

**Mini Social Media**

**Software Requirement Specification**

**Project Code: Midia**

**Document Code: Midia – v1.0**

– Hanoi, June 2025 –

**Table of Contents**

[Chapter 1. Record of Changes 3](#_heading=h.iznvygw4d8v6)

[Chapter 2. Software Requirement Specification 4](#_heading=h.6n2ql69lro11)

[1. Product Overview 4](#_heading=h.y8jsajiuh4yc)

[2. User Requirements 5](#_heading=h.9nrpj1ym574p)

[2.1 Actors 5](#_heading=h.j4onym1mgx8f)

[2.2 Use Cases 5](#_heading=h.o4brx8dmsn8b)

[3. Software Features 8](#_heading=h.yp81ry97qlem)

[3.1 Functional Overview 8](#_heading=h.mynokzeyr42n)

[3.2 Notification 13](#_heading=h.14q22l67zr77)

[3.3 Content Management 18](#_heading=h.6sl90jo9q84y)

[3.4 User Settings 25](#_heading=h.eoqetp5kyuww)

[3.5 Follow System 28](#_heading=h.cw0tljjouupf)

[3.6 Profile Management 31](#_heading=h.ybr0uouo3xeb)

[3.7 Account Management 38](#_heading=h.ag9815jha0g)

[3.8 Home 44](#_heading=h.atwkctjqu00n)

[4. Non-Functional Requirements 47](#_heading=h.2sjortjay854)

[4.1 External Interfaces 47](#_heading=h.1ff0n3miqrlc)

[4.2 Quality Attributes 47](#_heading=h.f9732wxe2314)

[5. Requirement Appendix 48](#_heading=h.7gyhlu2vrpw)

[5.1 Business Rules 48](#_heading=h.3b182z1oundb)

[5.2 Common Requirements 49](#_heading=h.3eu1om947yh7)

[5.3 Application Messages List 49](#_heading=h.v8qdkuhs3end)

# Chapter 1. Record of Changes

| **Date** | **A\* M, D** | **In charge** | **Change Description** |
| --- | --- | --- | --- |
| 13/06/25 | A | ThangLT | First implementation. |
| 13/06/25 | A | SonDH | Add actor and actor description |
| 13/06/25 | D | SonDH | Delete unnecessary actor |
| 13/06/25 | A | SonDH | Add screen description Add screen flow Add screen authorization Add non-screen functions |
| 13/06/25 | A | ThangLT | Add context diagram and description. |
| 13/06/25 | A | SonDH | Add use case diagram |
| 13/06/25 | A | HoanTX | Add use case specification for User Setting |
| 14/06/25 | A | SonDH | Add entity relationship diagram and entity description  Add entity detail |
| 14/06/25 | A | ThangLT | Add mockup, screen item description for notification feature. Add Check Recent Noti, Atc on a Noti use case. |
| 14/06/25 | A, M | HoanTX | Add non-functional requirements Modify the use case’s specification |
| 15/06/25 | A | HoanTX | Add Edit profile, Post details and Follow suggestions use case specification |
| 15/06/25 | M | HoangMH | Modify ERD, entity detail |
| 15/06/25 | A | SonDH | Add use case View Posts, Create Post |
| 15/06/25 | A | KhanhNT | Add use case for home page screen. |
| 15/06/25 | M | ThangLT | Modify use case for post details Modify context diagram |

\*A - Added M - Modified D - Deleted

# Chapter 2. Software Requirement Specification

## 1. Product Overview

The Midia System is a lightweight social-media platform. Release 1.0 replaces the current set of static UI screens with full backend logic, APIs, and databases, allowing guests to browse public content and authenticated users to post, follow, and interact in real time. The context diagram (Figure 1-1) shows the boundary of the Midia application and its two external actors for the MVP. Future releases are expected to introduce third-party integrations such as push-notification services, content-moderation pipelines, etc.

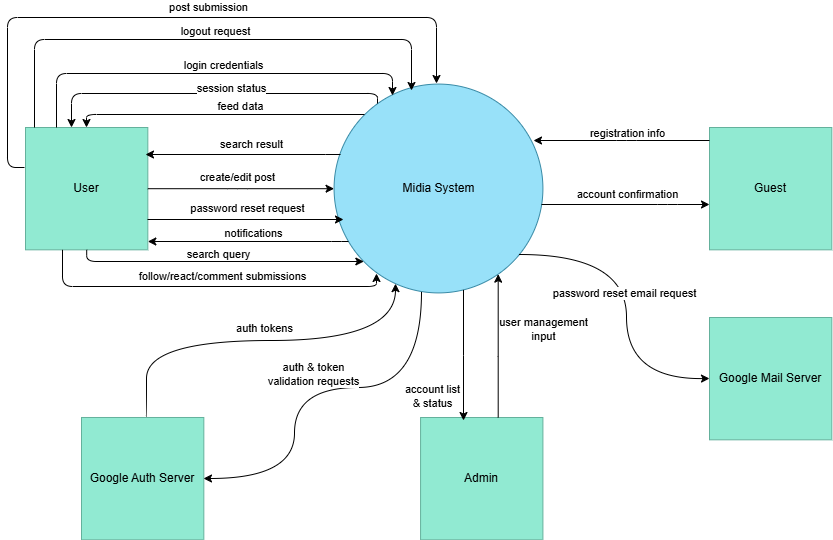


Figure 1.1: Midia System overview

As reflected in Figure 1-1 (Context Diagram), Midia exchanges information with two external entities. Each entity is outside the system boundary and communicates only through the labelled data flows.

* **Guest:** An unauthenticated visitor who consumes public-feed request, profile-view request, and search query flows. Midia replies with public-feed data, profile dataset, and search-result set. Guests cannot modify data or access private content.
* **Authenticated User:** A registered user with an active session token. Sends content payloads (new post, edit, like, comment), relationship records (follow/unfollow), real-time message packets, settings documents, and search queries. Receives personalised-feed data, post-confirmation set, interaction notifications, message packets, and search-result sets.

## 2. User Requirements

### 2.1 Actors

| **#** | **Actor** | **Description** |
| --- | --- | --- |
| 1 | Guest | A person who has not signed in or signed up. Can access login/register page, home or global search |
| 2 | User | An user who has signed up or signed in. Can create/edit posts, like, comment, follow others, manage profile, use messaging and interact with the system |
| 3 | Admin | An user which not only can use all of function of user, but also can manage user list and manage account status |

