M.KUMARASAMY COLLEGE OF ENGINEERING
NAAC Accredited Autonomous Institution
Approved by AICTE & Affiliated to Anna University
ISO 9001:2015 & ISO 14001:2015 Certified Institution
Thalavapalayam, Karur – 639 113.

A Project Report

on

NEWS TRACKER APPLICATION

Submitted in partial fulfillment for the award of the degree

of

BACHELOR OF ENGINEERING

in

COMPUTER SCIENCE AND ENGINEERING

Under the Guidance of

Mr. V. Rajeshram
Assistant Professor/CSE

Submitted by

TEAM ID: PNT2022TMID15525

927619BCS4090 - PAVITHRA R

927619BCS4094 - PRAVEENA S

927619BCS4100 - RESHEMA R

927619BCS4118 - SWETHA S

NAALAIYA THIRAN – EXPERIENTIAL PROJECT BASED LEARNING INITIATIVE

18CSE040L - PROFESSIONAL READINESS FOR INNOVATION, EMPLOYABILITY AND ENTERPRENURSHIP

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

M.KUMARASAMY COLLEGE OF ENGINEERING

(Autonomous)

Karur - 639 113

November, 2022

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

CHAPTER 1

INTRODUCTION

1.1 PROJECT OVERVIEW

Newspaper contains limited, non user/reader specific, location specific news. There are multiple news-sharing apps available which can be used by a single user and are often spammed with notifications. There is also a lot of unwanted news which gets shared. So it may take a lot of time for the user to find the news he/she likes. A news-sharing app wants to help users find relevant and important news easily every day and also provide explicitly news from that users locality/region which may of help to the user. In today's world we have a lot of work and there is no time to cope up everything in the schedule. So, it is not possible to read the newspaper. By using this application, the user can access the news they are interested and get a quick peek.

1.2 PURPOSE

There are multiple news-sharing apps used by a single user and are often spammed with notifications. There is also a lot of fake news which gets shared. A news-sharing app wants to help users find relevant and important news easily every day and also understand explicitly that the news is not fake but from proper sources. This news tracker application helps the user to get all information about commodities, sports, technology, education etc. The user can register using their personal details and interacts with the application and save their time. Creating an app that benefits with the smart news filtering functionality where it allows its users to segregate various news sections, comprising of business, sports, political, international, and more. The users can opt for filters like breaking news, most popular, and more. Creating a user-friendly news app that would not just have good number of features, but will also be able to get accessed by different types of users keeping the experience

CHAPTER 2

LITERATURE SURVEY

2.1 EXISTING PROBLEM

S.No.	TITLE	AUTHOR	YEAR	DRAWBACKS
1	Following the fed with a news tracke	Michael william mccracken	2012	The paper is not a technical paper but is essentially a statistical paper on how should one conclude whether the data have come in stronger, weaker or as expected.
2	Topic detection and tracking in news articles	Sagar patel, sanket suthar, sandip patel, neha patel	2015	By effectively organizing large-scale documents, a method for the evolution of news topics over time is proposed in this paper to realize the tracking and evolution of topicsin the news text set.
3	Exploring mobile news reading interactions for news app personalisation	Marios constantinides,john dowell , david johson, sylvain malacria	2015	We first surveyed users news reading preferences and behaviours; analysis revealed three primary types of reader.
4	An end-to-end weakly- supervised news aggregation framework	xijin tang, xiaohui huang	2022	The framework combines snorkel based weakly supervised classification and topic signal detection model to classify and aggregate unlabeled news texts and ultimately generate visualized result.

Table 2.1 - Existing Problem

2.2 REFERENCES

1. Michael william Mccracken, "Following the Fed with a News Tracker", January 2012.
2. Sagar Patel, Sandip Patel, Nehal Patel, Sanket Suthar, "Topic Detection and Tracking in News Articles", March 2015.
3. Marios Constantinides, John Dowell, David Johnson, Sylvain Malacria, "Exploring mobile news reading interactions for news app personalisation", August 2015.
4. Xiaohui Huang, Xijin Tang, "An End-to-end Weakly- supervised News Aggregation Framework", June 2022.

2.3 PROBLEM STATEMENT DEFINITION

Newspaper contains limited, non user/reader specific, Location specific news. There are multiple news-sharing apps available which can be used by a single user and are often spammed with notifications. There is also a lot of unwanted news which gets shared. So it may take a lot of time for the user to find the news he/she likes. A news-sharing app wants to help users find relevant and important news easily every day and also provide explicitly news from that users locality/region which may of help to the user.In today's world we have a lot of work and there is no time to cope up everything in the schedule. So, it is not possible to read the newspaper. By using this application, the user can access the news they are interested and get a quick peek for the day. Digital news continues to evolve, encouraged by a various innovations in recent time, from groundbreaking new technologies like virtual reality and automated reporting to experiments on social platforms that have altered campaign coverage. Topic detection and tracking is challenging topic in information retrieval technology that can be used in the text mining. In topic detection we finding the most important topics in a collection of news articles. Regarding the current events, a system is required to detect and track topics within news articles. We would be choosing any one or two domain from politics, sports, science and discovery etc. Focusing on chose domain goal is to implement a system that gives quite satisfying results about current events with all related stories using the optimal approach.

CHAPTER 3

IDEATION AND PROPOSED SOLUTION

3.1 EMPATHY MAP CANVAS

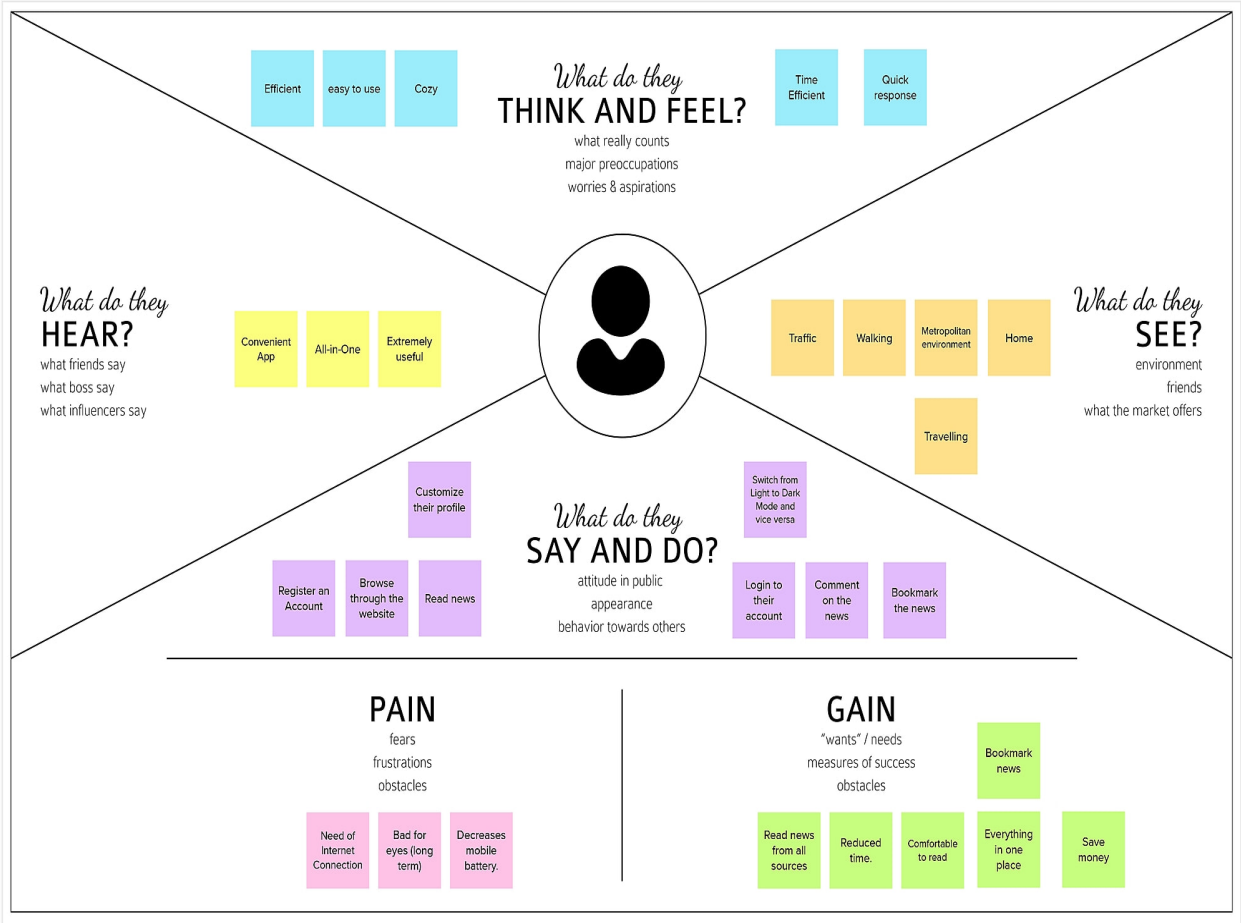

Figure 3.1 - Empathy Map

An empathy map is a widely-used visualization tool within the field. In relation to empathetic design, the primary purpose of an empathy map is to bridge the understanding of the end user.

