Table of Contents

Bracket 8 2

Account function 2

Get User () 2

Save username () 2

Tweet key () 2

Current user () 3

Update profile () 3

Get following flag () 3

Change follow status () 3

Get tweets by username () 3

Get user by username () 3

Get tweets for current user () 3

Get tweet by id () 4

Delete tweet () 4

Search tweet () 4

Format time () 4

Get tweet by key () 4

Edit profile 4

Get () 4

Post () 4

Get template values () 4

Home page 5

Get () 5

Post () 5

Get template values () 5

Profile page 5

Get () 5

Post () 5

Template value () 5

Search user () 5
Save tweet () 5
Update tweet () 6
Show profile 6
Get () 6
Post () 6
Template value () 6
Bracket 9 6
User model 6
Tweet model 6
Bracket 10 7
UI consideration 7
Colors 7
Forms 7
Messages 7

Bracket 8

Account function

Get User ()

- It will get current user and check whether the user exists or not in the data store.
    - If the user exist then allows the user to login.
- If the user does not exist then it will add a record to the Data store and redirected to account.
- In this method it will send the login URL, login text and current user.

Save username ()

- It will get the username from request.
- Then it will get the email from current user and check whether it exists in the data store or not.
- Username will be updated in the data store for the current user.

Tweet key ()

- It will get the username from the request.
- It will get the current time stamp.
- It will return the tweet key by combining username and time stamp.

Current user ()

- It will get the email from the current user.
- It will check against data store user entity.
- If match found user account will be returned.

Update profile ()

- It will get request object from the update form submission.
- It will get the username and bio from the request.
- If the bio is greater than 280 characters then it will shows an error.
- If there is no error found then the profile details will be updated.
- Success message will be returned.

Get following flag ()

- It will get the current user.
- It will check whether the following user is in the following list or not.
- If the user is in the following list we can unfollow.
- If not we can follow.

Change follow status ()

- Get the current user.
- Then the user that we are trying to follow.
- It will get the follow status of the user.
- It will add the followers list to the current user following list.
- It will update the other user followers. List with current user, username.
- After unfollowing the other user, the current user and the other user profiles also updated.

Get tweets by username ()

- Get the tweets by combining current user and his following list.
- Then order by date and time.
- If tweets not found an error will be return, otherwise list of tweets will be return.

Get user by username ()

- It will get the username and check it with tweet entity in data store.
- It will be ordered by date and time.
- If it is found, the records will be returned. Otherwise error will be returned.

Get tweets for current user ()

- It will get the current user.
- Current user and username will be appended to following list.
- Then it will check the tweets entity by combining current user and his following list.
- It will be ordered by date and time.
- Last 50 records will be fetched.
- If fetch found result will be returned.

- Otherwise false will be returned.

Get tweet by id ()

- It will accept the tweet id and will generate the key by combining username.
- It will check in the data store in the tweet entity.
- If match found tweet will be returned.
- Otherwise false will be returned.

Delete tweet ()

- It will accept the tweet id.
- It will combine the username and generate tweet key.
- Then check it with data store.
- If match found the tweet will be deleted.
- And the user entity counts of tweets also deducted by 1.

Search tweet ()

- It will accept the tweet search parameter.
- It will trim the white spaces.
- If the search parameter is empty error message will be returned.
- Tweet will be search in the full text search.
- If matches found, it will check against the tweet entity.
- And matching results will be returned, otherwise error will be returned.

Format time ()

- It will accept the time stamp.
- Format it to human readable format.

Get tweet by key ()

- It will accept the tweet id.
- It will check with the tweet entity.
- If match found tweet will be returned.

Edit profile

Get ()

- It will accept the request and shows the edit profile page.

Post ()

- It will accept the request.
- It will find the users with following count and unfollow counts.
- Template object will be returned.

Get template values ()

- It will accept the request.

- Then it will find the users with following count and unfollow count.
- Template object will be returned.

Home page

Get ()

- It will accept the get request and shows the home page.

Post ()

- It will accept the post request and will call the save username form to update username.
- If the username is already updated, user will be returned to profile page.

Get template values ()

- It will get the current user and return the template value object.

Profile page

Get ()

- It will edit the tweet details by matching edit tweet parameter.
- It will also call the delete tweet function by matching delete tweet parameter.
- It will show the existing tweet details by tweet id.

Post ()

- It will match the form name by button value.
- If the button value matches with search user it will call search user functions and shows user details.
- If the button value matches with save tweet it will call the save tweet function.
- If the button matches with update tweet, then update tweet function will be called.
- If the button matches with the search tweet, it will shows matched tweets.

Template value ()

- It will load the current user details including following and followers count.
- It will load current users tweet and send those values to get, post methods.

Search user ()

- It will get the username.
- If the user is empty then error will be shown.
- It will check the username in the data store whether it exists or not.
- If exists it will return otherwise an error will be returned.

Save tweet ()

- It will accept the tweet text and check whether the text is empty or not. If it is empty error will be shown.
- If image is selected it checks whether the image is JPEG or PNG.

- If file types doesn't match error will be shown in the data store.
- User entity will be updated.
- Document will be added to full text search.

Update tweet ()

- It will check whether the text is empty or not.
- It will update the existing tweet.
- Full text index search will be deleted and new record will be added.

Show profile

Get ()

- Here other user profiles will be shown.
- If the other user is same user that already logged in it will redirected to profile page.

Post ()

- It will call the change followers status function.
- It will show the show profile page.

Template value ()

- It will return the tweets, followers and following counts to the get and post methods.

Bracket 9

Two models have been generated they are user model and tweet model. Both have entities in the Data store.

User model

- It stores the user information.
- In this model, email id is unique for all the users so, the email that we entered is considered as a key for all entities and it will be added to the string property.
- In this model bio, first name and last name are added and taken as string property as they contain letters and words.
- Followers and Following are considered as String Properties

Tweet model

- It stores the tweet records for every user.
- For user and tweet entity the tweet key should be unique and unique for every entry. The key will be generated by combing user email and current timestamp.
- Text field considered as a string property because it stores the tweet.
- Tweet image is considered as a blob key property. Key will be generated after saving the image in blob store.
- Tweet added time considered as a date time property with auto added feature. This will shows the tweets by date and time.

- We can search the tweets by user by using user name and email.
- User name and user email considered as string property.

Bracket 10

UI consideration

    - At the top of every page the main navigation has been located in the header.
- Forms
    - Labels have been placed beside each other for every HTML component.
- Tweet, followers and following counts are aligned at the top of a page.
- In the right side of the menu tweet search and user was shown and can be navigated to all other pages.
- Name, username, posted date, image and tweet is shown beside each other and considered as one group.
- Save tweet and tweet list are shown beside each other. User can see it in the timeline when a new tweet is added.
- Tweet saves and delete are displayed in the timeline.

Colors

- Color scheme was chosen to make the site look nice. To give high contrast for visually impaired.

Forms

- Each structure has a heading at the top so the reason behind the form is clear.
- In every html structure part, marks have been shown near one another. Likewise, all the textboxes contain a placeholder for further distinguishing proof of the particular field.

Messages

- Green color was used to show the success message.
- Red color was used to show the warning messages for invalid data.

.