### 2.2 Use Cases

#### 2.2.1 Diagram(s)

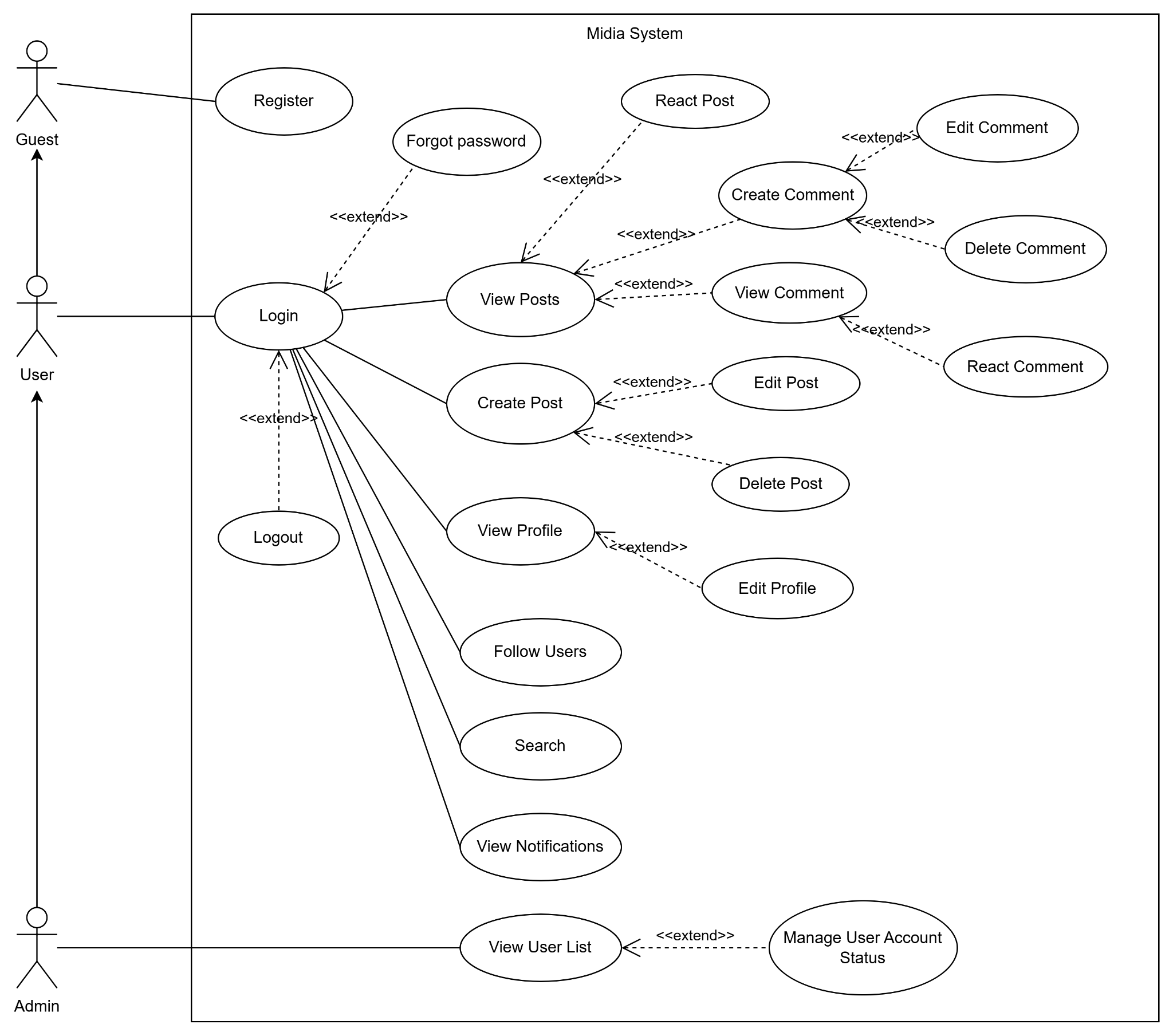
******

Figure 2-1: Use case diagram of Midia system

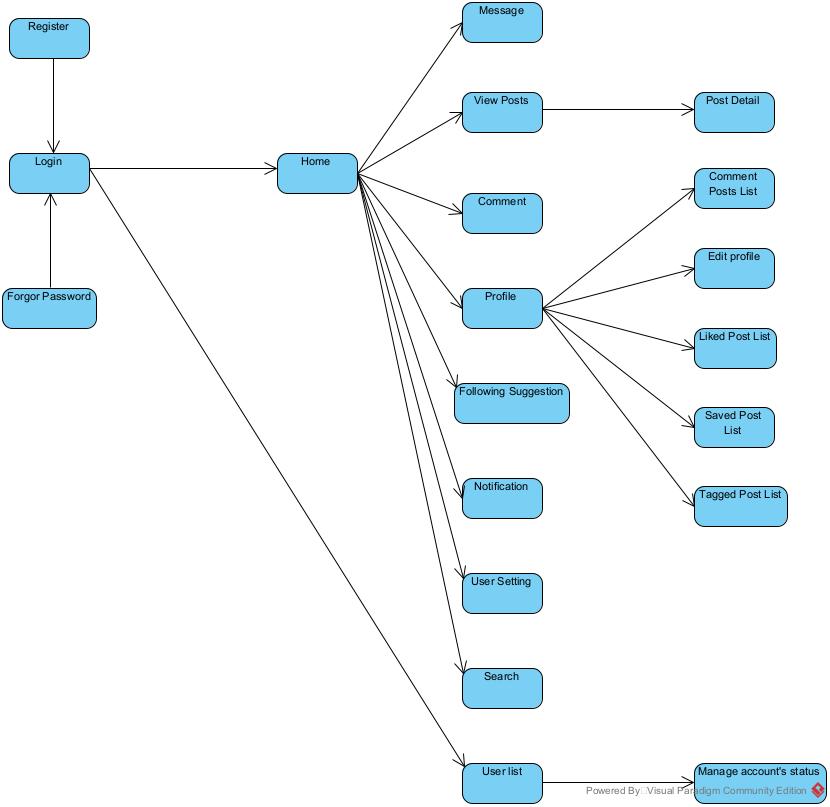
#### 2.2.2 Descriptions

| **ID** | **Group function** | **Use Case** | **Actors** | **Use Case Description & Main Flow** |
| --- | --- | --- | --- | --- |
| AM-01 | Account Management | Register | Guest | Allows a new guest to create an account.  Guest enters info → System validates → Creates account |
| AM-02 | Login | User | Allows a registered user to access the system.  User enters credentials → System authenticates → Grants access. |
| AM-03 |  | Forgot password | User | Allows a user to reset their password. User enters email → System sends reset link → User sets new password. |
| AM-04 |  | Logout | User | Allows a logged-in user to end their session. User clicks logout → System terminates session. |
| PM-01 | Profile Management | View Profile | User | Allows a user to view their own or another user's profile.  User navigates to profile → System displays info. |
| PM-02 | Edit Profile | User | Allows a user to update their profile information.  User navigates to edit profile page → Submits changes → System saves. |
| PM-03 | Follow Users | User | Allow user follow another user. User click Follow → System creates relationship → Updated. |
| PO-01 | Post Management | Create Post | User | Allows a user to share a new post with media and a caption.  User uploads media → Adds caption → Clicks share. |
| PO-02 | View Posts | User | Allows a user to view posts from followed users in their feed.  User opens feed → System displays posts. |
| PO-03 | Edit Post | User | Allows a user to modify their own post's caption.  User selects 'edit' on post → Makes changes → System updates. |
| PO-04 | Delete Post | User | Allows a user to delete their own post.  User selects 'delete' on post → Confirms → System removes post. |
| PO-05 | React Post | User | Allows a user to "like" or react to a post.  User clicks react button → System records the reaction. |
| CM-01 | Comment Management | Create Comment | User | Allows a user to add a comment to a post.  User types comment → Clicks post → System saves comment. |
| CM-02 | View Comment | User | Allows a user to view all comments on a specific post.  User opens post → System displays comments. |
| CM-03 | Edit Comment | User | Allows a user to modify their own comment.  User selects 'edit' → Makes changes → System updates comment. |
| CM-04 | Delete Comment | User | Allows a user to delete their own comment.  User selects 'delete' → Confirms → System removes comment. |
| CM-05 | React Comment | User | Allows a user to "like" or react to a comment.  User clicks react on a comment → System records reaction. |
| EF-01 | Extra Features | Search | User | Allows a user to search for other users or posts.  User enters query → System displays matching results. |
| EF-02 | View Notifications | User | Allows a user to see updates (likes, new comments, follows).  User opens notifications → System displays list. |
| AD-01 | Admin Functions | View User List | Admin | Allows an admin to see a list of all registered users.  Admin navigates to user list → System displays all users. |
| AD-02 | Manage User Account Status | Admin | Allows an admin to ban or activate a user account.  Admin selects user → Chooses action → System updates status. |

## 3. Software Features

### 3.1 Functional Overview

#### 3.1.1 Screens Flow



#### 3.1.2 Screen Descriptions

| **#** | **Feature** | **Screen** | **Description** |
| --- | --- | --- | --- |
| 1 | Authentication & Session | Login | Allows users to log into their account using email/username and password. |
| 2 | Authentication & Session | Register | Enables new user to create an account by providing email, username, password |
| 3 | Navigation | Global Layout | Provides consistent header and navigation bar visible across authenticated screens includes link to Home, Search and Message |
| 4 | Home Feed | Newsfeed Page | Display a list of posts from followed users, sorted by recency. Support infinite scroll |
| 5 | Content Management | Create Post | Modal dialog allowing users to create a new post, including for content, media uploads and tags |
| 6 | Content Management | PostDetails View | Detailed view of a single post including full content, comments, like count and user interaction |
| 7 | Content Management | Archive View | Display a list of archived posts for the logged-in user in grid or list format |
| 8 | Social Interaction | Comments | Interface showing comments on a post. Allows adding, editing and deleting own comments. |
| 9 | Social Interaction | Likes Post List | Shows a list of users who liked a particular post, along with follow/unfollow actions |
| 10 | User Profiles | Profile Page | Display user’s posts, saved content and tagged posts in tabbed views- |
| 11 | User Profiles | Edit Profile | Allows users to update their profile information including avatar, bio and personal links |
| 12 | Notifications | Notifications Center | List recent notifications (likes, comments, follows) with link to related content |
| 13 | Follow System | Follow Suggestions | Display recommended users to follow with “Follow” button for each |
| 14 | Global Search | Search | Allows users to search posts, users and tags. Display results in categorized sections |
| 15 | User settings | User settings | Provides UI to manage account settings, privacy options and notification preferences |
| 16 | Real-Time Messaging | Message | Real-time chat interface showing list of conversations and individual chat threads with live updates |
| 17 | User Management | User list | Display a table list of users. When click detail, show all the information of that user |
| 18 | User Management | Manage account |  |

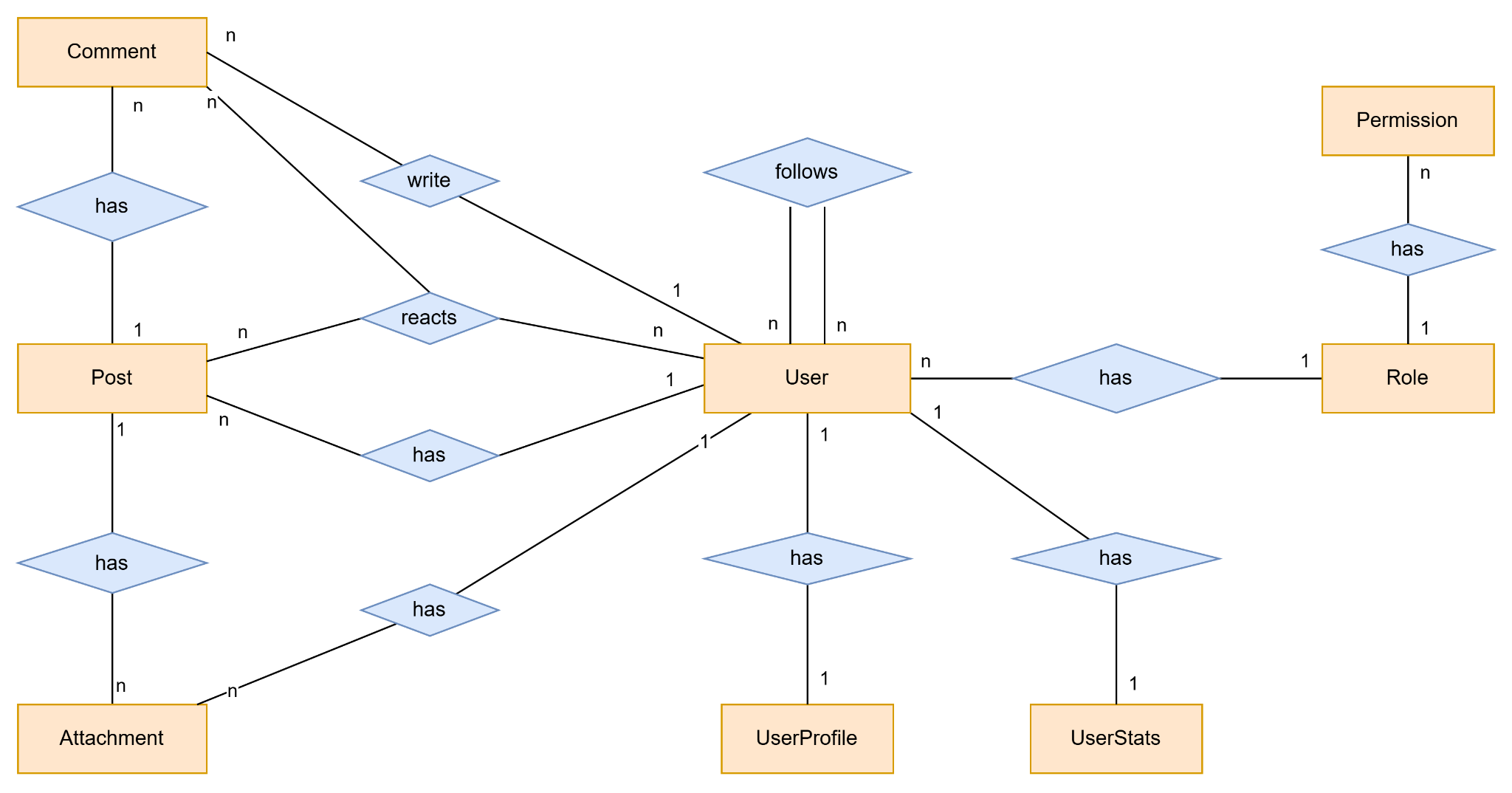
#### 3.1.3 Screen Authorization

| **#** | **Screen** | **Guest** | **User** |
| --- | --- | --- | --- |
| 1 | Login | X |  |
| 2 | Register | X |  |
| 3 | Newsfeed Page |  | X |
| 4 | Create post |  | X |
| 5 | Comment |  | X |
| 6 | Post Details View |  | X |
| 7 | Likes Post List |  | X |
| 8 | Profile Page |  | X |
| 9 | Edit Profile |  | X |
| 10 | Notification Center |  | X |
| 11 | Follow Suggestions |  | X |
| 12 | Search |  | X |
| 13 | User settings |  | X |
| 14 | Message |  | X |

#### 3.1.4 Non-Screen Functions

| **#** | **Feature** | **System Function** | **Description** |
| --- | --- | --- | --- |
| 1 | Authentication & Session | Auth API | Provides login, registration, and logout endpoints to manage authentication via RESTful APIs. |
| 2 | Authentication & Session | JWT / Session Management | Issues, validates, and refreshes JWT tokens to maintain secure session state across the system. |
| 3 | Real-Time Messaging | WebSocket Server | Manages real-time bidirectional communication between users for chat and typing indicators |

#### 3.1.5 Entity Relationship Diagram



**Entities Description**

| **#** | **Entity** | **Description** |
| --- | --- | --- |
| 1 | User | Represents user’s information such as id, username, email, encrypted password, followers, followings, status and audit information. |
| 2 | UserProfile | Contains personal information of users, including full name, avatar, phone number, date of birth and bio. |
| 3 | UserStats | Represents user statistics data including number of posts, number of followers, number of followings |
| 4 | Post | Stores user-generated posts, including caption, attachments, visibility and interaction statistics |
| 5 | Comment | Stores user comments on posts, which may include attachments and reaction statistics |
| 6 | Attachment | Represents media files such as images, videos attached to posts or comments |
| 7 | Role | Represents user’s role in the system |
| 8 | Permission | Represents actions user can perform in the system |

#### 3.1.6 Entity Details

##### 3.1.6.1 User

| **#** | **Attribute name** | **PK** | **Type** | **Mandatory** | **Description** |
| --- | --- | --- | --- | --- | --- |
| 1 | id | x | uuid | Yes | Primary key that uniquely identifies each user. |
| 2 | user\_name |  | string | Yes | Display name of the user |
| 3 | email |  | string | Yes | User’s email address |
| 4 | password |  | string (hashed) | Yes | Encrypted password |
| 5 | created\_at |  | timestamp | Yes | Timestamp when the user was created. |
| 6 | updated\_at |  | timestamp | No | Timestamp when the user was updated. |
| 7 | deleted\_at |  | timestamp | No | Soft-delete timestamp. If set, the account is considered deleted. |
| 8 | is\_locked |  | boolean | Yes (default: false) | Indicates whether the user's account is locked |
| 9 | profile |  | Reference | Yes | Reference to user’s profile |
| 10 | stats |  | Reference | Yes | Reference to user’s stats |
| 11 | followers |  | Set<Reference> | No | A set of references to other users who follow this user. |
| 12 | followings |  | Set<Reference> | No | A set of references to other users this user follows |
| 13 | uid |  | string | Yes | Automatic unique generated internal id |

