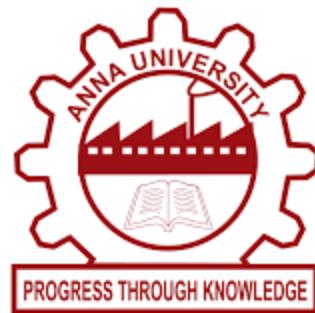


# NEWS TRACKER APPLICATION



**TEAM ID :** PNT2022TMID39302

**TEAM MEMBER :**

Logeshwaran C

# **INDEX**

## **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

## **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**

# **1. INTRODUCTION**

## **1.1 Project Overview**

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 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.

## **1.2 Purpose**

- This portal is open to public and end user or customer can come here and browse the desired news.
- You don't have to wait for newspaper in morning.
- Any time any news related to different fields will be available.
- We are developing an admin panel where our reputed clients can check day wise uploads, editing happen by authorized users.
- We are uploading the news of all categories happen in the world and it will be maintained by using database and real time algorithms.
- News sites offer the information to the public, political, social, sports, health, entertainment etc.
- Instant & latest news from all over the world.
- Easier accessibility.
- Fastest and Latest news updates as per the interest.
- Breaking News can update Minute-to-minute.
- Pictures of specials news events

## **2. LITERATURE SURVEY**

## 2.1 EXISTING PROBLEM:

YEAR	AUTHOR	TECHNIQUE	PROBLEM STATEMENT	PROS & CONS
<b>2018 &amp; MobiFeed: A Location-Aware News Feed System for Mobile Users*</b>	Wenjian Xu, Chi-Yin Chow, Man Lung Yiu, Qing Li, Chung Keung Poon	Location-aware social networks, news feed, mobile comput- ing, online scheduling	There is no location aware system in most of the application	Easy to access data  It doesn't get instant feedback.
<b>2018 &amp; Newsfeed System for Mobile Users</b>	Anuja Pise, Renuka Gangurde,Pooja Khairnar,Poonam Gamane	Spatial, Newsfeed, Prediction, Point-of interest, Diversity, Relevance.	News push notification ,News broadcast, Able to Automatically schedule newsfeed for mobile user based on their current and future location	Newsfeed Generation system for Android user Require Data/wifi to get Online.

<b>2019 &amp; GeoFeed: A Location-Aware News Feed System</b>	Require Data/wifi to get Online. Costly to Maintain.	The Spatial Pull Approach,The Spatial Push Approach,location aware system	(a) minimizes the system overhead for delivering the location-aware news feed, and (b) guarantees a certain response time for each user to obtain her location-aware news feed.	GeoFeed also supports location-aware news feed function for its mobile users.  Memory management
<b>2020 News portal</b>	Abkash khatua	It makes use of various technologies to get required crime oriented information more quick, easily,colorfully and attractively	Data gathering, Data Analysis	It provides a wonderful user interface that attracts more and more users.  We don't manage news reporting system .

2021 <b>Android online news application</b>	Rumpa madhu halder	It is a very long process for someone is looking for some specific sector news ,then it got tough for them to get the news directly.	AP news , Feed, Flip board, Podcast addict, Reddit	Online portals are fast. It don't have update for database.
--	--------------------	--	--	---

## **2.2.REFERENCES:**

- [1]. Z. Abbassi, V. S. Mirrokni, and M. Thakur."Diversity maximization under matroid constraints" In ACM KDD, 2013.
- [2]. G. Adomavicius and Y. Kwon. "Improving aggregate recommendation diversity using ranking-based techniques" IEEE TKD 24(5):896911,2012.
- [3]. J. Bao, M. F. Mokbel, and C.-Y. Chow. "GeoFeed: A location-aware news feed system" In IEEE ICDE, 2012
- [4]. W. Xu, C.-Y. Chow, M. L. Yiu, Q. Li, and C. K. Poon. "MobiFeed: A location-aware news feed system for mobile users" In ACM SIGSPATIAL GIS, 2012.
- [5]. J. Bao, M. F. Mokbel, and C.-Y. Chow. GeoFeed: A location-aware newsfeed system. In IEEE ICDE, 2012.
- [6]. S. Vargas and P. Castells."Rank and relevance in novelty and diversity metrics for recommender systems" In ACM RecSys, 2011.
- [7]. H. Jeung, M. L. Yiu, X. Zhou, and C. S. Jensen. "Path prediction and predictive range querying in road network databases" VLDB Journal, 19(4):585602, 2010.
- [8]. C.-Y. Chow, J. Bao, and M. F. Mokbel. Towards location-based social networking services. In ACM SIGSPATIAL LBSN, 2010.
- [9]. B. Carterette and P. Chandar. "Probabilistic models of novel document rankings for faceted topic retrieval" In ACM CIKM, 2009.
- [10]. L. Wang, Y. Jia, and W. Han. Instant message clustering based on extended vector space model. In ISICA, 2007.
- [11]. F. Radlinski and S. T. Dumais. "Improving personalized web search using result diversification" In ACM SIGIR, 2006.
- [12]. F. Radlinski and S. T. Dumais. Improving personalized web search using result diversification. In ACM SIGIR, 2006.
- [13]. C.-N. Ziegler, S. M. McNee, J. A. Konstan, and G. Lausen. "Improving recommendation lists through topic diversification" In WWW, 2005.

**[14].** J. Carbonell and J. Goldstein. "The use of mmr, diversity-based reranking for reordering documents and producing summaries" In ACM SIGIR, 1998.

