



Department of Computer Science and Engineering  
The Chinese University of Hong Kong

# Testing Document

LITTLE TWITTER – GROUP F1

1155157657 CHAN SIU CHUNG

1155109984 CHOW CHUN MAN

1155175870 CHENG TSZ KIN

1155158936 HO CHUN WAI

1155172012 HUI ROCKY

2023-05-06

## Contents

Test Overview.....	2
Test Plan .....	2
Frontend Tests.....	2
Frontend Test – Page Routing .....	2
Frontend Test – User Authentication - Register .....	4
Frontend Test – User Authentication – Login.....	5
Frontend Test – Post Tweet .....	5
Frontend Test – Like (Tweet, Retweet, comment) .....	7
Frontend Test – Bookmark (Tweet, Retweet, comment) .....	7
Frontend Test – Follow user.....	8
Frontend Test – Retweet.....	8
Frontend Test – Tweet Showing.....	9
Frontend Test – Searching .....	10
Frontend Test– Profile Page .....	10
Frontend Test – User Management Page.....	11
Frontend Test – Post a Comment.....	13
Frontend Test – Post a Nested Comment .....	13
Frontend API Test – Chatbox.....	14
Backend Tests .....	15
Backend API Test – Tweet and Comment APIs.....	15
Backend API Test – Chatbox.....	17
Backend API Test – Account Related APIs .....	17
Backend API Test – Admin Related APIs.....	20

# Test Overview

We have conducted black box testing on multiple features of the application, encompassing both backend APIs and frontend requirements. As of the report submission date, the application has successfully passed all test scenarios.

## Test Plan

The frontend features are organized into distinct use cases, and are tested manually by providing various inputs and verifying the expected outputs. For the backend APIs, Postman is utilized to directly send test requests to the server.

## Frontend Tests

### Frontend Test – Page Routing

#### Test case 1

Purpose	To check if guest user can access homepage
Inputs	Role: guest url=http://localhost:8123/
Expected Outputs	Homepage should be shown with limited feature
Pass/Fail Criteria	It is passed if expected outputs are returned

#### Test case 2

Purpose	To check if guest user can access the user protected page
Inputs	Role: guest url=http://localhost:8123/bookmark
Expected Outputs	Return to homepage immediately
Pass/Fail Criteria	It is passed if expected outputs are returned

#### Test case 3

Purpose	To check if guest user can access the admin protected page
Inputs	Role: guest url=http://localhost:8123/manage/users
Expected Outputs	Show error message
Pass/Fail Criteria	It is passed if expected outputs are returned

#### Test case 4

Purpose	To check if normal user can access homepage
Inputs	Role: normal user url=http://localhost:8123/
Expected Outputs	Homepage should be shown with all feature (except admin related)
Pass/Fail Criteria	It is passed if expected outputs are returned

#### Test case 5

Purpose	To check if normal user can access the user protected page
Inputs	Role: normal user url=http://localhost:8123/bookmark
Expected Outputs	Show the bookmark of the user
Pass/Fail Criteria	It is passed if expected outputs are returned

#### Test case 6

Purpose	To check if normal user can access the admin protected page
Inputs	Role: normal user url=http://localhost:8123/manage/users
Expected Outputs	Show error message
Pass/Fail Criteria	It is passed if expected outputs are returned

#### Test case 7

Purpose	To check if admin user can access homepage
Inputs	Role: admin url=http://localhost:8123/
Expected Outputs	Homepage should be shown with all

	feature
Pass/Fail Criteria	It is passed if expected outputs are returned

#### Test case 8

Purpose	To check if admin user can access the user protected page
Inputs	Role: admin url=http://localhost:8123/bookmark
Expected Outputs	Show the bookmark of the user
Pass/Fail Criteria	It is passed if expected outputs are returned

#### Test case 9

Purpose	To check if admin user can access the admin protected page
Inputs	Role: guest url=http://localhost:8123/manage/users
Expected Outputs	Show user management dashboard
Pass/Fail Criteria	It is passed if expected outputs are returned

### Frontend Test – User Authentication- Register

#### Test case 1

Purpose	To check if validation works
Inputs	Click on the footer to register without filling anything.
Expected Outputs	All inputs become red in colour and error message are shown below each input.
Pass/Fail Criteria	It is passed if the UI behaves exactly like the expected outputs.

#### Test case 2

Purpose	To check if the form submission works
Inputs	Fill with valid data: name: "testuser" tweetID: "unusedid", password: "1234abcd" password2: "1234abcd"
Expected Outputs	A network request should be sent to the

	API server. A success notification shows.
Pass/Fail Criteria	It is passed if expected outputs are achieved.

