Use Case Diagram

Use Case Diagram for Microblogging Application showing the User and Another User actors connected through a Generalization relationship. The User actor connects to use cases including: login (which «include» Verify pasword, which «include» hide password function, and «extend» display login error), quote tweet, Retweet (with undo Retweet «include»), create tweet («extend» to Retweet, «include» sizing), like tweet (with undo like «extend»), new feeds, comment (with reply, «extend» undo bookmark), Bookamrk, change password, edit profile («extend» to add Bio, edit name, location, add cover, add website, add profile picture, Go to profle help), signup («extend» to Check uniqe function, go to privacy help), and logout.

Database Specification

ERD

Tables

Bookmarl

PK	id
FK	user_id
FK	post_id
	time

User

PK	id
	username
	Email
	password
	name
	img
	imgCover
	Bio
	Location
	website

refresh_token

PK	id
	created_date
	refersh_token

Retweet

PK	post:id
	retweet_msg
FK	tweet_id
FK	retweet_id

posts

PK	ID
FK	user_id
	post_on

Tweet

PK	post:id
	created_date
	like_counter
	reply_counter
	text
	type
	tweet_id
	user_id

like

PK	id
FK	user_id
FK	post_id
	time

Comments

PK	id
	comment_
FK	user_id
FK	pst_id
	time

replies

PK	id
Fk	Comment_id
	user_id
	reply
	time

Class Diagram

Bookmark
-ID int
+tweet: Tweet
+created: bool

comment
+ID
+text
+Date
+author
+craete_comment()

User
-ID: int
+username: string
-Email: string
-password: string
+phone: number
+gender
+B_day

+signup()
+login()
+logout()
+follow(user.User)
+Unfollow(user.User)
+editPassword()
+postTweet(tweet text.string)
+likeTweet(): like
+getUserinfo(): list<User>
+replyToTweet(): Reply
+Bookmark()

tweet
-ID int
+text: string
+Date: Date
+user: user
+retweet
+likes

+post()
+getTweetinfo()
+getData()
+editTwett()
+getLikesCounter()

Reply
-ID int
+text string
+author: user
+Date: Date

+createreply()

profile
+profilePlcPath: string
+name: string
+tweet: list<Tweet>
+Bio: string

+addProfilePicpath()
+editname()
+addBio()

like
-ID int
+tweet: Tweet
+who: User
+created: bool

Quote tweet
+text: string

Retweet
-ID
+tweet: tweet
+user: user
+Date

+retweet()

make
create
have
have
make
make
make
make
have

0   1   1   1   1   0...*   0...*   1   1   0...*   0..*   1   1   0...*   0....*   0...*   0

OCL

Context profile

Inv:

--Constraint User must have profile picture
not self.profilepicture().isEmpty()

-- Constraint Profile name cannot exceed 15 characters
Slef.bio.size()<=15

-- Constraint User bio cannot exceed 160 characters
  slef.bio.size()<=160

Context User
Inv:
-- Constraint: Username must be unique
  User.allInstances()->forAll(u | u <> self implies u.username <> self.username)

  -- Constraint: Password size must be between 5 and 15 characters
  self.password.size() >= 5 and self.password.size() <= 15

-- Constraint: Email must be valid
  self.email.isValid()

Constraint: Username should have at least 3 characters
  self.username.size() >= 3

  -- Constraint: First 2 characters of the username should be letters
  self.username.at(1).isLetter() and self.username.at(2).isLetter()

Pre:
--Constraint User creation requires username , Email and password
 not self.username().isEmpty() and not self.email().isEmpty() and not self.password.isEmpty()

Context tweet

Inv :

--Constraint tweet must not be empty
Self.text<> ""

-- Constraint: tweet content cannot exceed 280 characters self.content.size() <= 280

Post:

--: Constraint After posting a tweet, the tweet count of the user should increase by 1
  self.author.tweetCount = self.author.tweetCount@pre + 1

Context like:

inv:
  -- Constraint : Like should have an associated user and tweet
  self.liker <> null and self.tweet <> null

pre:
  -- Constraint : User must be authenticated to like a tweet
  self.liker.isAuthenticated = true

-- Constraint : User should not be able to like the same tweet more than once
  not self.liker.likedTweets->includes(self.tweet)

post:

-- Constraint After liking a tweet, the like count of the tweet should increase by 1
  self.tweet.likeCount = self.tweet.likeCount@pre + 1
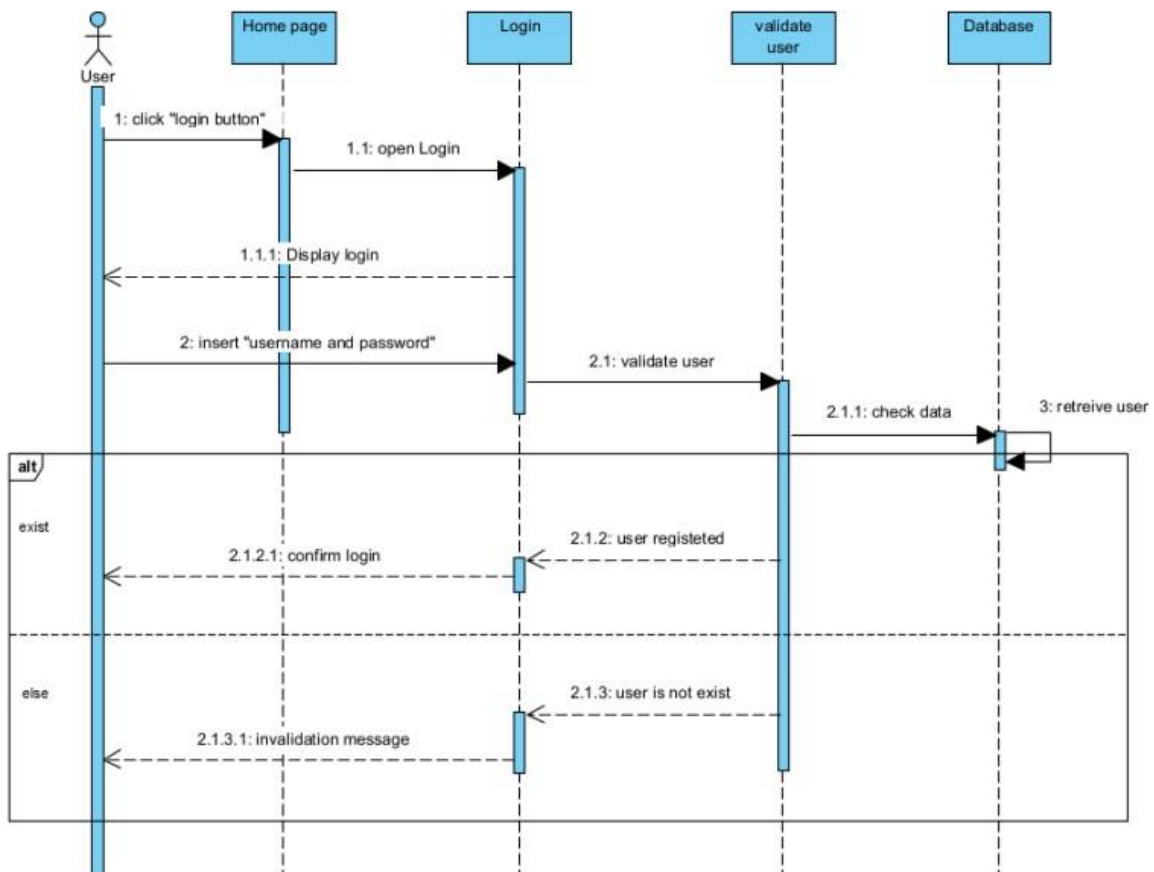
  -- Constraint : After liking a tweet, the tweet should be added to the like list of liked tweets
  self.bookmarker.bookmarkedTweets->includes(self.tweet)