## 2.3. PROBLEM STATEMENT DEFINITION:

Jimmy is a busy business man who needs to read news on the go without any hassles while travelling because he considers carrying around a physical newspaper a nuisance to him and the people around him. Vijay needs to read news in such a way that he doesn't have to worry about ever buying physical newspapers or carrying them everywhere. Something which should fit in the palm of his hands, which he could carry everywhere, access from everywhere, something digital such as an Application hosted on the internet which could be accessed from any device that is connected to the Internet. Such as smartphones and computers.

### **who does the problem affect?**

Customers such as jimmy .

### **What are the boundaries of the problem?**

Geographic, workflow, daily life.

### **What is the issue?**

Physical newspapers are old fashioned in this digital era. They cost money to buy, can easily be damaged, limited amount of information, not flexible to modifications, poor quality. Fixing these in such a way can make physical newspapers become extinct and their use can be abolished. If the problem isn't solved, it would become a greater problem to integrate people to become digitally aware.

### **When does this issue occur?**

The issue occurs when the customer wants to read the news.

### **Where is the issue occurring?**

The issue occurs when people try to access information i.e during the usage.

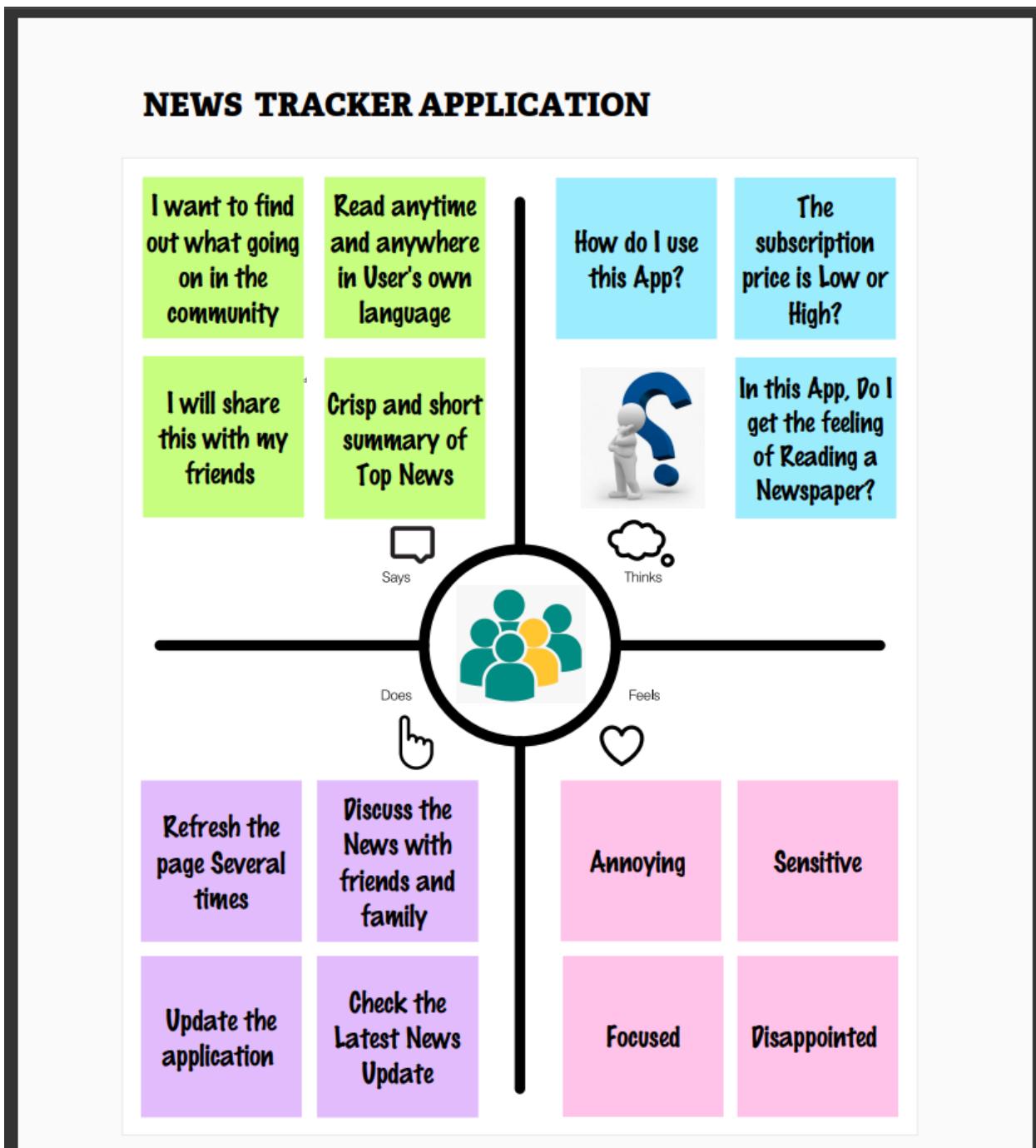
Accessing information is flawed through the usage of physical newspapers in the digital era.