3.2 IDEATION AND BRAINSTORMING

Step 1: Team Gathering, Collaboration and Select the Problem Statement

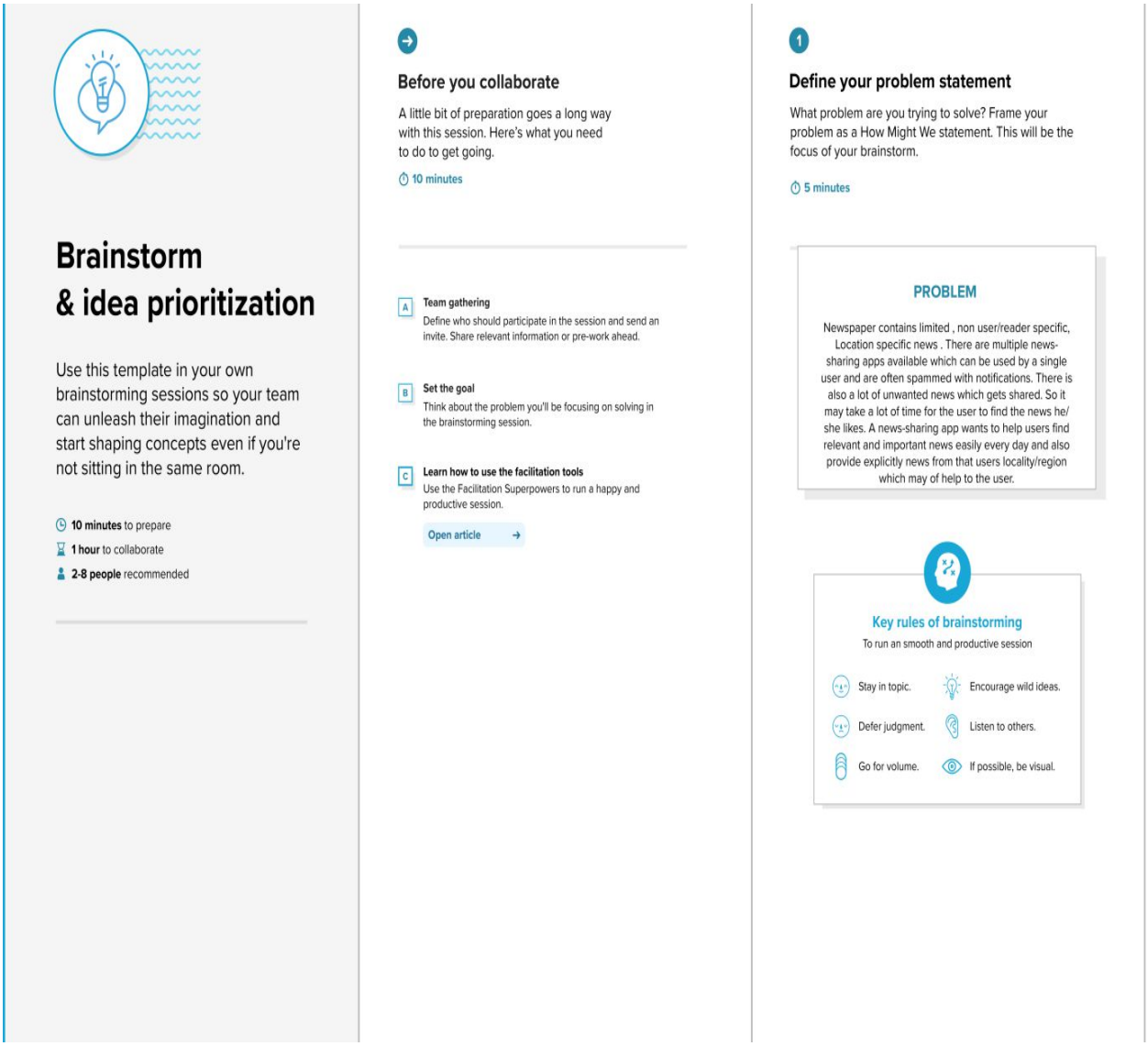

Figure 3.2 - Ideation and Brainstorming

A principal difference between ideation and brainstorming is that ideation is commonly more thought of as being an individual pursuit, while brainstorming is almost always a group activity.

Step 2: Brainstorm, Idea Listing and Grouping

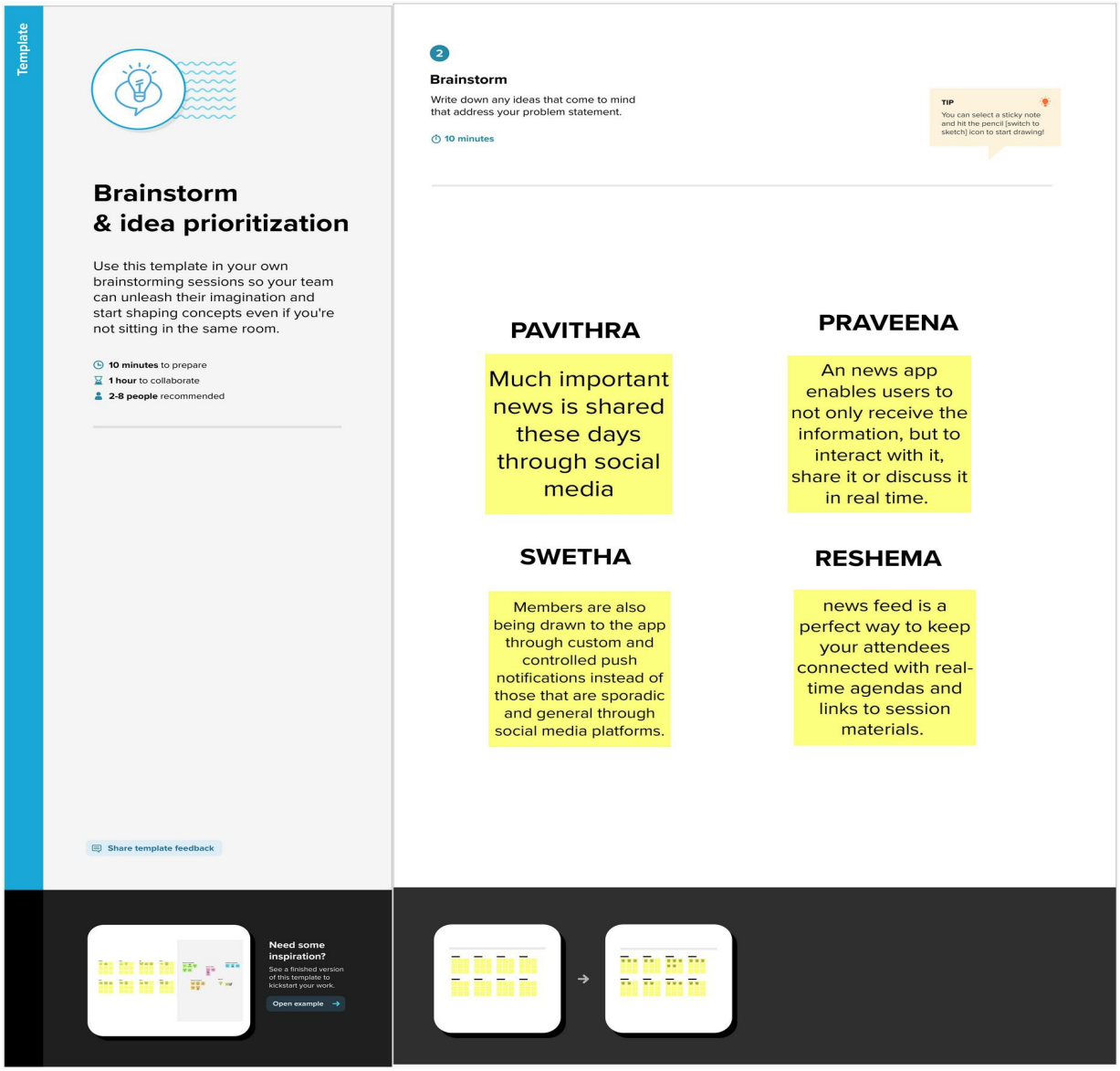

Figure 3.3 - Brainstorm, Idea Listing and Grouping

The idea listing and grouping is used to organize and analyse large numbers of ideas by categorising them. By organising and reorganising ideas, students gain a better appreciation of, and dialogue about, their ideas. As students create idea clusters, new contexts and connections among themes emerge.