## Frontend Test – User Authentication – Login

### Test case 1

Purpose	To check if validation works
Inputs	Click on the footer to login without filling anything.
Expected Outputs	All inputs become red in colour and error message are shown below each input.
Pass/Fail Criteria	It is passed if the UI behaves exactly like the expected outputs.

### Test case 2

Purpose	To check if the form submission works
Inputs	Mock the database with user: tweetID: "hello" password: "1234abcd" and login with the same credentials.
Expected Outputs	The user should be re-directed to the profile page.
Pass/Fail Criteria	It is passed if expected outputs are achieved.

## Frontend Test – Post Tweet

### Test case 1

Purpose	To check if empty tweet can be posted
Inputs	Text: "" Image: null
Expected Outputs	Cannot post and message of requiring input is shown
Pass/Fail Criteria	It is passed if expected outputs are returned

### Test case 2

Purpose	To check if only text tweet can be
---------	------------------------------------

	posted
Inputs	Text: "test" Image: null
Expected Outputs	Tweet posted successfully
Pass/Fail Criteria	It is passed if expected outputs are returned

### Test case 3

Purpose	To check if text and image tweet can be posted
Inputs	Text: "test" Image: img.jpg
Expected Outputs	Tweet posted successfully
Pass/Fail Criteria	It is passed if expected outputs are returned

### Test case 4

Purpose	To check if only image tweet can be posted
Inputs	Text: "" Image: img.jpg
Expected Outputs	Cannot post and message of requiring input is shown
Pass/Fail Criteria	It is passed if expected outputs are returned

### Test Case 5

Purpose	To check whether a user can create and post a text-only tweet successfully.
Inputs	Content = "Hello World"
Expected Outputs	The user should see a new tweet with correct content on his/ her home page.
Pass/ Fail Criteria	It is pass if the new tweet exists on home page

### Test Case 6

Purpose	To check whether a user can create and post a image-only tweet successfully.
Inputs	Content = "" Images = "okinawa.jpg"
Expected Outputs	The user should see a new tweet with correct content on his/ her home page.
Pass/ Fail Criteria	It is pass if the new tweet exists on home page

#### Test Case 7

Purpose	To check whether a user can create and post a tweet contains both text and image successfully.
Inputs	Content = "This is the coolest creatures in the world" Images = "whaleshark.jpeg"
Expected Outputs	The user should see a new tweet with correct content on his/ her home page.
Pass/ Fail Criteria	It is pass if the new tweet exists on home page

#### Frontend Test – Like (Tweet, Retweet, comment)

##### Test case 1

Purpose	To check if user can like
Inputs	Like a tweet that has not been liked
Expected Outputs	Like successfully, like icon border turn to red
Pass/Fail Criteria	It is passed if expected outputs are returned

##### Test case 2

Purpose	To check if user can get back the like
Inputs	Like a tweet that has been liked
Expected Outputs	Get back like successfully, like icon border turn to grey
Pass/Fail Criteria	It is passed if expected outputs are returned

#### Frontend Test – Bookmark (Tweet, Retweet, comment)

##### Test case 1

Purpose	To check if user can bookmark
Inputs	Bookmark a tweet that has not been bookmarked
Expected Outputs	Bookmark successfully, bookmark button changes to unbookmark. That tweet is shown in bookmark page.
Pass/Fail Criteria	It is passed if expected outputs are returned



### Test case 2

Purpose	To check if user can unbookmark
Inputs	Unbookmark a tweet that has been bookmarked
Expected Outputs	Unbookmark successfully, unbookmark button changes to bookmark. That tweet is removed from bookmark page.
Pass/Fail Criteria	It is passed if expected outputs are returned

### Frontend Test – Follow user

#### Test case 1

Purpose	To check if user follow a user
Inputs	Follow a user that has not been followed
Expected Outputs	Follow successfully, follow button changes to unfollow. The tweet related to that user will be shown in homepage.
Pass/Fail Criteria	It is passed if expected outputs are returned

#### Test case 2

Purpose	To check if user can unfollow a user
Inputs	Unfollow a user that has been followed
Expected Outputs	Unfollow successfully, unfollow button changes to follow. The tweet related to that user will not be shown in homepage.
Pass/Fail Criteria	It is passed if expected outputs are returned

