Object Oriented Analysis and Design
Course project - Application and Recruitment Environment

Instructor: Assoc. Prof. Dr. Truong Ninh Thuan

Students:
Vu Ngoc Hien	19021268
Le Truong Giang	19021260
Nguyen Vinh Quang	1902????

October 25, 2022

Contents

List of Figures

1 Requirements

1.1 Problem statement

1.1.1 Problem

something about this shit?

1.1.2 Solution

something about this shit?

1.2 Glossary

something huh

1.3 Supplementary specifications

something huh

1.4 Use-case model

Figure 1: Use-case model

1.4.1 View profile

Description:

- As a visitor or logged in user, views the profile information of other users on the system.
- Depending on the permission level, the information that are made available is different.

Actors:

- Visitor.
- Applicant.
- Employer.
- Administrator.
- User database.

Trigger:

- Users clicking on another user's name or profile picture within the website.
- Users visiting the direct link to another user's profile page.
- Logged in users clicking on their own profile picture.

Preconditions:

- The user exists in the user database with their basic information
- The user's account is not locked.

Postconditions:

- The user is presented with the profile.
- Sections and options are provided based on authentication status, permissions and available information.
- The visit gets logged if the profile owner chose to do so (for statistical reasons).

Flow of events:

- User click on another user's feature (e.g., name, avatar) within the site (in post, feeds, suggestions).
- The system determines what data get shown and what get hidden based on the permission level and authentication status.
- The information is displayed.

Alternative flows:

- Flow 1:
  - The user clicks on their own profile.
  - The page is displayed with all available information with ability to edit the information stored.

- Flow 2:
  - The user clicks a link from a different website linking to the user profile.
  - The system determines what data get shown and what get hidden based on the permission level and authentication status.
  - The information is displayed.

Exception flows:

- Flow 1:

- The owner blocked public visitors from seeing their profile.
- The site displays an error message.

- Flow 2:
  - User clicking a link.
  - The user does not exist in the database.
  - The site displays an error message.

1.4.2 Create Account

Description:

- As a visitor, create an account on the system in order to access its personalized features, or features that requires identity confirmation.

Actor:

- Visitor.
- Authentication backends.

Trigger:

- User wanting to logs into the system when they don't have an existing account.

Precondition:

- User does not have an account.
- User has internet connection.

Postcondition:

- User gets logged into the application.
- Basic user authentication information is written into the database.

Flow of events:

- User accesses the website
- User intentionally opt to register / User attempts to access a website part that requires authentication
- User inputs basic information such as their email address, name and password of choice
- Authentication systems sent an email to the chosen inbox.
- User verifies by opening their email inbox and click the link
- The system saves the user's sign in information and ask for extra information as needed

Alternative flows: None.
Exceptions:

- User does not have internet connection.
  - User cannot register.
- User already have an account associated with an email.
  - Notifies user and offer to reset password.
- User fails to confirm their password.
- User's email is blocked from registering.

Business rules:

- Deny account registration if there are more than three registration attempts from the same computer within the span of a day (24 hours).

Non-functional requirements:

- User passwords are hashed with a salt for security.

1.4.3 Login

Description:

- As an existing user (applicant, employer or admin), login to an existing account.

Actor:

- Applicant.
- Employer.
- Admin.
- Authentication backends.

Trigger:

- User choosing to login on the interface
- User visiting contents that requires authentication

Precondition:

- User already have an account in the authentication database.
- User account have proper permissions to login.
- User has internet connection.

Postcondition:

- User gets logged in to the account.
- The activity is logged into the user's Activity Log.

Flow of events:

- User choose to login from the user interface or access a page that requires authentication.
- The system checks for any existing session token on the device.
- The user provides their email address and password of their choosing.
- The system compares the information the user provides with what is stored in the authentication database.
- The system logs the user in and set a session token on the device.

Alternative flows: None.
Exceptions flow:

- If the system found an existing session token on the device, instead of prompting the user for the account info, logs the user in if the token is valid.
- If the user-provided information is not correct, prompt the user and offer password reset.
- If the account is blocked from login, notifies the user and sign out automatically.
- If the user does not have internet connection before the data is sent to the server, the system will cache the data.

1.4.4 Update profile

Description:

- User go to profile page and update information. New information will be saved and displayed on profile page after user click update button.

- This feature apply to both employer and applicant.

Actor:

- Applicant or employer.

Trigger:

- User wanting to update profile information.

Precondition:

- User has an account (not administrator account).

- User logged in.

- User has internet connection.

Postcondition:

- User profile information is updated.

Flow of events:

- User go to profile page.

- User update new information. User can add new section information (e.g. education, experience, etc.) or update existing information.

- User click update button.

- System save new information.

- System display new information on profile page.

Alternative flow:

- User click cancel button.

- System display old information on profile page.

Exception flow:

- User has no internet connection.

- System display error message.

Business rules:

- User can only update information of his/her own profile.

Non-functional requirements: None

1.4.5 Apply into job description post

Description:

- User go to job description post page and apply for the job. User can apply for multiple jobs.

- This feature apply to applicant only.

Actor:

- Applicant.

- Backend system.

Trigger:

- User wanting to apply for a job in a job description post.

Precondition:

- User has an applicant account.
- User logged in.
- User has internet connection.

Postcondition:

- User apply for a job in a job description post.

Flow of events:

- User go to job description post page.
- User click apply button.
- System shows a textbox for user to write a cover letter or simply a message/comment (optional).
- System save user's application.
- System send notification to employer and applicant to confirm.

Alternative flow:

- User click cancel button.
- System display job description post page.

Exception flow:

- User has no internet connection.
- System display error message.

Business rules:

- Each user can apply for only onetime in a job description post.

Non-functional requirements: None

1.4.6   Interact with job description

Description:

- User go to job description post page and interact with the job description post. User can like, dislike, comment the job description post.
- This feature apply to both applicant and employer.

Actor:

- Applicant or employer.
- Backend system.

Trigger:

- User wanting to interact with a job description post.

Precondition:

- User has an account (not administrator account).
- User logged in.
- User has internet connection.

Postcondition:

- User's interaction with a job description post is saved and displayed.

Flow of events:

- User go to job description post page.
- User click like, dislike button or leave a comment.
- System save user's interaction.
- System display user's interaction on job description post page.

Alternative flow:

- User stop commenting on job description post.
- System display job description post page.

Exception flow:

- User has no internet connection.
- System display error message.

Business rules:

- User can only have one like or dislike on a job description post.
- User can leave multiple comments on a job description post.

Non-functional requirements: None

1.4.7 Post job description

Description:

- Employer click create recruitment announcement and post a job description. Employer can post multiple job descriptions.
- This feature apply to employer only.

Actor:

- Employer.
- Backend service and database.

Trigger:

- Employer wanting to post a job description for recruitment.

Precondition:

- Employer has an employer account.
- Employer logged in.
- Employer has internet connection.

Postcondition:

- Employer post a job description for recruitment.

Flow of events:

- Employer click create recruitment announcement button.
- Employer fill in job description information.
- Employer click post button.

- System save job description information.

- System display job description post page.

Alternative flow:

- Employer stop editing job description information and close browser/tab or direct to another site or press cancel button.

- System display a warning message to ask employer weather to save a draft or not.

- Employer choose to save a draft or not. If yes, system save job description information as a draft in user's local storage. If no, system discard job description information.

Exception flow:

- Employer has no internet connection.

- System auto save job description information as a draft in employer's profile page by caching the draft in employer's device.

- System display error message.

Business rules:

- Employer can only post one job description in a recruitment announcement.

- Employer can post multiple job descriptions in multiple recruitment announcements.

1.4.8 View job description post

Description:

- User go to job description post page and view the job description post.

Actor:

- Applicant.

- Employer.

- Administrator.

- Visitor.

- Backend service and database.

Trigger:

- User wanting to view a job description post.

- User search for a specific job description post.

Precondition:

- User has internet connection.

- Database service is available.

Postcondition:

- Job description post is displayed.

Flow of events:

- User goes to the website.

- User type a keyword related to the job in the search box.

- The website displays several job descriptions posts.

- User selects a job description post.

- The website displays the full job description post.

Alternative flow:

- Flow 1:
    - User goes to the website.
    - User type a keyword related to the job in the search box.
    - The website displays several job descriptions posts.
    - User selects a job description post.
    - The website displays the full job description post.

- Flow 2:
    - User click on a link to a job description post.
    - The website displays the full job description post.

Exception flow:

- Flow 1:
    - User cannot find any job description posts.
    - Display a notification message.

- Flow 2:
    - User accesses a deleted job description post.
    - Display an error message.

Business rules:

- User cannot access a deleted job description post.

1.4.9 Edit Job description post

Description:

- Employer wants to change the content of a job description post.

Actor:

- Employer.
- Backend service and database.

Trigger:

- Employer wants to change the content of a job description post.

Precondition:

- Employer has an employer account.
- Employer logged in.
- Employer has internet connection.
- Employer is viewing his/her job description post.

Postcondition:

- Employer has changed the content of a job description post.

Flow of events:

- User accesses the website.
- User selects a job description post.

- User click edit button.

- User modifies the content of the job description post.

- User saves the changes.

- The change has updated on the system.

- System display job description post page.

Alternative flow:

- User accesses the website.

- User selects a job description post.

- User click edit button.

- User modifies the content of the job description post.

- User close browser/tab or direct to another site or press cancel button.

- System display a warning message to ask user if the user wants to keep editing. If yes, the browser cancel closing/redirect keep editing. If no, system discard the changes.

Exception flow:

- User attempt to open more than 1 editor for 1 post at a time point.

- Display an error message.

Business rules:

- User can only open an editor for one post at a time point.

- User cannot edit other employer's job description posts.

1.4.10   Delete Job description post

Description:

- User, as an employer, delete a job description posts.

- User, as an administrator, delete a post that is considered not a job description.

Actor:

- Employer.

- Administrator.

- Backend service and database.

Trigger:

- A job description post has expired or no longer needed.

Precondition:

- User has an internet connection.

- User has an employer/administration account.

- User logged in.

- The employer owned the job description post (this is not required for administration).

Postcondition:

- A job description post is removed from the database.

Flow of events:

- User accesses the website.

- User selects a description post or view a description post.

- User selects the remove option.

- The system ask for confirmation.

- The description post is removed from the database.

Exception flow:

- User attempt to delete a deleted job description post.

- Display an error message.

Business rules:

- User can use only one delete request on one job description post.

- User cannot delete other employer's job description post (except administrator).

1.4.11 Delete Account

Description:

- The user can delete their account along with their associated data from the respective databases of the site.

- An administrator can delete a user's account upon their request or due to site rules violations.

Actor:

- Employer.
- Applicant.
- Administrator.
- Backend authentication.

Trigger:

- The user selecting the option to delete their account.

- An admin deleting an account from administration tools.

Precondition:

- The user is logged in and able to access their account settings.

- An administrator is authorized to manage user accounts.

- Employer or Applicant has internet connection.

Postcondition:

- The user login info and user data are deleted from all systems.

- The user may or may not re-register an account with the same email.

Flow of events:

- The user logs into their account.

- The user opens the account settings.

- The user chooses to delete their account.

- The user confirms with their password.

- The system verifies the password and ask for confirmation.

- The system deletes the user's data and authorization information.

Alternative flow:

- An admin logs into his/her account.

- The admin selects a user account and choose the option to delete it.

- The system asks if the user should be able to recreate an account with the same email.

- The system asks for confirmation.

- The system logs the deletion in the Audit log and sets a flag if the user is denied from re-registering.

Exception flow:

- Flow 1:

  - The user has no internet connection.
  - The system displays an error message.

- Flow 2:

  - The user fails to verify their password upon deletion.

1.4.12 Searching tool

Description:

- User search information from the website about job descriptions, employers, and applicants.

Actor:

- Applicant.

- Employer.

- Administrator.

- Visitor.

- Backend search engine.

Trigger:

- User wants to search for information about a typical job, an employer, or an applicant.

Precondition:

- User has internet connection.

Postcondition:

- The website displays several results about the information user is looking for.

Flow of events:

- User accesses the website.

- User clicks the search bar and type keywords in the search box.

- The website displays several results about the information user is looking for.

Alternative flow:

- User click on a link to a search result.

- The website displays several results about the information user is looking for.

Exception flow:

- The search engine does not find any information.

- The system displays a message not finding any job description post or employer or applicant.

Business rules:

- The search engine will search for information in the database.

- The search engine will search for information in the job description post, employer, and applicant.

