

Team: hus206 , ung200
Name: Hitarthi Shah, Udita Gupta

NYU Tandon School of Engineering
Computer Science and Engineering
CS 6083 INET, Spring 2018

REPORT : Project Part 2

[Introduction](#)

[Relational Schema](#)

[Website based User Interface](#)

[Signup and login page](#)

[Edit Profile](#)

[Add Friend](#)

[Pending Request](#)

[Accept Request](#)

[Decline Request](#)

[Activity Post](#)

[Diary Entry](#)

[Location](#)

[Like and Dislike](#)

[Privileges](#)

[Search](#)

[User Interface](#)

[User Session](#)

[Features](#)

[Timeline](#)

Introduction

- We have chosen University Students Social Network as our Application Scenario.
- In project part 2, we are expanding the work that we did in part 1 by creating web based user interface for the database.
- For the backend we have created and maintained our database in **PHPMyadmin**.
- We have implemented all the feature that were mentioned and have added some extra functionality to the system which we will mention further in the report.

Relational Schema:

Primary Keys are shown in bold

1. Student(username, **student_id**, password, name, age, gender, city, group_id_list
friends_list, latest_login_timestamp, latest_logout_timestamp, liked_location_list
email, bio, story, profile_pic, telephone)

- **Assumption:**

liked_location_list column: all the locations that the user has only liked will be added.

Group_id_list: all the groups ids of which the student is a member of

Friends_list: a list of all friends of the student.

- **New addition to the table:**

We are taking additional details from user to so he can edit/update his profile if he wants to. Also we have set student id to be auto incremented.

Story	User can add few lines to share his mood, hobbies, and status.
Bio	User can add his introduction in few lines such as from which class or batch he belongs.
Telephone	User will be able to add his phone number.
Profile pic	User can add a display picture.

2. Diary (student_id, **diary_id** , title, diary_timestamp, body_text, body_multimedia, privacy_control_id, location_id, like_counter, dislike_counter)

Foreign Key: student_id, privacy_control_id, location_id

- **New addition to the table:**

We have now added dislike counter to keep track of the user's dislike on his or other's post.

We also made each diary id to auto increment so that we can record unique entries in that database.

3. Diary_comment (student_id, diary_id, comment_id , comment_body, comment_timestamp)

Foreign Key: student_id, diary_id

4. Location (location_id , location_name, latitude, longitude, like_counter, dislike counter)

- **New addition to the table:**

As mentioned above, we are keeping tracks of dislikes with the like counters.

5. Friends (student_id_1, student_id_2, friendship_timestamp, friendship_status)

Foreign Key: student_id_1, student_id_2 references Student (student_id)

Assumption : When user1 sends friend request to user2 then user1 id will go in student_id_1 and user2 id will go in student_id_2.

6. Student_activity (from_student_id, to_diary_id, activity_id , activity_timestamp, activity_type, privacy_control_id)

Foreign Key: from_student_id, to_diary_id, privacy_control_id

Assumption:

from_student_id is the student_id of the student who will be liking or commenting a post of his own or his friends, or his friends of friends etc.

To_diary_id : is the diary_id on which the activity was performed.

- **New addition to the table:**

activity_id is made auto incremented.

7. Groups (group_id , group_name, student_id, group_member_list, group_creation_timestamp, privacy_control_id)

Foreign Key: student_id, privacy_control_id

Assumption:

student_id is the id of the group admin.

