|  |  |  |
| --- | --- | --- |
| **Title: Write a client application that utilizes an Apache Camel / Apache Karaf Container framework and implements the Twitter REST APIs** | **Group : 4Spartans** | |
| **Tarun Shetty (Team Lead)** | **012449697** |
| **SaloniSharad Bhalerao** | **012466623** |
| **Hemambhujam Veeraraghavan** | **012482392** |
| **Anu Sebastian** | **012496276** |

|  |  |
| --- | --- |
| **GitHub Repository** | |
| **Tarun Shetty – 4Spartans Repo. link** | https://github.com/tarunshetty20/4-Spartans/tree/master/karaf |
| Saloni Sharad Bhalerao | https://github.com/SaloniBhalerao/Enterprise\_Sw\_Platforms/tree/master/Karaf |
| HemambhujamVeeraraghavan | https://github.com/hemambujamV/Twitter-API-implementation-using-apache-camel-Karaf |
| Anu Sebastian | https://github.com/anuksebastian/ESP-Karaf |

**ABSTRACT**

The purpose of this project is to write a client application which uses apache karaf container framework (a small OSGi based runtime) which provides a lightweight container onto which various components and applications can be deployed) to implement 8 twitter API’s. The application is built around the twitter account of the developer. The developer request for the authentication key and token for the twitter account to work on the twitter APIs. Using these tokens and keys, the developer gets to choose the from the list of twitter APIs from the twitter website and call these APIs locally. A HTML page is designed as a part of user interface with CSS styling which runs on a local host. This UI contains the list of the APIs that were implemented by the team. Each of these APIs have a unique function to perform. When the client clicks on any of these API’s it redirects to a new page which displays the response to the   
API Request. Whenever the user clicks on these APIs, the tokens and keys get authenticated before displaying the result of these API calls.

**Pre-requisite**

* Create Twitter App (Twitter Display Name: 4Spartans) [Get the OAuth credentials to use in the application]

<http://docs.inboundnow.com/guide/create-twitter-application/#toc-0>

* Set up the development environment - install JDK, MAVEN, KARAF and ECLIPSE for IDE

**Commands issued on the Karaf terminal (w.r.t feature: twitter-api)**

1.Invoke karaf.bat

2.karaf@root()> feature:repo-add mvn:homework/api/1.0.0/xml/features

3.karaf@root()> feature:install twitter-api

**In-order to rebuild after code change**

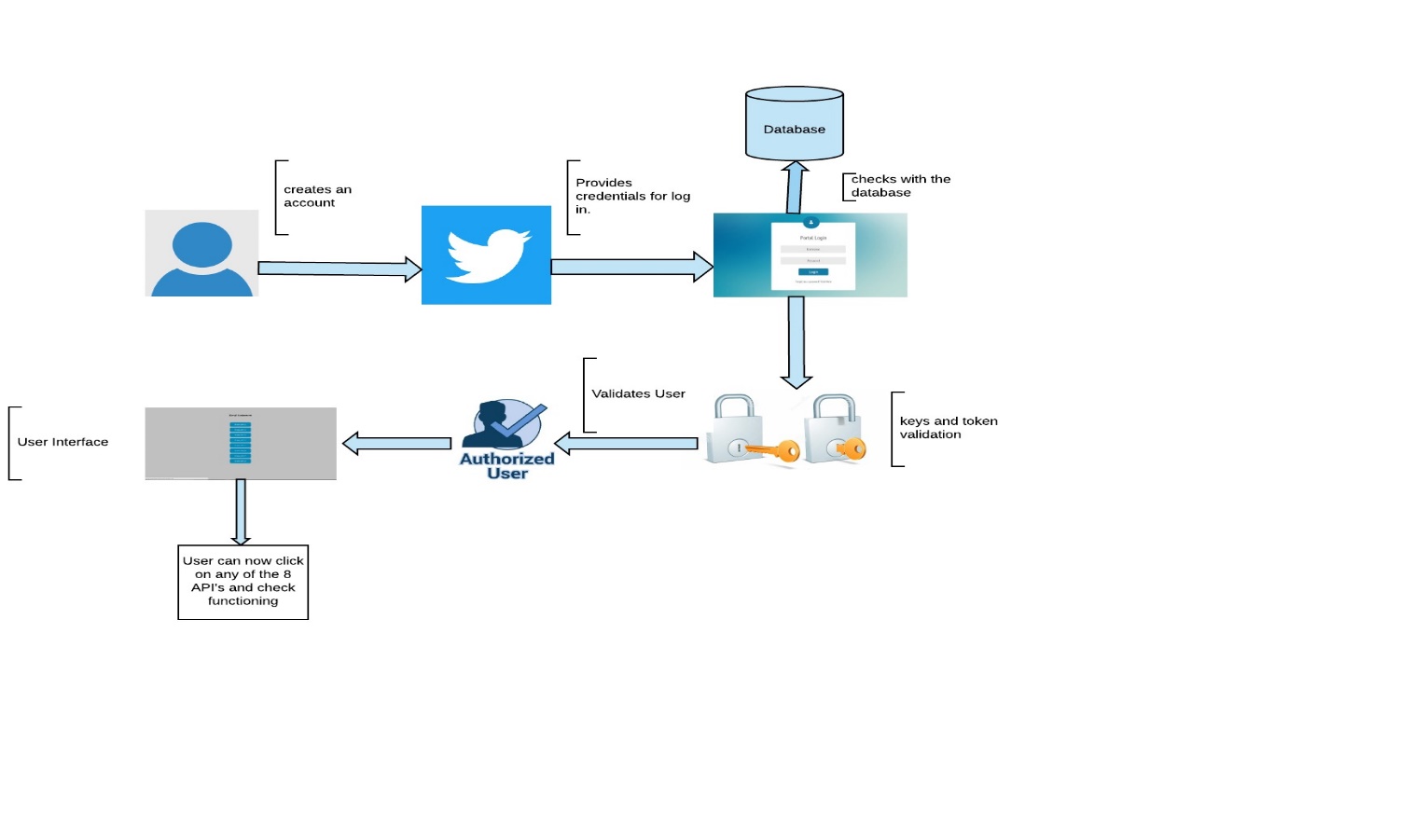
1.$ mvn clean install

2.karaf@root()> feature:uninstall twitter-api

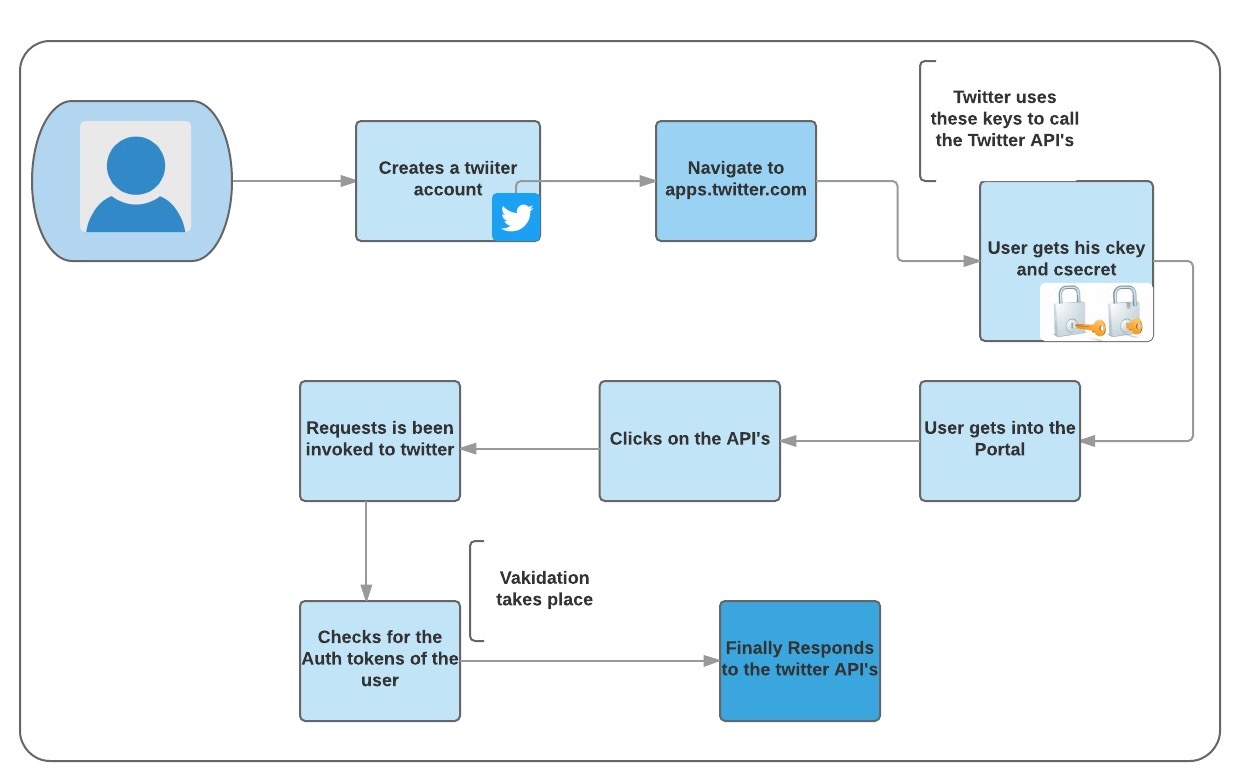
3.karaf@root()> feature:repo-remove mvn:homework/api/1.0.0/xml/features