### Frontend Test – Retweet

#### Test case 1

Purpose	To check if user can retweet
Inputs	Retweet a tweet that has not been retweeted
Expected Outputs	Retweet successfully, retweet icon border turn to green
Pass/Fail Criteria	It is passed if expected outputs are

	returned
--	----------

#### Test case 2

Purpose	To check user can get back the retweet or retweet again
Inputs	Retweet a tweet that has been retweeted
Expected Outputs	Nothing happened
Pass/Fail Criteria	It is passed if expected outputs are returned

#### Test Case 3

Purpose	To check whether a user can retweet a tweet successfully.
Inputs	CorrTweetID = 1011
Expected Outputs	The user should see a retweet in home page with correspond tweet's content.
Pass/ Fail Criteria	It is pass if the retweet exists in home page with retweet indicator.

#### Test Case 4

Purpose	To check whether a user can retweet a tweet successfully.
Inputs	CorrTweetID = 1
Expected Outputs	The user should see a retweet in home page with correspond tweet's content.
Pass/ Fail Criteria	It is pass if the retweet exists in home page with retweet indicator.

### Frontend Test – Tweet Showing

#### Test case 1

Purpose	To check if tweet can show and order by recommendation
Inputs	In <a href="http://localhost:8123/explore">http://localhost:8123/explore</a> , click Recommend tab
Expected Outputs	The tweets is order by the recommend order
Pass/Fail Criteria	It is passed if expected outputs are returned

#### Test case 2

Purpose	To check if tweet can show and order by Like
Inputs	In <a href="http://localhost:8123/explore">http://localhost:8123/explore</a> , click Popular tab
Expected Outputs	The tweets is order by the Like
Pass/Fail Criteria	It is passed if expected outputs are returned

### Test case 3

Purpose	To check if tweet can show and order by created time
Inputs	In <a href="http://localhost:8123/explore">http://localhost:8123/explore</a> , click Recent tab
Expected Outputs	The tweets is order by the created time
Pass/Fail Criteria	It is passed if expected outputs are returned

## Frontend Test – Searching

### Test case 1

Purpose	To check if empty search treats as not searching
Inputs	Search: ""
Expected Outputs	Show all the tweet or people (select suitable tab) by the recommendation (not followed)
Pass/Fail Criteria	It is passed if expected outputs are returned

### Test case 2

Purpose	To check if searching work (including partial searching)
Inputs	Search: "test"
Expected Outputs	Show all the tweet or people (select suitable tab) contain "test"
Pass/Fail Criteria	It is passed if expected outputs are returned

## Frontend Test– Profile Page

### Test Case 1

Purpose	To check if the profile works with the currently logged in user
Inputs	Logged as user with tweetID "TestingID" and navigate to /profile
Expected Outputs	The page loads. Shows the name, tweetID, and other personal information. Also show an edit button.
Pass/Fail Criteria	It is passed if expected outputs are presented.

#### Test Case 2

Purpose	To check if other's profile can be loaded
Inputs	Assume the user with tweetID "New" exists (by mocking). Navigate to /profile/New
Expected Outputs	The page loads. Shows the personal information of user "New". Do not show the edit button.
Pass/Fail Criteria	It is passed if expected outputs are presented.

#### Test Case 3

Purpose	To check if error messages are shown properly for not logged in users.
Inputs	Logout then navigate to /profile
Expected Outputs	Shows a error message including the string "not logged in"
Pass/Fail Criteria	It is passed if expected outputs aer presented.

#### Test Case 4

Purpose	To check if other's profile still viewable without logging in
Inputs	Logout and assume the user with tweetID "New" exists (by mocking). Navigate to /profile/New.
Expected Outputs	Same as Test Case 2.
Pass/Fail Criteria	It is passed if expected outputs are presented.

#### Test Case 1

Purpose	To check whether the users' table loads correctly.
Inputs	Assume there are two users in the database (by mocking). Log in to a admin account and navigate to /manage/users.
Expected Outputs	A user table with two users shown
Pass/Fail Criteria	It is passed if expected outputs are presented.