8. Student_logs (student_id, student_login_timestamp, student_logout_timestamp, location_id)

Foreign Key: student_id, location_id

9. Privacy_control (privacy_control_id , privacy_control_type)

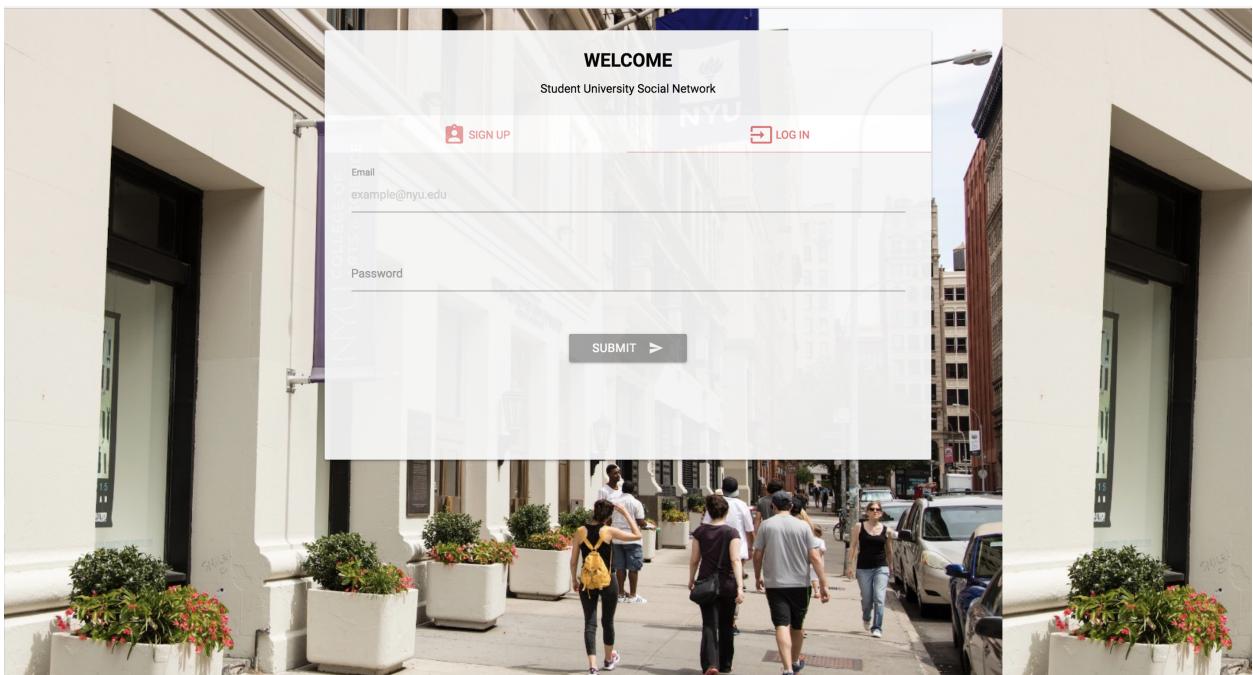
Web App Design

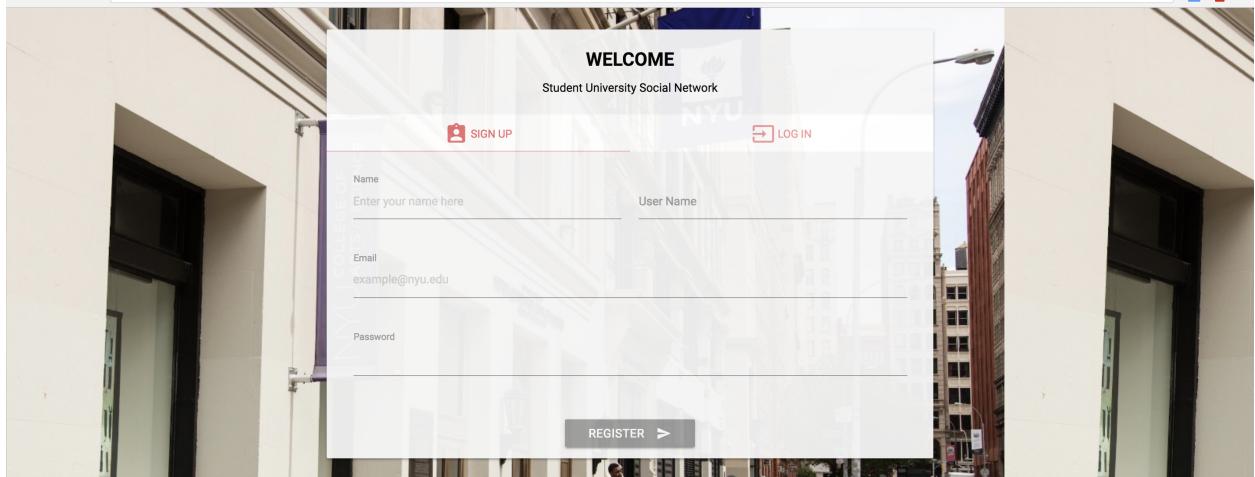
Signup and login page

- **Signup** - Every new user will first have to sign up to the system to gain access for website.
- User will create an account for this by providing his name, username, email id and password.
- We have taken email id as primary key so that when the new user will try to make the account, we will check if some user with the same user email exists in the database or not. If exist, student will be denied to make account with that email id.
- We are also checking if the user enters valid email id or not by checking the format of the email.