2 Use-case realisations

2.1 Sequence diagrams
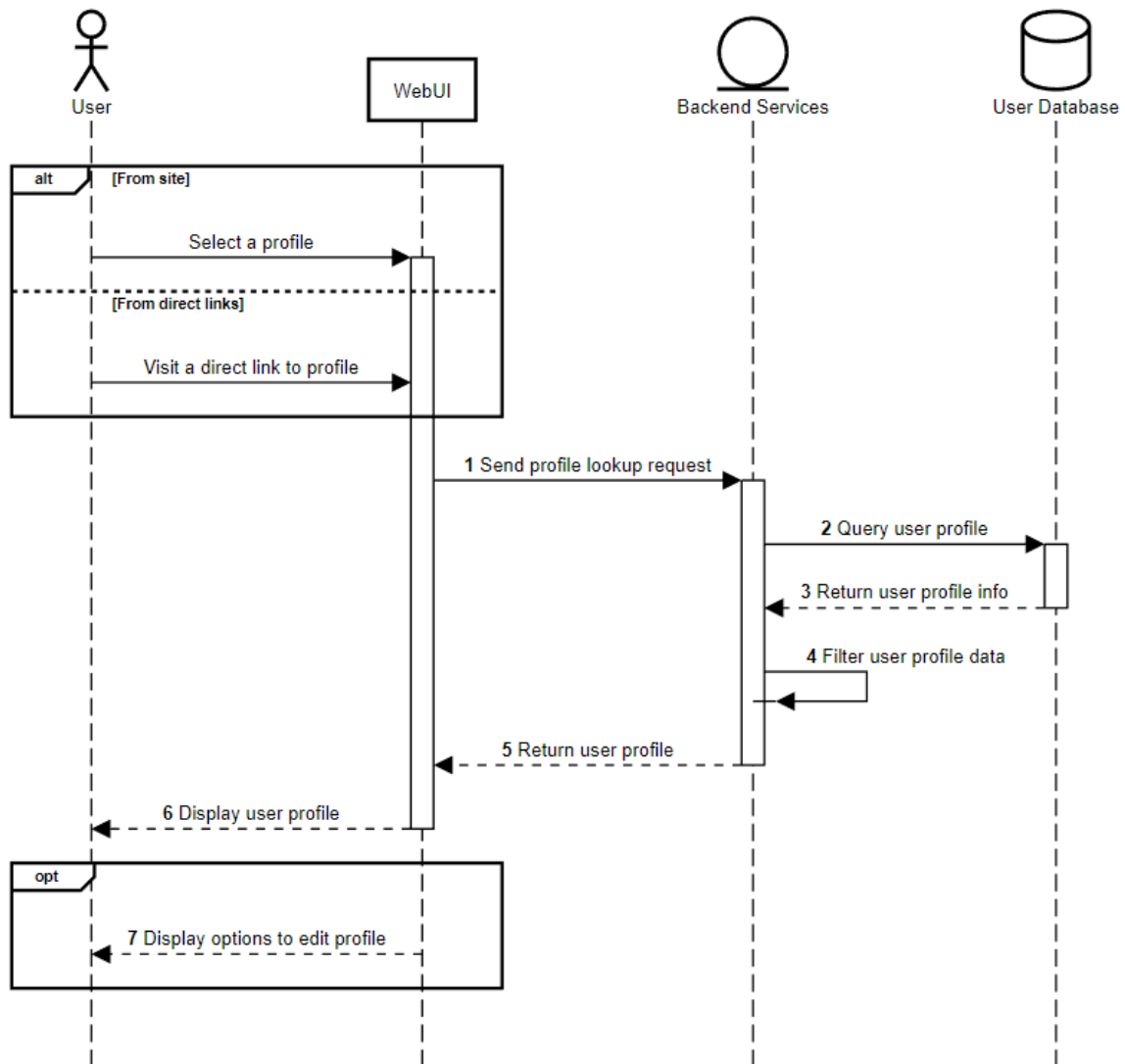
Figure 2: Sequence diagram for the use profile use case (basic flow)
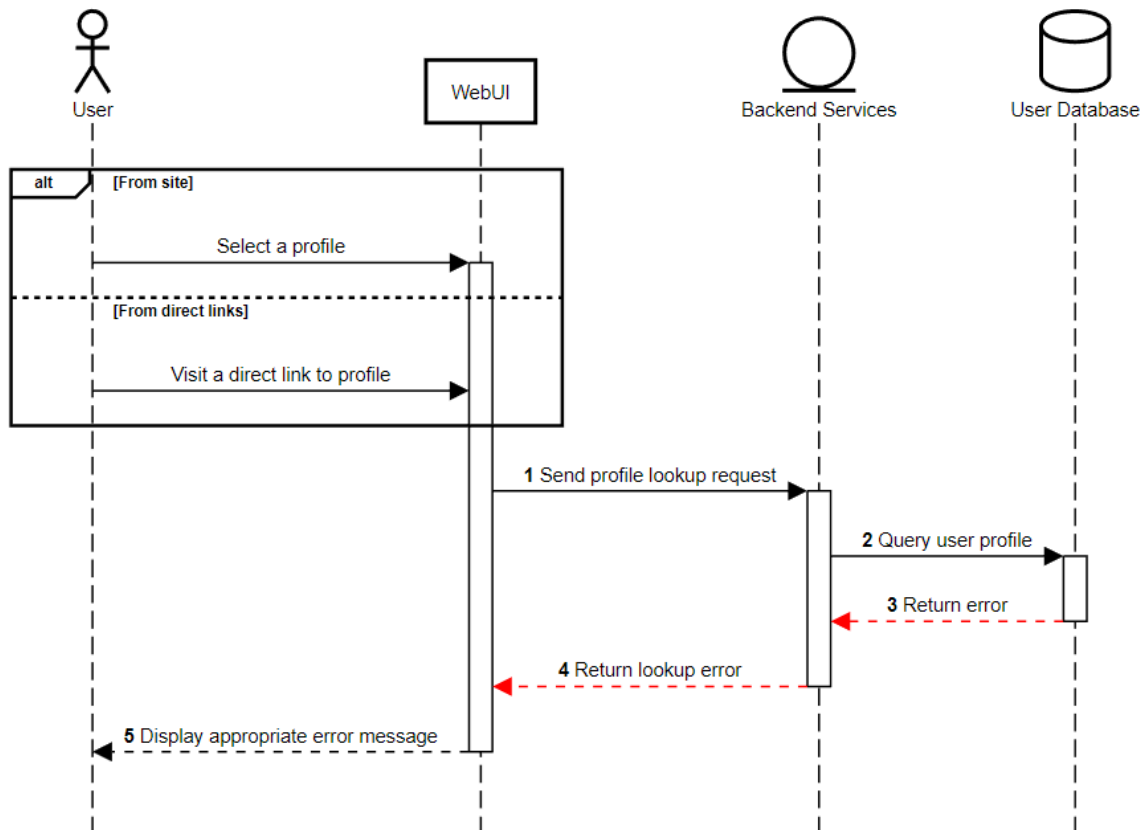
Figure 3: Sequence diagram for the use profile use case (exception flow)
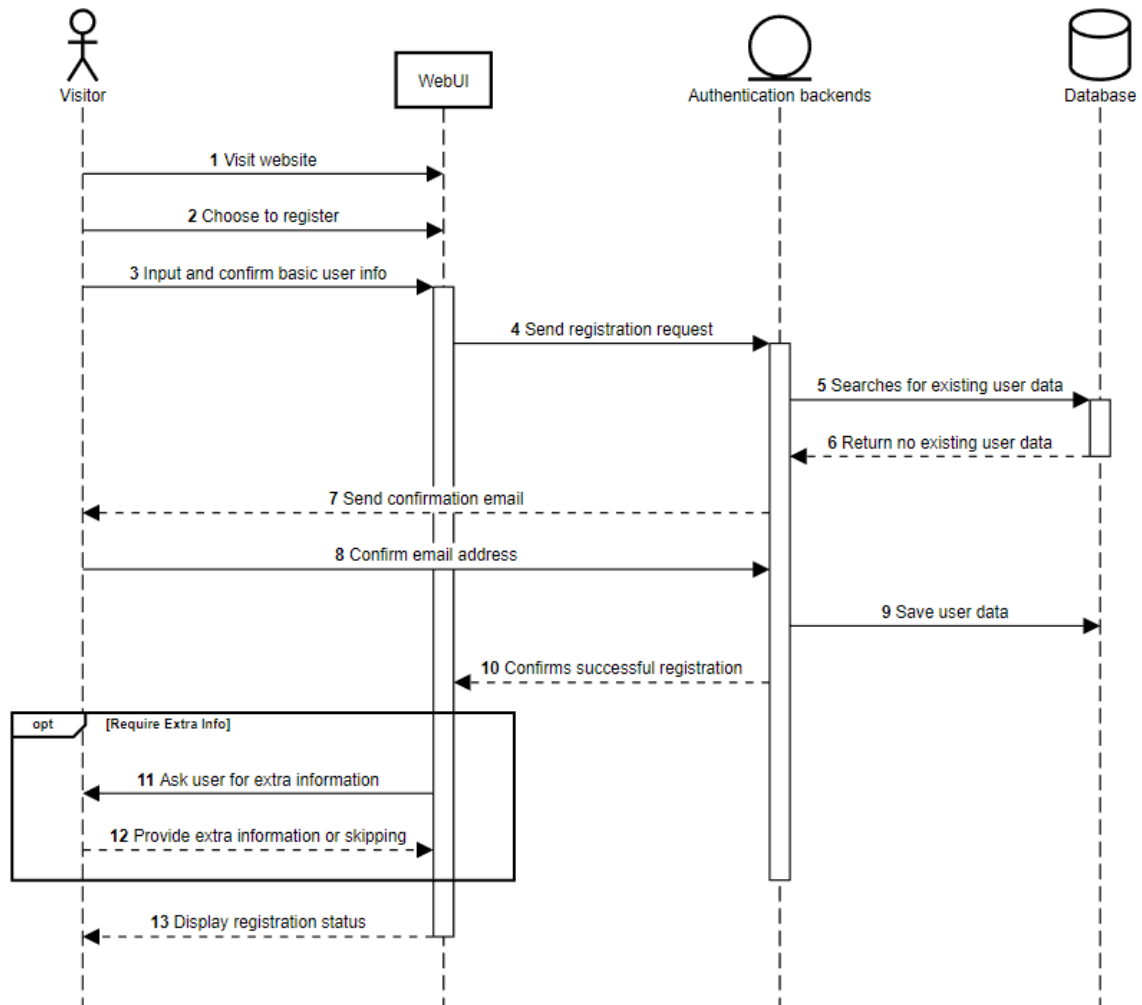
Figure 4: Sequence diagram for the create account use case (basic flow)
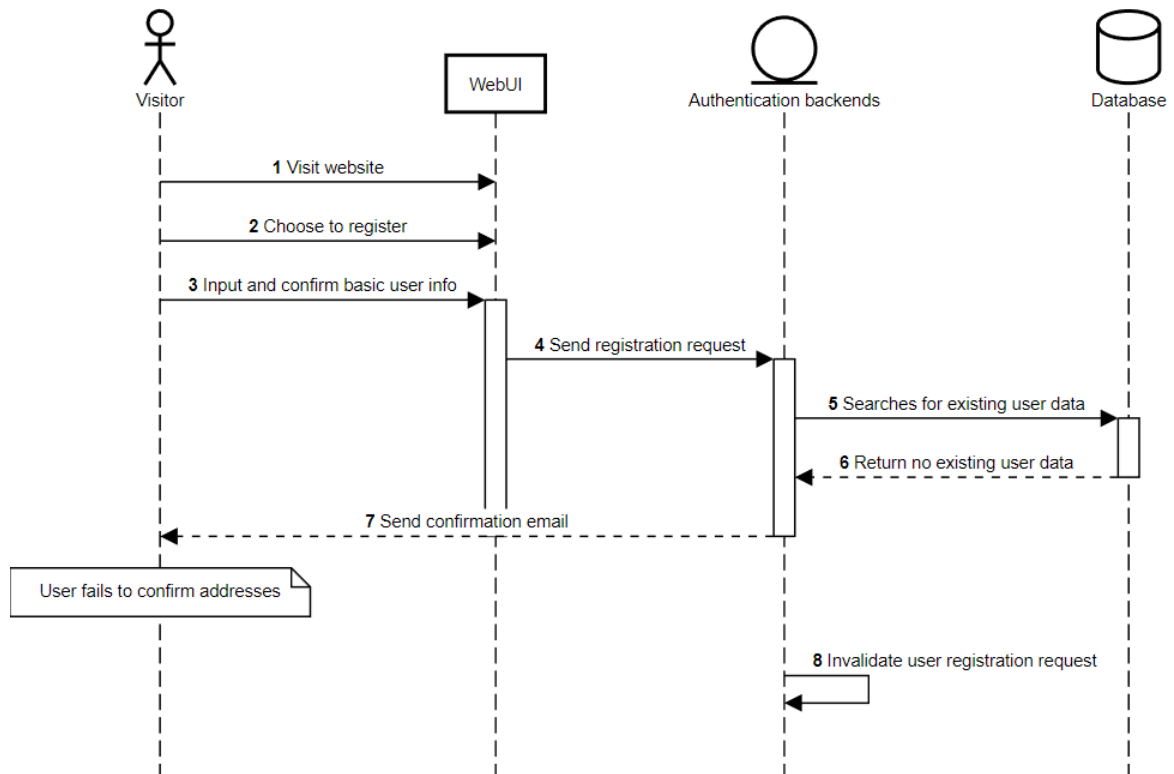
Figure 5: Sequence diagram for the create account use case (unconfirmed email exception)
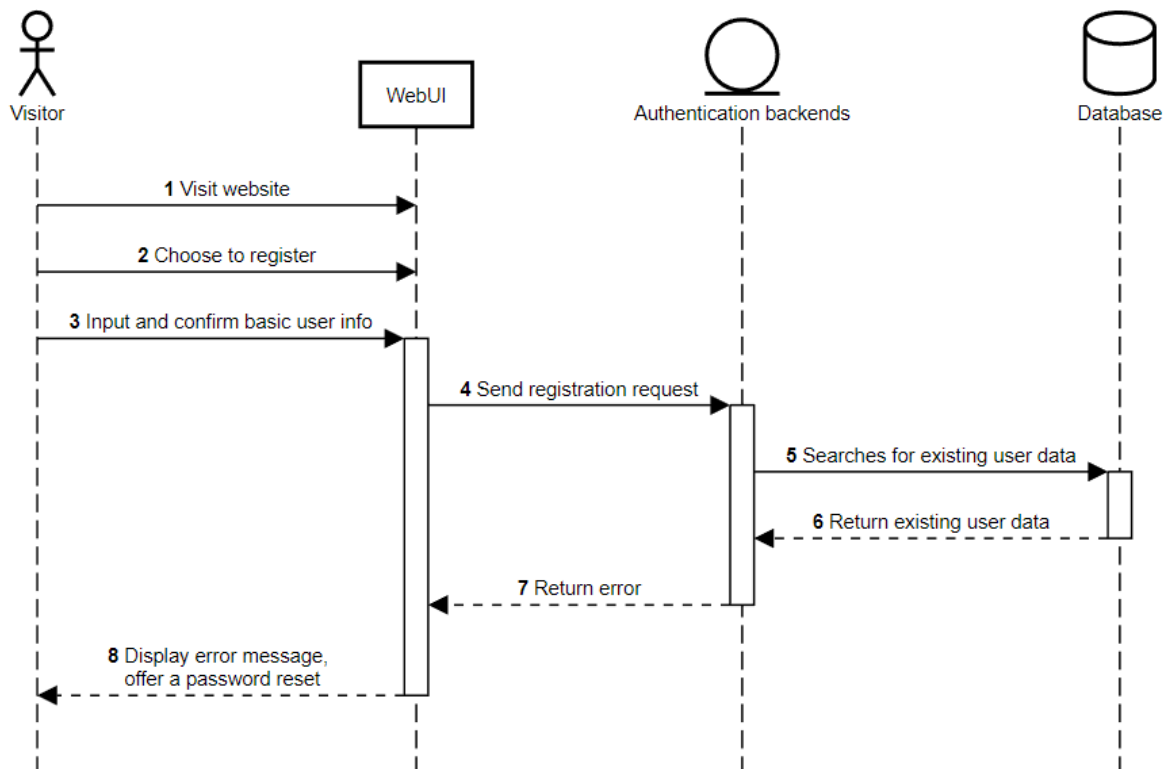
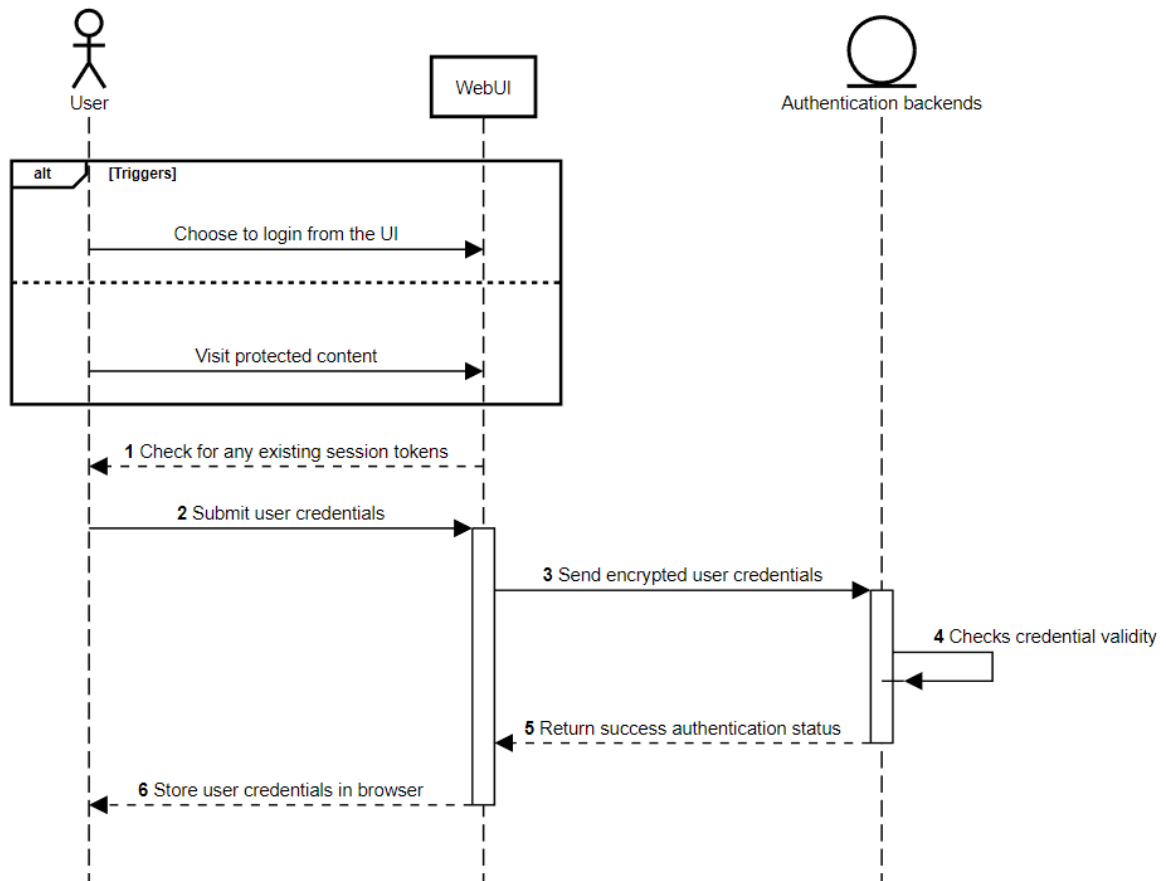**Figure 6:** Sequence diagram for the create account use case (user exist exception)