Context Retweet

inv:
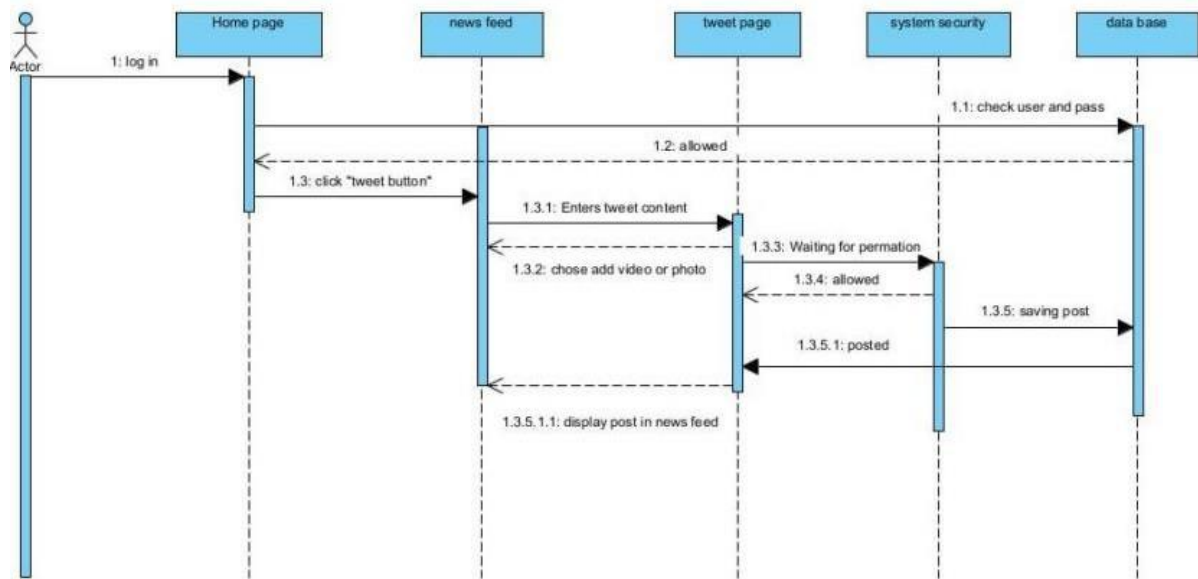 -- Constraint : Retweet should have an associated user and original tweet
  self.retweetedBy <> null and self.originalTweet <> null

pre:

  -- Constraint : User cannot retweet the same tweet more than once
  not self.retweetedBy.retweetedTweets->includes(self.originalTweet)

post:
-- Constraint : After retweeting, the original tweet's retweet count should increase by 1
  self.originalTweet.retweetCount = self.originalTweet.retweetCount@pre + 1

Context Reply

inv:
  -- Constraint : Reply content should not be empty
  self.content <> ''

pre:
  -- Constraint : User must be authenticated to reply to a tweet
  self.replier.isAuthenticated = true

post:
  -- Constraint : After replying to a tweet, the reply should be added to the original tweet's replies
  self.originalTweet.replies->includes(self)

  -- Constraint : After replying to a tweet, the reply count of the original tweet should increase by 1
  self.originalTweet.replyCount = self.originalTweet.replyCount@pre + 1 Constraint

Context Quote tweet:

Inv:
-- Constraint: Quote content should not be empty
  self.quoteContent <> ''

post:
-- Constraint: After quoting a tweet, the quoted tweet's quote count should increase by 1
  self.quotedTweet.quoteCount = self.quotedTweet.quoteCount@pre + 1

Context Bookmark

inv:
  -- Constraint: Bookmark must belong to a user
  self.user <> null

  -- Constraint: Bookmark must reference a valid tweet
  self.tweet <> null

Pre:
  -- Constraint: User cannot bookmark the same tweet more than once
  not self.bookmarker.bookmarkedTweets->includes(self.tweet)

post:
  -- Constraint: After bookmarking a tweet, the tweet should be added to the bookmarker's list of bookmarked tweets
self.bookmarker.bookmarkedTweets->includes(self.tweet)

Sequence Diagrams

Sign_up:

Login:

Tweet:

Likeatweet:

	Home page	news feed	tweet page	data base
Actor				

1: log in

1.1: check user and pass

1.2: allowed

1.3: chose the post

1.3.1: click"like button"