### **Why is it important that we fix this problem?**

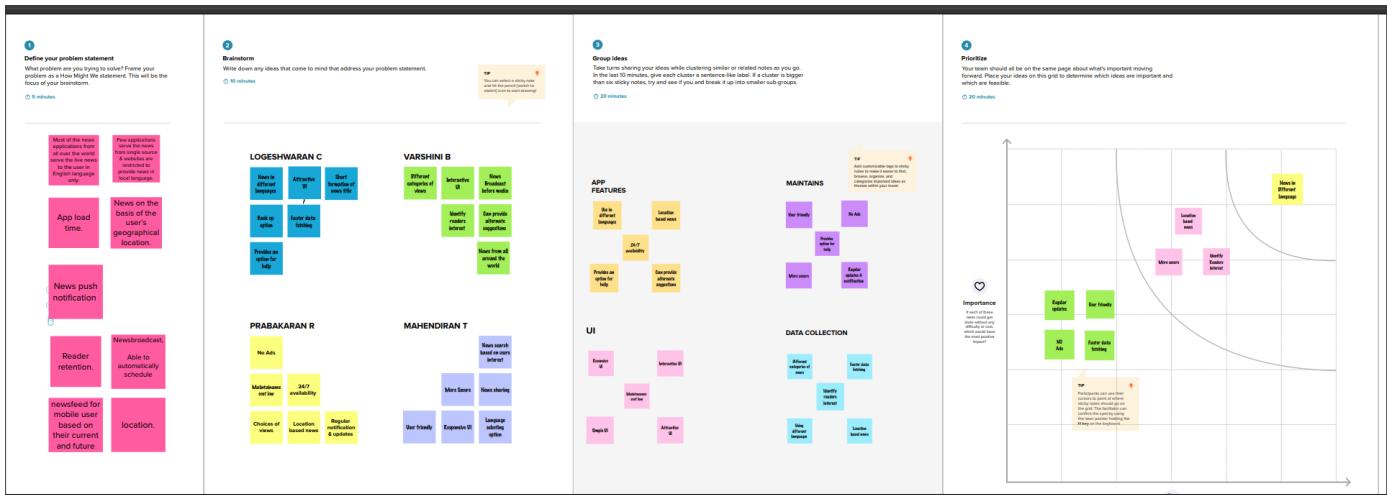
It is important to fix this problem because since everything in our day to day life is becoming/being converted to digital from a physical/analog entity. It is necessary to access information from anywhere. It reduces the hassles of getting knowledge and information that has happened before and will promote everyone to know better about the current events that's happening and will motivate everyone to gain a profound knowledge in the field of their interests. It wouldn't look weird.

**3. IDEATION  
&  
PROPOSED SOLUTION**

### 3.1 Empathy Map Canvas :



### **3.2 Ideation & Brainstorming :**



### **3.3 Proposed Solution :**

## PROBLEM STATEMENT

In today's fast life it is difficult for users to get news at their move. News push notification, news broadcast, Able to automatically schedule newsfeed for mobile user based on their current and future location. Most of the news app serve news to the user in English language only and in single source.

## **IDEA/SOLUTION DESCRIPTION**

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. It helps the user to read the news in their own national or international language. we create a scalable, responsive and user-friendly news tracker application.

## **NOVELTY / UNIQUENESS**

The online news can be browsed on the basis of categorirs, city, countries, interest, etc..., from anywhere of anytime using a mobile device. A section for comments or discussions should be there below every post. Readers can share their opininos on a specific topic. Engage with each other and discuss things, etc... .

## **SOCIAL IMPACT / CUSTOMER SATISFACTION**

The world is going to be more digital, and digitization is the need of every business, every Industry. Online News Sites play a vital role in educating and informing Mass with latest updates, current happenings around the globe. People don't have much more time to read the Printed newspaper with yesterday's happening, so they always follow web portals or Electronic Media for getting latest News ,With Electronic media, there is a certain problem like Costing of On Air is higher, and everyone is not always in front of Television set, but they can access their mobile, social sites, Whatsapp, Facebook etc. so they will definitely update them self very fast and less time-consuming methods. Online News Website and social media can give the fastest news Updates. Now every publisher has to build their web identity.