Figure 7: Sequence diagram for the login use case (basic flow)
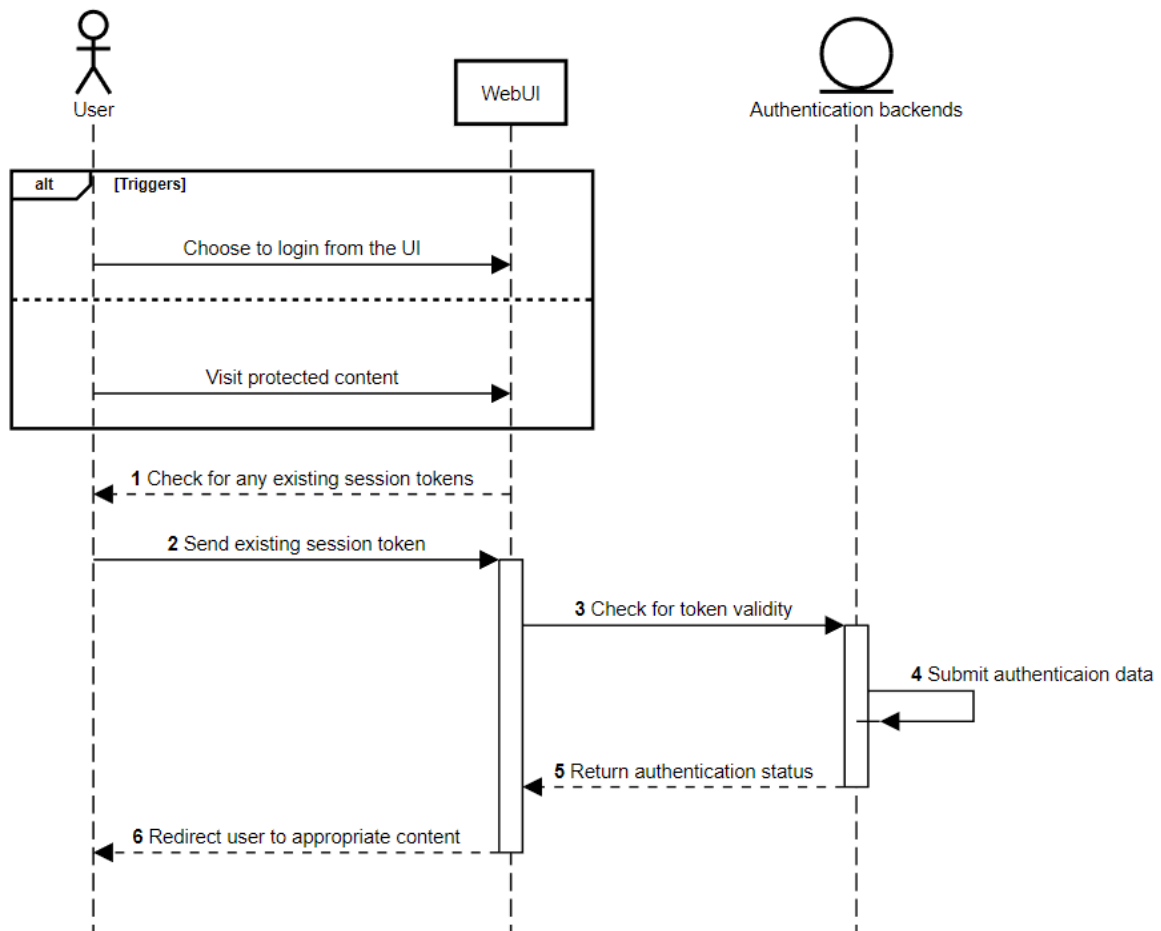
Figure 8: Sequence diagram for the login use case (alternative flow)
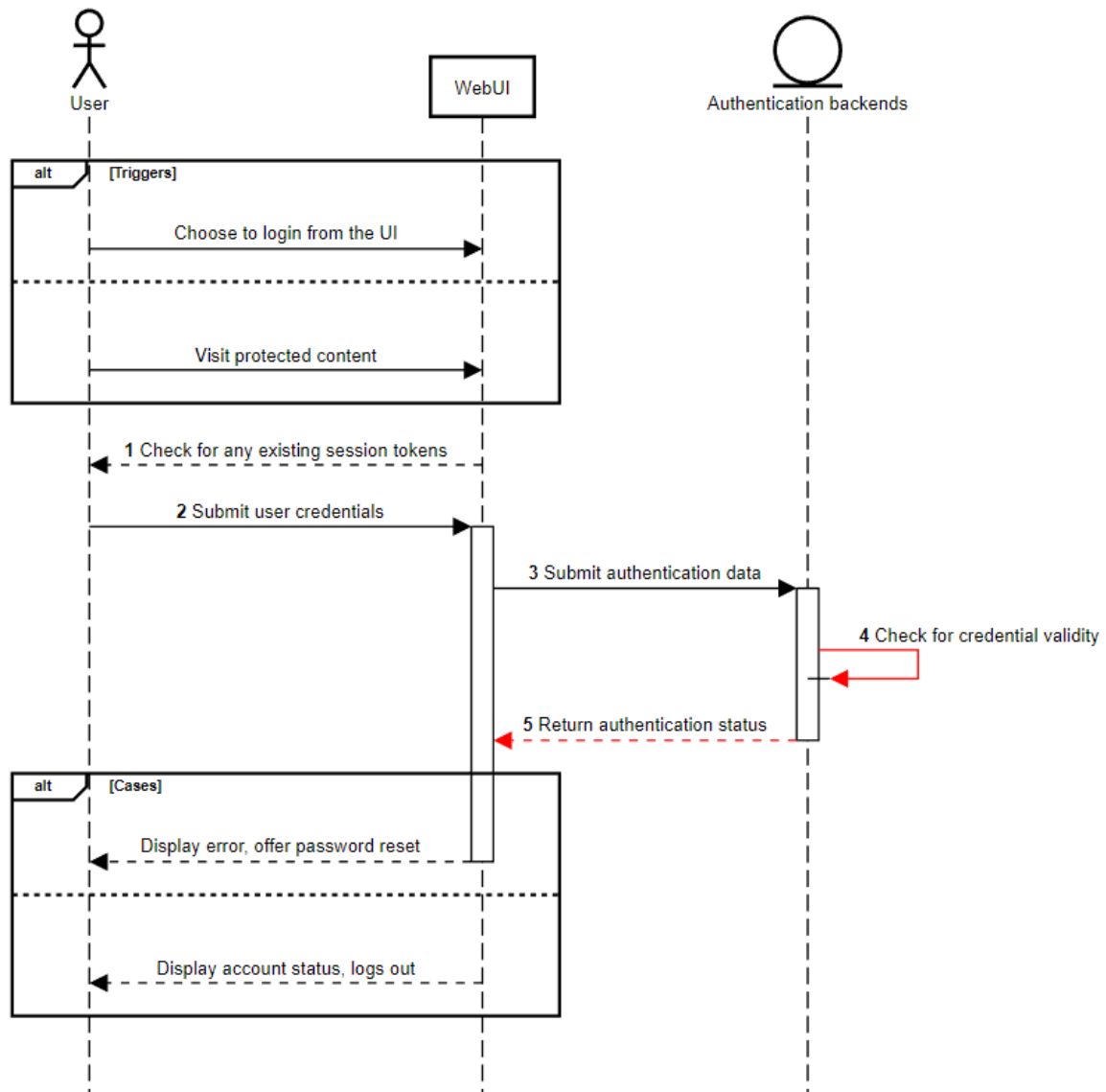
Figure 9: Sequence diagram for the login use case (exception flow)
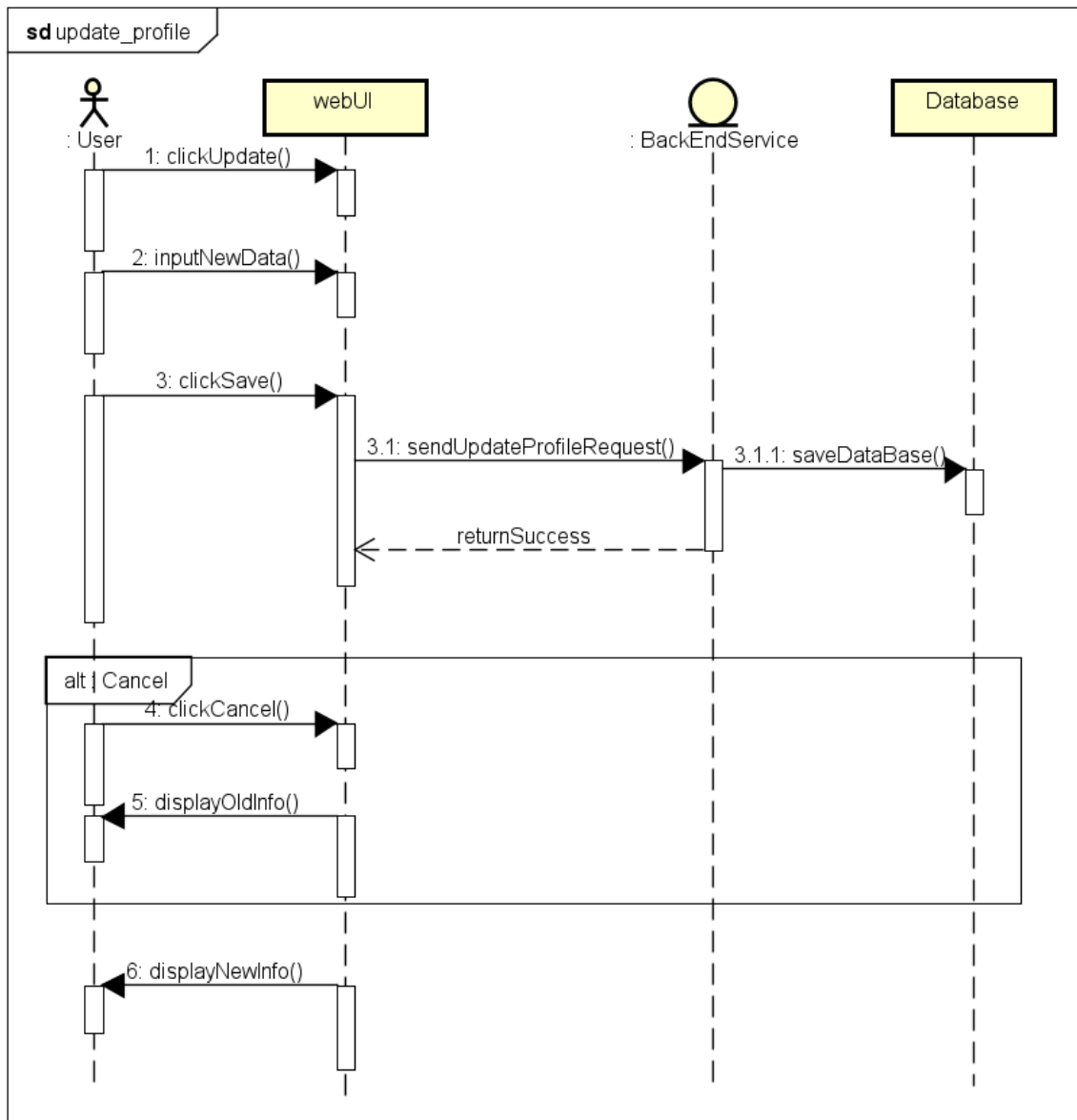
Figure 10: Sequence diagram for the update profile use case
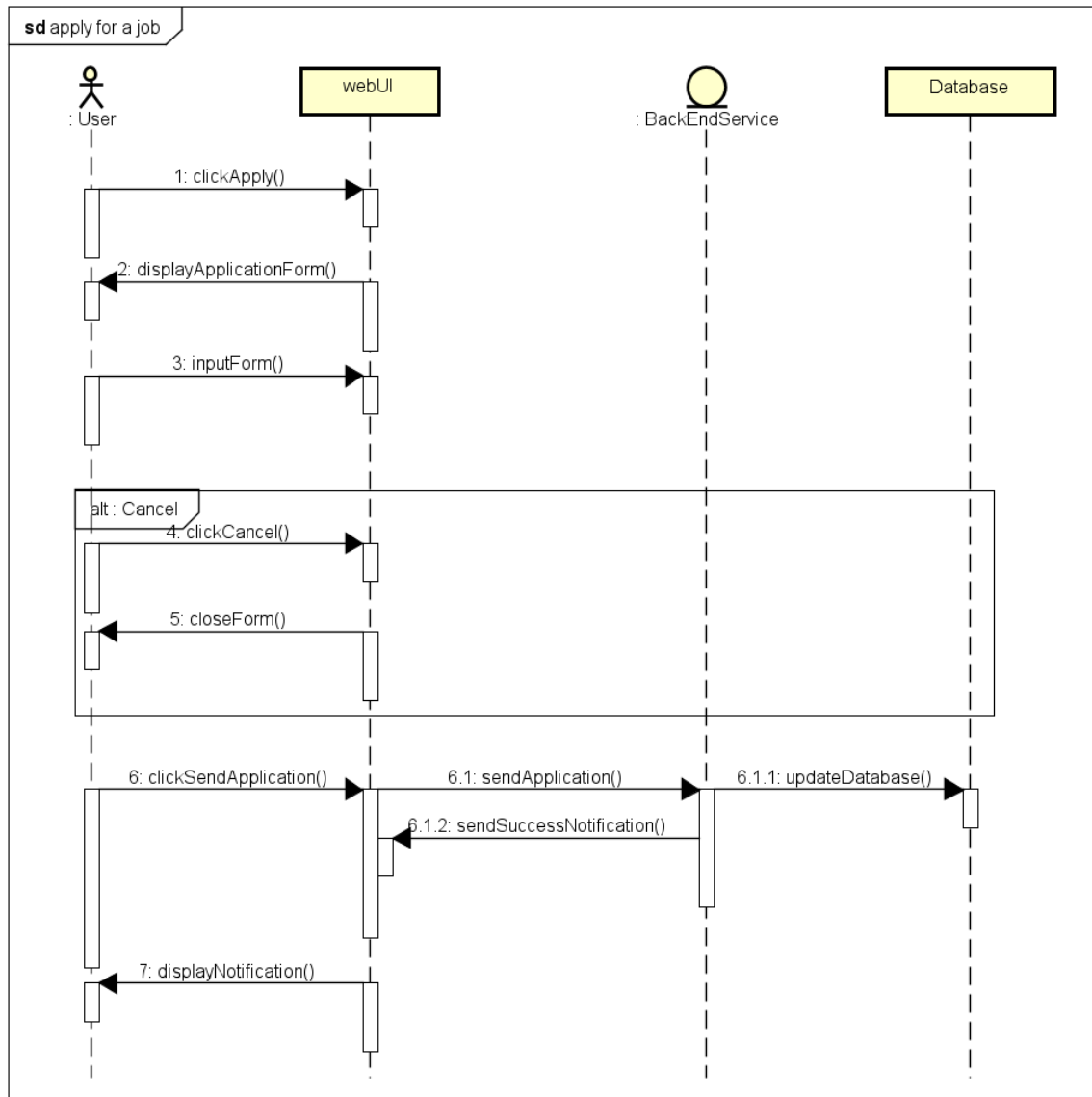
Figure 11: Sequence diagram for the job description application use case
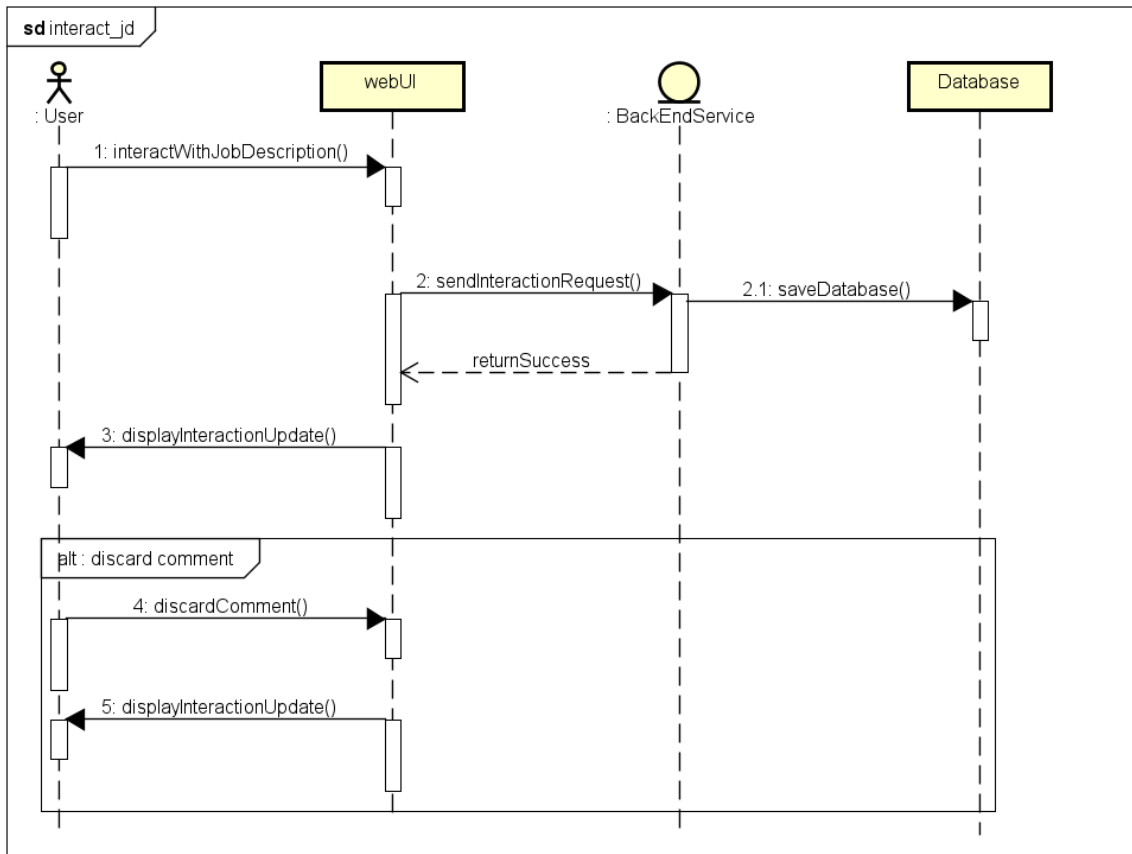
Figure 12: Sequence diagram for the job description interaction use case
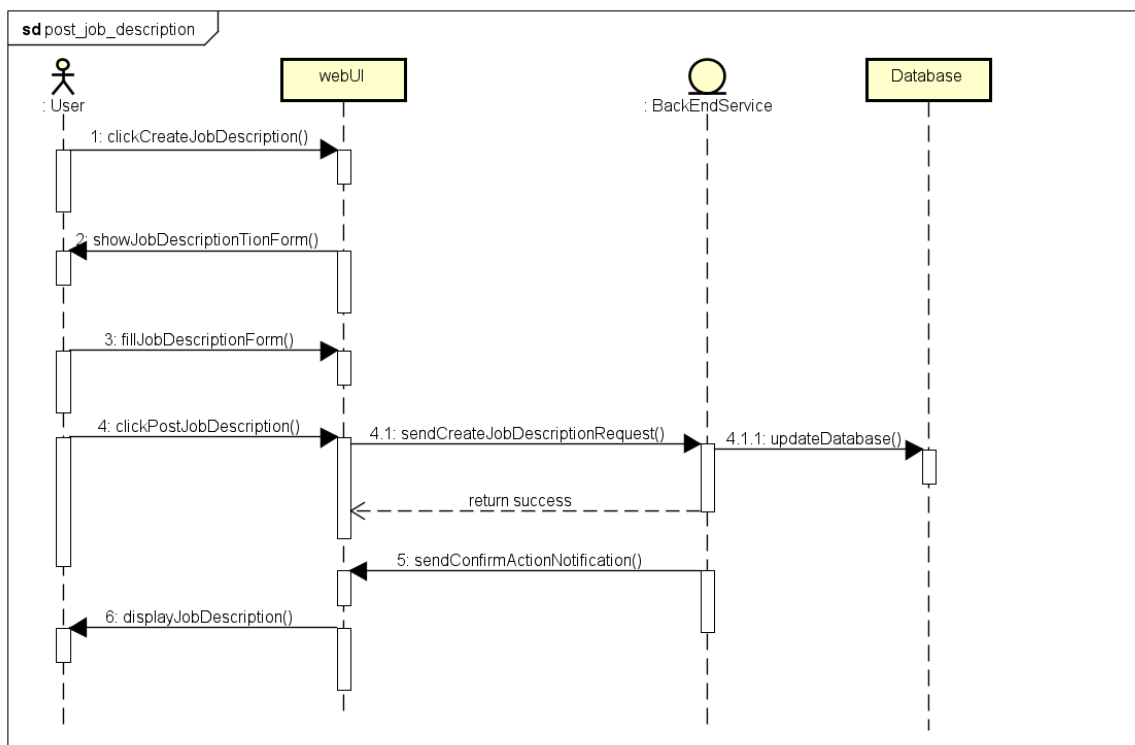
Figure 13: Sequence diagram for the post job description use case

