

**NEWSTRACKER APPLICATION**

**PROJECT REPORT TEAM ID: PNT2022TMID23357**

**COLLEGE NAME: VELAMMAL ENGINEERING COLLEGE,SURAPET**

**NACHIAPPAN A  
NISHANTH SARATHY M  
RISHIKUMAR R  
MAHESHKUMAR SELVAKUMAR**

# **Project Report Format**

1. **INTRODUCTION**
  - 1.1 Project Overview
  - 1.2 Purpose
2. **LITERATURE SURVEY**
  - 2.1 Existing problem
  - 2.2 References
  - 2.3 Problem Statement Definition
3. **IDEATION & PROPOSED SOLUTION**
  - 3.1 Empathy Map Canvas
  - 3.2 Ideation & Brainstorming
  - 3.3 Proposed Solution
  - 3.4 Problem Solution fit
4. **REQUIREMENT ANALYSIS**
  - 4.1 Functional requirement
  - 4.2 Non-Functional requirements
5. **PROJECT DESIGN**
  - 5.1 Data Flow Diagrams
  - 5.2 Solution & Technical Architecture
  - 5.3 User Stories
6. **PROJECT PLANNING & SCHEDULING**
  - 6.1 Sprint Planning & Estimation
  - 6.2 Sprint Delivery Schedule
  - 6.3 Reports from JIRA
7. **CODING & SOLUTIONING (Explain the features added in the project along with code)**
  - 7.1 Feature 1
  - 7.2 Feature 2
  - 7.3 Database Schema (if Applicable)
8. **TESTING**
  - 8.1 Test Cases
  - 8.2 User Acceptance Testing
9. **RESULTS**
  - 9.1 Performance Metrics
10. **ADVANTAGES & DISADVANTAGES**
11. **CONCLUSION**
12. **FUTURE SCOPE**
13. **APPENDIX**
  - Source Code
  - GitHub & Project Demo Link

# INTRODUCTION

## **Project Overview:**

News is one of the primary source of gaining information about the actions and events that happen all around. It may be an event that happened in the past, happening now or going to happen in the future. In the present days where there is a rapid increase in the development and adaptability of technologies throughout all the demographic of people, it is necessary to provide news in such a way that it is interconnected with the current technological trends. As our lives are very busy these days, we often feel we need more than 24 hrs. a day to cope up with everything we have in our schedule. Well, that's not possible but reducing the time by changing the conventional method of reading news can help. Just tell us what market news you're interested in and get a quick peek for the day. Only read what you feel is relevant and save your time. This app helps you to query for all information about Indices, Commodities, Currencies, Future Rates, Bonds and so on as on official websites.

## **Purpose**

The purpose is to develop an application, which will eliminate the problems faced in the current scenario. This application will provide all the information and news related to cyber security, Esport, Science, and Technology that are in trend in one place. So, it will save time and efforts for the users by making it more efficient. Using, this application will terminate the possibility of information redundancy

# LITERATURE SURVEY

## Existing problem

The issue occurs when readers try to access news feeds when travelling. Accessing information is restricted through the usage of physical newspapers in the digital era.

## References

### 1. Android News App

**Author:** Brijesh Joshi , Nehal Patel

**Journal:** International Journal of Applied Engineering Research ISSN 0973-4562 Volume 13, Number 11 Published

**Year:** 2018 2. Other security Precautions:

### 2. Exploring mobile news reading interactions for news app personalization

**Author:** Marios Constantinides , Dowel

**Journal:** International Journal of Applied Engineering Research ISSN 0973-4562 Volume 13, Number 11 Published

**Year:** August 2015

### 3. Topic Detection and Tracking in News Article

**Author:** Sagar Patel, Sandip Patel,Sanket Suthar

**Journal:** DOI:10.1007/978-3-319-63645-0\_48

**Published Year:** March 2017

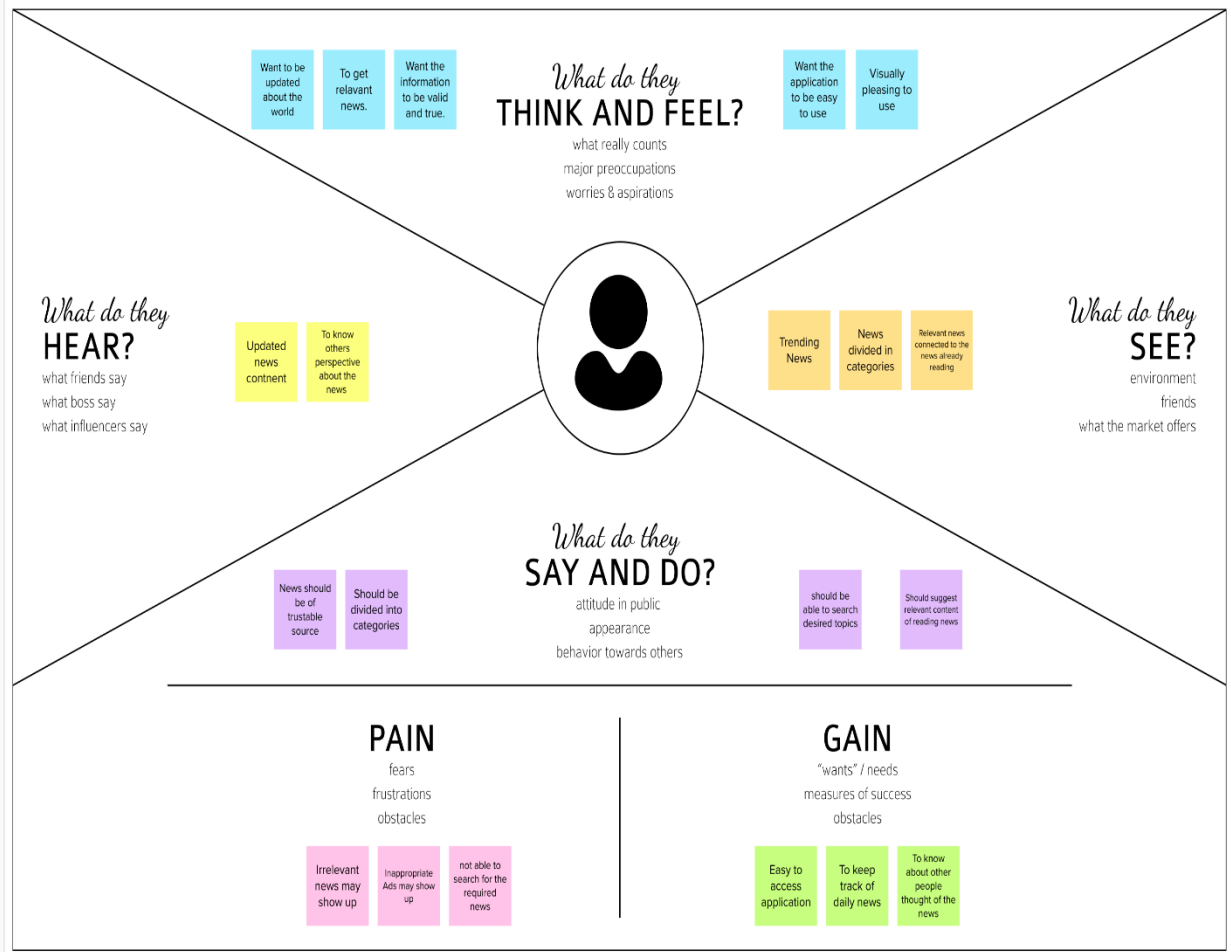
## Problem Statement Definition

As our lives are very busy these days, we often feel we need more than 24 hrs. a day to cope up with everything we have in our schedule. Well, that's not possible but reducing the time by changing the conventional method of reading news can help. Just tell us what market news you're interested in and get a quick peek for the day. Only read what you feel is relevant and save your time. This app helps you to query for all information about Indices, Commodities, Currencies, Future Rates, Bonds, etc.... as on official websites.

# IDEATION & PROPOSED SOLUTION

## Empathy Map Canvas

### Empathy Map for News Tracker Application



## Ideation & Brainstorming

The image displays five distinct templates for brainstorming and idea prioritization, arranged in a grid-like fashion. Each template is designed to guide a user through a structured creative process.  
  
1. **Template 1 (Top Left):** Features a lightbulb icon. The main heading is "Brainstorm & idea prioritization". It includes a section "Before you collaborate" with a tip about preparation and a 10-minute timer. Below this is a section "Define your problem statement" with a tip about problem definition and a 5-minute timer. The "Brainstorm" section has a tip about writing down ideas and a 10-minute timer. The "Idea prioritization" section includes a tip about ranking ideas and a 10-minute timer. A "Show template feedback" button is at the bottom.  
  
2. **Template 2 (Top Middle):** Features a lightbulb icon. The main heading is "Brainstorm & idea prioritization". It includes a section "Before you collaborate" with a tip about preparation and a 10-minute timer. Below this is a section "Define your problem statement" with a tip about problem definition and a 5-minute timer. The "Brainstorm" section has a tip about writing down ideas and a 10-minute timer. The "Idea prioritization" section includes a tip about ranking ideas and a 10-minute timer. A "Show template feedback" button is at the bottom.  
  
3. **Template 3 (Top Right):** Features a lightbulb icon. The main heading is "Brainstorm & idea prioritization". It includes a section "Before you collaborate" with a tip about preparation and a 10-minute timer. Below this is a section "Define your problem statement" with a tip about problem definition and a 5-minute timer. The "Brainstorm" section has a tip about writing down ideas and a 10-minute timer. The "Idea prioritization" section includes a tip about ranking ideas and a 10-minute timer. A "Show template feedback" button is at the bottom.  
  
4. **Template 4 (Bottom Left):** Features a lightbulb icon. The main heading is "Brainstorm & idea prioritization". It includes a section "Before you collaborate" with a tip about preparation and a 10-minute timer. Below this is a section "Define your problem statement" with a tip about problem definition and a 5-minute timer. The "Brainstorm" section has a tip about writing down ideas and a 10-minute timer. The "Idea prioritization" section includes a tip about ranking ideas and a 10-minute timer. A "Show template feedback" button is at the bottom.  
  
