Twitter Marshmellow

Team Members:

|  |  |
| --- | --- |
| Name | SJSU ID |
| Pavani Vellal | 010702497 |
| Preetusha Patne | 010708477 |
| Sayori Patil | 010756681 |
| Tanya Sanjay | 010766249 |

ABSTRACT

*Twitter bases its application programming interfaces (API) on the REST architecture (Representational State Transfer).REST is an architectural style and approach to communications that is often used in the development of web services.*

*For Twitter a REST architecture in part means that the service works with most syndication formats. Twitter is compatible with two of them Really Simple Syndication (RSS) and Atom Syndication Format (Atom).Both formats retrieve data from one resource and send it to another.*

*Application only authentication is a form of authentication where your application makes API requests on its own behalf, without a user context. API calls are still rate limited per API method, but the pool each method draws from belongs to your entire application at large, rather than from a per-user limit. API methods that support this form of authentication will contain two rate*

*limits in their documentation, one that is* per user *(for application-user*

*authentication) and the other is* per app *(for this form of application-only authentication). Not all API methods support application-only authentication.*

*The REST API’s provide programmatic access to read and write Twitter data. Author a new Tweet, read author profile and follower data, and more. The REST API identifies Twitter applications and users using OAuth; and the responses are available in JSON.*

*The following 8 REST API’s used in this project are*

 *GET* ***search/tweets***

*Returns a collection of relevant Tweets matching a specified*

*query.*

 *GET* ***statuses/user\_timeline***

*Returns a collection of the most recent Tweets posted by the user indicated by the screen\_name or user\_id parameters.*

 *GET* ***followers/ids***

*Returns a cursored collection of user ids for every user following the specified user*

 *GET* ***statuses/retweets/:id***

*Returns up to 100 of the first retweets of a given tweet.*

 *GET* ***statuses/show/:id***

. *Returns a single Tweet, specified by the id parameter*

 GET ***friends\_list***

*Returns a cursored collection of user objects for every user the*

*specified user is following (otherwise known as their “friends”).*

 *GET* ***statuses/lookup***

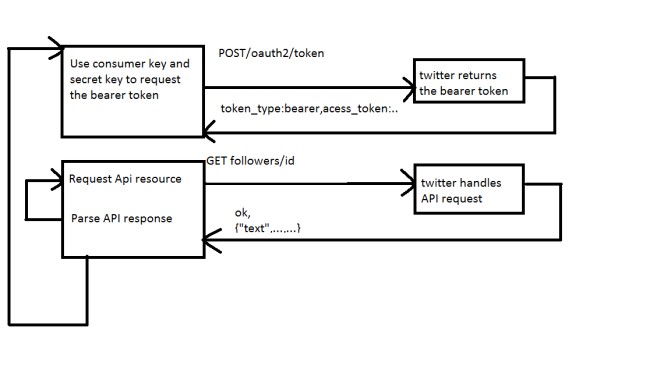
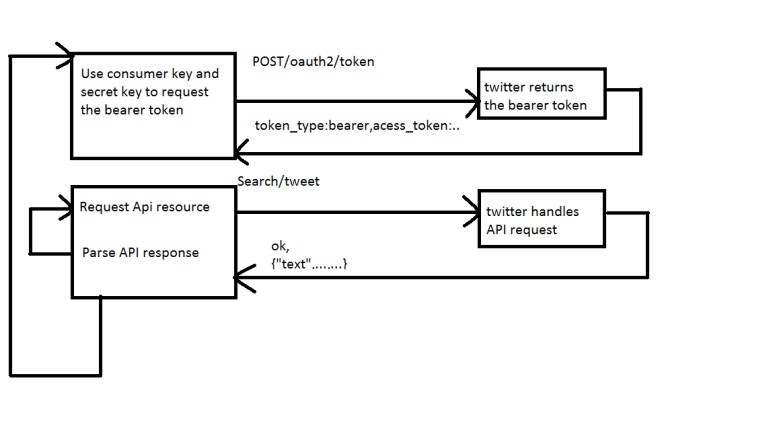
*Returns fully-hydrated tweet objects for up to 100 tweets per request, as specified by comma-separated values passed to the id parameter.*

 *GET* ***Users/lookup***

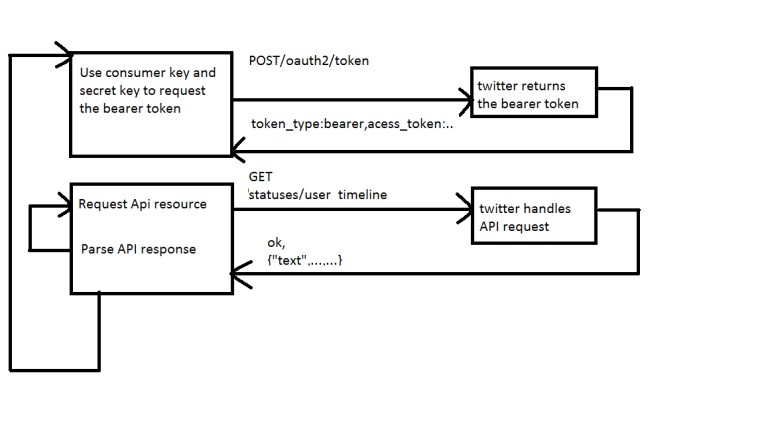
*Returns fully hydrated user objects for up to 100 users per request, as specified by comma separated values passed to the user\_id and/or screen\_name parameters.*