##### 3.1.6.2 UserProfile

| **#** | **Attribute name** | **PK** | **Type** | **Mandatory** | **Description** |
| --- | --- | --- | --- | --- | --- |
| 1 | uid | x | string | Yes | Automatic unique generated profile id |
| 2 | avatar |  | string | Yes | URL or reference to avatar image |
| 3 | bio |  | string | No | User’s biography |
| 4 | user\_id |  | Reference | Yes | Associated user |

##### 3.1.6.3 UserStats

| **#** | **Attribute name** | **PK** | **Type** | **Mandatory** | **Description** |
| --- | --- | --- | --- | --- | --- |
| 1 | uid | x | string | Yes | Automatic unique generated profile id |
| 2 | num\_posts |  | int | Yes | Number of posts made by the user |
| 3 | num\_following |  | int | Yes | Number of users the user is following |
| 4 | num\_followers |  | int | Yes | Number of followers |
| 5 | user\_id |  | Reference | Yes | Associated user |

##### 3.1.6.4 Post

| **#** | **Attribute name** | **PK** | **Type** | **Mandatory** | **Description** |
| --- | --- | --- | --- | --- | --- |
| 1 | id | x | uuid | Yes | Primary key that uniquely identifies the posts. |
| 2 | caption |  | string | Yes | The content of the posts |
| 3 | created\_at |  | timestamp | Yes | Time when the post was created |
| 4 | visibility |  | enum | Yes | Visibility setting of the post (‘PUBLIC’,‘PRIVATE’, ‘FRIENDS’) |
| 5 | deleted\_at |  | timestamp | No | Time when post is deleted. |
| 6 | updated\_at |  | timestamp | No | Timestamp when the post was updated. |
| 7 | author\_id |  | Reference | Yes | Reference to user who create the post |

##### 3.1.6.5 Comment

| **#** | **Attribute name** | **PK** | **Type** | **Mandatory** | **Description** |
| --- | --- | --- | --- | --- | --- |
| 1 | id | x | uuid | Yes | Primary key that uniquely identifies the comments. |
| 2 | content |  | string | Yes | The text content of the comment |
| 3 | created\_at |  | timestamp | Yes | Time when comment was created |
| 4 | update\_at |  | timestamp | No | Timestamp when the comment was updated. |
| 5 | deleted\_at |  | timestamp | Yes | Timestamp when the comment was created. |
| 6 | post\_id |  | Reference | Yes | Associated post |
| 7 | user\_id |  | Reference | Yes | User who wrote the comment |

##### 3.1.6.6 Attachment

| **#** | **Attribute name** | **PK** | **Type** | **Mandatory** | **Description** |
| --- | --- | --- | --- | --- | --- |
| 1 | id | x | uuid | Yes | Primary key that uniquely identifies the attachment. |
| 2 | title |  | String | No | The title of the attachment |
| 3 | description |  | string | Yes | The detail text for the attachment |
| 4 | original\_link |  | string | Yes | Link to the original file (full size, original quality) |
| 5 | created\_at |  | datetime | Yes | Time when the attachment was created. |
| 6 | metadata |  | json | No | Additional information (e.g. size, resolution, duration, format). |
| 7 | optimized\_link |  | string | No | Link to the optimized,compressed version. |
| 8 | delete\_at |  | timestamp | No | Time when the attachment is deleted |

##### 3.1.6.7 Role

| **#** | **Attribute name** | **PK** | **Type** | **Mandatory** | **Description** |
| --- | --- | --- | --- | --- | --- |
| 1 | uid | X | string | Yes | Unique ID of the role |
| 2 | name |  | enum | Yes | Role name (default: ‘GUEST’, ‘USER’, ‘ADMIN’) |

##### 3.1.6.8 Permission

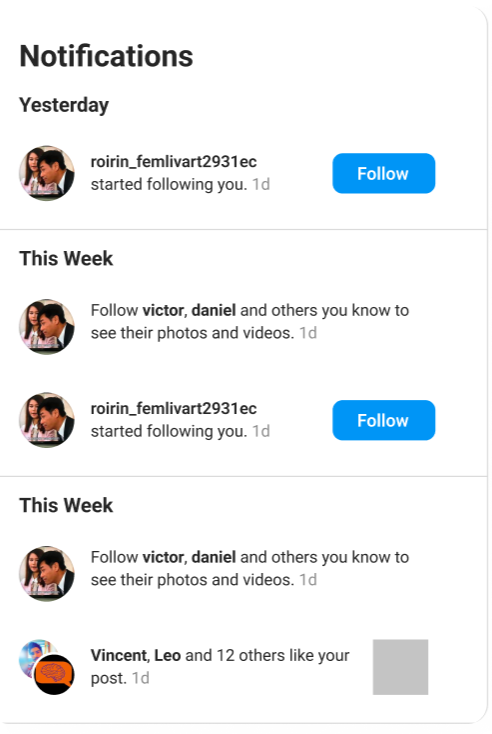
| **#** | **Attribute name** | **PK** | **Type** | **Mandatory** | **Description** |
| --- | --- | --- | --- | --- | --- |
| 1 | uid | X | string | Yes | Unique ID of the permission |
| 2 | name |  | string | Yes | Name of permission |
| 3 | role\_id |  | Reference | Yes | Which role this permission belongs to. |

### 

### 3.2 Notification

#### 3.2.1 Check Recent Notifications

##### 3.2.1.1 Screen Mock-up



**Figure 3-1**: Notification modal

**Table 3-1: Screen Definition**

| **#** | **Field Name** | **Type** | **Mandatory** | **Max Length** | **Description** |
| --- | --- | --- | --- | --- | --- |
| 1 | ctaButtonLabel | String | No | N/A | Caption on the call-to-action button when the notification requires a response (e.g., **“Follow”**, **“Following”**). Hidden for non-interactive notifications. |
| 2 | targetURL | URL | Yes | 2083 | With this field in place, every notification item can seamlessly deep-link to the related post or user profile. |
| 3 | timeAgoLabel | String | Yes | 10 | Relative time stamp (“1 d”, “3 h”). Concise ≤ 5 chars plus a trailing space if displayed inline. |
| 4 | activityMessage | String | Yes | 200 | Full descriptive sentence that follows the actors (e.g., *“started following you.”*, *“… like your post.”*). Includes punctuation. |
| 5 | actorNamesText | String | Yes | 100 | Human-readable list of actors shown in bold (e.g., *“roirin\_femlivart2931ec”* or *“Vincent, Leo and 12 others”*). |
| 6 | sectionHeader | String | Yes | 20 | Group label that partitions the list chronologically (e.g., “Yesterday”, “This Week”). |

##### 3.2.1.2 Use Case Description

| **Use Case ID** | | **EF-02** | **Use Case Name** | | **Check Recent Notifications** | |
| --- | --- | --- | --- | --- | --- | --- |
| **Author** | | **ThangLT** | **Version** | **1.0** | **Date** | **14/06/2025** |
| **Actor** | | Authenticated User | | | | |
| **Description** | | Allows a signed-in user to review the most-recent notifications (likes, comments, follows, system alerts) and have them automatically marked as read once displayed. | | | | |
| **Precondition** | | PRE-01: User is already authenticated and holds a valid JWT / session cookie.  PRE-02: Notification records exist in the database for this user. | | | | |
| **Trigger** | | TRG-01: User clicks the notification bell icon. | | | | |
| **Post-Condition** | | POS-01: A paginated list (latest ≤ 20 items) is presented to the user in reverse-chronological order.  POS-02: Any notifications delivered and displayed in the current view are flagged readAt ≠ null in both PostgreSQL and Dgraph.  POS-03: Header/badge unread counter is updated to reflect remaining unread items, if any. | | | | |
| **Main flows** | | | | | | |
| ***Step*** | ***Actor*** | ***Action*** | | | | |
| *1* | Auth User | Clicks the bell icon. | | | | |
| *2* | Client | Sends GET /api/notifications?limit=20 with JWT. | | | | |
| *3* | Midia | Authorises request, queries Dgraph & PostgreSQL for the latest 20 notifications, sorted by createdAt DESC. | | | | |
| *4* | Midia | Returns JSON payload containing notification list and paging cursor. | | | | |
| 5 | Client | Renders list grouped by time buckets (“Today”, “Yesterday”, “This Week”). | | | | |
| 6 | Client | Fires POST /api/notifications/mark-read with IDs of the items now visible. | | | | |
| 7 | Midia | Updates readAt timestamps; returns new unread count. | | | | |
| 8 | Client | Updates header badge, leaving it at 0 if all items are read. | | | | |

**Alternative flows**

| **AT1** | At step 2 in the main flows**,** if there is an internal error in the system, | |
| --- | --- | --- |
| **Sub step** | **Actor** | **Action** |
| 2.1 | Midia | Fails to fetch notifications due to an unhandled exception. |
| 2.2 | Midia | Responds with HTTP 500 + error code. |
| 2.3 | Client | Displays an error banner: “Unable to load notifications. Please try again later.” |

| **AT2** | At step 3 in the main flows**,** if there are no new notifications. | |
| --- | --- | --- |
| **Sub step** | **Actor** | **Action** |
| 3.1 | Midia | Returns empty list (HTTP 200). |
| 2.2 | Client | Shows placeholder illustration + text: “You’re all caught up!” and sets badge to *0*. |

***Business Rules***

| **#** | **Rule Description** |
| --- | --- |
| BR-01 | Maximum of 20 notifications are fetched per request; subsequent pages use cursor-based pagination. |
| BR-02 | Notifications are considered unread when readAt is null. |
| BR-03 | Returned list must be strictly ordered by createdAt descending before grouping into sections. |
| BR-04 | Notifications older than 30 days are auto-archived and excluded from the “recent” endpoint. |
| BR-05 | The unread badge should never exceed “99+” characters; counts above 99 display as “99+”. |

#### 3.2.2 Check Recent Notifications

##### 3.2.2.1 Use Case Description

| **Use Case ID** | | **EF-03** | **Use Case Name** | | **Check Recent Notifications** | |
| --- | --- | --- | --- | --- | --- | --- |
| **Author** | | **ThangLT** | **Version** |  | **Date** | **14/06/2025** |
| **Actor** | | Authenticated User | | | | |
| **Description** | | The user performs an in-context action from a notification card (e.g., Follow back, Accept request, Open post) and the notification is then marked read. | | | | |
| **Precondition** | | PRE-01: User is already authenticated and holds a valid JWT / session cookie.  PRE-02: Notification records exist in the database and/or Dgraph for this user.  PRE-03: Any context-specific prerequisites are met (e.g., target user’s account is public and not yet followed). | | | | |
| **Trigger** | | TRG-01: User taps either the CTA button (e.g., Follow) or anywhere on the notification row. | | | | |
| **Post-Condition** | | POS-01: Context action completes successfully (follow created, request accepted, post opened, etc.).  POS-02: readAt timestamp is set for that notification in PostgreSQL and Dgraph.  POS-03: Header badge decrements and the notification card shows read styling / updated CTA state. | | | | |
| **Main flows** | | | | | | |
| ***Step*** | ***Actor*** | ***Action*** | | | | |
| *1* | Auth User | Clicks CTA button (e.g., Follow) on a notification. | | | | |
| *2* | Client | Sends API request corresponding to the action:  • POST /api/users/{targetId}/follow or  • Navigates to targetUrl if the row (not CTA) was clicked. | | | | |
| *3* | Midia | Validates JWT, executes business logic (create follow edge, log audit, etc.). | | | | |
| *4* | Midia | Returns 200 OK with updated status payload (followState: FOLLOWING) | | | | |
| 5 | Client | Updates CTA label/icon to “Following” (or navigates to target screen). | | | | |
| 6 | Client | Calls POST /api/notifications/{notifId}/mark-read. | | | | |
| 7 | Midia | Sets readAt, returns new unread count. | | | | |
| 8 | Client | Removes blue dot, updates header badge; flow ends. | | | | |

