Assignment P02_ScreenSketches

Team members:
Caelan Herzberg, Josh Polich, Jian Shi, Wenqin Wu

Team Number:
1_YN_6

Date:
3/24/2022

Professor:
Simanta Mitra

TA:
Yididiya Nadew

Content

Step 2: Use Cases (Extension may just a idea for now and will not write any more use cases for extension currently):

Use-cases1: Register account
Actors: Guest User
Triggering Event: Users click on the register button if they do not have an account.
Success Guarantee: Provided a registration DB
Preconditions: User information(Basic info, Name, password, username, DOB…)
Main Success Scenario:

1. User can use the comment functions after registration
2. Users can switch accounts based on the registration information.(Log in, log out)

Extensions: Add on email or phone number register confirmation.

Use-cases2: user profile menu
Actors: User
Triggering Event: User selects profile button and clicks on it
Success Guarantee: Menu for User profile is opened
Preconditions: User must have a registered account
Main Success Scenario:

1. User profile menu is opened
2. User can choose to change profile picture, change password, change username, delete account, or return to home screen by selecting corresponding buttons
   Extensions:

Use-cases3: change profile picture
Actors: User
Triggering Event: User presses change profile picture button in change user profile menu
Success Guarantee: User profile picture is changed
Preconditions: User has a registered account and is in the user profile menu
Main Success Scenario:

1. User selects change profile picture button
2. User is prompted to select photo from phone gallery
3. User profile picture is changed to the selected profile picture

Extensions:

Use-cases4: change username
Actors: User
Triggering Event: User presses change username button
Success Guarantee: User username is changed
Preconditions: User has a registered account and is in the user profile menu
Main Success Scenario:

1. User selects change username button

2. User is prompted with textbox
3. User types in new desired username
4. User username is changed

Extensions:

Use-cases5: change password
Actors: User
Triggering Event: User presses change password button
Success Guarantee: User password is changed
Preconditions: User has a registered account and is in the user profile menu
Main Success Scenario:

1. User selects change password button
2. User is prompted with textbox
3. User types in new desired password
4. User username is changed

Extensions:

Use-cases6: delete account
Actors: User
Triggering Event: User clicks delete user account button
Success Guarantee: User account is deleted from database
Preconditions: User has registered account and is in profile menu button
Main Success Scenario:

1. User selects delete user account button
2. User is given a confirmation of if they really want to delete user account or not
3. User is prompted in a textbox to enter their password
4. User inputs correct password and account is set to be terminated in the database

Extensions:

Use-cases7: leave profile menu
Actors: User
Triggering Event: User clicks back button on profile menu
Success Guarantee: User is returned to the home page
Main Success Scenario:

1. User selects back button on profile menu
2. User is sent back to home page

Extensions:

Use-cases8: Leave Review
Actors: User
Triggering Event: User click comments box and write info in the box to commit
Success Guarantee: Comments are in the database and other user can see it

Preconditions: There is a database for the comments
Main Success Scenario:

1. The comments box has been selected
2. User type in some words
3. User selected done
4. Comments upload to the server database

Extensions:

User can edit the comments after pushing to the server

Use-cases9: View recommendations
Actors: User, Guest User
Triggering Event: User clicks on the item they want to view recommendations on
Success Guarantee: recommendations are opened
Preconditions: none
Main Success Scenario:

1. Recommendation page is opened
2. User can look through their recommendations
3. User can click on an item to leave a review

Extensions:

Use-Cases10: Ban user's account
Actors: Admin
Triggering Event: Admin selected the user and click ban selection
Success Guarantee: A banned user can't log in
Preconditions: A user violate the rules and user account is in active status
Main Success Scenario:

1. User has been selected
2. Admin ban account
3. The account has been diabled and can't login anymore

Extensions:

Admin can re-enable the account if that is a mistake.

Use-Case11: Delete comments
Actors: Admin
Triggering Event: Admin spot some toxic content in the comment
Success Guarantee: delete the comment in the DB
Preconditions: User posted comments
Main Success Scenario:

1. Society is more friendly and less dirty.
2. Users can delete comments themselves if they want to.

Extensions:

Add a report function so that the user can as admin to delete some potential toxic comment.

Use-Case12: Change text font size

Actors: Eye issue user or older user

Triggering Event: User click on the setting and change the app text size to larger or smaller

Success Guarantee: Text changed after submit changes

Preconditions: User want to change the text size

Main Success Scenario:

1. User click on the setting button
2. User choose the size of text in the setting menu
3. User submit the changes

Extensions:

The app can read something for disability people such as blind people.

Step 3 Non-functional Requirement:

External requirement:

Two-factor authentication: In our idea so far, the login function has no verification yet. In some cases, if anyone knows your password and username, they will log in to the App easily. We want to add a security feature when you log in. We will verify your phone number by a text message code and let you type in your codes to verify you are using the App right now.

Organizational requirement:

Response time should be less than 1-2 secs: We don't want users to wait too long when they click on a button or transfer screens. We may change the efficiency of the algorithm to help the app response time much faster.

Product requirement:

LTE use: The program would have the option to reduce element downloading for pictures/videos to save on data usage when the user is not on wifi. This would be an optional setting.