View Job description post
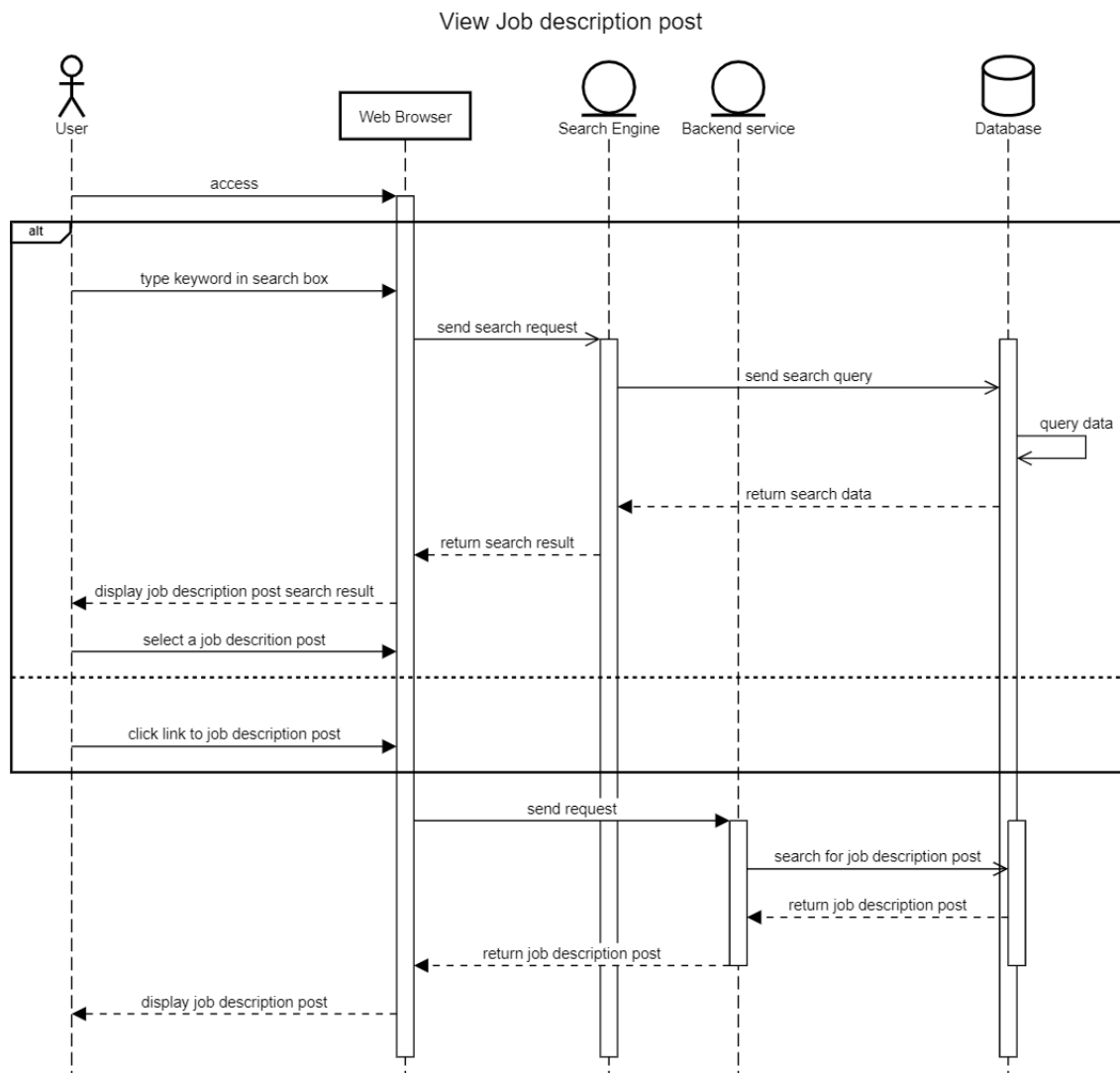
Figure 14: Sequence diagram for the view job description use case