UML DIAGRAMS

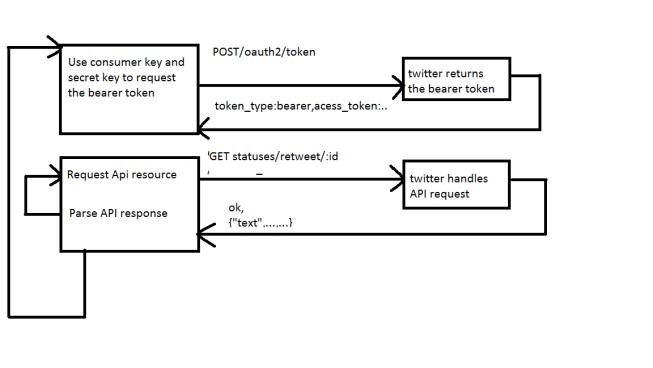
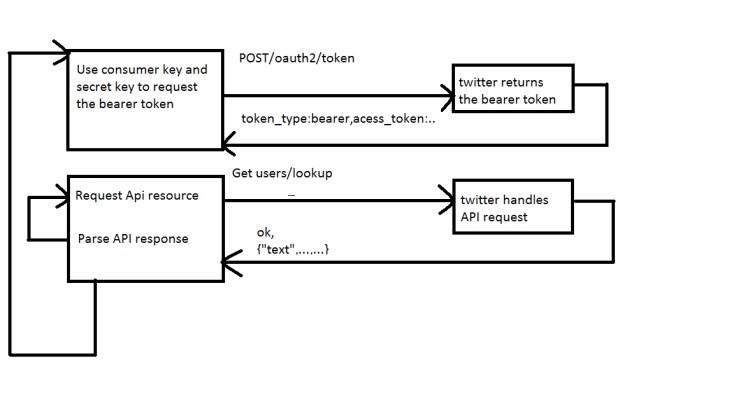
Search/tweets



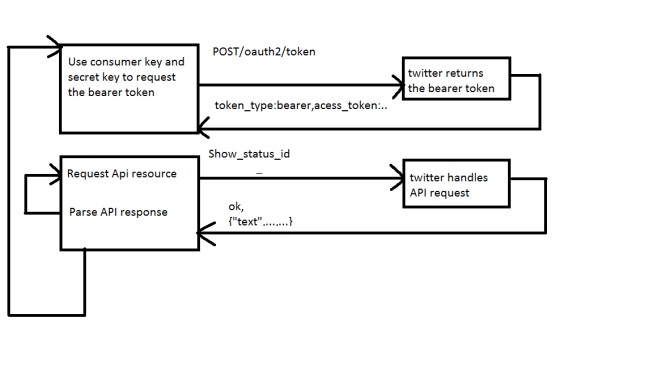
GET statuses/user\_timeline



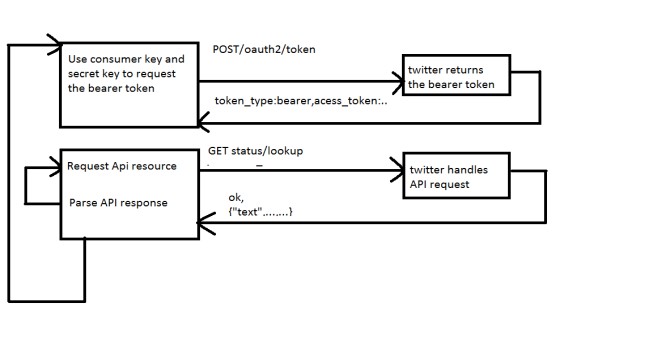
GET users/lookup



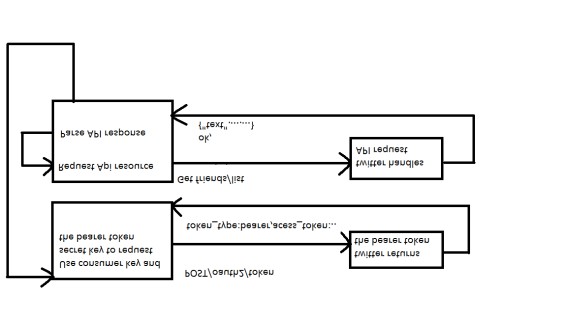
GET status\_id



Get status/lookup



Get friends\_list



USECASES

User Twitter Twitter

OAuth

Search

User Request access

Application request token

Request token and

URL is provided to application

User navigates to URL at Titter and logs in to Twitter

Access token and URL passed to user

User navigates to

URL with access token

User is granted access

Application validates

Access token

Access token is

Validated, Authentication success

User request Search/tweets API

OAuth takes over search API

Validates for consumer key and consumer secret from user

If successful executes user request

User Twitter Twitter

OAuth

Search

User Request access

Application request token

Request token and

URL is provided to application

User navigates to URL at Titter and logs in to Twitter

Access token and URL passed to user

User navigates to

URL with access token

User is granted access

Application validates

Access token

Access token is

Validated, Authentication success

User request GET followers/ids

OAuth takes over requested API Validates for consumer key and consumer secret from user

If successful executes user request

User Twitter Twitter

OAuth

Search

User Request access

Application request token

Request token and

URL is provided to application

User navigates to URL at Titter and logs in to Twitter

Access token and URL passed to user

User navigates to

URL with access token

User is granted access

Application validates

Access token

Access token is

Validated, Authentication success

User request statuses/

user\_timelines API

OAuth takes over requested API Validates for consumer key and consumer secret from user If successful executes user request

User Twitter Twitter

OAuth

Search

User Request access

Application request token

Request token and

URL is provided to application

User navigates to URL at Titter and logs in to Twitter

Access token and URL passed to user

User navigates to

URL with access token

User is granted access

Application validates

Access token

Access token is

Validated, Authentication success

User request Get User/lookup API

OAuth takes over requested API Validates for consumer key and consumer secret from user

If successful executes user request

User Twitter Twitter

OAuth

Search

User Request access

Application request token

Request token and

URL is provided to application

User navigates to URL at Titter and logs in to Twitter

Access token and URL passed to user

User navigates to

URL with access token

User is granted access

Application validates

Access token

Access token is

Validated, Authentication success

User request GET statuses/ Retweets/:id API

OAuth takes over requested API Validates for consumer key and consumer secret from user If successful executes user request

User Twitter Twitter

OAuth

Search

User Request access

Application request token

Request token and

URL is provided to

Application

User navigates to URL at Titter and logs in to Twitter

Access token and URL passed to user

User navigates to

URL with access token

User is granted access

Application validates

Access token

Access token is

Validated, Authentication success

User request POST

statuses/show/:id API

OAuth takes over requested API Validates for consumer key and consumer secret from user If successful executes user request