**Alternative flows**

| **AT1** | At step 3 in the main flows**,** if there is an internal error in the system, | |
| --- | --- | --- |
| **Sub step** | **Actor** | **Action** |
| 3.1 | Midia | Detects unrecoverable error (e.g., DB timeout) and responds HTTP 500. |
| 3.2 | Client | Shows toast: **“**Action failed. Please try again later.**”** and leaves CTA unchanged. |

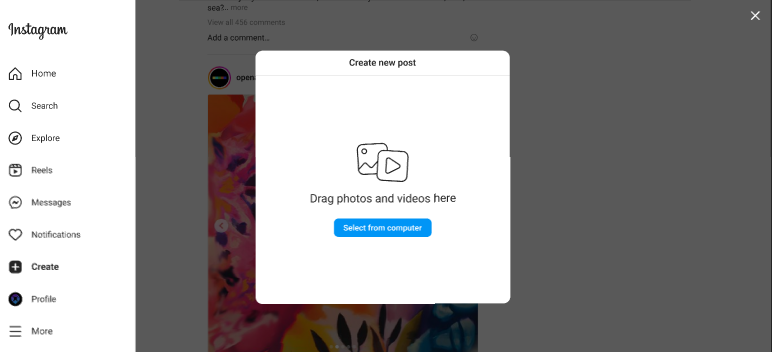
***Business Rules***

| **#** | **Rule Description** |
| --- | --- |
| BR-06 | CTA mapping table determines which API endpoint is invoked per eventType (e.g., FOLLOW\_REQUEST, MENTION, LIKE). |
| BR-07 | A notification is auto-marked read after any successful CTA action or row navigation. |
| BR-08 | Follow/unfollow actions are rate-limited to 30 per hour per user to mitigate spam. |
| BR-09 | If the action has already been performed (e.g., user is already following), the service returns 409 Conflict; client shows inline warning without altering unread count. |

### 3.3 Content Management

#### 3.3.1 Create Post

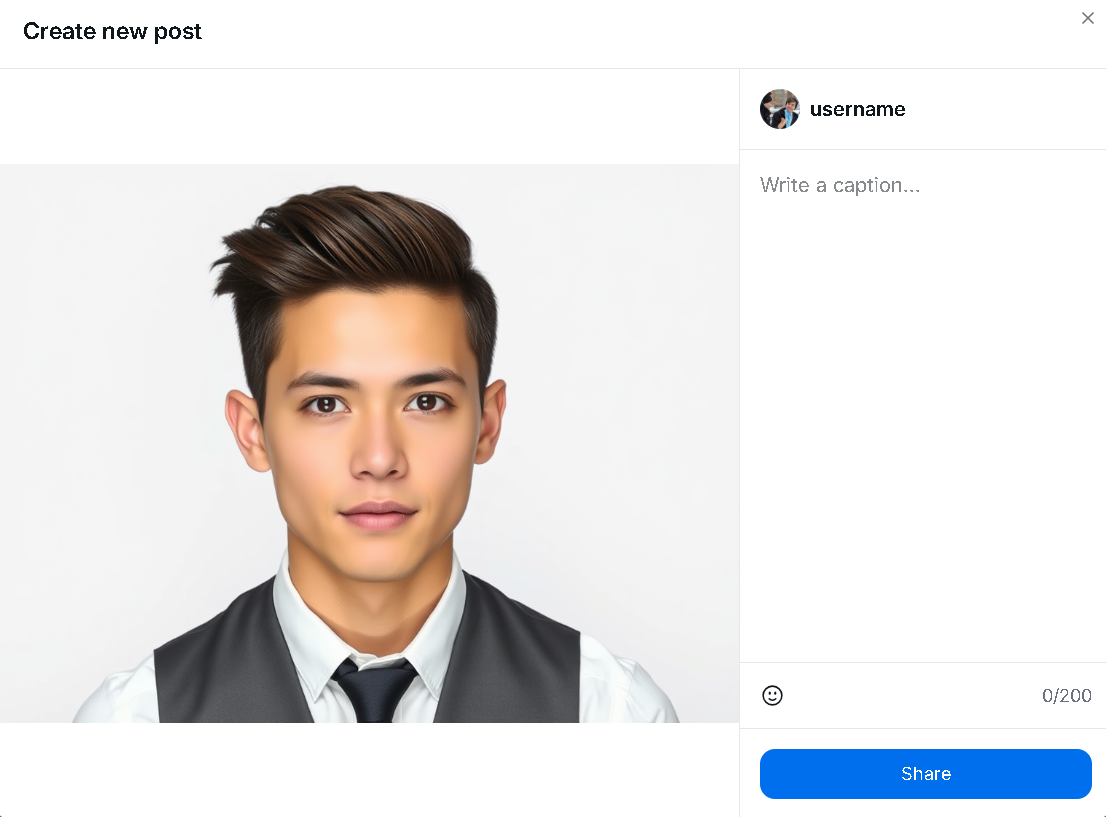
##### 3.3.1.1 Screen Mock-up

****

**Figure 3.3-1**: Screen Design of Create Post

**Table 3.3-1: Screen Definition**

| **#** | **Field Name** | **Type** | **Mandatory** | **Max Length** | **Description** |
| --- | --- | --- | --- | --- | --- |
| 1 | Select from Computer | Button | Yes | N/A | Import image or video from system |

******

**Figure 3.2-2**: Screen Design of post sharing

| **#** | **Field Name** | **Type** | **Mandatory** | **Max Length** | **Description** |
| --- | --- | --- | --- | --- | --- |
| 1 | Image | image | Yes | N/A | Display image of post. Can click to image to tag other user |
| 2 | Write a caption | text area | Yes | 255 | Write the caption of the post |
| 3 | Emoji icon | button | No | N/A | Choose emoji for the caption for more beautiful caption |
| 4 | Share | button | Yes | N/A | Submit to create a new post |

**3.3.1.2 Use Case Description**

| **Use Case ID** | | **CM-01** | **Use Case Name** | | **Create Post** | |
| --- | --- | --- | --- | --- | --- | --- |
| **Author** | | **SonDH** | **Version** | **1.0** | **Date** | **15/06/2025** |
| **Actor** | | User | | | | |
| **Description** | | This function allows an authenticated user to create and share a new post, which includes a media file (image) and a text caption. The new post will be visible on their profile and in the feeds of their followers. | | | | |
| **Precondition** | | PRE-01: The User must be successfully logged into the Midia system.  PRE-02: The User's account must be active (i.e., not banned or suspended). | | | | |
| **Trigger** | | TRG-01: The User clicks the "Create Post" button or navigates to the post creation screen within the application. | | | | |
| **Post-Condition** | | POS-01: (Success) A new post record is created, stored in the database, and associated with the User's account.  POS-02: (Success) The new post is added to the User's profile feed and distributed to the feeds of their followers.  POS-03: (Failure) The system state remains unchanged if the process is cancelled or fails. | | | | |
| **Main flows** | | | | | | |
| ***Step*** | ***Actor*** | ***Action*** | | | | |
| *1* | User | Selects the option to create a new post. | | | | |
| *2* | Midia | Displays the post creation form. | | | | |
| *3* | User | Uploads a media file (image). | | | | |
| *4* | User | Enters a text caption for the post. | | | | |
| *5* | User | Clicks the "Share" button to submit the post. | | | | |
| *6* | Midia | Validates the submitted media and text against business rules (BR-01, BR-02, etc.). | | | | |
| *7* | Midia | Saves the post to the database, linking it to the User's profile. | | | | |
| *8* | Midia | Confirms the successful creation by displaying a success message (e.g., "Post shared successfully"). | | | | |
| 9 | Midia | Navigates the User back to their main feed or profile page, where the new post is now visible. | | | | |

**Alternative flows**

| **AT1** | At any step before step 5 in the main flow, if the user decides to cancel: | |
| --- | --- | --- |
| **Sub step** | **Actor** | **Action** |
| 1.1 | User | Clicks the "Cancel" or "Back" button. |
| 1.2 | Midia | Discards the form data and returns the user to the previous screen. The use case ends. |

| **AT2** | At step 6 in the main flow, if the uploaded file does not meet the required criteria (e.g., wrong format, too large) | |
| --- | --- | --- |
| **Sub step** | **Actor** | **Action** |
| 6.1 | Midia | Rejects the submission and does not save the post. |
| 6.2 | Midia | Displays an informative error message on the creation form (e.g., "Invalid file format. Please use JPG, PNG"). The flow resumes at step 3. |

| **AT3** | At step 6 in the main flow, if the caption fails validation (e.g., exceeds character limit) | |
| --- | --- | --- |
| **Sub step** | **Actor** | **Action** |
| 6.1 | Midia | Rejects the submission and does not save the post. |
| 6.2 | Midia | Displays an informative error message on the creation form (e.g., "Caption cannot exceed 2,200 characters."). The flow resumes at step 4. |

| **AT4** | At step 7 in the main flow, if there is an internal error in the system (e.g., database connection failure) | |
| --- | --- | --- |
| **Sub step** | **Actor** | **Action** |
| 6.1 | Midia | Displays an "Error" page with the message "Internal System Error. Your post could not be shared. Please try again later." The use case ends. |

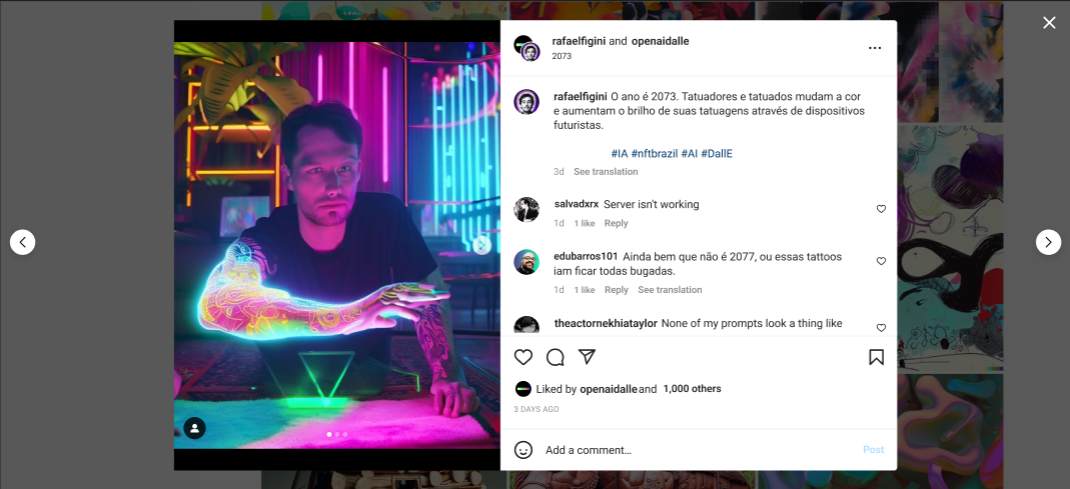
***Business Rules***

| **#** | **Rule Description** |
| --- | --- |
| BR-10 | Media Required: A post must contain exactly one media file (image or video). |
| BR-11 | Supported File Formats: The system shall only accept media files with the following extensions: .jpg, .jpeg, .png for images, and .mp4 for videos. |
| BR-12 | Maximum File Size: The size of the uploaded media file must not exceed 10MB for images and 100MB for videos. |
| BR-13- | Caption Length: The text caption is optional, but if provided, it must not exceed 2,200 characters. |