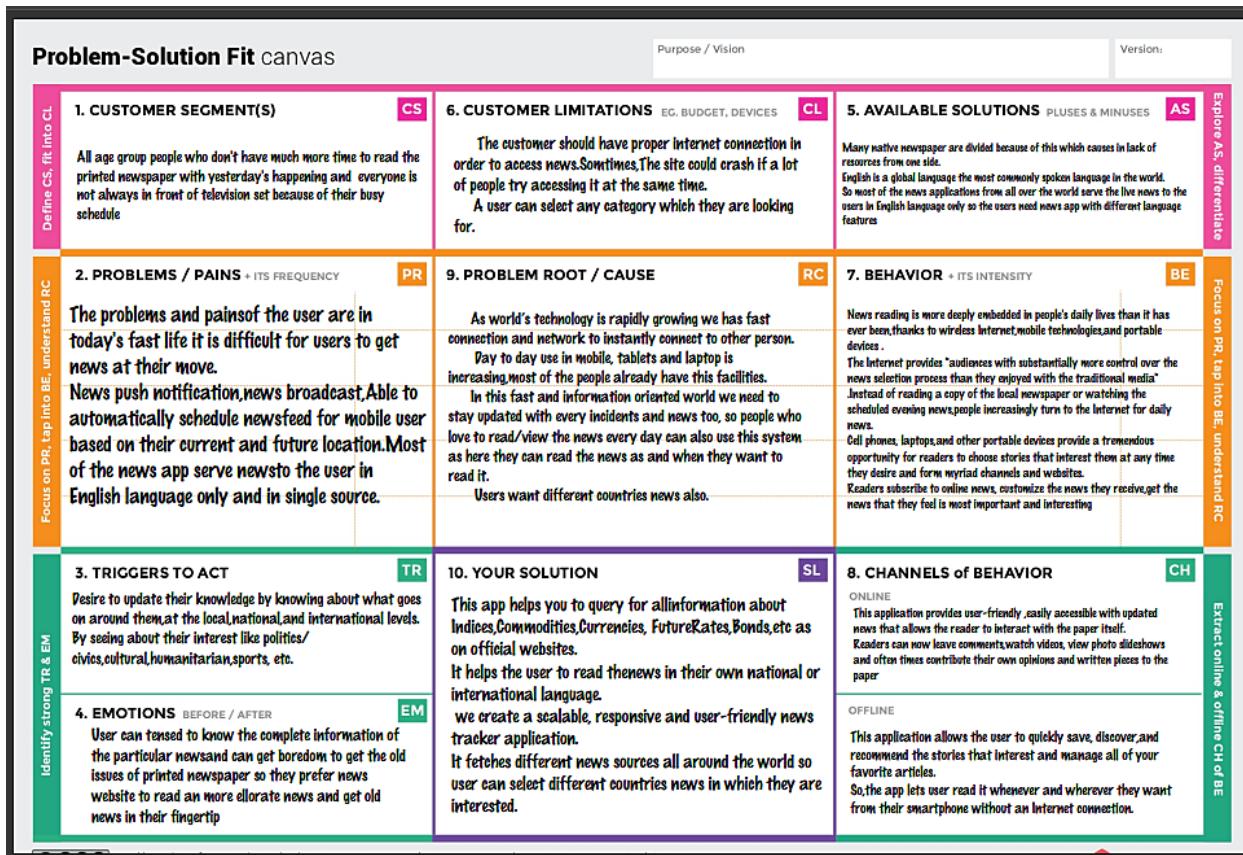
## **BUSINESS MODEL**

Showing third-party advertisements on the platform are the most efficient way of making money from a news website or app. The monetization option works successfully if you have a large audience base. Share the latest news at the earliest, trusted content. Some additional benefits with the subscription. You can also allow other writers, journalists, and brands to share content with you in the form of news , article, columns, etc... For publishing this content, you can charge some amount.

## **SCALABILITY OF SOLUTION**

Our solution that we provide based on cloud application like news tracker application scaling up makes hardware stronger; scaling out adds additional nodes. It requires more data storage capacity or processing power, you want a system that can scale easily and quickly.

## 3.4 Problem Solution fit :



## **4. REQUIREMENT ANALYSIS**

## 4.1 Functional requirement :

Following are the functional requirements of the proposed solution

FR No.	Functional Requirement (Epic)	Sub Requirement (Story / SubTask)
FR-1	User Registration	Registration through Form Registration through Gmail Registration through Linked IN
FR-2	User Confirmation	Confirmation via Email Confirmation via OTP
FR-3	Use Login	Login through application directly by entering user name & password Login through Gmail
FR-4	Uploading item	Uploading function can be done by user who has registered on the application. it is a news item is determined & edited by the administrator or editor & then it is displayed on the home page. Registered user can also add comment on the other news as well.

FR -5	Search Topic	Search function does not require any authentication from its user so any user can perform this function. If a user searcher for a news item then the news will be displayed on screen . If it related to the search topic.
FR - 6	Sharing	The application should have tools to share the news in social

		networks.
--	--	-----------

## 4.2 Non-Functional requirements :

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

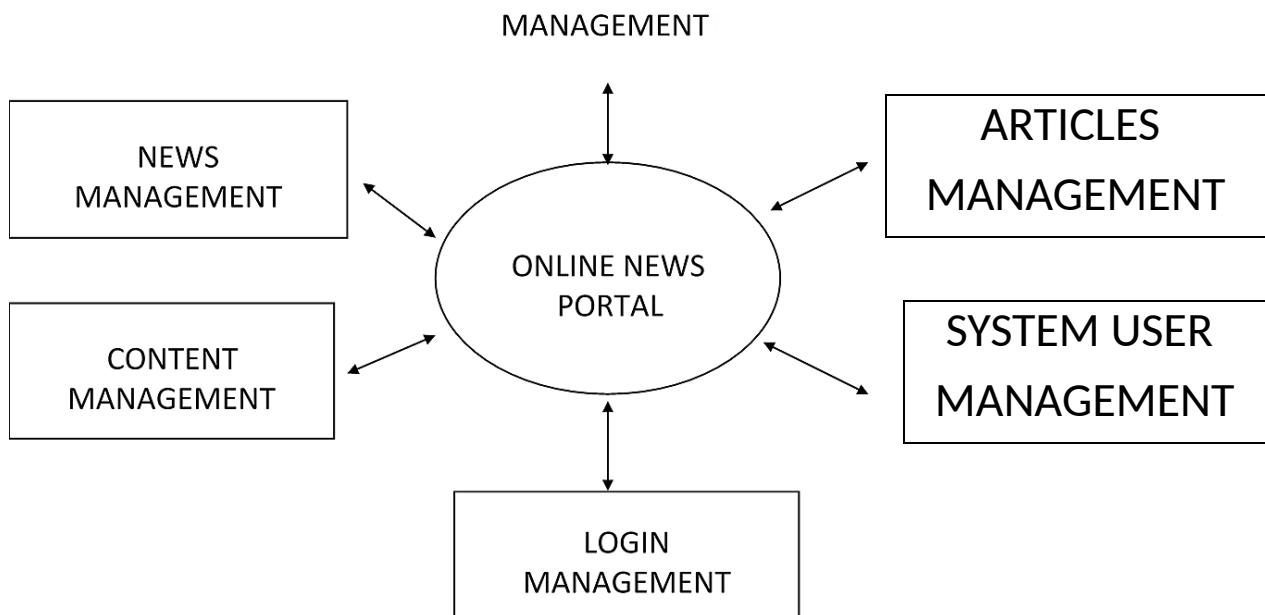
FR No.	Non-Functional Requirement	Description
NFR-1	Usability	Checking that the system is easy to handle & navigate in the most expected way with no delays. In that case the system reacts accordingly & transverse quickly between its states.
NFR-2	Security	All the system data must be backed up every day & the backup copies stored in another server at different location for disaster recovery.so no data loss occurs .user can only login then they use the correct email & password. Any eligible person can become a content writer only after allowed by the administrator.
NFR-3	Reliability	The system shall have a minimum uptime of 99% excluding time pre-schedule for maintenance for upgrade. We assure that the problem is completely authenticated in order to enhance security & corruptions of data base as the application. The person is given as an access only if he/she has a valid email & password both for user & content writer.
NFR-4	Performance	The system respond time must be less than 30 sec for the user interface. or else the system will show TIMED OUT. Application gets the authenticated content writers & other sources which are available in other sources portion

