

**Capstone Project Report**

**Report 3 – Software Requirement Specification**

– Hanoi, January 2021 –

**Table of Contents**

[I. Project Report 3](#_Toc63755073)

[1. Status Report 3](#_Toc63755074)

[2. Team Involvements 3](#_Toc63755075)

[3. Issues/Suggestions 3](#_Toc63755076)

[II. Software Requirement Specification 4](#_Toc63755077)

[1. Overall Description 4](#_Toc63755078)

[1.1 Product Overview 4](#_Toc63755079)

[1.2 Business Rules 4](#_Toc63755080)

[2. User Requirements 5](#_Toc63755081)

[2.1 Overview 5](#_Toc63755082)

[2.2 Feature for guest 10](#_Toc63755083)

[2.3 Feature for user 13](#_Toc63755084)

[2.4 Feature for manager 39](#_Toc63755085)

[3.1 System Functional Overview 54](#_Toc63755086)

[3.2 Mockup Screen 59](#_Toc63755087)

[4. Non-Functional Requirements 77](#_Toc63755088)

[4.1 External Interfaces 77](#_Toc63755089)

[4.2 Quality Attributes 78](#_Toc63755090)

[5. Other Requirements 80](#_Toc63755091)

[5.1 Appendix1 - Messages List 80](#_Toc63755092)

# I. Project Report

## 1. Status Report

|  |  |  |  |
| --- | --- | --- | --- |
| **#** | **Work Item** | **Status** | **Notes (Work Item in Details)** |
| 1 |  | Pending |  |
| 2 |  | In Progress |  |
| 3 |  | Completed |  |

## 2. Team Involvements

|  |  |  |  |
| --- | --- | --- | --- |
| **#** | **Task** | **Member** | **Notes (Task Details, etc.)** |
| 1 |  | HoangNH |  |
| 2 |  | DungHV |  |
| 3 |  | VanLT |  |
| 4 |  | TungNT |  |
| 5 |  | VienMV |  |

## 3. Issues/Suggestions

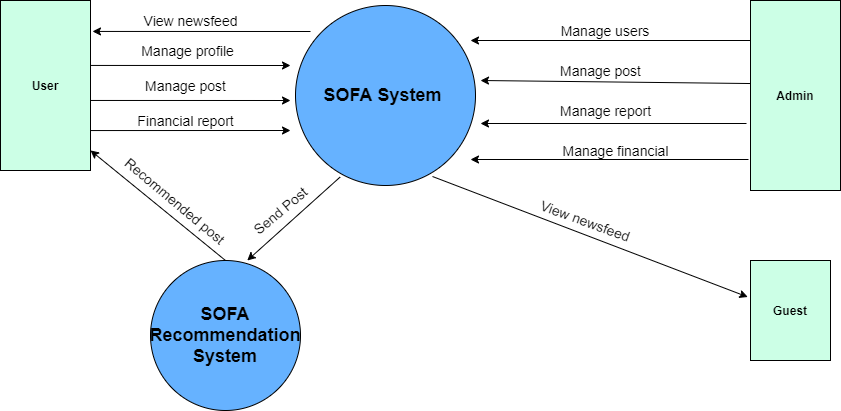
|  |  |  |  |
| --- | --- | --- | --- |
| **#** | **Issue** | **Status** | **Notes (Solution, Suggestion, etc.)** |
| 1 |  | Pending |  |
| 2 |  | In Progress |  |
| 3 |  | Completed |  |

# II. Software Requirement Specification

## 1. Overall Description

### 1.1 Product Overview

SOFA is a new social network that people can share all about fashion information. For example, they can share their beautiful moments, the place that they bought their outfits, or how to mix outfits of the day. Besides, the app can suggest posts of people who have the same size and shape of the body as the users' bodies. This makes SOFA different from the other apps. The context diagram below illustrates the external entities and system interfaces for release 1.0. The app is expected to evolve over several releases and become popular and useful for users.



### 1.2 Business Rules

|  |  |
| --- | --- |
| **ID** | **Rule Definition** |
| BR-01 | Only system accounts are allowed to access the management system. |
| BR-02 | Only admin accounts can create a staff account for the system. |
| BR-03 | Administrators are not allowed to delete user accounts from the system, only allowed to ban |
| BR-04 | Administrators or Staffs are not allowed to know users’ password |
| BR-05 | Administrators or Staffs are not allowed to change users’ balance |
| BR-06 | All users are allowed to know the time when the system will be maintained. |
| BR-07 | Users are allowed to know the reasons when her/his account is banned |
| BR-08 | The length of the content Maximum 2000 character |
| BR-09 | Before posting, the images in the post will be moderated for infringing content |
| BR-10 | Only posters can modify their post and all information in the post |
| BR-11 | The users can like, mark up all posts in their newsfeed |
| BR-12 | The users can rate star (1-5) in the other users’ posts |
| BR-13 | The users can send messages and follow the user |
| BR-14 | Post must include image about fashion |
| BR-15 | The users can advertise their posts |
| BR-16 | Users can send report request to admin |
| BR-17 | All input fields that include (\*) must not be blank |
| BR-18 | Phone number and email must be valid |
| BR-19 | User, phone number, Email must be unique |
| BR-20 | User must be login |
| BR-21 | User not login on app |

## 2. User Requirements

### 2.1 Overview

#### a. Use Case Diagram



*Figure 1. Common use cases*



*Figure 2. User use cases*



*Figure 3. Manager use cases*

#### b. System Actors

|  |  |  |
| --- | --- | --- |
| **#** | **Actor** | **Description** |
| 1 | Guest | The person who don’t have account and not login |
| 2 | User | The person have account and login |
| 3 | Manager | The person who manager the hole system, users, posts, reports, balance and recommendation data. |

#### c. Use Cases List

|  |  |  |  |
| --- | --- | --- | --- |
| **ID** | **Use Case** | **Actors** | **Use Case Description** |
| 01 | View newsfeed | Guest | Allow guests to view newsfeed but can’t react the post |
| 02 | Register | Guest | Allows guests to create an account for the system |
| 03 | Login | Guest | Allow user to login to the system |
| 04 | Forgot password | User | Allows users to reset a new password after authenticating owning an account |
| 05 | Verification | User | Send OTP code for the user |
| 06 | Logout | User | Allows users log out from the application |
| 07 | View newsfeed | User | Allows app users to access newsfeeds and view posts |
| 08 | View post detail | User | View the detail content of the post |
| 09 | Like post | User | Allow user to like the post |
| 10 | Rate post | User | Allow user to rate the post from 1 to 5 star |
| 11 | Comment post | User | Allow user to comment in the post |
| 12 | Report post | User | Allow user to report the post with some reason which have been classified |
| 13 | Markup | User | Allow user to save the post |
| 14 | Post a post | User | Allow user to create a post and post it |
| 15 | Update post | User | Allow user to update their post |
| 16 | Delete post | User | Allow user to delete their post |
| 17 | Follow users | User | Allow user to follow other users to see their post at the first of the newsfeed |
| 18 | Unfollow users | User | Allow user to unfollow other users |
| 19 | Block users | User | Allow user to block other users and the person who is blocked can’t find or communicate with the blocker |
| 20 | Report users | User | Allow user to report the user with some reason which have been classified |
| 21 | View other user’s profile | User | Allow user to view each other profile |
| 22 | Chat | User | Allow user to communicate with each other |
| 23 | Delete message in the conversation | User | Allow user to delete the message in the conversation with other user in their side or both side |
| 24 | Delete conversation | User | Allow user to delete one or many conversations |
| 25 | View list conversation | User | Allow user to view list of their conversation |
| 26 | Search a conversation | User | Allow user search to conversation by the name of the conversation |
| 27 | View my profile | User | Allow user to view their profile |
| 28 | Update my profile | User | Allow user to update their information then save it to their profile |
| 29 | Search people | User | Allow user to search others by their name |
| 30 | Search post | User | Allow user to search posts by their title, content, tag, caption |
| 31 | View list notification | User | View top newest notification which user have received |
| 32 | View a notification | User | View a notification: when user click to the notification system will direct to another screen |
| 33 | View hot posts | User | View fashionista posts and high engagement articles |
| 34 | Recommend | User | Recommend posts based on a user's metrics |
| 35 | Show suggest place of sale | User | Show people who sell products |
| 36 | View balance | User | Allows users to view account balances, transaction history |
| 37 | Top up account | User | Allows users to top up the account |
| 38 | View list voucher | User | View list voucher which user have received |
| 39 | View voucher detail | User | Allow user to view detail content of a voucher |
| 40 | Login | Manager | Allow manager to login to the web server |
| 41 | Logout | Manager | Allow manager to logout from the web server |
| 42 | Create new staff | Manager | Admins with higher authority will be allowed to create accounts with smaller roles for system administration |
| 43 | View list user | Manager | Allow manager to view all user |
| 44 | Search user | Manager | Allow manager to search user by their name |
| 45 | Ban user | Manager | Allow manager to ban user |
| 46 | Reset password for user | Manager | Reset password for user when they can connect recovery password |
| 47 | View list post | Manager | Allow manager to view all user’s post |
| 48 | View post detail | Manager | Allow manager to view detail content of a post |
| 49 | Search user’s post | Manager | Allow manager to search posts by their title, content, tag, caption |
| 50 | Delete post | Manager | Allow manager to delete a post |
| 51 | Manage voucher | Manager | Allow manager to offer a voucher to users |
| 52 | View list report | Manager | Allow manager to view list report |
| 53 | View report detail | Manager | Allow manager to view report detail |
| 54 | Manage recommend data | Manager | See a list of articles that score enough points to be included in the recommend system |
| 55 | Manage user’s balance | Manager | Allow manager to top up or down user’s balance |
| 56 | Manage my profile | Manager | Allow manager to update their information then save it to their profile |

### 2.2 Feature for guest

1. ***Use case 01 (Actor guest)***

|  |  |  |  |
| --- | --- | --- | --- |
| ID and Name: | **UC-01 View newsfeeds** | | |
|  |  | | |
| Created By: | Mai Văn Viên | Date Created: | 26/01/2021 |
| Primary Actor: | Guest | Secondary Actors: | None |
| Description: | An guest accesses the app from mobile phone view newsfeeds | | |
| Trigger: | An guest accesses the app from mobile phone and skip login | | |
| Preconditions: | None | | |
| Post-conditions: | View screen newsfeeds | | |
| Normal Flow: | 1. **Guest view newsfeeds**   1. The guest accesses the app from a mobile phone  2. The guest clicks “skip login”  3. Display screen newsfeeds | | |
| Alternative Flows: | None | | |
| Exceptions: | None | | |
| Priority: | Medium | | |
| Frequency of Use: | Average a guest view newsfeeds one time before register account in-app SOFA | | |
| Business Rules: | BR-21 | | |
| Other Information: | View post public sort by date decrease | | |
| Assumptions: | None | | |

1. ***Use case 02 (Actor guest)***

|  |  |  |  |
| --- | --- | --- | --- |
| ID and Name: | **UC-02 Register** | | |
| Created By: | Mai Văn Viên | Date Created: | 26/01/2021 |
| Primary Actor: | Guest | Secondary Actors: | None |
| Description: | When a guest wants to use the functions of the system, it is imperative that they register for an account | | |
| Trigger: | Guest clicks on the button “Register” on Home screen | | |
| Preconditions: | None | | |
| Post-conditions: | Guest submit to send information to the system | | |
| Normal Flow: | **1.0 Guest Register**   1. The guest accesses the app from a mobile phone 2. The guest clicks “register” 3. System displays screen register 4. The guest inputs email or mobile phone number and password to register an account on the app | | |
| Alternative Flows: | **1.0 Guest Register in screen “login”**   1. In screen “login” 2. The guest clicks “register” | | |
| Exceptions: | 1. User already exists 2. Email already exists 3. Phone number already exists 4. Password invalid 5. Phone number invalid 6. Email invalid | | |
| Priority: | High | | |
| Frequency of Use: | None | | |
| Business Rules: | BR-17, BR-18, BR-19 | | |
| Other Information: | 1. Use when the user does not have an account on the system 2. When successfully registered, the default role is the user | | |
| Assumptions: | None | | |

