 1.1 User clicks delete but decides not to click "confirm delete". 2. Delete comment does not exist. 2.1 User confirms to delete comment but comment has been deleted by admin or owner or because post has been deleted.
Postconditions	1. User cancels delete confirmation: No comments were deleted. 2. Delete comment does not exist: Message "Something went wrong”
Special requirements	There is no special requirements for this use case.
Extension points	There is no extension point for this use case.

2.16. Delete React Use Case

Use Case ID	U016
Name	Create react
Summary	User create comment
Actors	User, Admin
Prerequisites	The user has registered an account and logged in successfully.
Basic flow	1. The user logs in to the forum. 2. The user browses posts. 3. If the user has had a reaction on a post before, the reaction is displayed to the user. The user can delete the reaction by clicking on the reaction button. 4. After the user clicks, the reaction will be deleted.
Postconditions	- Reaction deleted successfully
Alternative flow	- Delete reaction on non-existent post: User delete reaction but the post that contains the reaction has been deleted by admin or owner.
Postconditions	- Delete reaction on non-existent post: Message "Something went wrong”
Special requirements	- There is no special requirements for this use case.
Extension points	- There is no extension point for this use case.

2.17. Create React Use Case

Use Case ID	U017
Name	Create react
Summary	User create comment
Actors	User, Admin
Prerequisites	The user has registered an account and logged in successfully.
Basic flow	1. The user logs in to the forum. 2. The user browses posts.a 3. When the user hovers over the reaction button, all types of button will be displayed. The user can create a new reaction by clicking on one of them.
Postconditions	- Reaction created successfully

Alternate flow	- Create reaction on non-existent post: User create reaction but the post that contains the reaction has been deleted by admin or owner.
Postconditions	- Delete reaction on non-existent post: Message "Something went wrong”
Special requirements	- There is no special requirements for this use case.
Extension points	- There is no extension point for this use case.

2.18. Update React Use Case

Use Case ID	**U018**
Name	Create react
Summary	User create comment
Actors	User, Admin
Prerequisites	The user has registered an account and logged in successfully.
Main flow	1. The user logs in to the forum. 2. The user browses posts. 3. If the user has had a reaction on a post before, the reaction is displayed to the user. The user can update the reaction by creating a new type of reaction.
Postconditions	- React updated successfully
Alternate flow	- Update reaction on non-existent post: User update reaction but the post that contains the reaction has been deleted by admin or owner.
Postconditions	- Update reaction on non-existent post: Message "Something went wrong”
Special requirements	- There is no special requirements for this use case.
Extension points	- There is no extension point for this use case.

2.19. Grant Permission

Use Case ID	**U021**
Name	Grant Permission
Brief description	Admin grant permission for user (update user’s role to admin)
Actors	User, Admin
Preconditions	- Admin has logged in
Basic flow	1. Admin find user by id or username 2. Update his/her role

Postconditions	- A snackbar with message “role of user” has been updated appears on admin’s screen - A notification is sent to user to notify that they are granted a new role
Alternative flow	- If user doesn’t exist, the page display a message “User doesn’t exist”
Postconditions	- Admin’s screen should appear the message
Special requirements	The granted user has to exist
Extension points	There is no extension point for this use case

2.20. Warning

Use Case ID	**U021**
Name	Warning
Brief description	Admin warn users about their violating post and delete it
Actors	User, Admin
Preconditions	- Admin has logged in - Admin is notified about the post
Basic flow	1. User posts a violating post 2. Admin is notified about the post 3. Admin reviews the post for policy violations 4. Admin warns user and deletes the post
Postconditions	- The post is deleted in database and disappear on the webpage - Warned user is notified about the activity and being saved as warned in database for one month. - A snackbar with message “User has been warned” appear on admin’s screen
Alternative flow	- If the post has been deleted, a snackbar with message “Post doesn’t exist” will appear on admin’s screen
Postconditions	- Admin click on “Home” button to go back to home
Special requirements	There’s a violating post
Extension points	There is no extension point for this use case

2.21. Logout

Use Case ID	U021
Name	Logout
Brief description	This use case describes how a user logs out the Forum Website.
Actors	User, Admin
Preconditions	User has registered an account in the system and needs to sign into the system.
Basic flow	1. User click on the avatar icon 2. User click on the Logout option 3. The system terminates the user session, clearing any session data (e.g., user information, authentication tokens). 4. The user is redirected to the login screen. 5. The system displays a message indicating the user has been logged out successfully.
Postconditions	1. The user is logged out and the session is closed. 2. The user is redirected to the login screen. 3. Any sensitive information is removed from the system. 4. The system is ready to authenticate a new user.
Alternative flow	1. User cancels the logout action (at step 1 or 2) 1.1 System return to the current user session without any changes. 2. System fails to terminate the session (due to a network issue or system's error) 2.1 System displays an error message and keep the user logged in.
Postconditions	1. User cancel the action: System remains the user at the current page without any changes. 2. System fails to terminate the session : System needs to display error message and keep the user logged in.
Special requirements	1. The logout process must be completed securely and clear all the sensitive information in cookies or sessions.
Extension points	There is no extension point for this use case

2.22. Forget Password

Use Case ID	U021
Name	Forget password
Brief description	This use case allows a user to reset their password if they have forgotten it.
Actors	User, Admin
Preconditions	1. User has registered an account in the system. 2. User has forgotten their password but knows their email address associated with the account. 3. System must be able to send emails.
Basic flow	1. User access login page 2. System displays the login form with fields for username, password, along with a “Login” button, “Remember me”, “Forgot password” 3. User selects the "Forgot Password" option. 4. System prompts for an email and user fill in the email that they have registered before and click “Send verification code” button. 5. System sends a password reset code to the provided emails. 6. System displays a form with fields for new password and verification code. 7. User enters required information and clicks “Change password” button 8. System validates information and updates database. 9. System display ‘Reset Password successfuly’ message and redirect user to login page.
Postconditions	1. New password is saved in database and the old one is deleted. 2. After resetting password successfully, system redirect user to login page. 3. User can login with new password.
Alternative flow	1. User cancels the reset password action 1.1 System redirect user to login page without any changes.
Postconditions	1. User cancel the action: System remains the user at the current page without any changes.
Special requirements	1. The sending verification email time need to less than 5 minutes.
Extension points	There is no extension point for this use case