That is, we will check for the '@' sign and '.com' in the email to validate it.

- **Login** - Once, the user has signed up, we will make him login. This will be mandatory step for each user to enter into the system.
- If user already has an account, he can directly log in and access the website.





Edit Profile

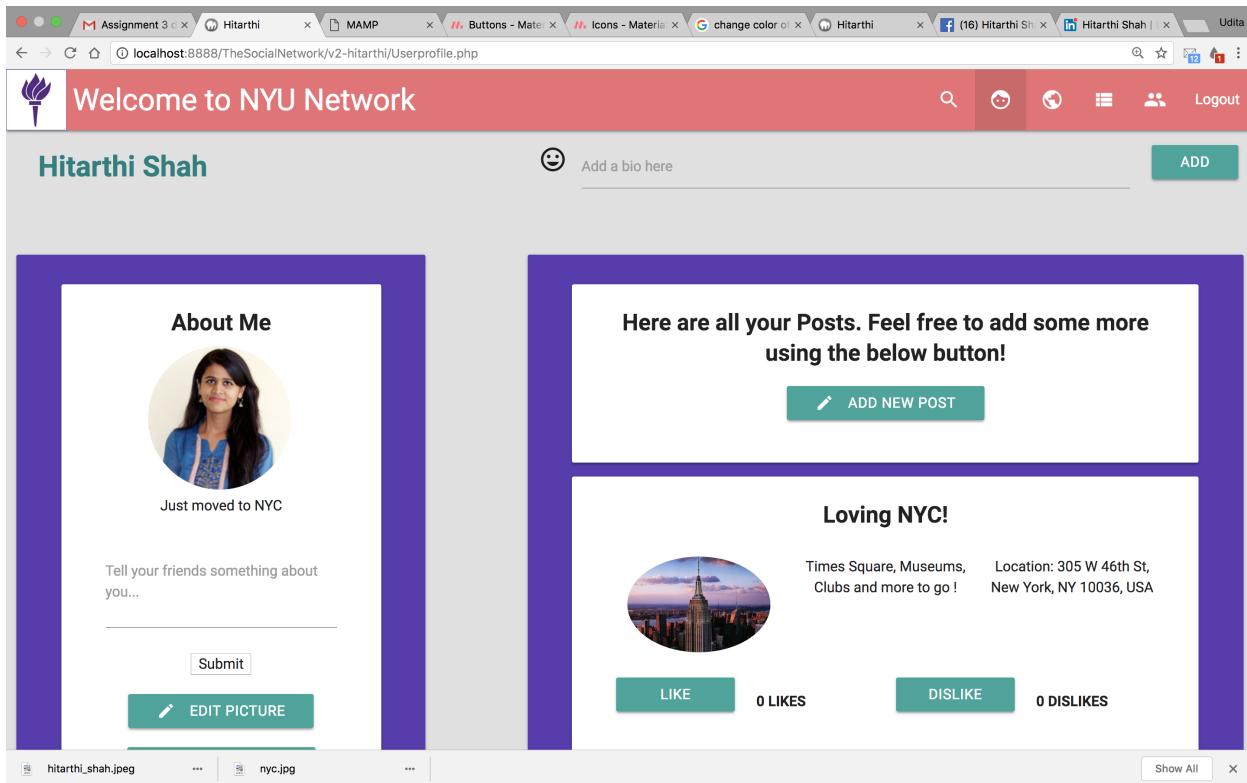
- Once user logs in, it enters into his profile.
- The homepage is divided into two parts.

Part 1 - will have options to edit and view profile for that particular user also user will be able to see his friends by clicking 'view friends' button.

- User can also add bio and story which is nothing but what user wants other to see on his timeline.
- User can edit his profile picture by clicking 'edit picture' button.