Step 3: Idea Prioritization

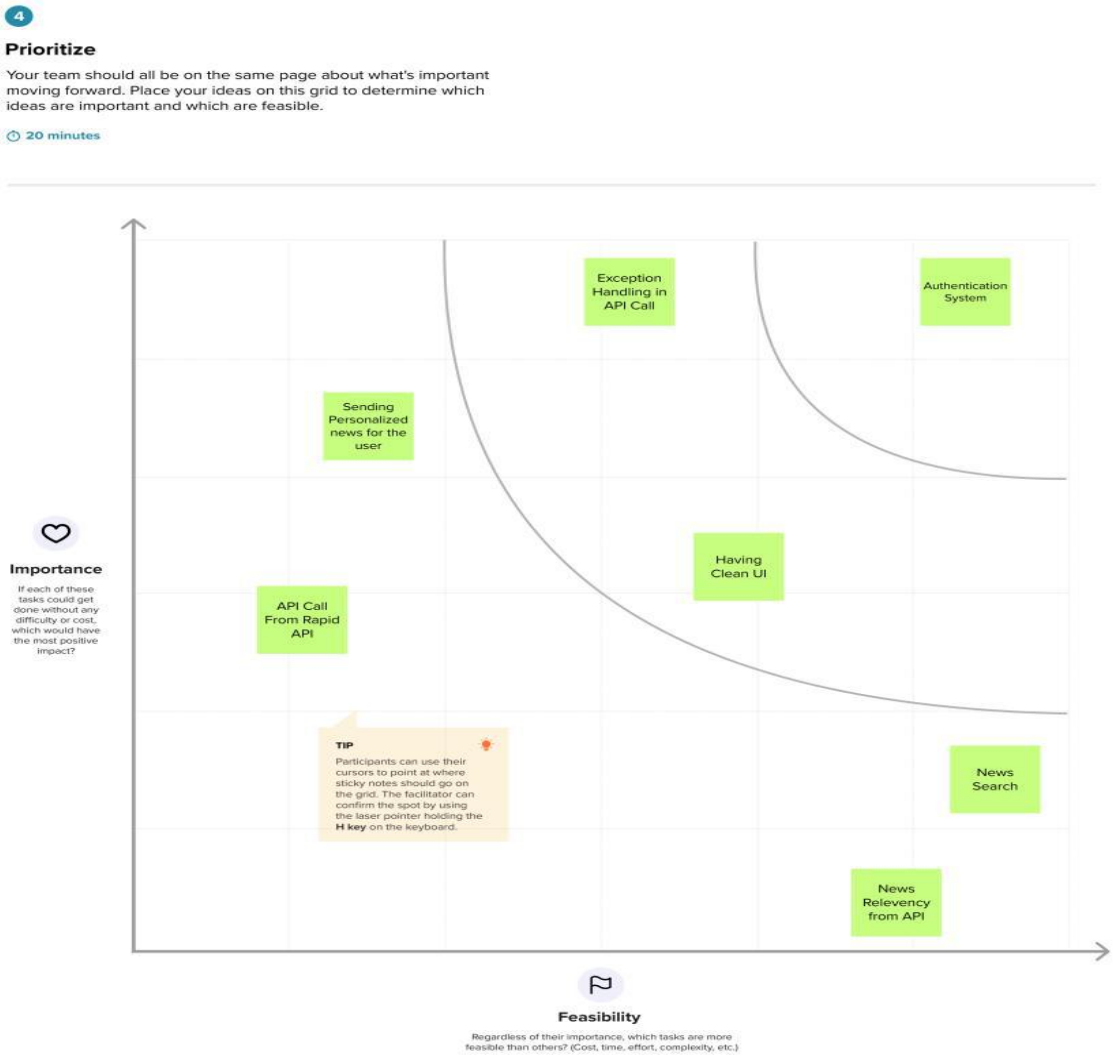

Figure 3.4 - Idea Prioritization

Idea prioritization is just a part of the idea management process. Having a structured idea management process and a systematic way of gathering, evaluating and prioritizing new ideas takes time. To make it work, the entire idea management process should be integrated to the everyday ways of working.

3.3 PROPOSED SOLUTION

S.No.	PARAMETER	DESCRIPTION
1.	Problem statement (problem to be solved)	Everyday, a lot of events happen world-wide and we rely on newspapers, television and news articles to get the reliable and trust-worthy information about these events. as a result, we created a platform that offers such news from reliable sources worldwide, in an organized and efficient manner.
2.	Idea / Solution Description	One platform for all local and worldwide news. trustworthy and reliable news, fast and efficient system. preventing spread of false information. data storage and backup, communication.
3.	Novelty / Uniqueness	A cloud computing-based news application that generates news and reports about the happenings around the world using computers and network. News based on most reliable and trustworthy resources around the world.developing the eco- friendly & sustainability based on centre.
4.	Social Impact / Customer satisfaction	Our platform eliminates the spread of false news and exposes the injustice and wrongdoings done by false groups. eliminating the fake news provides better understanding of the real-events happening in the world and the spread of knowledge.
5.	Business Moel (revenue model)	Our business model will be monetized and generate income by showing advertisements and operating on monthly and yearly plan.
6.	Scalability of the Solution	Cloud scalability will help to increase the user-base by increasing the resource allocation and meeting the changing demands without sacrificing the efficiency or quality of our customer service and internal operations.

Table 3.1 - Proposed Solution

3.4 PROBLEM SOLUTION FIT

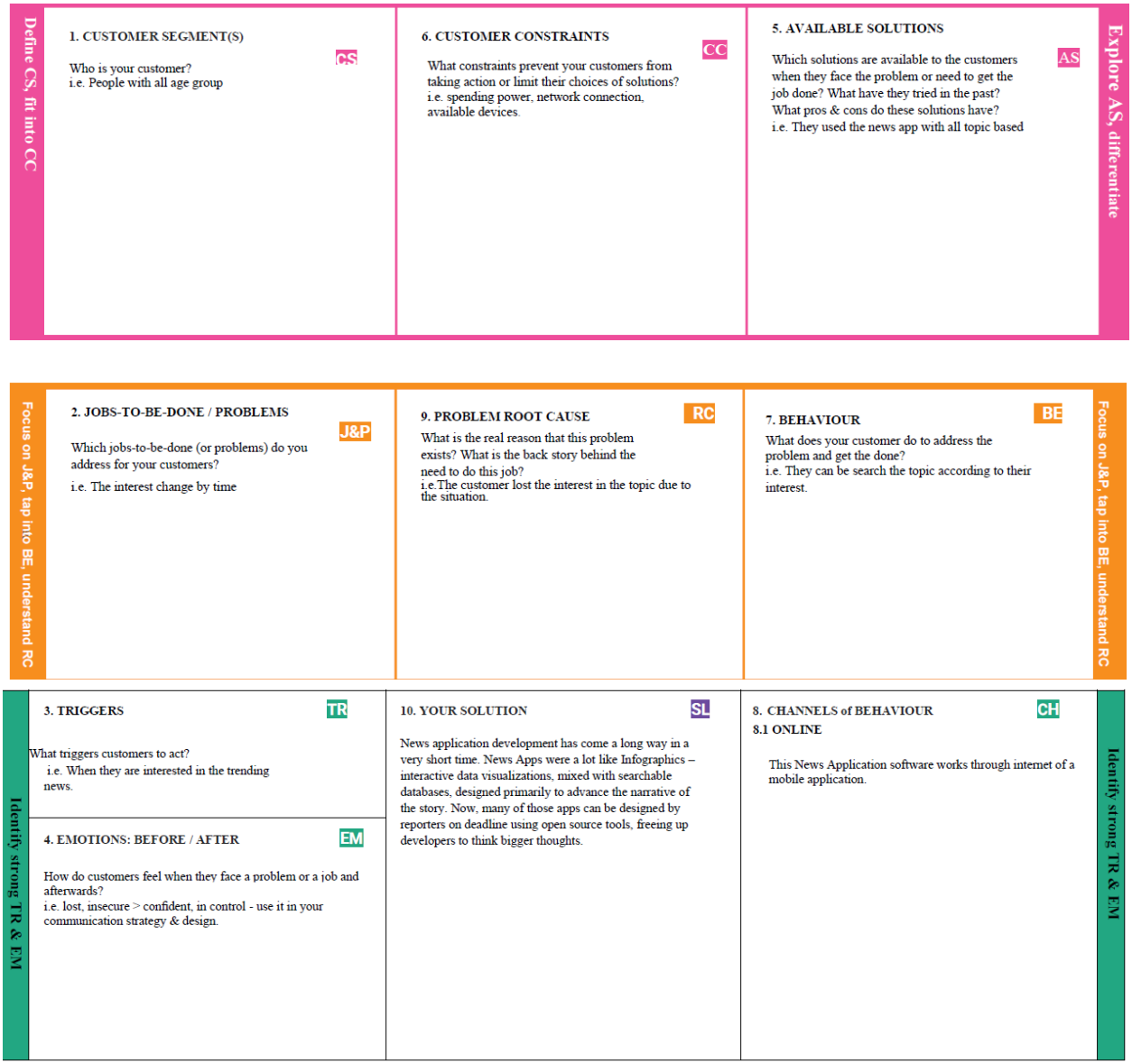

Figure 3.5 - Problem Solution Fit

The Problem Solution Fit simply means that you have found a problem with your customer and that the solution you have realized for it actually solves the customer's problem.

CHAPTER 4

REQUIREMENT ANALYSIS

4.1 FUNCTIONAL REQUIREMENTS

FR No.	FUNCTIONAL REQUIREMENTS (EPIC)	SUB REQUIREMENT(STORY/SUB-TASK)
FR-1	User Installation	User can install the app from google play store or from the website
FR-2	User Registration	Registration through form registration through gmail.
FR-3	User Confirmation	Confirmation via email confirmation via otp.
FR-4	User Login	User should login the app with the user's name and password

Table 4.1 - Functional Requirements

4.2 NON-FUNCTIONAL REQUIREMENTS

NFR No.	NON-FUNCTIONAL REQUIREMENTS	DESCRIPTION
NFR-1	**USABILITY**	Need to update their knowledge towards their interested news.
NFR-2	**SECURITY**	The information collected from user would be very safe and secure.
NFR-3	**RELIABILITY**	It will operate in a defined environment without failure.
NFR-4	**PERFORMANCE**	it is easy to use and the application is easy to access.
NFR-5	**AVAILABILITY**	The information whatever the user needed everything is available in this application.
NFR-6	**SCALABILITY**	It is a open free source and support to all platform where millions of people can use this application.