Edit job description post

User | WebUI | Backend service | Database

access

select a job description post

choose edit button

modify content

alt [cancel]

close / direct to other site

display warning message

[save changes]

save the changes

send request

send update query

update data

alt [success]

return successful

return successful message

display successful message

[error]

return error

return error message

display error message

Figure 15: Sequence diagram for the view job description use case
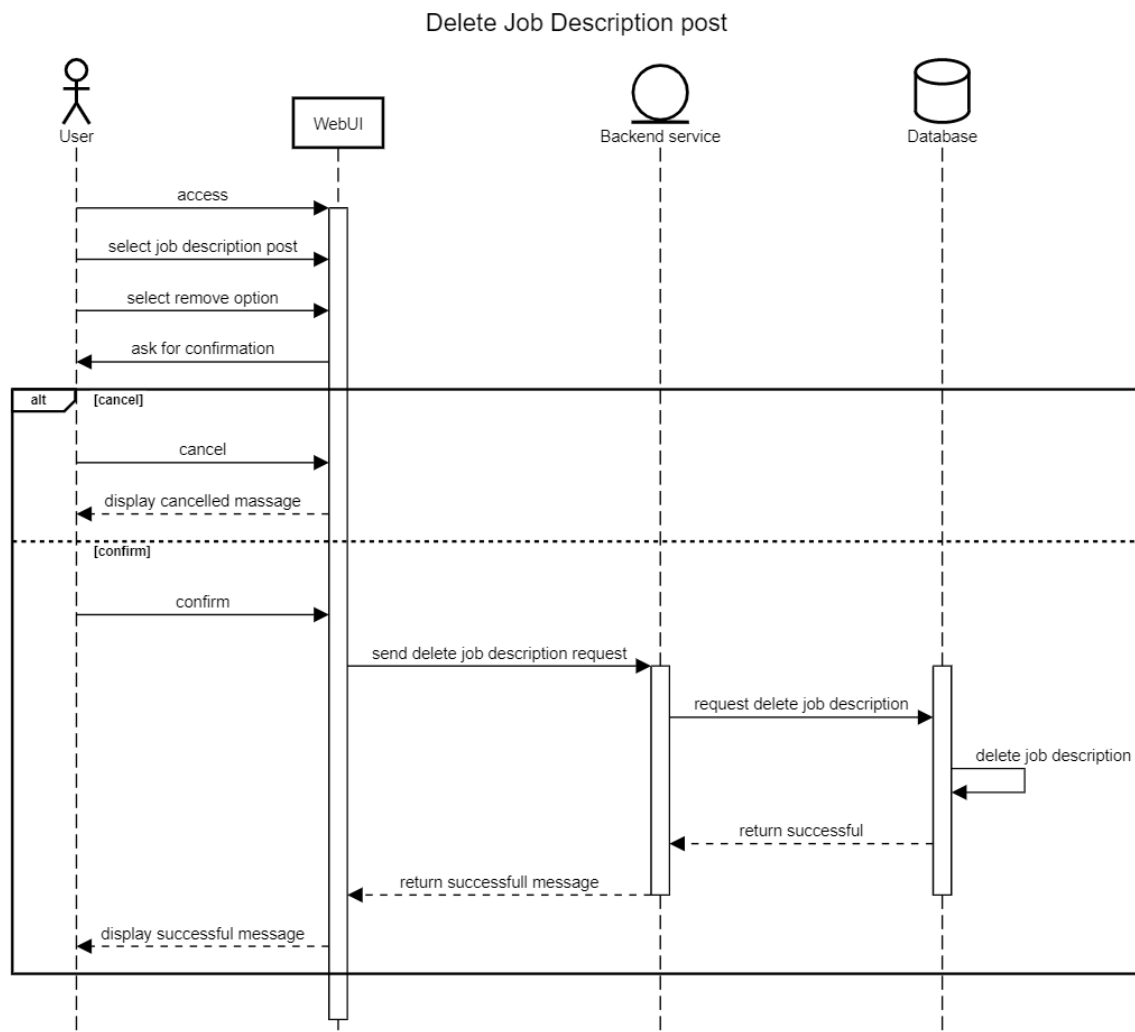
Figure 16: Sequence diagram for the delete job description use case
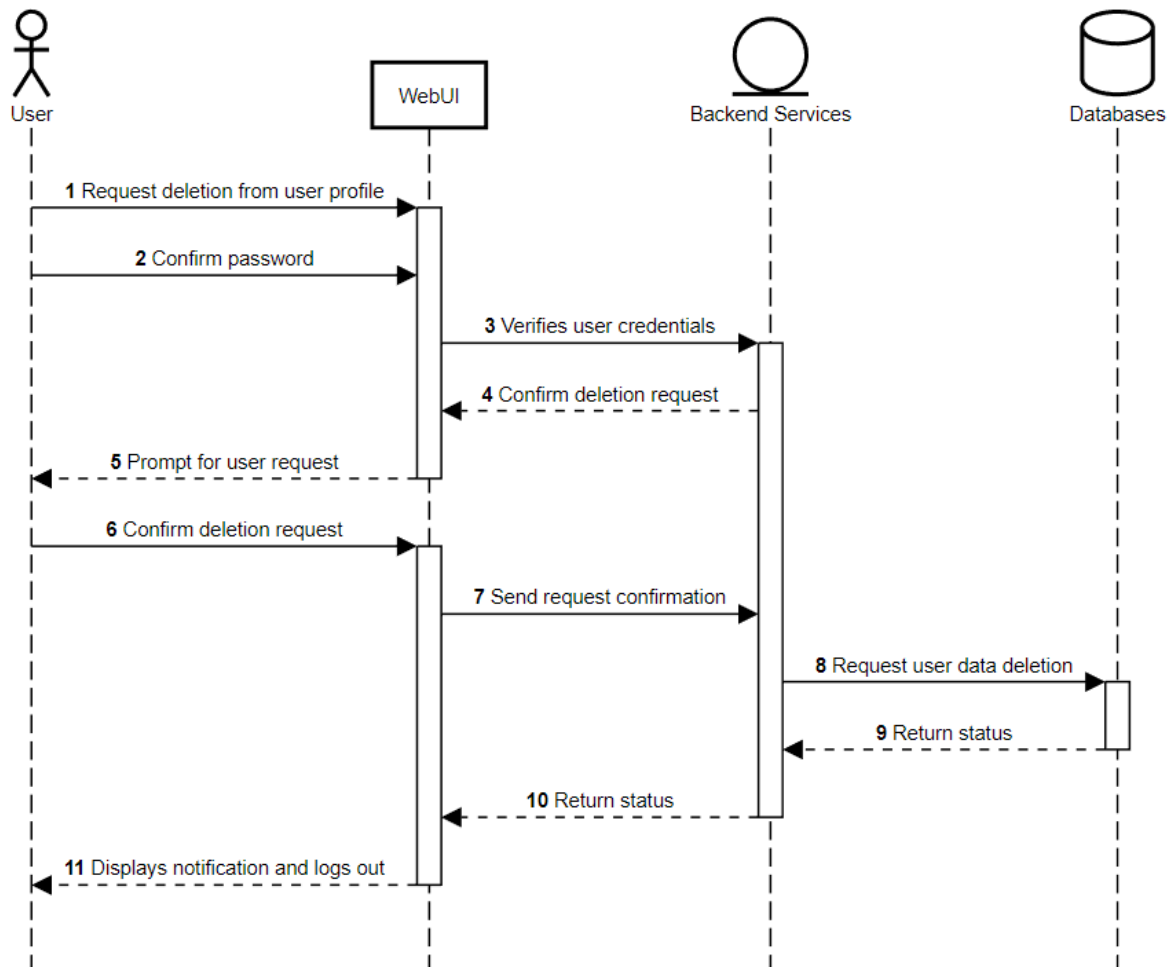
Figure 17: Sequence diagram for the delete account use case (basic flow)
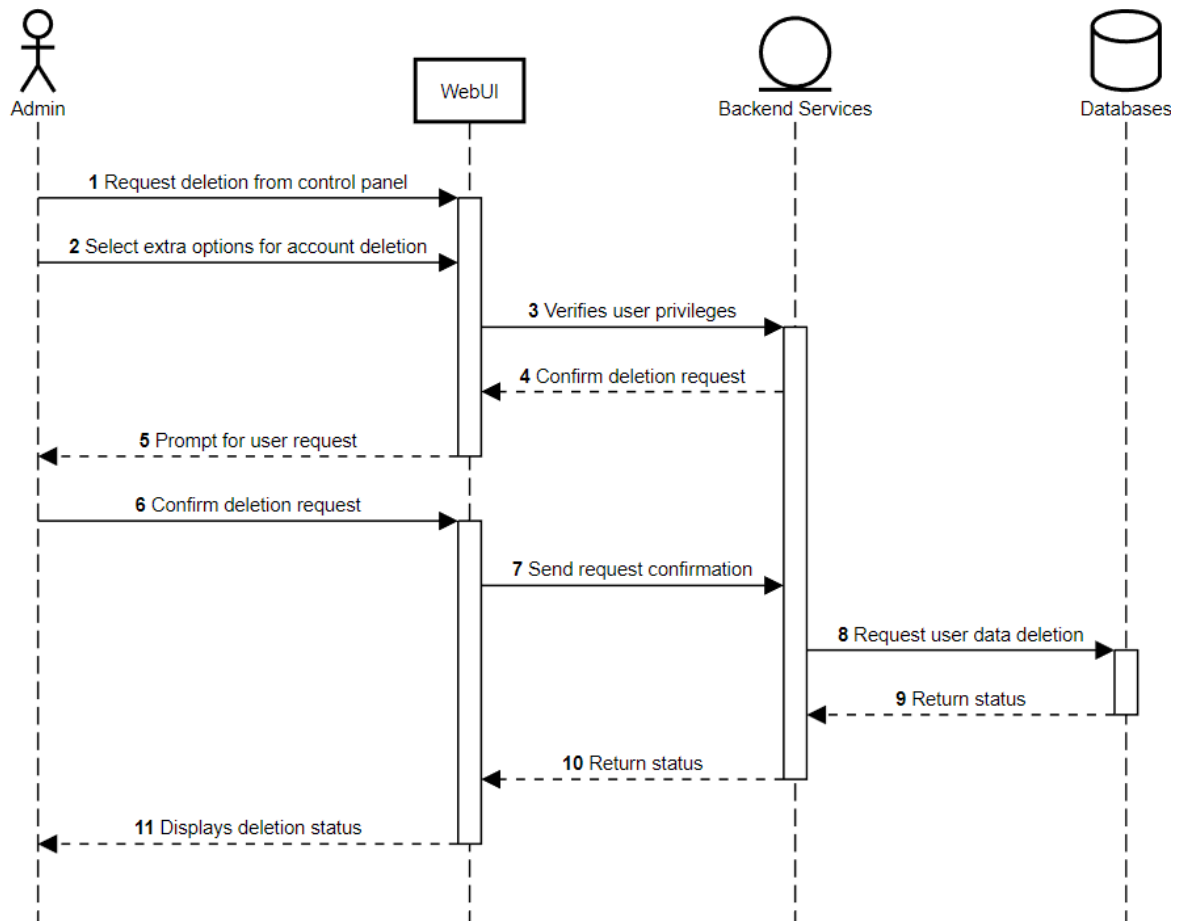
Figure 18: Sequence diagram for the post job description use case (admin alternative)
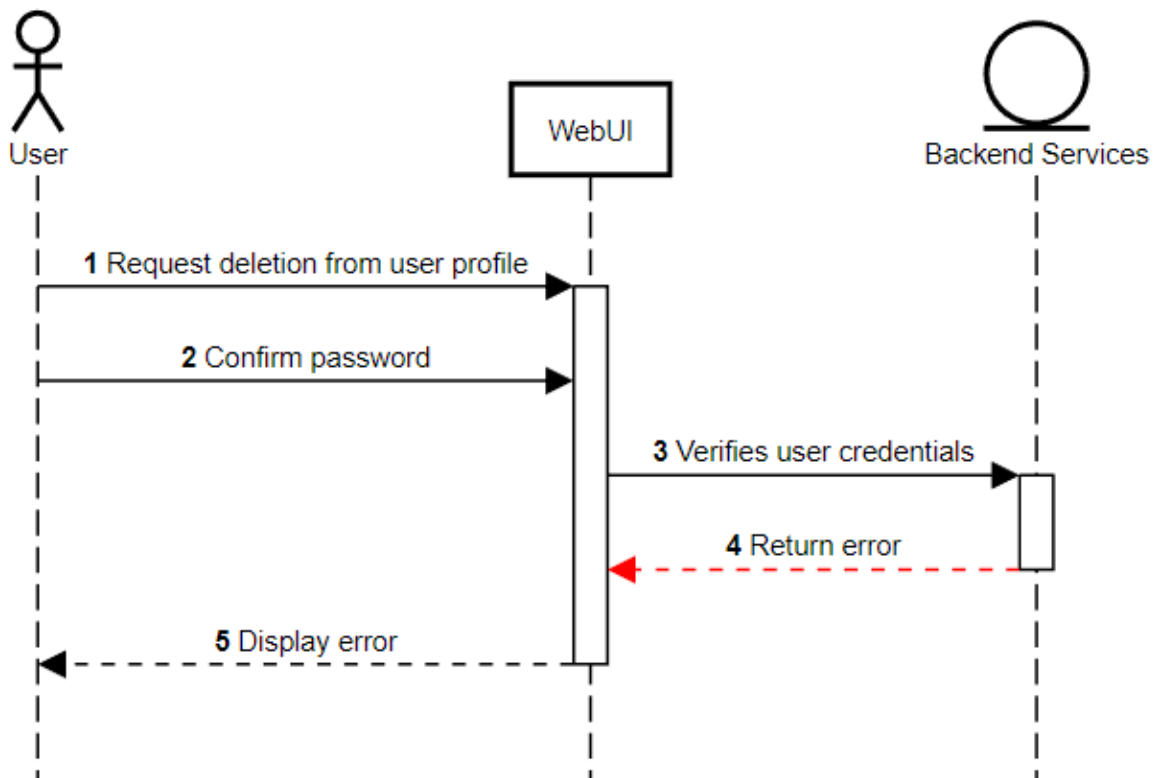
Figure 19: Sequence diagram for the delete account use case (exception)
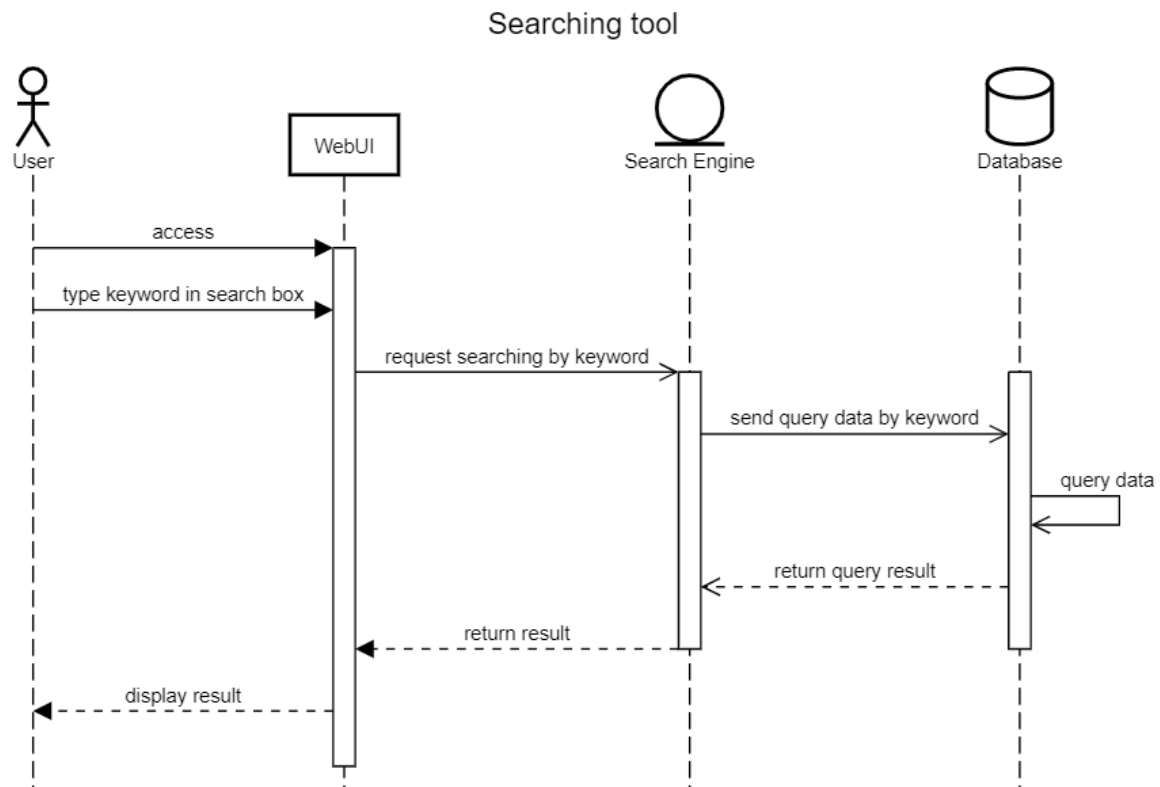
Figure 20: Sequence diagram for the search engine use case