Part 2 - second half of that page will allow user to add a new post.

- The details about the post is mentioned further in Diary Entry.
- We have showed demo post format so that user can have the idea of what all things he can add to the post and how will it look.
- Once the post is added, it will be shown on that page separately with its timestamp, name, body text, likes and dislikes.
- If user has added image or location, it will also be shown.
- User can like or dislike his own post and can see that how many likes and dislikes he has received from other users.



Add Friend

Pending Request

- User will be able to see all the request from other users who have send him a request to be their friend.
- User can either accept it or decline it.
- User can also do nothing with the request and it will stay there.
- We are also notifying user that he has received a friend request.

Accept Request

- Once user accepts the friend request, both, the user and the friend will receive a notification saying the other person has accepted the friend request.

Decline Request

- If user declines any request, no one will be notified about it and the request will get deleted from the users account.

localhost:8888/TheSocialNetwork/v2-hitarthi/Userprofile.php

Welcome to NYU Network

Hitarthi Shah

About Me



my first storyyy

Tell your friends something about you...

Submit

EDIT PICTURE

VIEW DETAILS

FRIENDS

PENDING REQUESTS



UDITA GUPTA

LOCATION

32 DISLIKES

7th post

localhost:8888/TheSocialNetwork/v2-hitarthi/Userprofile.php

Welcome to NYU Network

Hitarthi Shah

About Me



my first storyyy

Tell your friends something about you...