#### Test Case 2

Purpose	To check if search users function properly.
Inputs	Prepare two users with name "abcde" and "efghk". Log in to admin account and navigate to /manage/users. Focus on the search bar and type "z"
Expected Outputs	No users should be shown.
Pass/Fail Criteria	It is passed if expected outputs are returned

#### Test Case 3

Purpose	To check if search users function properly.
Inputs	Prepare two users with name "abcde" and "efghk". Log in to admin account and navigate to /manage/users. Focus on the search bar and type "a"
Expected Outputs	Only the user "abcde" should be shown.
Pass/Fail Criteria	It is passed if expected outputs are returned

#### Test Case 4

Purpose	To check if search users function properly.
Inputs	Prepare two users with name "abcde" and "efghk". Log in to admin account and navigate to /manage/users. Focus on the search bar and type "z" and remove the "z"
Expected Outputs	First it should no users but upon the removal of "z", the two users should be shown again.
Pass/Fail Criteria	It is passed if expected outputs are returned

## Frontend Test – Post a Comment

### Test Case 1

Purpose	To check whether a user can create and post a comment successfully.
Inputs	CorrTweetID = 1011 Content = “what a pretty miracle”
Expected Outputs	The user should see a comment under the Tweet’s (id:1011) page with correct content.
Pass/ Fail Criteria	It is pass if the new comment exists in tweet’s page (id:1011)

### Test Case 2

Purpose	To check whether a user can create and post a comment successfully.
Inputs	CorrTweetID = 1011 Content = “You are right”
Expected Outputs	The user should see comments under the Tweet’s (id:1011) page with correct content.
Pass/ Fail Criteria	It is pass if the new comment exists in tweet’s page (id:1011)

## Frontend Test – Post a Nested Comment

### Test Case 1

Purpose	To check whether a user can create and post a comment to a comment successfully.
Inputs	CorrTweetID = 1011 CorrCommentID = 51 Content = “yea this is cool”
Expected Outputs	The user should see a comment under the Comment’s (id:1011) page with correct content.
Pass/ Fail Criteria	It is pass if the new comment exists in tweet’s page (id:1011)

### Test Case 2

Purpose	To check whether a user can create and post a comment to a comment successfully.
---------	--

Inputs	CorrTweetID = 1011 CorrCommentID = 51 Content = "You are right"
Expected Outputs	The user should see a comment under the Comment's (id:1011) page with correct content.
Pass/ Fail Criteria	It is pass if the new comment exists in tweet's page (id:1011)

## Frontend API Test – Chatbox

### Test Case 1

Purpose	To check whether User can search another user via search bar
Inputs	Other User ID
Expected Outputs	Related UserID with its avastar is displayed
Pass/Fail Criteria	It is passed if expected outputs are returned

### Test Case 2

Purpose	To check whether User can create the private chat box with another users
Inputs	User kick the target User under search result
Expected Outputs	(1)Specific chat box is created (2)the chat box is displayed
Pass/Fail Criteria	It is passed if expected outputs are returned

### Test Case 3

Purpose	To check whether User can send the message to the chatbox
Inputs	Message to be used
Expected Outputs	The message should immediately shown in the chatbox
Pass/Fail Criteria	It is passed if expected outputs are returned

# Backend Tests

## Backend API Test – Tweet and Comment APIs

### Test Case 1

Purpose	To check whether Creating Tweet function create tweets successfully at POST localhost:8000/home
Inputs	Form Data with Content as Key to contain the tweet contents
Expected Outputs	Status 200 OK with output as follow: { "message" : "success, posted tweet" }
Pass/Fail Criteria	It is passed if expected outputs are returned

### Test Case 2

Purpose	To check whether Like (Dislike) Tweet function create tweets successfully at PATCH localhost:8000/home/(dislike)likeTweet/:TweetID
Inputs	Target TweetID as input n URL
Expected Outputs	Status 200 OK with output as follow: { "state" : "Success", "LikeCount" : 5, "DisLikeCount" : 0 }
Pass/Fail Criteria	It is passed if expected outputs are returned

### Test Case 3

Purpose	To check whether Editing Tweet function create tweets successfully at PATCH localhost:8000/home/editTweet/:TweetID
Inputs	(1)Target TweetID as input in URL (2)Edited Content in form Data
Expected Outputs	Status 200 OK with output as follow: { "state" : "Success" }
Pass/Fail Criteria	It is passed if expected outputs are returned



#### Test Case 4

Purpose	To check whether Retweeting Tweet function create tweets successfully at POST localhost:8000/home/retweet/:TweetID
Inputs	(1)Target TweetID as input in URL
Expected Outputs	Status 200 OK with output as follow: { "message" : "success, posted retweet" }
Pass/Fail Criteria	It is passed if expected outputs are returned