<b>FR No.</b>	<b>Non-Functional Requirement</b>	<b>Description</b>
NFR-5	Availability	Checking that the system always has something to function & always popup error messages component failure. In that case the error messages appear when something goes wrong so to prevail availability problems .The web application is available 24/7 at any where
NFR-6	Scalability	Our solution that we provide based on cloud application like news tracker application scaling up make hardware strong : scaling out adds additional node . It refers more data storage capacity or processing power , u want a system that can scale easily & quickly.

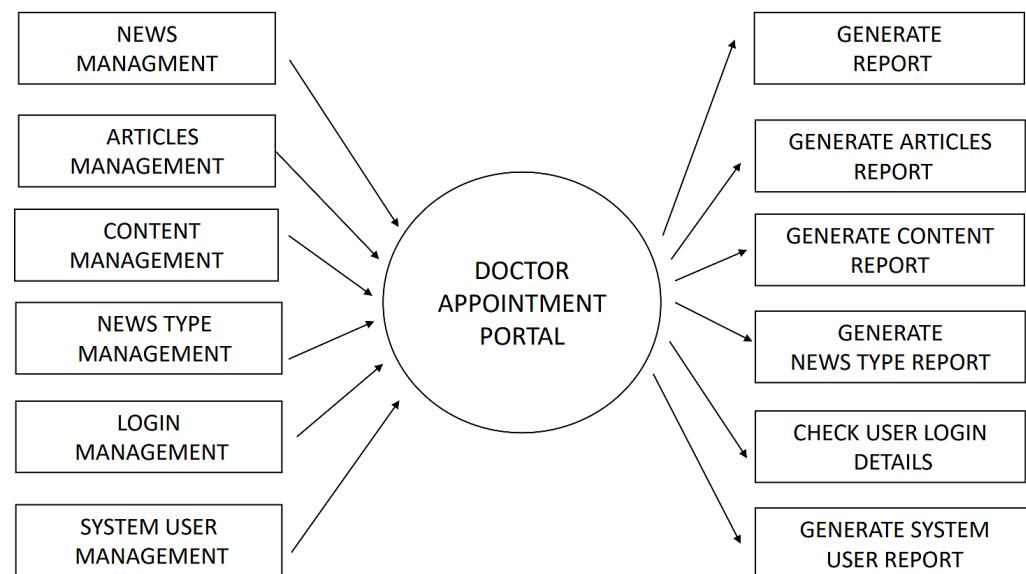
## **5. PROJECT DESIGN**

## 5.1 Data Flow Diagrams :

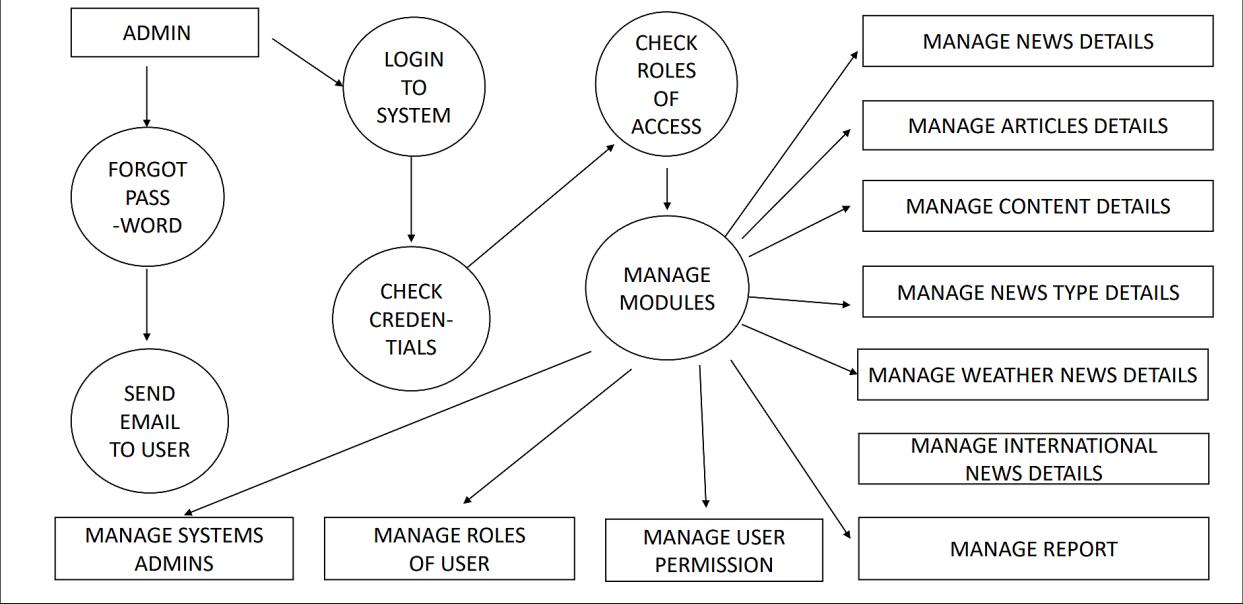
### ZERO LEVEL DFD - ONLINE NEWS PORTAL



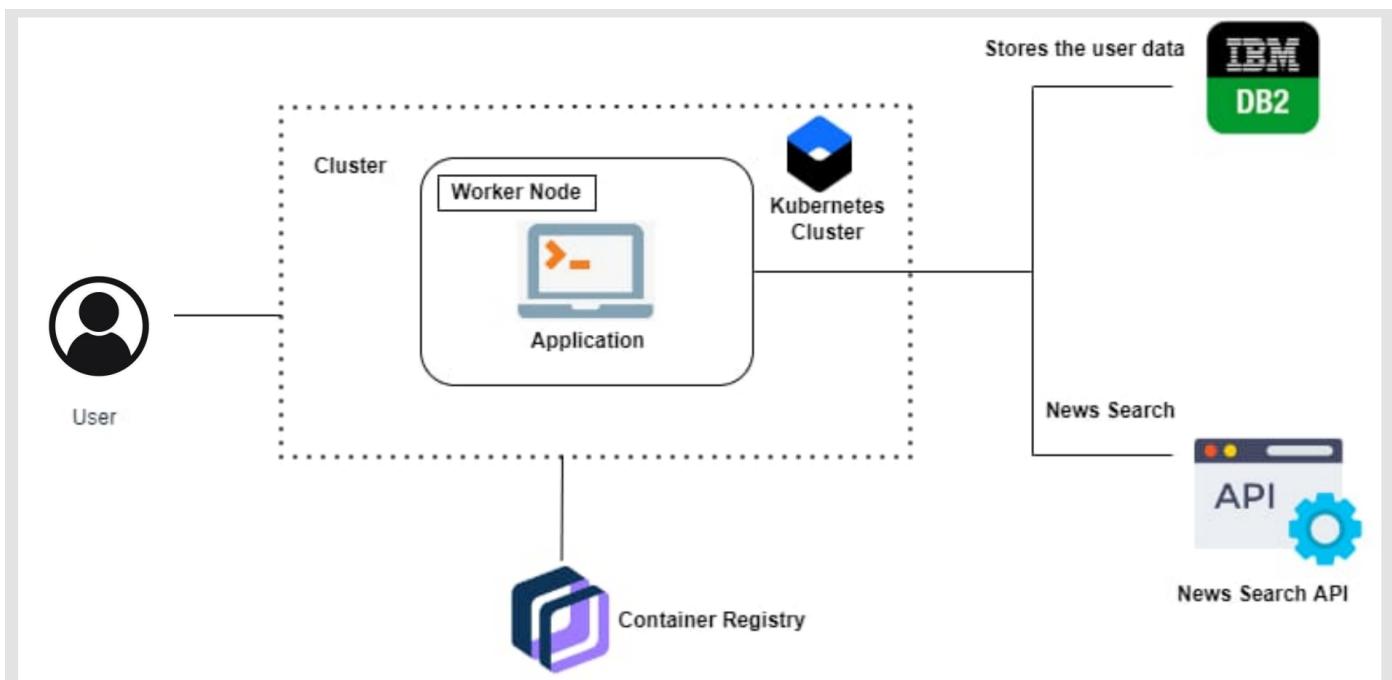
### FIRST LEVEL DFD - ONLINE NEWS PORTAL



## SECOND LEVEL DFD - ONLINE NEWS PORTAL



## 5.2 Solution & Technical Architecture:



## 5.3 User Stories:

User Stories						
Use the below template to list all the user stories for the product.						
User Type	Functional Requirement (Epic)	User Story Number	User Story / Task	Acceptance criteria Priority Release	Priority	Release
Customer (Mobile user)	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
		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
		USN-3	As a user, I can register for the application through Facebook	I can register & access the dashboard with Facebook Login	Low	Sprint-2
		USN-4	As a user, I can register for the application through Gmail		Medium	Sprint-1
	Login	USN-5	As a user, I can log into the application by entering email & password		High	Sprint-1
	Dashboard	USN-6	As a user, I can access the news content in the home page or dashboard	I can access the dashboard	High	Sprint-2
Customer (Web user)		USN-7	As a user, I can send a feedback to the admin	I can question the admin	High	Sprint-1
Customer Care Executive		USN-8	As a user care executive, I will provide clarification to the user instantly	I can clarify question that user ask	Medium	Sprint-3
Administrator		USN-9	As an admin, I can add & edit the news content		High	Sprint-4
		USN-10	As an admin, I can access the user to upload news		Medium	Sprint-4
		USN-11	As an admin, I will update the application with new features		High	Sprint-4
		USN-12	As an admin, I will maintain the application process		High	Sprint-4