1. ***Use case 03 (Actor guest)***

|  |  |  |  |
| --- | --- | --- | --- |
| ID and Name: | **UC-03 Login** | | |
| Created By: | Mai Văn Viên | Date Created: | 26/01/2021 |
| Primary Actor: | Guest | Secondary Actors: | None |
| Description: | If the guest has an account register in-app, they can access the app from their mobile phone | | |
| Trigger: | Guest clicks on the button “Login” on Home screen | | |
| Preconditions: | Account must be register in app SOFA | | |
| Post-conditions: | Login to mobile app SOFA | | |
| Normal Flow: | **1.0 Guest login**   1. The guest accesses the app from a mobile phone 2. Guest clicks “Login” 3. System displays screen login 4. The guest inputs email or mobile phone number and password registered on the app | | |
| Alternative Flows: | **1.0 Guest login when comment post in new feed**   1. In the screen “new feed”, guest comment in post 2. Display screen “login” | | |
| Exceptions: | 1. User doesn't exist 2. Email doesn't exist 3. Phone number doesn't exist 4. User or password don’t correct | | |
| Priority: | High | | |
| Frequency of Use: | This use case happens very regularly | | |
| Business Rules: | BR17 | | |
| Other Information: | When successful login, the application will go to the newsfeed screen | | |
| Assumptions: | None | | |

### 2.3 Feature for user

**a) *Use case 04 (Actor User)***

|  |  |  |  |
| --- | --- | --- | --- |
| ID and Name: | **UC-04 Forgot password** | | |
| Created By: | Mai Văn Viên | Date Created: | 26/01/2021 |
| Primary Actor: | User | Secondary Actors: | None |
| Description: | When the user forgets the password to log in, the system will send OTP to their email or mobile phone number to reset the password in the system | | |
| Trigger: | Guest clicks on the button “Forgot password” on Home screen | | |
| Preconditions: | Email or mobile phone number must be registered in-app SOFA | | |
| Post-conditions: | Send new password to email or mobile phone number of the user | | |
| Normal Flow: | **1.0 User forgot password**   1. On Home screen, User clicks “login” 2. System displays screen login 3. User click “Forgot password” 4. System display screen forgot password 5. User input email or password to input field and submit 6. System sends a new password to email or phone number 7. Display screen “Verification” | | |
| Alternative Flows: | None | | |
| Exceptions: | 1. User doesn't exist 2. Email doesn't exist 3. Phone number doesn't exist | | |
| Priority: | High | | |
| Frequency of Use: | This use case happens sometimes when the user forgot the password | | |
| Business Rules: | BR-17, BR-18 | | |
| Other Information: | None | | |
| Assumptions: | None | | |

**b) *Use case 05 (Actor User)***

|  |  |  |  |
| --- | --- | --- | --- |
| ID and Name: | **UC-05 Verification** | | |
| Created By: | Mai Văn Viên | Date Created: | 26/01/2021 |
| Primary Actor: | User | Secondary Actors: | None |
| Description: | When the user registers on the app, the system will send one OTP to their email or phone number. User must input valid OTP to continue use functions of the app | | |
| Trigger: | None | | |
| Preconditions: | None | | |
| Post-conditions: | None | | |
| Normal Flow: | **1.0 User Verification**   1. After the user submit info in screen “Register” 2. System displays screen “Verification” 3. User input OTP and submit | | |
| Alternative Flows: | None | | |
| Exceptions: | 1. OTP incorrect 2. System don’t send OTP to email or phone number 3. Their email or phone number incorrect | | |
| Priority: | High | | |
| Frequency of Use: | None | | |
| Business Rules: | BR-17, BR-18 | | |
| Other Information: | None | | |
| Assumptions: | None | | |

**c) *Use case 06 (Actor User)***

|  |  |  |  |
| --- | --- | --- | --- |
| ID and Name: | **UC-03 Logout** | | |
| Created By: | Mai Văn Viên | Date Created: | 26/01/2021 |
| Primary Actor: | User | Secondary Actors: | None |
| Description: | User logs out of the app | | |
| Trigger: | User clicks on the button “Logout” | | |
| Preconditions: | User login the app | | |
| Post-conditions: | None | | |
| Normal Flow: | **1.0 User Logout**   1. In the menu, User clicks “Logout” | | |
| Alternative Flows: | **1.0 User Logout when change password**   1. In the screen “Profile”, User clicks “Change password” 2. User input all fields and submit 3. Display toast message “Change password success!” 4. User logout from session | | |
| Exceptions: | None | | |
| Priority: | Low | | |
| Frequency of Use: | None | | |
| Business Rules: | BR-17, BR-18, BR-20 | | |
| Other Information: | None | | |
| Assumptions: | None | | |

**d) *Use case 07 (Actor User)***

|  |  |  |  |
| --- | --- | --- | --- |
| ID and Name: | **UC-03 View newsfeed** | | |
| Created By: | Mai Văn Viên | Date Created: | 26/01/2021 |
| Primary Actor: | User | Secondary Actors: | None |
| Description: | User view list of post that other user-submitted to the system | | |
| Trigger: | None | | |
| Preconditions: | The user has successfully logged in to the system | | |
| Post-conditions: | None | | |
| Normal Flow: | **1.0 User views newsfeeds**   1. In the navigation bar, click tab “Newsfeed” | | |
| Alternative Flows: | **1.0 User views newsfeeds after login**   1. The guest accesses the app from a mobile phone 2. Guest clicks “Login” 3. System displays screen login 4. The guest inputs email or mobile phone number and password registered on the app and submit 5. Display screen newsfeeds | | |
| Exceptions: | None | | |
| Priority: | High | | |
| Frequency of Use: | This use case regularly happens and usage load for this use case is all day | | |
| Business Rules: | None | | |
| Other Information: | None | | |
| Assumptions: | None | | |

1. ***Use case 08 (Actor User)***

|  |  |  |  |
| --- | --- | --- | --- |
| ID and Name: | **UC-08 View post detail** | | |
| Created By: | Nguyễn Huy Hoàng | Date Created: | 26/01/2021 |
| Primary Actor: | User | Secondary Actors: | None |
| Description: | View the detail content of the post | | |
| Trigger: | An user accesses the app from mobile phone then click on the post on the newsfeed | | |
| Preconditions: | PRE-1. User have login into SOFA | | |
| Post-conditions: | POST-1. The post is now exist in the system | | |
| Normal Flow: | **1.0 View post detail**   1. In the navigation bar, click tab “Newsfeed” 2. Click on the post that you want to view detail | | |
| Alternative Flows: | **1.0 View post detail when user click on post notification**   1. In the navigation bar, click tab “Notification” 2. Click on the notification about the post that you want to view detail | | |
| Exceptions: | None | | |
| Priority: | High | | |
| Frequency of Use: | This use case happen very regularly and usage load for this use case is all day | | |
| Business Rules: | None | | |
| Other Information: | None | | |
| Assumptions: | None | | |

1. ***Use case 09 (Actor User)***

|  |  |  |  |
| --- | --- | --- | --- |
| ID and Name: | **UC-09 Like post** | | |
| Created By: | Nguyễn Huy Hoàng | Date Created: | 26/01/2021 |
| Primary Actor: | User | Secondary Actors: | None |
| Description: | User like the post which they see | | |
| Trigger: | User click on the Like button on the post | | |
| Preconditions: | PRE-1. User have login into SOFA | | |
| Post-conditions: | POST-1. The post is now exist in the system | | |
| Normal Flow: | * 1. **User views a post**  1. User click on the Like button on the post | | |
| Alternative Flows: | None | | |
| Exceptions: | None | | |
| Priority: | High | | |
| Frequency of Use: | This use case happen very regularly and usage load for this use case is all day | | |
| Business Rules: | None | | |
| Other Information: | None | | |
| Assumptions: | None | | |

1. ***Use case 10 (Actor User)***

|  |  |  |  |
| --- | --- | --- | --- |
| ID and Name: | **UC-10 Rate post** | | |
| Created By: | Nguyễn Huy Hoàng | Date Created: | 26/01/2021 |
| Primary Actor: | User | Secondary Actors: | None |
| Description: | Allow user to rate the post from 1 to 5 star | | |
| Trigger: | User click on the Rate bar on the post | | |
| Preconditions: | PRE-1. User have login into SOFA | | |
| Post-conditions: | POST-1. The post is now exist in the system | | |
| Normal Flow: | * 1. **User views a post**  1. User click on the Rate bar from 1 to 5 on the post | | |
| Alternative Flows: | None | | |
| Exceptions: | None | | |
| Priority: | High | | |
| Frequency of Use: | This use case happen regularly and usage load for this use case is all day | | |
| Business Rules: | None | | |
| Other Information: | None | | |
| Assumptions: | None | | |

1. ***Use case 11 (Actor User)***

|  |  |  |  |
| --- | --- | --- | --- |
| ID and Name: | **UC-11 Comment post** | | |
| Created By: | Nguyễn Huy Hoàng | Date Created: | 26/01/2021 |
| Primary Actor: | User | Secondary Actors: | None |
| Description: | Allow user to leave a comment on the post | | |
| Trigger: | User input comment on the text field then enter | | |
| Preconditions: | PRE-1. User have login into SOFA | | |
| Post-conditions: | POST-1. The post is now exist in the system | | |
| Normal Flow: | * 1. **User views a post**  1. User input comment on the text field then enter | | |
| Alternative Flows: | None | | |
| Exceptions: | None | | |
| Priority: | High | | |
| Frequency of Use: | This use case happen regularly and usage load for this use case is all day | | |
| Business Rules: | BR-08 | | |
| Other Information: | None | | |
| Assumptions: | None | | |

1. ***Use case 12 (Actor User)***

|  |  |  |  |
| --- | --- | --- | --- |
| ID and Name: | **UC-12 Report post** | | |
| Created By: | Nguyễn Huy Hoàng | Date Created: | 26/01/2021 |
| Primary Actor: | User | Secondary Actors: | None |
| Description: | Allow user to report a post | | |
| Trigger: | When user choose a report button on the post | | |
| Preconditions: | PRE-1. User have login into SOFA | | |
| Post-conditions: | POST-1. The post is now exist in the system | | |
| Normal Flow: | * 1. **User views a post**  1. User click on the Report button on the post | | |
| Alternative Flows: | None | | |
| Exceptions: | None | | |
| Priority: | High | | |
| Frequency of Use: | This use case rarely happen and usage load for this use case is all day | | |
| Business Rules: | None | | |
| Other Information: | None | | |
| Assumptions: | None | | |

1. ***Use case 13 (Actor User)***

|  |  |  |  |
| --- | --- | --- | --- |
| ID and Name: | **UC-13 Markup** | | |
| Created By: | Nguyễn Huy Hoàng | Date Created: | 26/01/2021 |
| Primary Actor: | User | Secondary Actors: | None |
| Description: | Allow user to save the post and the post will be save in the user’s library | | |
| Trigger: | When user choose a save button on the post | | |
| Preconditions: | PRE-1. User have login into SOFA | | |
| Post-conditions: | POST-1. The post is now exist in the system | | |
| Normal Flow: | * 1. **User views a post**  1. User click on the Save button on the post | | |
| Alternative Flows: | None | | |
| Exceptions: | None | | |
| Priority: | High | | |
| Frequency of Use: | This use case sometimes happen and usage load for this use case is all day | | |
| Business Rules: | None | | |
| Other Information: | None | | |
| Assumptions: | None | | |

1. ***Use case 14 (Actor User)***

|  |  |  |  |
| --- | --- | --- | --- |
| ID and Name: | **UC-14 Post a post** | | |
| Created By: | Nguyễn Huy Hoàng | Date Created: | 26/01/2021 |
| Primary Actor: | User | Secondary Actors: | None |
| Description: | Allow user to post text and images the their profile | | |
| Trigger: | When user write a text and upload pictures then choose post | | |
| Preconditions: | PRE-1. User have login into SOFA | | |
| Post-conditions: | None | | |
| Normal Flow: | **1.0 User are in newsfeed tab**  1. User click on the Post button  2. User enter text then choose pictures to update  3. User choose post this post | | |
| Alternative Flows: | None | | |
| Exceptions: | None | | |
| Priority: | High | | |
| Frequency of Use: | This use case regularly happen and usage load for this use case is all day | | |
| Business Rules: | BR-08, BR-09 | | |
| Other Information: | None | | |
| Assumptions: | None | | |