1.3.1.1: add the like to the like counter

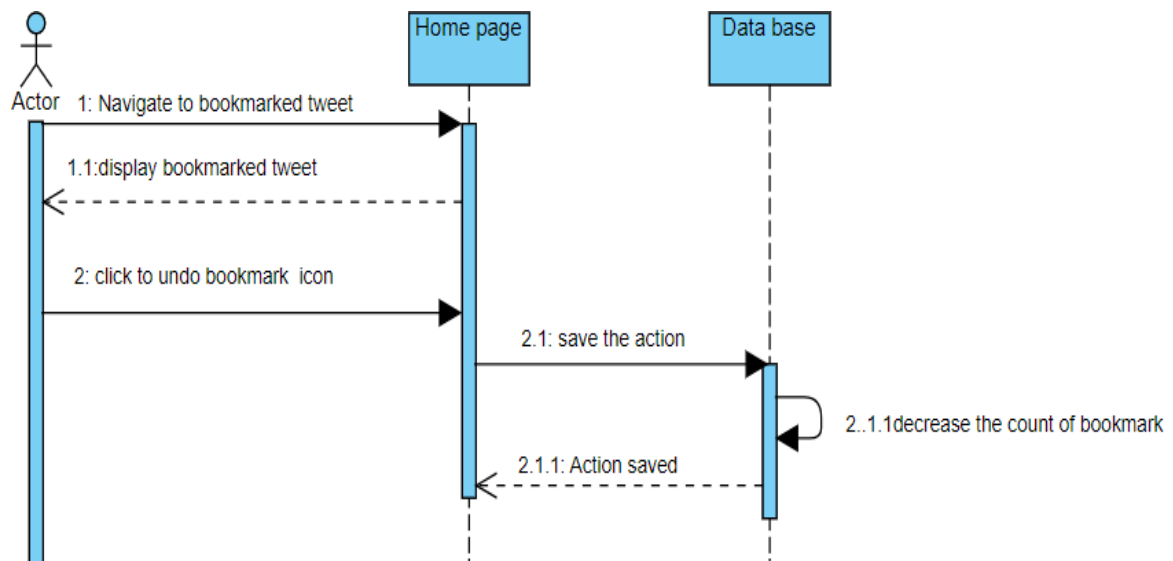1.3.1.1.1: liked
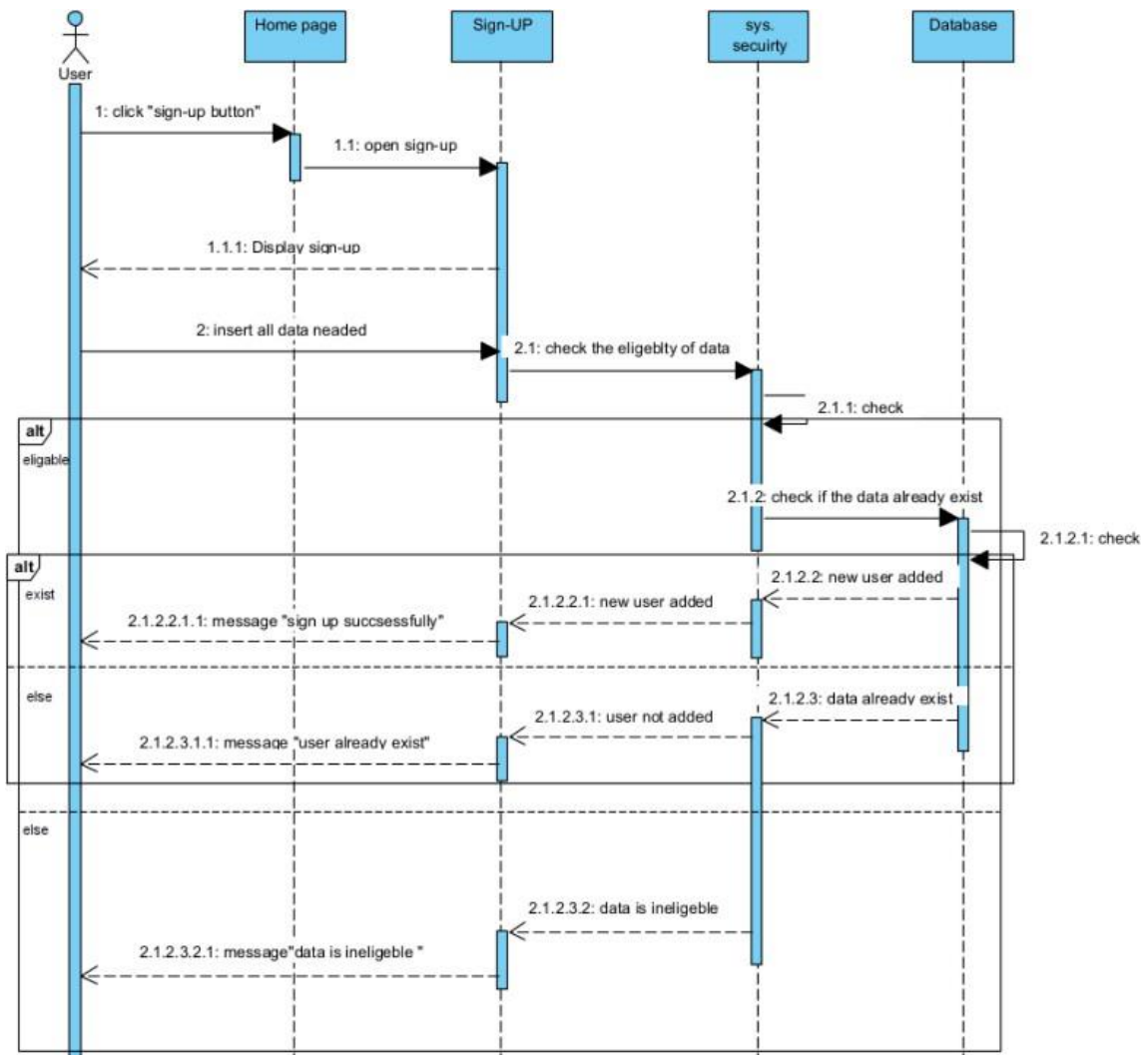