# **6. PROJECT PLANNING & SCHEDULING**

## 6.1 Sprint Planning & Estimation :

### Product Backlog, Sprint Schedule, and Estimation (4 Marks)

Use the below template to create product backlog and sprint schedule

Sprint	Functional Requirement (Epic)	User Story Number	User Story / Task	Story Points	Priority	Team Members
Sprint-1	User Panel	USN-1	As a user, I can register for application using either my phone number or email address & then login to the application. If the user have an account they can login to application & choose the language in which they will be likely to read the news & check the news as per his/her choice.	20	High	LOGESHWARAN C VARSHINI B
Sprint-2	Admin Panel	USN-2	As an admin, I'm able to oversee all application activity & can accept, alter, or reject user uploaded content & I can also restrict the users too.	20	High	PRABAKARAN R MAHENDIRAN T
Sprint-3	News Feed Panel	USN-3	All the news by the admin is updated into this panel, this news as many sections & each section is specific to a particular domain. Admin can update the news into this panel & thus readers can read the news from the feed	20	High	LOGESHWARAN C PRABAKARAN R MAHENDIRAN T

Sprint-4	Final Delivery	USN-4	All the data's are stored in the IBM data base & deployment & submit the application.	20	High	LOGESHWARAN C VARSHINI B MAHENDIRAN T
----------	----------------	-------	---	----	------	---

## 6.2 Sprint Delivery Schedule :

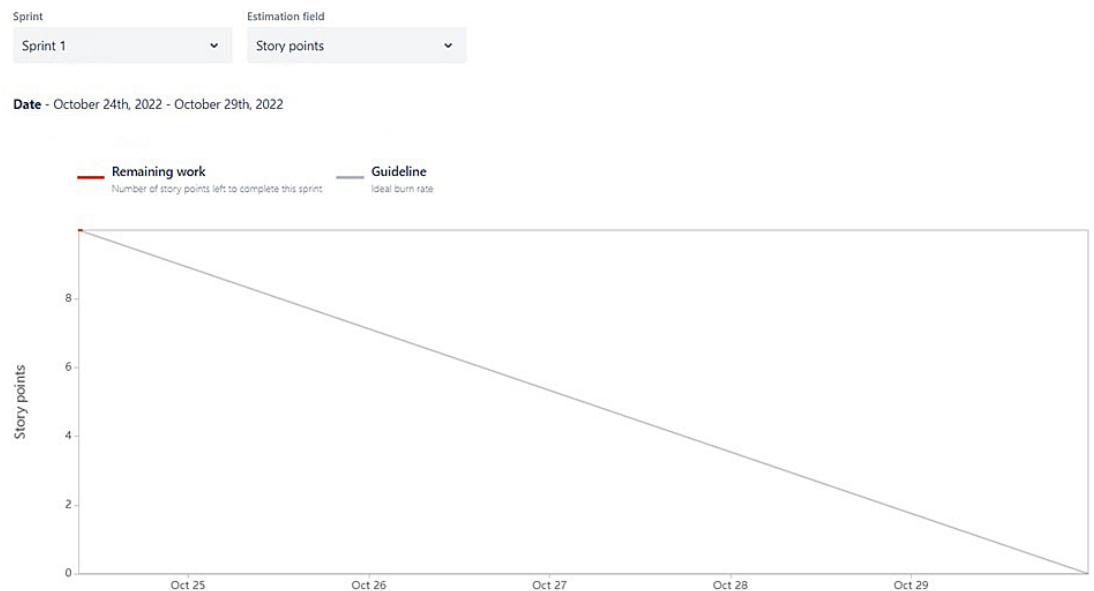
Sprint	Total Story Points	Duration	Sprint Start Date	Sprint End Date (Planned)	Story Points Completed (as on Planned End Date)	Sprint Release Date (Actual)
Sprint-1	20	6 Days	24 Oct 2022	29 Oct 2022		29 Oct 2022
Sprint-2	20	6 Days	31 Oct 2022	05 Nov 2022		05 Nov 2022
Sprint-3	20	6 Days	07 Nov 2022	12 Nov 2022		12 Nov 2022
Sprint-4	20	6 Days	14 Nov 2022	19 Nov 2022		19 Nov 2022

## 6.3 Reports from JIRA:

### Burndown chart sprint 1

#### Sprint burndown chart

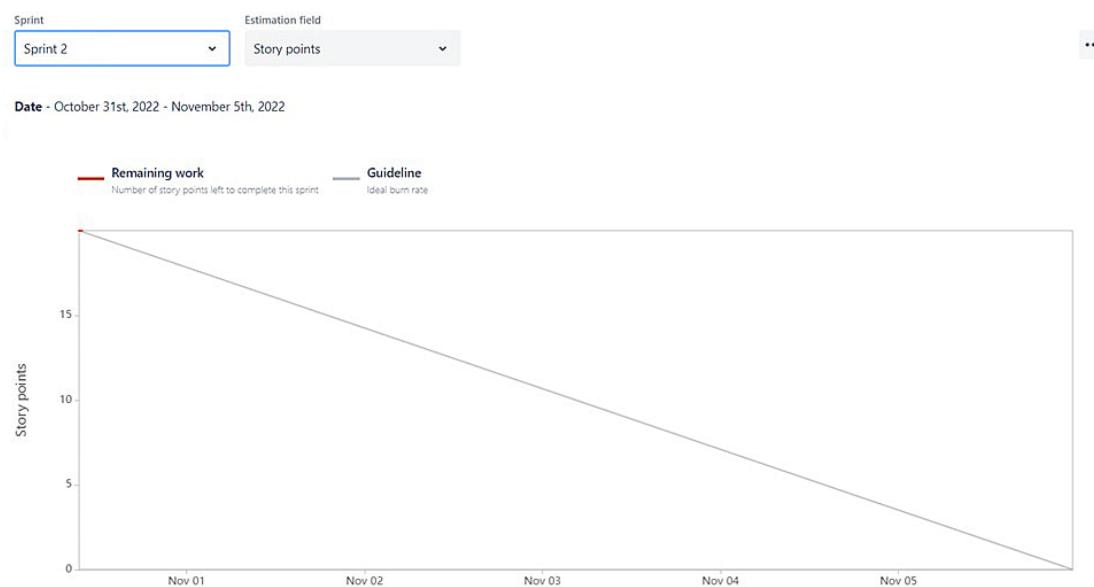
[How to read this report](#)