1. ***Use case 15 (Actor User)***

|  |  |  |  |
| --- | --- | --- | --- |
| ID and Name: | **UC-15 Update a post** | | |
| Created By: | Nguyễn Huy Hoàng | Date Created: | 26/01/2021 |
| Primary Actor: | User | Secondary Actors: | None |
| Description: | Allow user to update their post | | |
| Trigger: | When user write a text and upload pictures then choose post | | |
| Preconditions: | PRE-1. User have login into SOFA | | |
| Post-conditions: | POST-1. The post is now existing in the system  POST-2. The post is owned by it user | | |
| Normal Flow: | **1.0 User are in their profile tab**  1. User choose the post they want to update  2. User click on the button update  3. User update this post then click Save | | |
| Alternative Flows: | None | | |
| Exceptions: | None | | |
| Priority: | High | | |
| Frequency of Use: | This use case sometimes happen and usage load for this use case is all day | | |
| Business Rules: | BR-08, BR-09 | | |
| Other Information: | None | | |
| Assumptions: | None | | |

1. ***Use case 16 (Actor User)***

|  |  |  |  |
| --- | --- | --- | --- |
| ID and Name: | **UC-16 Delete a post** | | |
| Created By: | Nguyễn Huy Hoàng | Date Created: | 26/01/2021 |
| Primary Actor: | User | Secondary Actors: | None |
| Description: | Allow user to delete their post | | |
| Trigger: | When user click on the button Delete on the post | | |
| Preconditions: | PRE-1. User have login into SOFA | | |
| Post-conditions: | POST-1. The post is now existing in the system  POST-2. The post is owned by it user | | |
| Normal Flow: | **1.0 User are in their profile tab**  1. User choose the post they want to delete  2. User click on the Delete button  3. User click Yes to confirm delete this post | | |
| Alternative Flows: | None | | |
| Exceptions: | None | | |
| Priority: | High | | |
| Frequency of Use: | This use case sometimes happen and usage load for this use case is all day | | |
| Business Rules: | None | | |
| Other Information: | None | | |
| Assumptions: | None | | |

1. ***Use case 17 (Actor User)***

|  |  |  |  |
| --- | --- | --- | --- |
| ID and Name: | **UC-17 Follow up Users** | | |
| Created By: | Hà Việt Dũng | Date Created: | 26/01/2021 |
| Primary Actor: | User | Secondary Actors: | None |
| Description: | An User accesses the app from mobile phone, chooses another user and decides to follow him/her | | |
| Trigger: | An User indicates that he/she wants to follow an other's account | | |
| Preconditions: | PRE-1. User logged into SOFA. | | |
| Post-conditions: | POST-1. The person who wants to follow someone is not being blocked by the other person.  POST-2. Account of user who uses this use case is not being banned  POST-3. The followed person’s account is not being banned  POST-4. Number of people who followed the person has not exceeded the limit | | |
| Normal Flow: | **1.0 Follow up an person without permission necessary.**   1. An User accesses another user’s profile 2. The user sends a follow-up request to the other. (see 1.0.E1) 3. Follow successfully | | |
| Alternative Flows: | **1.1 Follow up an person with permission necessary**  Step 1,2 of normal flow  The followed user see the follow-up request (see 1.1.E1) | | |
| Exceptions: | **1.0.E1 Number of people who followed the person has exceeded the limit**  1. The followed person has too many followers  2. The follow-up request is not sent to the followed person  **1.1.E1 The followed person decides not to accept the request**  1a. If the followed user accepts the request, the return to step 3 of normal flow  1b. Else if the followed user declines the request, the SOFA terminates use case. | | |
| Priority: | Medium | | |
| Frequency of Use: | Average a person follow-ups 3-5 people per day. Usage load for this use case is all day | | |
| Business Rules: |  | | |
| Other Information: | 1. Each person can see who is following him/her and the people whom he/she is following.  2. Number of followers can be shown in everyone’s profile if he/she wants. | | |
| Assumptions: | Assume that a person will follow 100 users and be followed by 100 users. | | |

***o) Use case 18 (Actor User)***

|  |  |  |  |
| --- | --- | --- | --- |
| ID and Name: | **UC-18 Unfollow up Users** | | |
| Created By: | Hà Việt Dũng | Date Created: | 26/01/2021 |
| Primary Actor: | User | Secondary Actors: | None |
| Description: | An User accesses the app from mobile phone, chooses an other user whom he/she is following and decides to unfollow. | | |
| Trigger: | An User indicates that he/she wants to unfollow an other's account | | |
| Preconditions: | PRE-1. User logged into SOFA. | | |
| Post-conditions: | POST-1. User is following at least 1 person.  POST-2. Account of user who uses this use case is not being banned | | |
| Normal Flow: | 1. **Unfollow an person’s account**   1. An User accesses another user’s profile who he/she is following  2. The user selects Unfollow  3. Unfollow successfully | | |
| Alternative Flows: | **None** | | |
| Exceptions: | **None** | | |
| Priority: | Medium | | |
| Frequency of Use: | Average a person unfollows 1,2 people per day. Usage load for this use case is all day | | |
| Business Rules: |  | | |
| Other Information: | None | | |
| Assumptions: | Assume that a person will unfollow 10 users and be unfollowed by 10 users. | | |

***p) Use case 19 (Actor User)***

|  |  |  |  |
| --- | --- | --- | --- |
| ID and Name: | **UC-19 Block users** | | |
| Created By: | Hà Việt Dũng | Date Created: | 26/01/2021 |
| Primary Actor: | User | Secondary Actors: | None |
| Description: | An User accesses the app from mobile phone, chooses an other user and decides to block him/her | | |
| Trigger: | An User indicates that he/she wants to block an other's account | | |
| Preconditions: | PRE-1. User logged into SOFA. | | |
| Post-conditions: | POST-1. Account of user who uses this use case is not being banned | | |
| Normal Flow: | 1. **Block an person’s account**   1. An User accesses another user’s profile  2. The user selects Block  3. User confirms her/his decision (see 1.0.E1)  4. Block successfully | | |
| Alternative Flows: | **None** | | |
| Exceptions: | **1.0.E1 User want to cancel decision**  1. If user cancel block decision, the SOFA terminates use case. | | |
| Priority: | Medium | | |
| Frequency of Use: | Average a person blocks 2,3 people per week. Usage load for this use case is all day | | |
| Business Rules: |  | | |
| Other Information: | Each person can see the people whom he/she is blocking. | | |
| Assumptions: | Assume that a person will block 10 users and be blocked by 5 users. | | |

***q) Use case 20 (Actor user)***

|  |  |  |  |
| --- | --- | --- | --- |
| ID and Name: | **UC-20 Report users** | | |
| Created By: | Hà Việt Dũng | Date Created: | 26/01/2021 |
| Primary Actor: | User | Secondary Actors: | None |
| Description: | An User accesses the app from mobile phone, chooses an other user and decides to report him/her | | |
| Trigger: | An User indicates that he/she wants to report an other's account because of some reasons | | |
| Preconditions: | PRE-1. User logged into SOFA. | | |
| Post-conditions: | POST-1. Account of user who uses this use case is not being banned | | |
| Normal Flow: | 1. **Report an person’s account**   1. An User accesses another user’s profile  2. The user selects Report  3. Choose (a) reason(s) why  4. User confirms her/his decision (see 1.0.E1)  5. Block successfully | | |
| Alternative Flows: | **None** | | |
| Exceptions: | **1.0.E1 User want to cancel decision**  1. If user cancel report decision, the SOFA terminates use case. | | |
| Priority: | High | | |
| Frequency of Use: | Average a person reports 1 people per week. Usage load for this use case is all day | | |
| Business Rules: |  | | |
| Other Information: | Each person can see the people whom he/she reported. | | |
| Assumptions: | Assume that a person will report 2 users. | | |

***r) Use case 21 (Actor user)***

|  |  |  |  |
| --- | --- | --- | --- |
| ID and Name: | **UC-21 View others’ profile** | | |
| Created By: | Hà Việt Dũng | Date Created: | 26/01/2021 |
| Primary Actor: | User | Secondary Actors: | None |
| Description: | An User accesses the app from mobile phone, chooses an other user to view his/her profile | | |
| Trigger: | An User indicates to view an other's profile. | | |
| Preconditions: | PRE-1. User logged into SOFA. | | |
| Post-conditions: | POST-1. The person who wants to view someone’s profile is not being blocked by the other person.  POST-2. Account of user who uses this use case is not being banned | | |
| Normal Flow: | 1. **View other people’s profile**   1. An User choose an other user  2. Select View Profile (see 1.0.E1) | | |
| Alternative Flows: | **None** | | |
| Exceptions: | **1.0.E1. The user whose account is viewed select to lock his/her account**  1. Noctice to the viewer  2. SOFA terminates the use case | | |
| Priority: | Medium | | |
| Frequency of Use: | Average a person views 5-10 people’s profile per day. Usage load for this use case is all day | | |
| Business Rules: |  | | |
| Other Information: | 1. Information that is displayed in someone’s profile is chosen by the owner. | | |
| Assumptions: | Assume that a person will view 1000 users’ profile. | | |

**s) Use case 22 (Actor User)**

|  |  |  |  |
| --- | --- | --- | --- |
| ID and Name: | **UC-22 Chat with other users** | | |
| Created By: | Hà Việt Dũng | Date Created: | 26/01/2021 |
| Primary Actor: | User | Secondary Actors: | None |
| Description: | An User accesses the app from mobile phone, chooses an other user to chat with him/her | | |
| Trigger: | An User indicates to chat with an other people. | | |
| Preconditions: | PRE-1. User logged into SOFA. | | |
| Post-conditions: | POST-1. The person who wants to chat with someone is not being blocked by the other person.  POST-2. Account of user who uses this use case is not being banned | | |
| Normal Flow: | 1. **Chat with someone**   1. An User choose another user  2. Select Chat  3. Type something or choose an image  4. Select Send (see 1.0.E1) | | |
| Alternative Flows: |  | | |
| Exceptions: | **1.0. E1. The file size of image that user want to send is too large**  1. Notice to the viewer  2. Return to step 3 of normal flow | | |
| Priority: | High | | |
| Frequency of Use: | Average a person chat with 10 people per day. Usage load for this use case is all day | | |
| Business Rules: |  | | |
| Other Information: | 1. User can see the messages which he/she chatted.  2. User can see a list of people whom he/she chatted with | | |
| Assumptions: | Assume that a person will chat with 1000 users. | | |

***t) Use case 23 (Actor user)***

|  |  |  |  |
| --- | --- | --- | --- |
| ID and Name: | **UC-23 Delete message in conversation** | | |
| Created By: | Hà Việt Dũng | Date Created: | 26/01/2021 |
| Primary Actor: | User | Secondary Actors: | None |
| Description: | An User accesses the app from mobile phone, choose Message, enter an conversation, select a message that he/she want to delete | | |
| Trigger: | An User indicates to delete a message in conversation. | | |
| Preconditions: | PRE-1. User logged into SOFA. | | |
| Post-conditions: | POST-1. User who uses this use case chatted with someone.  POST-2. Account of user who uses this use case is not being banned | | |
| Normal Flow: | **1.0. Delete message in conversation**  1. An User chooses Message  2. Select a conversation  3. Select a message  4. Choose type of delete (Erase or Withdraw)  5. Confirm Delete (see 1.0.E1)  6. Delete finish | | |
| Alternative Flows: |  | | |
| Exceptions: | **1.0.E1. User want to cancel decision**  1. If user cancel delete decision, the SOFA terminates use case. | | |
| Priority: | High | | |
| Frequency of Use: | Average a person deletes 3-5 messages per weeks. Usage load for this use case is all day | | |
| Business Rules: |  | | |
| Other Information: | 1. If user chooses “Erase”, the other person still see the message.  2. If user chooses “With draw”, both people can’t see the message anymore. | | |
| Assumptions: | Assume that a person will delete about 200 messages. | | |