5. **Template 5 (Bottom Right):** Features a lightbulb icon. The main heading is "Brainstorm & idea prioritization". It includes a section "Before you collaborate" with a tip about preparation and a 10-minute timer. Below this is a section "Define your problem statement" with a tip about problem definition and a 5-minute timer. The "Brainstorm" section has a tip about writing down ideas and a 10-minute timer. The "Idea prioritization" section includes a tip about ranking ideas and a 10-minute timer. A "Show template feedback" button is at the bottom.

### Group ideas

Take Lears sharing your ideas with a clustering similar or related notes as you go. Once all sticky notes have been grouped, give each cluster a centered, distinct title. The center is bigger than the sticky notes, try and note if you need central it up into smaller sub groups.

20 minutes

### Prioritize

Your team should all be on the same page about which is important, moving towards. Place your ideas on this grid and discuss which ideas are an outlier and which are feasible.

20 minutes

### After you collaborate

You can export this report as a bridge to port to share with team members or your company's website. It will be a PDF.

Quick add-ons

- Show the report**  
Show a new link to the report to share others looking at the report.
- Export the report**  
Export the report as a PDF or CSV file for others to use.

Keep moving forward

- Strategy Blueprint**  
Show the components of your strategy.
- Customer experience journey map**  
Map the customer journey to the end of the journey.
- Strengths, weaknesses, opportunities & threats**  
Identify strengths, weaknesses, opportunities & threats.

Show template feedback

## Proposed Solution

S.No.	Parameter	Description
1.	Problem Statement (Problem to be solved)	As our lives are very busy these days, we often feel we need more than 24 hrs. a day to cope with everything we have in our schedule. Well, that's not possible but reducing the time by changing the conventional method of reading news can help. Just tell us what market news you're interested in and get a quick peek for the day. Only read what you feel is relevant and save your time. To change the method of news reading from traditional newspaper to the digital format of daily news.
2.	Idea / Solution description	Instead of using daily newspaper which has limited information we can create a way to access the news on any computing device. This provides an opportunity to access the required information anywhere and anytime.
3.	Novelty / Uniqueness	<ul style="list-style-type: none"> <li>News with various topics is displayed.</li> <li>Trending news at that particular time is displayed.</li> <li>The news feed updates based on your search history</li> <li>Can bookmark the news article and can be accessed later.</li> </ul>
4.	Social Impact / Customer Satisfaction	<ul style="list-style-type: none"> <li>The news feeds are displayed based on the interest of the users which is determined from the previous searches.</li> <li>Users are asked to select interesting categories during the initial period.</li> <li>Theme can be changed by the user according to their interest.</li> </ul>
5.	Business Model (Revenue Model)	<ul style="list-style-type: none"> <li>The primary source of revenue for the application is the ads that are displayed within the interface.</li> <li>Based on the number of users who view particular news, the news channel that</li> </ul>

		published that news article will have to pay a small amount as a commission
6.	Scalability of the Solution	This application can handle data across machines and data that will not fit into memory. It supports clusters and can handle machine failures, rebuilding machines easily.

## Problem Solution fit

Problem-Solution fit						
Define CS, fit into CC	<b>1. CUSTOMER SEGMENT(S)</b> Who is your customer? I.e. working parents of 0-5 y.o. kids  People who have the habit of reading news regularly	CS	<b>6. CUSTOMER CONSTRAINTS</b> What constraints prevent your customers from taking action or limit their choices of solutions? I.e. spending power, budget, no cash, network connection, available devices.  1. Minimum cost 2. Instant updation on trending news 3. Weather report on daily basis based on the location	CC	<b>5. AVAILABLE SOLUTIONS</b> Which solutions are available to the customers when they face the problem or need to get the job done? What have they tried in the past? What pros & cons do these solutions have? I.e. pen and paper is an alternative to digital notetaking  Cell Phones Can Be Accused of Less Current Instead of TVs and Radios, Helpful to Utilize and Can be Effectively Conveyed to All over.	AS
	<b>2. JOBS-TO-BE-DONE / PROBLEMS</b> Which jobs-to-be-done (or problems) do you address for your customers? There could be more than one; explore different sides.  1. Fake news to be detected 2. Reduce time delay on news updates 3. Decrease the amount of irrelevant news 4. Categorization of news	J&P	<b>9. PROBLEM ROOT CAUSE</b> What is the real reason that this problem exists? What is the back story behind the need to do this job? I.e. customers have to do it because of the change in regulations.  1. Due to the increase of information sharing through the internet there is more possibility of origin of fake news. 2. Making the users to trust the information provided by the application	RC	<b>7. BEHAVIOUR</b> What does your customer do to address the problem and get the job done? I.e. directly related: find the right solar panel installer, calculate usage and benefits; indirectly associated: customers spend free time on volunteering work (I.e. Greenpeace)  1. It should provide information based on the interest of user. 2. Easy access of user interface. 3. Easy search options	BE
Identify strong TR & EM	<b>3. TRIGGERS</b> What triggers customers to act? I.e. seeing their neighbour installing solar panels, reading about a more efficient solution in the news.  The users is unable to reach and fulfill the needs of their customers due to which the news tracker app is not getting improved and wants a solution.	TR	<b>10. YOUR SOLUTION</b> If you are working on an existing business, write down your current solution first, fill in the canvas, and check how much it fits reality. If you are working on a new business proposition, then keep it blank until you fill in the canvas and come up with a solution that fits within customer limitations, solves a problem and matches customer behaviour.  1. Improve app usage and speed. 2. Use privacy and data security 3. Authorized news from trustable sources 4. Responsive user interface	SL	<b>8. CHANNELS of BEHAVIOUR</b> <b>8.1 ONLINE</b> What kind of actions do customers take online? Extract online channels from #7  1. Simple and friendly user interface 2. Periodical notification of trending news 3. On going development and support strategy 4. Multiplatform Support	CH
	<b>4. EMOTIONS: BEFORE / AFTER</b> The creators and developers of the application feels less interactive with the customer as we are not getting instant Notifications and updates upto the market and feels insecure and drowns to express and causes failure	EM				



# REQUIREMENT ANALYSIS

## Functional Requirement

Following are the functional requirements of the proposed solution.

FR No.	Functional Requirement (Epic)	Sub Requirement (Story / Sub-Task)
FR-1	User Installation	Users can access the website using any browser.
FR-2	User Registration	Registration throughForm 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 nameand password

## Non Functional Requirement

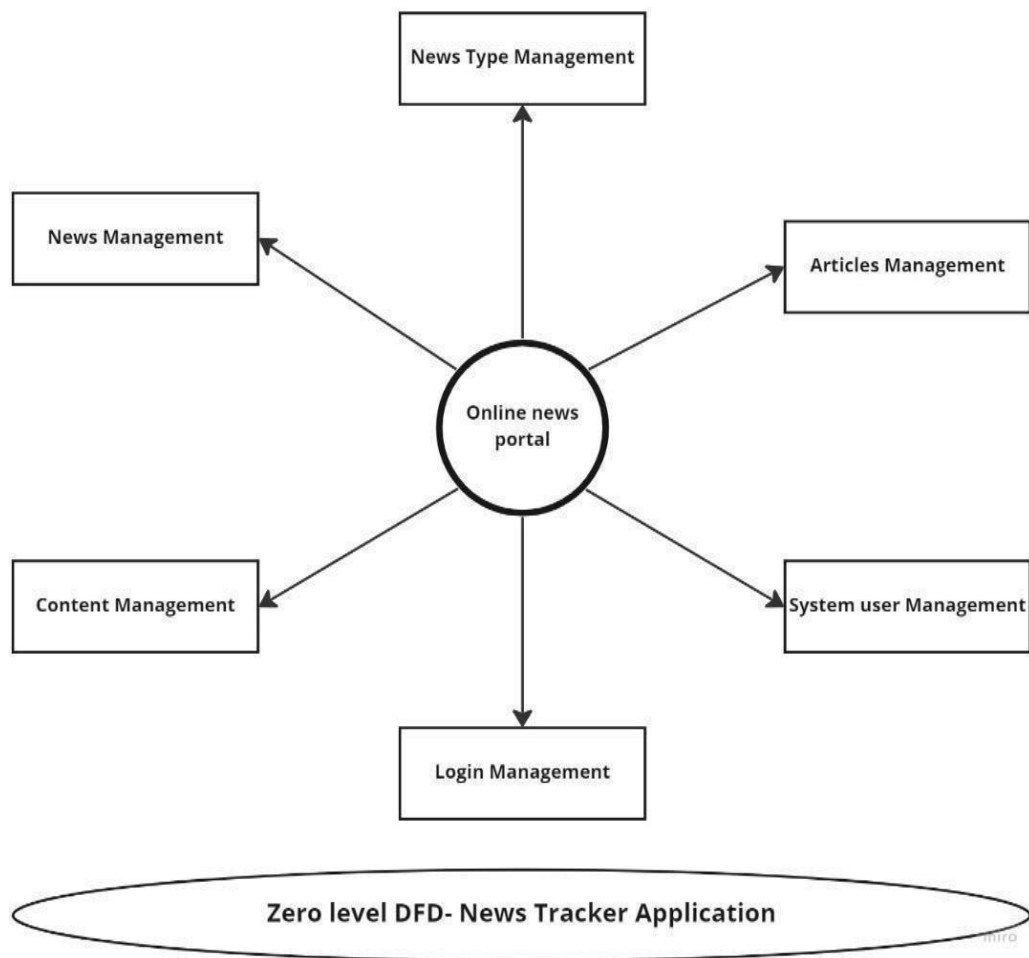
Following are the non-functional requirements of the proposed solution.