Table 4.2 - Non-Functional Requirements

CHAPTER 5

PROJECT DESIGN

5.1 DATA FLOW DIAGRAMS

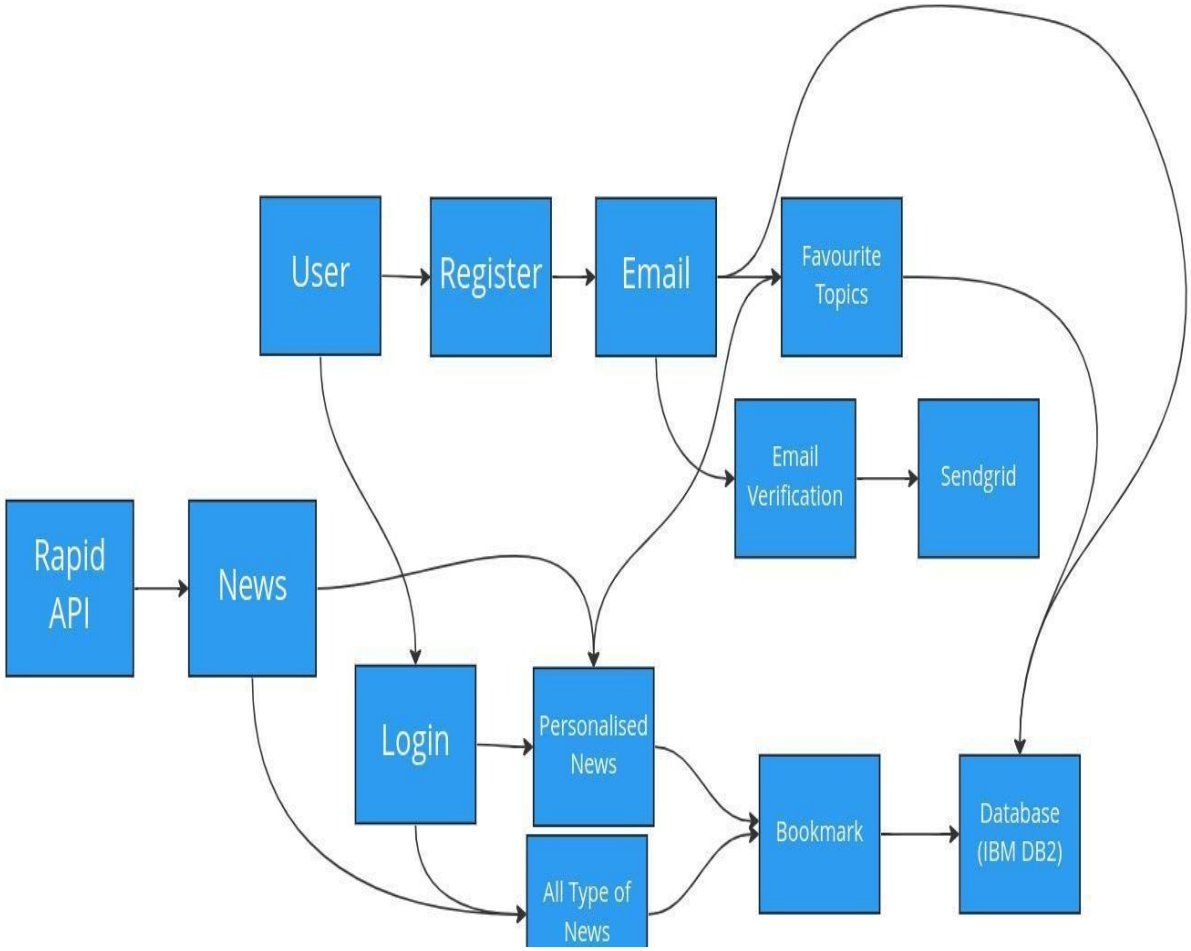

Figure 5.1 - Describes the flow of the project deployment

A data flow diagram shows the way information flows through a process or system. It includes data inputs and outputs, data stores, and the various subprocesses the data moves through.

5.2 SOLUTION AND TECHNICAL ARCHITECTURE

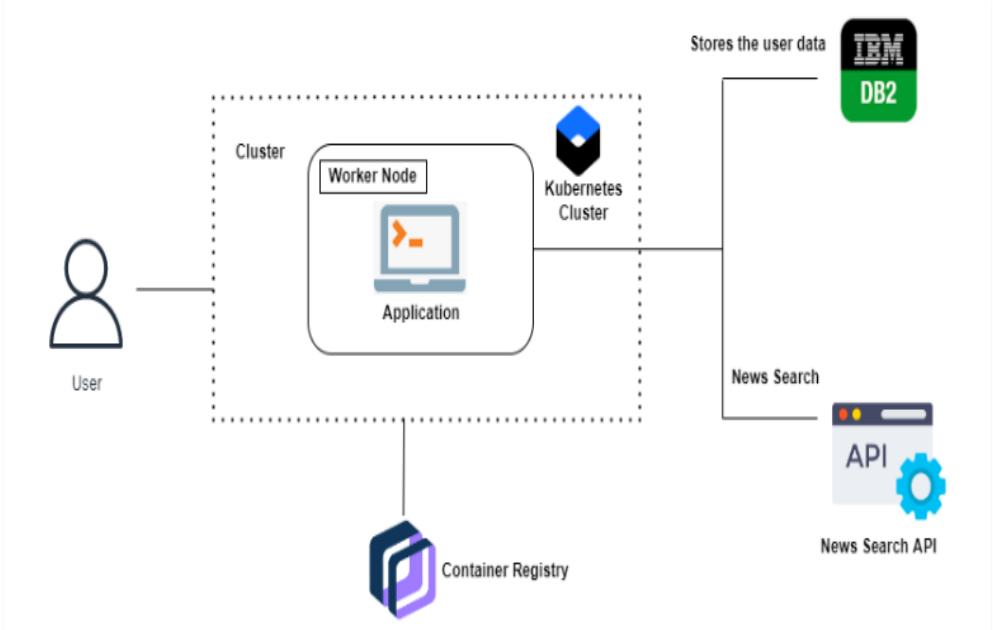

Figure 5.2 - Solution and technical architecture

- The user interacts with the application.
- Registers by giving the details.
- Integrate the application with news APIs and store the data in the database.
- The database will have all the details and the user can search the news by using a search bar.

5.3 USER STORY

User type	Functional requirement (epic)	User story no	User story / task	Acceptace criteria	Priority	Relese
Step 1	Registration	USN-1	As a user, i can register for the application by entering my email, password, and confirming my password.	I can access my account / dashboard	High	Sprint-1
Step 2	Email verification	USN-2	As a user, i will receive confirmation email once i have registered for the application	I can receive confirmation email & click confirm	High	Sprint-1
Step 3	Favourite topics	USN-3	As a user, i can choose my favourite topics	I can see all my preferred news under for you category	Medium	Sprint-2
Step 4	Login	USN-4	As a user, i can login with the email and password into the app	I can access to all the news	High	Sprint-1
Step 5	Dashboard	USN-5	As a user, i can see all the news under specific tab	I can view all the news	Medium	Sprint-3
Step 6	Bookmark	USN-6	As a user, i can bookmark my favourite news topicsas a user.	We can later view my bookmarked news	Low	Sprint-2
Step 7	Access	USN-7	As a user, i can access the site anywhere and everywhere	I can view the site by typing the url of the site	High	Sprint-4

Table 5.1 - User story

CHAPTER 6

PROJECT PLANNING & SCHEDULING

6.1 SPRINT PLANNING & ESTIMATION

Sprint	Functional requirement (epic)	User story no	User story / task	Story points	Priority	Team Members
Sprint-1	Registration	USN-1	As a user, i can register for the application by entering my email, password, and confirming my password.	10	High	Praveena S Swetha S
Sprint-1	Email verification	USN-2	As a user, i will receive confirmation email once i have registered for the application	10	High	Pavithra R Reshema R Praveena S Swetha S
Sprint-2	Favourite topics	USN-3	As a user, i can choose my favourite topics	7	Medium	Pavithra R Reshema R
Sprint-2	Login	USN-4	As a user, i can login with the email and password into the app	10	High	Swetha S
Sprint-3	Dashboard	USN-5	As a user, i can see all the news under specific tab	7	Medium	Praveena S
Sprint-3	Bookmark	USN-6	As a user, i can bookmark my favourite news topicsas a user.	7	Medium	Praveena S Swetha S
Sprint-4	Access	USN-7	As a user, i can access the site anywhere and everywhere	10	High	Pavithra R Praveena S Reshema R Swetha S

Table 6.1 - Sprint planning

Sprint	Total Story Points	Duration	Sprint Start Date	Sprint End Date (Planned)	Story Points Completed (as on Planned End Date)	Sprint Release Date (Actual)
Sprint-1	10	6 Days	24 Oct 2022	29 Oct 2022	8	29 Oct 2022
Sprint-2	10	6 Days	31 Oct 2022	05 Nov 2022	7	05 Nov 2022
Sprint-3	10	6 Days	07 Nov 2022	12 Nov 2022	8	12 Nov 2022
Sprint-4	10	6 Days	14 Nov 2022	19 Nov 2022	7	19 Nov 2022