***u) Use case 24 (Actor usecase)***

|  |  |  |  |
| --- | --- | --- | --- |
| ID and Name: | **UC-24 Delete conversation** | | |
| Created By: | Hà Việt Dũng | Date Created: | 26/01/2021 |
| Primary Actor: | User | Secondary Actors: | None |
| Description: | An User accesses the app from mobile phone, choose Message, choose an conversation that he/she want to delete | | |
| Trigger: | An User indicates to delete a conversation. | | |
| Preconditions: | PRE-1. User logged into SOFA. | | |
| Post-conditions: | POST-1. User who uses this use case chatted with someone.  POST-2. Account of user who uses this use case is not being banned | | |
| Normal Flow: | **1.0. Delete conversation**  1. An User chooses Message  2. Select a conversation  3. Choose Delete  4. Confirm Delete (see 1.0.E1)  5. Delete finish | | |
| Alternative Flows: |  | | |
| Exceptions: | **1.0.E1. User want to cancel decision**  1. If user cancel delete decision, the SOFA terminates use case. | | |
| Priority: | High | | |
| Frequency of Use: | Average a person deletes 1 messages per weeks. Usage load for this use case is all day | | |
| Business Rules: |  | | |
| Other Information: | 1. If user Delete a conversation, the other person still see this conversation. | | |
| Assumptions: | Assume that a person will delete about 20 conversation. | | |

***v) Use case 25 ( Actor User )***

|  |  |  |  |
| --- | --- | --- | --- |
| ID and Name: | **UC-25 View list of conversations** | | |
| Created By: | Hà Việt Dũng | Date Created: | 26/01/2021 |
| Primary Actor: | User | Secondary Actors: | None |
| Description: | An User accesses the app from mobile phone, choose Message to view list of conversations. | | |
| Trigger: | An User indicates to view list of conversations. | | |
| Preconditions: | PRE-1. User logged into SOFA. | | |
| Post-conditions: | POST-1. User who uses this use case chatted with someone.  POST-2. Account of user who uses this use case is not being banned | | |
| Normal Flow: | **1.0. View list of conversations**  1. An User chooses Message  2. View a list of conversations | | |
| Alternative Flows: | **None** | | |
| Exceptions: | **None** | | |
| Priority: | High | | |
| Frequency of Use: | Average a person view list conversation 10 times per day. Usage load for this use case is all day | | |
| Business Rules: |  | | |
| Other Information: | 1. List of conversation is sorted by time of latest message | | |
| Assumptions: | None | | |

***w) Use case 26 ( Actor User )***

|  |  |  |  |
| --- | --- | --- | --- |
| ID and Name: | **UC-26 Search conversation** | | |
| Created By: | Hà Việt Dũng | Date Created: | 26/01/2021 |
| Primary Actor: | User | Secondary Actors: | None |
| Description: | An User accesses the app from mobile phone, choose Message, type something in Search Box to search conversation. | | |
| Trigger: | An User indicates to search conversation by name of other users. | | |
| Preconditions: | PRE-1. User logged into SOFA. | | |
| Post-conditions: | POST-1. User who uses this use case chatted with someone.  POST-2. Account of user who uses this use case is not being banned | | |
| Normal Flow: | **1.0. Search conversation**  1. An User chooses Message  2. Type something into Search Box to search (See 1.0.E1)  3. View result list of conversations | | |
| Alternative Flows: | **None** | | |
| Exceptions: | **1.0.E1 User don’t enter anything into Search Box**  **1. Notice for user**  **2. SOFA terminate the use case** | | |
| Priority: | Medium | | |
| Frequency of Use: | Average a person search conversation 3-5 times per week. Usage load for this use case is all day | | |
| Business Rules: |  | | |
| Other Information: | None | | |
| Assumptions: | None | | |

***x) Use case 27 ( Actor User )***

|  |  |  |  |
| --- | --- | --- | --- |
| ID and Name: | **UC-27 View my profile** | | |
| Created By: | Hà Việt Dũng | Date Created: | 26/01/2021 |
| Primary Actor: | User | Secondary Actors: | None |
| Description: | An User accesses the app from mobile phone, choose Profile to view | | |
| Trigger: | An User indicates to view his/her profile. | | |
| Preconditions: | PRE-1. User logged into SOFA. | | |
| Post-conditions: | POST-1. Account of user who uses this use case is not being banned | | |
| Normal Flow: | **1.0. Search conversation**  1. An User chooses Profile  2. Display profile | | |
| Alternative Flows: | **None** | | |
| Exceptions: | **None** | | |
| Priority: | Medium | | |
| Frequency of Use: | Average a person view his/her profile 1-2 times per day. Usage load for this use case is all day | | |
| Business Rules: |  | | |
| Other Information: | None | | |
| Assumptions: | None | | |

***y) Use case 28 ( Actor User )***

|  |  |  |  |
| --- | --- | --- | --- |
| ID and Name: | **UC-28 Update my profile** | | |
| Created By: | Hà Việt Dũng | Date Created: | 26/01/2021 |
| Primary Actor: | User | Secondary Actors: | None |
| Description: | An User accesses the app from mobile phone, choose Profile to update | | |
| Trigger: | An User indicates to update his/her profile. | | |
| Preconditions: | PRE-1. User logged into SOFA. | | |
| Post-conditions: | POST-1. Account of user who uses this use case is not being banned | | |
| Normal Flow: | **1.0. Search conversation**  1. An User chooses Profile  2. Display profile  3. Choose update  4. Enter information that he/she wants to update  5. Choose Save (see 1.0.E1)  6. Update sucessfully | | |
| Alternative Flows: | **None** | | |
| Exceptions: | **1.0.E1. Some information is wrong format**  **1. Notice user for correct**  **2. Direct to the info which is wrong for user to correct** | | |
| Priority: | Medium | | |
| Frequency of Use: | Average a person view his/her profile 1-2 times per day. Usage load for this use case is all day | | |
| Business Rules: |  | | |
| Other Information: | None | | |
| Assumptions: | None | | |

***z) Use case 29 ( Actor User )***

|  |  |  |  |
| --- | --- | --- | --- |
| ID and Name: | **UC-29 Search people** | | |
| Created By: | Hà Việt Dũng | Date Created: | 26/01/2021 |
| Primary Actor: | User | Secondary Actors: | None |
| Description: | An User accesses the app from mobile phone, type something in Search Box to search people. | | |
| Trigger: | An User indicates to search other users by name. | | |
| Preconditions: | PRE-1. User logged into SOFA. | | |
| Post-conditions: | POST-1. Account of user who uses this use case is not being banned | | |
| Normal Flow: | **1.0. Search other users**  1. Type something into Search Box to search (See 1.0.E1)  2. View result list of other users | | |
| Alternative Flows: | **None** | | |
| Exceptions: | **1.0.E1 User don’t enter anything into Search Box**  **1. Notice for user**  **2. SOFA terminate the use case** | | |
| Priority: | Medium | | |
| Frequency of Use: | Average a person search other users 1-2 times per day. Usage load for this use case is all day | | |
| Business Rules: |  | | |
| Other Information: | Result list of searched people is sorted by interaction level. | | |
| Assumptions: | None | | |

***aa) Use case 30 ( Actor User )***

|  |  |  |  |
| --- | --- | --- | --- |
| ID and Name: | **UC-30 Search posts** | | |
| Created By: | Hà Việt Dũng | Date Created: | 26/01/2021 |
| Primary Actor: | User | Secondary Actors: | None |
| Description: | An User accesses the app from mobile phone, type something in Search Box to search posts. | | |
| Trigger: | An User indicates to search posts. | | |
| Preconditions: | PRE-1. User logged into SOFA. | | |
| Post-conditions: | POST-1. Account of user who uses this use case is not being banned | | |
| Normal Flow: | **1.0. Search other users**  1. Type something into Search Box to search (See 1.0.E1)  2. Switch into Post to search posts  3. View result list of posts | | |
| Alternative Flows: | **None** | | |
| Exceptions: | **1.0.E1 User don’t enter anything into Search Box**  **1. Notice for user**  **2. SOFA terminate the use case** | | |
| Priority: | Medium | | |
| Frequency of Use: | Average a person search other users 1-2 times per day. Usage load for this use case is all day | | |
| Business Rules: |  | | |
| Other Information: | Posts can be search by content, title, tag, caption,… | | |
| Assumptions: | None | | |

***ab) Use case 31 ( Actor User )***

|  |  |  |  |
| --- | --- | --- | --- |
| ID and Name: | **UC-31 View list of notifications** | | |
| Created By: | Hà Việt Dũng | Date Created: | 26/01/2021 |
| Primary Actor: | User | Secondary Actors: | None |
| Description: | An User accesses the app from mobile phone, choose Notification to view list of notifications. | | |
| Trigger: | An User indicates to view list of notifications. | | |
| Preconditions: | PRE-1. User logged into SOFA. | | |
| Post-conditions: | POST-1. Account of user who uses this use case is not being banned | | |
| Normal Flow: | **1.0. View list of notifiactions**  1. An User chooses Notification  2. View a list of conversations | | |
| Alternative Flows: | **None** | | |
| Exceptions: | **None** | | |
| Priority: | Medium | | |
| Frequency of Use: | Average a person view list notification 10 times per day. Usage load for this use case is all day | | |
| Business Rules: |  | | |
| Other Information: | 1. List of conversation is sorted by time of latest notification | | |
| Assumptions: | Each user views notification list about 1000 times | | |

***ac) . Use case 32 ( Actor User )***

|  |  |  |  |
| --- | --- | --- | --- |
| ID and Name: | **UC-32 View a notifications** | | |
| Created By: | Hà Việt Dũng | Date Created: | 26/01/2021 |
| Primary Actor: | User | Secondary Actors: | None |
| Description: | An User accesses the app from mobile phone, choose Notification and select a notification to view. | | |
| Trigger: | An User indicates to view a notification. | | |
| Preconditions: | PRE-1. User logged into SOFA. | | |
| Post-conditions: | POST-1. Account of user who uses this use case is not being banned  POST-2. Account has at least 1 notification | | |
| Normal Flow: | **1.0. View list of notifiactions**  1. An User chooses Notification  2. View a list of conversations  3. Select an notification to view (see 1.0.E1)  4. Direct to the post of notification | | |
| Alternative Flows: | **None** | | |
| Exceptions: | **1.0.E1 The post of notification is deleted**  **1. Notice user**  **2. SOFA terminate use case** | | |
| Priority: | Medium | | |
| Frequency of Use: | Average a person view a notification 15 times per day. Usage load for this use case is all day | | |
| Business Rules: |  | | |
| Other Information: | None | | |
| Assumptions: | Each user a notification about 3000 times | | |

***ad) Use case 33 ( Actor User )***

|  |  |  |  |
| --- | --- | --- | --- |
| ID and Name: | **UC-33 View hot posts** | | |
| Created By: | Hà Việt Dũng | Date Created: | 26/01/2021 |
| Primary Actor: | User | Secondary Actors: | None |
| Description: | An User accesses the app from mobile phone, choose HOT tab to view list of hot posts. | | |
| Trigger: | An User indicates to view list of hot posts. | | |
| Preconditions: | PRE-1. User logged into SOFA. | | |
| Post-conditions: | POST-1. Account of user who uses this use case is not being banned | | |
| Normal Flow: | **1.0. View list of hot post**  1. An User chooses HOT tab  2. Display list of hot post | | |
| Alternative Flows: | **None** | | |
| Exceptions: | **None** | | |
| Priority: | Medium | | |
| Frequency of Use: | Average a person view list of hot posts 5-10 times per day. Usage load for this use case is all day | | |
| Business Rules: |  | | |
| Other Information: | 1. Hot posts is the posts which belongs to fashionista or are the post popular post. | | |
| Assumptions: | Each user views list hot posts about 1000 times | | |

***ae) Use case 34 (Actor User)***

|  |  |  |  |
| --- | --- | --- | --- |
| ID and Name: | **UC-34 View recommended posts** | | |
| Created By: | Hà Việt Dũng | Date Created: | 26/01/2021 |
| Primary Actor: | User | Secondary Actors: | None |
| Description: | An User accesses the app from mobile phone, choose Recommend tab to view list of recommended posts. | | |
| Trigger: | An User indicates to view list of recommended posts. | | |
| Preconditions: | PRE-1. User logged into SOFA. | | |
| Post-conditions: | POST-1. Account of user who uses this use case is not being banned | | |
| Normal Flow: | **1.0. View list of hot post**  1. An User chooses Recommend tab  2. Display list of recommened posts | | |
| Alternative Flows: | **None** | | |
| Exceptions: | **None** | | |
| Priority: | Medium | | |
| Frequency of Use: | Average a person view list of hot posts 5-10 times per day. Usage load for this use case is all day | | |
| Business Rules: |  | | |
| Other Information: | 1. Recommended posts is the posts which is suitable with each users by measurements | | |
| Assumptions: | Each user views list recommended posts about 2000 times | | |