FR No.	Non-Functional Requirement	Description
NFR-1	<b>Usability</b>	Everyone can understand the process of using the appeasily by the commands given in the app.
NFR-2	<b>Security</b>	It is a secured app. No fake news can be shared.
NFR-3	<b>The performance</b>	Performance of the app is very great
NFR-4	<b>Availability</b>	More subcategories are available

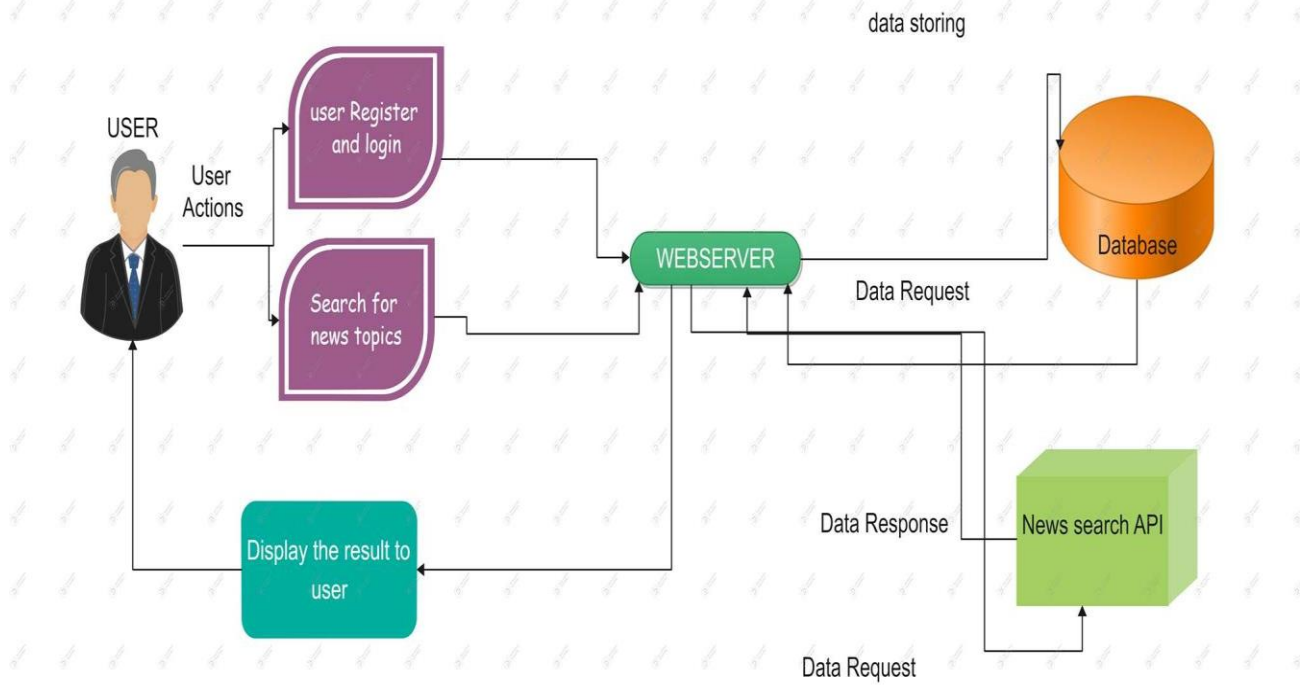
# PROJECT DESIGN

## Data Flow Diagrams

A Data Flow Diagram (DFD) is a traditional visual representation of the information flows within a system. A neat and clear DFD can depict the right amount of the system requirement graphically. It shows how data enters and leaves the system, what changes the information, and where data is stored.



## Solution & Technical Architecture



## User Stories

User Type	Functional Requirement (Epic)	User Story Number	User Story / Task	Acceptance criteria	Priority	Release
Customer (Mobile user)	Installation	USN-1	As a user, I installed the app for day-to-day updates and feeds.	My app will be installed on home screen		
Customer (Mobile user)		USN-2	As a user, I can register for the application by entering my email, password, confirming my password and phone number.	I can access my account / dashboard	High	Sprint-1
		USN-3	As a user, I will receive a confirmation email once I have registered for the	I can receive confirmation emails &	High	Sprint-1
		USN-4	As a user, I can register for the application through Gmail	I can register & access the dashboard with a Gmailaccount Login	Medium	Sprint-1
	Login	USN-6	As a user, I can log into the application by entering email & password	I can login to the official page	High	Sprint-1
	Dashboard	USN-7	Day-to-day news, feeds, categories, technews and other updates	I can see all the news which I wanted	High	Sprint-1
Customer (Web user)	Browsing	<u>USN-8</u>	Enter the web site on the browser	I can even login through browser	Medium	Sprint-1

## PROJECT PLANNING & SCHEDULING

### Sprint Planning & Estimation

Sprint	Functional Requirement (Epic)	User Story Number	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 and confirming my password.	10	High	Nachiappan Nishanth Rishi Maheshkumar
Sprint-1		USN-2	As a user, I will receive a confirmation email once I have registered for the application	10	High	Nachiappan Nishanth Rishi Maheshkumar
Sprint-2	Login	USN-3	As a user, I can log into the application by entering my email & password	5	medium	Nachiappan Nishanth
Sprint 2	Dashboard	USN-4	As a user, I should be able to navigate and access all the features hassle-free	5	low	Nishanth Maheshkumar
Sprint 2	Home page	USN-5	Interface for displaying the news headlines and important news with related images.	5	High	Nachiappan Nishanth Rishi Maheshkumar
Sprint 2		USN-6	Search related news	5	Medium	Nishanth Rishi

Sprint	Functional Requirement (Epic)	User Story Number	User Story / Task	Story Points	Priority	Team Members
Sprint 3	News Card	USN-7	Display the selected news article with its related images and related content	20	high	Nachiappan Nishanth Rishi Maheshkumar
Sprint 4	Save News Article	USN-8	Enable the user to save the news article for later use.	20	medium	Nachiappan Nishanth Rishi Maheshkumar

## Sprint Delivery Schedule

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	20	6 Days	24 Oct 2022	29 Oct 2022	20	29 Oct 2022
Sprint-2	20	6 Days	31 Oct 2022	05 Nov 2022	20	05 Nov 2022
Sprint-3	20	6 Days	07 Nov 2022	12 Nov 2022	20	05 Nov 2022
Sprint-4	20	6 Days	14 Nov 2022	19 Nov 2022	20	19 Nov 2022

$$AV = \frac{\textit{sprint duration}}{\textit{velocity}} = \frac{20}{10} = 2$$

# **CODING & SOLUTIONING**

## **FEATURE 1:**

This application allows users to search news and also view news based on set of categories like sports, entertainment , politics,etc

### **Code:**

```
@app.route('/category/<category>')
def newsCategory(category):
    news_articles = getCategory(category)
    return render_template("news.html", news_articles=news_articles)

@app.route('/search/<text>')
def newsSearch(text):
    news_articles = searchNews(text)
    return render_template("news.html", news_articles=news_articles)

if __name__ == "__main__":
    app.run(host='0.0.0.0', port=5000, debug=True)

import requests
from decouple import config
NEWS_API_KEY = config('NEWS_API_KEY')
# COUNTRY = 'in'

def get_latest_news():
    news_data = requests.get(f'https://newsapi.org/v2/top-headlines?country=us&apiKey={NEWS_API_KEY}').json()
    return news_data['articles']

def getCategory(category):
    news_data = requests.get(f'https://newsapi.org/v2/top-headlines?country=us&category={category}&apiKey={NEWS_API_KEY}').json()
    news_data = requests.get(f'https://newsapi.org/v2/top-headlines?country=us&category={category}&apiKey={NEWS_API_KEY}').json()
```

```
return news_data['articles']
```

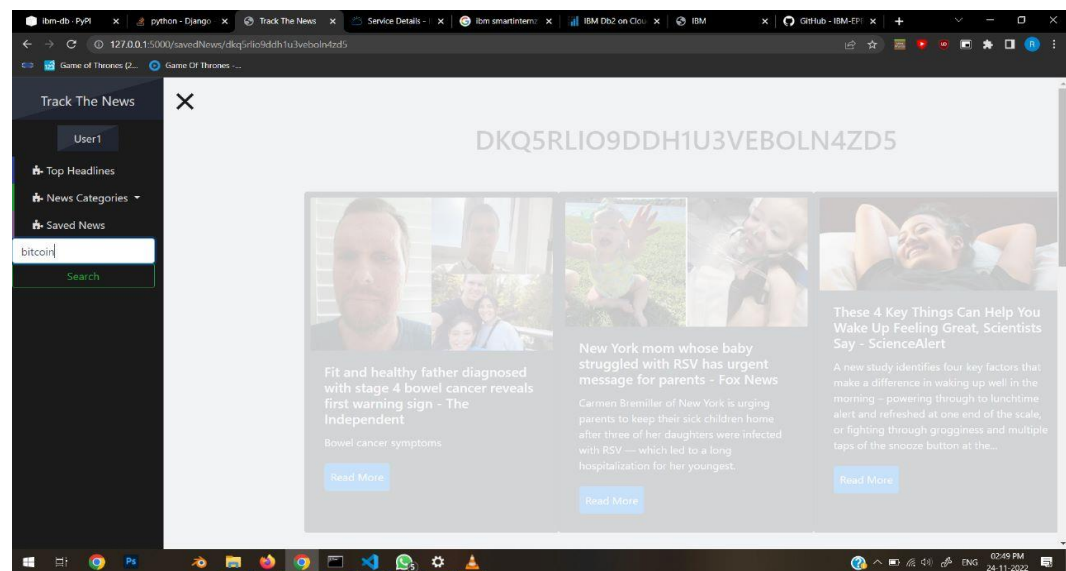
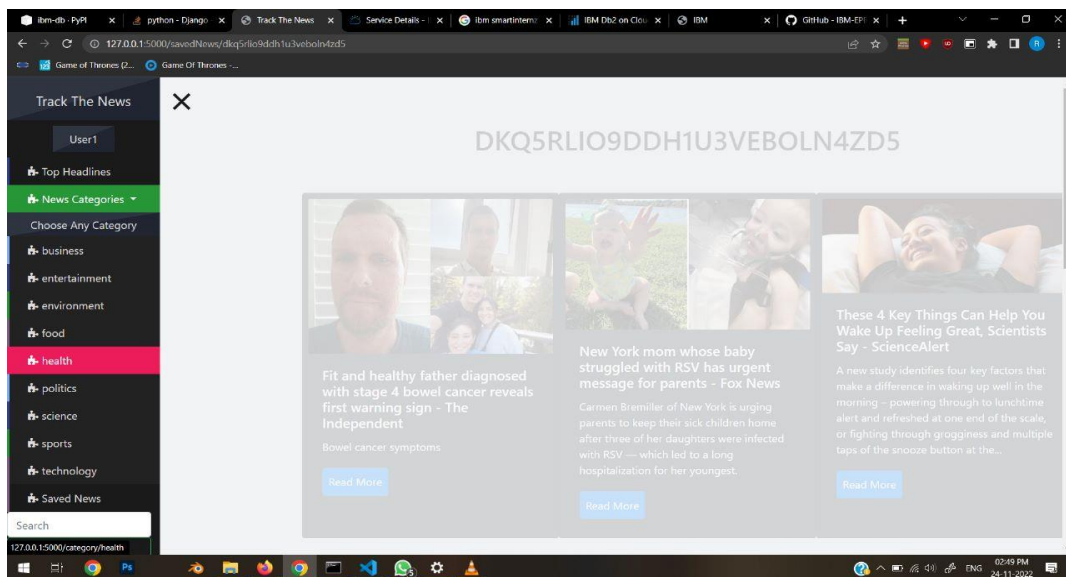
```
def searchNews(text):
```

```
    news_data = requests.get(f'https://newsapi.org/v2/top-  
headlines?country=us&q={text}&apiKey={NEWS_API_KEY}').json()
```

```
    news_data =
```

```
    requests.get(f'https://newsapi.org/v2/everything?q={text}&apiKey={NEWS_API_KEY}').json()
```

```
    return news_data['articles']
```