### 

#### 3.3.2 View Post Details

##### 3.3.2.1 Screen Mock-up



**Figure 3.3-2**: Screen Design of Post Details

**Table 3.3-2: Screen Definition**

| **#** | **Field Name** | **Type** | **Mandatory** | **Max Length** | **Description** |
| --- | --- | --- | --- | --- | --- |
| 1 | Image | Image | Yes |  | Display images of a post |
| 2 | Next Image | Button | Yes |  | Allow users to move to next image of this post if there are more than 2 images |
| 3 | Previous Image | Button | Yes |  | Allow users to move to previous image of this post if there are more than 2 images |
| 4 | Avatar | Image | Yes |  | Display users’ avatars |
| 5 | Username | Label | Yes |  | Display users’ username |
| 6 | Full name | Label | Yes |  | Display the current user’s full name |
| 7 | Post time | Label | Yes |  | Display how long this post was created |
| 8 | Post content | Label | Yes |  | Display the description of image that users import in a post |
| 9 | Comment field | Label | No |  | Display the comments of other users |
| 10 | Comments time | Label | Yes |  | Display how long this comment was posted |
| 11 | Comments like | Icon | Yes |  | Display how many likes that this comment has |
| 12 | Reply | Link | Yes |  | Allow users to answer someone’s comment |
| 13 | Like icon | Icon | Yes |  | Allow users to like comments |
| 14 | Comment icon | Icon | Yes |  | Allow users to create a comment |
| 15 | Share icon | Icon | Yes |  | Allow users to share this post |
| 16 | Number of liked | Label | Yes |  | Display the number of liked that users’ post have and their friends’ username |
| 17 | Feeling icon | Icon | Yes |  | Allow users to add some icons in their comment |
| 18 | Add a comment | Text | Yes |  | Allow users to type their comment |
| 19 | Post | Button | Yes |  | Allow users to post their comment |

**3.3.2.2 Use Case Description**

| **Use Case ID** | | **CM-02** | **Use Case Name** | | **Post Details** | |
| --- | --- | --- | --- | --- | --- | --- |
| **Author** | | **ThangLT** | **Version** | **1.0** | **Date** | **15/06/2025** |
| **Actor** | | Registered User | | | | |
| **Description** | | This function allows a user to view a single post in a detailed, focused view. It displays the post's media, caption, author, and all associated social interactions (likes, comments). From this view, the user can perform further actions on the post, such as liking, commenting, or saving it. | | | | |
| **Precondition** | | PRE-01: The User must be successfully logged into the Midia system.  PRE-02: The post being viewed must exist and be accessible to the User (i.e., not from a private account the User doesn't follow). | | | | |
| **Trigger** | | TRG-01: The User clicks on a post thumbnail from any feed (e.g., home feed, profile grid, search results). | | | | |
| **Post-Condition** | | POS-01: (Success) The system displays the detailed post view to the User. The system state remains unchanged until a subsequent action is taken.  POS-02: (Failure) An error message is displayed if the post cannot be loaded, and the User remains on the previous screen. | | | | |
| **Main flows** | | | | | | |
| ***Step*** | ***Actor*** | ***Action*** | | | | |
| *1* | User | Clicks on a post from a feed. | | | | |
| *2* | Midia | Retrieves all data associated with the selected post (media, caption, author info, likes, comments). | | | | |
| *3* | Midia | Displays the detailed post view, populated with:The post's image or video. The author's username and profile picture.The post's caption.The total like count.A list of existing comments.Action buttons (Like, Comment, Share, Save). | | | | |

**Alternative flows**

| **AT1** | At step 3 of the main flow: | |
| --- | --- | --- |
| **Sub step** | **Actor** | **Action** |
| 3.1 | User | Clicks the "Like" (heart) button. |
| 1.2 | Midia | Toggles the like status for the user on the post, updates the like count, and provides visual feedback (e.g., fills in the heart icon). |

| **AT2** | At step 3 of the main flow: | |
| --- | --- | --- |
| **Sub step** | **Actor** | **Action** |
| 3.1 | User | Enters text into the "Add a comment..." field and clicks "Post". |
| 3.2 | Midia | Submits and saves the new comment, then updates the comment list in the UI. |

| **AT3** | At step 3 of the main flow: | |
| --- | --- | --- |
| **Sub step** | **Actor** | **Action** |
| 3.1 | User | Clicks the "Save" (bookmark) button |
| 3.2 | Midia | Toggles the saved status of the post for the user and provides visual feedback. |

| **AT4** | At step 3 of the main flow, if the current User is the author of the post: | |
| --- | --- | --- |
| **Sub step** | **Actor** | **Action** |
| 3.1 | User | Clicks the "Options" (three dots) menu and selects "Edit" or "Delete". |
| 3.2 | Midia | Initiates the "Edit Post" or "Delete Post" use case. |

| **AT5** | At any point after step 3 of the main flow: | |
| --- | --- | --- |
| **Sub step** | **Actor** | **Action** |
| 3.1 | User | Clicks the "Close" ('X') button or presses the 'Escape' key. |
| 3.2 | Midia | Closes the detailed post view and returns the User to the previous feed screen. |

| **AT6** | At step 2 of the main flow, if the system cannot retrieve the post data: | |
| --- | --- | --- |
| **Sub step** | **Actor** | **Action** |
| 2.1 | User | Displays an error message (e.g., "This post is unavailable."). The use case ends. |

***Business Rules***

| **#** | **Rule Description** |
| --- | --- |
| BR-55 | **Author-Specific Options:** The "Edit" and "Delete" options for a post must only be visible and accessible to the user who authored the post. |
| BR-56 | **Like State:** A user can only like a post once. The system must toggle the like status (liked/unliked) on subsequent clicks. |
| BR-57 | **Comment Display Order:** Comments shall be displayed in chronological order, with the oldest comments appearing first. |

### 

### 3.4 User Settings

#### **3.4.1 User Settings**

##### 3.4.1.1 Screen Mock-up

##### 

**Figure 3-4**: Screen Design of User Settings

**Table 3-4: Screen Definition**

| **#** | **Field Name** | **Type** | **Mandatory** | **Max Length** | **Description** |
| --- | --- | --- | --- | --- | --- |
| 1 | Name | Text | Yes |  | Display real name of user who has logged in |
| 2 | Username | Text | Yes |  | Display username of account which has logged in |
| 3 | Website | Text | No |  | Display users’ website link. This field can be edited if users are using a mobile app and optional |
| 4 | Bio | Text | No |  | Display short biography of users. Users can only type about 150 characters. This field is optional |
| 5 | Email | Text | Yes |  | Display user’s email. Email must follow the format <name>@<domain>. This field cannot be edited |
| 6 | Phone number | Text | Yes | 11 | Display user’s phone number. Phone number must contain 11 digits. |
| 7 | Gender | Dropdown box | Yes |  | Display the user's gender. This field contains Male, Female and Prefer not to say |
| 8 | Show account suggestions | Checkbox | No |  | Allow users to see similar account suggestions on their profiles |
| 9 | Learn more | Link | Yes |  | Display the link that navigates to the information about changing username policies |
| 10 | Change profile photo | Link | Yes |  | Display the link that navigates to change profile photo screen |
| 11 | Submit | Button | Yes |  | Allow to save all changes |
| 12 | Temporarily deactivate my account | Link | Yes |  | Allow users to deactivate their account if they want |
| 13 | Edit profile | List item | Yes |  | Allow users to open the edit profile box. They can change their information stored in the system |

**3.4.1.2 Use Case Description**

| **Use Case ID** | | **US-01** | **Use Case Name** | | **User Settings** | |
| --- | --- | --- | --- | --- | --- | --- |
| **Author** | | **HoanTX** | **Version** | **1.0** | **Date** | **13/06/2025** |
| **Actor** | | Registered User | | | | |
| **Description** | | The function allows users to edit their basic and personal information in the system. | | | | |
| **Precondition** | | PRE-01: Users have unbanned accounts  PRE-02: Users have a stable Internet connection  PRE-03: Users access the website with the correct URL  PRE-04: Users have signed in successfully | | | | |
| **Trigger** | | TRG-01: Users click the Setting icon in the header of website | | | | |
| **Post-Condition** | | POS-01: Basic information and Personal information are displayed  POS-02: User’s avatar and information are displayed  POS-03: Input fields, buttons and navigation links are ready to interact | | | | |
| **Main flows** | | | | | | |
| ***Step*** | ***Actor*** | ***Action*** | | | | |
| *1* | User | Type URL into location field of internet browser and then press enter | | | | |
| *2* | Midia | Display Home page with gear icon in the right side of header | | | | |
| *3* | User | Click the gear icon | | | | |
| *4* | Midia | Navigate to User Setting page | | | | |

**Alternative flows**

| **AT1** | Users click the “Learn more” link in the Username field’s description | |
| --- | --- | --- |
| **Sub step** | **Actor** | **Action** |
| 1.1 | Midia | Navigate to System’s policies screen |
|  | | |
| **AT2** | Users click the “Temporarily deactivate my account” link in the right of “Submit” button | |
| **Sub step** | **Actor** | **Action** |
| 2.1 | Midia | Display the confirmation popup screen in order to make sure that users really want to deactivate their account |
| 2.2 | User | Click “Ok” or “Cancel” button |
| 2.3 | Midia | **AT2.1 Users clicks the Ok button**   * Navigate to Login screen, remove all cookie and session using in the system   **AT2.2 Users clicks the Cancel button  -** Exit the popup screen |

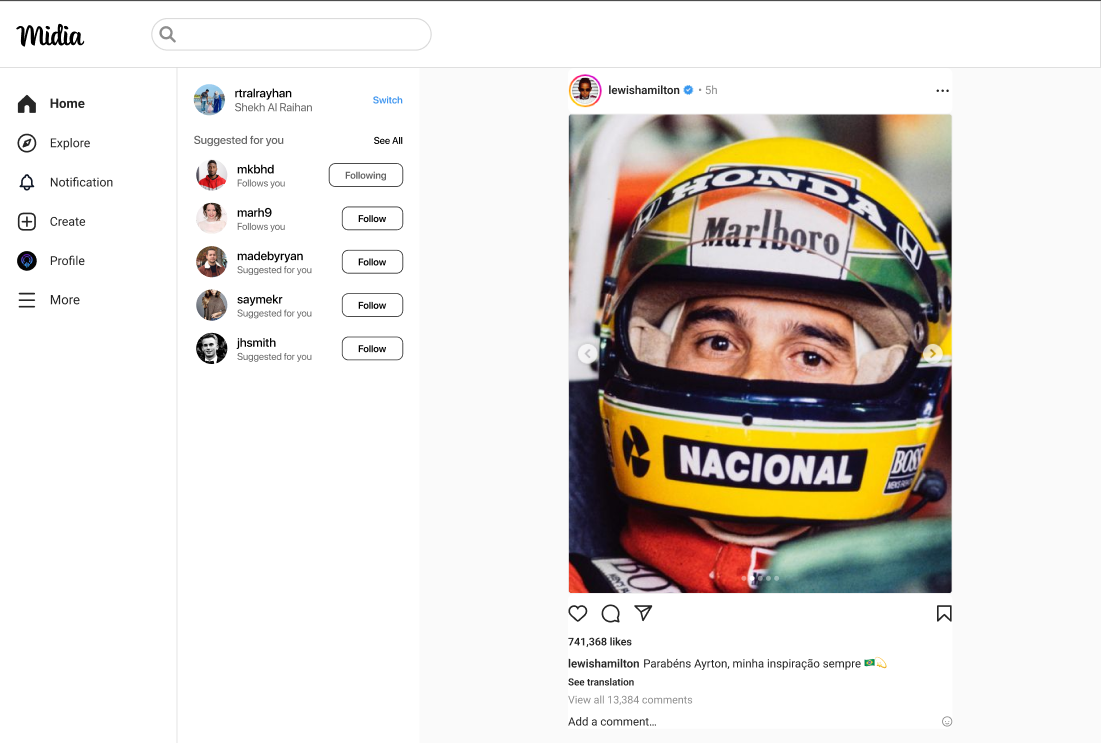
***Business Rules***

| **#** | **Rule Description** |
| --- | --- |
| BR-14 | Phone number must have 11 digits and is existed |
| BR-15 | Username must be unique |