Usability Requirement:

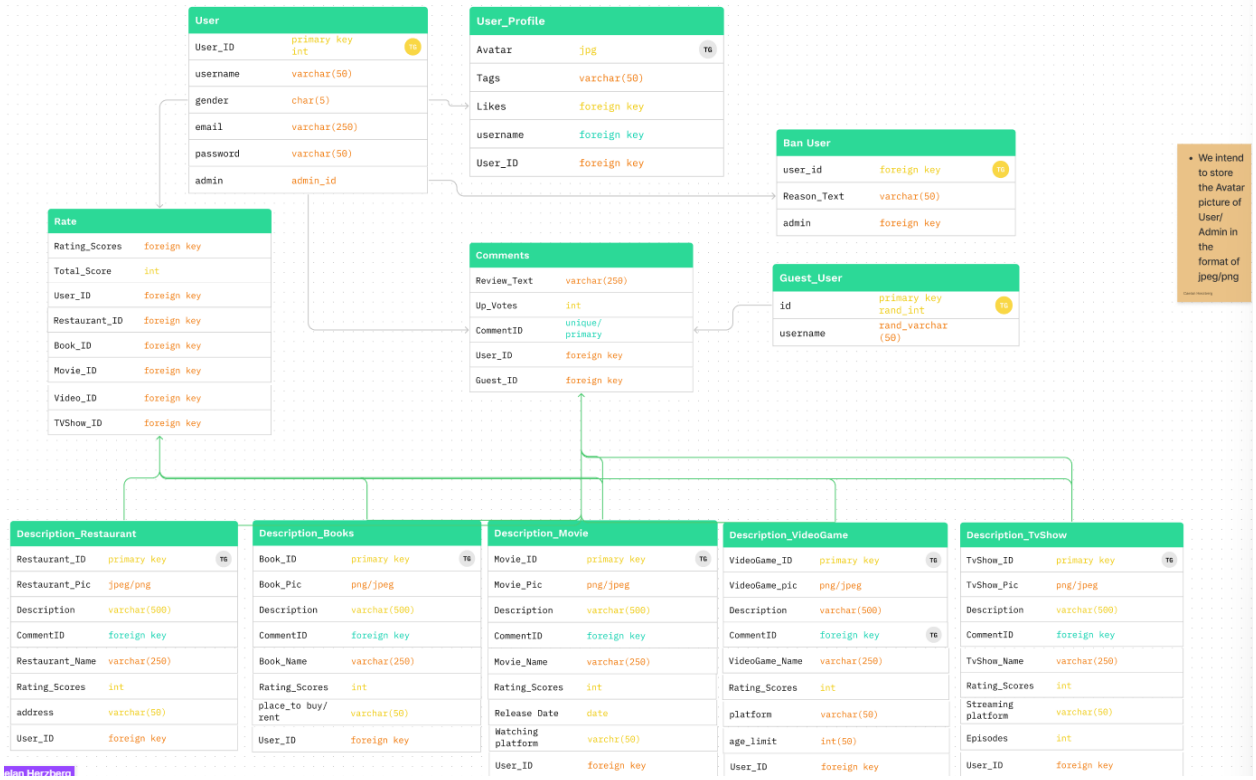
Graphical interface of the user face must be functional, but would also be better if it were easy on the eyes. As of now we are using built in graphics from figma/android studio but we could explore other avenues for graphics in our app.

Usability Requirement:

Our app currently does not have disability accommodations, but we could include some. Currently we are considering allowing users to enlarge text for visual impairment and a color blind setting for color blindness.

Step 4 (Database and relationship):

Tables_Relationship



Roles tables:

Here is a small change about all tables. So far each table has an id (primary key) used for searching, editing, and so on.

User table:

Attributes: userID(int), Username(Varchar), Password(Varchar), Avatar(jpeg), gender(varchar), email(varchar), admin(admin_ID)

Primary Key Int: userID

Foreign Key:

Guest_User(Still fixing and figure out a way):

Primary Key: id(rand_int), username(rand_varchar)

Attributes: id(int), Username(Varchar)

Ban User:
Primary key: id (long)
Attributes: Reason_text(varchar)
Foreign key: userID, admin
—----------

Admin:
Primary key: admin_id(long)
Attributes: admin_id(long), user_id(long)
Foreign key: user_id

—----------

Functional tables:

Comments table:
Attributes: CommentID(Integer), ReviewText(Varchar), upVotes(Integer), userID(int), guestID(int), movie_id(long), book_id(long), video_game_id(long), tv_show_id(long), restaurant_id(long)
Primary Key Int: CommentID
Foreign Key: UserID, movie_id, book_id, video_game_id, tv_show_id, restaurant_id

—----------

Rate table (Still thinking about this part):
Attributes: Total_Score(int)
Primary Key: id(long)
Foreign Key: Rating_Scores, Total_Score, UserID, Resturaunt_ID, Book_ID, Movie_ID, Video_ID, TVShow_ID

—----------

Description_Movie:
Attributes: Movie_ID(int), Movie_pic(Unsure type), Description(Varchar), Movie_Name(Varchar), Rating_Score(int), Release_Date(date), Watching_Platform(varchar)
Primary Key: Movie_ID
Foreign Key: User_ID, Comment_ID

—----------

Description_VideoGame:
Attributes: VideoGame_ID(int), VideoGame_pic(Unsure type), Description(Varchar), VideoGame_Name(Varchar), Rating_Scores(int), platform(Varchar), age limit(varchar)
Primary Key: VideoGame_ID
Foreign Key: User_ID, Comment_ID

—----------

Description_Restaurant:

Attributes: Restaurant_ID(int), Restaurant_pic(Unsure type), Description(Varchar), Restaurant_Name(varchar), Rating_Scores(int), address(Varchar)

Primary Key: Restaurant_ID

Foreign Key: User_ID, Comment_ID

—----------

Description_TvShow:

Attributes: TvShow_ID(int), TvShow_pic(Unsure type), Description(Varchar), TvShow_name(varchar), Rating_Scores(int), Streaming platform(varchar), Episodes(int)

