**README**

Web pages with behavior logging services assignment. The web page will store the user’s details like user name and his log in history. Another page displays the events that has been logged while the user was using the stackoverflow page.

**Getting Started**

The web application has been built on angularjs. The server-side API calls have been built on FLASK. The database that is being used by the application is SQLite. Additionaly to track and log events on the stackoverflow page, chrome extension has been used.

**Prerequisites**

For the application to run successfully, the environment requires

1. Python version 2.7.6
2. Google Chrome
3. Chrome extension

**Installing**

**For server-side programming:**

Install Python version 2.7.12 and Python libraries. The required libraries are as follows

*Example: pip install flask*

1. flask
2. Werkzeug
3. jinja2
4. itsdangerous
5. click

**Deployment**

**To start the server:**

Navigate to the folder where the project has been downloaded and navigate to \assignment1. Open the command prompt from this location and use command *python Endpoints.py* to start the server.

**To start the chrome on Developer mode:**

Chrome must be in developer mode to get the chrome extension running. To do that, open chrome and go to *chrome://extensions/*

**To add the chrome extensions:**

Since, this assignment is available over online and offline, there are two extensions that has been provided in the project folder,

1. assignment 1\extOnline
2. assignment 1\extOffline

Use these two folders accordingly. Use 1 while running the application on EC2 instance and use 2 while running the application on your local machine.

On *chrome://extensions/* , click Load unpacked extension… and load the extensions appropriately as mentioned above.

**To run the application:**

On local machine: run <http://localhost:8082> on browser

On EC2 instance: run <http://ec2-18-220-227-179.us-east-2.compute.amazonaws.com:8082> on browser

**Working of the application**

1. The starting page of the application is the index.html page. From this the user can navigate to login and register.
2. **The user has to be first registered to login.**
3. After registration, the user is redirected to the login page.
4. After login, the page is redirected to home page and displays the user name and his log in history. Additionally, there are four other buttons on home page namely,

* stackoverflow – redirects to stackoverflow page
* view event logs – A page that shows behavioral interaction logs
* why logged these events – A page that answers the question “why do you need to log these actions?”
* Logout

1. While on creation of the account, userDetails table gets created. While logging the events, userLogs table gets created.
2. View event logs page gets refreshed by clicking the back button on the page and then clicking the view event logs button on home page.

-------------------------------------------------------------------------------------------------------------------------------

**VERSION 1.1:**

This version of the project contains social visualization of the users. For running this version successfully, please do the following in addition to the previous version’s requirements.

**Installing**

**For server-side programming:**

*sudo apt-get install python-numpy*

*sudo apt-get install python-dev*

*pip install pandas*