Submit

EDIT PICTURE

VIEW DETAILS

FRIENDS

PENDING REQUESTS

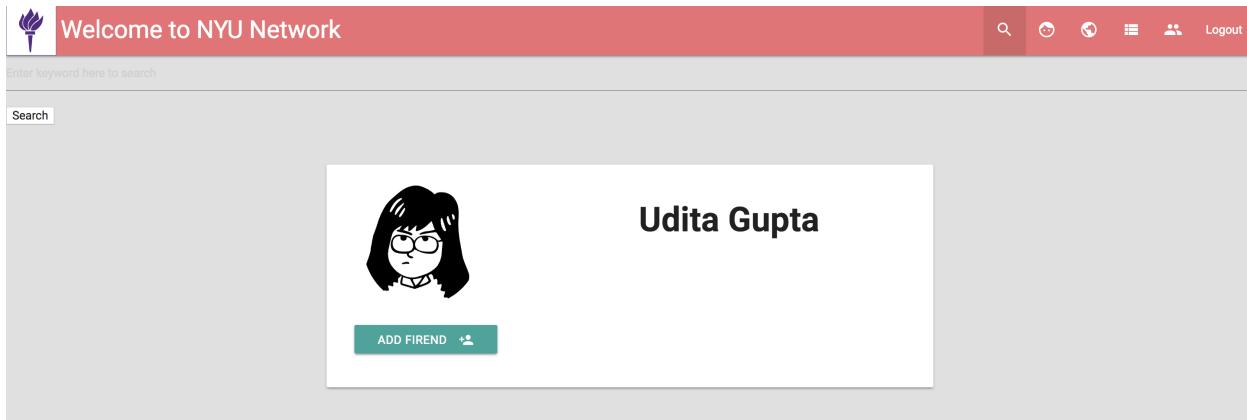


UDITA GUPTA

LOCATION

32 DISLIKES

7th post



Welcome to NYU Network

Enter keyword here to search

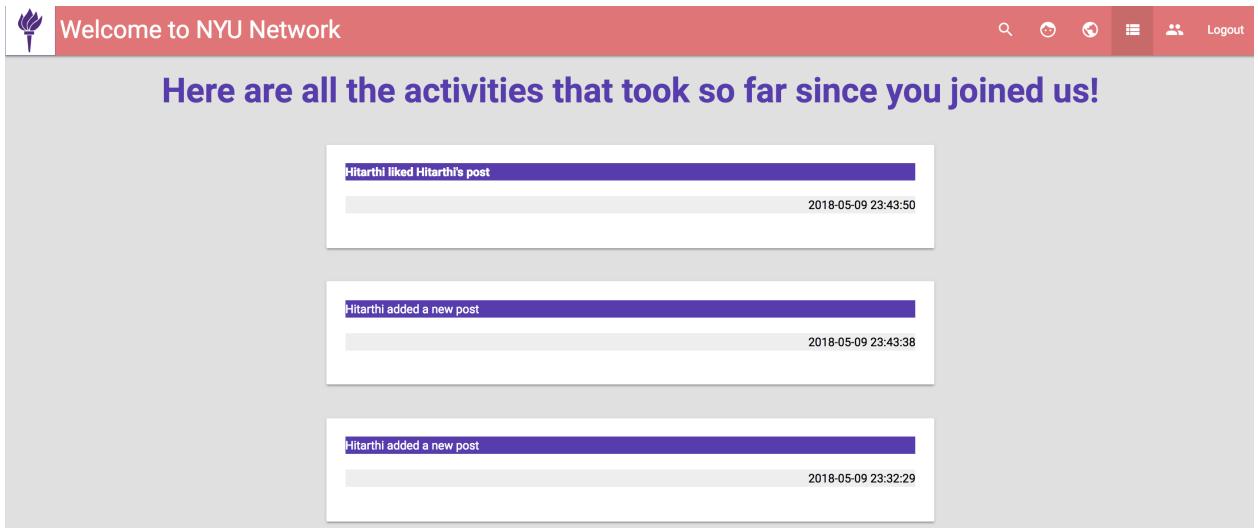
Search

Urita Gupta

ADD FRIEND +1

Activity Post

- User can see what his friends are posting and sharing.
- He can see the notifications in that section from other users for example if he has become friends with some other person or if someone has posted any post.
- In general, what all other friends of that user are doing will be shown in activity post.
- The user will only be able to see the post and notifications from his friends and not all.
- User will be able to see only the posts that he has privilege to see.



Welcome to NYU Network

Here are all the activities that took so far since you joined us!