User Twitter Twitter

OAuth

Search

User Request access

Application request token

Request token and

URL is provided to application

User navigates to URL at Titter and logs in to Twitter

Access token and URL passed to user

User navigates to

URL with access token

User is granted access

Application validates

Access token

Access token is

Validated, Authentication success

User request Get statuses/Lookup API

OAuth takes over requested API Validates for consumer key and consumer secret from user If successful executes user request

User Twitter Twitter

OAuth

Search

User Request access

Application request token

Request token and

URL is provided to application

User navigates to URL at Titter and logs in to Twitter

Access token and URL passed to user

User navigates to

URL with access token

User is granted access

Application validates

Access token

Access token is

Validated, Authentication success

User request GET friends\_list API

OAuth takes over requested API Validates for consumer key and consumer secret from user

If successful executes user request

T

User Twitter Twitter

OAuth

Search

User Request access

Application request token

Request token and

URL is provided to application

User navigates to URL at Titter and logs in to Twitter

Access token and URL passed to user

User navigates to

URL with access token

User is granted access

Application validates

Access token

Access token is

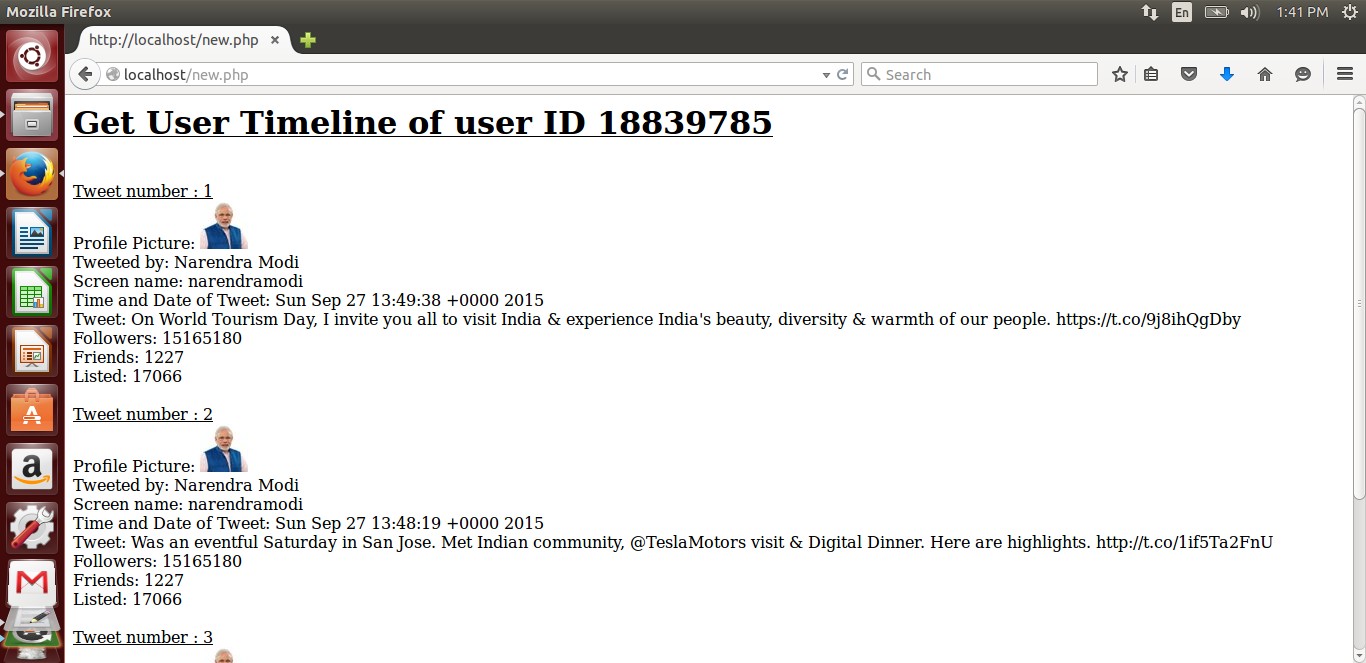
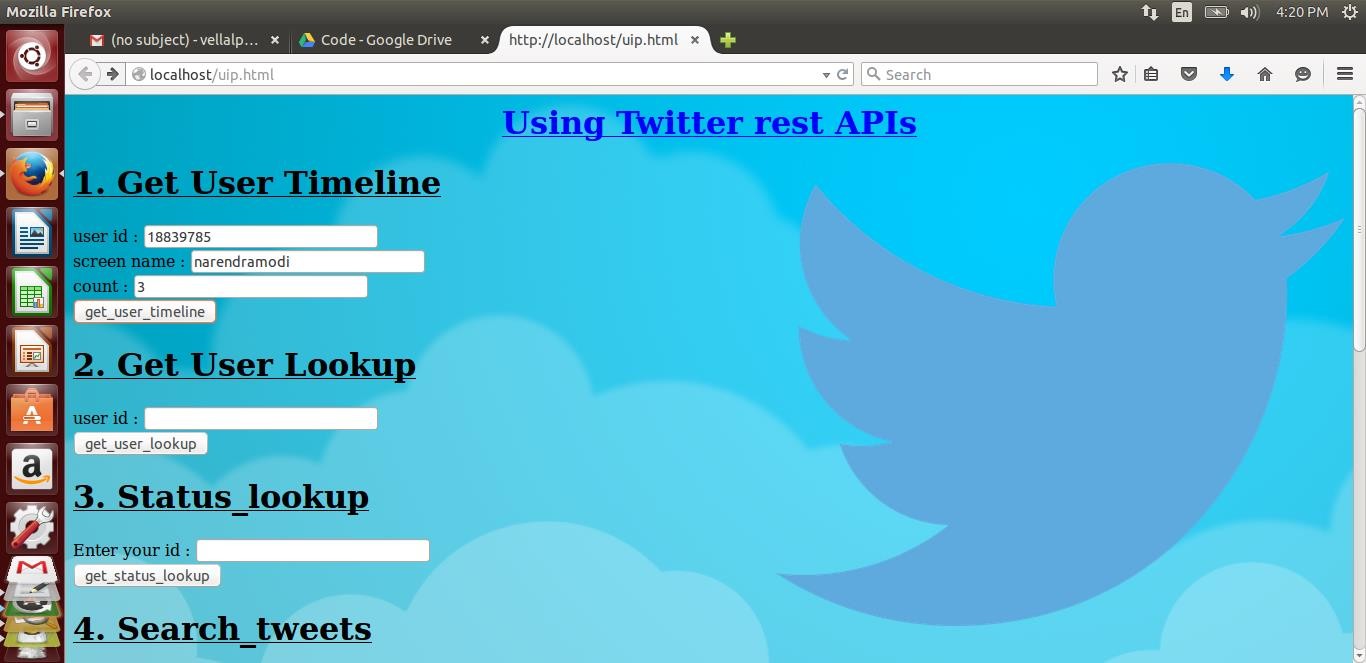
Validated, Authentication success