### 3.5 Follow System

#### **3.5.1 Follow Suggestions**

##### 3.5.1.1 Screen Mock-up



**Figure 3-5**: Screen Design of Follow Suggestions

**Table 3-4: Screen Definition**

| **#** | **Field Name** | **Type** | **Mandatory** | **Max Length** | **Description** |
| --- | --- | --- | --- | --- | --- |
| 1 | Username | Label | Yes |  | Display usernames of accounts. |
| 2 | Full name | Label | Yes |  | Display the full name of user who is using the website |
| 3 | Avatar | Image | Yes |  | Display accounts’ avatars |
| 4 | Follows you/Suggested for you | Label | Yes |  | Displays a label next to another user's name in lists or search results.  **Follows you**: Indicates that the user is currently following you. Helps you quickly identify mutual connections.  **Suggested for you**: Shown for users that Instagram recommends based on your behavior, interests, or mutual connections. These users are not necessarily following you. |
| 5 | Switch | Link | Yes |  | Allow users to use another saved account or login with unsaved account |
| 6 | See All | Link | Yes |  | Navigate to screen that list all accounts that users can follow |
| 7 | Following/Follow | Button | Yes |  | Allow users to start follow or cancel follow other accounts  **Follow:** Displayed when the user *has not followed* the account. Clicking it sends the following request.  **Following**: Displayed when the user *is currently following* the account. Clicking it may open an "Unfollow" option or directly trigger the unfollow action (depending on the design). |

**3.5.1.2 Use Case Description**

| **Use Case ID** | | **FS-01** | **Use Case Name** | | **Follow Suggestions** | |
| --- | --- | --- | --- | --- | --- | --- |
| **Author** | | **HoanTX** | **Version** | **1.0** | **Date** | **15/06/2025** |
| **Actor** | | Registered User | | | | |
| **Description** | | Displays a list of accounts suggested by the system for the user to follow, based on interests, mutual connections, previous interactions, or for newly created accounts. | | | | |
| **Precondition** | | PRE-01: Users have unbanned accounts  PRE-02: Users have a stable Internet connection  PRE-03: Users access the website with the correct URL  PRE-04: Users have signed in successfully | | | | |
| **Trigger** | | TRG-01: Users access the website | | | | |
| **Post-Condition** | | POS-01: Users avatar, username and fullname are displayed  POS-02: List of 5 accounts with labels, buttons are rendered and ready to interact | | | | |
| **Main flows** | | | | | | |
| ***Step*** | ***Actor*** | ***Action*** | | | | |
| *1* | User | Type URL into location field of internet browser and then press enter | | | | |
| *2* | Midia | Display Home page with suggestions in the right of left sidebar | | | | |

**Alternative flows**

| **AT1** | Users click the “Switch” link | |
| --- | --- | --- |
| **Sub step** | **Actor** | **Action** |
| 1.1 | Midia | Navigate to Login screen |
|  | | |
| **AT2** | Users click the “See all” link | |
| **Sub step** | **Actor** | **Action** |
| 2.1 | Midia | Navigate to explore users screen |
|  | | |
| **AT3** | Users click the Follow/Following button | |
| 3.1 | Midia | **AT3.1 Users click the “Follow” button**   * Change button from “Follow” to “Following” * Change label from “Suggested for you” to “Follows you” * Add this account to the followings list of current account   **AT3.2 Users click the “Following” button**   * Change button from “Following” to “Follow” * Change label from “Follows you” to “Suggested for you” * Remove this account from the followings list of current account |

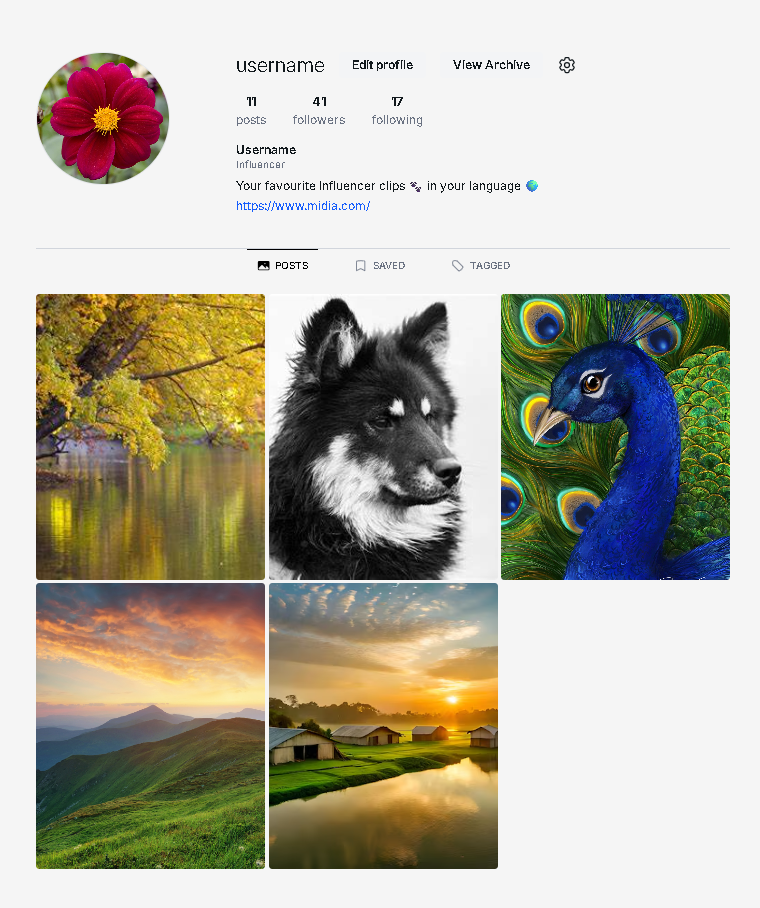
***Business Rules***

| **#** | **Rule Description** |
| --- | --- |
| BR-16 | List of suggestions must be limited in 5 accounts |

### 3.6 Profile Management

#### 3.6.1 Profile page

##### 3.6.1.1 Screen Mock-up



**Figure 3-5**: Screen Design of Follow Suggestions

**Table 3-6: Screen Definition**

| **#** | **Field Name** | **Type** | **Mandatory** | **Max Length** | **Description** |
| --- | --- | --- | --- | --- | --- |
| 1 | Username | Label | Yes | N/A | Display usernames of accounts. |
| 2 | Edit profile | Button | No | N/A | Navigate to edit profile page |
| 3 | Avatar | Image | Yes | N/A | Display accounts’ avatars |
| 4 | Number of posts | Text | Yes | 10 | Display number of posts of that user |
| 5 | Number of followers | Text | Yes | 10 | Display number of followers of that user |
| 6 | Number of following | Text | Yes | 10 | Display number of following user of that user |
| 7 | Description | Text | No | 100 | Display the description of that user profile |
| 8 | Link to other page | Text | No | 100 | Display the link of other profile page |
| 9 | Post navigation | Link | No | N/A | Display all the post of that user |
| 10 | Saved navigation | Link | No | N/A | Display all the saved post of that user |
| 11 | Tagged navigation | Link | No | N/A | Display all the post of that user that been tagged by others |
| 12 | Image Posts | Image | Yes | N/A | Display the image of the posts. When hover, show the number of comment and like. When click, show modal of image, comment and number of like. User can interact inside that modal |
| 13 | Setting icon | Button | No | N/A | Navigate user to User Setting screen page |

**3.6.1.2 Use Case Description**

| **Use Case ID** | | **PM-01** | **Use Case Name** | | **Profile page** | |
| --- | --- | --- | --- | --- | --- | --- |
| **Author** | | **SonDH** | **Version** | **1.0** | **Date** | **15/06/2025** |
| **Actor** | | User | | | | |
| **Description** | | Displays information of user profile, include username, description, … User can edit profile, show comment,number of comment,number of like , followers, following, posts and number of posts. | | | | |
| **Precondition** | | PRE-01: User is authenticated.  PRE-02: The target profile exists. | | | | |
| **Trigger** | | TRG-01: User clicks on their profile picture or username link. | | | | |
| **Post-Condition** | | POS-01: Profile information is displayed on screen.  POS-02: If updated, profile data is saved and reflected in UI. | | | | |
| **Main flows** | | | | | | |
| ***Step*** | ***Actor*** | ***Action*** | | | | |
| *1* | User | Navigates to the profile section from header or post. | | | | |
| *2* | Client | Send request to backend to fetch user profile data | | | | |
| *3* | System | Retrieves user profile, including username, bio, stats (posts, likes, comments), followers/following count, and recent posts. | | | | |
| *4* | System | Send the response back to the client. | | | | |
| *5* | Client | Renders the profile UI with all data. | | | | |
| *6* | User | User can click “Edit profile” to edit profile, click Setting icon to change setting, or click post to show modal contain image post, comment and stats | | | | |

**Alternative flows**

| **AT1** | At the step 1, user click into user profile that been deleted or change status to “LOCK” | |
| --- | --- | --- |
| **Sub step** | **Actor** | **Action** |
| 1.1 | Client | Show status 404 message “User not found” |

***Business Rules***

| **#** | **Rule Description** |
| --- | --- |
| BR-17 | Phone number must have 11 digits and is existed |
| BR-18 | Username must be unique |

#### **3.6.2 UP02\_Edit Profile**

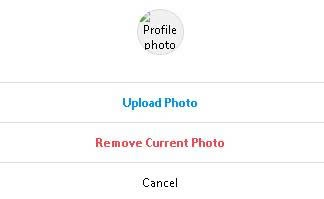
##### 3.6.2.1 Screen Mock-up

##### 

**Figure 3-6.2.1**: Screen Design of Edit Profile

**Table 3-6.2.1: Screen Definition**

| **#** | **Field Name** | **Type** | **Mandatory** | **Max Length** | **Description** |
| --- | --- | --- | --- | --- | --- |
| 1 | Avatar | Image | Yes |  | Display accounts’ avatars |
| 2 | Username | Text | Yes |  | Display username of account which has logged in |
| 3 | Full name | Label | Yes |  | Display the full name of user who is using the website |
| 4 | Change photo | Button |  |  | Allow users to replace the old avatar in the popup screen |
| 5 | Website | Text | No |  | Display users’ website link. This field can be edited if users are using a mobile app and optional |
| 6 | Bio | Text | No |  | Display short biography of users. Users can only type about 150 characters. This field is optional |
| 7 | Gender | Dropdown box | Yes |  | Display the user's gender. This field contains Male, Female and Prefer not to say |
| 8 | Show account suggestions | Switch | No |  | Allow users to see similar account suggestions on their profiles |
| 9 | Submit | Button | Yes |  | Allow to save all changes |



**Figure 3-6.2.2**: Screen Design of Change photo popup screen

**Table 3-6.2.2: Screen Definition**

| **#** | **Field Name** | **Type** | **Mandatory** | **Max Length** | **Description** |
| --- | --- | --- | --- | --- | --- |
| 1 | Profile photo | Image | Yes |  | Display accounts’ avatars |
| 2 | Upload photo | Button | Yes |  | Allow users to upload a new photo |
| 3 | Remove current photo | Button | Yes |  | Allow users to remove the current avatar |
| 4 | Cancel | Button | Yes |  | Allow users to cancel all changes and go back to previous screen |