## FEARTUE 2:

This application also allows the users to save the news which is already viewed. The saved news can be accessed later.

### Code:

```
@app.route('/addNews',methods=['POST'])
def addNews():
    value =
addNewsDb(json.loads(request.get_data())['id'],json.loads(request.get_data())['newsLink'],json.loads(request.get_data())['newsTitle'],json.loads(request.get_data())['newsImage'],json.loads(request.get_data())['newsDescrp'])
    return value

@app.route('/savedNews/<id>', methods=['GET', 'POST'])
def showSavedNews(id):
    news_articles = get_savedNews(id)
    return render_template("savedNews.html", news_articles=news_articles)

def get_savedNews(id):
    stmt = ibm_db.exec_immediate(conn, "select * from news where USERID='"+id+"'")
    article = []
    if ibm_db.fetch_row(stmt)== True:
        # print(ibm_db.fetch_assoc(stmt))
        value = ibm_db.fetch_assoc(stmt)

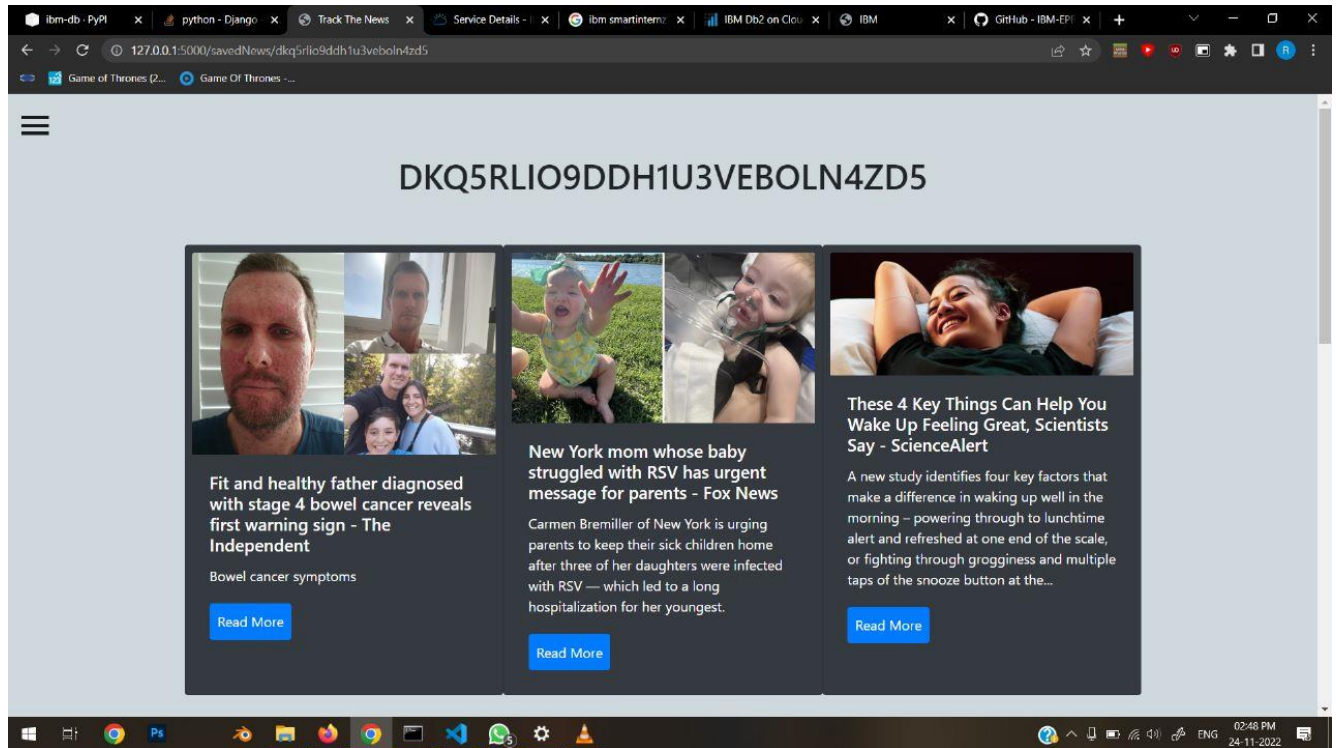
        while value != False:
            dict = {'image_url':value["IMAGE"] , 'title': value["TITLE"] , 'link': value["LINK"],'description':
value["DESCRP"]}
            article.append(dict)
            value = ibm_db.fetch_assoc(stmt)
        return article
    return article

def addNewsDb(id,link,title,image , descrp):
    stmt = ibm_db.exec_immediate(conn, "select * from news where LINK='"+link+"'")
    if ibm_db.fetch_row(stmt)== False:
```

```

stmt2 = "insert into news (USERID, TITLE, IMAGE, LINK ,DESCRP ) values
('"+id+"','"+title+"','"+image+"','"+link+"', '"+descrip+"');"
if ibm_db.exec_immediate(conn, stmt2):
    return '{"result": "Done"}'
return '{"result": "Already Found"}'
return '{"result": "Already Found"}'

```



# TESTING

## ACCEPTANCE TESTING:

### 1. Purpose of Document

The purpose of this document is to briefly explain the test coverage and open issues of the[newsTracker ] project at the time of the release to User Acceptance Testing (UAT).

### 2. Defect Analysis

This report shows the number of resolved or closed bugs at each severity level, and how they were resolved

Resolution	Severity1	Severity2	Severity3	Severity4	Subtotal
By Design	1	3	2	1	7
Duplicate	1	0	0	0	1
External	0	0	0	1	1
Fixed	2	3	2	1	8
Not Reproduced	0	0	0	0	0
Skipped	0	0	1	0	1
Won'tFix	0	0	0	0	0
Totals	4	6	5	3	18

### 3. Test Case Analysis

This report shows the number of testcases that have passed, failed, and untested

Section	Total Cases	Not Tested	Fail	Pass
External API	8	0	0	8
Input Fields(XSS)	28	0	0	28
Input Fields(CSRF)	12	0	0	12
Python Template Injection	8	0	0	8

Exception Reporting	10	0	0	10
JSON Parsing	80	0	0	80

**PERFORMANCE TESTING:**


NFT - Risk Assessment									
S.No	Project Name	Scope/feature	Functional Changes	Hardware Changes	Software Changes	Impact of Downtime	Load/Volume Changes	Risk Score	Justification
1	News Alholic	Existing	Low	No Changes	No Changes		>5 to 10%	GREEN	As we have seen the chnages
2	News Alholic	SignIn	High	No Changes	Code flow and DB	Unable to SignIn	>5 to 15%	RED	Minimal data is processed for improved latency
3	News Alholic	SignUp	High	No Changes	Code flow and DB	Unable to SignUp	>5 to 15%	RED	Minimal data is processed for improved latency
4	News Alholic	Explore	Moderate	No Changes	Code flow and API	Explore page not loading	>5 to 20%	GREEN	API data is processed to BSON
5	News Alholic	Saved	Moderate	No Changes	Code flow and API	Saved changes not reflected	>5 to 20%	GREEN	API data is processed to BSON
NFT - Detailed Test Plan									
S.No	Project Overview	NFT Test approach	Assumptions/Dependencies/Risks	Approvals/SignOff					
1	News Alholic	Usability	Testing for UI	Usability					
2	News Alholic	Security	Endpoints testing and User signin testing	Security					
3	News Alholic	Performance	Performance testing and lighthouse	Performance					
4	News Alholic	Maintainability	Testing for code readability	Maintainability					
End Of Test Report									
S.No	Project Overview	NFT Test approach	NFR - Met	Test Outcome	GO/NO-GO decision	Recommendations	Identified Defects (Detected/Closed/Open)	Approvals/SignOff	
1	News Alholic	Usability	Testing for UI	UI breaks for mediaquery	GO	Fix mediaquery and breakpoint	Mobile View - open	Mobile View - Testing	
2	News Alholic	Security	Endpoints testing and User signin testing	Endpoints exposed	GO	Add security to endpoints	Endpoints security - open	Endpoints security - Testing	
3	News Alholic	Performance	Performance testing and lighthouse	Performance issue for fetch	GO	Fix unwanted db Access and API fetch	API fetch - Detected	API fetch - Testing	
4	News Alholic	Maintainability	Testing for code readability	No comments for certain places	GO	Write comments	Write comments - fixed	Write comments - Testing	

# LIGHTHOUSE TOOL TESTING:

## DESKTOP:

11/25/22, 5:43 PM

about:blank

 http://159.122.187.194:32662/

89

Performance

94

Accessibility

92

Best Practices

89

SEO

PWA

PWA

89


Performance

Values are estimated and may vary. The [performance score is calculated](#) directly from these metrics. [See calculator.](#)