4.karaf@root()> feature:repo-add mvn:homework/api/1.0.0/xml/features

5.karaf@root()> feature:install twitter-api

**ARCHITECTURE DIAGRAM**

**BLOCK DIAGRAM**



**Tools**

* Apache Karaf
* Maven
* Postman
* Eclipse

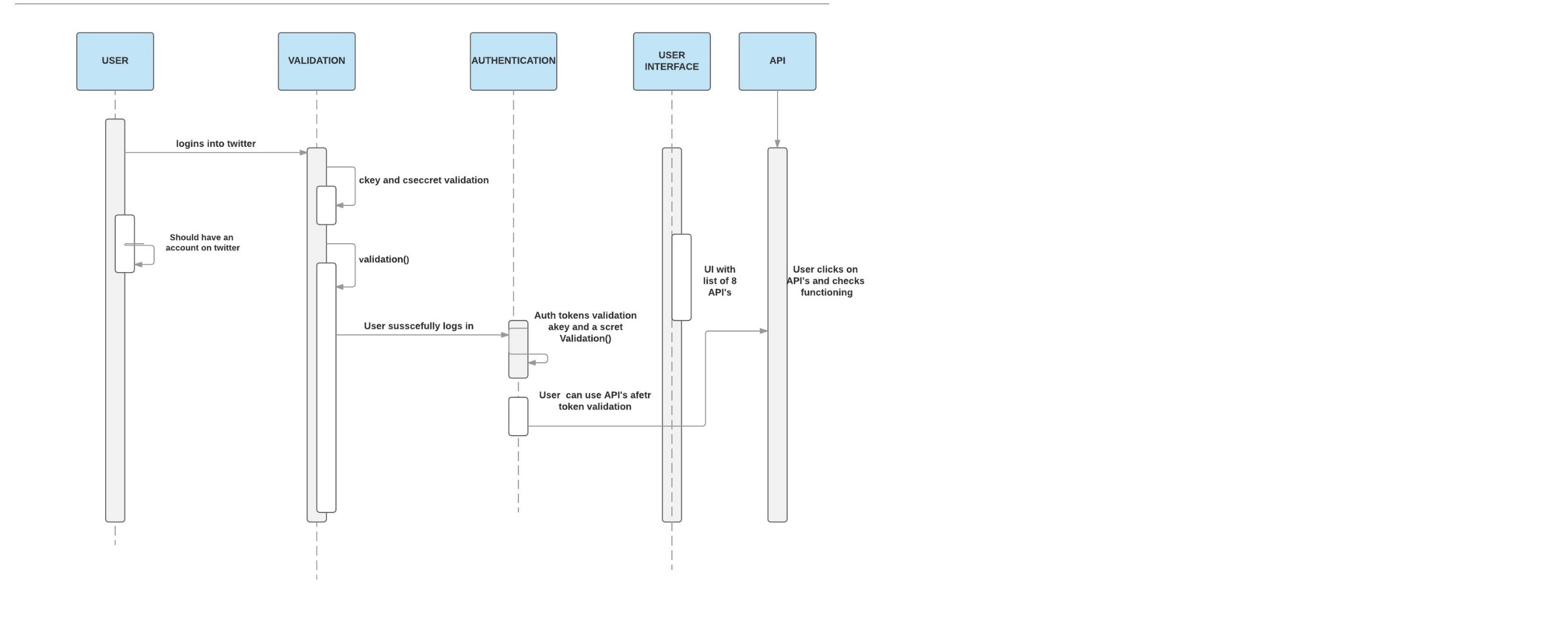
**Technologies**

* JAVA
* HTML, CSS

**API’s used**

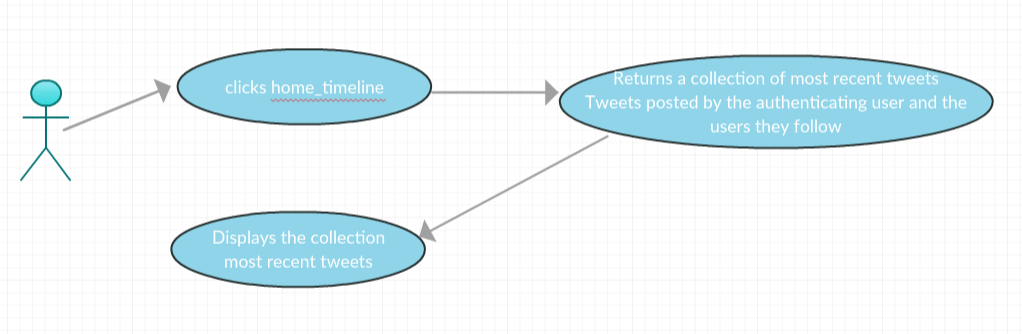
1. statuses/home\_timeline (<https://developer.twitter.com/en/docs/tweets/timelines/api-reference/get-statuses-home_timeline>)
2. statuses/mentions\_timeline (<https://developer.twitter.com/en/docs/tweets/timelines/api-reference/get-statuses-mentions_timeline>)
3. statuses/user\_timeline(<https://developer.twitter.com/en/docs/tweets/timelines/api-reference/get-statuses-user_timeline>)
4. friends/ids (<https://developer.twitter.com/en/docs/accounts-and-users/follow-search-get-users/api-reference/get-friends-ids>)
5. followers/list(<https://developer.twitter.com/en/docs/accounts-and-users/follow-search-get-users/api-reference/get-followers-list>)
6. account/settings(<https://developer.twitter.com/en/docs/accounts-and-users/manage-account-settings/api-reference/get-account-settings>)
7. help/privacy (<https://developer.twitter.com/en/docs/developer-utilities/privacy-policy/api-reference/get-help-privacy>)
8. help/tos(<https://developer.twitter.com/en/docs/developer-utilities/terms-of-service/api-reference/get-help-tos>)

**UML sequence diagram**

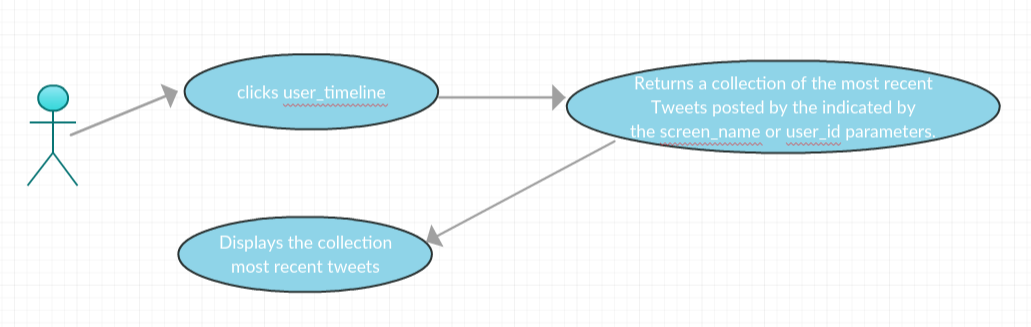


**Use cases, Test cases with screen shots**

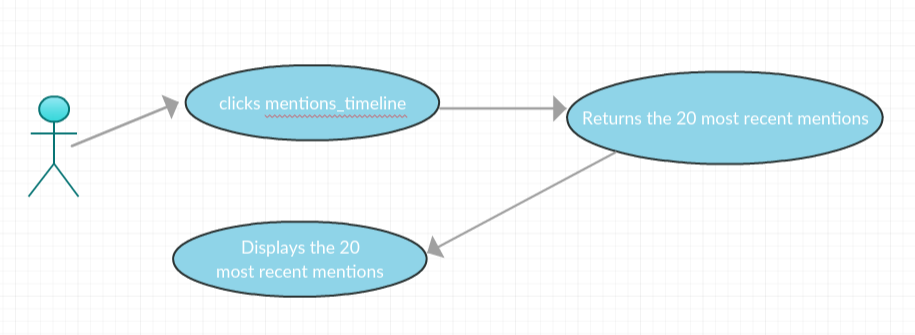
home\_timeline: Returns a collection of most recent tweets by the user and the users they follow. When the user clicks on home\_timeline the user is returned with the response of tweets.