Primary Key: TvShow_ID

Foreign Key: TvShow_ID, CommentID

—----------

Description_Books:

Attributes: Book_ID(int), Book_pic(Unsure type), Description(Varchar), Book_Name(Varchar), Rating_Scores(int), Place to buy/rent(varchar)

Primary Key: Book_ID

Foreign Key: User_ID, CommentID

—----------

Movie_Comment:

Attributes: user_id, movie_id

Primary Key: id

Foreign Key: User_ID, Movie_id

—----------

Book_Comment:

Attributes: user_id, book_id

Primary Key: id

Foreign Key: User_ID, book_id

—----------

Video Game_Comment:

Attributes: user_id, video_game_id

Primary Key: id

Foreign Key: User_ID, video_game_id_id

—----------

TV Show_Comment:

Attributes: user_id, tv_show_id

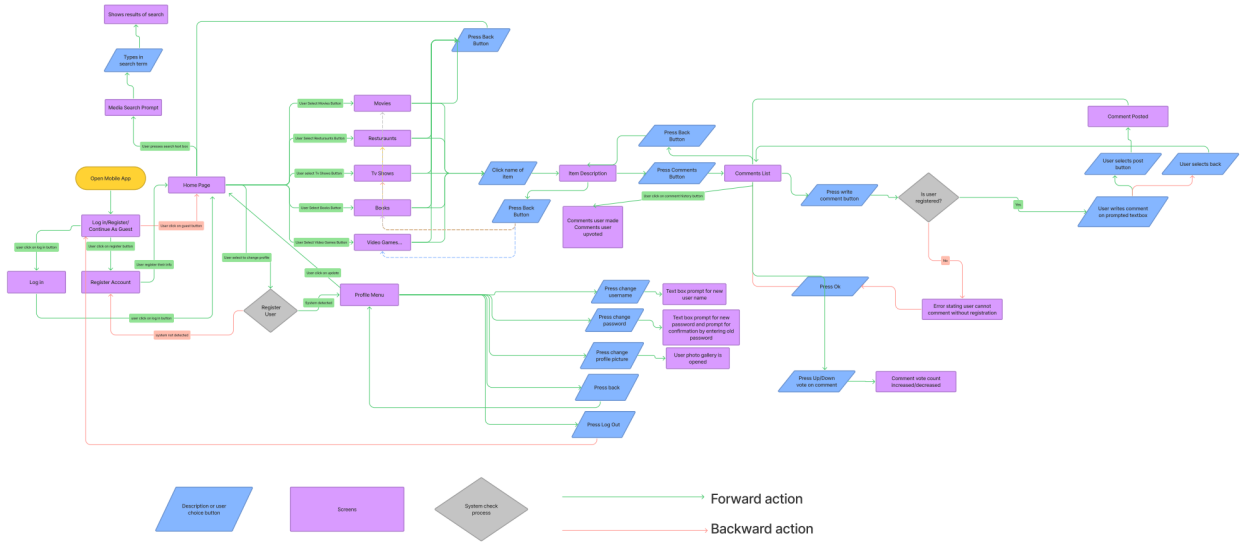
Primary Key: id

Foreign Key: User_ID, tv_show_id

—----------

Restaurant_Comment:
Attributes: user_id, Restaurant_id
Primary Key: id
Foreign Key: User_ID, Restaurant_id

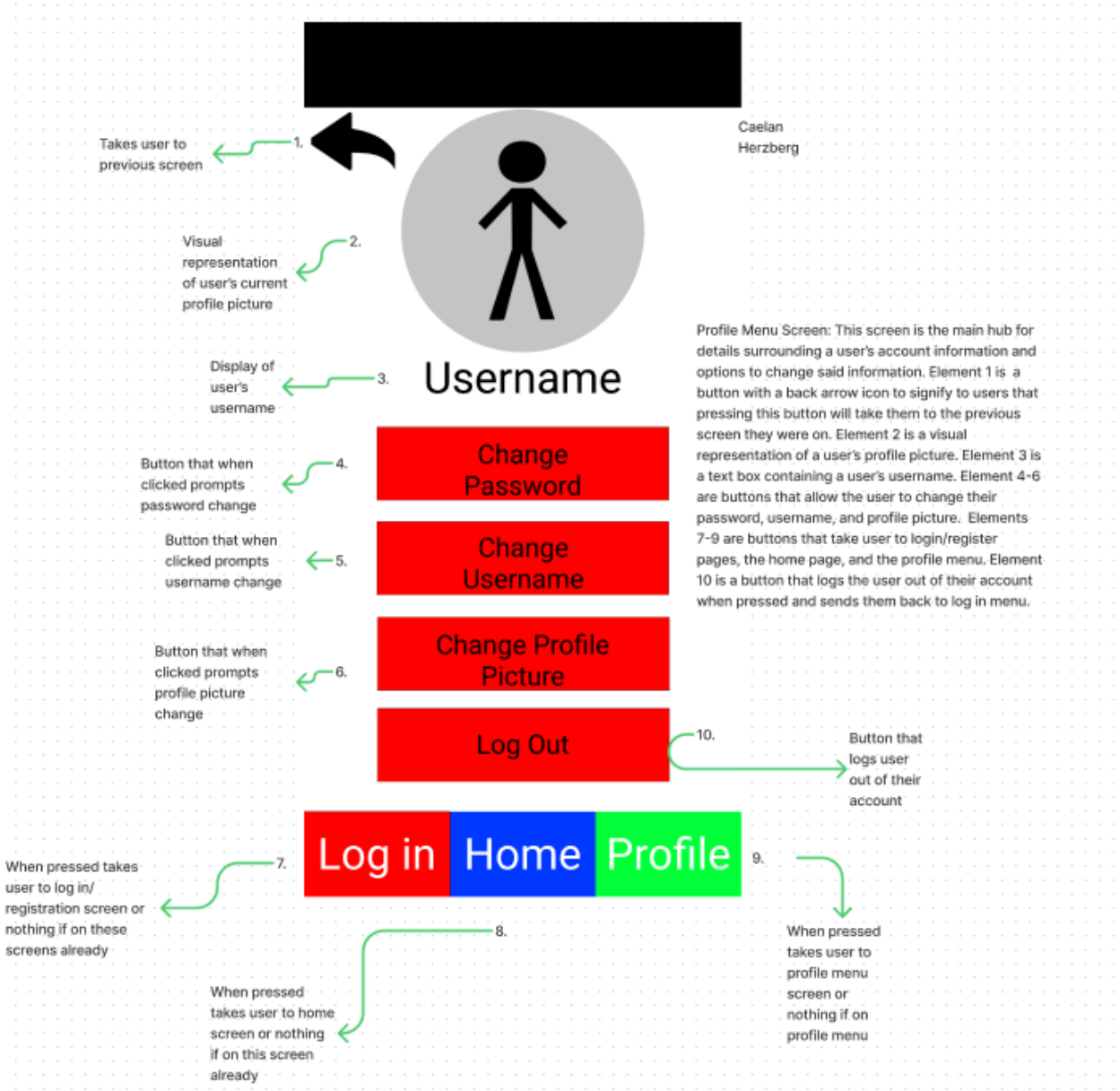
Step 5 Screen Flow Chart:

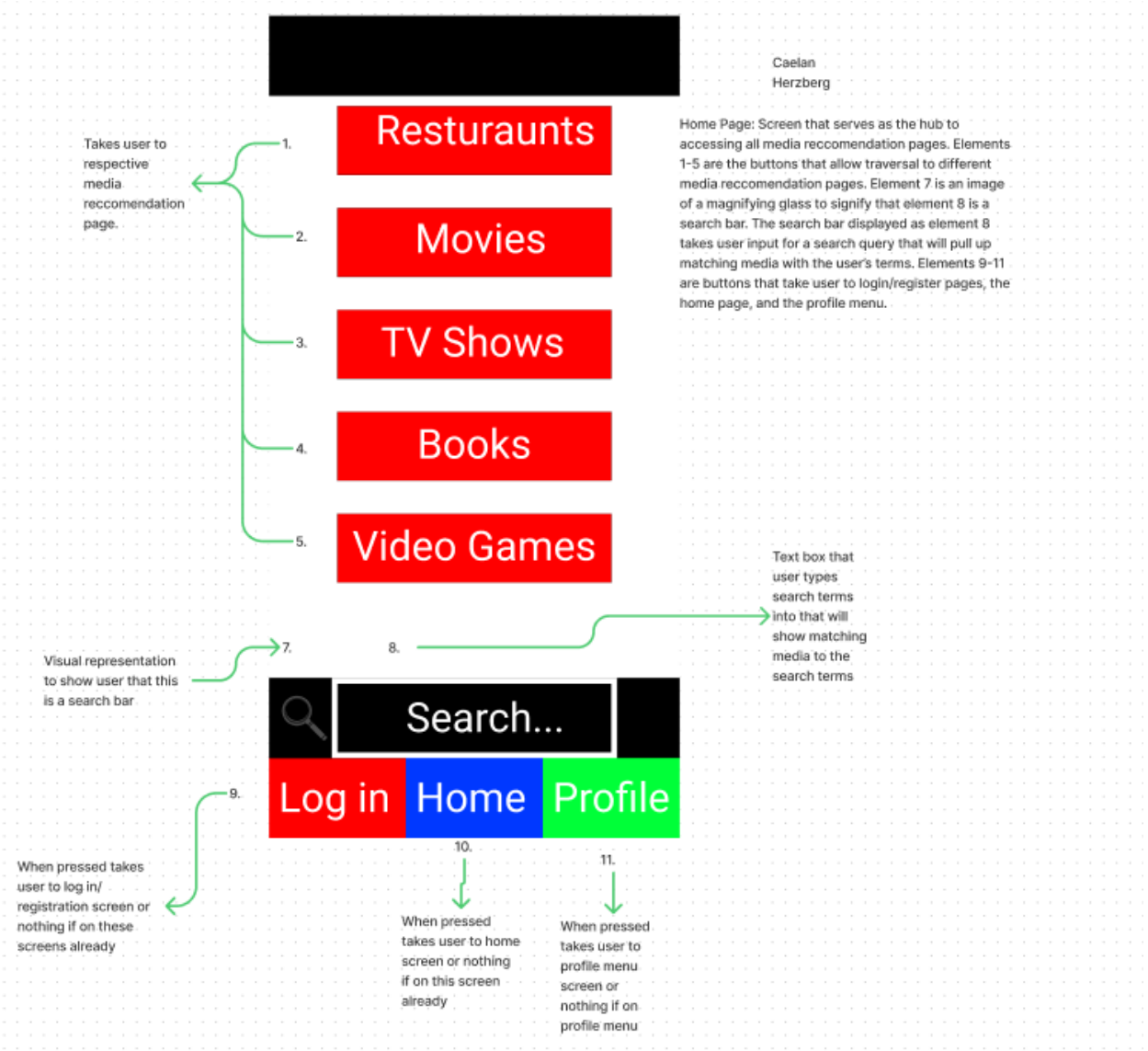


Link for better vision (iastate.edu):