Table 6.2 - Sprint Estimation

VELOCITY:

Imagine we have a 10-day sprint duration, and the velocity of the team is 20 (points per sprint). Let's calculate the team's average velocity (AV) per iteration unit (story points per day.

$$AV = \frac{\text{Sprint duration}}{\text{Velocity}} = 6/10 = 0.6$$

6.2 PROJECT DELIVERY SCHEDULE

TITLE	DESCRIPTION	DATE
Literature Survey & Information Gathering	Literature survey on the selected project & gathering information by referring the technical papers, research publications, journals etc.	1 SEPTEMBER 2022
Prepare Empathy Map	Prepare Empathy Map Canvas to capture the user Pains and Gains, prepare list of problem Statements that are to be solved by this project.	7 SEPTEMBER 2022 & 9 SEPTEMBER 2022
Ideation	List the ideas by organizing a brainstorming session and prioritize the top three ideas based on the feasibility and importance.	15 SEPTEMBER 2022
ProposedSolution	Prepare the proposed solution document, which includes novelty, feasibility of idea, revenue model, social impact, scalability of solution, etc.	22 SEPTEMBER 2022
Problem Solution Fit	Prepare problem - solution fit document.	30 SEPTEMBER 2022
SolutionArchitecture	Prepare solution architecture document.	30 SEPTEMBER 2022
CustomerJourney	Prepare the customer journey maps to understand the user interactions and experiences with the application (entry to exit).	6 OCTOBER2022
Functional Requirement	Prepare the functional requirement document.	11 OCTOBER 2022
Data FlowDiagrams	Draw the data flow diagrams and submit for review.	11 OCTOBER 2022
TechnologyArchitecture	Prepare the technology architecture diagram.	14 OCTOBER 2022
Prepare Milestone &Activity List	Prepare the milestones and activity list of the project.	21 OCTOBER 2022
Project Development - Delivery of Sprint-1, 2, 3 &4	Develop and submit the developed code by testing it.	IN PROGRESS...

Table 6.3 - Project delivery

6.3 REPORTS FROM JIRA

BURNDOWN CHART

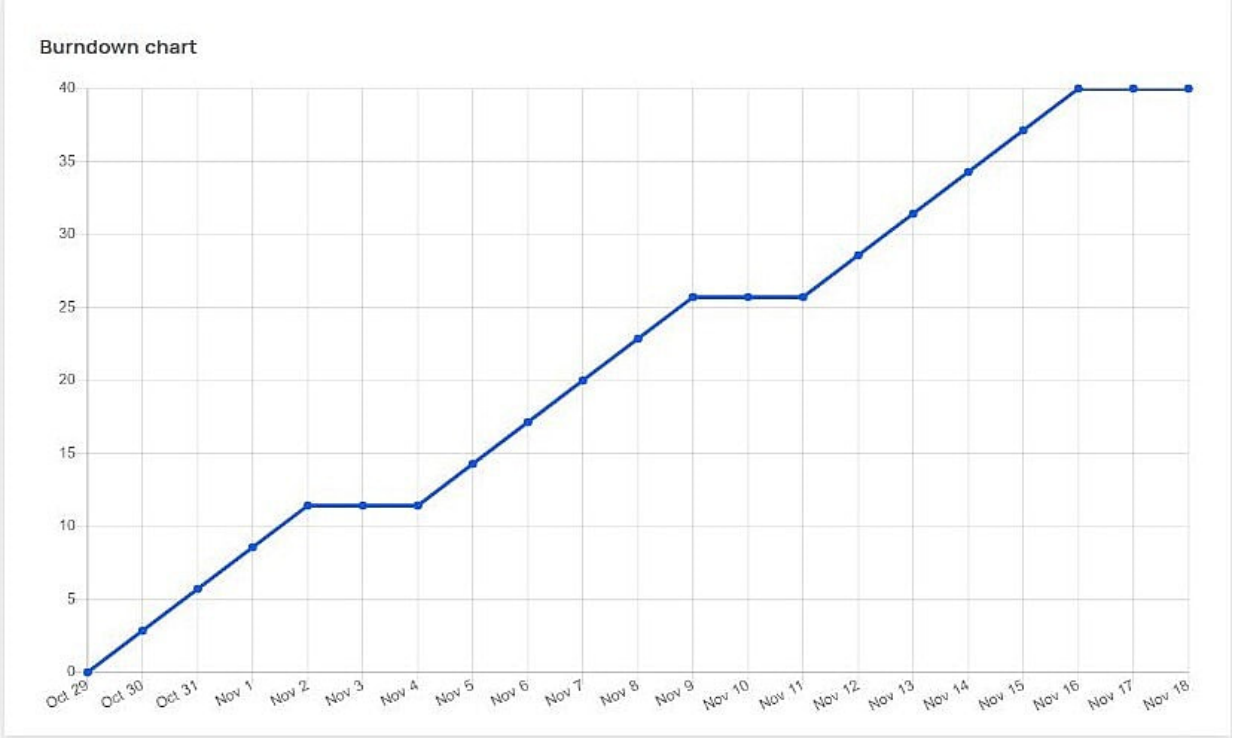

Figure 6.1 - Burndown chart

A burndown chart is a tool used by Agile teams to gather information about work completed on a project and work to be done in a given time period. Often, teams can use their burndown chart as a prediction tool that allows them to visualize when their project will be completed.

BURNUP CHART

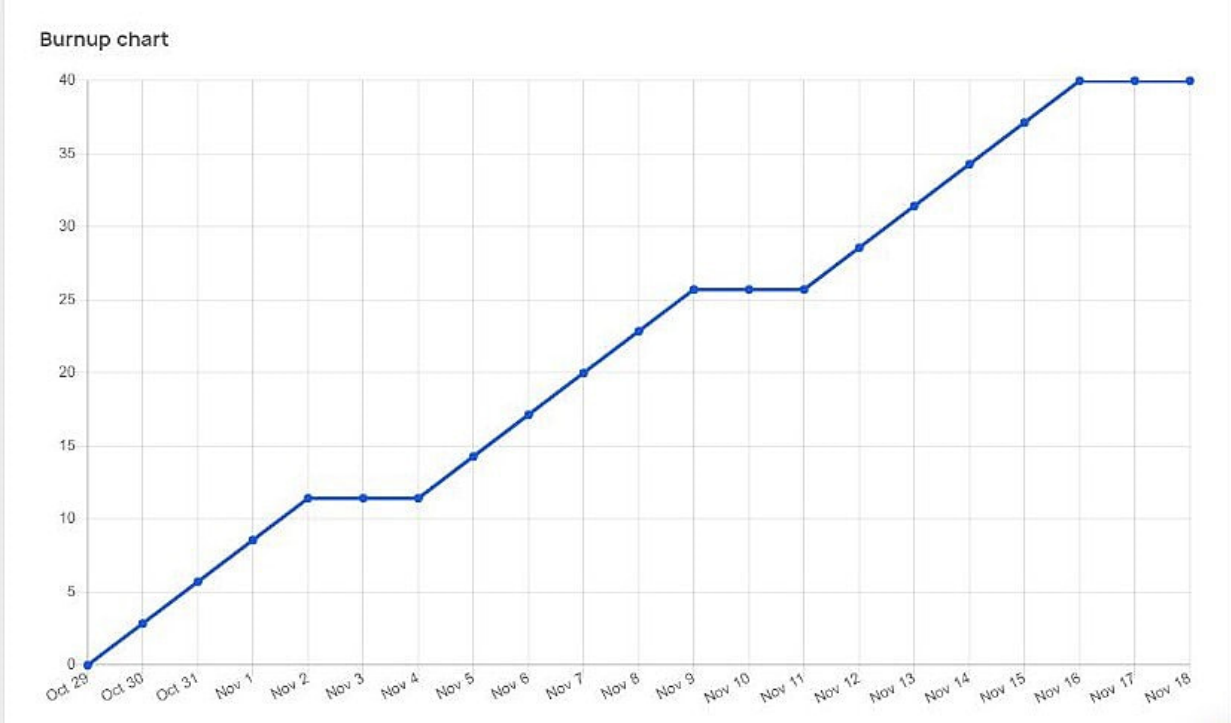

Figure 6.2 - Burnup Chart

A burn up chart is a visual diagram commonly used on Agile projects to help measure progress. Agile burn up charts allow project managers and teams to quickly see how their workload is progressing and whether project completion is on schedule.