***af) Use case 35 (Actor User )***

|  |  |  |  |
| --- | --- | --- | --- |
| ID and Name: | **UC-35 Show suggest place of sale** | | |
| Created By: | Hà Việt Dũng | Date Created: | 26/01/2021 |
| Primary Actor: | User | Secondary Actors: | None |
| Description: | An User accesses the app from mobile phone, choose a post, select a product to see suggested place of sale | | |
| Trigger: | An User indicates to see where to buy a product. | | |
| Preconditions: | PRE-1. User logged into SOFA. | | |
| Post-conditions: | POST-1. Account of user who uses this use case is not being banned | | |
| Normal Flow: | **1.0. Show suggest place of sale of a product**  1. An User chooses a post  2. Select product in post  3. Display a list of place where sale the product | | |
| Alternative Flows: | **None** | | |
| Exceptions: | **None** | | |
| Priority: | Medium | | |
| Frequency of Use: | Average a person view suggested place of sale 10 times per day. Usage load for this use case is all day | | |
| Business Rules: |  | | |
| Other Information: | If in database this is no place sale the product, the list is empty. | | |
| Assumptions: | Each user views suggested place of sale about 500 times | | |

***ag) Use case 36 (Actor User)***

|  |  |  |  |
| --- | --- | --- | --- |
| ID and Name: | **UC-36 View Balance** | | |
| Created By: | Mai Văn Viên | Date Created: | 26/01/2021 |
| Primary Actor: | User | Secondary Actors: | None |
| Description: | User views balance and history transaction | | |
| Trigger: | None | | |
| Preconditions: | User has successfully logged in to the system | | |
| Post-conditions: | None | | |
| Normal Flow: | **1.0 User view balance**   1. In the screen “Profile”, user click “ Manage balance” | | |
| Alternative Flows: | None | | |
| Exceptions: | None | | |
| Priority: | High | | |
| Frequency of Use: | This use case sometimes happens and usage load for this use case is all day | | |
| Business Rules: | None | | |
| Other Information: | None | | |
| Assumptions: | None | | |

***ah) Use case 37 (Actor User)***

|  |  |  |  |
| --- | --- | --- | --- |
| ID and Name: | **UC-37 Top up account** | | |
| Created By: | Mai Văn Viên | Date Created: | 26/01/2021 |
| Primary Actor: | User | Secondary Actors: | None |
| Description: | User add more money to the account | | |
| Trigger: | None | | |
| Preconditions: | User has successfully logged in to the system | | |
| Post-conditions: | Balance of user up | | |
| Normal Flow: | **1.0 Top up account**   1. In the menu, the user clicks “Top up account” 2. System display screen “TopUp” 3. User enters “PIN code” and “Scratch card no” | | |
| Alternative Flows: | None | | |
| Exceptions: | None | | |
| Priority: | High | | |
| Frequency of Use: | This use case sometimes happens and usage load for this use case is all day | | |
| Business Rules: | BR-17, BR-18 | | |
| Other Information: | None | | |
| Assumptions: | None | | |

***ai) Use case 38 (Actor User)***

|  |  |  |  |
| --- | --- | --- | --- |
| ID and Name: | **UC-38 View list voucher** | | |
| Created By: | Mai Văn Viên | Date Created: | 26/01/2021 |
| Primary Actor: | User | Secondary Actors: | None |
| Description: | When user views a list of all voucher, they can see the number of vouchers in their account | | |
| Trigger: | View all voucher in account | | |
| Preconditions: | User has successfully logged in to the system | | |
| Post-conditions: | View all voucher in account | | |
| Normal Flow: | **1.0 View list voucher**   1. In the menu, user click “Manage voucher” 2. System displays all voucher of account | | |
| Alternative Flows: | None | | |
| Exceptions: |  | | |
| Priority: | Low | | |
| Frequency of Use: | This use case sometimes happens and usage load for this use case is all day | | |
| Business Rules: | None | | |
| Other Information: | None | | |
| Assumptions: | None | | |

***aj) Use case 39 (Actor User)***

|  |  |  |  |
| --- | --- | --- | --- |
| ID and Name: | **UC-36 View voucher detail** | | |
| Created By: | Mai Văn Viên | Date Created: | 26/01/2021 |
| Primary Actor: | User | Secondary Actors: | None |
| Description: | When the user views a list of all vouchers, they can see the number of vouchers in their account. When they click on one voucher, the system will redirect to a new screen to show these voucher details | | |
| Trigger: | View details of one voucher | | |
| Preconditions: | User has successfully logged in to the system | | |
| Post-conditions: | View details of one voucher | | |
| Normal Flow: | **1.0 View list voucher**   1. In the menu, the user clicks “Manage voucher” 2. System displays all voucher of account 3. User clicks once voucher 4. System displays voucher details | | |
| Alternative Flows: | None | | |
| Exceptions: | This use case sometimes happens and usage load for this use case is all day | | |
| Priority: | High | | |
| Frequency of Use: | None | | |
| Business Rules: | None | | |
| Other Information: | None | | |
| Assumptions: | None | | |

### 2.4 Feature for manager

#### a. Login

|  |  |  |  |
| --- | --- | --- | --- |
| ID and Name: | **UC-40 Login for Management system** | | |
| Created By: | TungNT | Date Created: | 26/01/2021 |
| Primary Actor: | Admin | Secondary Actors: | N/A |
| Description: | System administrators need to log into the system to perform management functions. | | |
| Trigger: | Users access the link "sofamanager/login". | | |
| Preconditions: | N/A | | |
| Postconditions: | N/A | | |
| Normal Flow: | **1.0 Login for Management system**   1. Users accesses the link "sofamanager/login". 2. The system displays login page. 3. User enters username and password. 4. User clicks on “Login” button. 5. The system displays home page. | | |
| Alternative Flows: | None | | |
| Exceptions: | **1.0.E1 User enters missing username or password fields and click “Login”**   1. The system notifies that the user must enter username and password   **1.0.E2 User entered incorrect username or password**   1. The system notifies that the user has entered an incorrect password or username | | |
| Priority: | High | | |
| Frequency of Use: | Approximately 5 users, average of one usage per day. | | |
| Business Rules: | **BR-01** | | |
| Other Information: | N/A | | |
| Assumptions: | N/A | | |

#### b. Log out

|  |  |  |  |
| --- | --- | --- | --- |
| ID and Name: | **UC-41 Logout of Management system** | | |
| Created By: | TungNT | Date Created: | 26/01/2021 |
| Primary Actor: | Admin | Secondary Actors: | N/A |
| Description: | Admin log out of the system | | |
| Trigger: | User clicks on the button “Logout” | | |
| Preconditions: | PRE-1. User is logged into the system | | |
| Postconditions: | N/A | | |
| Normal Flow: | 1. **Log out of Management system** 2. User click on the button “Log out” on the right top of page. 3. Log out the user from the system and move to the login page | | |
| Alternative Flows: | None | | |
| Exceptions: | N/A | | |
| Priority: | High | | |
| Frequency of Use: | N/A | | |
| Business Rules: | **BR-01** | | |
| Other Information: | N/A | | |
| Assumptions: | N/A | | |

#### c. Create new staff

|  |  |  |  |
| --- | --- | --- | --- |
| ID and Name: | **UC-42 Create new staff** | | |
| Created By: | TungNT | Date Created: | 26/01/2021 |
| Primary Actor: | Admin | Secondary Actors: | N/A |
| Description: | Admins with higher authority will be allowed to create accounts with smaller roles for system administration (except for users). | | |
| Trigger: | User clicks on the button “Create a new staff”. | | |
| Preconditions: | PRE-1. User is logged into the system as role of admin | | |
| Postconditions: | POST-1. Create accounts with smaller roles for system administration. | | |
| Normal Flow: | * 1. **Create new staff**  1. User is logged into the system as role of admin. 2. User click on the button “Account management”. 3. The system displays the page that show all staff of system. 4. User click on the button “Create a new staff”. 5. The system displays the “Add new staff” page. 6. User enters information of new account. 7. User click on the button “Add”. 8. The system notifies that the account has successfully added and redirects to the staff list page. | | |
| Alternative Flows: | None | | |
| Exceptions: | **1.0.E1 User enters missing some of the fields and click “Add”**   1. The system notifies that the user must not leave fields blank.   **1.0.E2 User enters an existing username**   1. The system notifies that username already exists and the user must enter another username. | | |
| Priority: | High | | |
| Frequency of Use: | N/A | | |
| Business Rules: | **BR-01**  **BR-02** | | |
| Other Information: | N/A | | |
| Assumptions: | N/A | | |

#### d. View User list

|  |  |  |  |
| --- | --- | --- | --- |
| ID and Name: | **UC-43 View User list** | | |
| Created By: | TungNT | Date Created: | 26/01/2021 |
| Primary Actor: | Admin | Secondary Actors: | N/A |
| Description: | The system administrator has the right to view all registered users that used the mobile application. | | |
| Trigger: | User clicks on the button “User management”. | | |
| Preconditions: | PRE-1. User is logged into the system as role of admin | | |
| Postconditions: | N/A | | |
| Normal Flow: | * 1. **View User list**  1. User is logged into the system. 2. The system displays home page with dashboard. 3. User click on the button “User management”. 4. The system displays list of all user that used the mobile application, sorted by date created. | | |
| Alternative Flows: | None | | |
| Exceptions: | **1.0.E1 No user to display**   1. The system notifies that there are currently no users using the application. | | |
| Priority: | High | | |
| Frequency of Use: | Approximately 5 users, average of one usage per day. | | |
| Business Rules: | **BR-01** | | |
| Other Information: | N/A | | |
| Assumptions: | N/A | | |

#### e. Search user

|  |  |  |  |
| --- | --- | --- | --- |
| ID and Name: | **UC-44 Search user** | | |
| Created By: | TungNT | Date Created: | 26/01/2021 |
| Primary Actor: | Admin | Secondary Actors: | N/A |
| Description: | The system administrator can search for users by name or username | | |
| Trigger: | User clicks on the button “Search”. | | |
| Preconditions: | PRE-1. User is logged into the system as role of admin | | |
| Postconditions: | POST-1. User must access to the user list page by click on the button “User management”. | | |
| Normal Flow: | * 1. **Search user**  1. User is logged into the system. 2. The system displays home page with dashboard. 3. User click on the button “User management”. 4. The system displays list of all user. 5. User enter the keyword to search input on the top of page. 6. User click on the button “Search” 7. The system displays the result that match with the keyword. | | |
| Alternative Flows: | None | | |
| Exceptions: | **1.0.E1 User doesn’t enter keyword and click “Search”**   1. The system does not respond if the user presses the “Search” button without a keyword.   **1.0.E2 The keyword entered by the user does not exist**   1. The system informs users that there are no user that match the keyword. | | |
| Priority: | High | | |
| Frequency of Use: | Approximately 5 users, average of many usage per day. | | |
| Business Rules: | **BR-01** | | |
| Other Information: | N/A | | |
| Assumptions: | N/A | | |

#### f. Ban user

|  |  |  |  |
| --- | --- | --- | --- |
| ID and Name: | **UC-45 Ban user** | | |
| Created By: | TungNT | Date Created: | 26/01/2021 |
| Primary Actor: | Admin | Secondary Actors: | N/A |
| Description: | The system administrator can ban an user that used the mobile application. | | |
| Trigger: | User clicks on the button “Ban”. | | |
| Preconditions: | PRE-1. User is logged into the system as role of admin | | |
| Postconditions: | POST-1. User must go to User management page.  POST-2. The status of user account must be “Active”.  POST-3. The user information can access by view user list and search. | | |
| Normal Flow: | * 1. **Ban user**  1. User is logged into the system. 2. The system displays home page with dashboard. 3. User click on the button “User management”. 4. The system displays list of all user. 5. User selects user and click to button “Ban”. 6. The system displays a popup to confirm. 7. User click on the button “Yes”. 8. The system notifies the user lock is successful, the user's status becomes inactive, the “Ban” button becomes “Unban” . | | |
| Alternative Flows: | None | | |
| Exceptions: | **1.0.E1 User click on “No” option on confirmation popup**   1. The system close the popup and do nothing. | | |
| Priority: | High | | |
| Frequency of Use: | Approximately 5 users, average of one usage per day. | | |
| Business Rules: | **BR-01**  **BR-03** | | |
| Other Information: | N/A | | |
| Assumptions: | N/A | | |