User request GET statuses/update API

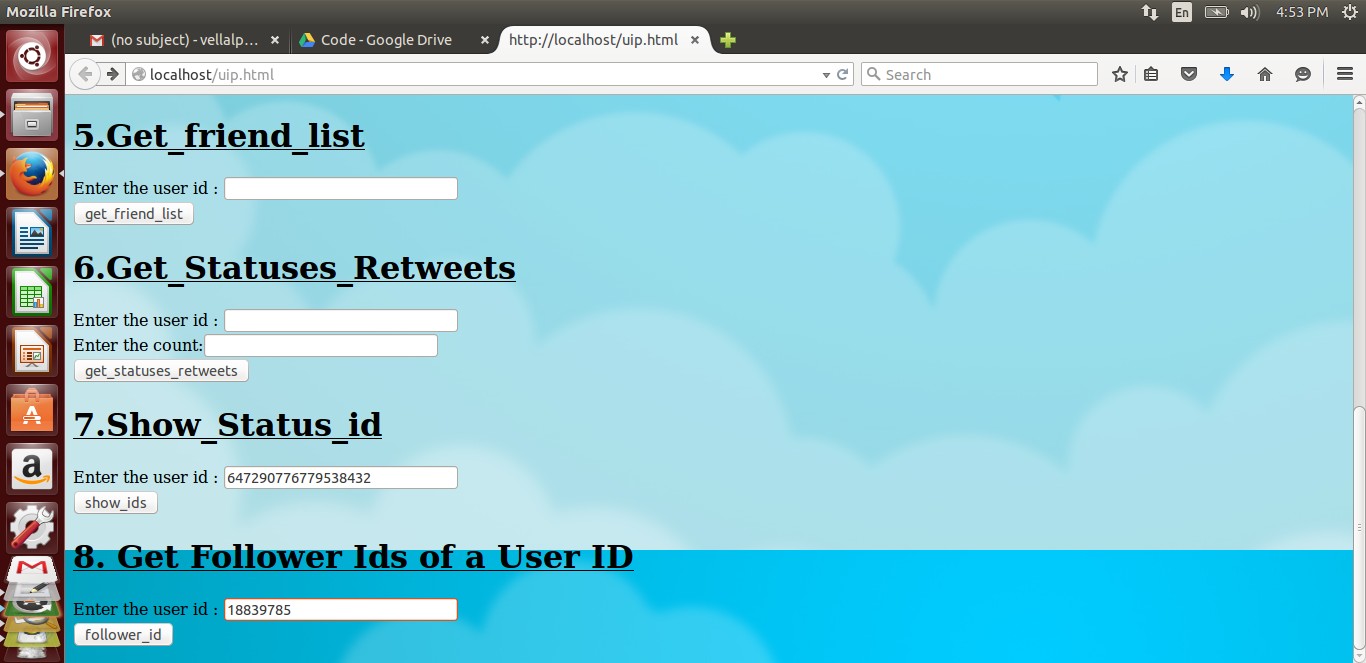
OAuth takes over requested API Validates for consumer key and consumer secret from user If successful executes user request