CHAPTER 7

CODING & SOLUTIONING

7.1 FEATURE 1

Front-end web development, also known as client-side development is the practice of producing HTML, CSS and JavaScript for a website or Web Application so that a user can see and interact with them directly.The layer above the back end is the front end and it includes all software or hardware that is part of a user interface. Human or digital users interact directly with various aspects of the front end of a program, including user-entered data, buttons, programs, websites and other featuresGlobal Support: Different type of newspaper will be available from all around the world in different languages with this user will be able to get news from all around the world.Short News: News will be displayed in short format with title, image and little description in list view. It will help user to access required news faster.

```
import json
 import requests
 def get_sources():
    json_sources = requests.get("https://newsapi.org/v1/sources").json()
    sources = []
    counter = 1
    for source in json_sources["sources"]:
        sources.append({
            "id": source['id'],
            "name": source['name'],
            "num": counter
        })
        counter += 1
    with open("sources.json", 'w') as out:
        json.dump(sources, out, indent=4)
    return sources
 get_sources(
```

7.2 FEATURE 2

The framework is the basis upon which software programs are built. It serves as a foundation for software developers, allowing them to create a variety of applications for certain platforms. It is a set of functions and predefined classes used to connect with the system software and handle inputs and outputs. It simplifies the life of a developer while giving them the ability to use certain extensions and makes the online applications scalable and maintainable. Flask is a web application framework written in Python. A Web Application Framework or a simply a Web Framework represents a collection of libraries and modules that enable web application developers to write applications without worrying about low-level details such as protocol, thread management, among other examples. Flask is a web application framework written in Python. It was developed by Armin Ronacher, who led a team of international Python enthusiasts called Poocco.

```
import json
import requests
def get_news(source):
  url =
"https://newsapi.org/v1/articles?apiKey=1d786ef6f8b54b8b88748b8b639d6a19&sourc
e={}".format(source)
  news_data = requests.get(url).json()
  news = []
  for article in news_data["articles"]:
    news.append({
      "author": article["author"],
      'title': article["title"],
      'desc': article["description"],
      'url': article["url"],
      'img': article['urlToImage']
    })

  return news
# def news_return(source):
```

```
#     # with open("sources.json") as sources:
#     #     sources = json.load(sources)
#     #     for source in sources:
#     #         print("{}. {}".format(source["num"], source["name"]))
#     #
#     #     source_num = int(input("Choose source: "))
#
#     data = get_news("reuters")
#
#     return data
 class news_article(object):
    def __init__(self, author, title, desc, url, img):
        self.author = author
        self.title = title
        self.desc = desc
        self.url = url
        self.img = img
    def print_article(self):
        print(self.author)
        print(self.title)
        print(self.desc)
 def main(source):
    news_data = get_news(source)
    articles = []
    for article in news_data:
        single_news_article = news_article(article["author"], article["title"], article["desc"],
article["url"],
                                           article["img"])
        articles.append(single_news_article)

    return articles
```

CHAPTER 8

TESTING

8.1 TEST CASES

TEST CASE ID	15525	**TEST CASE DESCRIPTION**	NEWS TRACKER APPLICATION

S.No.	PREREQUISITES	TEST DATA
1	Access to Application.	Enter into the application by clicking it.
2	Register by giving required details.	Details must be in valid format.
3	Login in to application check for favorite topics.	Can be customized to user's expectation.
4	Favourite topics can be bookmarked.	Click the bookmark icon to bookmark news.

Table 8.1 - TEST CASES

8.2 USER ACCEPTANCE TESTING

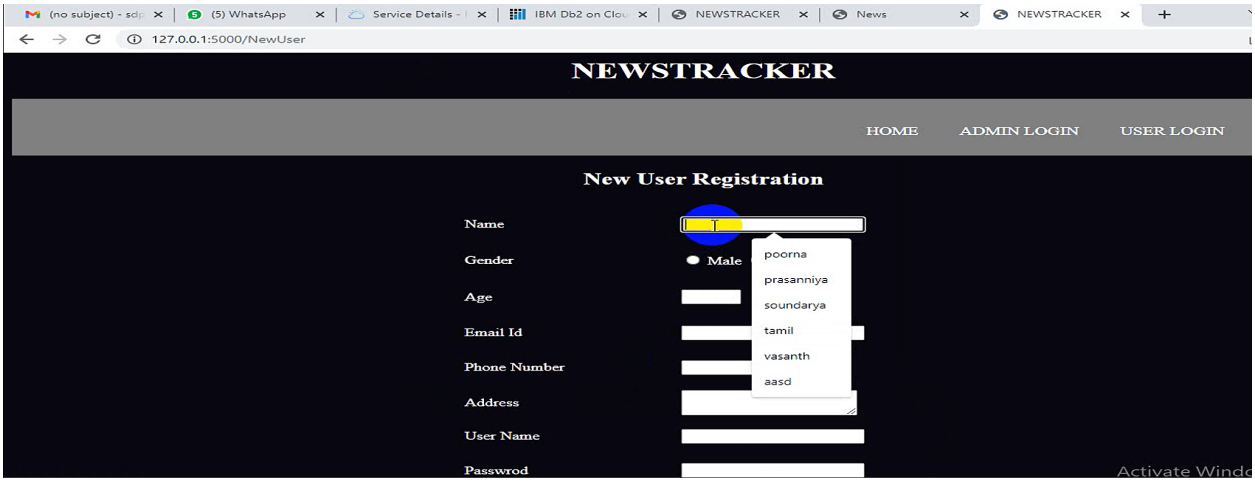

Figure 8.1 - Signup Page

The user needs to enter the valid details and they click join. After login we can enter through sign in page. The Details provided in the sign up page is stored.

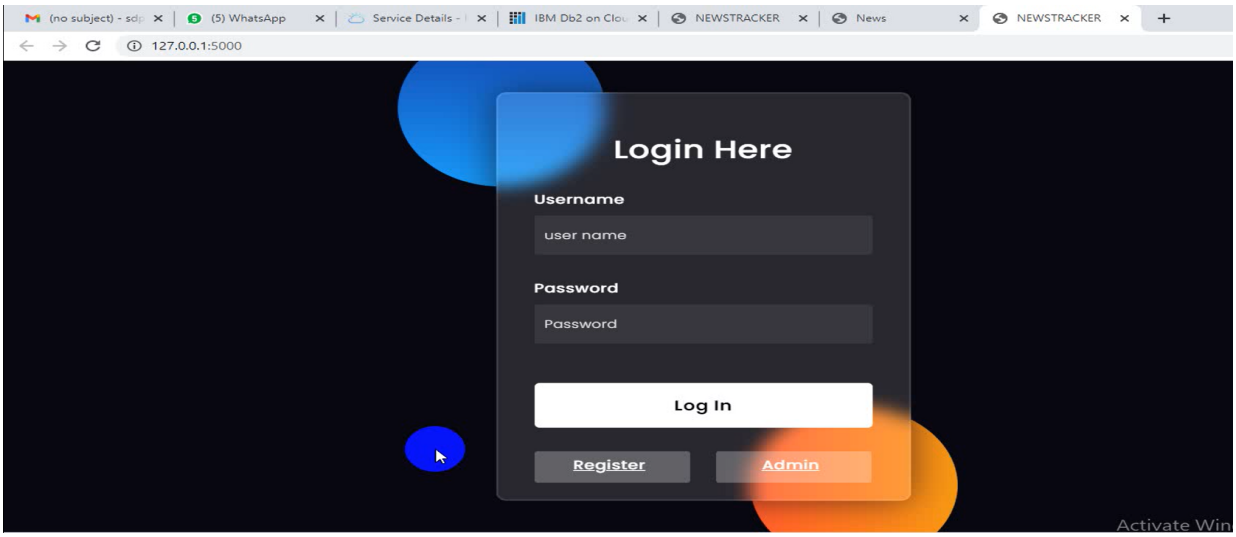

Figure 8.2 - Sign in page

The user needs to login in to the page by entering the credentials they entered during the sign up. The database validate the information provided and allows to go to next page.

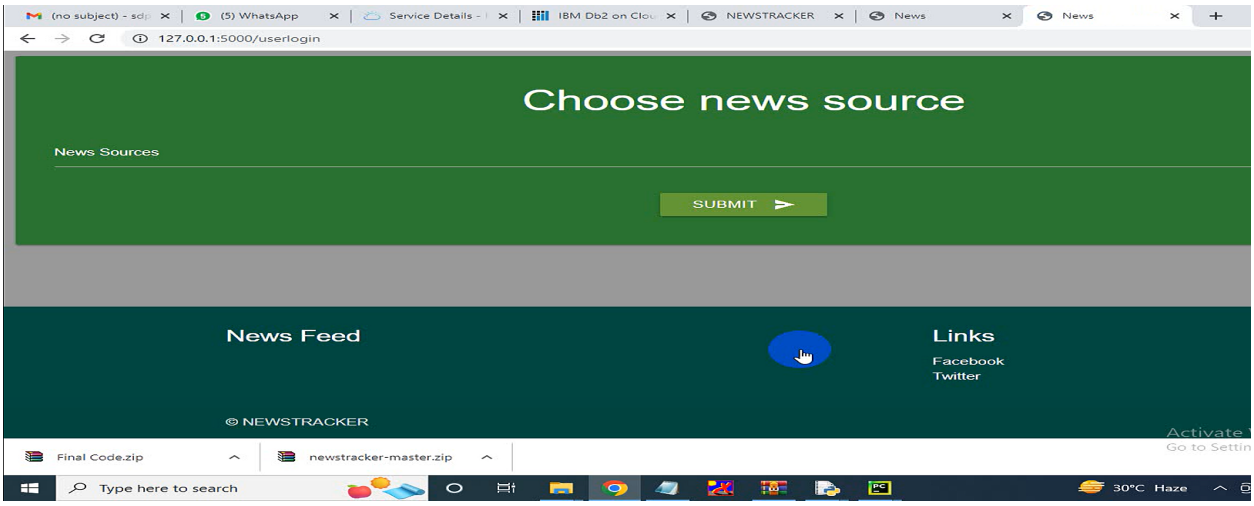

Figure 8.3 - Dashboard 1

After entering into their account they can search the news they want to view and they can customize and view the relevant news.