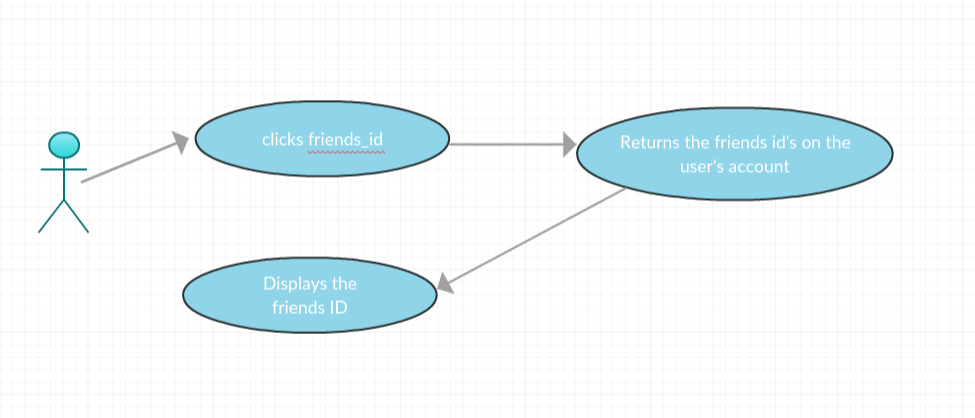


user\_timeline: This returns the collection of tweets made by a particular user by screen-name.When the user clicks on the user\_timeline he is redirected to the page where the response is displayed.

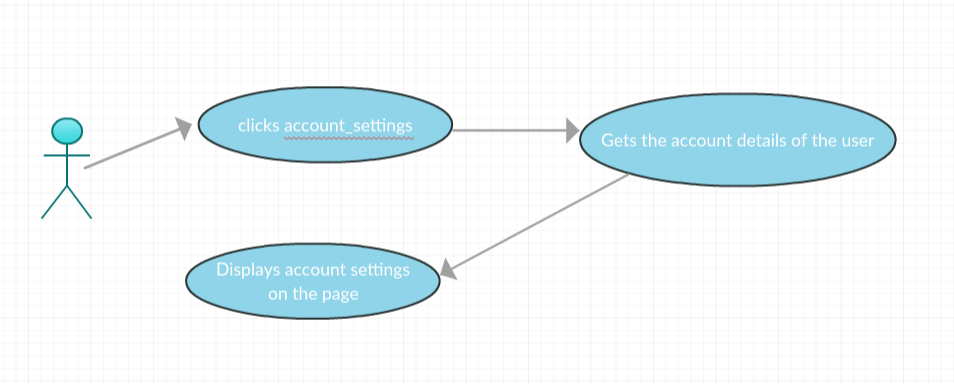


mentions\_timeline: returns the 20 recent tweets ,this is displayed to the user.

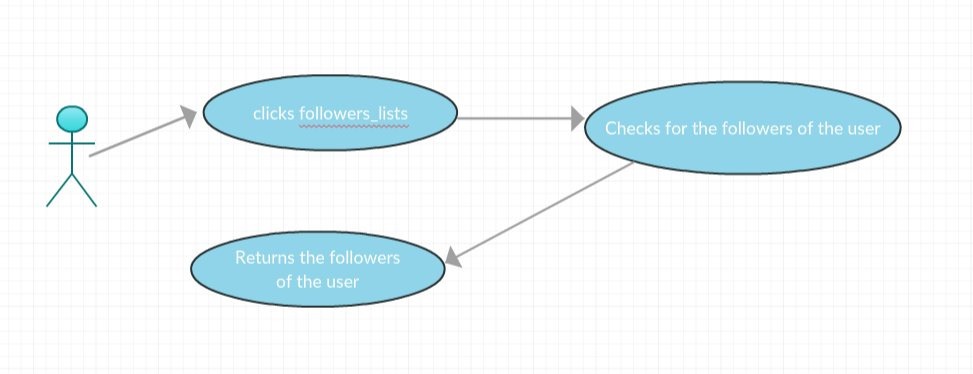


friends\_id: The response for friends/id returns a list of ids the user is following. These friends ids are displayed as response when the user clicks on the friends\_id. 

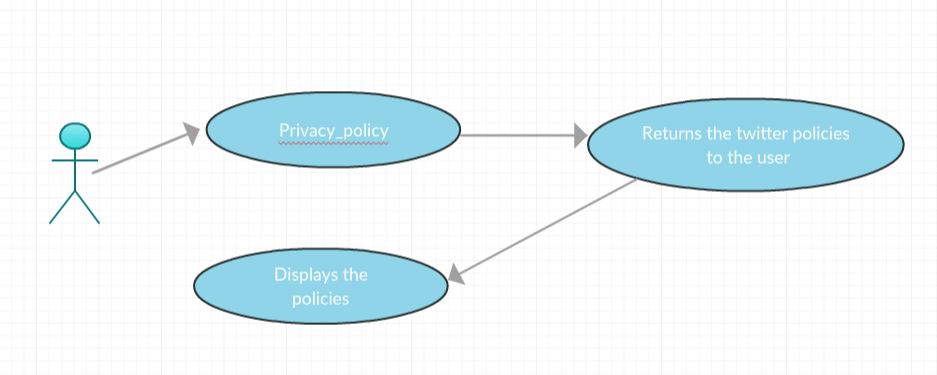
accounts\_settings:This gives the account settings of the user (4Spartans) , like screen name and other details on the account.



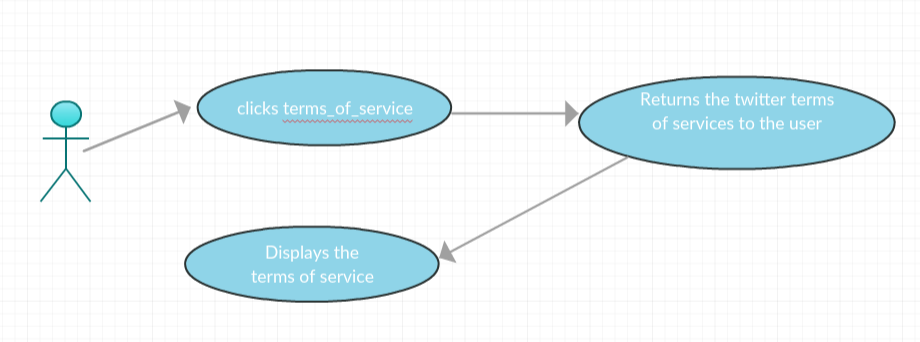
followers\_lists:Gives the list of users following 4Spartans