Hitarthi liked Hitarthi's post
2018-05-09 23:43:50

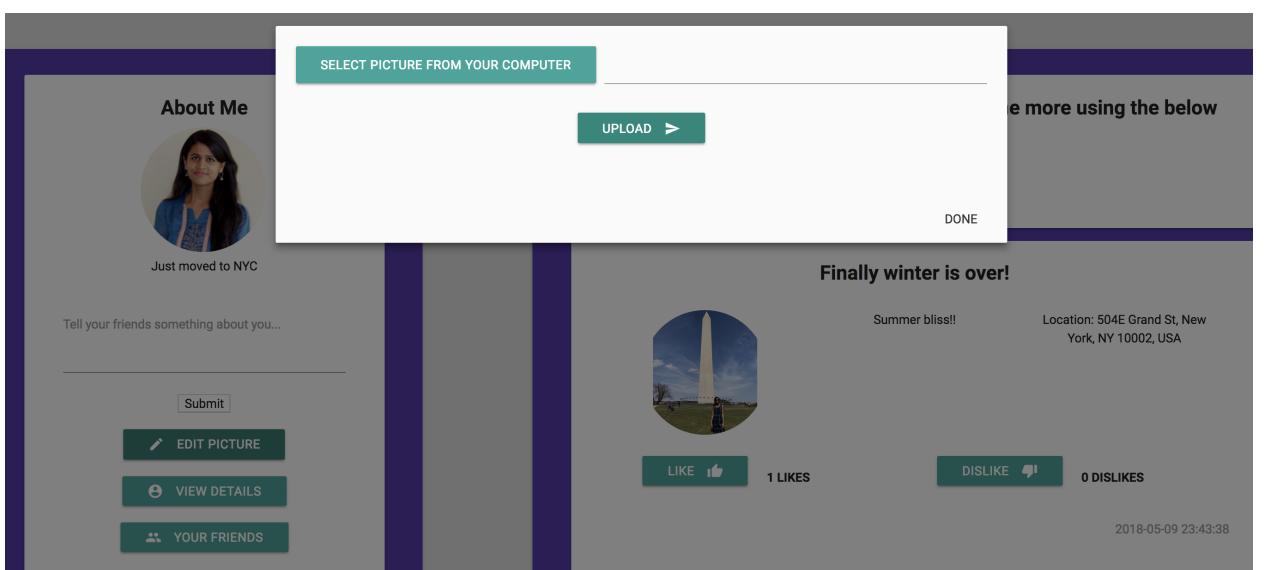
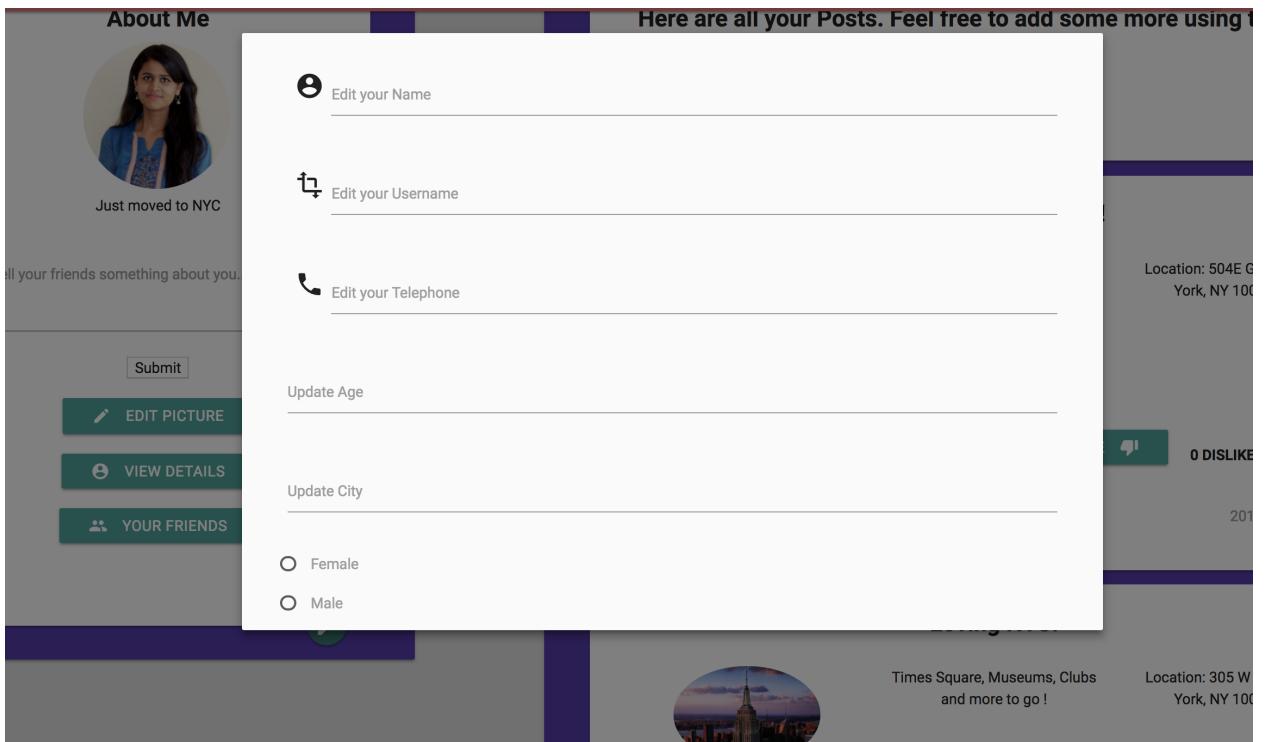
Hitarthi added a new post
2018-05-09 23:43:38