**TESTCASES**

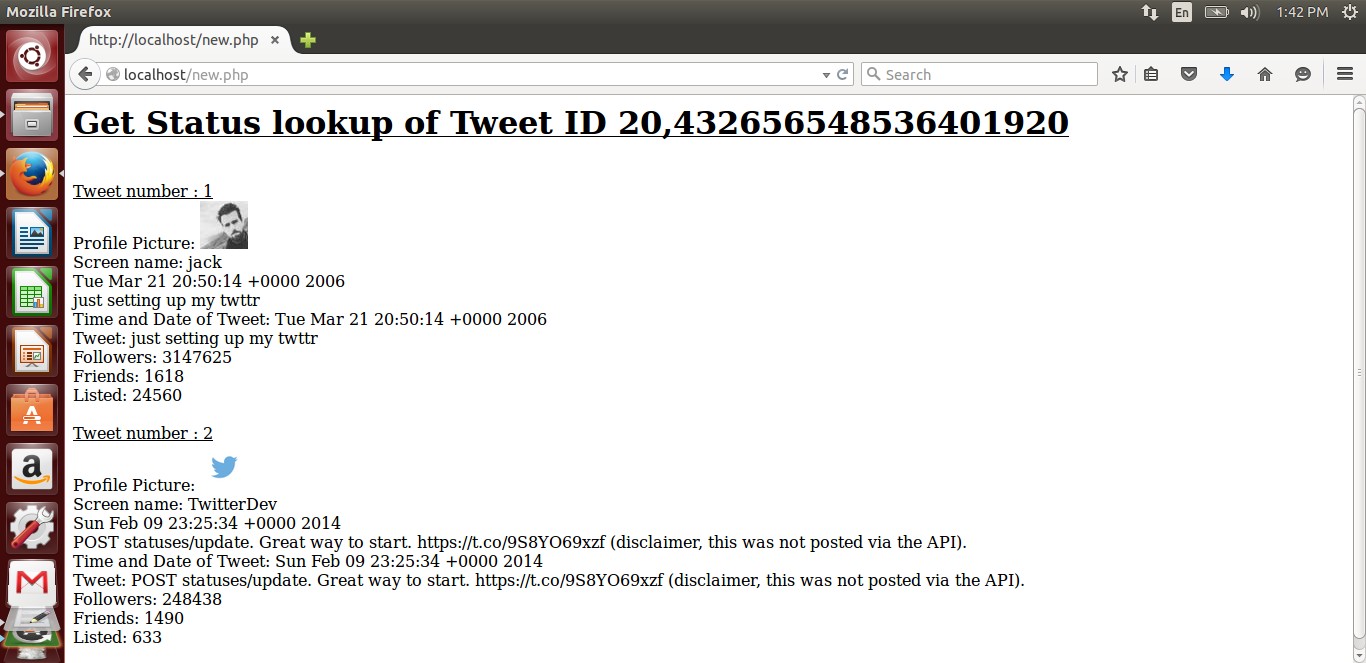
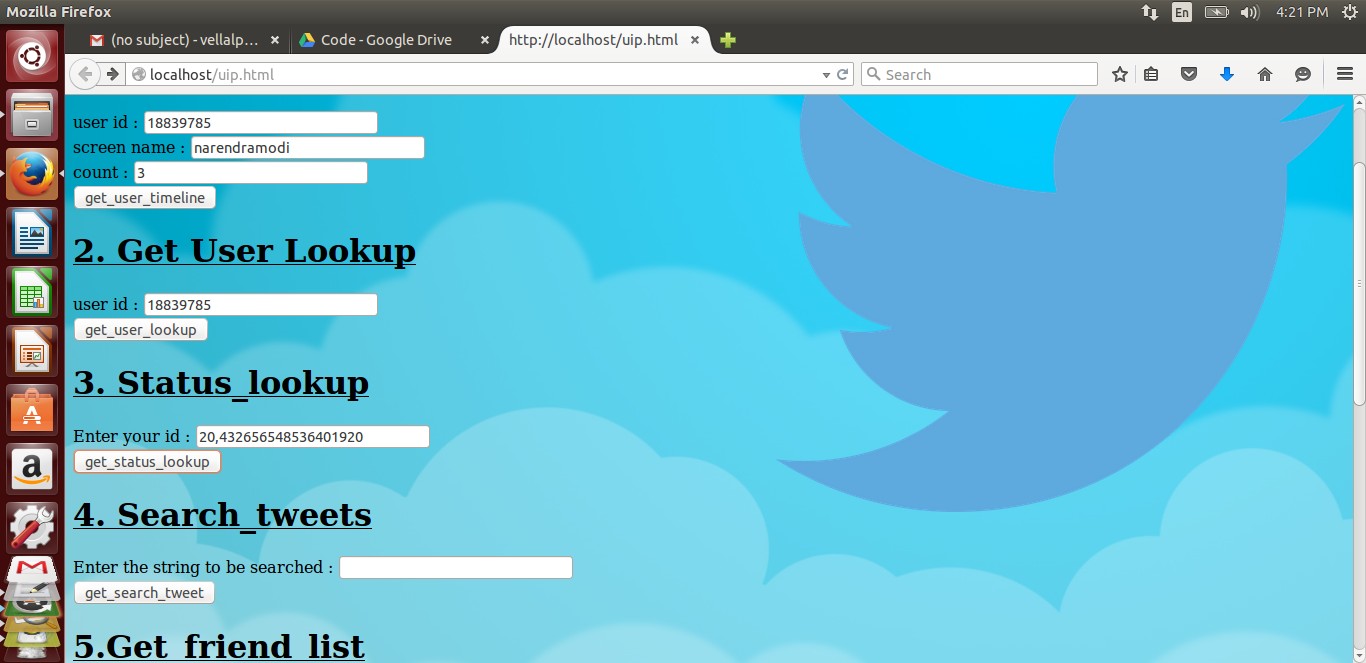
**GET user/timeline**