2.2 Views of participating classes
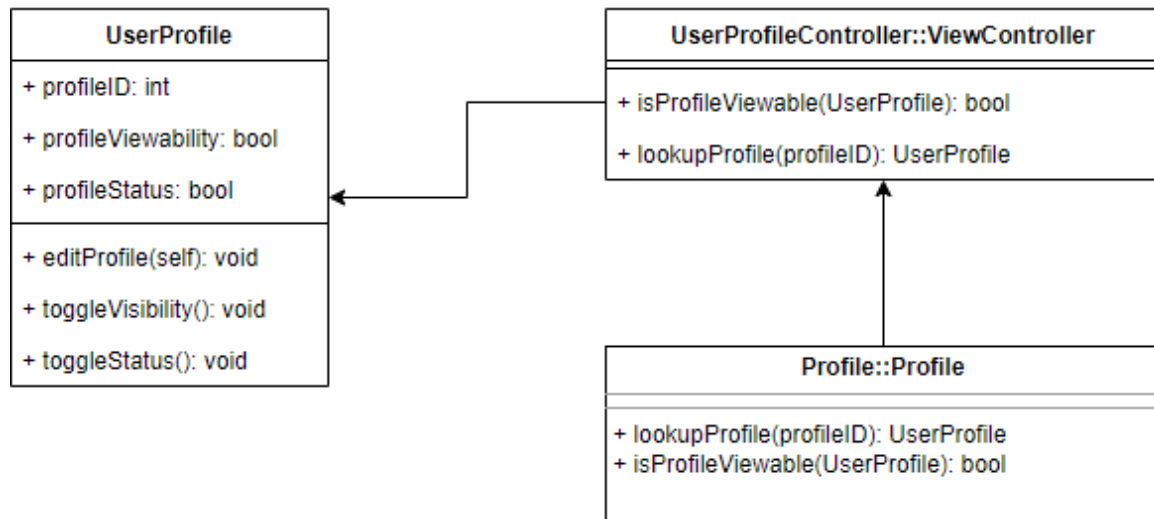
Figure 21: Vopc of view profile use case
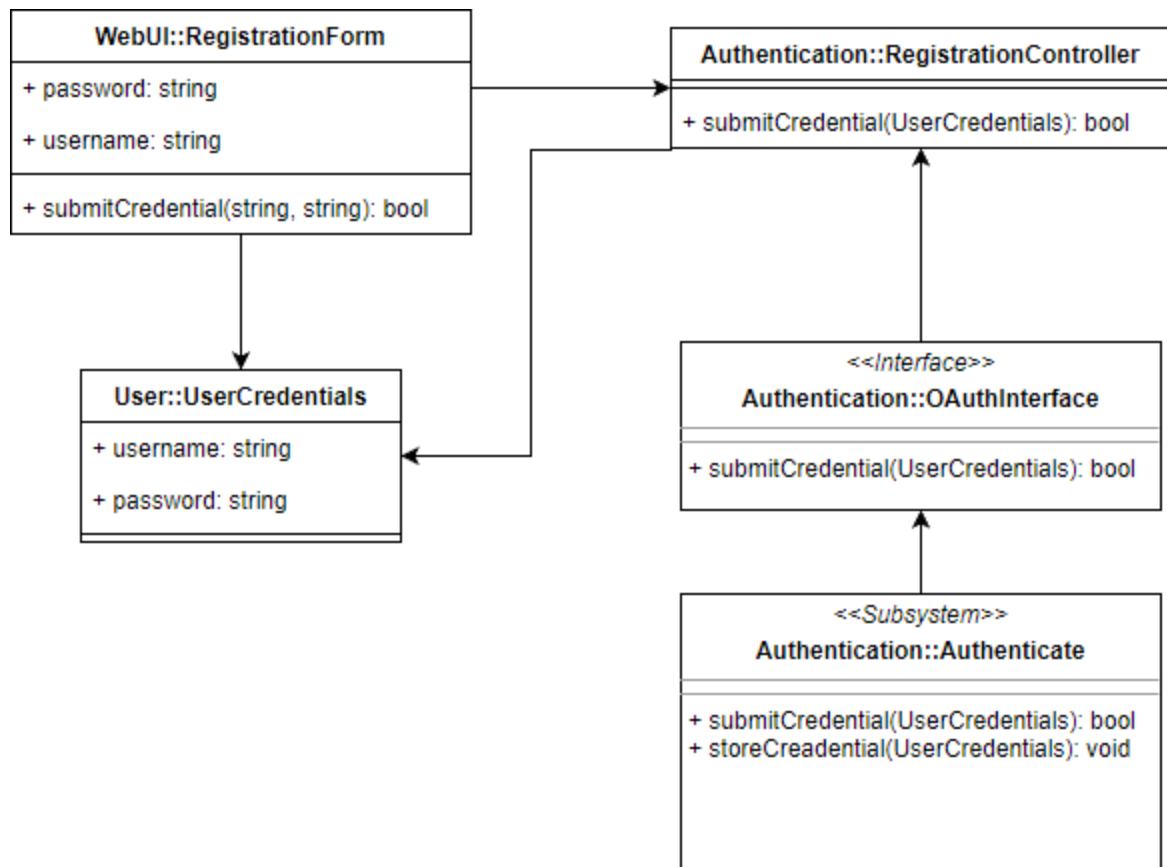
Figure 22: Vopc of create account use case
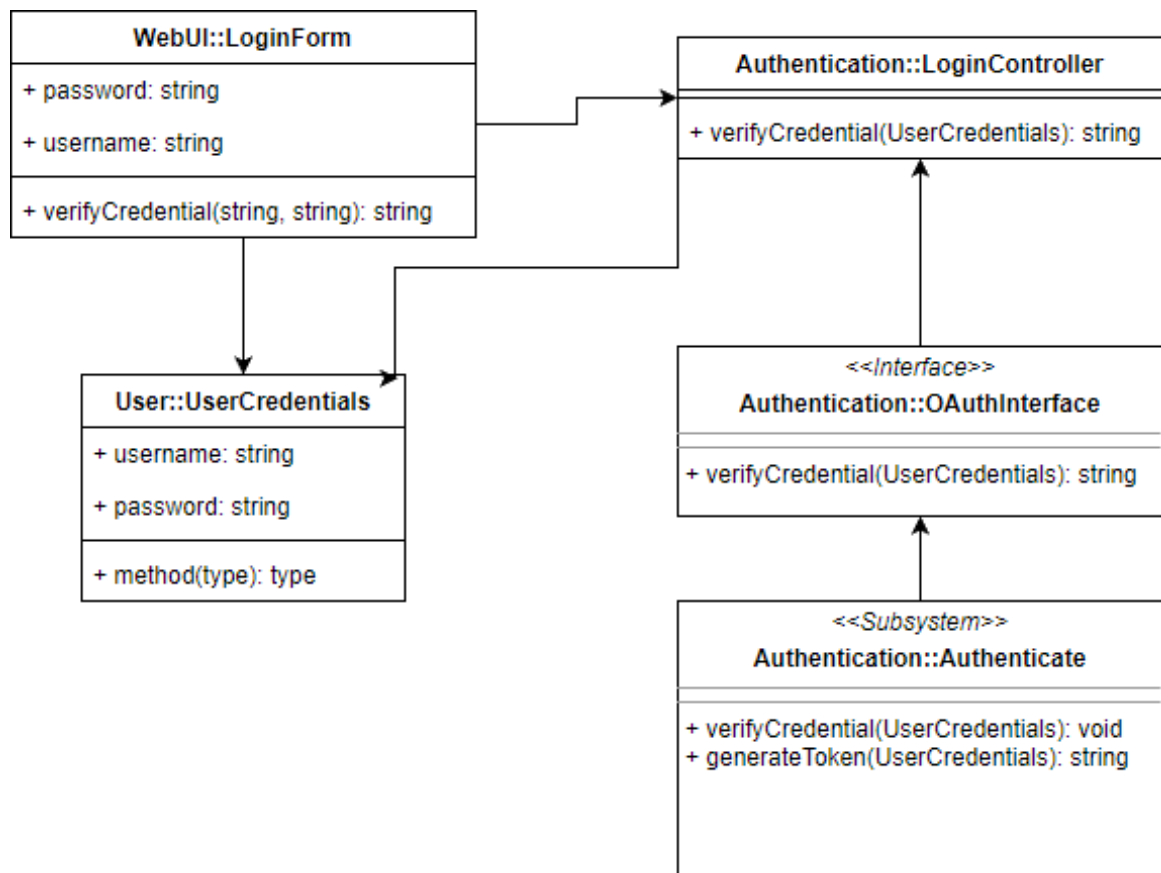
Figure 23: Vopc of login use case
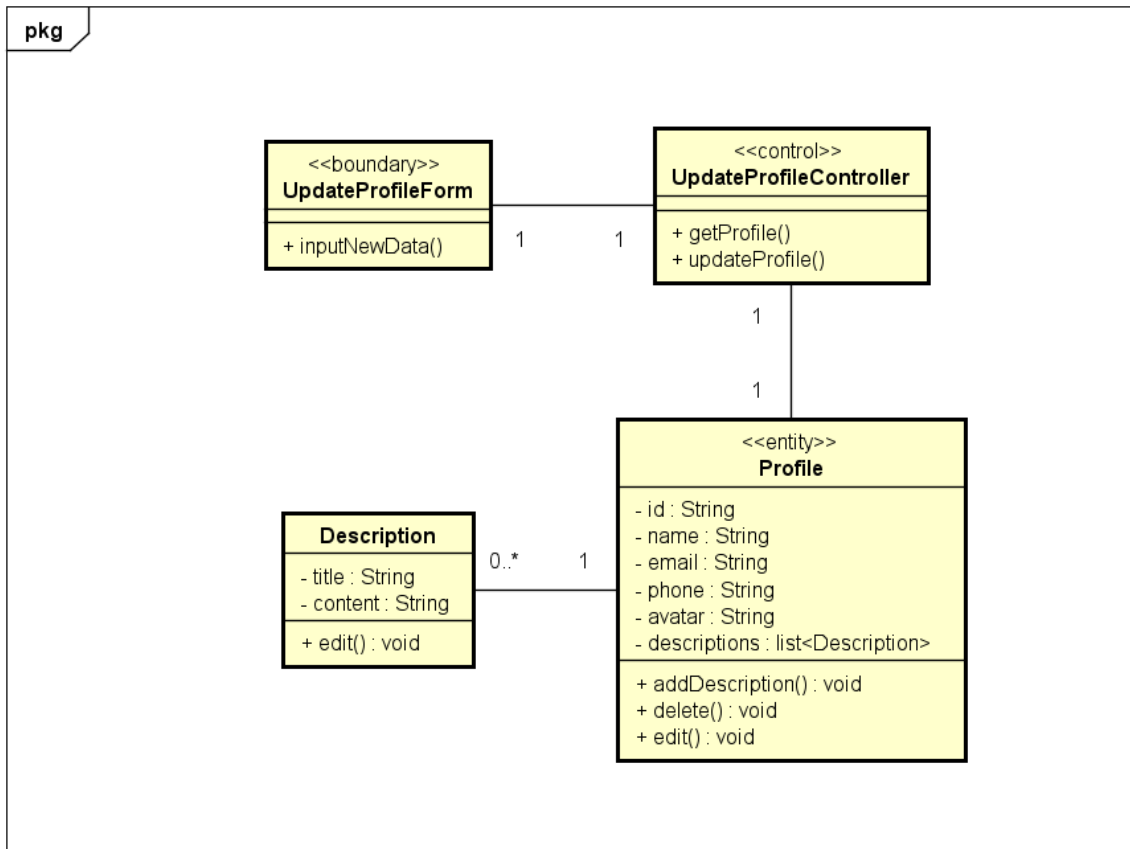
Figure 24: Vopc of update profile use case
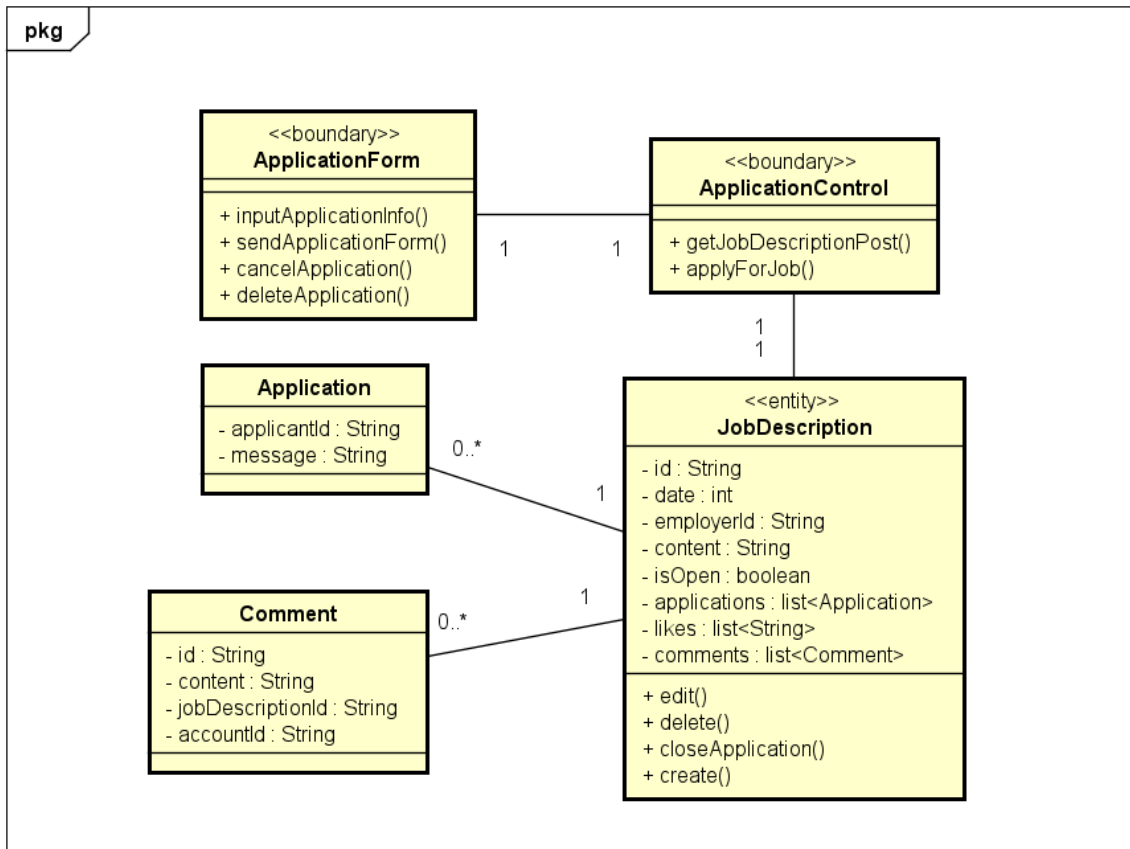
Figure 25: Vopc of job application use case
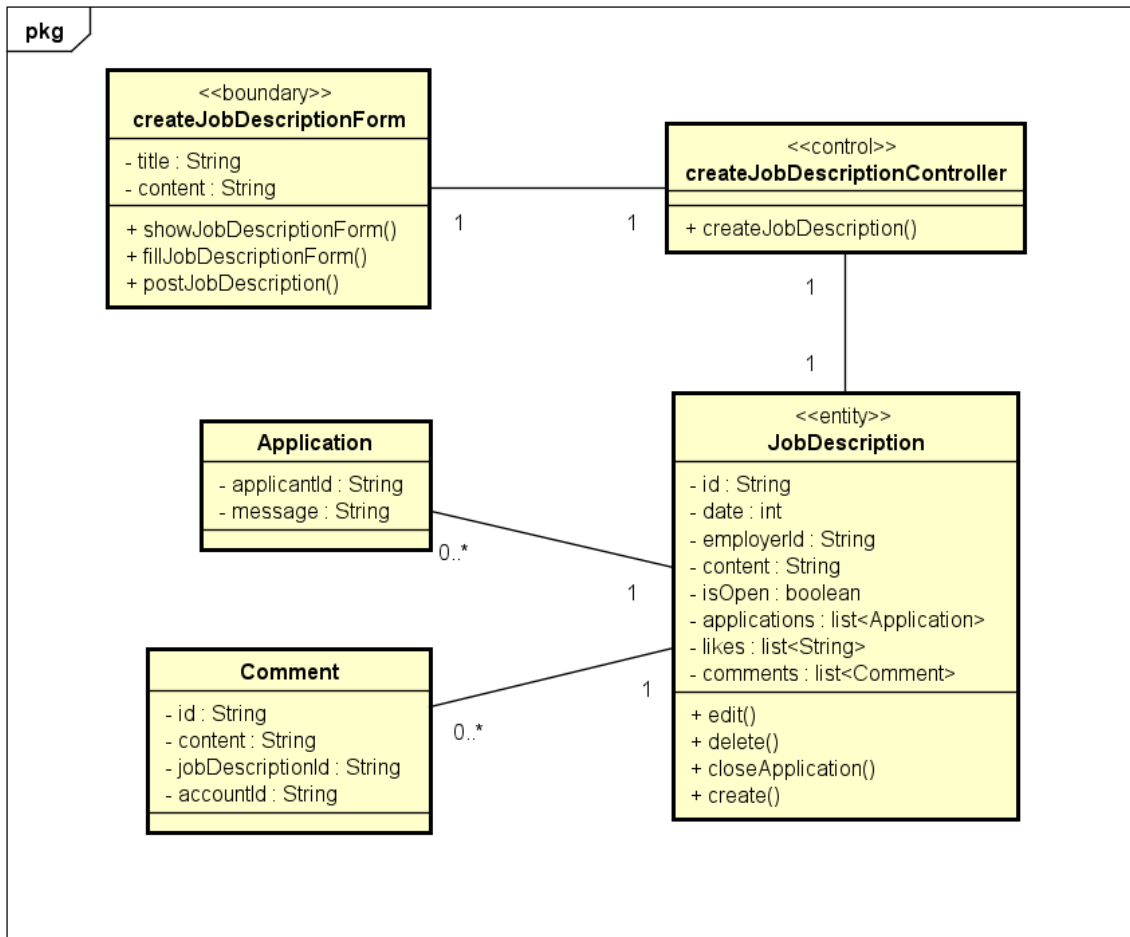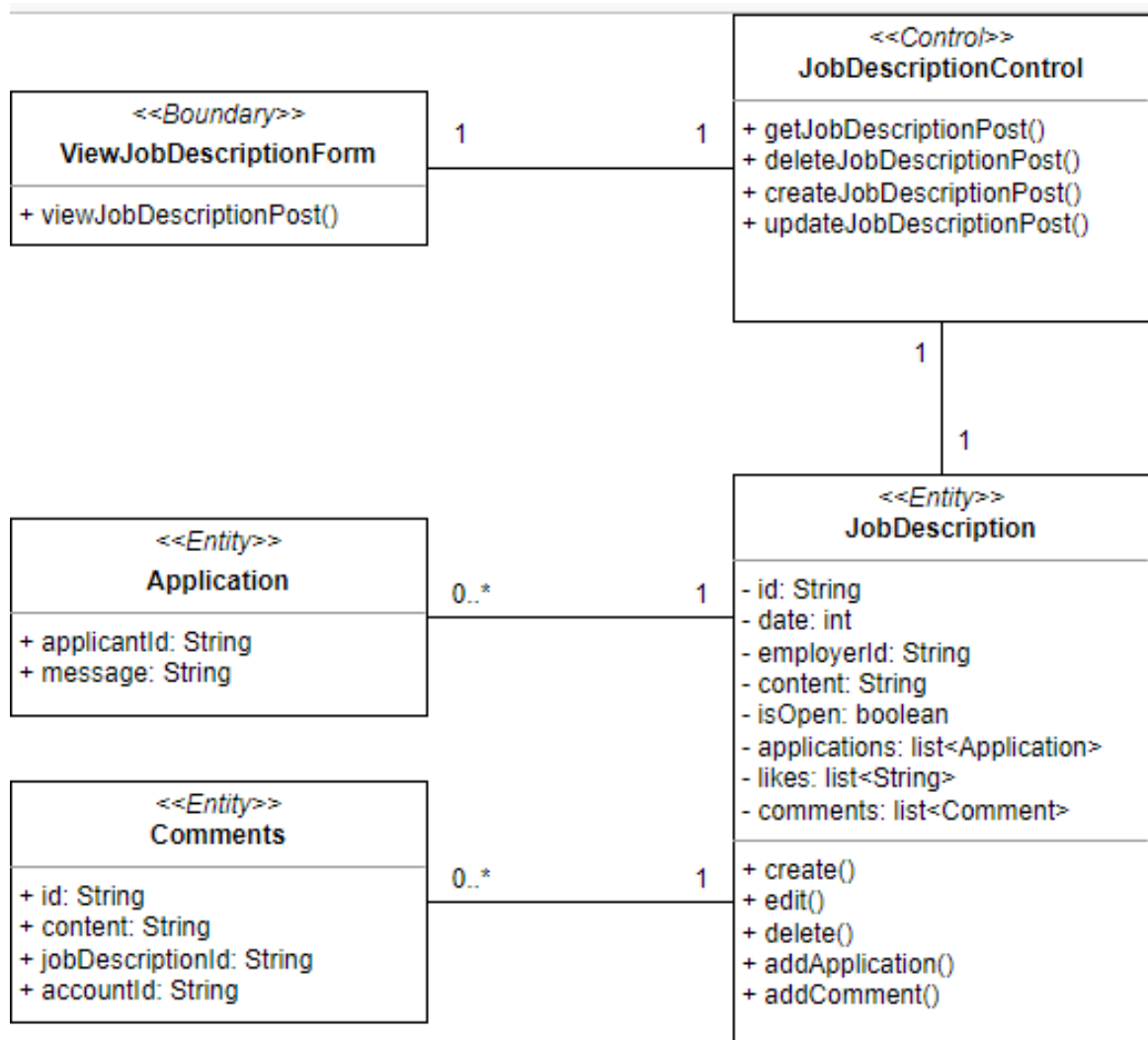
Figure 26: Vopc of post job description use case