Hitarthi added a new post
2018-05-09 23:32:29

Diary Entry

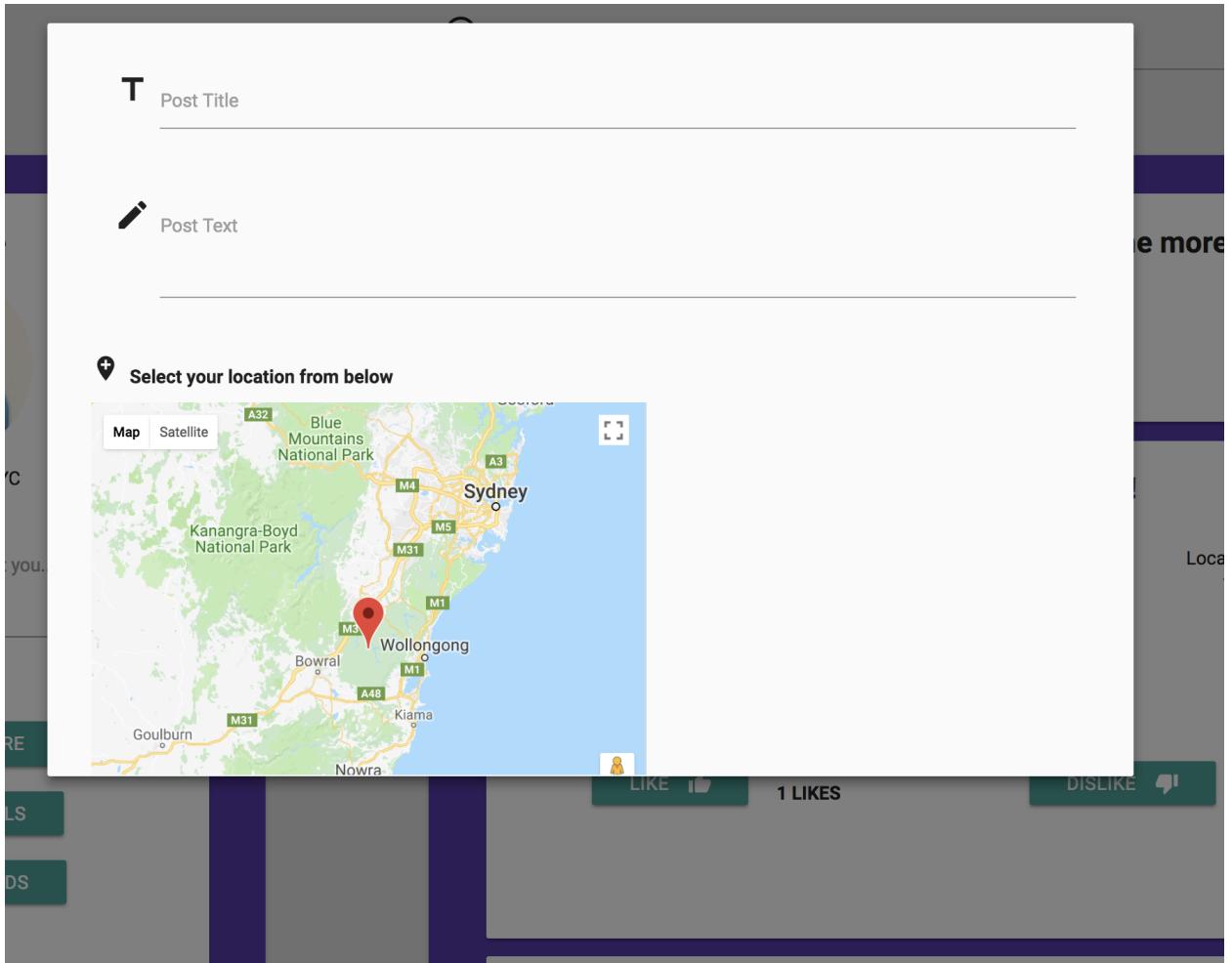
- User can upload a post, which is called diary entry.
- User can add following details to a post.

- Post title
- Post text
- Location
- Image
- He can set the post to public, private or friends and friends of friends.



Location

- We have added an option to choose location for each post.
- It is implemented using Google Maps API.
- Person can drag a pointer on the world map to select a location and the latitude and longitude for that location, along with its name and full address will be saved in the entry of post.



Like and Dislike

- A user can like and dislike his own post.
- A user can like and dislike other friends post.
- Like and Dislike counter will be updated like wisely and will be stored in Diary table.

Privileges

- Privacy_control consists of different types of privacy levels like friends, friends of friends, only me, public etc.

- User can only see the data which is made public to him.
- He can choose to keep his post private or public.

Search

- We have implemented search for two tasks.
- **Task 1** - User can search a keyword to find some other user to be friends with him or to see his public content.
 - In this when he will enter the keywords of the name of the other user, he will be able to see all the users having that keywords.
- **Task 2** - User can search for a post keyword and all the posts containing that keyword will be shown.
- Whenever the keywords will be entered, both usernames and posts containing that keyword will be shown.

User Session

- We have implemented Session throughout the website.

Timeline