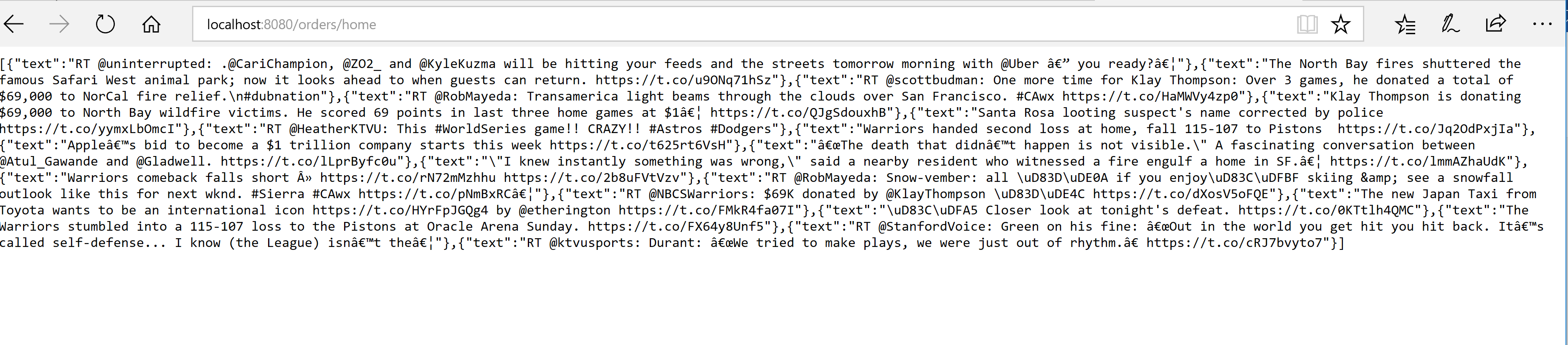
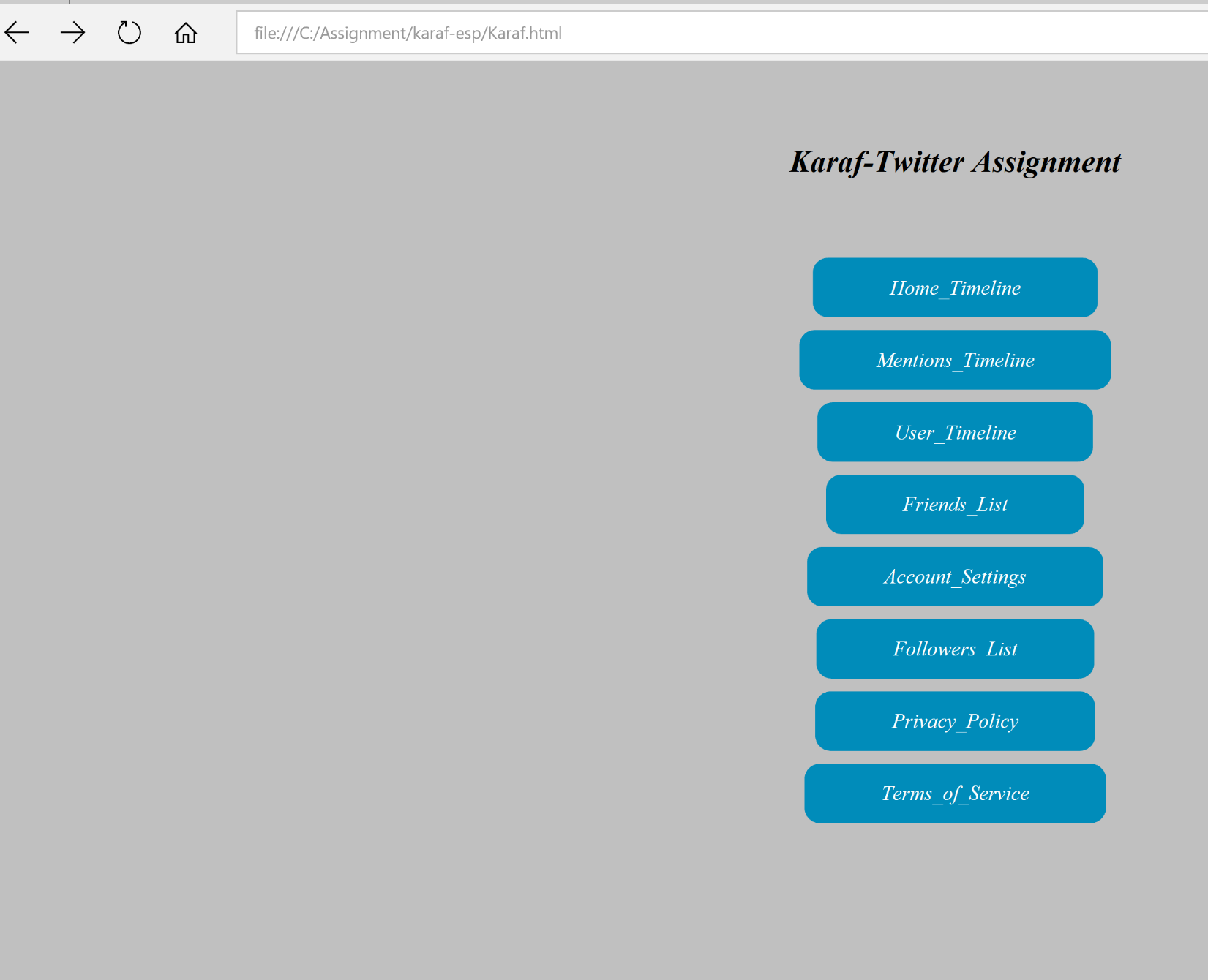