#### Test Case 5

Purpose	To check whether Fetching Tweet function create tweets successfully at GET localhost:8000/home/fetchHomeTweet
Inputs	Target TweetID as input in URL
Expected Outputs	Status 200 OK with output as follow: { "message" : List of Tweet related information } Example for one Tweet related information: <pre>"TweetID": "1", "CreatorUserID": "ansontest", "CreatorUserName": "ansoni", "CreatorAvastar": "uploads\\avatar-1681435414727.jpg", "CreateTime": "2023-04-01T10:18:43.268Z", "Content": "This is our First Tweet!!!!!!", "LikeCount": 1, "DisLikeCount": 0, "ReTweetCount": 1, "UrlList": [], "ImageList": [], "CommentCount": 9</pre>
Pass/Fail Criteria	It is passed if expected outputs are returned

#### Test Case 6

Purpose	To check whether Commenting Tweet function create tweets successfully at POST localhost:8000/home/comment/:Tweetid
Inputs	(1)Target TweetID as input in URL (2)Comment Content as form data
Expected Outputs	Status 200 OK with output as follow:

	{ "message" : "success, posted comment" }
Pass/Fail Criteria	It is passed if expected outputs are returned

## Backend API Test – Chatbox

### Test Case 1

Purpose	To check whether Creating Chatbox function create chatbox successfully at POST localhost:8000/conservation/create/:firstUserID/:secondUserID
Inputs	Current User ID and Target User ID as input in URL
Expected Outputs	Status 200 OK with output as follow: { Created Conservation items }
Pass/Fail Criteria	It is passed if expected outputs are returned

### Test Case 2

Purpose	To check whether Sending Message function create tweets successfully at POST localhost:8000/message/
Inputs	Form Data as input with the following items: (1)ChatboxID (2)SenderUserID (3)MessageContent
Expected Outputs	Status 200 OK with output as follow: { Saved Conservation items }
Pass/Fail Criteria	It is passed if expected outputs are returned

## Backend API Test – Account Related APIs

### Test Case 1

Purpose	To check whether the register function is work successfully at POST localhost:8000/registration
Inputs	tweetID, name, email, password

Expected Outputs	{ message: "account created" }
Pass/Fail Criteria	It is passed if expected outputs are returned

#### Test Case 2

Purpose	To check whether the register function is work successfully with user entering a duplicated email or tweetID at POST localhost:8000/registration
Inputs	tweetID that already in database, name, email that already in database, password
Expected Outputs	{ error: "Account already exist" }
Pass/Fail Criteria	It is passed if expected outputs are returned

#### Test Case 3

Purpose	To check whether the login function is work successfully at POST localhost:8000/login
Inputs	email, password
Expected Outputs	{ state: "Success", data: userObject }
Pass/Fail Criteria	It is passed if expected outputs are returned

#### Test Case 4

Purpose	To check whether the register function is work successfully with user entering a wrong email, tweetID or password at POST localhost:8000/login
Inputs	Wrong email, wrong tweetID and wrong password
Expected Outputs	{ error: "password doesn't match" }
Pass/Fail Criteria	It is passed if expected outputs are returned

#### Test Case 5

Purpose	To check whether the SearchUserById function is work successfully with user entering a tweetID at GET localhost:8000/search/SearchUserById/:id
Inputs	userid
Expected Outputs	{ data: user, state: "Success" }
Pass/Fail Criteria	It is passed if expected outputs are returned

#### Test Case 6

Purpose	To check whether the UserRecommendation function is work successfully at GET localhost:8000/search/UserRecommendtion
Inputs	-
Expected Outputs	{ User_Recommended }
Pass/Fail Criteria	It is passed if expected outputs are returned

## Backend API Test – Admin Related APIs

### Test Case 1

Purpose	To check whether the AdminDeleteComment function is work successfully at DELETE localhost:8000/dashboard/AdminDeleteComment/:userID
Inputs	userID
Expected Outputs	{ state: "Success" }
Pass/Fail Criteria	It is passed if expected outputs are returned

### Test Case 2

Purpose	To check whether the AdminDeleteTweet function is work successfully at DELETE localhost:8000/dashboard/AdminDeleteTweet/:userID
Inputs	userID
Expected Outputs	{ state: "Success" }
Pass/Fail Criteria	It is passed if expected outputs are returned

### Test Case 3

Purpose	To check whether the ListAllUser function is work successfully at GET localhost:8000/dashboard
Inputs	-
Expected Outputs	{ data: userObject state: "Success" }
Pass/Fail Criteria	It is passed if expected outputs are returned

### Test Case 4

Purpose	To check whether the DeleteUser function is work successfully at DELETE localhost:8000/dashboard/:tweetID
Inputs	tweetID
Expected Outputs	{ data: userObject

	state: "Success" }
Pass/Fail Criteria	It is passed if expected outputs are returned