#### g. Reset password for user

|  |  |  |  |
| --- | --- | --- | --- |
| ID and Name: | **UC-46 Reset password for user** | | |
| Created By: | TungNT | Date Created: | 26/01/2021 |
| Primary Actor: | Admin | Secondary Actors: | N/A |
| Description: | The system administrator can reset password for user that used the mobile application when they can’t recovery password. | | |
| Trigger: | User clicks on the button “Reset password”. | | |
| Preconditions: | PRE-1. User is logged into the system as role of admin | | |
| Postconditions: | POST-1. User must go to User management page.  POST-2. The status of user account must be “Active”.  POST-3. The user information can access by view user list and search. | | |
| Normal Flow: | * 1. **Reset password for user**  1. User is logged into the system. 2. The system displays home page with dashboard. 3. User click on the button “User management”. 4. The system displays list of all user. 5. User selects user and click to edit icon. 6. The system shows the user detail page. 7. User click to button “Reset password”. 8. The system displays a popup to confirm. 9. User click on the button “Yes”. 10. The system notifies the user password has reset successful. | | |
| Alternative Flows: | None | | |
| Exceptions: | **1.0.E1 User click on “No” option on confirmation popup**  1. The system close the popup and do nothing. | | |
| Priority: | High | | |
| Frequency of Use: | Approximately 5 users, average of one usage per day. | | |
| Business Rules: | **BR-01**  **BR-04** | | |
| Other Information: | N/A | | |
| Assumptions: | N/A | | |

#### h. View post list

|  |  |  |  |
| --- | --- | --- | --- |
| ID and Name: | **UC-47 View post list** | | |
| Created By: | TungNT | Date Created: | 26/01/2021 |
| Primary Actor: | Admin | Secondary Actors: | N/A |
| Description: | The system administrator has the right to view all post that posted by users on the mobile application. | | |
| Trigger: | User clicks on the button “Post management”. | | |
| Preconditions: | PRE-1. User is logged into the system as role of admin | | |
| Postconditions: | N/A | | |
| Normal Flow: | * 1. **View post list**  1. User is logged into the system. 2. The system displays home page with dashboard. 3. User click on the button “Post management”. 4. The system displays list of all user that posted by users on the mobile application, sorted by date created. | | |
| Alternative Flows: | None | | |
| Exceptions: | **1.0.E1 No user to display**   1. The system notifies that there are currently no posts that posted by user. | | |
| Priority: | High | | |
| Frequency of Use: | Approximately 5 users, average of many usage per day. | | |
| Business Rules: | **BR-01** | | |
| Other Information: | N/A | | |
| Assumptions: | N/A | | |

#### i. View post detail

|  |  |  |  |
| --- | --- | --- | --- |
| ID and Name: | **UC-48 View post detail** | | |
| Created By: | TungNT | Date Created: | 26/01/2021 |
| Primary Actor: | Admin | Secondary Actors: | N/A |
| Description: | The system administrator can view the details of a post posted by the user on mobile application. | | |
| Trigger: | User clicks on a post in post list page. | | |
| Preconditions: | PRE-1. User is logged into the system as role of admin | | |
| Postconditions: | POST-1. User access the post list page by click on “Post management”. | | |
| Normal Flow: | * 1. **View post detail**  1. User is logged into the system. 2. The system displays home page with dashboard. 3. User click on the button “Post management”. 4. The system displays list of all post that posted by users on the mobile application, sorted by date created. 5. User lick on a post in post list. 6. The system displays the detail post page, show the detail of the post that clicked. | | |
| Alternative Flows: | None | | |
| Exceptions: | **1.0.E1 If the current post has been deleted**   1. The system notifies that this post does not exist or has been deleted. | | |
| Priority: | High | | |
| Frequency of Use: | Approximately 5 users, average of many usage per day. | | |
| Business Rules: | **BR-01** | | |
| Other Information: | N/A | | |
| Assumptions: | N/A | | |

#### j. Search user’s post

|  |  |  |  |
| --- | --- | --- | --- |
| ID and Name: | **UC-49 Search user’s post** | | |
| Created By: | TungNT | Date Created: | 26/01/2021 |
| Primary Actor: | Admin | Secondary Actors: | N/A |
| Description: | The system administrator can search for post by title | | |
| Trigger: | User clicks on the button “Search”. | | |
| Preconditions: | PRE-1. User is logged into the system as role of admin | | |
| Postconditions: | POST-1. User must access to the user list page by click on the button “Post management”. | | |
| Normal Flow: | * 1. **Search user’s post**  1. User is logged into the system. 2. The system displays home page with dashboard. 3. User click on the button “Post management”. 4. The system displays list of all user’s post. 5. User enter the keyword to search input on the top of page. 6. User click on the button “Search” 7. The system displays the result that match with the keyword. | | |
| Alternative Flows: | None | | |
| Exceptions: | **1.0.E1 User doesn’t enter keyword and click “Search”**   1. The system does not respond if the user presses the “Search” button without a keyword.   **1.0.E2 The keyword entered by the user does not exist**   1. The system informs users that there are no user that match the keyword. | | |
| Priority: | High | | |
| Frequency of Use: | Approximately 5 users, average of many usage per day. | | |
| Business Rules: | **BR-01** | | |

#### k. Delete user’s post

|  |  |  |  |
| --- | --- | --- | --- |
| ID and Name: | **UC-50 Delete user’s post** | | |
| Created By: | TungNT | Date Created: | 26/01/2021 |
| Primary Actor: | Admin | Secondary Actors: | N/A |
| Description: | The system administrator can delete user’s post. | | |
| Trigger: | User clicks on the delete icon. | | |
| Preconditions: | PRE-1. User is logged into the system as role of admin | | |
| Postconditions: | POST-1. User must access to the user’s post list page by click on the button “Post management”. | | |
| Normal Flow: | * 1. **Delete user’s post**  1. User is logged into the system. 2. The system displays home page with dashboard. 3. User click on the button “Post management”. 4. The system displays list of all user’s post. 5. User select a post and click on delete icon. 6. The system displays a popup to confirm. 7. User click on the button “Yes”. 8. Post’s is deleted and remove from list. | | |
| Alternative Flows: | None | | |
| Exceptions: | **1.0.E1 User click on “No” option on confirmation popup**   1. The system close the popup and do nothing. | | |
| Priority: | High | | |
| Frequency of Use: | Approximately 5 users, average of many usage per day. | | |
| Business Rules: | BR-01 | | |
| Other Information: | N/A | | |

#### l. Manage voucher

|  |  |  |  |
| --- | --- | --- | --- |
| ID and Name: | **UC-51 Manage voucher** | | |
| Created By: | TungNT | Date Created: | 26/01/2021 |
| Primary Actor: | Admin | Secondary Actors: | N/A |
| Description: | The system administrator has the right to view all voucher of the shop. | | |
| Trigger: | User clicks on the button “Voucher management”. | | |
| Preconditions: | PRE-1. User is logged into the system as role of admin | | |
| Postconditions: | N/A | | |
| Normal Flow: | * 1. **Manage voucher**  1. User is logged into the system. 2. The system displays home page with dashboard. 3. User click on the button “Voucher management”. 4. The system displays list of all voucher that created by shop on the mobile application, sorted by date created. | | |
| Alternative Flows: | None | | |
| Exceptions: | **1.0.E1 No voucher to display**   1. The system notifies that there are currently no vouchersthat created by user. | | |
| Priority: | High | | |
| Frequency of Use: | Approximately 5 users, average of many usage per day. | | |
| Business Rules: | **BR-01** | | |
| Other Information: | N/A | | |
| Assumptions: | N/A | | |

#### m. View report list

|  |  |  |  |
| --- | --- | --- | --- |
| ID and Name: | **UC-52 View report list** | | |
| Created By: | TungNT | Date Created: | 26/01/2021 |
| Primary Actor: | Admin | Secondary Actors: | N/A |
| Description: | The system administrator has the right to view all report that reported from mobile application user. | | |
| Trigger: | User clicks on the button “Report management”. | | |
| Preconditions: | PRE-1. User is logged into the system as role of admin | | |
| Postconditions: | N/A | | |
| Normal Flow: | * 1. **View report list**  1. User is logged into the system. 2. The system displays home page with dashboard. 3. User click on the button “Report management”. 4. The system displays list of all voucher that reported from mobile application user, sorted by date created. | | |
| Alternative Flows: | None | | |
| Exceptions: | **1.0.E1 No report to display**   1. The system notifies that there are currently no reportsthat reported by user. | | |
| Priority: | High | | |
| Frequency of Use: | Approximately 5 users, average of many usage per day. | | |
| Business Rules: | **BR-01** | | |
| Other Information: | N/A | | |
| Assumptions: | N/A | | |

#### n. View report detail

|  |  |  |  |
| --- | --- | --- | --- |
| ID and Name: | **UC-53 View report detail** | | |
| Created By: | TungNT | Date Created: | 26/01/2021 |
| Primary Actor: | Admin | Secondary Actors: | N/A |
| Description: | The system administrator can view the details of a report that report by the user on mobile application. | | |
| Trigger: | User clicks on a post in post list page. | | |
| Preconditions: | PRE-1. User is logged into the system as role of admin | | |
| Postconditions: | POST-1. User access the post list page by click on “Report management”. | | |
| Normal Flow: | * 1. **View report detail**  1. User is logged into the system. 2. The system displays home page with dashboard. 3. User click on the button “Report management”. 4. The system displays list of all user report that reported from mobile application user, sorted by date created. 5. User lick on a report in report list. 6. The system displays the detail report page, show the detail of the report that clicked. | | |
| Alternative Flows: | None | | |
| Exceptions: | N/A | | |
| Priority: | High | | |
| Frequency of Use: | Approximately 5 users, average of many usage per day. | | |
| Business Rules: | BR-01 | | |
| Other Information: | N/A | | |
| Assumptions: | N/A | | |

#### o. Manage recommendation data

|  |  |  |  |
| --- | --- | --- | --- |
| ID and Name: | **UC-54 Manage recommendation data** | | |
| Created By: | TungNT | Date Created: | 26/01/2021 |
| Primary Actor: | Admin | Secondary Actors: | N/A |
| Description: | The system administrator can view the data of the recommendation system, posts are highly interactive, enough points to be included in the recommendation system. | | |
| Trigger: | User clicks on the button “Recommendation data management” | | |
| Preconditions: | PRE-1. User is logged into the system as role of admin | | |
| Postconditions: | N/A | | |
| Normal Flow: | 1. **Manage recommendation data** 2. User is logged into the system. 3. The system displays home page with dashboard. 4. User click on the button “Recommendation data management”. 5. The system displays list of all post are highly interactive, enough points to be included in the recommendation system, sorted by interactive and point. | | |
| Alternative Flows: | None | | |
| Exceptions: | N/A | | |
| Priority: | High | | |
| Frequency of Use: | Approximately 5 users, average of many usage per day. | | |
| Business Rules: | **BR-01** | | |
| Other Information: | N/A | | |
| Assumptions: | N/A | | |

#### p. Manage user balance

|  |  |  |  |
| --- | --- | --- | --- |
| ID and Name: | **UC-55 Manage user balance** | | |
| Created By: | TungNT | Date Created: | 26/01/2021 |
| Primary Actor: | Admin | Secondary Actors: | N/A |
| Description: | The system administrator can view all deposit transactions and advertisements of users on the system, sorted by time. | | |
| Trigger: | User clicks on the button “Balance management” | | |
| Preconditions: | PRE-1. User is logged into the system as role of admin | | |
| Postconditions: | N/A | | |
| Normal Flow: | 1. **Manage user balance** 2. User is logged into the system. 3. The system displays home page with dashboard. 4. User click on the button “Balance management”. 5. The system displays list of all deposit transactions and advertisements of users on the system, sorted by time. | | |
| Alternative Flows: | None | | |
| Exceptions: | **1.0.E1 No balance to display**   1. The system notifies that there are currently no balancesthat created by user. | | |
| Priority: | High | | |
| Frequency of Use: | Approximately 5 users, average of many usage per day. | | |
| Business Rules: | **BR-01**  **BR-05** | | |
| Other Information: | N/A | | |
| Assumptions: | N/A | | |