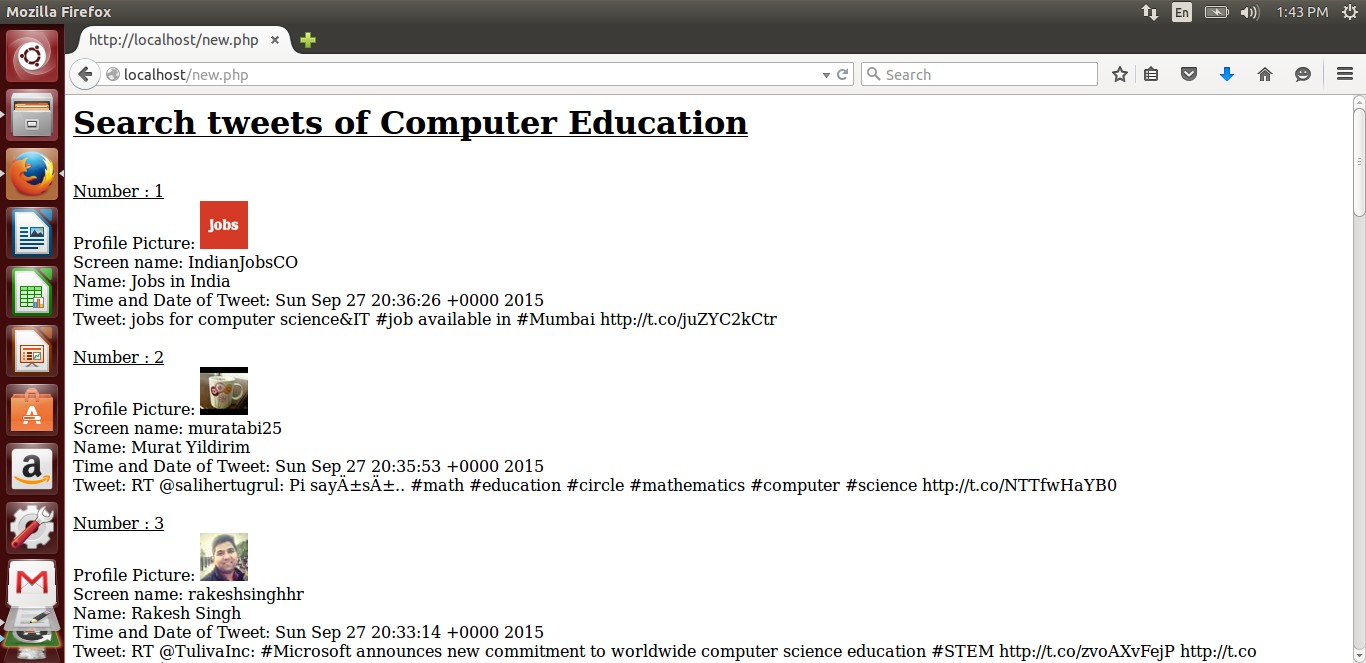
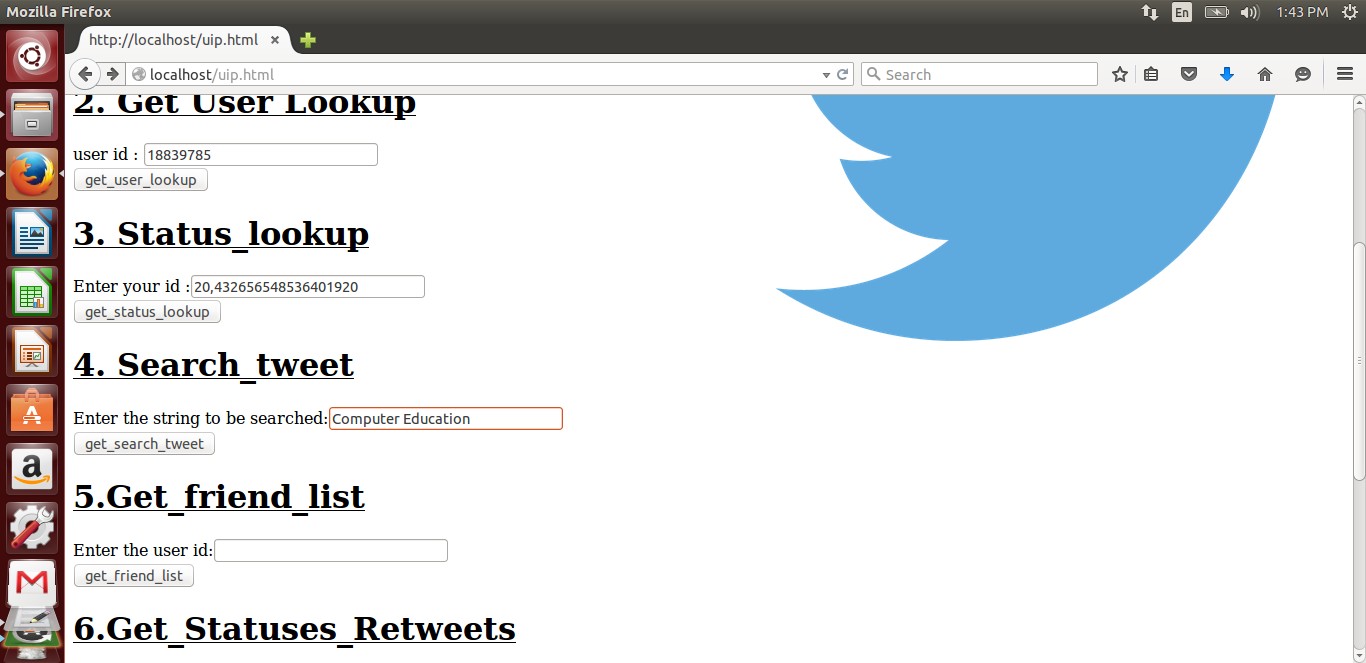
**GET followers/ids**