▲ 0–49

50–89

90–100



METRICS

Expand view

First Contentful Paint

0.7 s

Time to Interactive

0.7 s

▲ Speed Index

16.4 s

Total Blocking Time

0 ms


Largest Contentful Paint


0.7 s

Cumulative Layout Shift

0

View Original Trace

 View Treemap



Show audits relevant to:

All

FCP

TBT

LCP

CLS

OPPORTUNITIES

Estimated Savings

Opportunity



about:blank

1/5

MOBILE:

11/25/22, 5:45 PM

about:blank

 http://159.122.187.194:32662/ 

84

Performance

94

Accessibility

92

Best Practices

91

SEO

PWA

PWA


84

Performance

Values are estimated and may vary. The [performance score](#) is calculated directly from these metrics. [See calculator.](#)

▲

0–4950–8990–100



METRICS

Expand view

First Contentful Paint

2.3 s

Time to Interactive

2.5 s

▲ Speed Index

30.5 s

Total Blocking Time

0 ms


Largest Contentful Paint


2.6 s

Cumulative Layout Shift

0

View Original Trace

 View Treemap



Show audits relevant to:

All

FCP

TBT

LCP

CLS

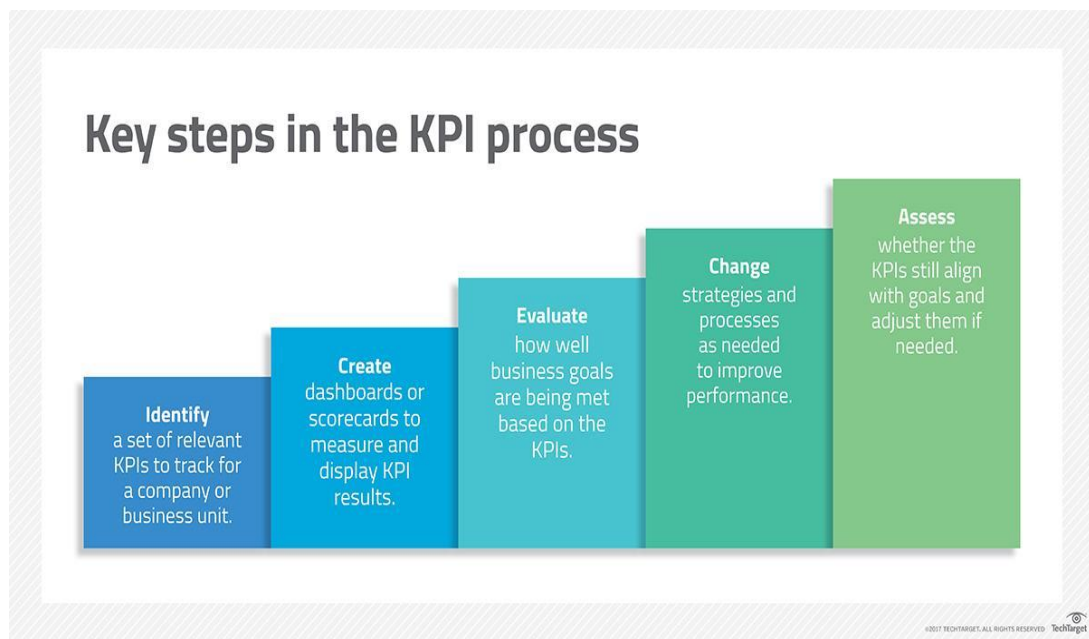
about:blank

1/5

## RESULTS

### PERFORMANCE MATRIXS

These application performance metrics, commonly known as key performance indicators (KPIs), are a quantitative measure of how effectively the organization achieves the business objectives. Capturing the right metrics will give you a comprehensive report and powerful insights into ways to improve your application.



# **ADVANTAGES & DISADVANTAGES**

## **ADVANTAGES**

- Real-time application is allowed and has live layout.
- It has a Built-in support for Google Platform.
- News accuracy of specialised API
- Authenticated users are allowed
- Up to date and daily information is available

## **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.



## **CONCLUSION**

In our project work, an attempt has been made to develop a News or information-based website. We develop this project that helps people and make them aware so that they can know any news. To establish this website we use various methodologies. To develop this project we faced many problems but we hardly tried to develop this project. The project we've offered here is just the beginning of a new way of interacting with our society. In the meantime, don't forget that compelling visual content will help you be more visible and viral than offline or online newspapers.

## **FUTURE SCOPE**

- ☐ We are in a process of developing an algorithm that will help the user to read the new postings and news from his recent data sources.
- ☐ In pandemic situations, offline news won't be delivered to anyone, in those times these news apps are the most suitable.
- ☐ In the future, we are going to develop new categories according to their user locations.

## APPENDIX

### SOURCE CODE:

#### App.py

```
# from crypt import methods
import json
from flask import Flask , request ,render_template , redirect , url_for
# from flask_cors import CORS
from DB_op import signinusersdb ,signupusersdb ,addNewsDb ,get_savedNews;
from utils import get_latest_news , getCategory,searchNews
app = Flask(__name__,template_folder = 'templates')

class sessionobj:
    def __init__(self,email):
        self.email = email

@app.route('/')
def welcome():
    return render_template("login.html")

@app.route('/signinusers',methods=['POST'])
def signinusers():
    value = signinusersdb(json.loads(request.get_data())['id'],json.loads(request.get_data())['password'])
    return value

@app.route('/signupusers',methods=['POST'])
def signupusers():
    value =
    signupusersdb(json.loads(request.get_data())['name'],json.loads(request.get_data())['email'],json.loads(request
.get_data())['password'])
    # print(json.loads(request.get_data()))
    return value

@app.route('/addNews',methods=['POST'])
def addNews():
    value =
    addNewsDb(json.loads(request.get_data())['id'],json.loads(request.get_data())['newsLink'],json.loads(request.
```

```
get_data())['newsTitle'],json.loads(request.get_data())['newsImage'],json.loads(request.get_data())['newsDesc  
rp']])
```

```
    return value
```

```
@app.route('/news')
```

```
def news_headlines():
```

```
    news_articles = get_latest_news()
```

```
    # print(news_articles)
```

```
    return render_template("news.html", news_articles=news_articles)
```

```
    # return news_articles
```

```
@app.route('/category/<category>')
```

```
def newsCategory(category):
```

```
    news_articles = getCategory(category)
```

```
    return render_template("news.html", news_articles=news_articles)
```

```
@app.route('/savedNews/<id>', methods=['GET', 'POST'])
```

```
def showSavedNews(id):
```

```
    news_articles = get_savedNews(id)
```

```
    return render_template("savedNews.html", news_articles=news_articles)
```

```
@app.route('/search/<text>')
```

```
def newsSearch(text):
```

```
    news_articles = searchNews(text)
```

```
    return render_template("news.html", news_articles=news_articles)
```

```
@app.route('/signup')
```

```
def signup():
```

```
    return render_template("signup.html")
```

```
@app.route('/dashboard')
```

```
def dashboard():
```

```
    return render_template("dashboard.html")
```

```
if __name__ == "__main__":
```

```
    app.run(host='0.0.0.0', port=5000, debug=True)
```

## DB\_op.py

```
import ibm_db

import random

import string

from hashing import *

conn =
    ibm_db.connect("DATABASE=bludb;HOSTNAME=2d46b6b4-
cbf6-40eb-bbce-
6251e6ba0300.bs2io90l08kqb1od8lcg.databases.appdomain.cloud
;PORT=32328;SECURITY=SSL;SSLServerCertificate=DigiCert
GlobalRootCA.crt;UID=bmd88020;PWD=wcxrFr0o8c29Ybco", "
", "")

print("Connection Successful")

print(conn)


def signinusersdb(email,password):

    stmt = ibm_db.exec_immediate(conn, "select
password,uid,random_key,name from users1 where
EMAIL='"+email+"';")

    while ibm_db.fetch_row(stmt) != False:

        value = ibm_db.result(stmt, 0)

        key = ibm_db.result(stmt, 1)

        randomid = ibm_db.result(stmt, 2)

        dname = ibm_db.result(stmt,3)

        # print(value)

        r = rehashing(value,password,key,randomid,dname)

        # print(r)

        return r

    return '{"result":"Not Found"}'
```

```

def signupusersdb(name,email,password):

    key = ".join(random.choices(string.ascii_lowercase +string.digits,
                                k=25))

    # key = ".join(random.choices(string.ascii_lowercase +string.digits,
                                k=25))

    arr = hashing(password)

    stmt = "insert into users1 (NAME, EMAIL, PASSWORD,
                                random_key) values ('"+name+"','"+ email+"','"+ str(arr.decode())
                                +','"+ key+"');"

    if ibm_db.exec_immediate(conn, stmt):

        return '{"result":"Done"}'

    return '{"result":"Not Found"}'

```

```

def get_savedNews(id):

    stmt = ibm_db.exec_immediate(conn, "select * from news where
    USERID='"+id+"';")

    article = []

    if ibm_db.fetch_row(stmt)== True:

        # print(ibm_db.fetch_assoc(stmt))

        value = ibm_db.fetch_assoc(stmt)

        while value != False:

            dict = {'image_url':value["IMAGE"] , 'title': value["TITLE"] ,
                    'link': value["LINK"],'description': value["DESCRP"]}

            article.append(dict)

            value = ibm_db.fetch_assoc(stmt)

        return article

    return article

```

```

def addNewsDb(id,link,title,image , descrp):

    stmt = ibm_db.exec_immediate(conn, "select * from news where
    LINK='"+link+"';")

```

```

if ibm_db.fetch_row(stmt)== False:

    stmt2 = "insert into news (USERID, TITLE, IMAGE, LINK
,DESCRP ) values ('"+id+"','"+title+"','"+image+"','"+link+"",
    '"+descrip+"');"

    if ibm_db.exec_immediate(conn, stmt2):

        return '{"result":"Done"}'

    return '{"result":"Already Found"}'

return '{"result":"Already Found"}'