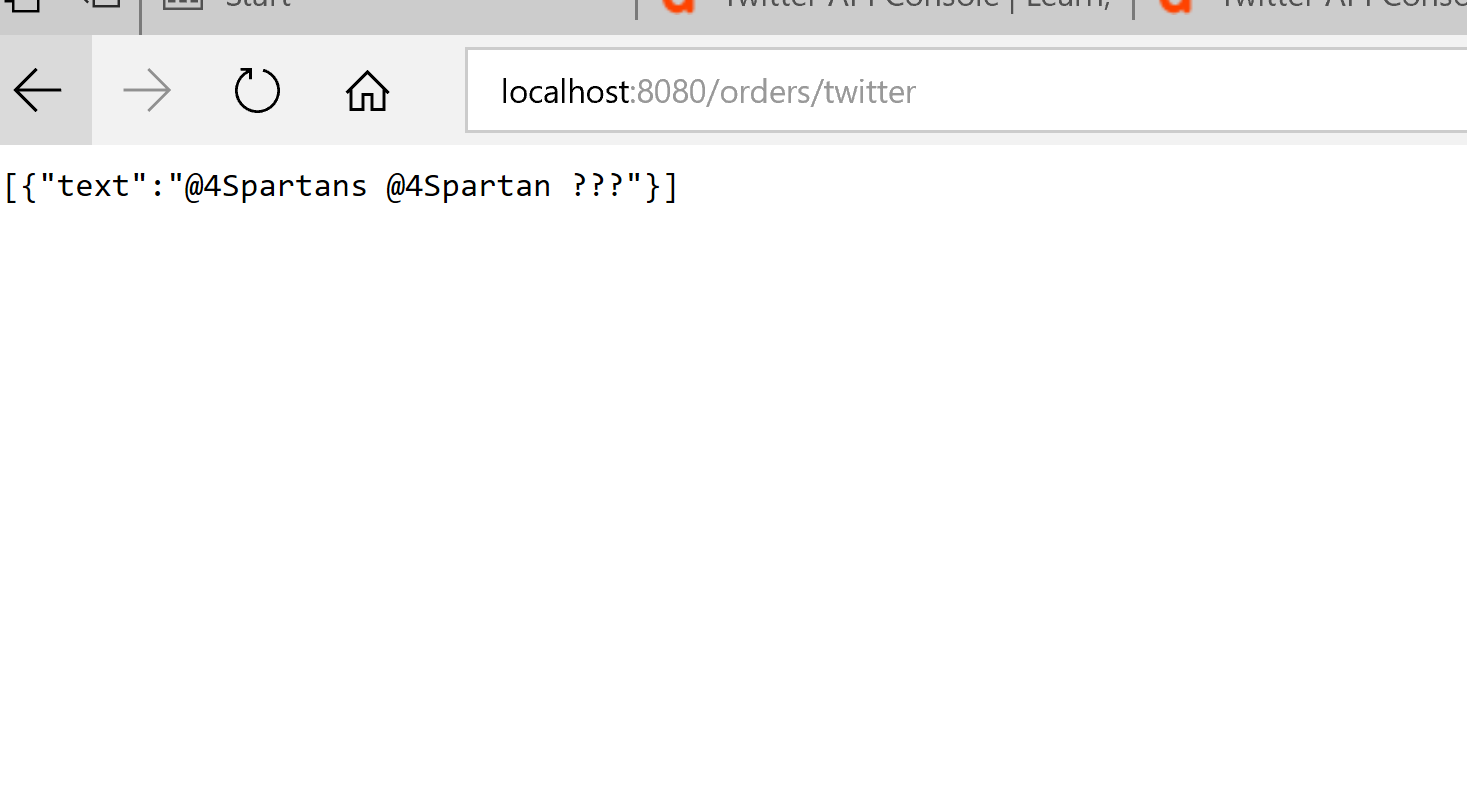
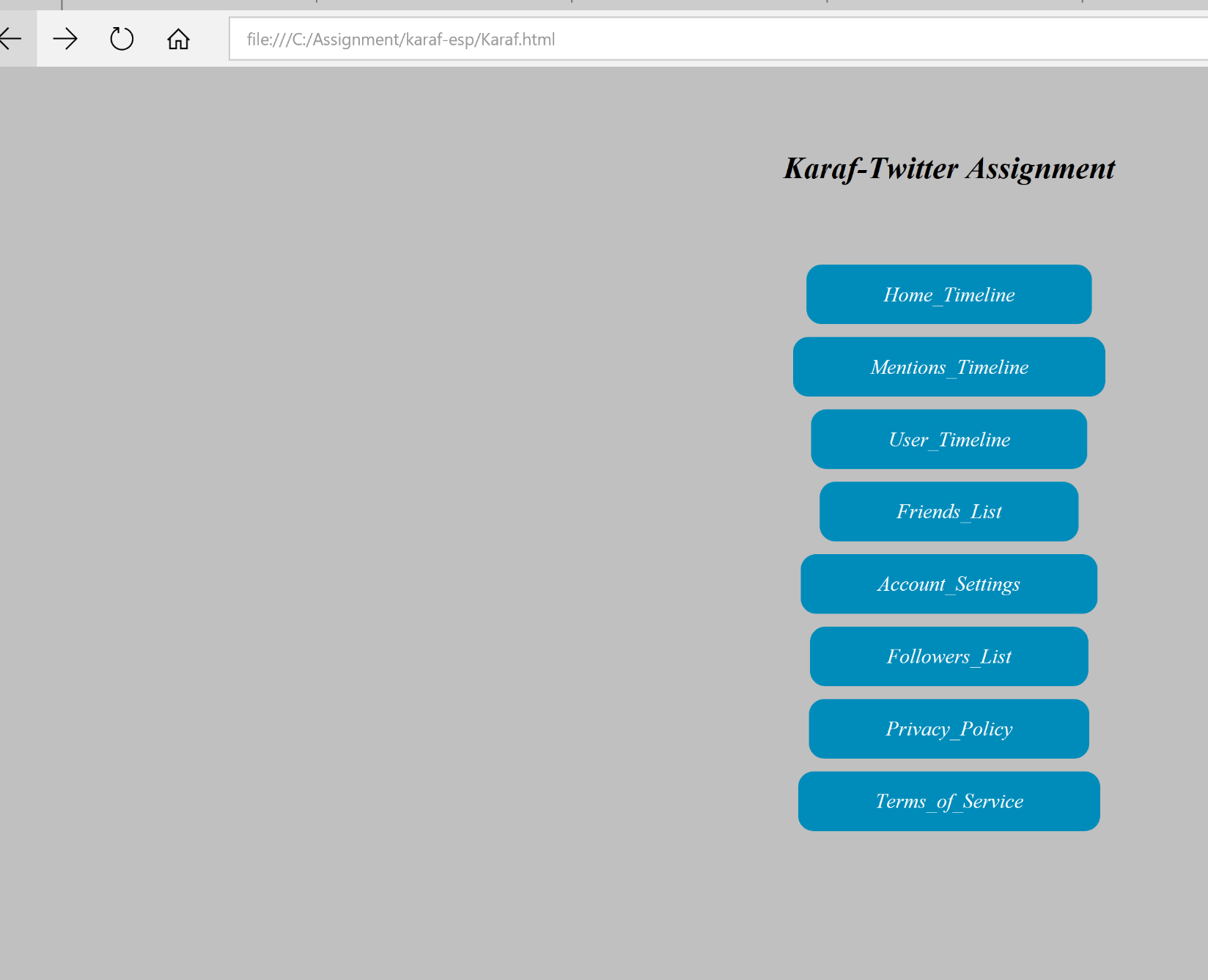
privacy\_policy:returns Twitters Privacy Policy



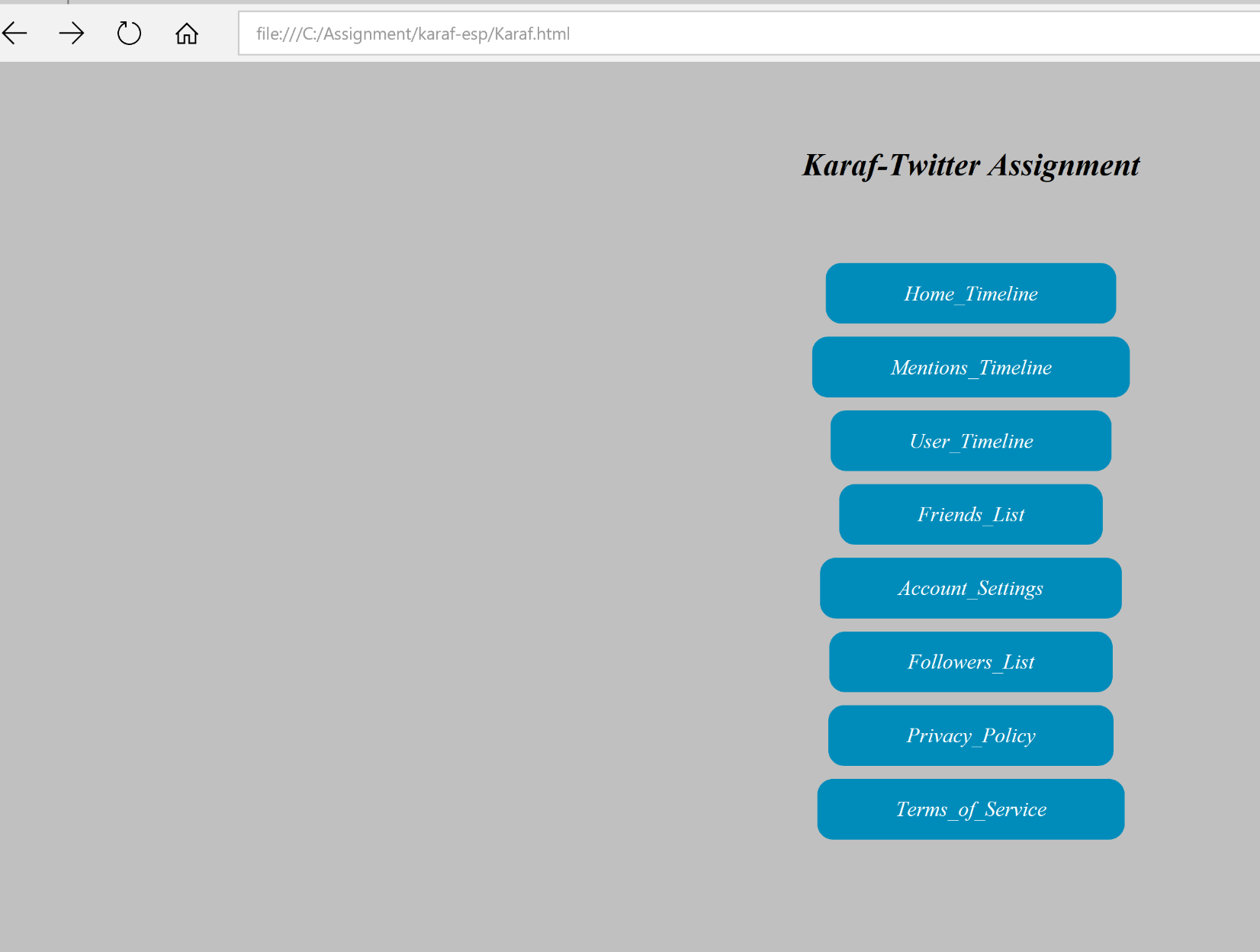
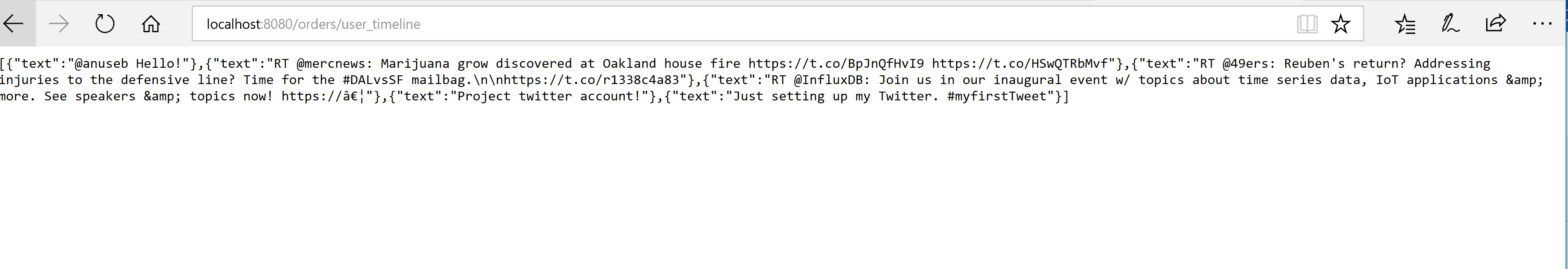
Terms of Service: Gives the details of the Terms of Service that Twitter has set.