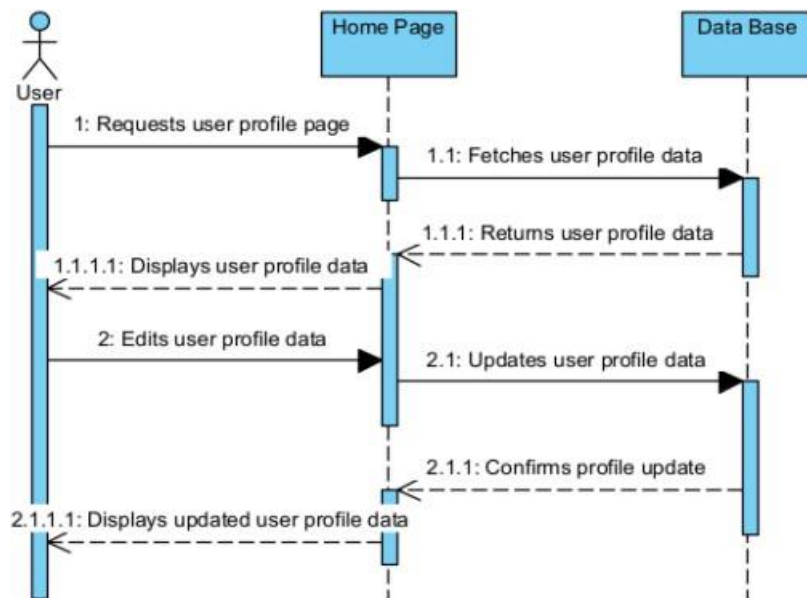1.3.1.2: display the like in post
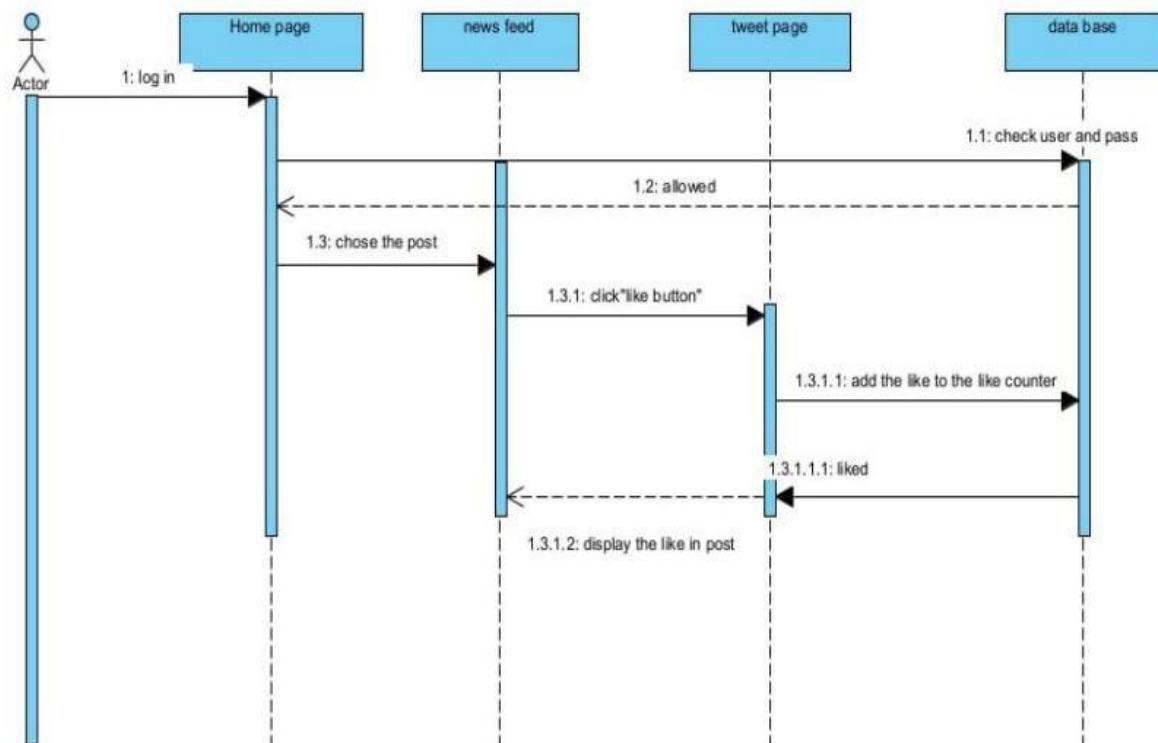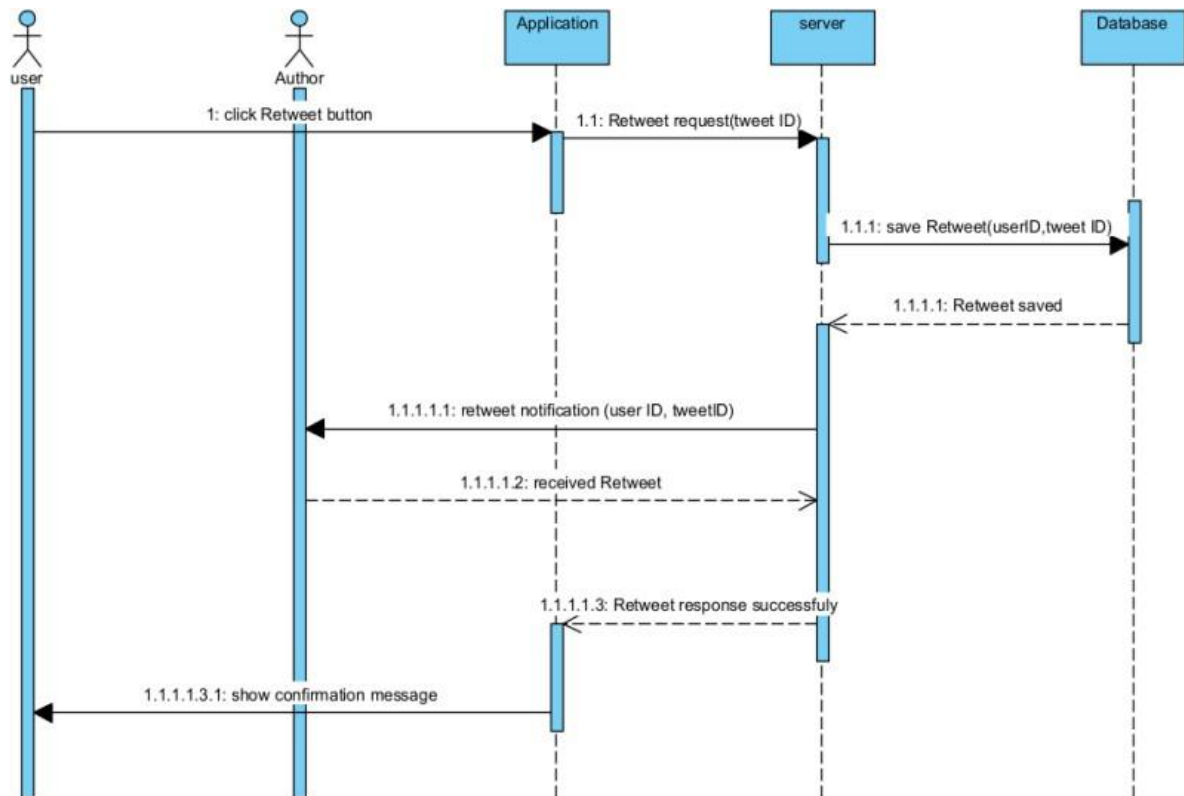