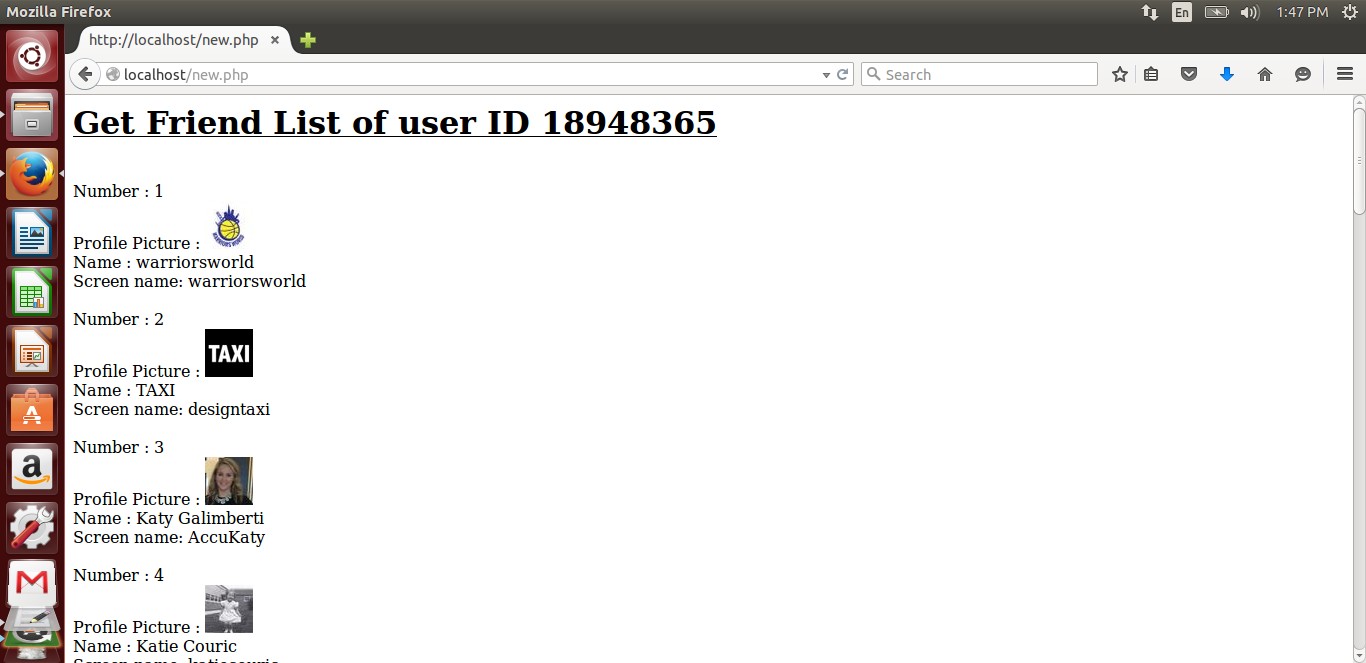
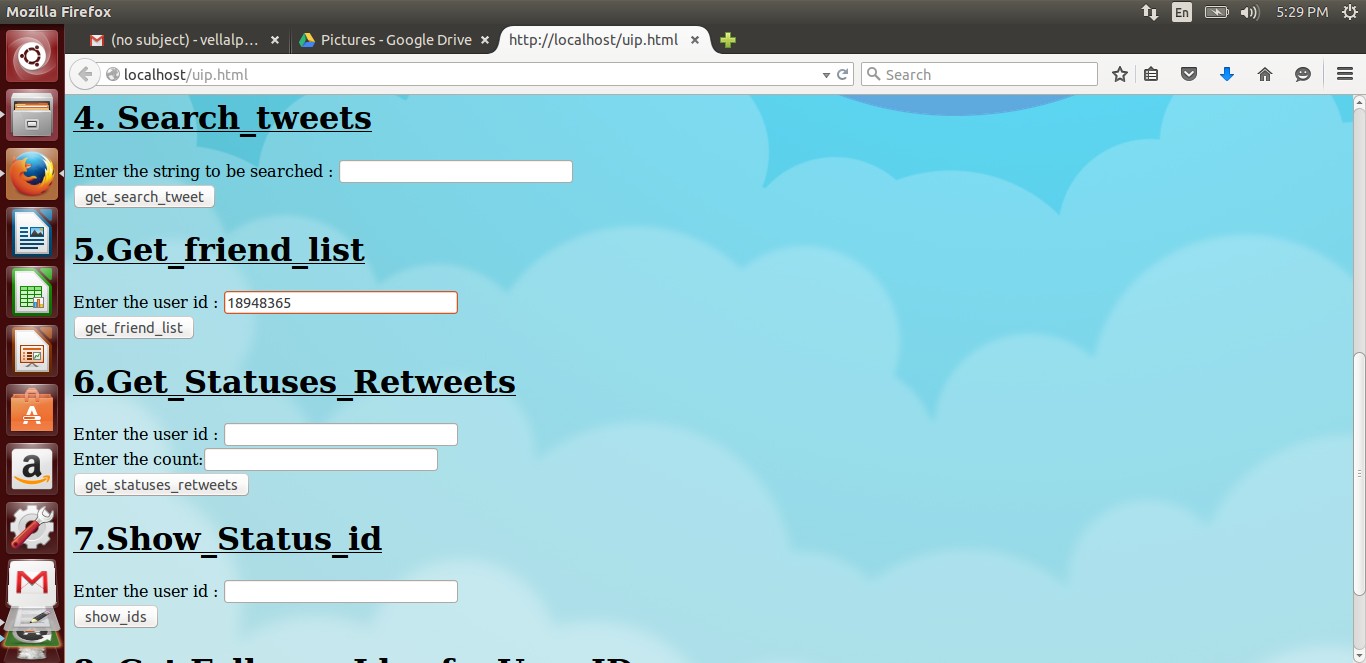
**GET status/lookup**