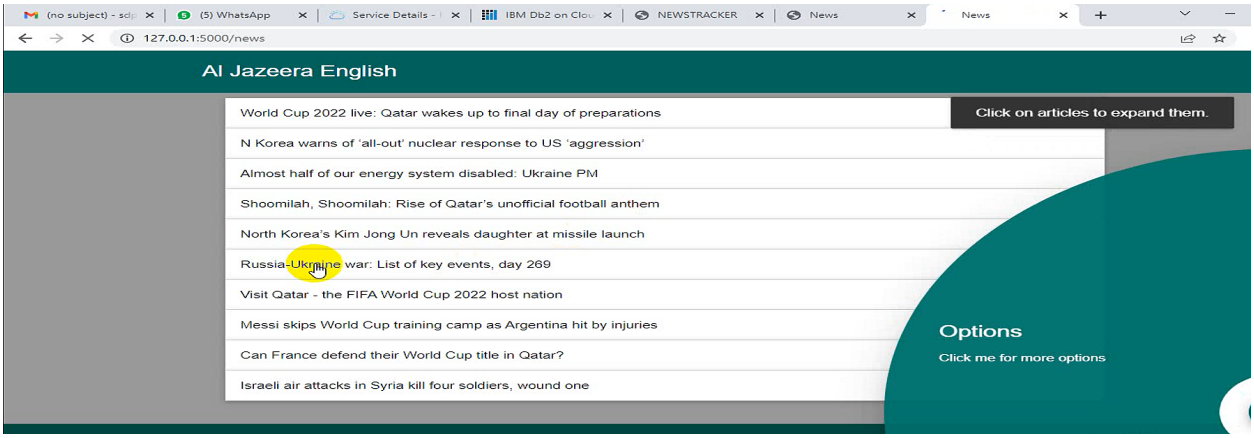

Figure 8.4 - Dashboard 2

The user can bookmark the favourite categories of news and they can see all the news related to their domain.

CHAPTER 9

RESULT

9.1 PERFORMANCE METRICS

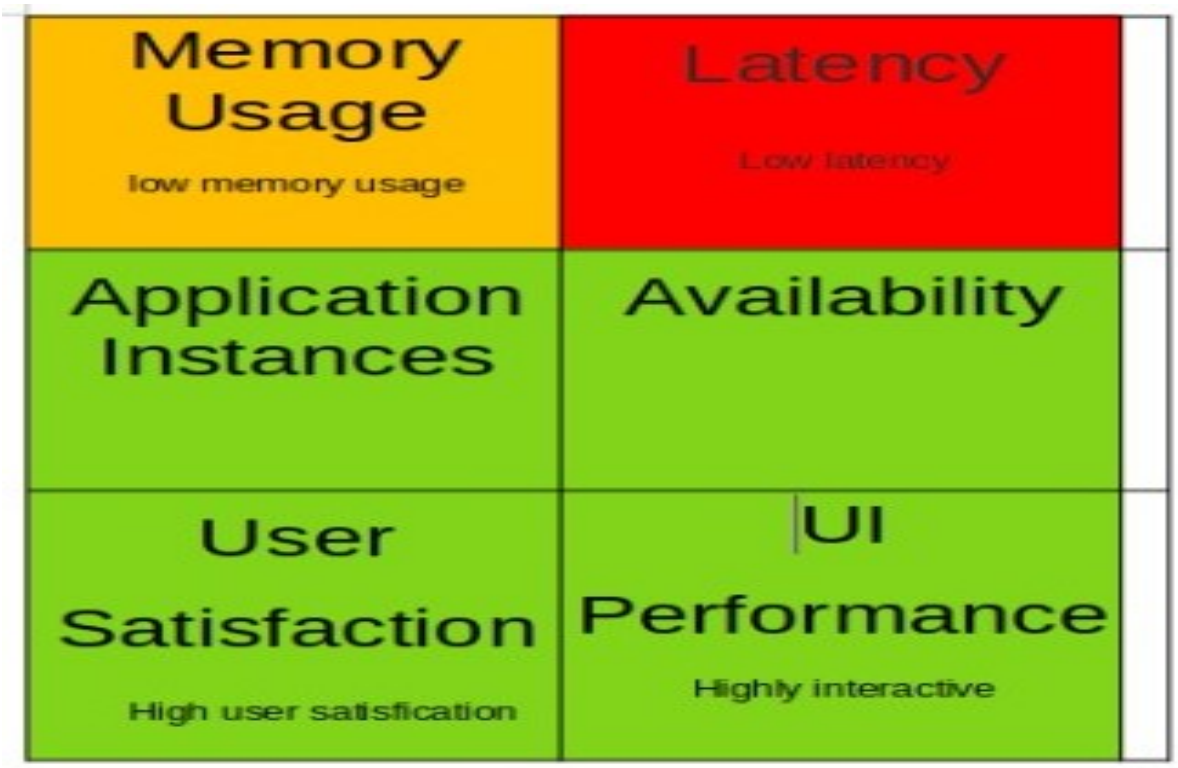

Fig 9.1 Performance Metrics

The performance of the app is optimized according to the user friendly interface. The memory is optimized using cloud and cloud database. It increase the performance and user statisfaction.

CHAPTER 10

ADVANTAGES AND DISADVANTAGES

10.1 ADVANTAGES

Whenever the reader pleases, he/she can use your News App as a one-stop point to get all the information they need. Whenever content is updated on the News App, the reader will be rolled towards your App with update notifications. An online newspaper allows the reader to interact with the paper itself. Readers can now leave comments, watch videos, view photo slideshows and oftentimes contribute their own opinions and written pieces to the paper.

- Better User Experience.
- Higher Engagement.
- Push notifications.
- Revenue Opportunities.
- App Store Presence.
- Engagement Tools.
- Social Media Integration.
- Registration and Personal Profiles.
- Personal Feed.
- Options to Interact With Content.

10.2 DISADVANTAGES

- Require data/wifi to get online.
- Companies not making as much money due to free reading for audiences.
- News spreads quicker online - people find out news before they should.
- Lose money - can't get people to pay for digital.
- Older audiences may not access digital platforms.You'll need Android and iOS applications and listings.
- Update and maintenance efforts are multiplied.
- There is extra Marketing pressure too.
- You need a separate URL.

CHAPTER 11

CONCLUSION

As world's technology is rapidly growing we has fast connection and network to instantly connect to other person. Day to day use in mobile, tablets and laptop is increasing, most of the people already have this facilities. In this fast and information oriented world we need to stay updated with every incidents and news too. This News app is android mobile application where user have access to latest news from 120+ newspapers from 50+ countries. The main focus of this application is to connect news articles from all around the world and deliver it to user as fast as possible in best visualize way.

CHAPTER 12

FUTURE SCOPE

Location feature with automation can be implemented which means as user move from one city to other local news will change as per it. Offline Reading can be improve will more efficient way on full articles. Data quality check needed. If API can't reach to certain article source it gives null value which can cause problem in JSON parsing. For a news provider, the smartphone screen is the the most challenging environment ever seen. There, chances are that a legacy media or a pure-player will find itself in direct competition, not only with the usual players in its field, but also with Facebook, Snapchat, Instagram and scores of gaming applications. Distraction is just one icon away; any weakness in functional or graphic design

