**Smart Bridge Internship**

Project Report on

**AI Powered News Search App**

By

K Kruthendar Reddy

Malla Reddy Institute of Engineering and Technology

Hyderabad, Telangana, India

**INTRODUCTION**

1. **1 Overview:**

The web is home to massive amounts of data, with more being created every day. Organizations can harness this constant stream of information to gain understanding, plan strategies, and find opportunities. Enriched news data can help your application make dynamic connections across current events faster.

* 1. **Purpose:**

The purpose of this project is to develop a user-friendly web application which makes it easier for the users to find relevant and recent news articles according to their interests and queries.

All we need to do is give a keyword regarding the required news to the chatbot so that it can fetch the news articles and links. The web application also analyses the sentiment present in the article.

**LITERATURE SURVEY**

1. **1 Existing Problem:**

The news applications that are currently available can be confusing, and contain huge data which is inappropriate to the user to search for a particular article from a substantial amount of data.

It is also not possible to determine the tone or the sentiment of the required news using the apps that are available.

* 1. **Proposed Solution:**

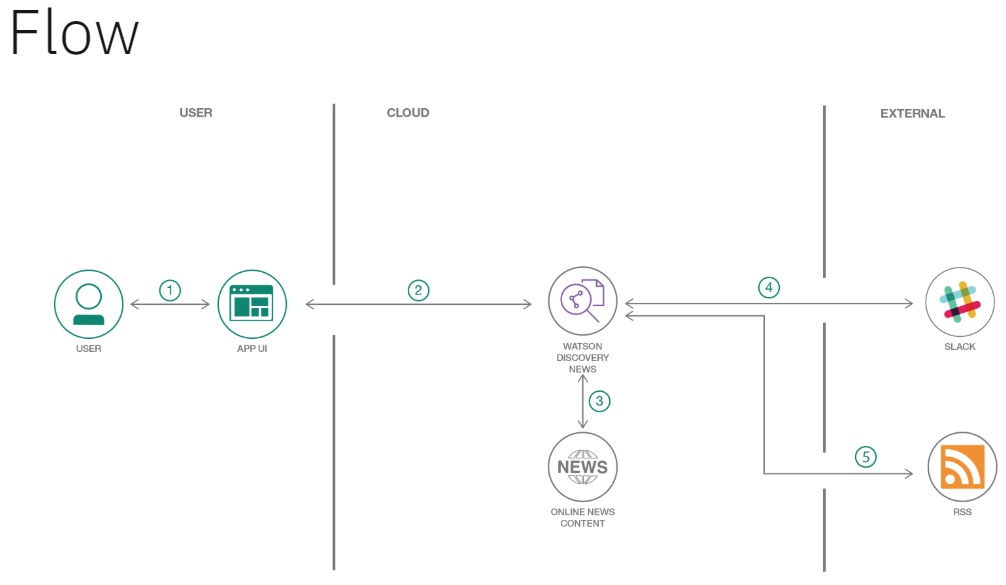
A news search web application which is created by using IBM discovery service available in IBM cloud.

This IBM discovery service when integrated with Node Red Flow, it creates a simple, organized, an easy to access user interface i.e. news hunting application which fetches the user with relevant news from a vast data since IBM discovery service always updates itself with the latest news.

By adding sentiment analysis to this application can make the app UI more unique and user friendly.

**THEORITICAL ANALYSIS**

**3.1 Block Diagram:**



The block diagram shows that the user interacts with the UI of the app created by the Node Red Flow. The UI is integrated with the Watson Discovery News which keeps updating itself with the latest news from the internet. This when integrating with slack bot makes an interactive news bot which fetches the relevant news according to the query of the user.

**3.2 Software Designing:**

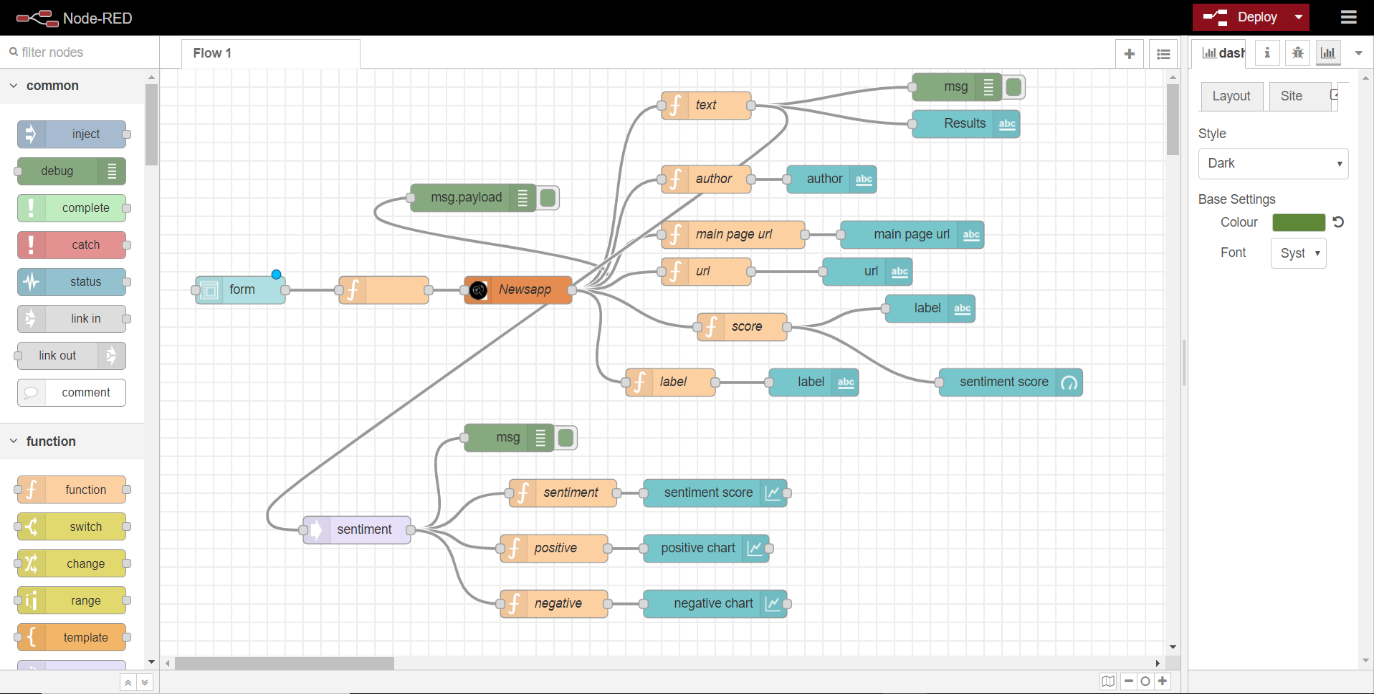
In order to build this app, the following things are need to be done:

1. Build a Server Side Application using Node-RED.
2. Use the pre-built Watson Discovery News collection.
3. Access the Watson Discovery Service through the Discovery API.
4. Use a Slack interface to query the data.
5. Push news alerts out to web notification.
6. Deploy the app on IBM Cloud.

**Setting up the development environment**

1. Install Git and create a GitHub account. Git installation can be done through the following website <https://git-scm.com/downloads>. Once this is done, open <https://github.com/> and create a git account.
2. Open <https://slack.com/intl/en-in/> and create a slack account and a channel.
3. Open <https://cloud.ibm.com/registration> and create an IBM Cloud account. Then start creating a Node-RED application.
4. Create and deploy Watson discovery news app locally.
5. Integrate Watson Discovery Service to the Node-RED application.
6. Install Node.js from <https://nodejs.org/en/download/>.
7. Open the slack channel and create a bot application.
8. Integrate slack bot with Watson discovery news app.

The node red UI can be made by using the debug node, function node, form node, sentiment node and a few chart and label nodes and connecting them as shown below.



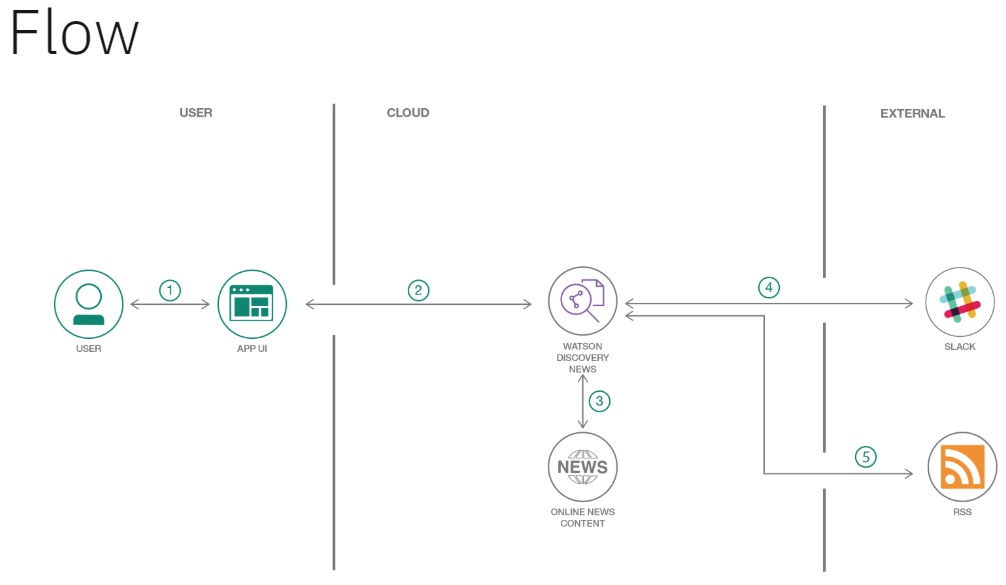
**EXPERIMENTAL ANALYSIS**

First we create the node red application by creating a flow using 5 simple nodes. This UI is then integrated with Watson Discovery Service. The user can now enter the query in the UI and get the relevant news from the Watson discovery (which collects the data/news from the net and keeps updating itself) and also know the sentiment of the news (if it’s positive or negative).

Then slack bot is integrated with Watson Discovery Service to make it more interactive and user friendly where the user can directly ask the bot to fetch a particular news using the relevant keywords.

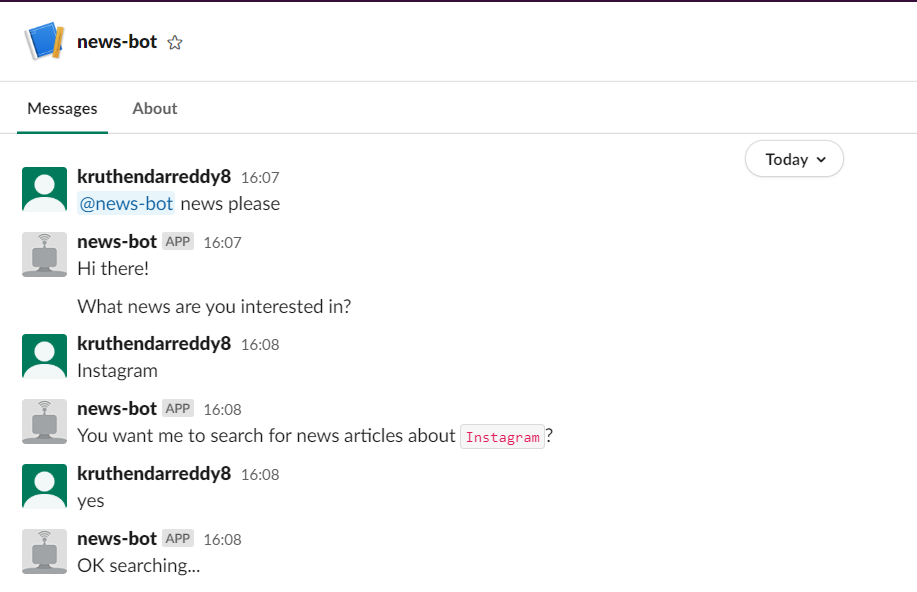
User can also search for the news in the application which will be available in browser at localhost - http://localhost:3000.

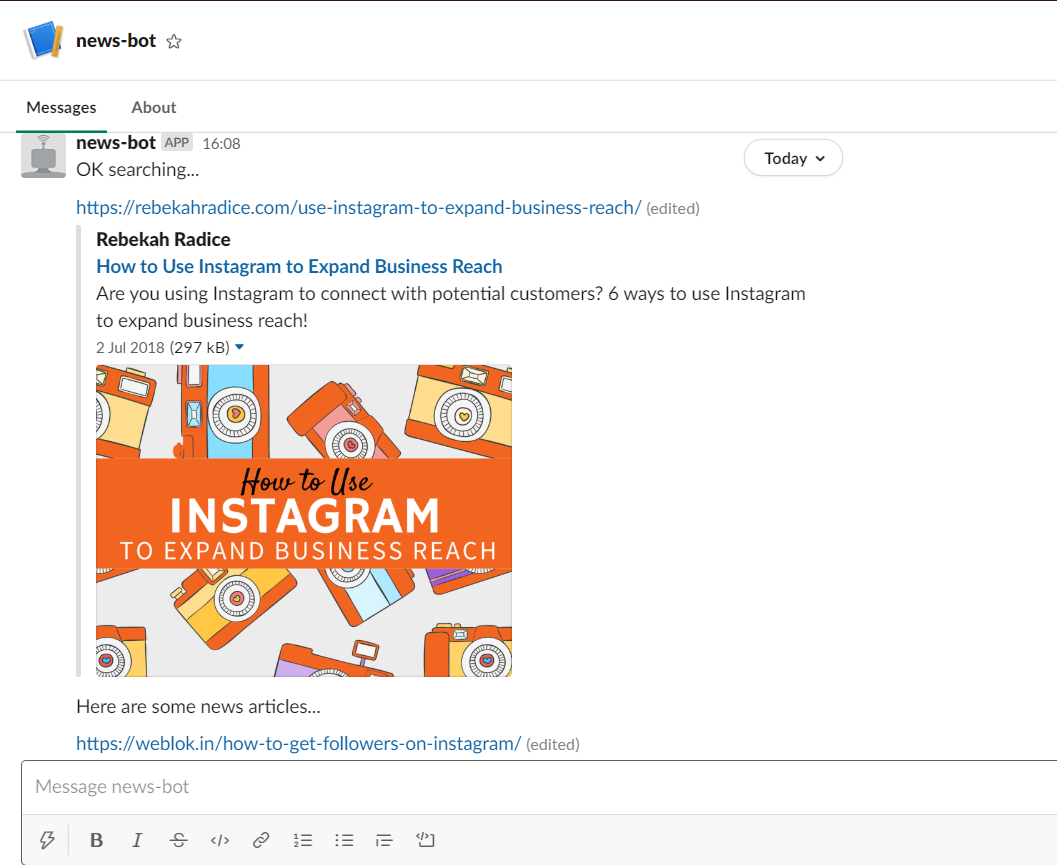
**FLOWCHART**

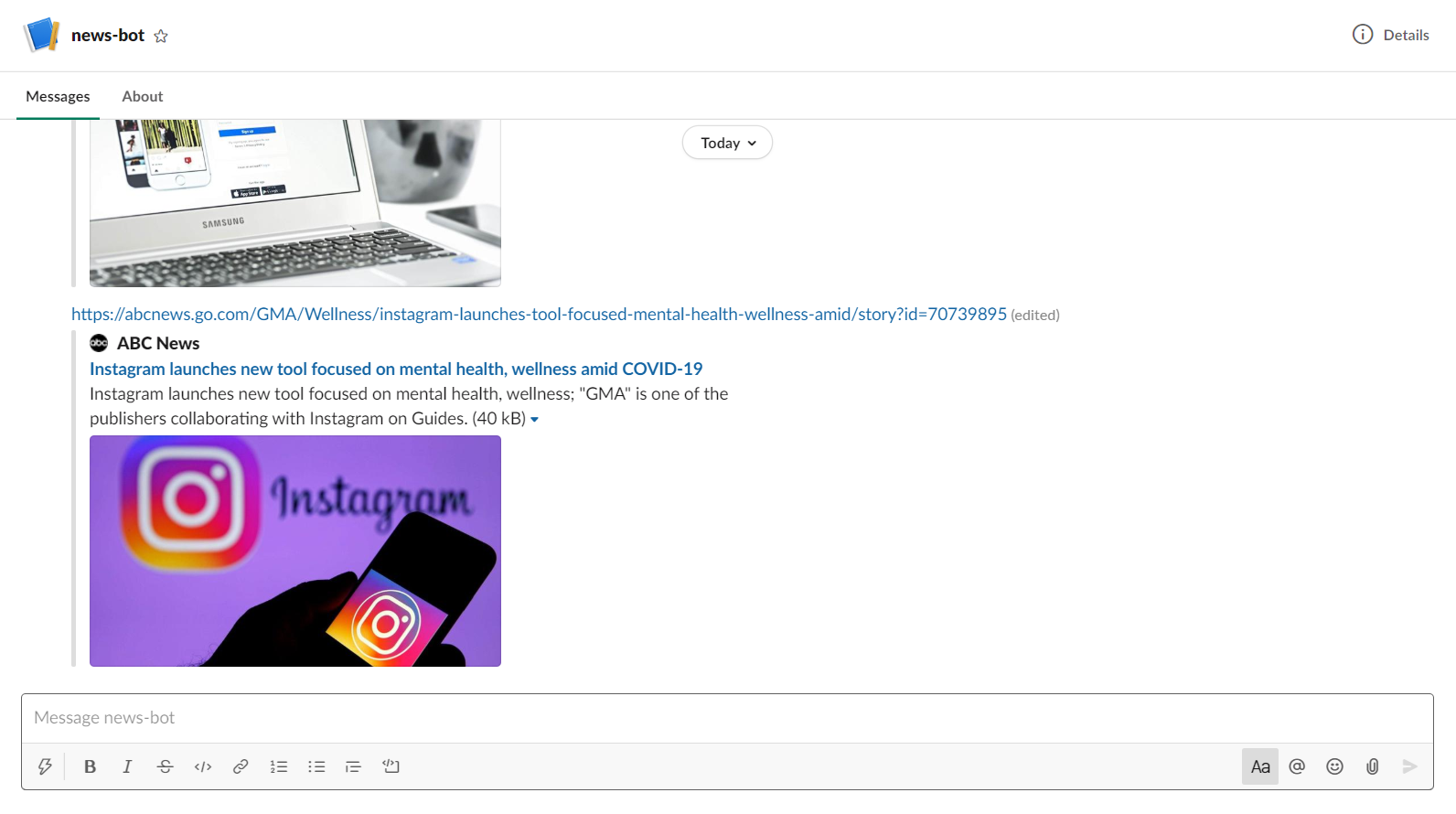
****

**RESULT**

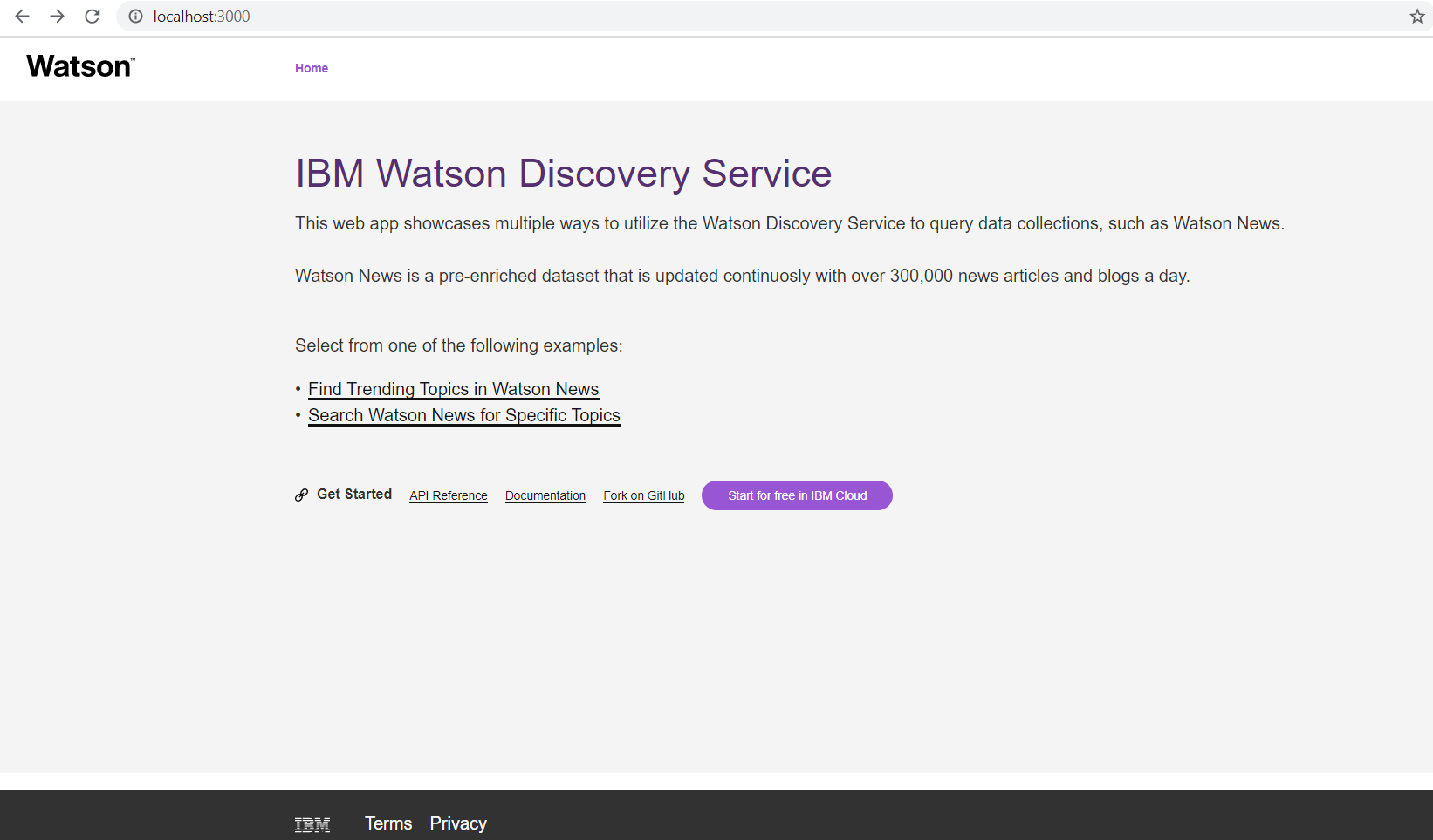
**Slack bot output:**

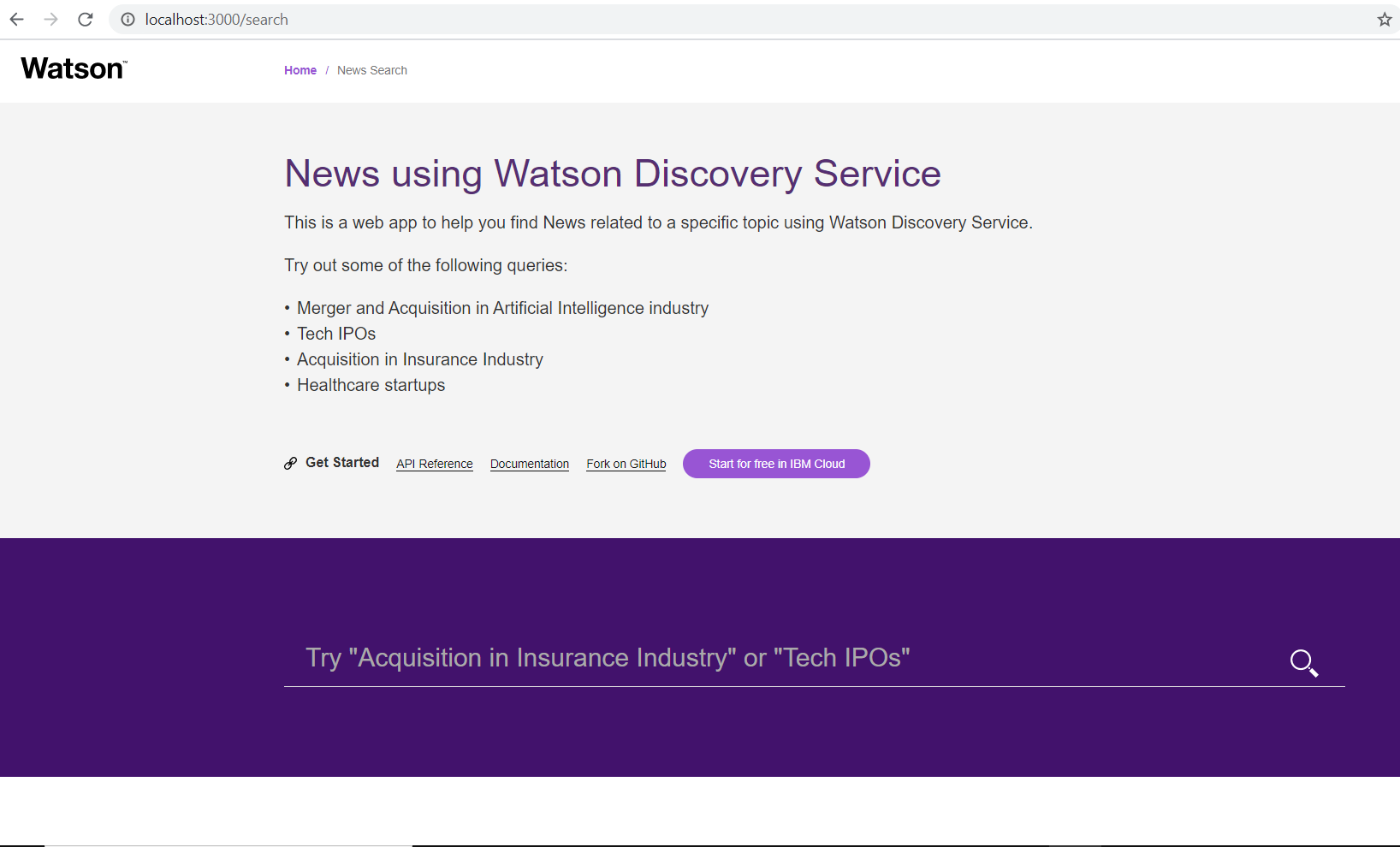


****

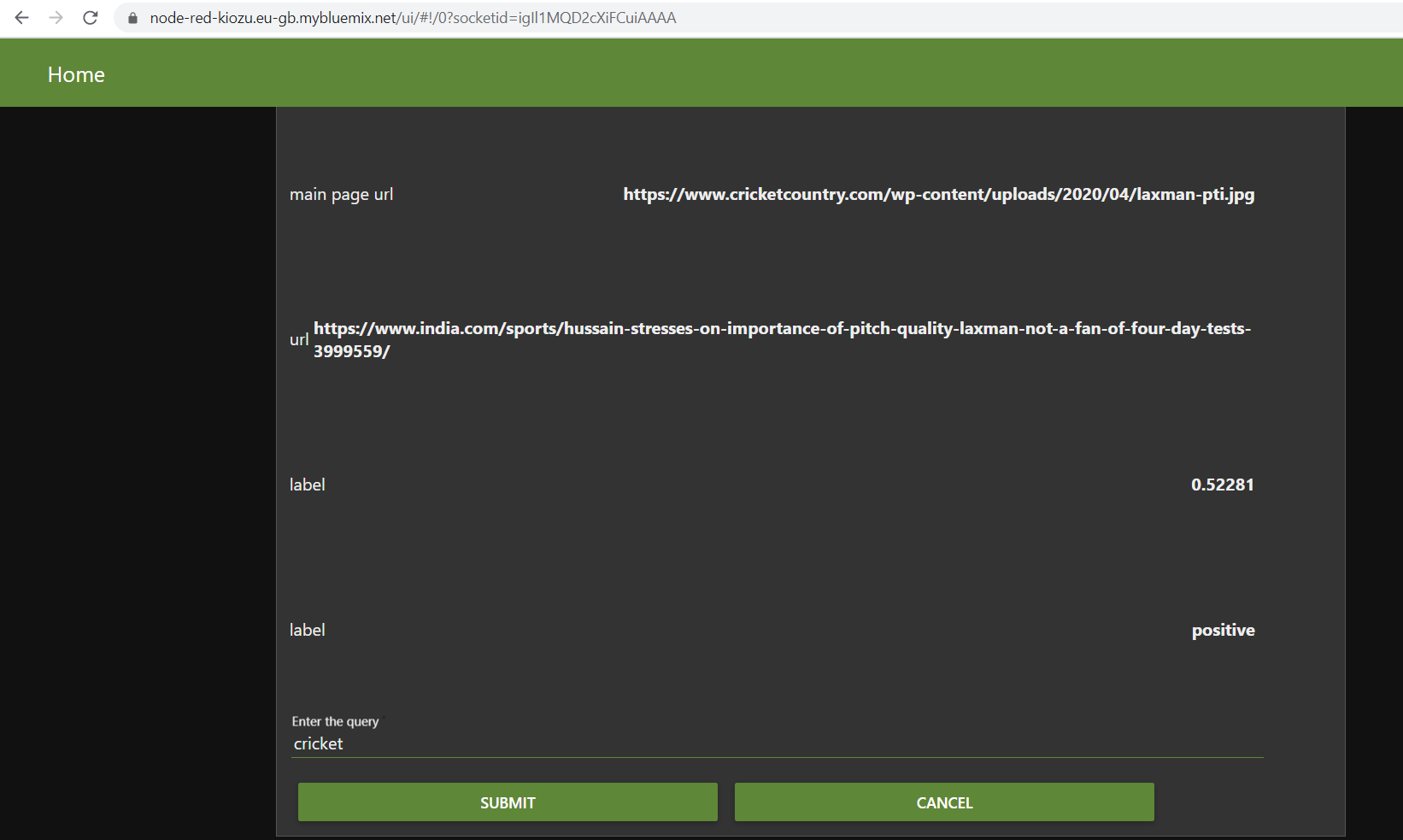
****

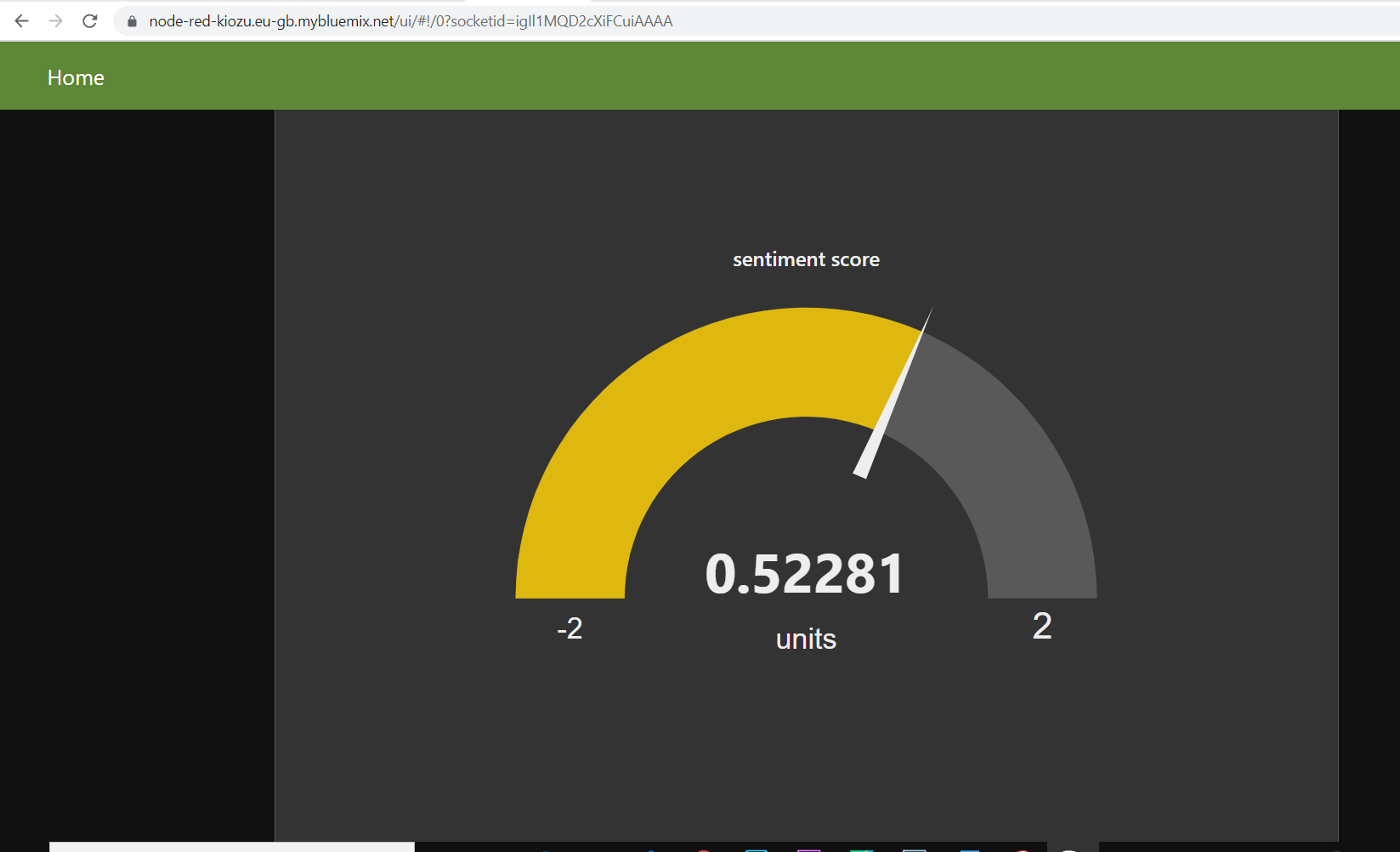
**Localhost sample:**

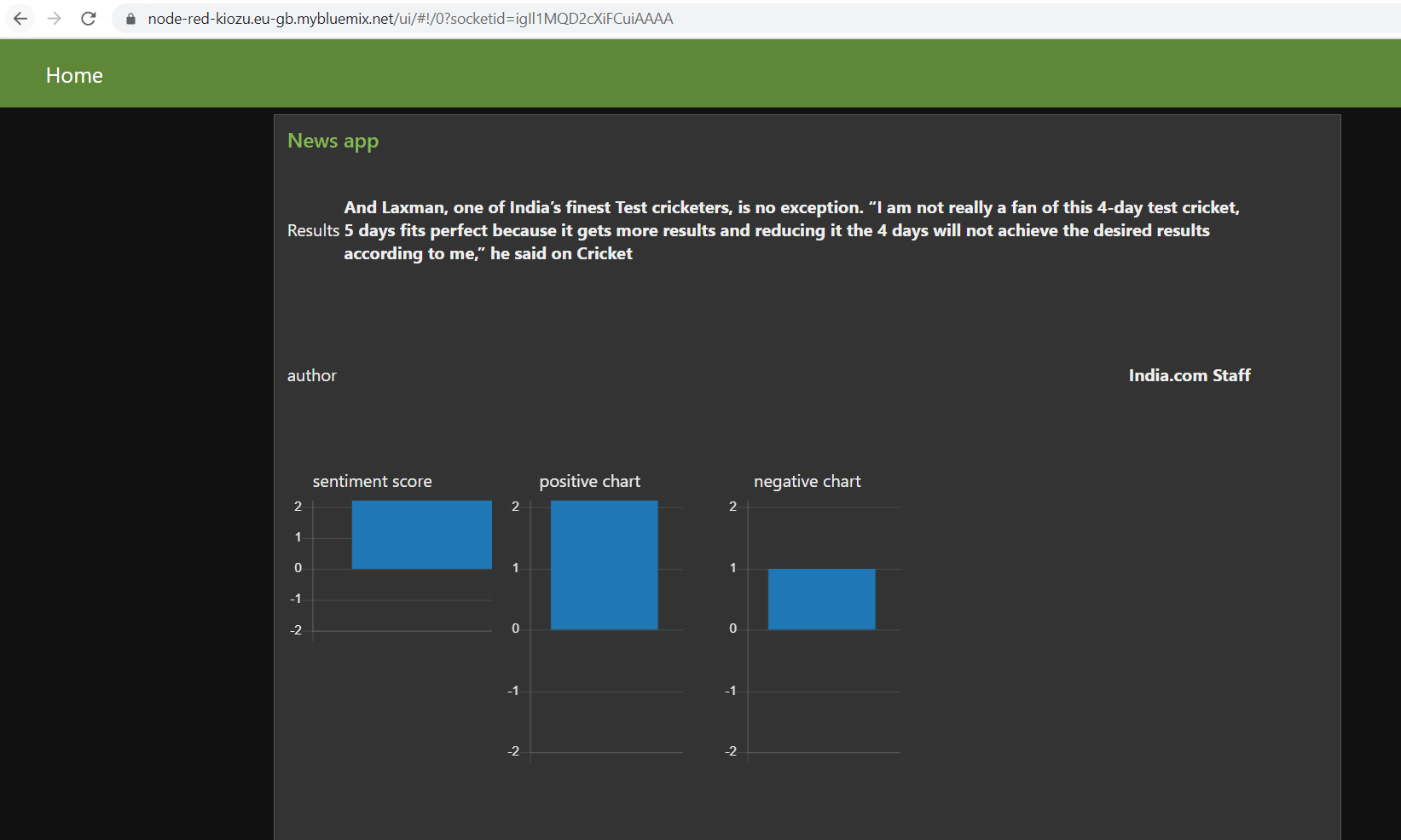