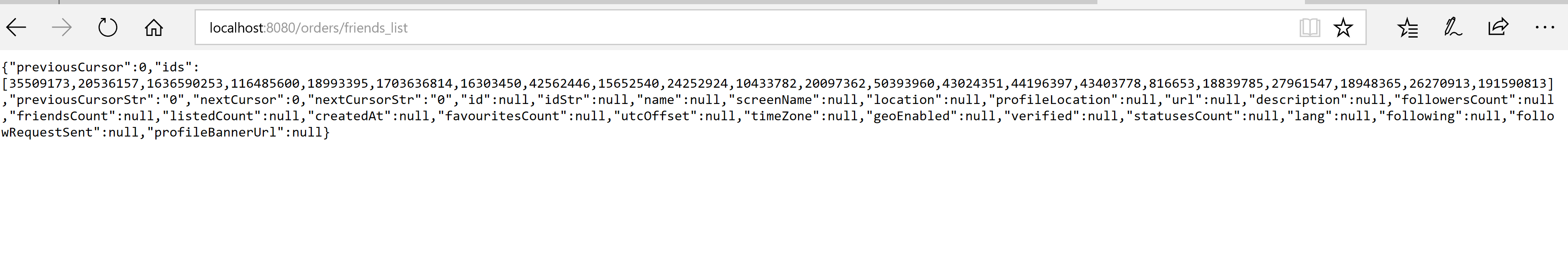
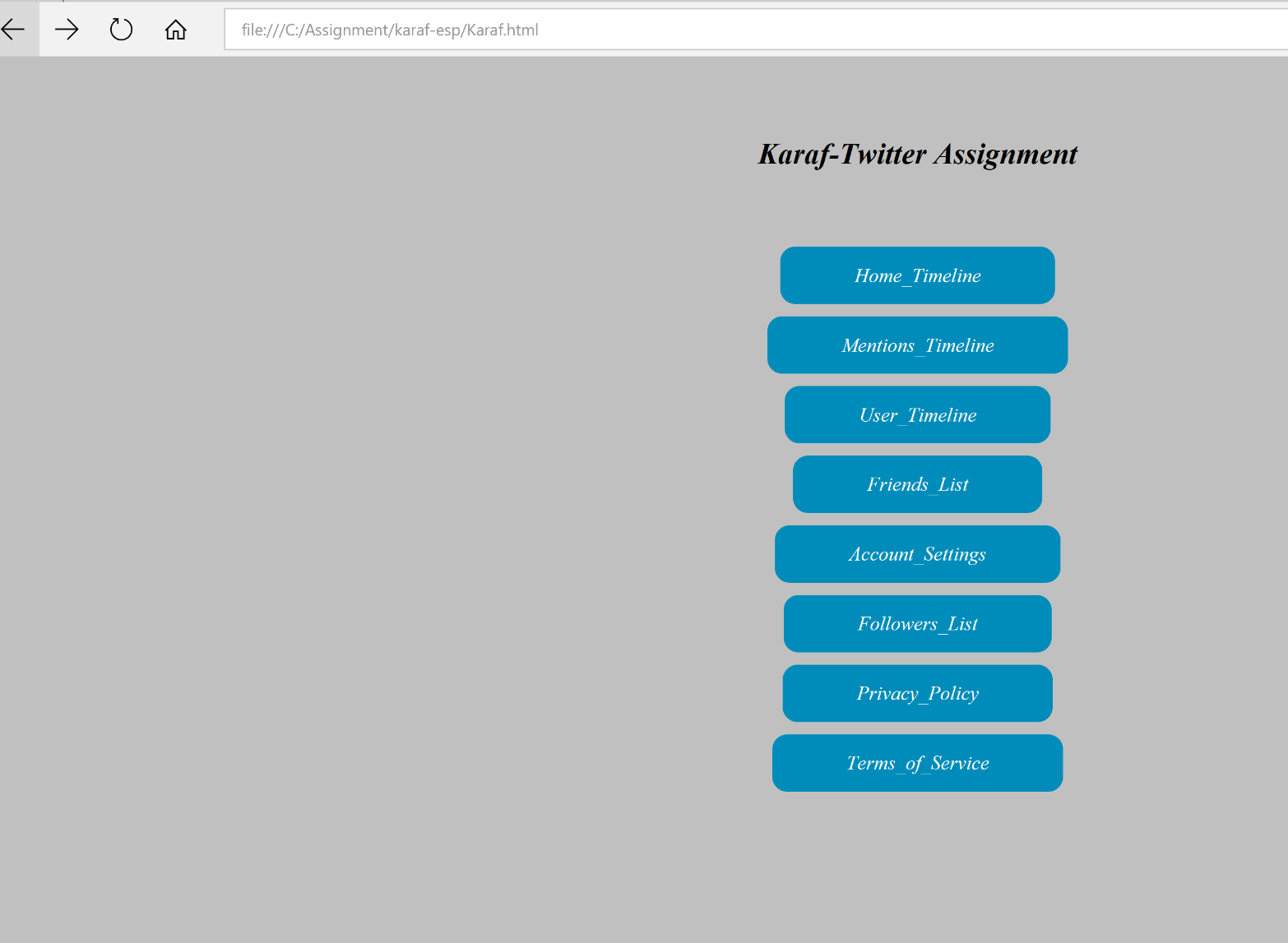
1. **Home\_Timeline** 
2. **mentions\_timeline**