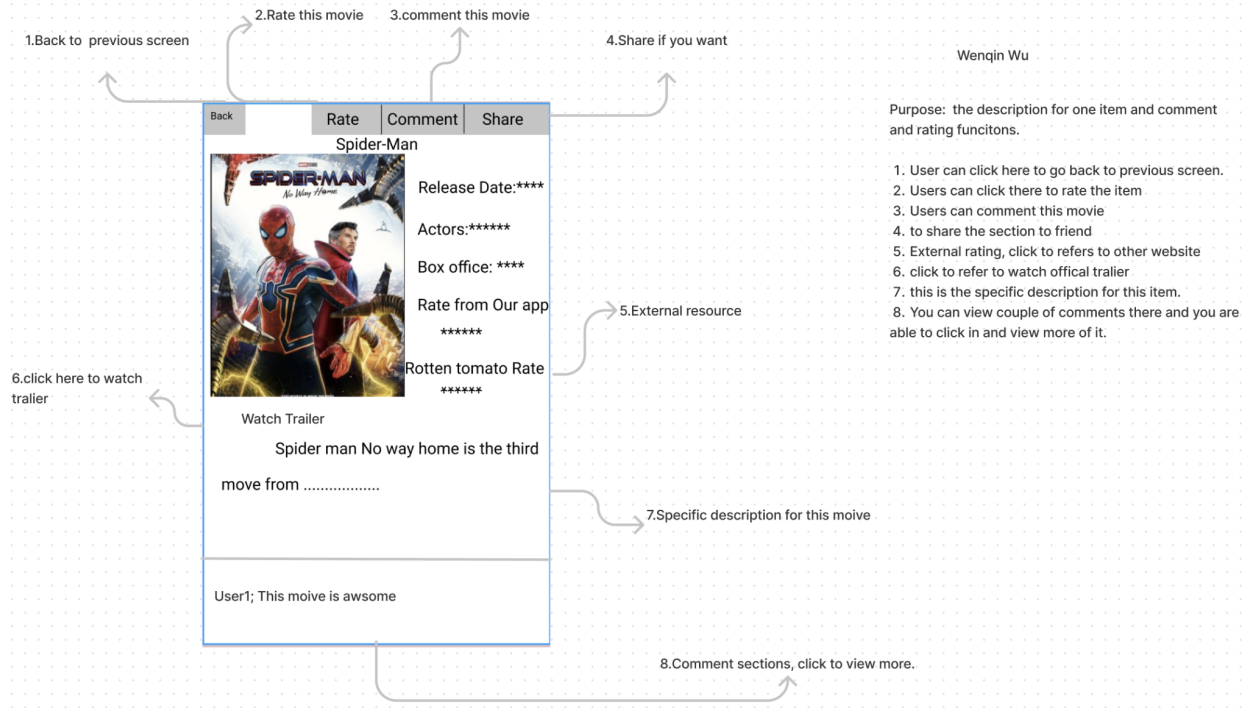
https://drive.google.com/file/d/1GRqFS7Yms6cBnKRayRUu_AHWzxn58zIE/view?usp=sharing

Step 6 (Screen Sketches individual works):

