**README**

Recommender System **Version 1.2** for stack overflow data set built on top of previous version of behavior logging website. For Recommender System, please navigate to **VERSION 1.2** section of the page.

**Getting Started**

**Version 1.0**: 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.

**Version 1.1**: This version of the project contains social visualization of the users.

**Version 1.2**: The web application has been built on angularjs. The server-side API calls have been built on FLASK. Scrapy is used for crawling the website. Elastic search has been used for indexing the content of the website.

**VERSION 1.0**

**Prerequisites**

For the application to run successfully, the environment requires

1. Python version 2.7.6

**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:**

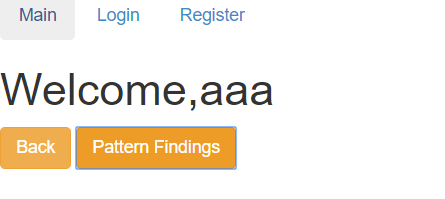
*sudo apt-get install python-numpy*

*sudo apt-get install python-dev*

*pip install pandas*

**Pattern findings**

Click on pattern findings to see explanation



**VERSION 1.2**

**Content-based recommender system** has been implemented in web application which is used for recommending similarity-based Java programming **wikibooks content/ Oracle The JavaTM tutorials** to the given dataset.

**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. elasticsearch
2. beautifulsoup4
3. pyes
4. Scrapy
5. ScrapyElasticsearch
6. spyder
7. urllib
8. urllib3

**Deployment**

**To start the elasticsearch server:**

Navigate to the folder where the project has been downloaded and navigate to \assignment 1\elasticsearch-5.6.3\elasticsearch-5.6.3\bin.

Then click the elasticsearch.bat file to start the elasticsearch server

**To start the server:**

Navigate to the folder where the project has been downloaded and navigate to \assignment 1

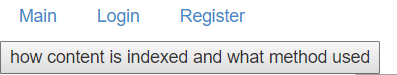
Open the command prompt from this location and use command *python Endpoints.py* to start the server.

**To run the application:**

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

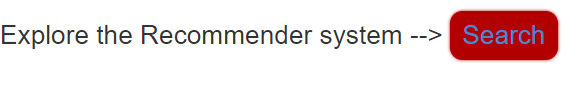
**To know how it is indexed and method used:**

Click on the how content is indexed and what method used button within the Recommender system page.



**Working of the application**

1. The starting page of the application is the index.html page. From this the user can navigate to Recommendation system.
2. By clicking on the Search button on the main page, it takes it to the recommendation system.



1. Once on the recommendation system page, the user can choose to crawl the data for the first time.



If not, the user can simply Click on the Recommender System button to show the system.



**IF ParsedData.csv file is opened, Please remember to close the sheet before running the crawler.**

1. when the crawler is run, then internally it crawls the wikibooks and java tutorial page. The crawled content is available at \assignment 1\ParsedData.csv. And then indexes it into the elasticsearch. Now an index is created in the elasticsearch. Hence, the user does not have to run the crawler each time he opens the application. He can simply start the recommender system to start searching for recommended content.
2. When the Recommender system is opened, the user can see the series of post. These posts are the dataset given.



1. On hover of the button, the content of each post is shown as a tooltip.
2. Click each post to show the content and then click on search to show the recommended content.
3. Custom search can also be done.
4. To know the method used and how indexing is done, click on *how content is indexed and what method used* button on top of the page