Undo like:
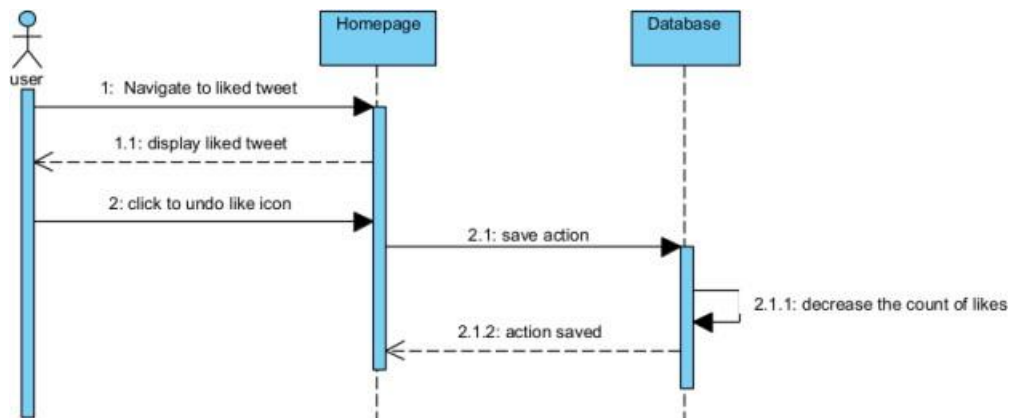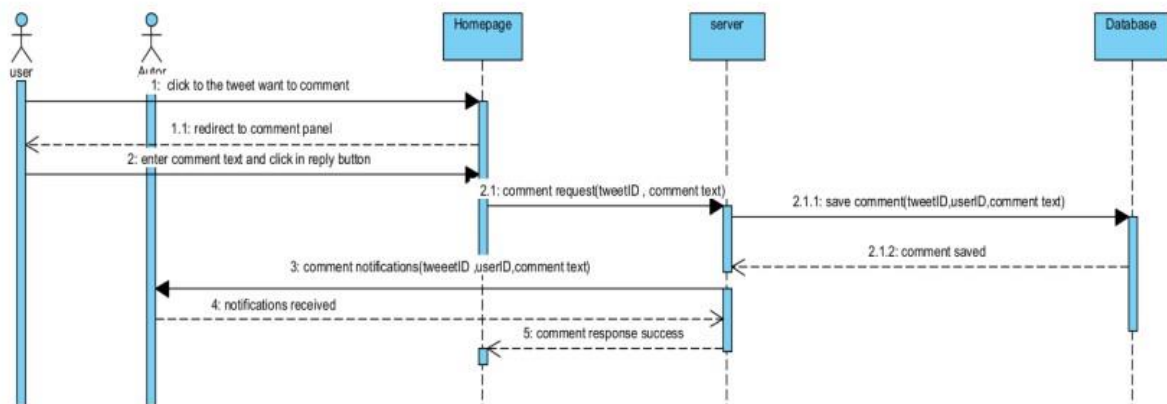
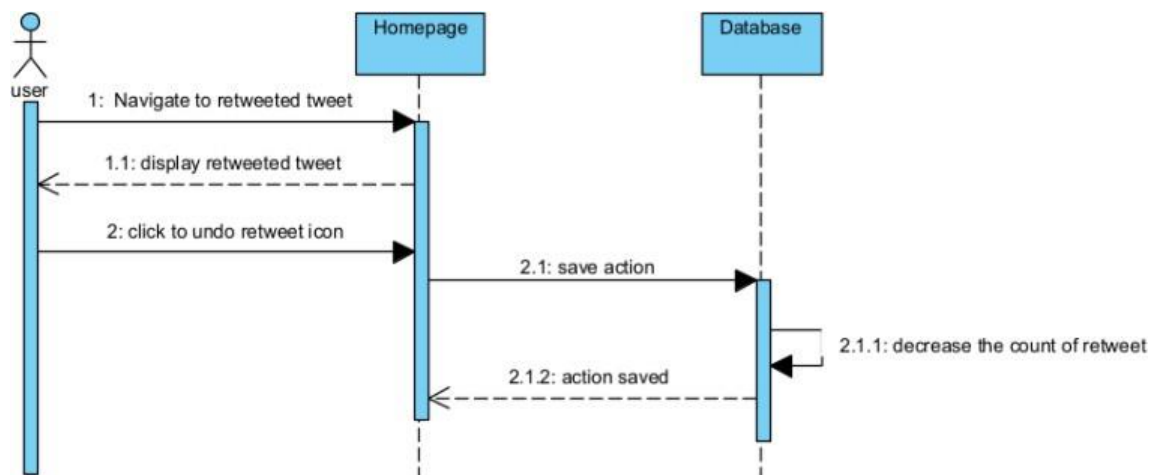
Comment:

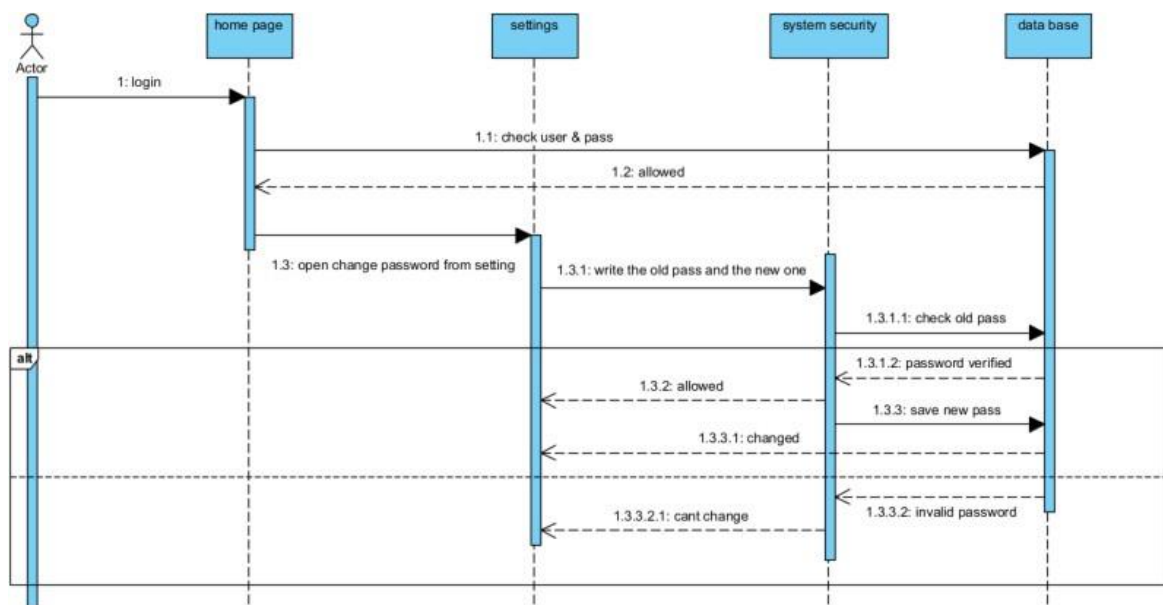# ReplyToComment:
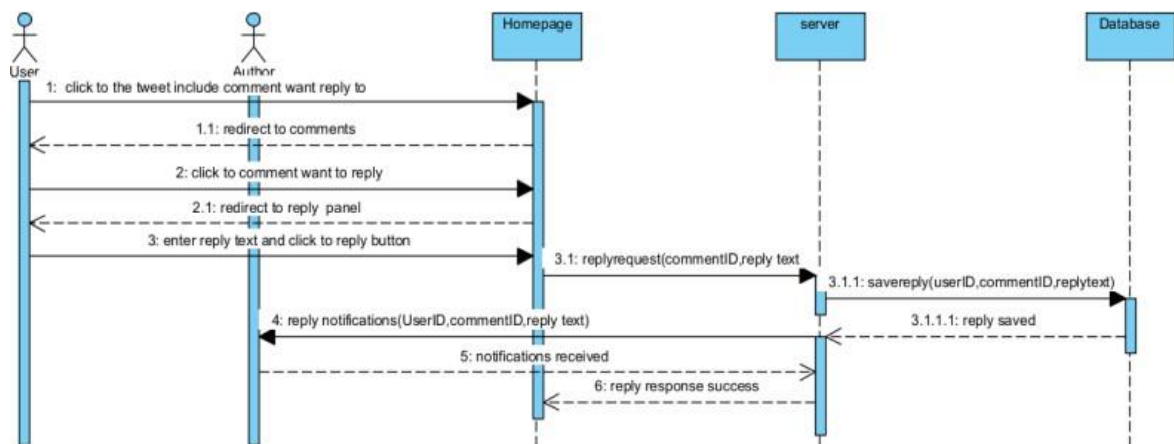
Bookmark:
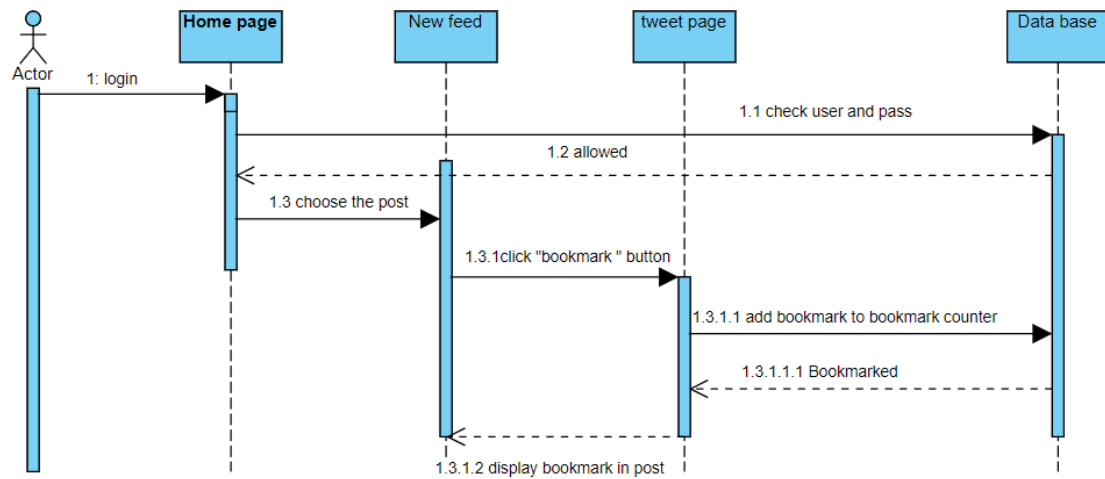
Undo Bookmark:

User Profile:

Quote tweet:

Sequence diagram showing interactions between user, Author, Application, server, and Database:

- 1: click quote tweet butoon (user → Application)
- 1.1: redirect to tweet panel (Application → user)
- 2: enter qoute text (user → Application)
- 2.1: quote request(tweet ID,quote text) (Application → server)
- 2.1.1: save quote tweet(User ID , tweet ID, quote text) (server → Database)
- 2.1.1.1: quote tweet saved (Database → server)
- 2.1.1.1.1: quote notification(User ID, tweet ID, quote text) (server → Application)
- 2.1.1.1.2: notification received (Application → server)
- 2.1.1.1.3: quote response success (server → Application)
- 2.1.1.1.3.1: show confirmation message (Application → user)
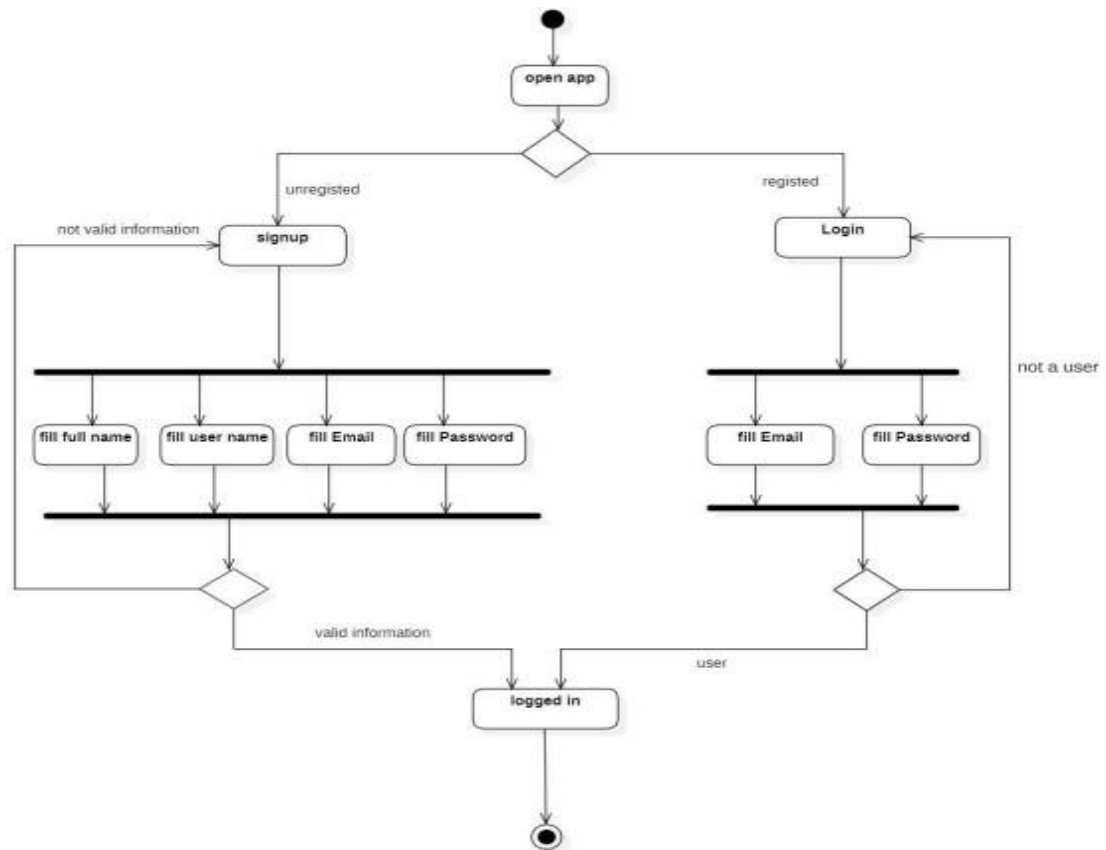
Retweet:

Undo Retweet:
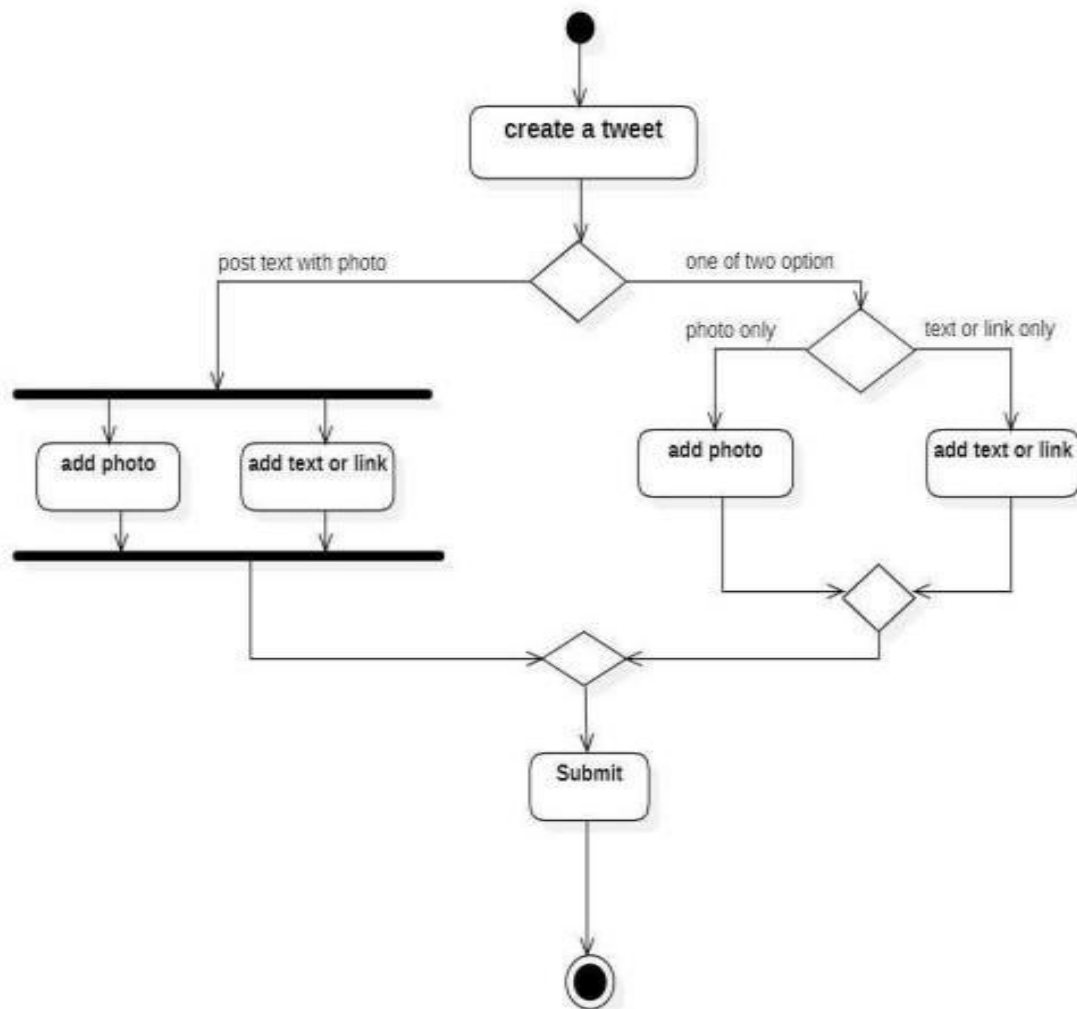
Change password:

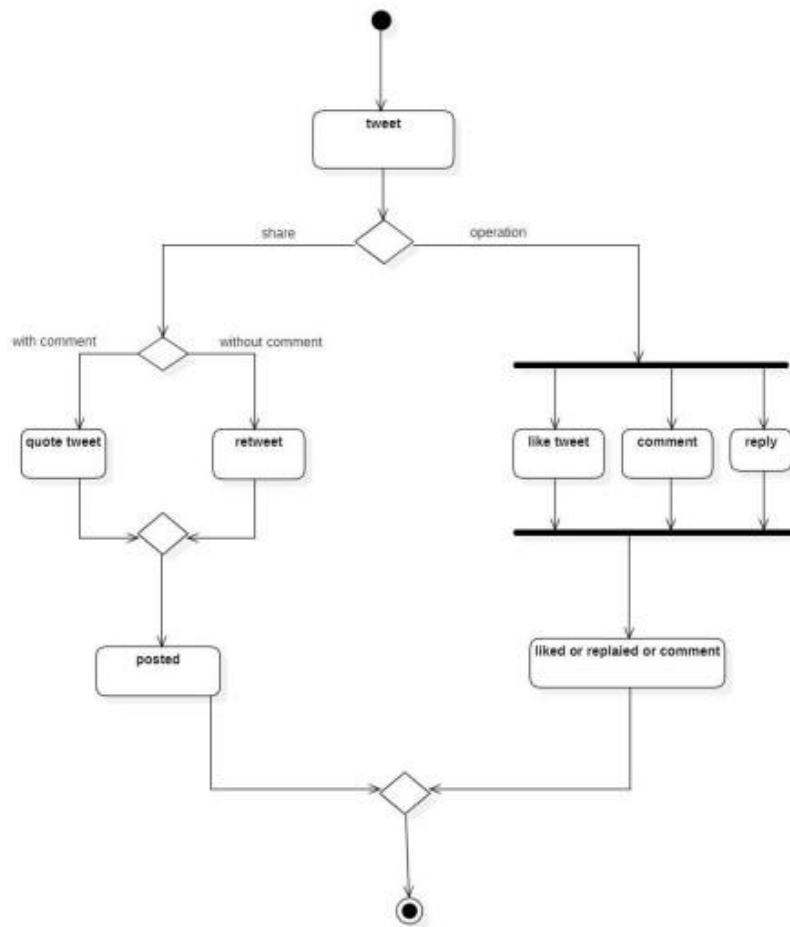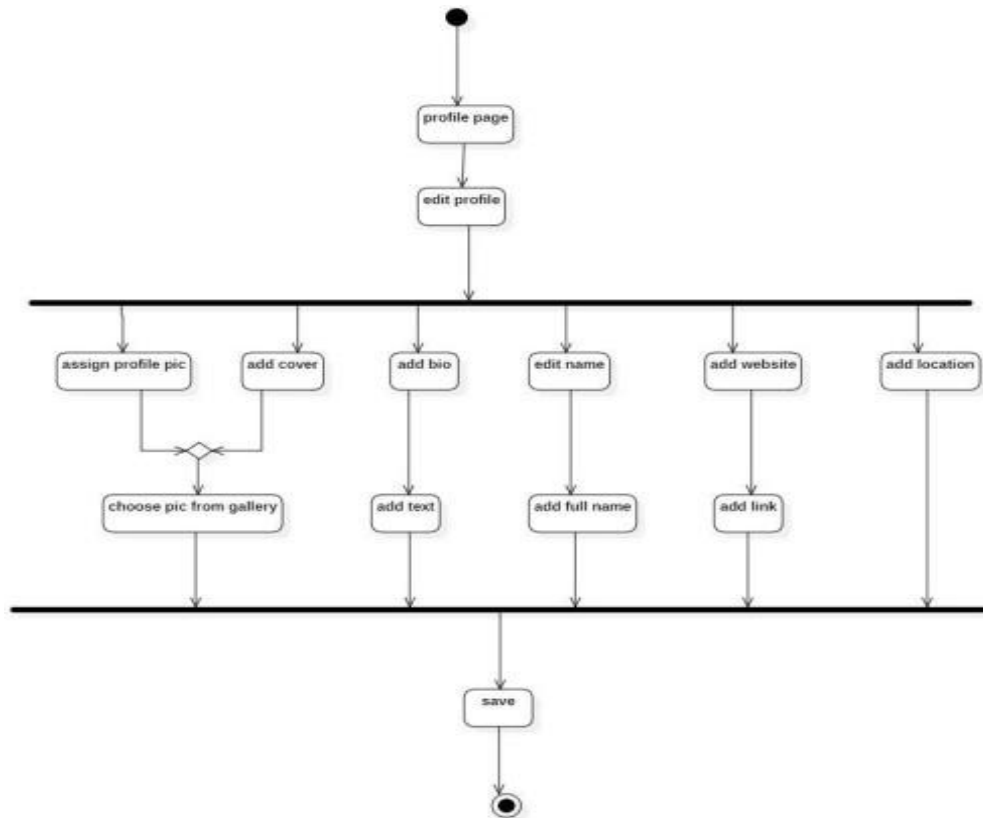**Activity Diagrams:**

Login&signup:

Create tweet:

Operations on a tweet:

Editing Profile:

Change password: