

# Data Processing & Management (DPM)

## NEWS SEARCH SYSTEM

GROUP NUMBER - 17

NIHARIKA PODDAR (MT21057)

SHUBHAM JAIN (MT21091)

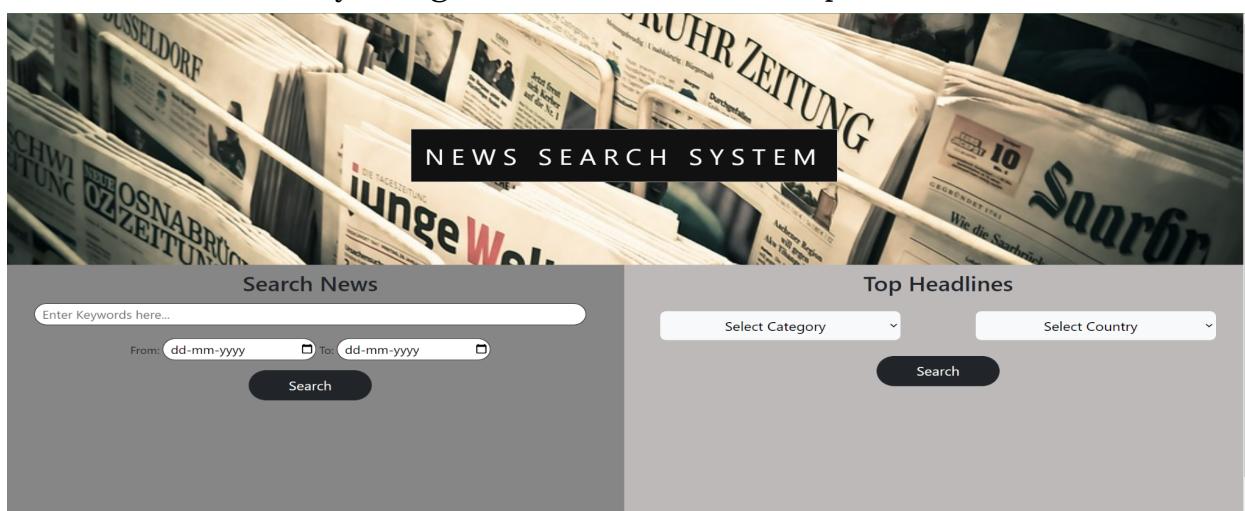
PARUL JAIN (MT21064)

### FEATURES :

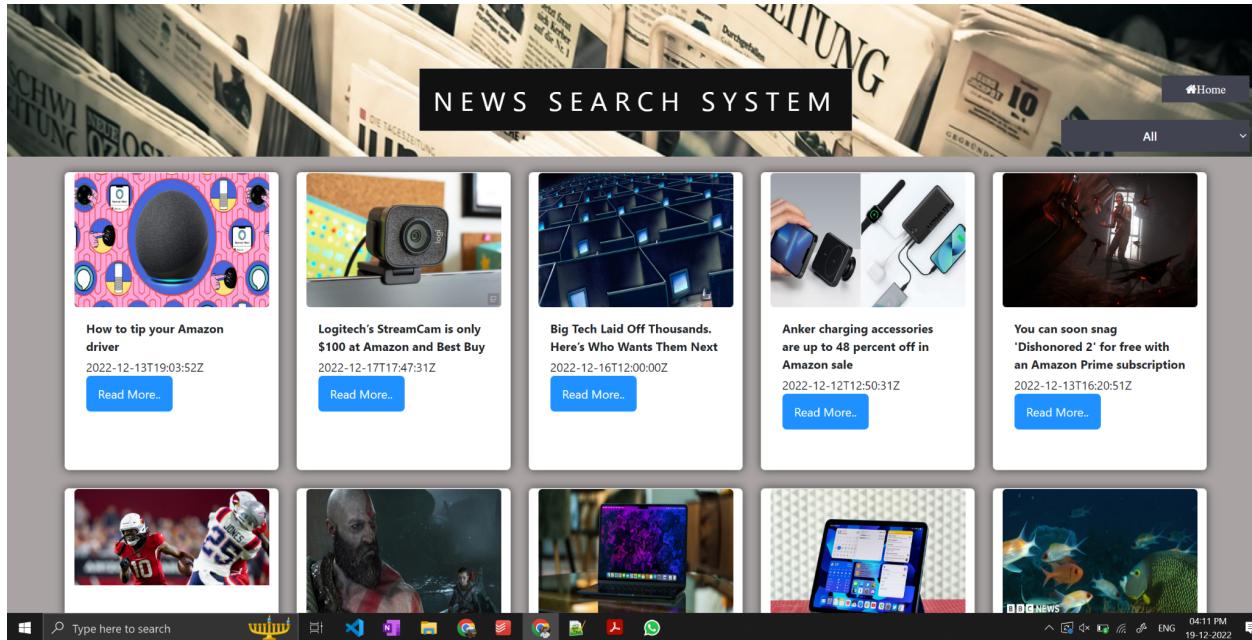
1. Today's Top Headlines based on
  - 7 Categories
  - 50 Countries
2. News Search based on
  - User Input Query/ Keyword
  - Start date
  - End date

### WEB PAGES :

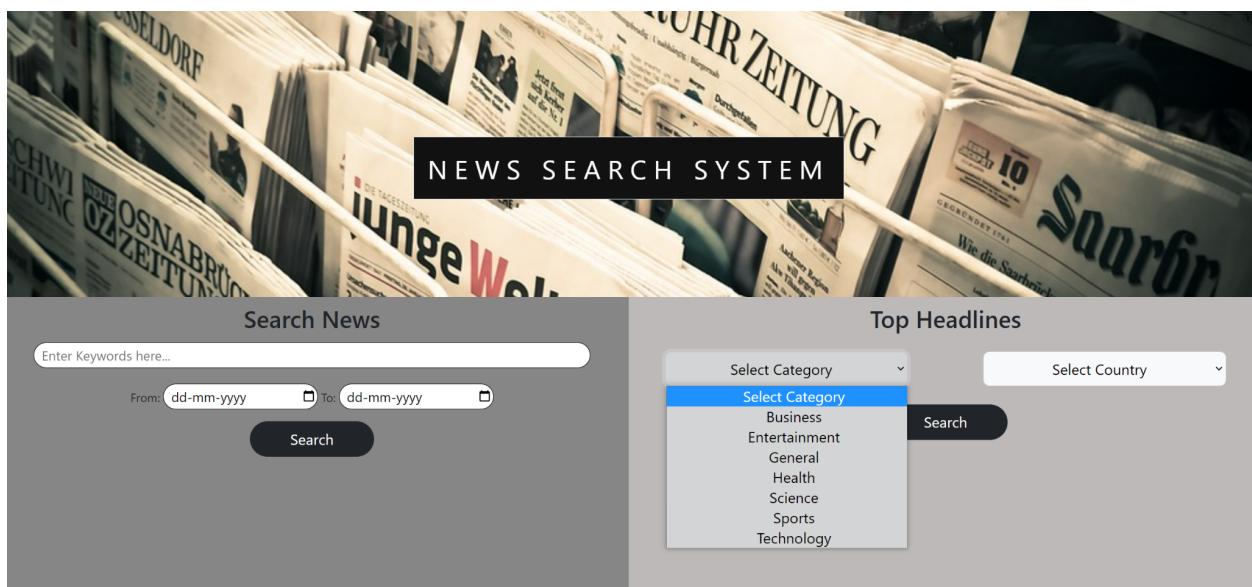
1. **Home Screen** - This page shows the home page of our website, which takes the input in two ways-
  - Search via a keyword/ Query
  - Search by categories and countries dropdown



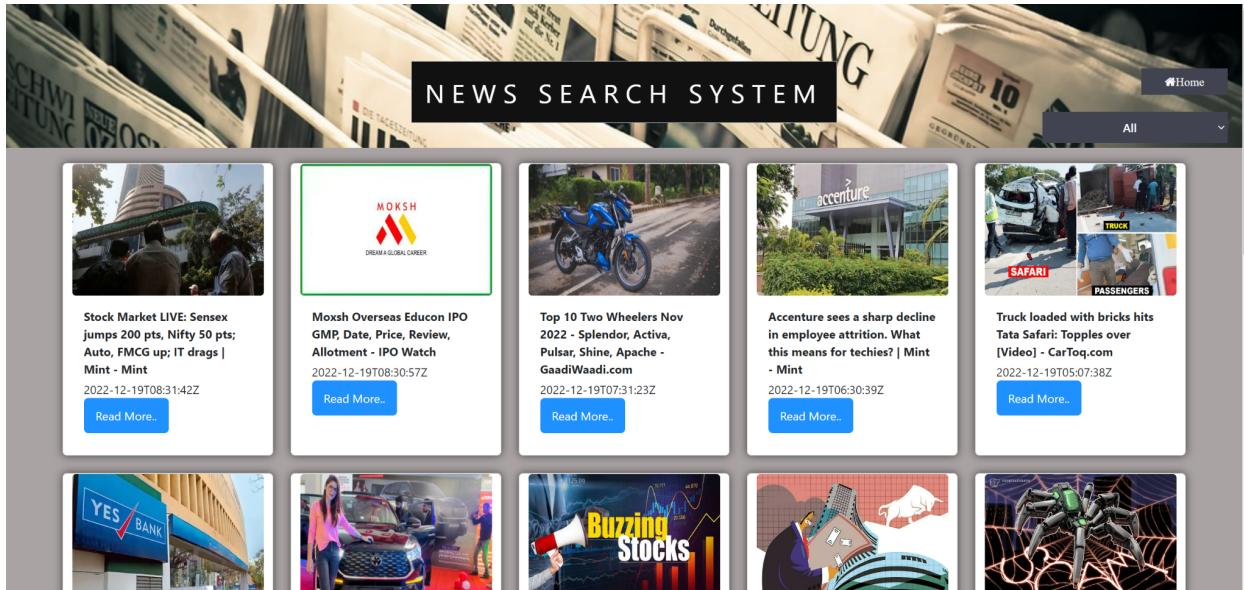
**2. News Page based on Search query** - This page shows all the news fetched from API based on some input keyword entered by the user.



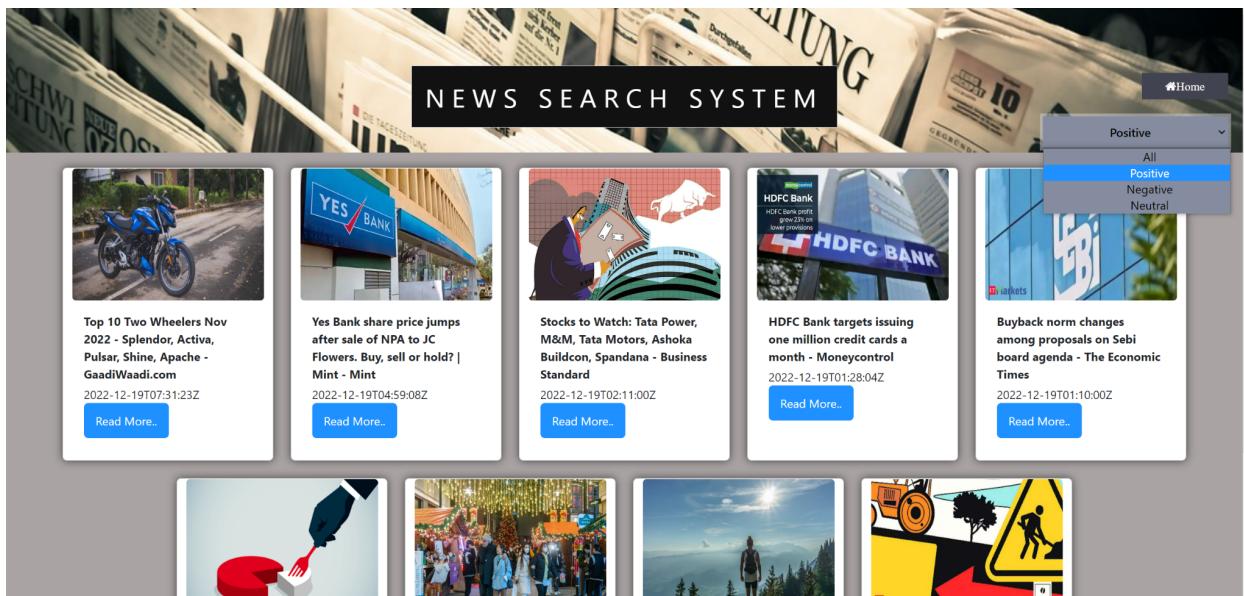
**3. Top Headlines**- It gives a dropdown list for categories and countries.



4. News Page based on Categories - This page shows all the Top headlines fetched from API based on category and country selected by the user.



5. News Based on Sentiment - This page shows the top headlines based on the sentiment of the news. It uses a model to classify the news into three categories that are positive, negative, and neutral.



## **How to Use the Website :**

Link: <http://143.244.135.235:8033/>

Users can directly interact with the website and utilize it to search the news based on the keywords and also based on the top headlines. Also, a user can filter the resulting news based on sentiments - positive or negative or neutral. For every news, there is a read button, with redirects user to the news page and user can read whole news from that page.

Users need not have to do any pre-processing steps or any kind of login/signup to use the application.

## **Contribution:**

### **Niharika-**

- learnt and implemented sentiment analysis for news
- Implemented News Search Page for displaying the news Results
- Flask Deployment

### **Parul-**

- learnt and implemented sentiment analysis for news
- Implemented home / index page for the application with the search and select functionalities
- Implemented Top Headlines Page for displaying the news Results

### **Shubham-**

- Flask development and Deployment
- Helping with Displaying Dynamic News Results
- Learning and Integration of News API

## **Technical Documentation**

### **1) Templates**

#### **a) Index.html -**

- i) It is the home page to the Application. It enables two features to be used 1) Searching of news based on the keywords and dates 2) Top headlines based on Categories and Countries.
- ii) The technologies used to create this webpage are HTML , CSS and JS
- iii) Jinja Template is used to read the data from backend sent via flask

#### **b) Newssearch.html**

- i) This page renders the news that are being fetched based on the keywords provided as input by the user
- ii) Uses HTML,CSS,JS
- iii) Jinja template and JS together are used to display the data dynamically on the UI

#### **c) Topheadlinespage.html**

- i) This page renders the news that are being fetched based on the keywords provided as input by the user
- ii) Uses HTML,CSS,JS
- iii) Jinja template and JS together are used to display the data dynamically on the UI

### **2) App.py**

#### **a) home :**

- i) Url : /
- ii) renders the index.html page

#### **b) searchnews:**

- i) Url : /searchnews

- ii) Receives the search parameters from the index.html page
  - iii) Provide search data to the news api and retrieves results
  - iv) Render the newssearch.html page and Pass the news data received from the api to it
- c) Topheadlinesmethod:
- i) Url : /topheadlines
  - ii) Receives the headlines parameters from the index.html page
  - iii) Provide search data to the news api and retrieves results
  - iv) Render the topheadlinespage.html page and Pass the news data received from the api to it
- d) filternewsmethod:
- i) Url: /filternews
  - ii) Receives the filter parameter from the newssearch and topheadline page
  - iii) Perform sentiment analysis on the news data
  - iv) Filter the news based on the filter parameter
  - v) Returns and renders the filtered news data back to the page requesting the filter