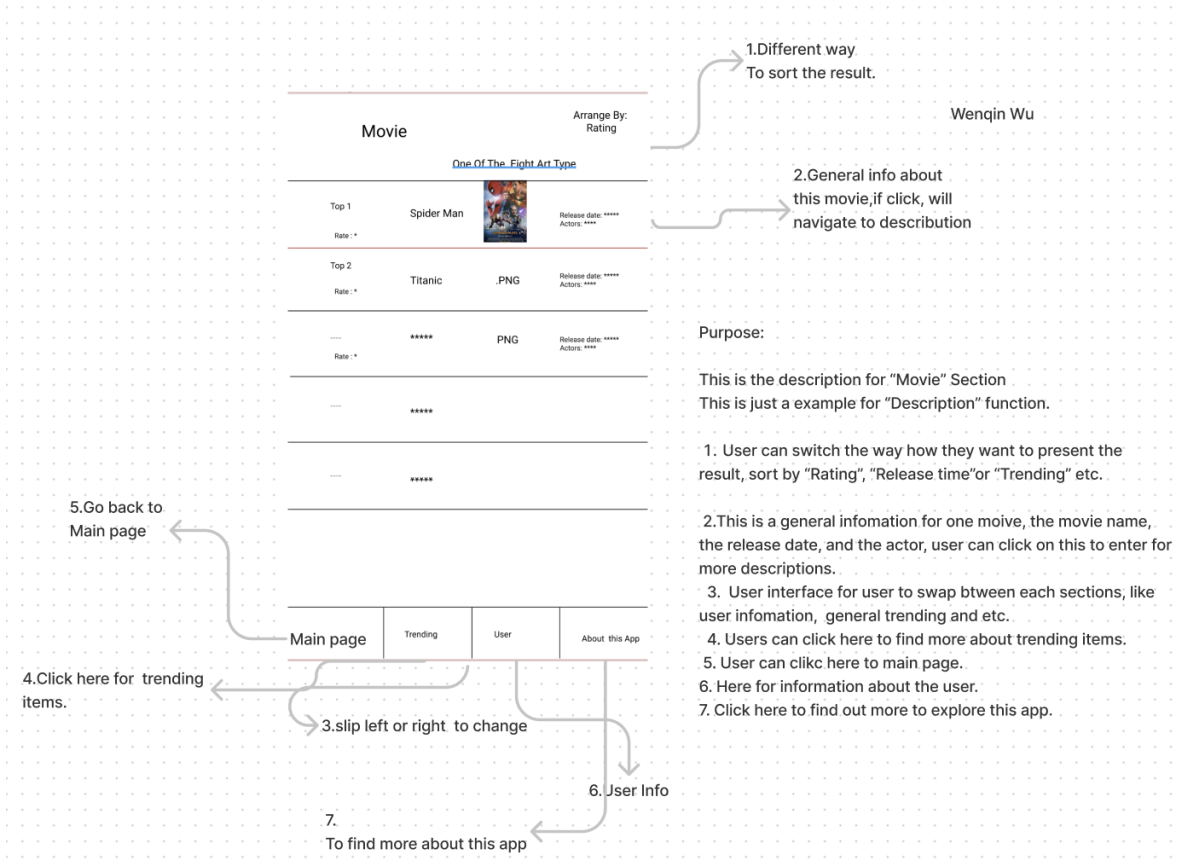


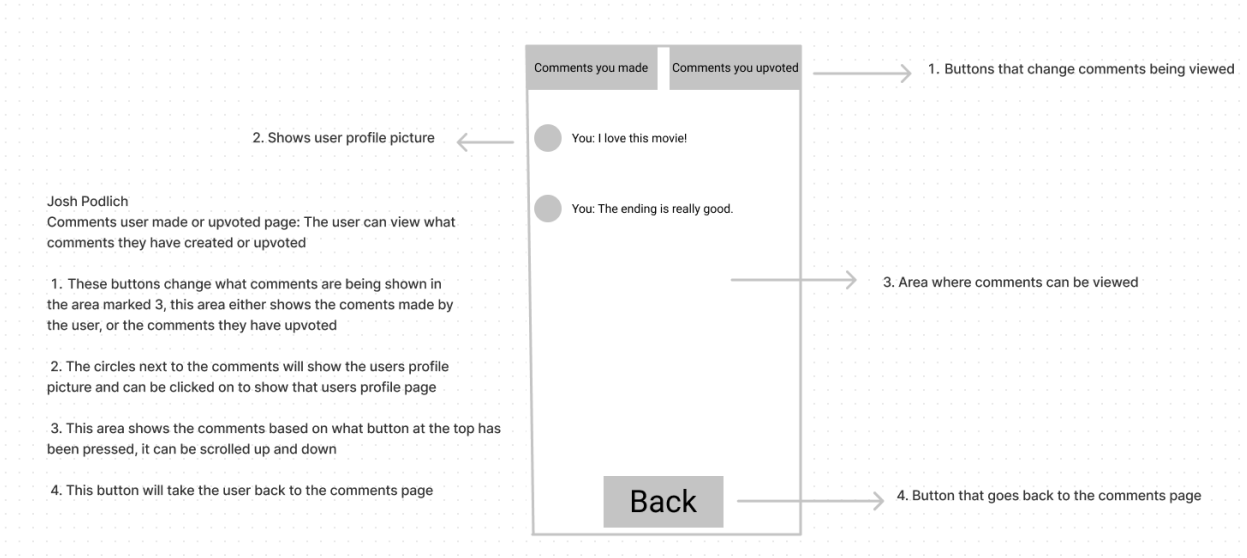

Caelan Herzberg
Home Page: Screen that serves as the hub to accessing all media reccomendation pages. Elements 1-5 are the buttons that allow traversal to different media reccomendation pages. Element 7 is an image of a magnifying glass to signify that element 8 is a search bar. The search bar displayed as element 8 takes user input for a search query that will pull up matching media with the user's terms. Elements 9-11 are buttons that take user to login/register pages, the home page, and the profile menu.
Takes user to respective media reccomendation page.
1.
Resturaunts
2.
Movies
3.
TV Shows
4.
Books
5.
Video Games
Text box that user types search terms into that will show matching media to the search terms
Visual representation to show user that this is a search bar
7.
8.
Search...
9.
Log in
Home
Profile
10.
11.
When pressed takes user to log in/ registration screen or nothing if on these screens already
When pressed takes user to home screen or nothing if on this screen already
When pressed takes user to profile menu screen or nothing if on profile menu