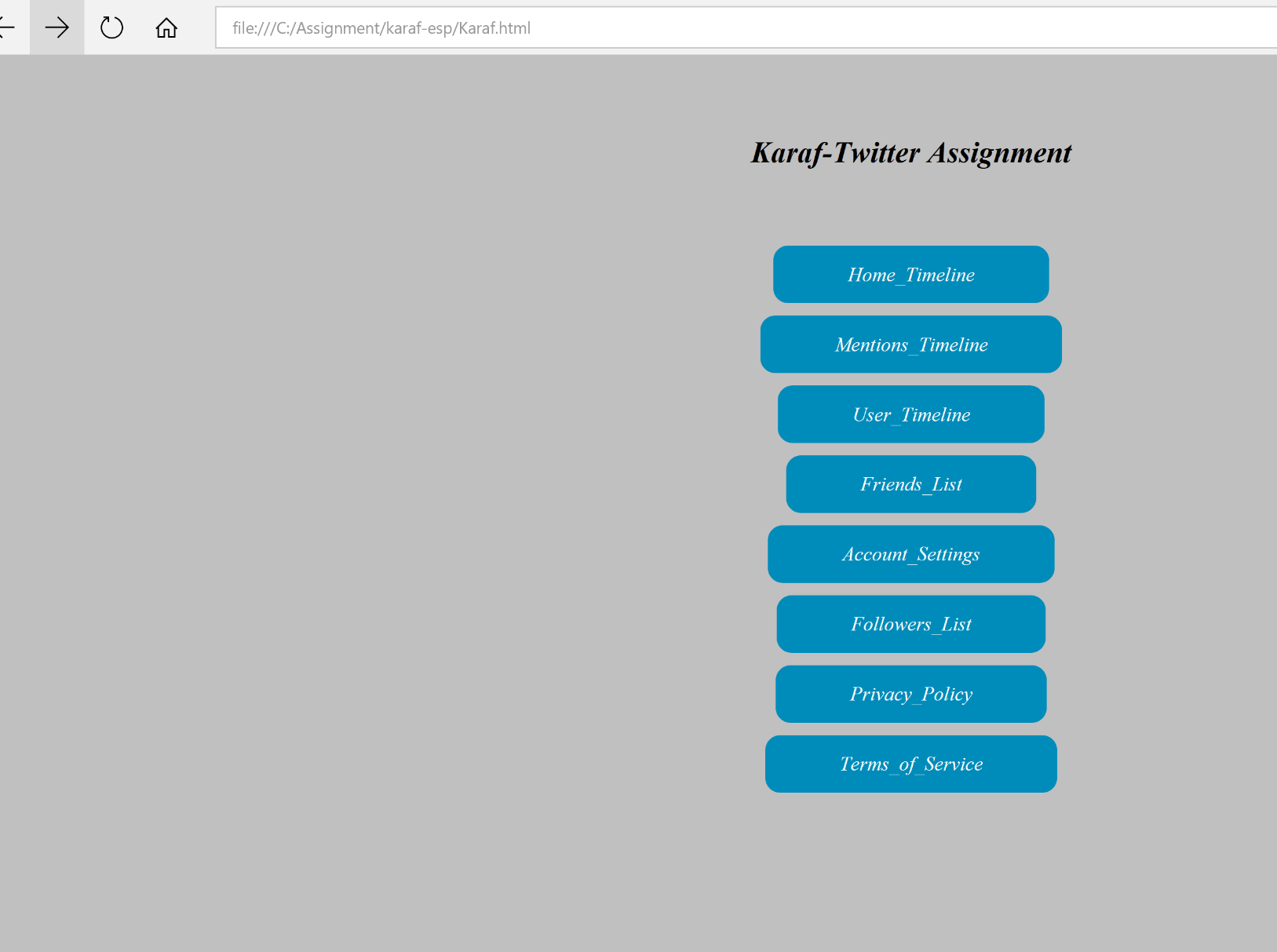
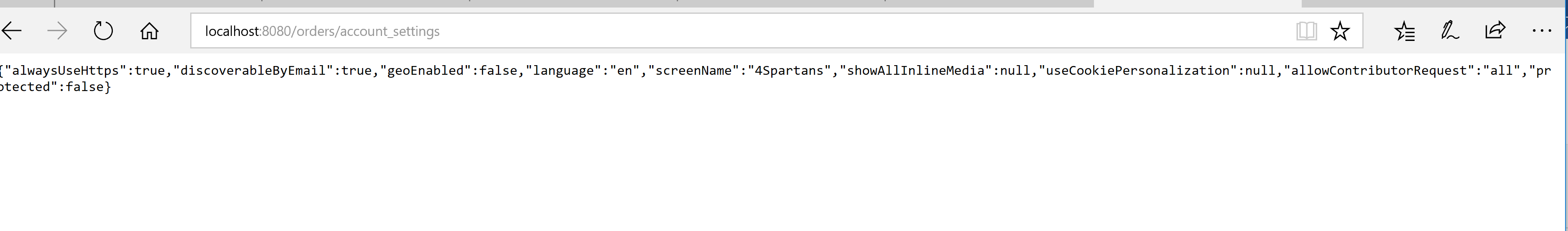
1. user\_timeline