CHAPTER 13

APPENDIX

```
 import news
from flask import Flask, render_template, request, jsonify, session
import datetime
import re
import ibm_db
import pandas
import ibm_db_dbi
from sqlalchemy import create_engine
engine = create_engine('sqlite://',echo = False)
dsn_hostname = "19af6446-6171-4641-8aba-
9dcff8e1b6ff.c1ogj3sd0tgtu0lqde00.databases.appdomain.cloud"
sn_uid = "wdn20062"
dsn_pwd = "2eFT80IyOuJQE3zV"
 dsn_driver = "{IBM DB2 ODBC DRIVER}"
dsn_database = "bludb"
dsn_port = "30699"
dsn_protocol = "TCPIP"
dsn_security = "SSL"
dsn = (
    "DRIVER={0};"
    "DATABASE={1};"
    "HOSTNAME={2};"
    "PORT={3};"
```

```
    "PROTOCOL={4};"
    "UID={5};"
    "PWD={6};"
    "SECURITY={7};").format(dsn_driver, dsn_database, dsn_hostname, dsn_port,
dsn_protocol, dsn_uid, dsn_pwd,dsn_security)
 try:    conn = ibm_db.connect(dsn, "", "")
print ("Connected to database: ", dsn_database, "as user: ", dsn_uid, "on host: ",
dsn_hostname)
except:
    print ("Unable to connect: ", ibm_db.conn_errormsg() )
app = Flask(__name__)
app.config.from_object(__name__)
app.config['SECRET_KEY'] = '7d441f27d441f27567d441f2b6176a'
@app.route("/")
def homepage():
  return render_template('UserLogin.html')
 @app.route("/alogin")
def alogin():
    return render_template('AdminLogin.html')
@app.route("/NewUser")
def NewUser():
return render_template('NewUser.html')
@app.route("/RNewUser", methods=['GET', 'POST'])
def RNewUser():
if request.method == 'POST':
        name1 = request.form['name']
        gender1 = request.form['gender']
```

```
        Age = request.form['age']

        email = request.form['email']

        address = request.form['address']

        pnumber = request.form['phone']

        uname = request.form['uname']

        password = request.form['psw']

        conn = ibm_db.connect(dsn, "", "")

        insertQuery = "INSERT INTO regtb VALUES ('" + name1 + "','" + gender1 + "','" + Age +
"','" + email + "','" + pnumber + "','" + address + "','" + uname + "','" + password + "')"

        insert_table = ibm_db.exec_immediate (conn, insertQuery)

        print(insert_table)

        return render_template('userlogin.html')

@app.route("/AdminHome")

def AdminHome():

    conn = ibm_db.connect(dsn, "", "")

    pd_conn = ibm_db_dbi.Connection(conn)

    selectQuery = "SELECT * from regtb "

    dataframe = pandas.read_sql(selectQuery, pd_conn)

    dataframe.to_sql('Employee_Data',con=engine,  if_exists='append')

    # run a sql query

    data = engine.execute("SELECT * FROM Employee_Data").fetchall()

    return render_template('AdminHome.html', data=data)

@app.route("/userlogin", methods=['GET', 'POST'])

def userlogin():

    error = None

    if request.method == 'POST':
```

```
        username = request.form['uname']
        password = request.form['password']
        session['uname'] = request.form['uname']
       conn = ibm_db.connect(dsn, "", "")
        pd_conn = ibm_db_dbi.Connection(conn)
        selectQuery = "SELECT * from regtb where uname='" + username + "' and
password='" + password + "'"
        dataframe = pandas.read_sql(selectQuery, pd_conn)
        if dataframe.empty:
        data1 = 'Username or Password is wrong'
         return render_template('goback.html', data=data1)
        else:
            print("Login")
            selectQuery = "SELECT * from regtb where uname='" + username + "' and
password='" + password + "'"
            dataframe = pandas.read_sql(selectQuery, pd_conn)
              dataframe.to_sql('Employee_Data'con=engine,if_exists='append')
            # run a sql query
            print(engine.execute("SELECT * FROM Employee_Data").fetchall())
            return render_template('index.html', data=engine.execute("SELECT * FROM
Employee_Data").fetchall())
@app.route("/adminlogin", methods=['GET', 'POST'])
def adminlogin():
    error = None
    if request.method == 'POST':
        username = request.form['uname']
        password = request.form['password']
```

```
    conn = ibm_db.connect(dsn, "", "")
     pd_conn = ibm_db_dbi.Connection(conn)
     selectQuery = "SELECT * from admintb where USERNAME='" + username + "' and
PASSWORD='" + password + "'"
     dataframe = pandas.read_sql(selectQuery, pd_conn)
     if dataframe.empty:
        data1 = 'Username or Password is wrong'
        return render_template('goback.html', data=data1)
     else:
        print("Login")
        selectQuery = "SELECT * from regtb "
        dataframe = pandas.read_sql(selectQuery, pd_conn)
        dataframe.to_sql('Employee_Data', con=engine,if_exists='append')
        # run a sql query
        print(engine.execute("SELECT * FROM Employee_Data").fetchall())
     return render_template('AdminHome.html', data=engine.execute("SELECT * FROM
Employee_Data").fetchall())
@app.route('/news', methods=['POST'])
def submit_data():
   vals = request.form['news-source']
    id = vals.split('_')[0]
   name = vals.split('_')[1]
   articles = news.main(id)
   print(articles)
   return render_template('news.html', articles=articles, name=name)
if __name__ == '__main__':
   app.run(host="0.0.0.0")
```

```
html
{% extends 'layout.html' %} {% block body %}
<div class="row">
  <div class="center col s12">
   <div class="card green darken-3 hoverable">
    <div class="card-content white-text">
     <span class="center card-title"><h3>Choose news source</h3></span>
      <div class="row">
       <form action='/news' method="POST" class="col s12">
        <div class="row">
   <div class="input-field center">
    <select id="news-source" name="news-source" required="" class="initialized">
   <option value="" disabled selected>News Sources</option>
   <option value="abc-news-au_ABC News (AU)">ABC News (AU)</option>
  <option value="al-jazeera-english_Al Jazeera English">Al Jazeera English</option>
  <option value="ars-technica_Ars Technica">Ars Technica</option>
 <option value="associated-press_Associated Press">Associated Press</option>
<option value="bbc-news_BBC News">BBC News</option>
 <option value="bbc-sport_BBC Sport">BBC Sport</option>
<option value="bild_Bild">Bild</option>
<option value="bloomberg_Bloomberg">Bloomberg</option>
 <option value="breitbart-news_Breitbart News">Breitbart News</option>
  <option value="business-insider_Business Insider">Business Insider</option>
  <option value="business-insider-uk_Business Insider (UK)">Business Insider (UK)
   </option>
    <option value="buzzfeed_Buzzfeed">Buzzfeed</option>
```

<option value="cnbc_CNBC">CNBC</option>

<option value="cnn_CNN">CNN</option>

<option value="daily-mail_Daily Mail">Daily Mail</option>

<option value="der-tagesspiegel_Der Tagesspiegel">Der Tagesspiegel</option>

<option value="die-zeit_Die Zeit">Die Zeit</option>

<option value="engadget_Engadget">Engadget</option>

<option value="entertainment-weekly_Entertainment Weekly">Entertainment Weekly</option>

<option value="espn_ESPN">ESPN</option>

<option value="espn-cric-info_ESPN Cric Info">ESPN Cric Info</option>

<option value="financial-times_Financial Times">Financial Times</option>

<option value="focus_Focus">Focus</option>

<option value="football-italia_Football Italia">Football Italia</option>

<option value="fortune_Fortune">Fortune</option>

<option value="four-four-two_FourFourTwo">FourFourTwo</option>

<option value="fox-sports_Fox Sports">Fox Sports</option>

<option value="google-news_Google News">Google News</option>

<option value="gruenderszene_Gruenderszene">Gruenderszene</option>

<option value="hacker-news_Hacker News">Hacker News</option>

<option value="handelsblatt_Handelsblatt">Handelsblatt</option>

<option value="ign_IGN">IGN</option>

<option value="independent_Independent">Independent</option>

<option value="mashable_Mashable">Mashable</option>

<option value="metro_Metro">Metro</option>

<option value="mirror_Mirror">Mirror</option>

<option value="mtv-news_MTV News">MTV News</option>

<option value="mtv-news-uk_MTV News (UK)">MTV News (UK)</option>

<option value="national-geographic_National Geographic">National Geographic</option>

_New Scientist">New Scientist</option>

<option value="newsweek_Newsweek">Newsweek</option>

<option value="new-york-magazine_New York Magazine">New York Magazine</option>

<option value="nfl-news_NFL News">NFL News</option>

<option value="polygon_Polygon">Polygon</option>

<option value="recode_Recode">Recode</option>

<option value="reddit-r-all_Reddit /r/all">Reddit /r/all</option>

<option value="reuters_Reuters">Reuters</option>

<option value="techcrunch_TechCrunch">TechCrunch</option>

<option value="techradar_TechRadar">TechRadar</option>

<option value="the-economist_The Economist">The Economist</option>

<option value="the-guardian-au_The Guardian (AU)">The Guardian (AU)</option>

<option value="the-guardian-uk_The Guardian (UK)">The Guardian (UK)</option>

<option value="the-hindu_The Hindu">The Hindu</option>

<option value="the-huffington-post_The Huffington Post">The Huffington Post</option>

<option value="the-lad-bible_The Lad Bible">The Lad Bible</option>

<option value="the-new-york-times_The New York Times">The New York Times</option>

<option value="the-next-web_The Next Web">The Next Web</option>

<option value="the-sport-bible_The Sport Bible">The Sport Bible</optio

<option value="the-telegraph_The Telegraph">The Telegraph</option>

<option value="the-times-of-india_The Times of India">The Times of India</option>

<option value="the-verge_The Verge">The Verge</option>

<option value="the-wall-street-journal_The Wall Street Journa

```
</option>

  <option value="the-washington-post_The Washington Post">The Washington
Post</option>

   <option value="time_Time">Time</option>

   <option value="usa-today_USA Today">USA Today</option>

    <option value="wired-de_Wired.de">Wired.de</op

<option value="wirtschafts-woche_Wirtschafts Woche">Wirtschafts Woche</option>

     </select>

    </div>

   </div>

      <button class="btn light-green darken-2 pulsewaves-effect waves-light waves-green"
type="submit"

      name="action">Submit<i class="material-icons right">send</i></button>

  </form>

  </div>

 </div>

{% endblock %}

 {% block script %}

<script src="{{ url_for('static', filename='select.js')}}"></script>

{% endblock
```

Github and Project Video Demo Link

Github Link:

https://github.com/IBM-EPBL/IBM-Project-21446-1659780401.git

Project Video Demo Link:

https://drive.google.com/file/d/1jXb1nn9Pu_c82APlnGO_z4-rt5wf591w/view?usp=share_link55