#### q. Manage my profile

|  |  |  |  |
| --- | --- | --- | --- |
| ID and Name: | **UC-56 Manage my profile** | | |
| Created By: | TungNT | Date Created: | 26/01/2021 |
| Primary Actor: | Admin | Secondary Actors: | N/A |
| Description: | The system administrator can view and update their profile. | | |
| Trigger: | User clicks on the button “My profile” or avatar icon | | |
| Preconditions: | PRE-1. User is logged into the system as role of admin | | |
| Postconditions: | N/A | | |
| Normal Flow: | 1. **View my profile** 2. User is logged into the system. 3. The system displays home page with dashboard. 4. User click on the button “My profile” or avatar icon 5. The system displays the detail profile of user.   **2.0 Update my profile**   1. User update their profile by edit fields on profile page. 2. User click on the button “Update” 3. The system notifies that update successful and reload the updated profile. | | |
| Alternative Flows: | None | | |
| Exceptions: | **2.0.E1 User enters missing some of the fields and click “Update”**   1. The system notifies that the user must not leave fields blank.   **2.0.E2 User doesn’t enter keyword and click “Search”**   1. The system does not respond if the user presses the “Update” button without a edit. | | |
| Priority: | High | | |
| Frequency of Use: | Approximately 5 users. | | |
| Business Rules: | **BR-01**  **BR-04** | | |
| Other Information: | N/A | | |
| Assumptions: | N/A | | |

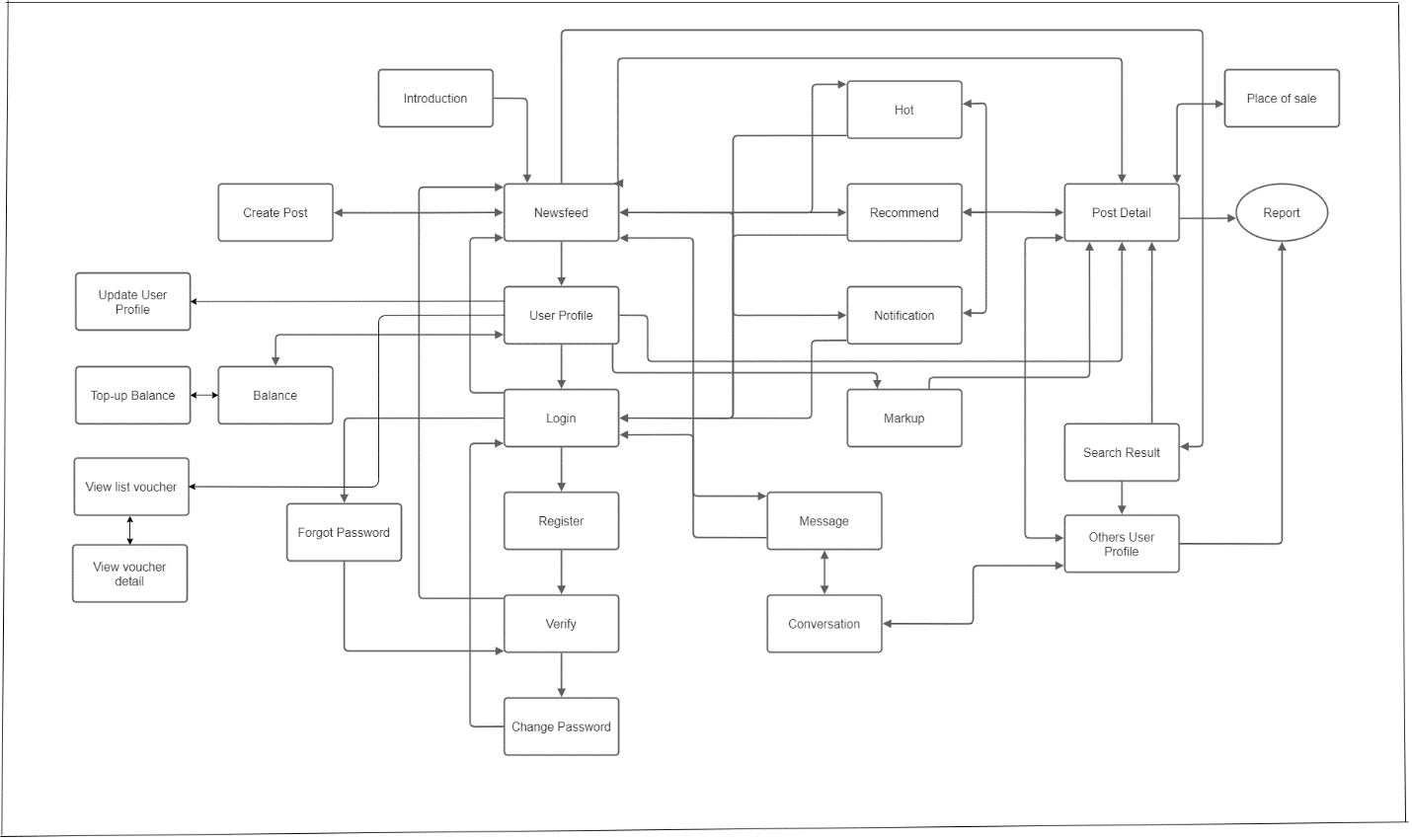
#### r. View user detail

|  |  |  |  |
| --- | --- | --- | --- |
| ID and Name: | **UC-57 View user detail** | | |
| Created By: | TungNT | Date Created: | 26/01/2021 |
| Primary Actor: | Admin | Secondary Actors: | N/A |
| Description: | The system administrator can view the details of an user that has used the system | | |
| Trigger: | User clicks on an user in user list page. | | |
| Preconditions: | PRE-1. User is logged into the system as role of admin | | |
| Postconditions: | POST-1. User access the post list page by click on “User management”. | | |
| Normal Flow: | * 1. **View user detail**  1. User is logged into the system as admin. 2. The system displays home page with dashboard. 3. Admin click on the button “User management”. 4. The system displays list of all user that has used the system, sorted by date created. 5. Admin lick on a user in user list. 6. The system displays the detail user page, show the detail of the user that clicked. | | |
| Alternative Flows: | None | | |
| Exceptions: | N/A | | |
| Priority: | High | | |
| Frequency of Use: | Approximately 5 users, average of many usage per day. | | |
| Business Rules: | **BR-01**  **BR-03**  **BR-04** | | |
| Other Information: | N/A | | |
| Assumptions: | N/A | | |

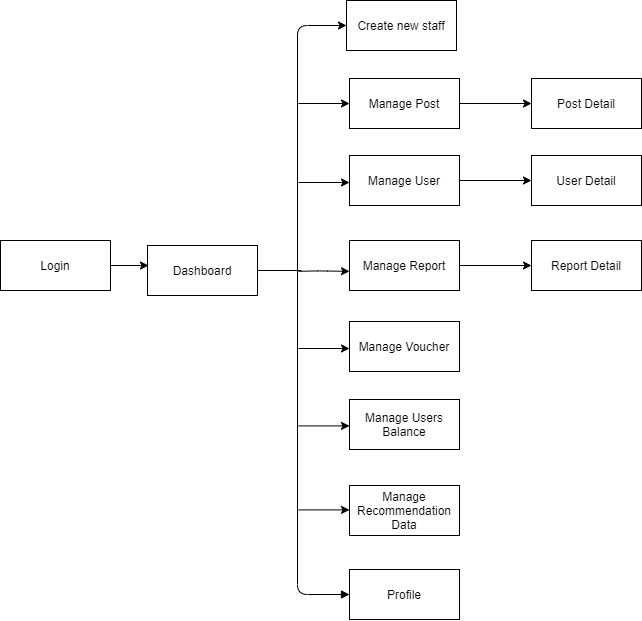
**3. Functional Requirements**

### 3.1 System Functional Overview

#### a. Screen Flow



Screen Flow - User and Guest



Screen Flow – Manager

#### b. Screen Details

|  |  |  |  |
| --- | --- | --- | --- |
| **#** | **Feature** | **Screen** | **Description** |
| 1 |  | Introduction | This screen use for introduce the application for user |
| 2 | View newsfeed | Newsfeed | This screen contain all posts which users have post |
| 3 | Logout, view my profile | User Profile | This screen use for view current user profile |
| 4 | Login | Login | This screen use for enter username and password the submit to the server for authentication |
| 5 | Register | Register | This screen allow guest can register their account |
| 6 | Forgot password, change password, verification | Verify | This screen use for get request send authentication code then check if it true it will forward to next step |
| 7 | Forgot password, change password | Change Password | This screen use for get password if user forgot their password or change password |
| 8 | Hot | Hot | This screen display all the post which have high react, rate or like, comment |
| 9 | Recommend | Recommend | This screen use for display the recommendation system |
| 10 | View list notification | Notification | This screen use to view list notification |
| 11 | Markup | Markup | This screen use to view all markup post |
| 12 | View post detail, like post, rate post, comment post, report post, markup, delete post | Post Detail | This screen use to view post detail |
| 13 | Show suggest place of sale | Place of Sale | This screen use to show place that sale the current item |
| 14 | Report | Report | This screen use for reporting |
| 15 | Search people, search post | Search Result | This screen use for searching people or post |
| 16 | Follow User, unfollow user, block user, report user, view other profile | Other User Profile | This screen use for view other user profile |
| 17 | Chat, delete conversation, view list conversation | Message | This screen use for viewing all of list conversation |
| 18 | Chat, delete message in conversation | Conversation | This screen use for chatting with other user |
| 19 | Create Post, Update Post | Create Post | This screen use for updating post, creating post |
| 20 | Update user profile | Update User Profile | This screen use for updating user profile |
| 21 | View balance, top-up account | Top-up Balance | This screen use for viewing balance and topping-up  balance |
| 22 | View list voucher | View list voucher | This screen use for viewing list voucher |
| 23 | View voucher detail | View voucher detail | This screen use for viewing voucher detail |