**3.6.2.2 Use Case Description**

| **Use Case ID** | | **PM-02** | **Use Case Name** | | **Edit Profile** | |
| --- | --- | --- | --- | --- | --- | --- |
| **Author** | | **HoanTX** | **Version** | **1.0** | **Date** | **15/06/2025** |
| **Actor** | | Registered User | | | | |
| **Description** | | This screen allows the user to update personal profile information, including website link, bio, gender, and account suggestion visibility. Users can also change their profile picture and view their username. | | | | |
| **Precondition** | | PRE-01: Users have unbanned accounts  PRE-02: Users have a stable Internet connection  PRE-03: Users access the website with the correct URL  PRE-04: Users have signed in successfully | | | | |
| **Trigger** | | TRG-01: Users click the “Edit profile” in the Profile screen | | | | |
| **Post-Condition** | | POS-01: User’s avatar and information are displayed  POS-02: Input fields, buttons and navigation links are ready to interact | | | | |
| **Main flows** | | | | | | |
| ***Step*** | ***Actor*** | ***Action*** | | | | |
| *1* | User | Type URL into location field of internet browser and then press enter | | | | |
| *2* | Midia | Display Home page | | | | |
| *3* | User | Click the Profile option in the left sidebar | | | | |
| *4* | Midia | Display the Profile screen with “Edit profile” button in the header | | | | |
| *5* | User | Click the “Edit profile” button | | | | |
| *6* | Midia | Navigate to “Edit profile” screen | | | | |

**Alternative flows**

| **AT1** | Users click the “Change photo” button | |
| --- | --- | --- |
| **Sub step** | **Actor** | **Action** |
| 1.1 | Midia | Open the change photo popup screen |
| 1.2 | User | **AT1.1 Users click the “Upload Photo” button**   * Choose an image * Click Ok button   **AT1.2 Users click the “Remove Current Photo” button**   * Confirm removing in the alert popup screen   **AT1.3 Users click the “Cancel” button** |
| 1.3 | Midia | **AT1.1 Users click the “Upload Photo” button**   * Change user’s avatar   **AT1.2 Users click the “Remove Current Photo” button**   * If user presses “OK” button, then remove his/her avatar * If user presses “Cancel” button, then close the alert modal   **AT1.3 Users click the “Cancel” button**   * Close the change photo popup screen |

***Business Rules***

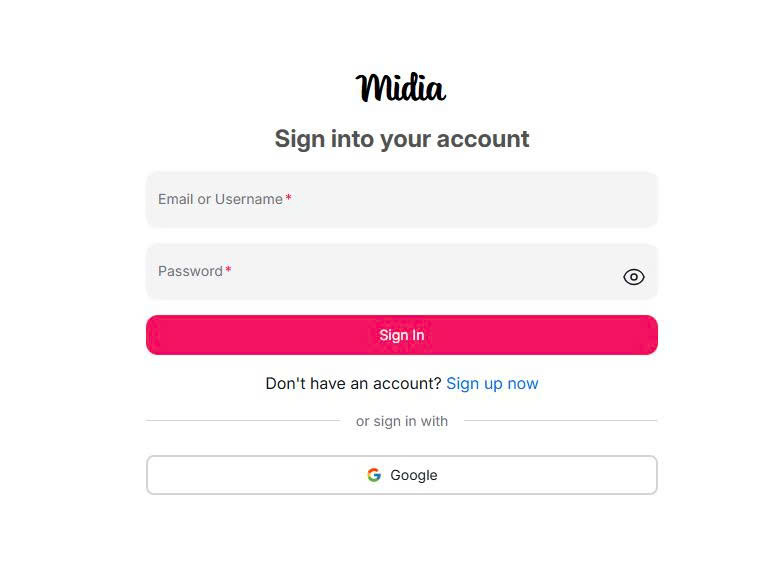
| **#** | **Rule Description** |
| --- | --- |
| BR-19 | Image file must be maximum 5MB |
| BR-20 | Image’s format should be JPG, PNG or WEBP |

### 

### 3.7 Account Management

#### **3.7.1 Login**

##### 3.7.1.1 Screen Mock-up



**Figure 3-7.1**: Screen Design of Follow Suggestions

**Table 3-7.1: Screen Definition**

| **#** | **Field Name** | **Type** | **Mandatory** | **Max Length** | **Description** |
| --- | --- | --- | --- | --- | --- |
| 1 | Username | Label | Yes | N/A | Display usernames of accounts. |
| 2 | Password | Label | Yes | N/A | Display password of accounts. |
| 3 | Sign In | Button | No | N/A |  |
| 4 | Sign up now | Link | No | N/A | Navigate to sign up page. |
| 5 | Google | Button | No | N/A | Sign in via google. |

**3.7.1.2 Use Case Description**

| **Use Case ID** | | **AM-01** | **Use Case Name** | | **Login** | |
| --- | --- | --- | --- | --- | --- | --- |
| **Author** | | **HoangMH** | **Version** | **1.0** | **Date** | **15/06/2025** |
| **Actor** | | Guest | | | | |
| **Description** | | This use case allows guests to log in to the system using a system-registered account (username and password). Optionally, users may log in via Google authentication. | | | | |
| **Precondition** | | PRE-01: The user has registered an account OR has a valid Google account.  PRE-02: The user is currently not logged in | | | | |
| **Trigger** | | TRG-01: User visits the website and chooses to log in | | | | |
| **Post-Condition** | | POS-01: User is authenticated and redirected to the home page.  POS-02: Authentication token or session is created. | | | | |
| **Main flows** | | | | | | |
| ***Step*** | ***Actor*** | ***Action*** | | | | |
| *1* | User | Clicks on the "Login" button on the homepage | | | | |
| *2* | Client | Navigate user to login screen with options:   * Username and password form * “Login with Google” button | | | | |
| *3* | User | Enters username and password and clicks “Login” | | | | |
| *4* | System | Validates the credentials with user database | | | | |
| *5* | System | If valid, logs in user and creates session or JWT | | | | |
| *6* | Client | Redirects user to the home page with authenticated session | | | | |

**Alternative flows**

| **AT1** | At the step 3, user choose “Login with Google” instead of enters username and password | |
| --- | --- | --- |
| **Sub step** | **Actor** | **Action** |
| 1.1 | User | Click “Login with Google” |
| 1.2 | Client | Redirect to Google OAuth login page |
| 1.3 | Google | Authenticates user and returns token |
| 1.4 | System | Verifies token and logs user in (creates user if first time) |
| 1.5 | Client | Navigate to home page |

| **AT2** | At the step 3, user enter wrong username/password | |
| --- | --- | --- |
| **Sub step** | **Actor** | **Action** |
| 2.1 | System | Rejects login due to incorrect credentials |
| 2.2 | Client | Displays error message: “Invalid username or password” |

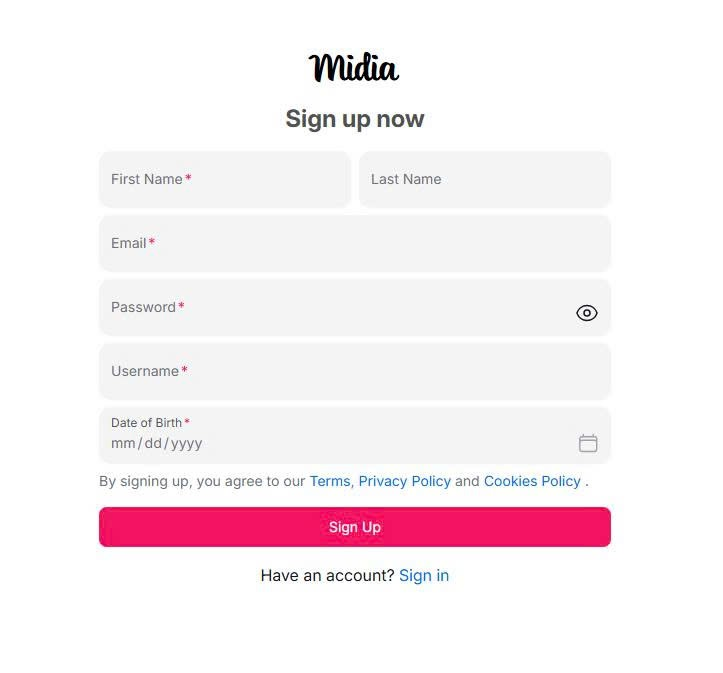
| **AT2** | At the step 4, the system find that user’s account status is lock or banned | |
| --- | --- | --- |
| **Sub step** | **Actor** | **Action** |
| 3.1 | System | Detects user status is LOCKED/BANNED |
| 3.2 | Client | Show message: “Your account has been locked” |

***Business Rules***

| **#** | **Rule Description** |
| --- | --- |
| BR-21 | Username must be unique and password must be hashed securely. |
| BR-22 | Only active users (not locked or banned) can log in. |
| BR-23 | Google login is optional and only available if the email is verified. |

#### **3.7.2 Register**

##### 3.7.2.1 Screen Mock-up



**Figure 3-7.2**: Screen Design of Register

**Table 3-7.2: Screen Definition**

| **#** | **Field Name** | **Type** | **Mandatory** | **Max Length** | **Description** |
| --- | --- | --- | --- | --- | --- |
| 1 | First Name | Input | Yes | 30 | Enter user’s first name |
| 2 | Last Name | Input | No | 30 | Enter user’s last name |
| 3 | Email | Input | Yes | 30 | Enter user’s email |
| 4 | Password | Input | Yes | 30 | Enter user’s password |
| 5 | Username | Input | Yes | 20 | Enter user’s username |
| 6 | Date of birth | Date | Yes | N/A | Choose user’s date of birth in calendar |
| 7 | Sign up | Button | Yes | N/A | Submit button to create a new account |
| 8 | Sign in | Button | No | N/A | Navigate user to login page |

**3.7.2.2 Use Case Description**

| **Use Case ID** | | **AM-02** | **Use Case Name** | | **Register** | |
| --- | --- | --- | --- | --- | --- | --- |
| **Author** | | **HoangMH** | **Version** | **1.0** | **Date** | **15/06/2025** |
| **Actor** | | Guest | | | | |
| **Description** | | This use case allows a guest user to create a new account by filling in required personal details. | | | | |
| **Precondition** | | PRE-01: User is not authenticated. | | | | |
| **Trigger** | | TRG-01: User clicks on “Sign up” from the login screen. | | | | |
| **Post-Condition** | | POS-01: A new account is created and verified (optionally).  POS-02: User is redirected to home or login page. | | | | |
| **Main flows** | | | | | | |
| ***Step*** | ***Actor*** | ***Action*** | | | | |
| *1* | User | Clicks on the "Sign up" button on the login screen | | | | |
| *2* | Client | Navigate user to sign up screen with fields:   * Full name * Username * Email * Password * Phone (optional) | | | | |
| *3* | User | Fills in registration form and submits | | | | |
| *4* | System | Validates data (e.g., username uniqueness, email format) | | | | |
| *5* | System | Creates a new user account, sets default profile settings | | | | |
| *6* | Client | Redirects to login screen | | | | |

**Alternative flows**

| **AT1** | At the step 4, the system found email that user enters is existed in database | |
| --- | --- | --- |
| **Sub step** | **Actor** | **Action** |
| 1.1 | System | Detects existing email or username |
| 1.2 | Client | Shows error message: “Email already in use” |

***Business Rules***

| **#** | **Rule Description** |
| --- | --- |
| BR-24 | Username and email must be unique |
| BR-25 | Password must be at least 8 characters |

#### **3.7.3 Forgot password**

##### 3.7.3.1 Screen Mock-up

**Figure 3-7.2**: Screen Design of forgot password

**Table 3-7.2: Screen Definition**

**3.7.2.2 Use Case Description**

| **Use Case ID** | | **AM-03** | **Use Case Name** | | **Forgot Password** | |
| --- | --- | --- | --- | --- | --- | --- |
| **Author** | | **HoangMH** | **Version** | **1.0** | **Date** | **15/06/2025** |
| **Actor** | | Guest | | | | |
| **Description** | | This use case allows a user to recover their password by submitting their email. A reset link will be sent if the email is valid. | | | | |
| **Precondition** | | PRE-01: The user has an existing account with a verified email. | | | | |
| **Trigger** | | TRG-01: User clicks “Forgot password?” on the login screen. | | | | |
| **Post-Condition** | | POS-01: A reset email is sent to the user’s inbox. | | | | |
| **Main flows** | | | | | | |
| ***Step*** | ***Actor*** | ***Action*** | | | | |
| *1* | User | Clicks on the "Forgot password?" button on the login screen | | | | |
| *2* | Client | Displays form asking for the user’s registered email | | | | |
| *3* | User | Enters email and submits | | | | |
| *4* | System | Checks if email exists and is verified | | | | |
| *5* | System | Sends password reset email with a secure token | | | | |
| *6* | Client | Display message “Reset link sent to your email” | | | | |
| *7* | User | User click reset link in their email | | | | |
| *8* | Client | Navigates user to "Reset Password" page where user can enter a new password | | | | |
| *9* | User | Enters new password and confirms it, then submits | | | | |
| *10* | System | Validates token, updates password in the database, and displays confirmation message | | | | |
| *11* | Client | Redirects user to login page with success notification | | | | |