```

## hashing.py

```

import bcrypt

import os

def hashing(password):

    hash = bcrypt.hashpw(password.encode('utf-8'),bcrypt.gensalt())

    print(hash)

    return hash


def rehashing(key,password_to_check,usrkey,id,name):

    if bcrypt.checkpw(password_to_check.encode('utf-8'),key.encode('utf-8')):

        return
        '{"result":"Done","key":'+str(usrkey)+'","userid":'+str(id)+'","name":'+name+'"}'

        # return
        '{"result":"Done","key":'+str(usrkey)+'","userid":'+str(id)+'","name":'+name+'"}'

    else:

return '{"result":"Not Found"}'

```

## utils.py

```
import requests

from decouple import config

NEWS_API_KEY = config('NEWS_API_KEY')


def get_latest_news():
    news_data = requests.get(f'https://newsapi.org/v2/top-headlines?country=us&apiKey={NEWS_API_KEY}').json()
    return news_data['articles']


def getCategory(category):
    news_data = requests.get(f'https://newsapi.org/v2/top-headlines?country=us&category={category}&apiKey={NEWS_API_KEY}').json()
    news_data = requests.get(f'https://newsapi.org/v2/top-headlines?country=us&category={category}&apiKey={NEWS_API_KEY}').json()
    return news_data['articles']


def searchNews(text):
    news_data = requests.get(f'https://newsapi.org/v2/top-headlines?country=us&q={text}&apiKey={NEWS_API_KEY}').json()
    news_data = requests.get(f'https://newsapi.org/v2/everything?q={text}&apiKey={NEWS_API_KEY}').json()
    return news_data['articles']
```

## script.js

```
$(document).ready(function () {
    var trigger = $('.hamburger'),
        overlay = $('.overlay'),
        isClosed = false;

    trigger.click(function () {
        hamburger_cross();
    });

    function hamburger_cross() {

        if (isClosed == true) {
            overlay.hide();
            trigger.removeClass('is-open');
```

```

        trigger.addClass('is-closed');
        isClosed = false;
    } else {
        overlay.show();
        trigger.removeClass('is-closed');
        trigger.addClass('is-open');
        isClosed = true;
    }
}

$('[data-toggle="offcanvas"]').click(function () {
    $('#wrapper').toggleClass('toggled');
});
});

```

## base.html

```

<!DOCTYPE html>
<html lang="en">

<head>
    <meta charset="UTF-8">
    <title>Track The News</title>

    <link rel='stylesheet' href='https://stackpath.bootstrapcdn.com/bootstrap/4.1.3/css/bootstrap.min.css'>
    <link rel='stylesheet' href='https://cdnjs.cloudflare.com/ajax/libs/animate.css/3.7.0/animate.min.css'>
    <link rel='stylesheet' href='https://maxcdn.bootstrapcdn.com/font-awesome/4.7.0/css/font-awesome.min.css'>
    <link rel='stylesheet' href='https://cdnjs.cloudflare.com/ajax/libs/tether/1.4.4/css/tether.min.css'>
    <link rel="stylesheet" href="{ {url_for('static',filename='base.css')} }">
</head>

<body>
    <!-- partial:index.partial.html -->
    { % block navbar % }
    <div id="wrapper">
        <div class="overlay"></div>

```



```
<!-- Sidebar -->
<nav class="navbar navbar-inverse fixed-top" id="sidebar-wrapper" role="navigation">
  <ul class="nav sidebar-nav">
    <div class="sidebar-header">
      <div class="sidebar-brand">
        <a href="#">Track The News</a>
      </div>
    </div>
    <div class="dropdown-header my-2 mx-auto" id="userName">User Name</div>
    <li><a href="/news">Top Headlines </a></li>
    <li class="dropdown">
      <a href="#works" class="dropdown-toggle" data-toggle="dropdown">News Categories <span
        class="caret"></span></a>
      <ul class="dropdown-menu animated fadeInLeft" role="menu">
        <div class="dropdown-header">Choose Any Category</div>
        <li><a href="/category/business">business</a></li>
        <li><a href="/category/entertainment">entertainment</a></li>
        <li><a href="/category/environment">environment</a></li>
        <li><a href="/category/food">food</a></li>
        <li><a href="/category/health">health</a></li>
        <li><a href="/category/politics">politics</a></li>
        <li><a href="/category/science">science</a></li>
        <li><a href="/category/sports">sports</a></li>
        <li><a href="/category/technology">technology</a></li>
        <!-- <li><a href="/category/top">top</a></li> -->
        <!-- <li><a href="/category/world">world</a></li> -->
      </ul>
    </li>
    <li><a id="savedNews">Saved News </a></li>

    <div class="d-flex flex-column">
      <input class="form-control me-2" type="text" id="searchContent" placeholder="Search" aria-label="Search">
      <button class="btn btn-outline-success" type="button" onclick="searchText(this)">Search</button>
    </div>
    <!-- <li><a href="#services">Services</a></li>
    <li><a href="#contact">Contact</a></li>
    <li><a href="#followme">Follow me</a></li> -->
```

</ul>

</nav>

<!-- /#sidebar-wrapper -->

<!-- Page Content -->

<div id="page-content-wrapper">

<button type="button" class="hamburger animated fadeInLeft is-closed" data-toggle="offcanvas">

<span class="hamb-top"></span>

<span class="hamb-middle"></span>

<span class="hamb-bottom"></span>

</button>

<div class="container">

<h1 class="text-center" id="change">Top HeadLines</h1>

<script>

var base\_url = window.location.origin

console.log(base\_url);

function searchText(e) {

var text = document.getElementById("searchContent").value

window.location.href = base\_url + "/search/" + text;

}

loc = window.location

arr = loc.toString().split("/")

document.getElementById("change").innerHTML = arr[arr.length - 1].toUpperCase()

document.getElementById("userName").innerHTML = localStorage.getItem("NTname")

document.getElementById("savedNews").href = "/savedNews/" + localStorage.getItem("NTid")

</script>

{ % endblock % }

{ % block content % }

{ % endblock % }

```

    </div>
</div>
<!-- /#page-content-wrapper -->

</div>
<!-- /#wrapper -->
<!-- partial -->
<script src='https://code.jquery.com/jquery-3.3.1.slim.min.js'></script>
<script src='https://cdnjs.cloudflare.com/ajax/libs/popper.js/1.14.3/umd/popper.min.js'></script>
<script src='https://stackpath.bootstrapcdn.com/bootstrap/4.1.3/js/bootstrap.min.js'></script>
<script src='https://cdnjs.cloudflare.com/ajax/libs/tether/1.4.4/js/tether.min.js'></script>
<script src="{ {url_for('static',filename='script.js')} }"></script>

</body>

</html>

```

## Login.html

```

<!doctype html>
<html lang="en">

<head>
  <!-- Required meta tags -->
  <meta charset="utf-8">
  <meta name="viewport" content="width=device-width, initial-scale=1, shrink-to-fit=no">

  <!-- Bootstrap CSS -->
  <link rel="stylesheet" href="https://cdn.jsdelivr.net/npm/bootstrap@4.6.2/dist/css/bootstrap.min.css"
    integrity="sha384-xOolHFLEh07PJGoPkLv1IbcEPTNtaed2xpHsD9ESMhqIYd0nLMwNLD69Npy4HI+N"
    crossorigin="anonymous">

  <link rel="stylesheet" href="{ {url_for('static',filename='login.css')} }">
  <!-- <script type="text/javascript" src="http://ajax.googleapis.com/ajax/libs/jquery/1.4.2/jquery.min.js"></script>
  <script src="https://code.jquery.com/jquery-3.4.1.min.js" integrity="sha256-
CSXorXvZcTkaix6Yvo6HppcZGetbYMGWSFlBw8HfCJo=" crossorigin="anonymous"></script> -->

```

```
<!-- <script src="https://accounts.google.com/gsi/client" async defer></script> -->
```

```
<title>News Tracker</title>
```

```
</head>
```

```
<body class="">
```

```
<div class="wrapper fadeInDown">
```

```
<div id="formContent">
```

```
<!-- Tabs Titles -->
```

```
<h2 class="active"> LOG IN </h2>
```

```
<!-- Icon -->
```

```
<!-- <div class="fadeIn first">
```

```

```

```
</div> -->
```

```
<!-- Login Form -->
```

```
<form>
```

```
<input type="text" id="exampleFormControlInput" class="fadeIn second" name="login" placeholder="Email">
```

```
<input type="text" id="exampleFormControlInput1" class="fadeIn third" name="login" placeholder="Password">
```

```
<input type="button" class="fadeIn fourth" onclick="signinusers()" value="Log In">
```

```
</form>
```

```
<p>Not Registered ? Please <a href="/signup">Signup</a></p>
```

```
</div>
```

```
</div>
```

```
<script>
```

```
function signinusers() {
```

```
  var uname = document.getElementById('exampleFormControlInput').value;
```

```
  var upass = document.getElementById('exampleFormControlInput1').value;
```

```
  fetchres = fetch('/signinusers', {
```

```
    method: 'POST',
```

```
    body: JSON.stringify({ id: uname, password: upass }),
```

```
    contentType: "application/json"
```

```
  })
```

```
  fetchres.then(res => { return res.json() })
```

```

.then(
  d => {
    console.log(d)
    if (d.result == "Done")
{localStorage.setItem("NTkey",String(d.key));localStorage.setItem("NTname",String(d.name));localStorage.setItem("NTi
d",String(d.userid));window.location.replace('/news');}
    else if (d.result == "Not Found") {
      document.getElementById("mainbody").innerHTML +=
      `

```

<!-- <div id="rockectimage" class="img-fluid">
  
</div>

```



```

<div id="newsreportimage" class="img-fluid">
  
</div>

```



```

<div id="chartimage" class="img-fluid">
  
</div> -->

```



```

<!-- Optional JavaScript; choose one of the two! -->

```



```

<!-- Option 1: jQuery and Bootstrap Bundle (includes Popper) -->

```



```

<script src="https://cdn.jsdelivr.net/npm/jquery@3.5.1/dist/jquery.slim.min.js"
  integrity="sha384-DfXdz2htPH0lsSSs5nCTpuj/zy4C+OGpamoFVy38MVBnE+IbbVYUew+OrCXaRkfj"
  crossorigin="anonymous"></script>

```