1. Description

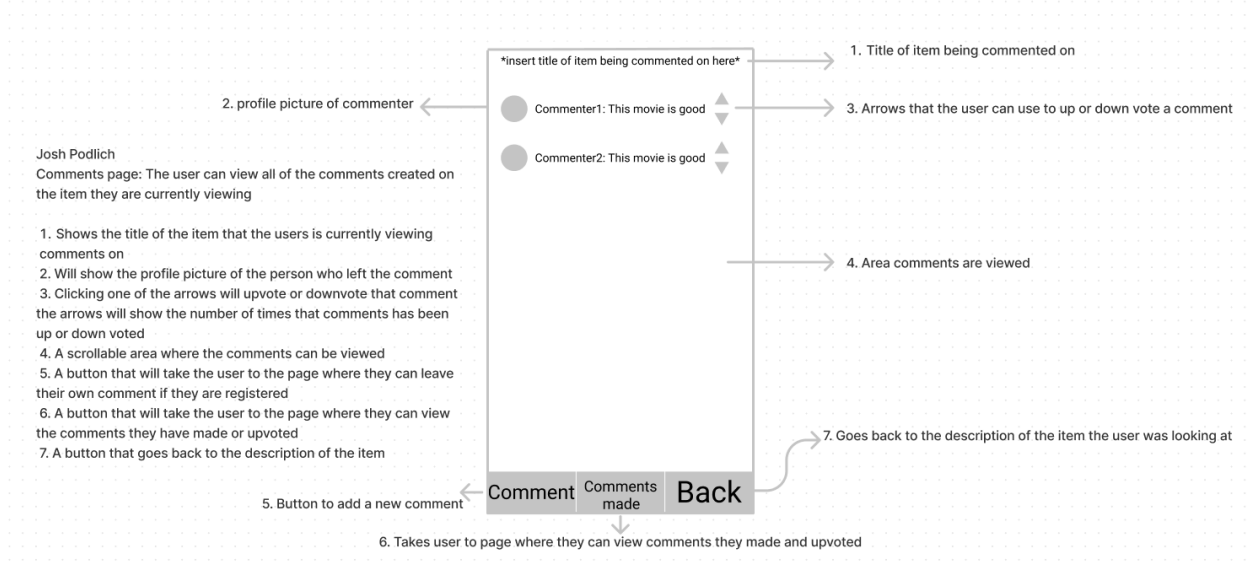


2.Category:

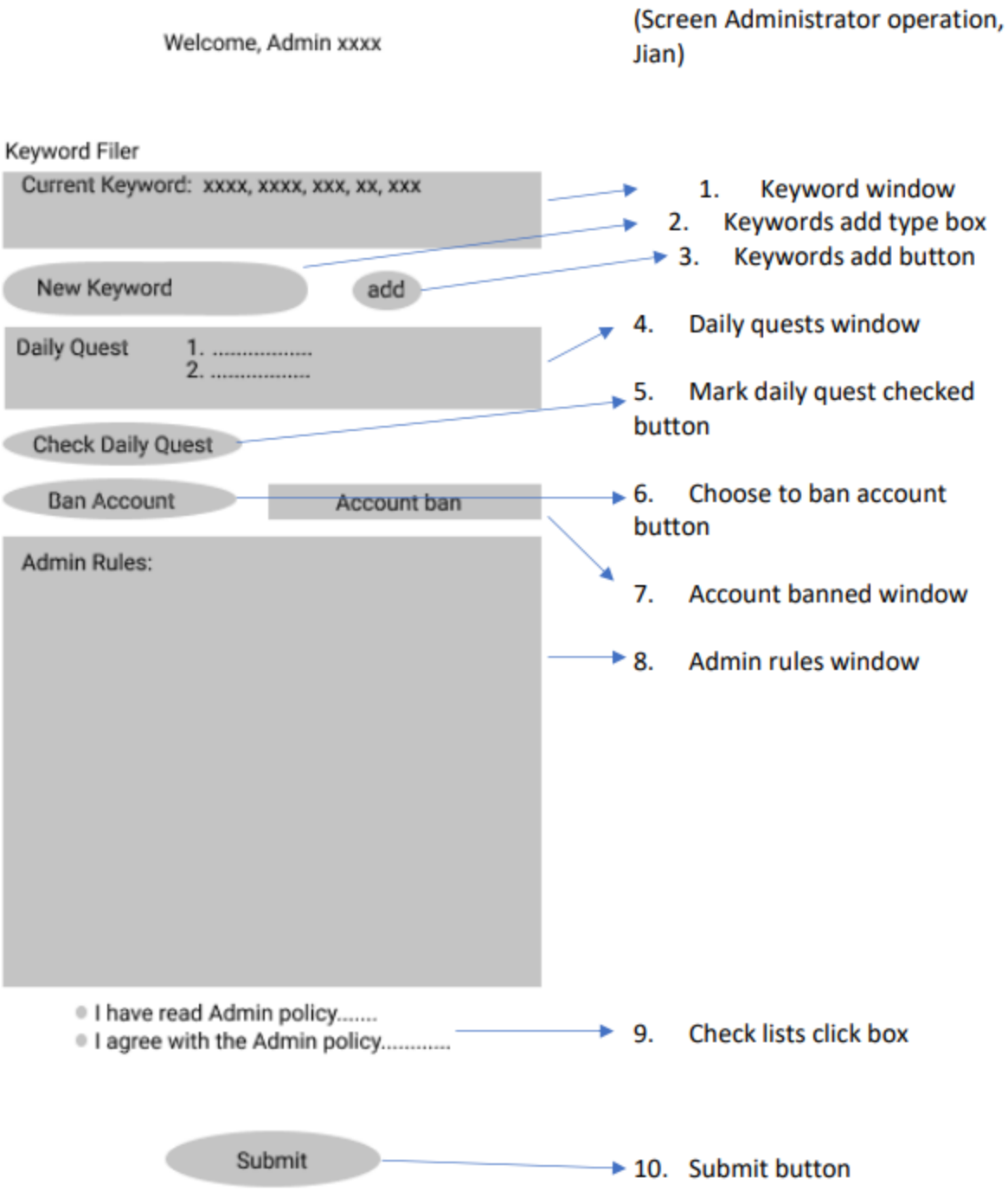




From Josh Podlich



From Josh Podlich


Welcome, Admin xxxx
(Screen Administrator operation, Jian)
Keyword Filer
Current Keyword: xxxx, xxxx, xxx, xx, xxx
New Keyword
add
Daily Quest
1. .....
2. .....
Check Daily Quest
Ban Account
Account ban
Admin Rules:
I have read Admin policy.....
I agree with the Admin policy.....
Submit
1. Keyword window
2. Keywords add type box
3. Keywords add button
4. Daily quests window
5. Mark daily quest checked button
6. Choose to ban account button
7. Account banned window
8. Admin rules window
9. Check lists click box
10. Submit button

Purpose:

This page is used for administrator control. Sometimes, when admin login, they have an extra screen for users. The operation screens are one of those. This page is mainly operated on keywords filter add on, ban account, and finish admin daily quests.