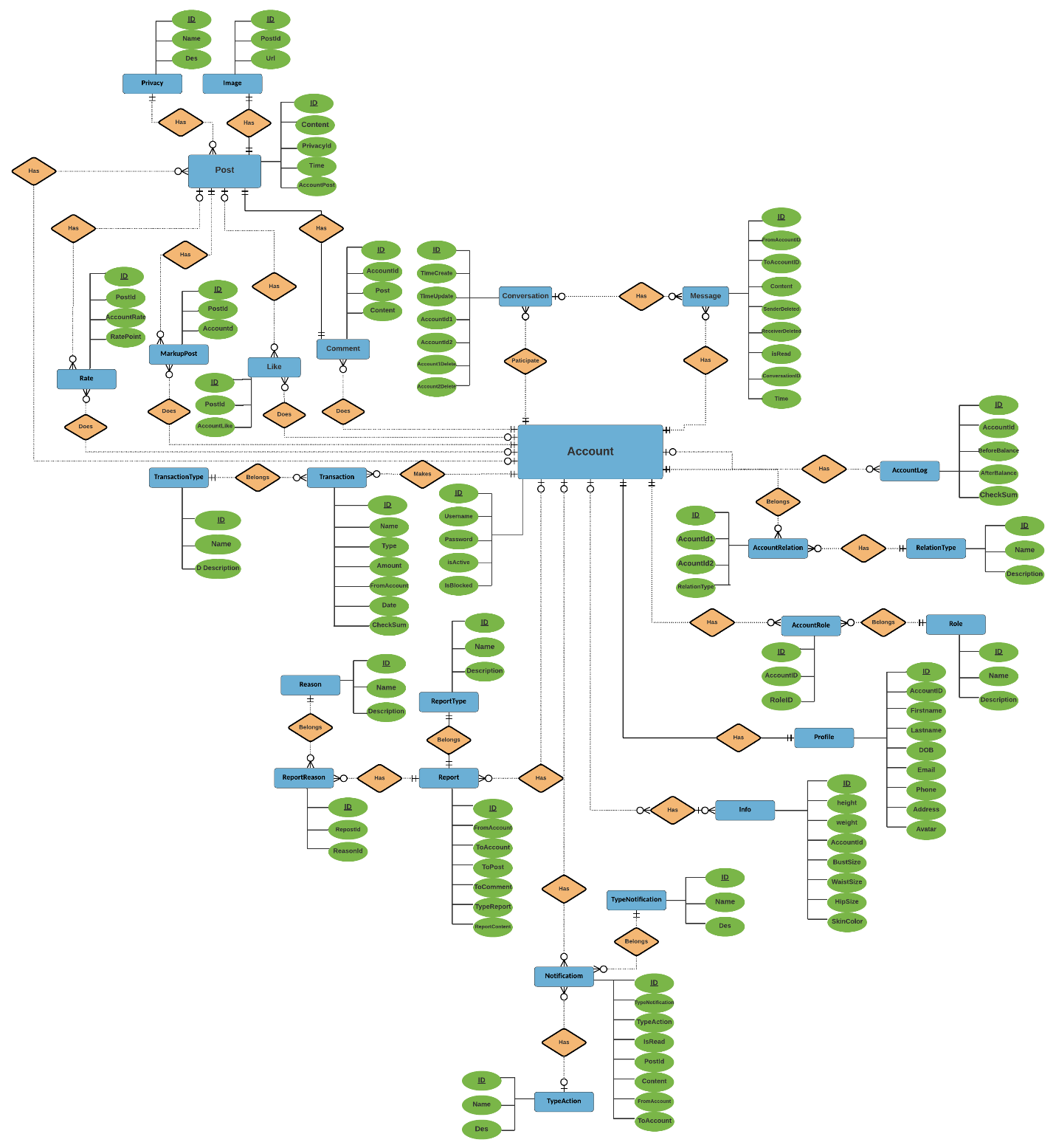
#### c. Screen Authorization

|  |  |  |  |
| --- | --- | --- | --- |
| **Screen** | **Guest** | **User** | **Manager** |
| Introduction | X | X |  |
| Newsfeed |  |  |  |
| * View newsfeed | X | X |  |
| Login | X |  |  |
| Register | X | X |  |
| Verify | X | X |  |
| Change Password |  | X |  |
| Forgot Password | X | X |  |
| Hot | X | X |  |
| Recommend |  | X |  |
| Notification |  | X |  |
| Mark-up |  | X |  |
| Post Detail |  |  |  |
| * View post detail | X |  |  |
| * Like post |  | X |  |
| * Rate post |  | X |  |
| * Comment |  | X |  |
| * Report |  | X |  |
| * Mark-up |  | X |  |
| * Delete |  | X |  |
| Place of Sale |  | X |  |
| Search Result |  | X |  |
| Other User profile |  | X |  |
| Message |  | X |  |
| Conversation |  | X |  |
| Create Post |  | X |  |
| Update User Profile |  | X |  |
| Balance |  | X |  |
| Top-up balance |  | X |  |
| View list voucher |  | X |  |
| View voucher detail |  | X |  |
| Manager Login |  |  | X |
| Dashboard |  |  | X |
| Create new Staff |  |  | X |
| Manage Post |  |  | X |
| Manage User |  |  | X |
| Manage User |  |  | X |
| Manage Report |  |  | X |
| Manage Voucher |  |  | X |
| Manage Users Balance |  |  | X |
| Manage Recommendation Data |  |  | X |
| Manager Profile |  |  | X |
| Post Detail |  |  | X |
| User Detail |  |  | X |
| Report Detail |  |  | X |

#### d. Non-Screen Functions

|  |  |  |  |
| --- | --- | --- | --- |
| **#** | **Feature** | **System Function** | **Description** |
| 1 | Web API | Web API | A web server or a web browser, knows as web development concept, usually limited to a web application's client-side, and thus usually does not include web server or browser implementation details. |
| 2 | Service | Architectural pattern | Applied within the service-orientation design paradigm, which aims to organize the services, within a service inventory, into a set of logical layers. Services that are categorized into a particular layer share functionality. |
| 3 | Recommendation | Clarifai API | Uses machine learning and deep neural networks to identify and analyze images. |
| 4 | Detect implicit content of image | Google Cloud's Vision API | Assign labels to images and quickly classify them into millions of predefined categories. Detect objects and faces, read printed and handwritten text, and build valuable metadata into your image catalog. |

#### e. Entity Relationship Diagram



### Mockup Screen

1. **Mockup Screen Newfeeds**



1. **Mockup Screen Hot**

****

1. **Mockup Screen Recommendation**

****

1. **Mockup Screen Post Detail**

****

1. **Mockup Screen Place of Sale**

****

1. **Mockup Screen Create Post**

****

1. **Mockup Screen Login**

****

1. **Mockup Screen Register**

****

1. **Mockup Screen Verify**

****

1. **Mockup Screen Change Password**

****

1. **Mockup Screen Introduction**

****

1. **Mockup Screen Notification**

****

1. **Mockup Screen User Profile**

****

1. **Mockup Screen Update user profile**
2. **Mockup Screen Markup**

****

1. **Mockup Screen Message**

****

1. **Mockup Screen Conversation**
2. **Mockup Screen Other user**

****

1. **Mockup Screen Balance**

****

1. **Mockup Screen Top-up balance**

****

1. **Mockup Screen View list voucher**

****

1. **Mockup Screen View voucher detail**

****

1. **Mockup Screen Report**

****

1. **Mockup Screen Search result**

****

1. **Mockup Screen Manager - Login Page**



1. **Mockup Screen Manager – Dashboard**



1. **Mockup Screen Manager - Create new staff**



1. **Mockup Screen Manager - Manage Post**



1. **Mockup Screen Manager - Manage User**



1. **Mockup Screen Manager - Manage Report**



1. **Mockup Screen Manager - Manage Voucher**



1. **Mockup Screen Manager - Manage User Balance**



1. **Mockup Screen Manager - Manage Recommendation Data**



1. **Mockup Screen Manager – Profile**



1. **Mockup Screen Manager - User detail**



1. **Mockup Screen Manager - Post Detail**



## 4. Non-Functional Requirements

### 4.1 External Interfaces

#### a. User Interfaces

UI-1: The SOFA application on smartphone is used entirely by touch and swipe.

UI-2: The SOFA application always includes a status bar at the bottom of the page to navigate between the application's features.

UI-3: The Management System conforms to the Dashboard Design User Experience Guidelines.

UI-4: The Management System in browser permits complete navigation by using the mouse alone, in addition to using mouse and keyboard combinations.

#### b. Software Interfaces

SI-1: Detect implicit content of image (Google cloud vision AI)

SI-1.1: After the photo is uploaded from the user's phone through the SOFA app, it will be sent through the image censorship system, while waiting for the censorship, the photo is only viewed as a preview.

SI-1.2: Through Vision AI, the system will evaluate and make a decision about whether a photo is suitable or not.

SI-1.3: When a posted photo is determined to be invalid, it is immediately removed from the preview interface.

SI-2: Recommendation System

SI-2.1: The Recommendation System collects information from photos posted from users.

SI-2.2: After collecting information, based on information about the user's body, the system will give suggestions on suitable outfits to the user through the recommendation page.

SI-3: Purchase Management System

SI-3.1: To allow users to top up the system: using Bao Kim's API

SI-3.2: To allows users to pay for advertising posts.

page.

SI-4: Virtual Search System (Google cloud vision AI)

SI-4.1: Through Google Cloud Vision AI, the system will allow users to find similar clothes by image

#### c. Hardware Interfaces

No hardware interfaces have been identified.

#### d. Communications Interfaces

CI-01: The system sends an email to the user to confirm transactions (top up, advertise) in the system.

CI-02: The system will send the OTP code to the phone number signed by the user to authenticate the action (top up, advertise, change information).

CI-03: In addition to the usual login, users can choose to log in with a Google or Facebook account.

### 4.2 Quality Attributes

#### a. Usability

* The application is designed with a trendy interface, simple and modern.
* The system has a friendly and good-looking GUI which makes it easy for users of all ages to access, helps users feel comfortable when using the application.
* The interface of main functions is learned, inherited and developed from previous successful systems such as Facebook, Instagram, ... This gives users a sense of familiarity when using the software and helps users access the functions of the system more effectively.

#### b. Reliability

* The system complies with business rules.
* The system must ensure smooth and stable running.
* The display of personal information such as date of birth, measurements, ... will be decided by the owner.
* The system must ensure that it does not crash when get exception.

#### c. Performance

* Two functions (Message and Notification) are real-time update. When a new message or notification appears, users will receive it almost immediately
* The system allows up to 100000 users
* The system allows 20000 users at a time

#### d. Security

* Each email is only validated for one account. User can use Google or Facebook account to login as long as it has the same email as the one used to authenticate the account.
* The system must use HTTPS to encrypt the information transferred between clients and servers.
* Use the JWT Bearer Token to authenticate users
* Before uploading a file to the system, the file will be verified for the correct file format which helps reduce the likelihood of being hacked with webshell
* The user authorization table is as follow:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Function** | | **Super Admin** | **Staff** | **Users** |
| Login | | ✔ | ✔ | ✔ |
| Logout | | ✔ | ✔ | ✔ |
| Manage staffs | | ✔ |  |  |
| Manage accounts | | ✔ | ✔ |  |
| Manage posts | Report posts |  |  | ✔ |
| Ban posts | ✔ | ✔ |  |
| Other managements |  |  | ✔ |
| Manage users | | ✔ | ✔ |  |
| Manage notifications | |  |  | ✔ |
| Manage messages | |  |  | ✔ |
| Manage vouchers | | ✔ | ✔ |  |
| Manage configure email | | ✔ |  |  |
| Manage react of posts | |  | ✔ | ✔ |
| Manage data of recommendation system | | ✔ |  |  |

#### e. Maintainability

This system uses RESTFUL API structure with ASP.NET Core, with the following advantages to maintain:

* We can easily maintain our application because of separation of concern.
* At the same time, we can split many developers' work at a time. It will not affect one developer work to another developer work.
* It supports TTD (test-driven development). We can create an application with unit test. We can write won test case.

#### f. Portability

* System is created by using API and React Native which can be developed in the future to supports responsiveness for all screens: mobiles, desktops, tablets, laptops (For candidate view)

#### g. Dependability

N/A

#### h. Supportability

N/A

#### i. Design Constraints

**N/A**

#### k. Support Documents

**N/A**

#### l. Purchased Components

**N/A**

## 5. Other Requirements

### 5.1 Appendix1 - Messages List

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **#** | **Message code** | **Message Type** | **Context** | **Content** |
| 1 | MSG01 | In line | There is not any search result | *No search result.* |
| 2 | MSG02 | In red, under the text box | Input-required fields are empty | *The \* field is required.* |
| 3 | MSG03 | Toast message | Updating information successfully | *Update profile successfully.* |
| 4 | MSG04 | Toast message | Login successfully | *Login successfully.* |
| 5 | MSG05 | Toast message | Change password successfully | *Change password successfully.* |
| 6 | MSG06 | Toast message | Confirming email sent successfully | *A confirmation email has been sent to {email\_address}.* |
| 7 | MSG07 | In line | Account has been blocked | *Account has been blocked, please check email or contact admin* |
| 8 | MSG08 | In line | Account or password incorrect | *Account or password incorrect please try again* |
| 9 | MSG09 | In red, under the text box | Input value length > max length | *Exceed max length of {max\_length}.* |
| 10 | MSG10 | In line | Username or password is not correct when clicking sign-in | *Incorrrect user name or password. Please check again.* |
| 11 | MSG11 | In red, under the text box | Email not valid | *Please enter valid email* |

### Appendix2 - API List

**5.2.1. Bao Kim API**

Bao Kim Payment Platform is an open payment platform, Bao Kim provides a full range of APIs that allow users to integrate their application (web / app) with Bao Kim in order to receive payment orders and checks. account, transaction control, automatic trading, ...

**5.2.2. Vision API Product Search**

Vision API Product Search allows retailers to create products, each containing reference images that visually describe the product from a set of viewpoints. Retailers can then add these products to product sets. Currently Vision API Product Search supports the following product categories: homegoods, apparel, toys, packaged goods, and general .

When users query the product set with their own images, Vision API Product Search applies machine learning to compare the product in the user's query image with the images in the retailer's product set, and then returns a ranked list of visually and semantically similar results.