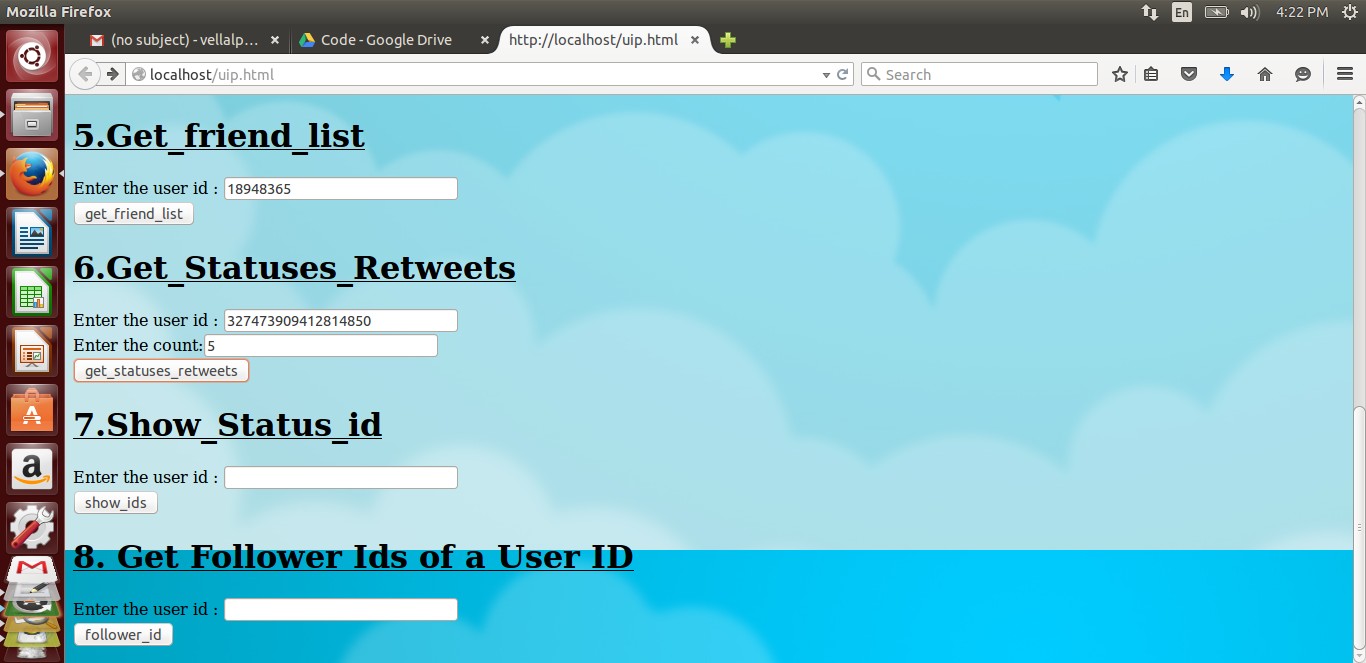
**GET search/tweet**



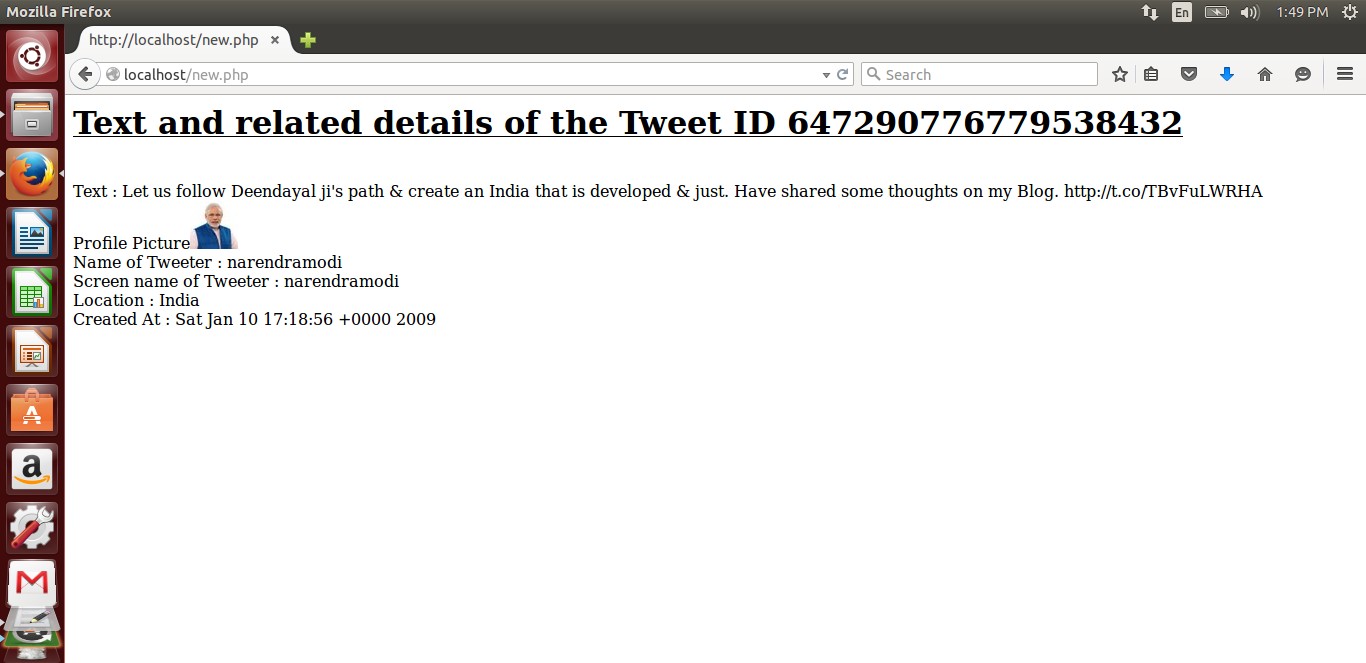
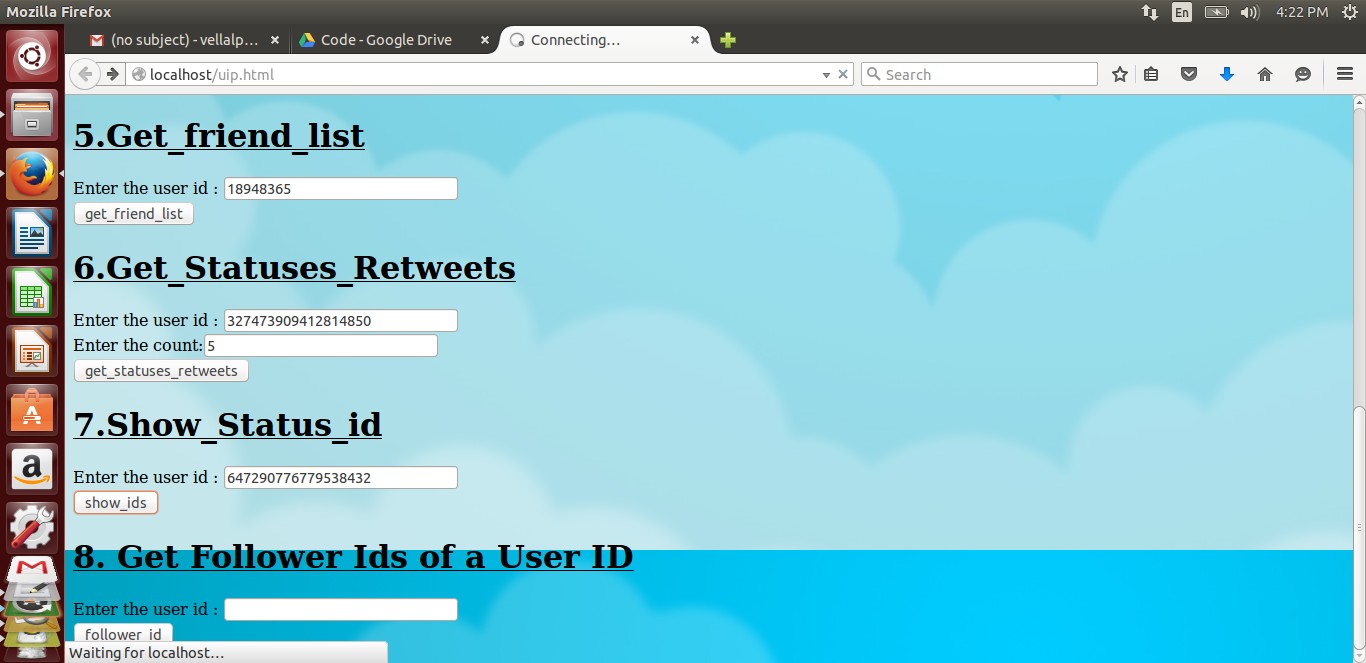
**GET friends\_list**



**GET statuses/retweets/:id**



**GET show\_id**



**GET users/lookup**