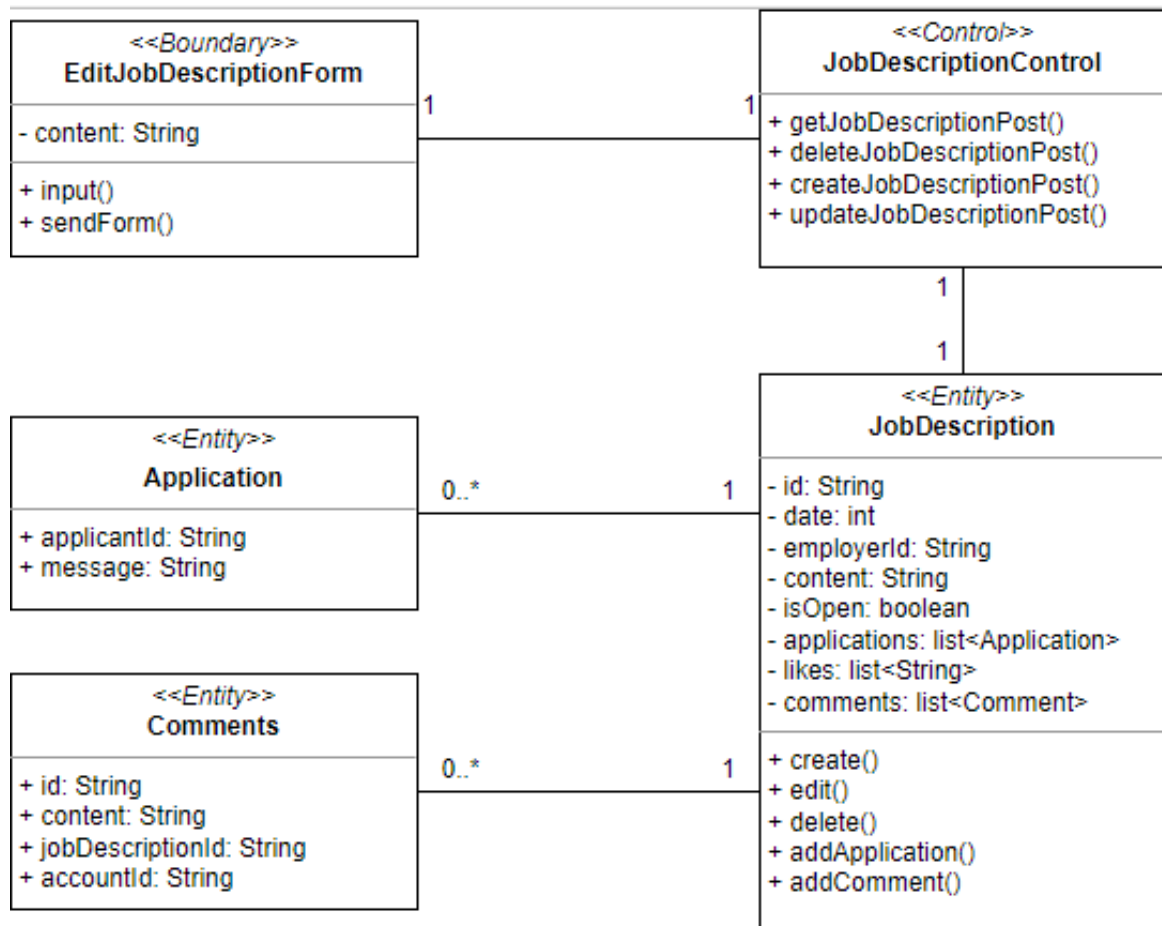**Figure 27:** Vopc of view job description use case