```

```

<script src="https://cdn.jsdelivr.net/npm/bootstrap@4.6.2/dist/js/bootstrap.bundle.min.js"
  integrity="sha384-Fy6S3B9q64WdZWQUiU+q4/2Lc9npb8tCaSX9FK7E8HnRr0Jz8D6OP9dO5Vg3Q9ct"
  crossorigin="anonymous"></script>
<!-- Option 2: Separate Popper and Bootstrap JS -->
<!--
  <script src="https://cdn.jsdelivr.net/npm/jquery@3.5.1/dist/jquery.slim.min.js" integrity="sha384-
DfXdz2htPH0lsSSs5nCTpuj/zy4C+OGpamoFVy38MVBnE+IbbVYUew+OrCXaRkfj"
  crossorigin="anonymous"></script>
  <script src="https://cdn.jsdelivr.net/npm/popper.js@1.16.1/dist/umd/popper.min.js" integrity="sha384-
9/reFTGAW83EW2RDu2S0VKA1Zap3H66lZ81PoYlFhbGU+6BZp6G7niu735Sk7lN"
  crossorigin="anonymous"></script>
  <script src="https://cdn.jsdelivr.net/npm/bootstrap@4.6.2/dist/js/bootstrap.min.js" integrity="sha384-
+sLIOodYLS7CIrQpBjl+C7nPvqq+FbNUBDunl/OZv93DB7Ln/533i8e/mZXLi/P+" crossorigin="anonymous"></script>
-->
</body>

</html>

```

## signup.html

```

<!doctype html>
<html lang="en">

<head>
  <!-- Required meta tags -->
  <meta charset="utf-8">
  <meta name="viewport" content="width=device-width, initial-scale=1, shrink-to-fit=no">

  <!-- Bootstrap CSS -->
  <link rel="stylesheet" href="https://cdn.jsdelivr.net/npm/bootstrap@4.6.2/dist/css/bootstrap.min.css"
    integrity="sha384-xOolHFLEh07PJGoPkLv1IbcEPTNtaed2xpHsD9ESMhqIYd0nLMwNLD69Npy4HI+N"
    crossorigin="anonymous">
  <link rel="stylesheet" href="{ {url_for('static',filename='login.css')}} ">
  <title>News Tracker</title>
</head>

```

```

<body class="mainbody container-lg">
  <div class="wrapper fadeInDown">
    <div id="formContent">
      <!-- Tabs Titles -->
      <h2 class="active"> SIGN UP </h2>
      <!-- Icon -->
      <!-- <div class="fadeIn first">
        
      </div> -->

      <!-- Login Form -->
      <div>
        <input type="text" id="exampleFormControlInput1" class="fadeIn second" name="signup" placeholder="Name">
        <input type="text" id="exampleFormControlInput2" class="fadeIn third" name="signup" placeholder="Email">
        <input type="text" id="exampleFormControlInput3" class="fadeIn third" name="signup" placeholder="Password">
        <input type="text" id="exampleFormControlInput4" class="fadeIn third" name="signup"
          placeholder="Confirm Password">
        <input type="button" class="fadeIn fourth" onclick="signupusers()" value="REGISTER" >
      </div>
      <p>Already Registered ? Please <a href="/">Login</a></p>

    </div>
  </div>

<script>
function signupusers() {
  var uname = document.getElementById('exampleFormControlInput1').value;
  var uemail = document.getElementById('exampleFormControlInput2').value;
  var upass = document.getElementById('exampleFormControlInput3').value;
  fetchres = fetch('/signupusers', {
    method: 'POST',
    body: JSON.stringify({ name: uname, email: uemail, password: upass }),
    contentType: "application/json"
  });
  console.log(fetchres);
}

```

```

fetchres.then(res => { return res.json() })
.then(
  d => {
    console.log(d)
    if (d.result == "Done") { window.location.replace('/');}
    else if (d.result == "Not Found") {
      document.getElementById("mainbody").innerHTML +=
        `<div class="alert alert-danger alert-dismissible fade show" role="alert">
          <strong>Username or Password is Incorrect!</strong>
          <button type="button" class="close" data-dismiss="alert" aria-label="Close">
            <span aria-hidden="true">&times;</span>
          </button>
        </div>`
    }
  })
}

```

```

// function signupusers(){
//   var uname = document.getElementById('exampleFormControlInput1').value;
//   var uemail = document.getElementById('exampleFormControlInput2').value;
//   var upass = document.getElementById('exampleFormControlInput3').value;
//   fetch('/signupusers',{
//     method:'POST',
//     body:JSON.stringify({ name: uname,email:uemail,password: upass}),
//     contentType: "application/json"}).then(response => {
//       console.log(response.status);
//       if(response.status == 200){
//         console.log("Entered Switching if")
//         window.location('/news')
//       }
//     })
// }
</script>

```

<!-- Optional JavaScript; choose one of the two! -->



```

<!-- Option 1: jQuery and Bootstrap Bundle (includes Popper) -->
<script src="https://cdn.jsdelivr.net/npm/jquery@3.5.1/dist/jquery.slim.min.js"
  integrity="sha384-DfXdz2htPH0lsSSs5nCTpuj/zy4C+OGpamoFVy38MVBnE+IbbVYUew+OrCXaRkfj"
  crossorigin="anonymous"></script>
<script src="https://cdn.jsdelivr.net/npm/bootstrap@4.6.2/dist/js/bootstrap.bundle.min.js"
  integrity="sha384-Fy6S3B9q64WdZWQUiU+q4/2Lc9npb8tCaSX9FK7E8HnRr0Jz8D6OP9dO5Vg3Q9ct"
  crossorigin="anonymous"></script>

<!-- Option 2: Separate Popper and Bootstrap JS -->
<!--
  <script src="https://cdn.jsdelivr.net/npm/jquery@3.5.1/dist/jquery.slim.min.js" integrity="sha384-
DfXdz2htPH0lsSSs5nCTpuj/zy4C+OGpamoFVy38MVBnE+IbbVYUew+OrCXaRkfj"
crossorigin="anonymous"></script>
  <script src="https://cdn.jsdelivr.net/npm/popper.js@1.16.1/dist/umd/popper.min.js" integrity="sha384-
9/reFTGAW83EW2RDu2S0VKA1zap3H66lZH81PoYlFhbGU+6BZp6G7niu735Sk7lN"
crossorigin="anonymous"></script>
  <script src="https://cdn.jsdelivr.net/npm/bootstrap@4.6.2/dist/js/bootstrap.min.js" integrity="sha384-
+sLIOodYLS7CIrQpBjl+C7nPvqq+FbNUBDunl/OZv93DB7Ln/533i8e/mZXLi/P+" crossorigin="anonymous"></script>
-->
</body>

</html>

```

## news.html

```

{% extends 'base.html' %}

{% block content %}

<div class="container d-flex justify-center">
  <div class="row mt-5 ">
    <div class="col-md-12 ">
      <div class="row ">
        {% for news in news_articles %}
          <div class="card col-md-5 col-lg-4 mb-5 p-2 bg-dark text-white rounded">
            
            <div class="card-body">

```

```

<h5 class="card-title">{{ news.title }}</h5>
<p class="card-text">{{ news.description }}</p>
<a href="{{ news.url }}" class="btn btn-primary mt-1 p-2">Read More</a>
<button class="btn btn-secondary mt-1 p-2" onclick="saveArticle(this)">Save Article</button>
</div>

</div>
{% endfor %}
</div>
</div>
</div>
</div>

<!-- sweet alert -->
<script src="https://unpkg.com/sweetalert/dist/sweetalert.min.js"></script>

<script>
const saveArticle = (e) => {
  var uid = localStorage.getItem("NTid")
  var parent = e.parentElement
  var link = parent.childNodes[5].href
  var title = parent.childNodes[1].innerHTML
  var image_url = parent.parentElement.childNodes[1].src
  var descrp = parent.childNodes[3].innerHTML
  // console.log(descrp);
  fetchres = fetch('/addNews', {
    method: 'POST',
    body: JSON.stringify({ id: uid, newsLink: link, newsTitle: title, newsImage: image_url , newsDescrp:descrp }),
    contentType: "application/json"
  })
  fetchres.then(res => { return res.json() })
    .then(
      d => {
        console.log(d)
        if (d.result == "Done") { swal("Successful", "News Saved", "success"); }
        else if (d.result == "Already Found") { swal("Warning", "Already Saved", "warning"); }
      })
}

```

```
}
</script>
```

```
<!-- Bootstrap Bundle with Popper -->
{% endblock % }
```