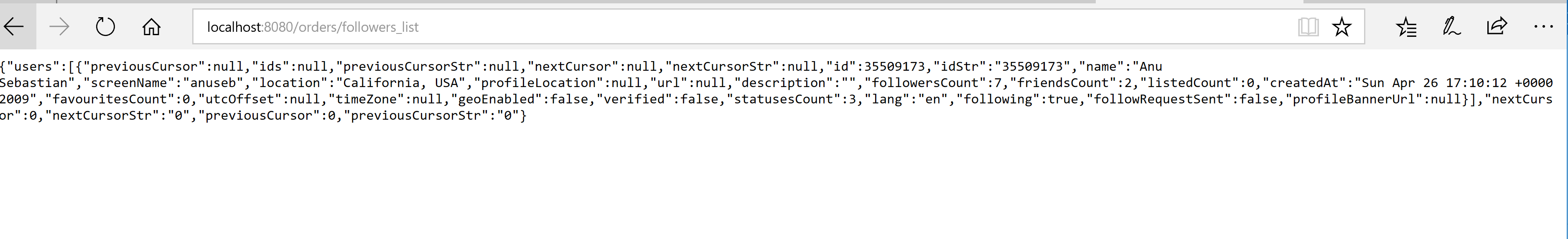
1. **friends\_list**

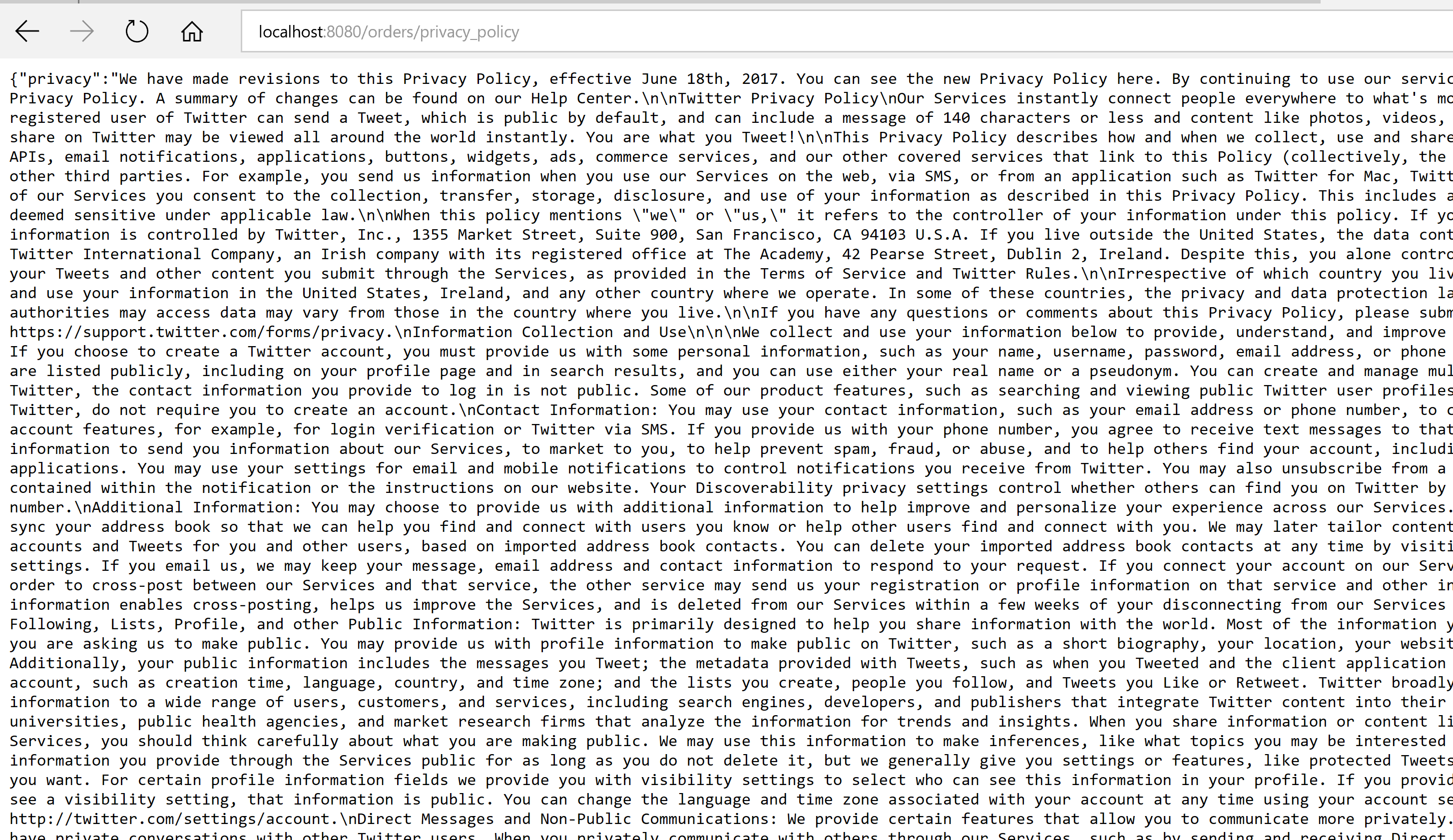
1. **account\_settings**



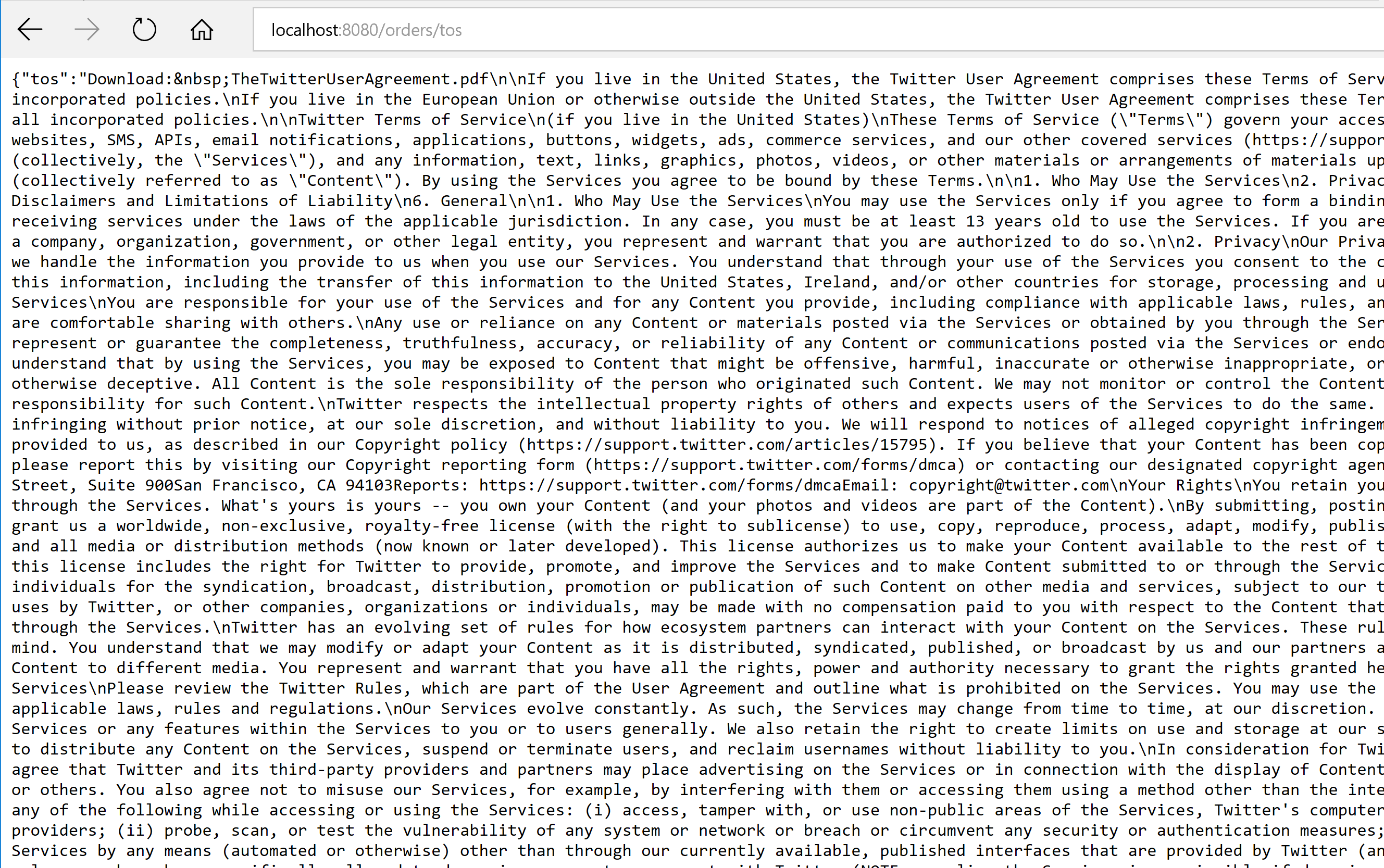
1. **Followers\_List**



1. **Privacy\_policy**



1. **Terms of Service**



**Test Cases**

TC#11: Verify that home\_timeline returns a non-empty and 200 OK response with the correct response data. - PASSED

TC#2:Verify that mentions\_timeline returns a non-empty and 200 OK response with the correct response data. -PASSED

TC#3:Verify that users\_timeline returns the a non-empty and 200 OK response -PASSED

TC#4: Verify that account settings of the user returns the account details of the user(owner). -PASSED

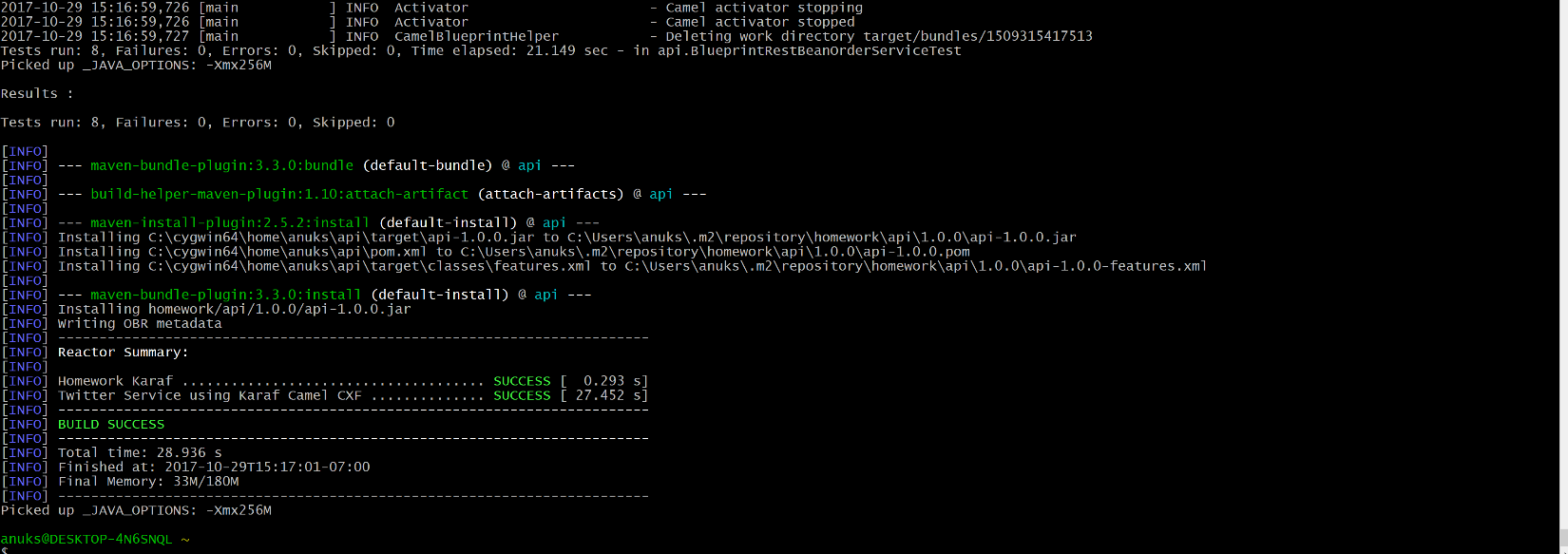
TC#5:Verify the friends\_ids returns the ids of the friends of the user. -PASSED

TC#6:Verify that followers list returns a non-empty and 200 OK response-PASSED

TC#7:Verify that privacy policy call returns twitters privacy policy. -PASSED

TC#8:Verify that terms of service returns a non-empty and 200 OK response-PASSED

**Screenshot of the TC’s during the maven build (under Tests Run)**



**References**

<https://developer.twitter.com>

<http://www.jsonschema2pojo.org/>

<http://docs.inboundnow.com/guide/create-twitter-application/#toc-0>