Label:

Keyword window is used for admin to see current keywords are filtering (1)

Keyword add type box is used for adding new keywords (2)

Keywords add button is used for adding to the server after clicking (3)

Daily quests window is a summarize preview daily quests for admin to look (4)

Check daily quests button is used for directing to a detailed screen with daily quests (5)

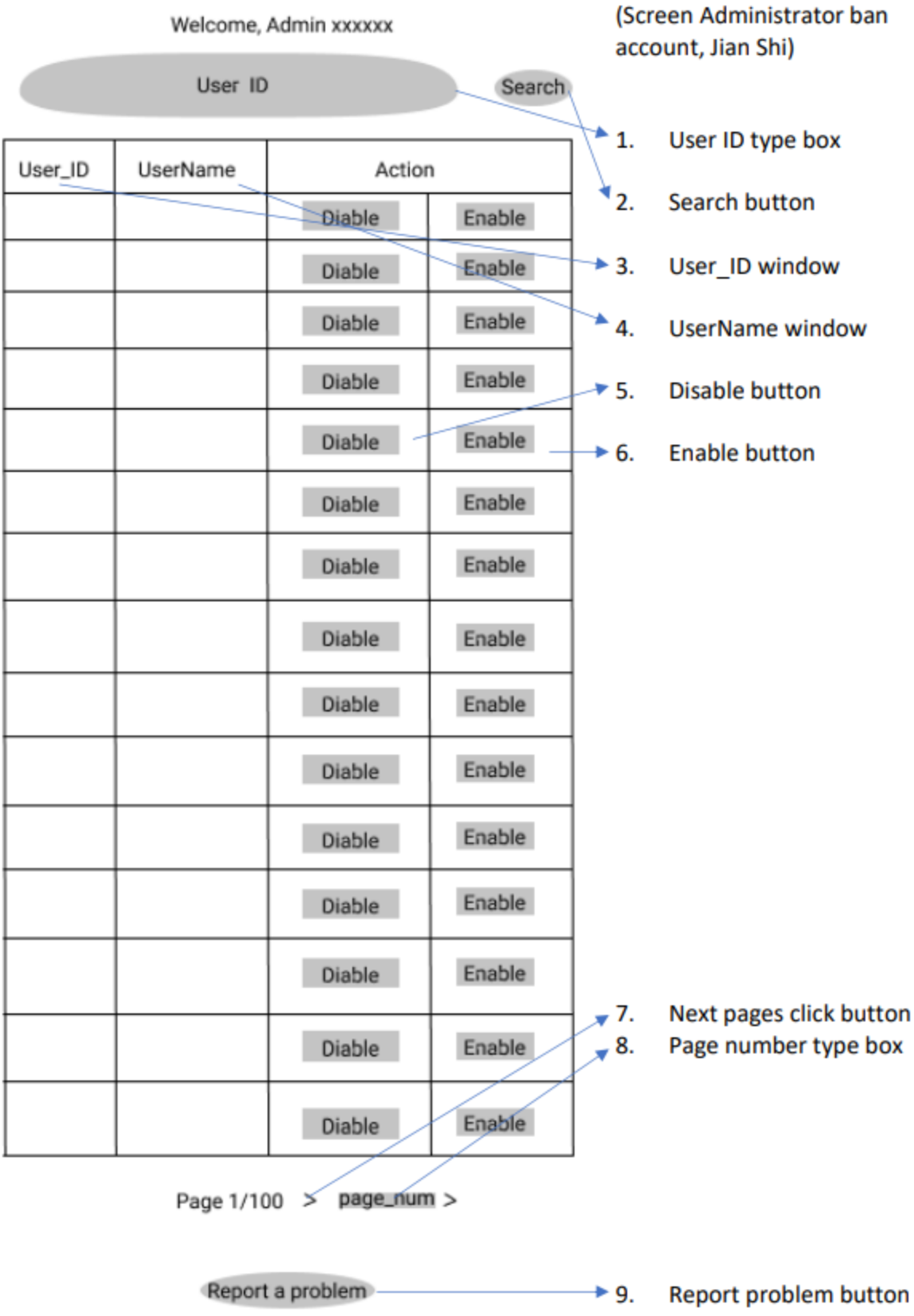
Ban account button is used for directing admin to the user ban page (6)

Account banned preview window is used for admin preview their selected user in the ban account screen (7)

Admin rules window is used for letting admin know rules (8)

Check lists click box is used for forcing admin to read the rules and agree with following rules (9)

Submit button is used for submitting the changes or any forms into the server after clicking.


Welcome, Admin xxxxxx
User ID
Search
User_ID
UserName
Action
Diable
Enable
Page 1/100 > page_num >
Report a problem
(Screen Administrator ban account, Jian Shi)
1. User ID type box
2. Search button
3. User_ID window
4. UserName window
5. Disable button
6. Enable button
7. Next pages click button
8. Page number type box
9. Report problem button

Purpose:

This page is used for administrators to ban an account by searching or finding the user in the list below. After clicking on the enable or disable, the user will be saved on the slot of waiting until the administrator clicks submit on the previous page.

Label:

User ID type box is used for a admin to search a user by ID (1)

Search button is just a confirm button after admin type in the user ID (2)

User ID window will show all register users with their username is the next window (3)(4)

Disable button is used for an admin to disable the account after confirming (5)

Enable button is used for an admin click disable by mistake that may cause a user account inactive (6)

Next pages click button is used for admin look up user manually and turn to next page (7)

Page number type box is used for admin to type a specific page they want to go (8)

Report a problem button is used for report problems while admin met during using app such like a mistake operation or some app bug they found (9)