## savednews.html

```
{% extends 'base.html' % }
```

```
{% block content % }
```

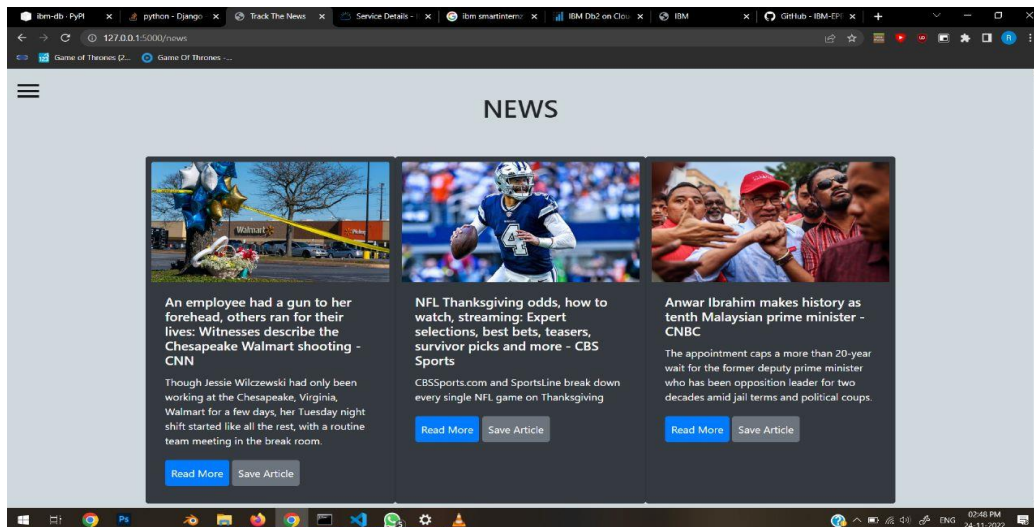
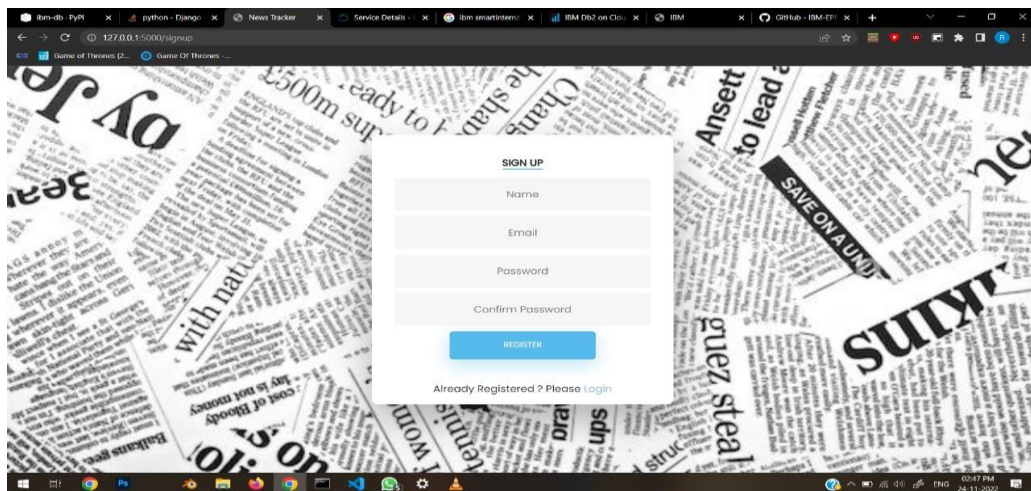
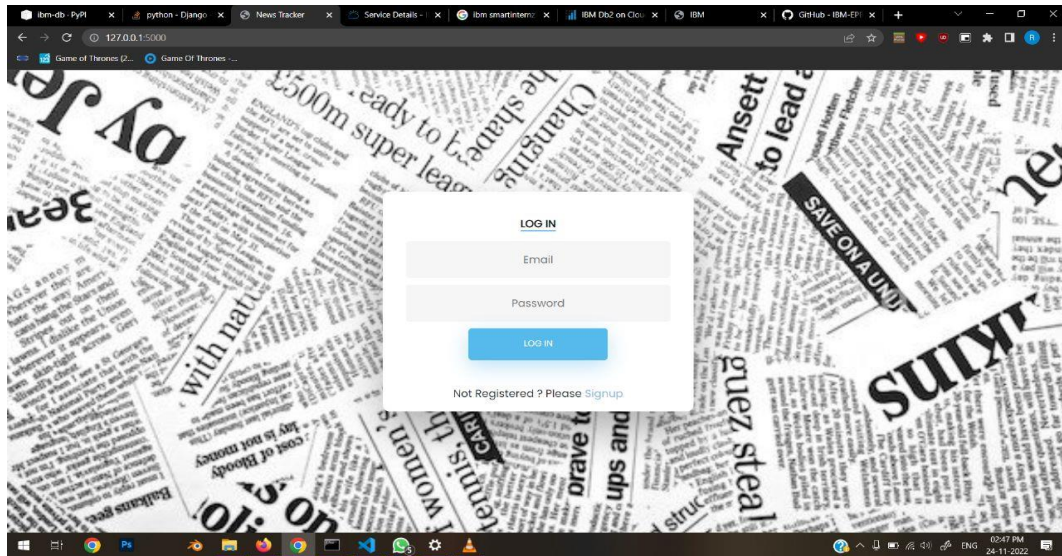
```
<div class="container d-flex justify-center">
  <div class="row mt-5 ">
    <div class="col-md-12 ">
      <div class="row ">
        {% for news in news_articles % }
          <div class="card col-md-5 col-lg-4 mb-5 p-2 bg-dark text-white rounded">
            
            <div class="card-body">
              <h5 class="card-title">{{ news.title }}</h5>
              <p class="card-text">{{ news.description }}</p>
              <a href="{{ news.link }}" class="btn btn-primary mt-1 p-2">Read More</a>
              <!-- <button class="btn btn-secondary mt-1 p-2" onclick="saveArticle(this)">Save Article</button> -->
            </div>
          </div>
        {% endfor % }
      </div>
    </div>
  </div>
</div>

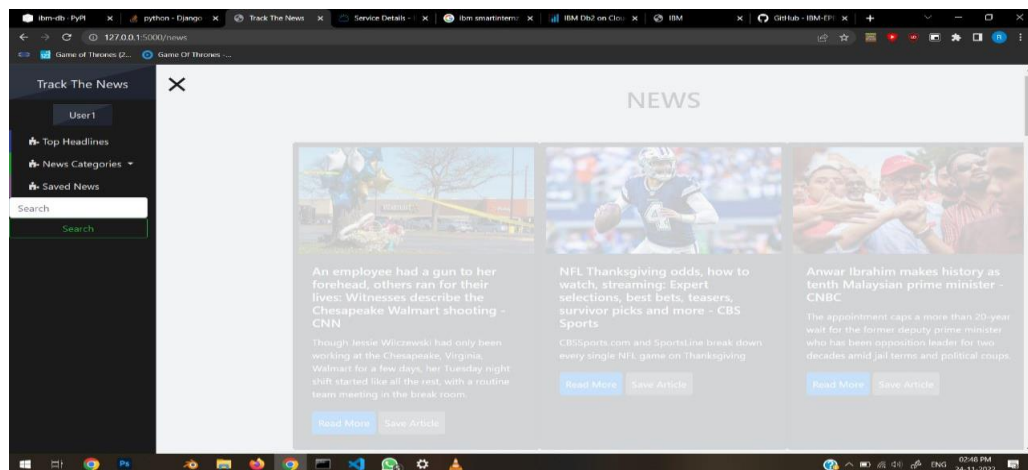
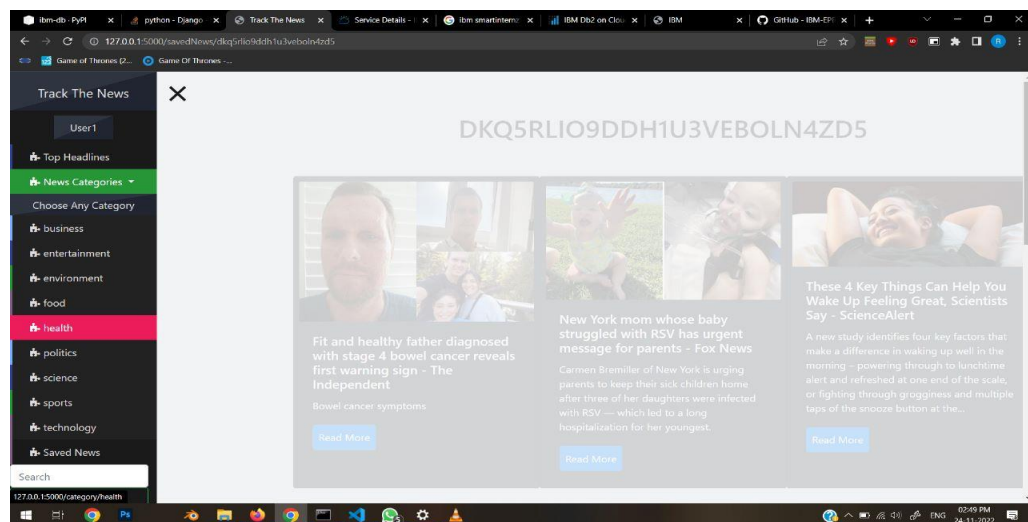
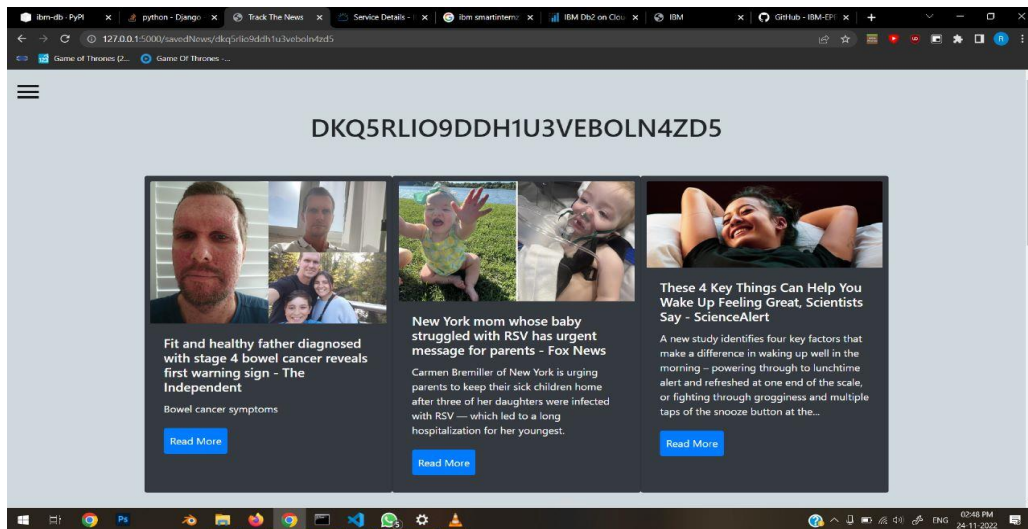
<!-- sweet alert -->

<!-- Bootstrap Bundle with Popper -->

{% endblock % }
```

## OUTPUT SCREENSHOTS :





**GITHUB LINK :** <https://github.com/IBM-EPBL/IBM-Project-25083-1659953504>

**PROJECT DEMO LINK :**

[https://drive.google.com/file/d/1\\_In25o5YYi9xWmCOL\\_zutV\\_hqRKTQABA/view?usp=sharing](https://drive.google.com/file/d/1_In25o5YYi9xWmCOL_zutV_hqRKTQABA/view?usp=sharing)