****

****

**Node Red UI output:**

****

****

****

**This can be seen in the below link:**

<https://node-red-kiozu.eu-gb.mybluemix.net/ui/#!/0?socketid=igIl1MQD2cXiFCuiAAAA>

**Git link:**

<https://eu-gb.git.cloud.ibm.com/kruthendarreddy8/NodeREDKIOZU.git>

**ADVANTAGES AND DISADVANTAGES**

* Provides an easy user friendly application to get news related to the topic that the user wants unlike the available apps which are difficult since the users need to search the required news on their own from a vast data.
* Sentiment analysis to make it more unique.
* Interactive news bot that searches and fetches news for the user.
* The bot understands only a certain fixed keywords.
* Does not have a single full-fledged app.

**APPLICATIONS**

* Can be used by companies that involve following the latest news.
* Can be used to make predictions in stock market.
* Can be used by anyone all around the world to get fast and accurate results.

**CONCLUSION**

In this project we’ll get to know the basics and build our own news mining web application using Node-RED / Python Web App and the IBM Watson Discovery Service. It helps us to know how things work in the corporate world. We learn how to use slack. It also makes us aware of the basic applications of AI.

**FUTURE SCOPE**

* Can be made into a full-fledged single mobile application.
* Can make the bot a little more advanced.
* The UI can be made more interesting by showing the latest news and adding more aesthetic front end designs.

**BIBLIOGRAPHY**

* <https://github.com/>
* <https://slack.com/intl/en-in/>
* <https://cloud.ibm.com/>