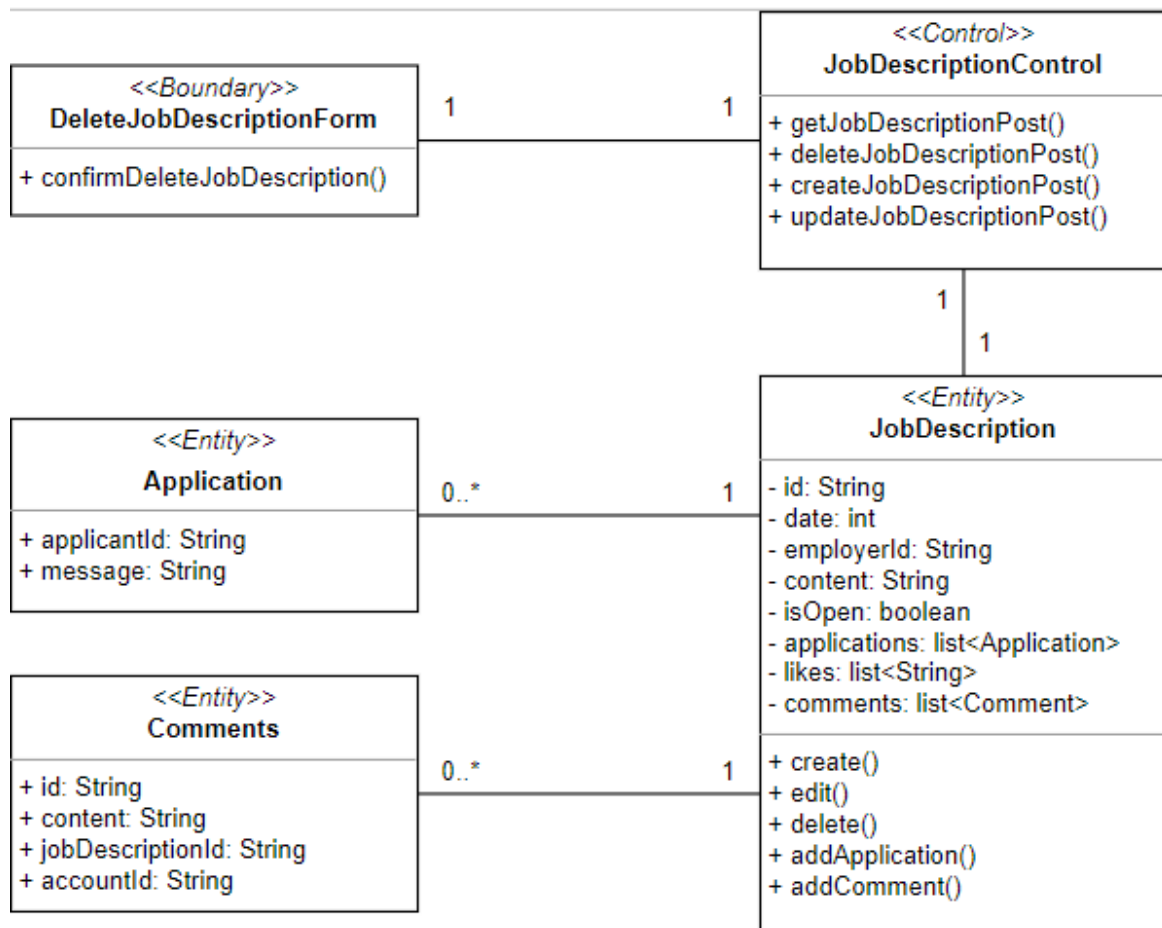**Figure 28:** Vopc of edit job description use case
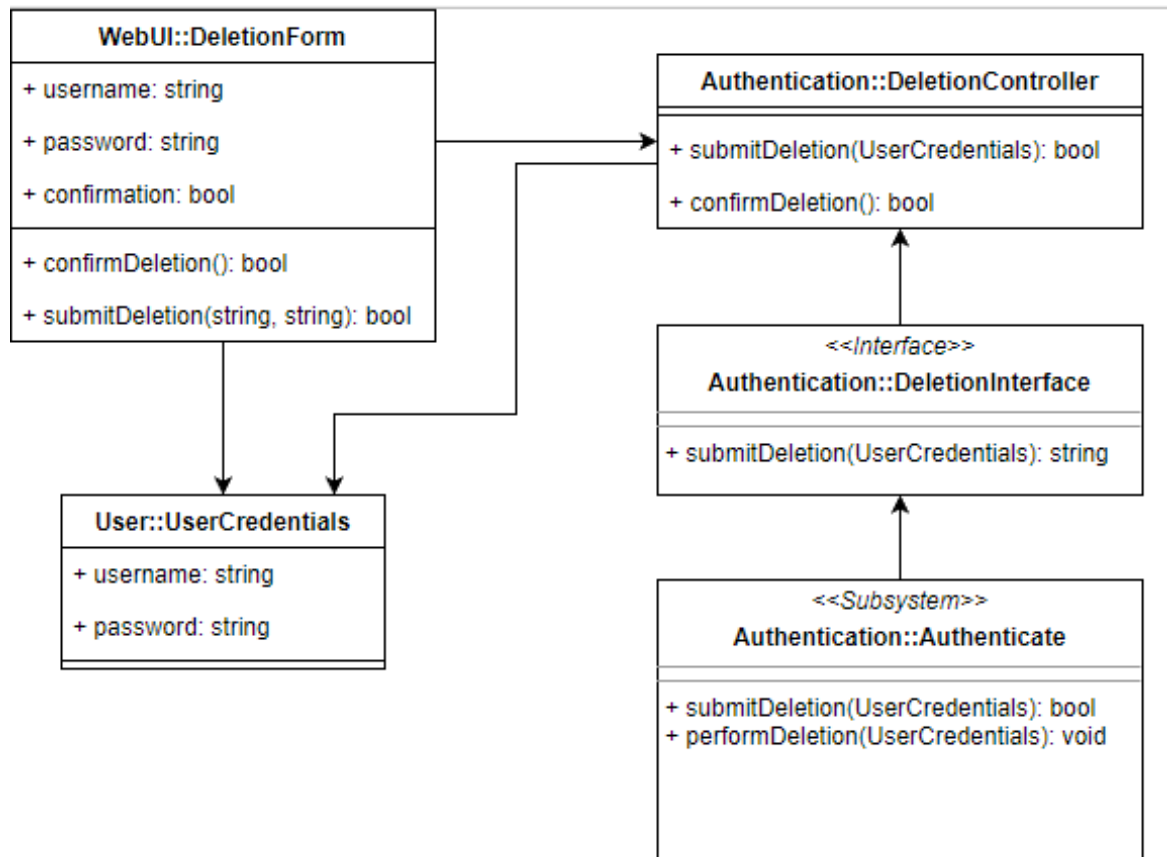
Figure 29: Vopc of delete job description use case
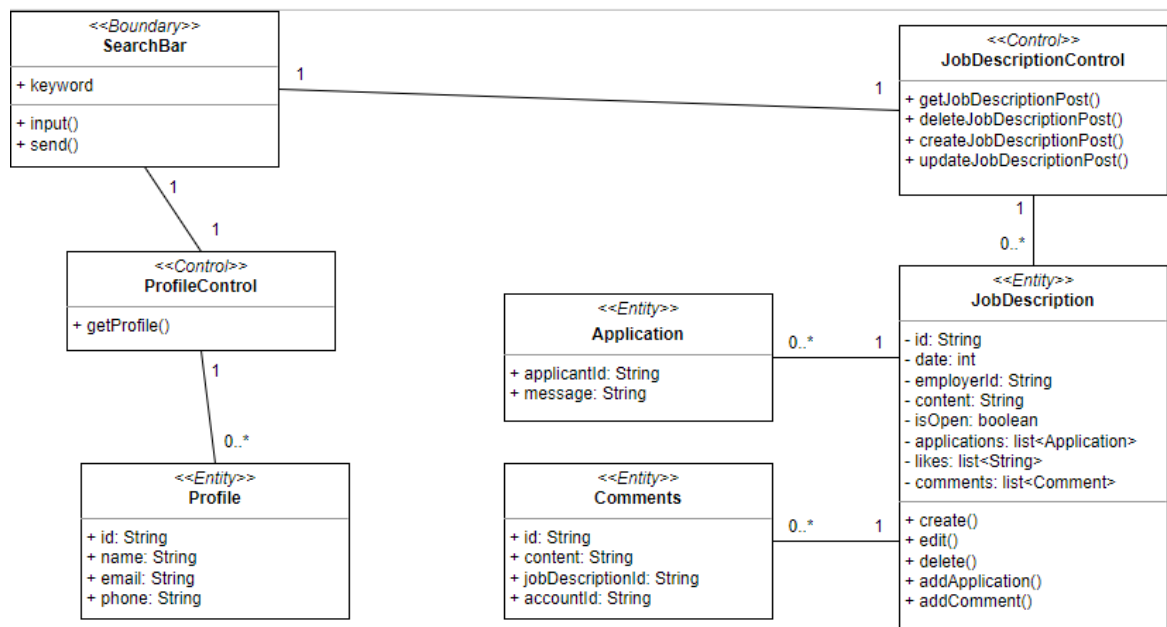
Figure 30: Vopc of delete account use case

Figure 31: Vopc of search engine use case

3 Use case design

3.1 Identify classes

Analysis class	Design element
Account	Account, Database subsystem
Job description	Job description, Database subsystem
UserProfileController	Map directly to design classes
RegistrationController	
LoginController	
UpdateProfileController	
JobApplicationController	
JobDescriptionController	
DeletionController	
RegistrationForm	
LoginForm	
UpdateProfileForm	
ApplicationForm	
CreateJobDescriptionForm	
EditJobDescriptionForm	
DeleteJobDescriptionForm	
DeleteAccountForm	
SearchBarForm	

3.2 Identify subsystem and interfaces
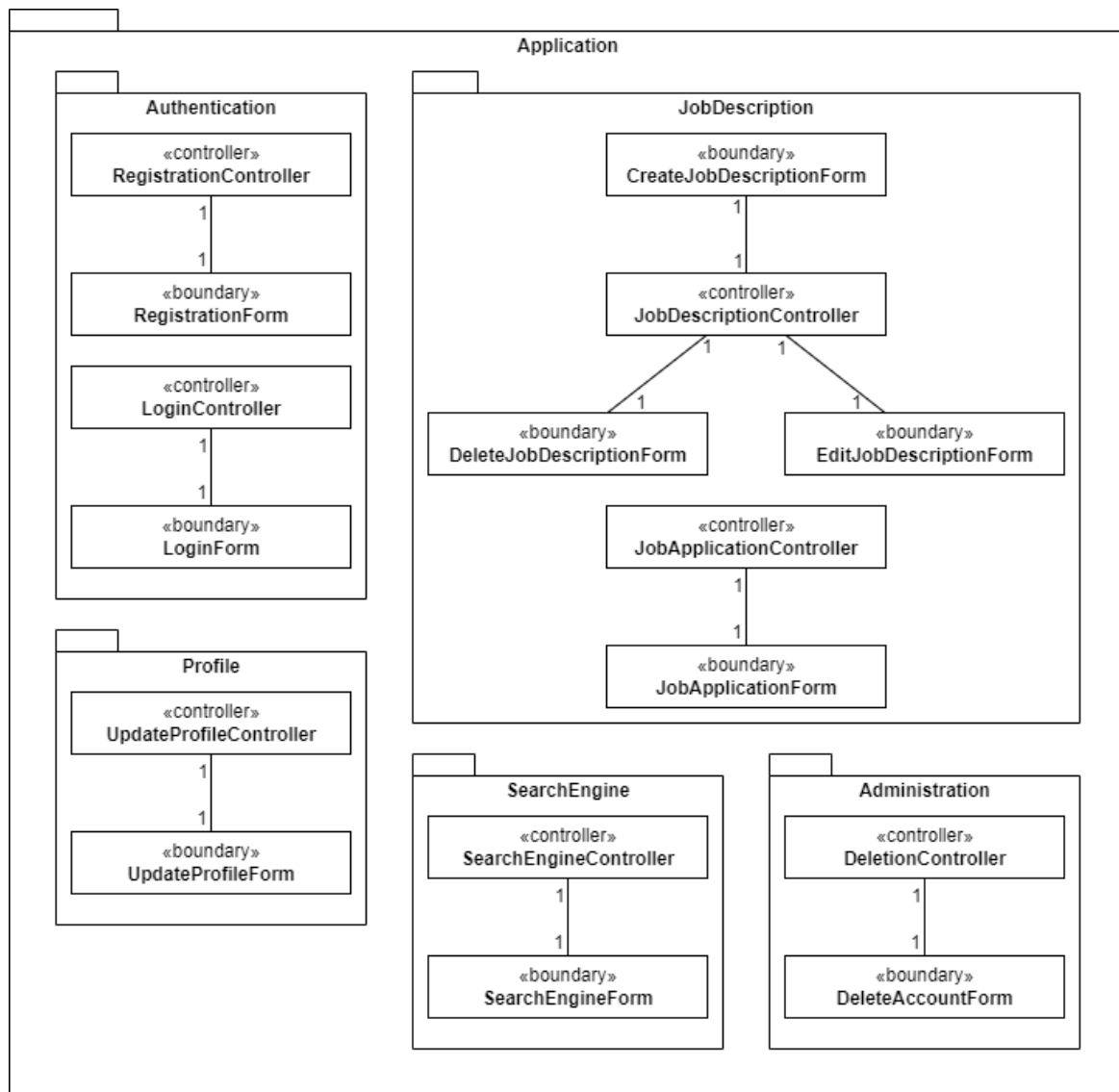
Giang's job

3.3 Identify packages

The Application package

Figure 32: Design of the application package