**Alternative flows**

| **AT1** | At the step 4, the system found that the email that user enters is not existed in database | |
| --- | --- | --- |
| **Sub step** | **Actor** | **Action** |
| 1.1 | System | Detects existing email |
| 1.2 | Client | Shows error message: “Email is not existed” |

| **AT2** | At the step 8, the token in reset link is invalid or expired | |
| --- | --- | --- |
| **Sub step** | **Actor** | **Action** |
| 2.1 | System | Return an error if the token is invalid or expired |
| 2.2 | Client | Shows message: “Reset link is invalid or expired” |
| 2.3 | Client | Provides option to resend reset link to set new password |

| **AT3** | At the step 9, the user enter password and confirm password is not matched | |
| --- | --- | --- |
| **Sub step** | **Actor** | **Action** |
| 2.1 | System | Return an error if password and confirm password is not matched |
| 2.2 | Client | Shows error message: “Confirm password is not match” and request user to enter again |

***Business Rules***

| **#** | **Rule Description** |
| --- | --- |
| BR-26 | User must enter a valid and registered email address to initiate password reset |
| BR-27 | Password reset token must be unique, secure |
| BR-28 | New password must meet minimum security requirements (e.g., minimum 8 characters, include letters and numbers) |
| BR-29 | Password and Confirm Password must match before submission |
| BR-30 | Only verified accounts are allowed to reset their password via email |

### 3.8 Home

#### **3.8.1 Home page**

##### 3.8.1.1 Screen Mock-up

##### 

**Figure 3-8**: Screen Design of Home page

**Table 3-8: Screen Definition**

| **#** | **Field Name** | **Type** | **Mandatory** | **Max Length** | **Description** |
| --- | --- | --- | --- | --- | --- |
| 1 | Username | Text(Link) | Yes | 20 | The display name of the user who posted. Clickable to navigate to profile page. |
| 2 | Post image | Text | Yes | N/A | The main image of the post. |
| 3 | Post caption | Text | No | 200 | The description or caption of the post. |
| 4 | Like count | Integer | Yes | N/A | The number of likes for the post. |
| 5 | Comment input | Text Input | No | N/A | Input field where user can type a comment. |
| 6 | Comment link | Icon Button | No | N/A | Link to view all comments of the post. |
| 7 | Post timestamp | Dropdown box | Yes | N/A | Opens a dropdown menu with actions (e.g., report, share). |
| 8 | Follow button | Button | No | N/A | Button to follow the suggested user. |
| 9 | More Options (⋯) | Icon Button | No | N/A | Opens a dropdown menu with actions (e.g., report, share). |
| 10 | Suggested User Name | Link | Yes | 20 | Username of the suggested account to follow. |
| 11 | View all comment | Button | Yes | N/A | Opens the comment section to view all comments. |
| 12 | Reaction Icons | Icon Group | No | N/A | Icons for actions: like, comment, share. |
| 13 | Add a Comment Placeholder | Placeholder text | No | N/A | Placeholder text in the comment input, e.g. “Add a comment…” |
| 14 | Suggestions for You - Card | Card Group | No | N/A | Cards showing user suggestions with avatar, name, and follow button. |
| 15 | Navigation Menu | Button group | Yes | N/A | Sidebar with navigation: Home, Explore, Notification, Create, Profile, More. |
| 16 | Profile picture | Image button | Yes | N/A | Avatar image of the user. Clickable to view their profile. |

**3.8.1.2 Use Case Description**

| **Use Case ID** | | **H-01** | **Use Case Name** | | **Home Page** | |
| --- | --- | --- | --- | --- | --- | --- |
| **Author** | | **KhanhNT** | **Version** | **1.0** | **Date** | **15/06/2025** |
| **Actor** | | Guest, User | | | | |
| **Description** | | This use case describes the behavior of the Home Page, where users can view posts, like them, comment, and interact with other users. It also includes suggested users to follow and navigation menu options. | | | | |
| **Precondition** | | PRE-01: Users have unbanned accounts  PRE-02: Users have a stable Internet connection  PRE-03: The user has successfully logged in.  PRE-04: Users have signed in successfully | | | | |
| **Trigger** | | TRG-01: The user signs in and is redirected to the Home Page. | | | | |
| **Post-Condition** | | POS-01: The Home Feed is displayed, showing posts from following users.  POS-02: All UI components such as post list, reaction icons, comment input, and sidebar navigation are rendered  POS-03: The user is able to interact with posts and navigate the app. | | | | |
| **Main flows** | | | | | | |
| ***Step*** | ***Actor*** | ***Action*** | | | | |
| *1* | User | Logs in successfully or navigates to home page via sidebar menu | | | | |
| *2* | System | Displays the Home Page with a feed of recent posts | | | | |
| *3* | System | For each post, display: Username (clickable), Profile Picture, Post Image, Caption, Like Count, Reaction Icons, Comment Input, Comment Link, Post Timestamp, More Options | | | | |
| *4* | User | Interacts with posts (like, comment, view comments, share, …) | | | | |
| *5* | System | Displays suggested users to follow in sidebar with: Avatar, Suggested Username, Follow Button | | | | |
| *6* | User | Click “Follow” button to follow a suggested user | | | | |
| *7* | System | Update follow status and suggests more users | | | | |
| *8* | User | Uses the Navigation Menu to switch to other sections (Explore, Notifications, Profile, etc.) | | | | |

**Alternative flows**

| **AT1** | User is a guest (not signed in) | |
| --- | --- | --- |
| **Sub step** | **Actor** | **Action** |
| 1.1 | Midia | displays a limited home page with login/register prompts and restricted interaction (no like/comment/post). |

***Business Rules***

| **#** | **Rule Description** |
| --- | --- |
| BR-31 | Users can only like or comment on a post if they are logged in |
| BR-32 | Each comment is limited to 200 characters |
| BR-33 | Username display is limited to 20 characters |
| BR-34 | "More Options" dropdown includes: Report Post, Share Post, Copy Link |
| BR-35 | Suggested users are based on similar interests, friends of friends, or system recommendation engine |

## 

## 4. Non-Functional Requirements

### 4.1 External Interfaces

* The system will interact with users with a standard web browser such as Microsoft Edge, Chrome, Opera,…
* **Database Storage**: Store accounts, posts,...

### 4.2 Quality Attributes

#### 4.2.1 Usability

* The UI must be simple and intuitive that normal users can understand how to use quickly and easily
* The UI must follow the usability standards such as
  + Naming elements clearly
  + Input validation notification
  + Navigation consistency
* The UI have cleaned and responsive layout

#### 4.2.2 Reliability

* Availability: The system should be available 99.98% hours of use, equivalent to downtime for 20 minutes per month.
* Mean Time Between Failures (MTBF): Should be larger than 3000 hours
* Mean Time To Repair (MTTR): Should be less than 30 minutes
* Backup & Recovery: Backup by block for each 6 hours and save it for a month
* Maximum Bugs: Must less than 10 bugs/KLOC

#### 4.2.3 Performance

* **Response Time**: Average response time is not larger than 1 second
* **Throughput**: The system can support concurrent active users accessing in one time (minimum 100 reqs/sec)
* **Latency for Media Delivery**: Preview image should be displayed no more than one second

## 5. Requirement Appendix

### 5.1 Business Rules

| **ID** | **Rule Definition** |
| --- | --- |
| BR-01 | Maximum of 20 notifications are fetched per request; subsequent pages use cursor-based pagination. |
| BR-02 | Notifications are considered unread when readAt is null. |
| BR-03 | Returned list must be strictly ordered by createdAt descending before grouping into sections. |
| BR-04 | Notifications older than 30 days are auto-archived and excluded from the “recent” endpoint. |
| BR-05 | The unread badge should never exceed “99+” characters; counts above 99 display as “99+”. |
| BR-06 | CTA mapping table determines which API endpoint is invoked per eventType (e.g., FOLLOW\_REQUEST, MENTION, LIKE). |
| BR-07 | A notification is auto-marked read after any successful CTA action or row navigation. |
| BR-08 | Follow/unfollow actions are rate-limited to 30 per hour per user to mitigate spam. |
| BR-09 | If the action has already been performed (e.g., user is already following), the service returns 409 Conflict; client shows inline warning without altering unread count. |
| BR-10 | Media Required: A post must contain exactly one media file (image or video). |
| BR-11 | Supported File Formats: The system shall only accept media files with the following extensions: .jpg, .jpeg, .png for images, and .mp4 for videos. |
| BR-12 | Maximum File Size: The size of the uploaded media file must not exceed 10MB for images and 100MB for videos. |
| BR-13- | Caption Length: The text caption is optional, but if provided, it must not exceed 2,200 characters. |
| BR-14 | Phone number must have 11 digits and is existed |
| BR-15 | Username must be unique |
| BR-16 | List of suggestions must be limited in 5 accounts |
| BR-17 | Phone number must have 11 digits and is existed |
| BR-18 | Username must be unique |
| BR-19 | Image file must be maximum 5MB |
| BR-20 | Image’s format should be JPG, PNG or WEBP |
| BR-21 | Username must be unique and password must be hashed securely. |
| BR-22 | Only active users (not locked or banned) can log in. |
| BR-23 | Google login is optional and only available if the email is verified. |
| BR-24 | Username and email must be unique |
| BR-25 | Password must be at least 8 characters |
| BR-26 | User must enter a valid and registered email address to initiate password reset |
| BR-27 | Password reset token must be unique, secure |
| BR-28 | New password must meet minimum security requirements (e.g., minimum 8 characters, include letters and numbers) |
| BR-29 | Password and Confirm Password must match before submission |
| BR-30 | Only verified accounts are allowed to reset their password via email |
| BR-31 | Users can only like or comment on a post if they are logged in |
| BR-32 | Each comment is limited to 200 characters |
| BR-33 | Username display is limited to 20 characters |
| BR-34 | "More Options" dropdown includes: Report Post, Share Post, Copy Link |
| BR-35 | Suggested users are based on similar interests, friends of friends, or system recommendation engine |

### 5.2 Common Requirements

### 5.3 Application Messages List

| **#** | **Message code** | **Message Type** | **Context** | **Content** |
| --- | --- | --- | --- | --- |
| 1 | MSG01 | In line | There is not any search result | *No search results.* |
| 2 | MSG02 | In red, under the text box | Input-required fields are empty | *The \* field is required.* |
| 3 | MSG03 | Toast message | Updating asset(s) information successfully | *Update asset(s) successfully.* |
| 4 | MSG04 | Toast message | Adding new asset successfully | *Add asset successfully.* |
| 5 | MSG05 | Toast message | Confirming email of asset hand-over is sent successfully | *A confirmation email has been sent to {email\_address}.* |
| 6 | MSG06 | Toast message | Resetting asset information successfully | *Return asset(s) successfully.* |
| 7 | MSG07 | Toast message | Deleting asset information successfully | *Delete asset(s) successfully.* |
| 8 | MSG08 | In red, under the text box | Input value length > max length | *Exceed max length of {max\_length}.* |
| 9 | MSG09 | In line | Username or password is not correct when clicking sign-in | *Incorrect username or password. Please check again.* |

### 