### Burndown chart sprint 2

#### Sprint burndown chart

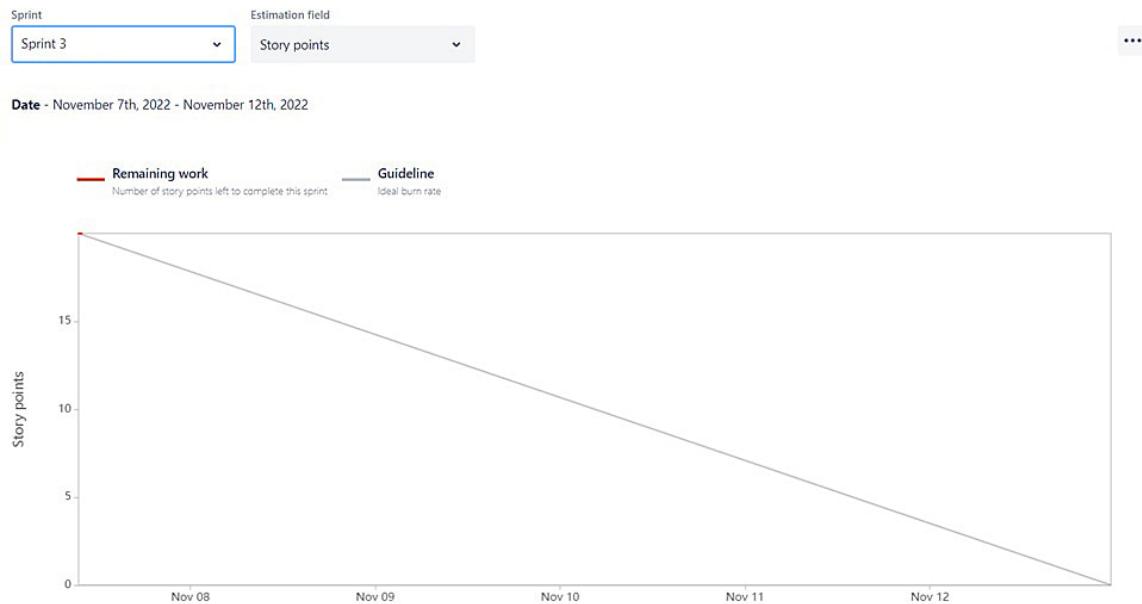
[How to read this report](#)



## Burndown chart sprint 3

### Sprint burndown chart

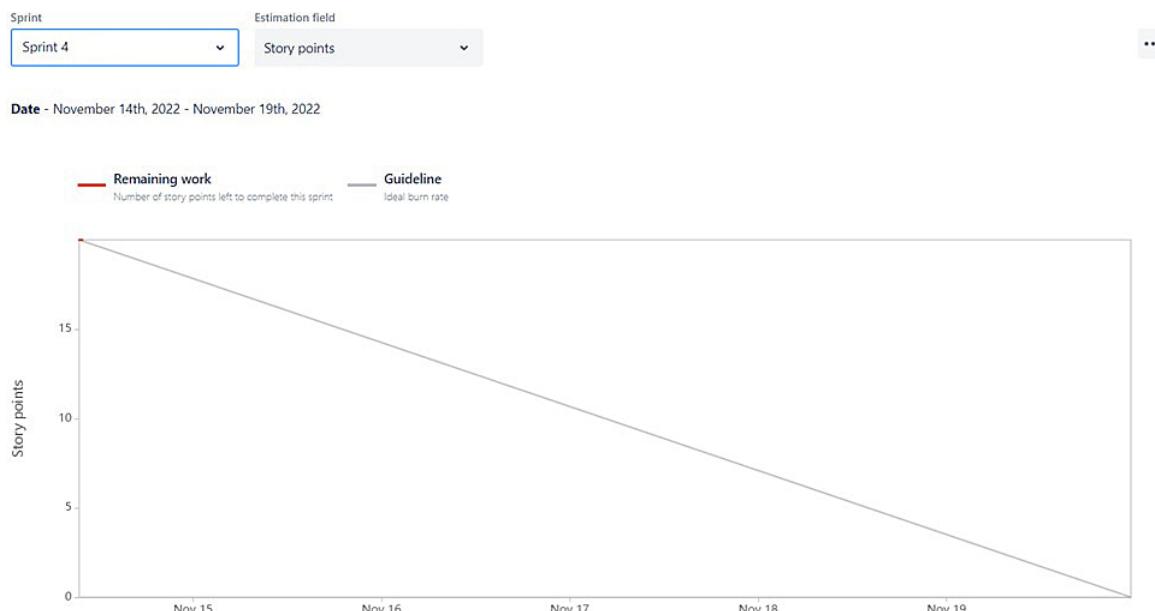
[How to read this report](#)



## Burndown chart sprint 4

### Sprint burndown chart

[How to read this report](#)



## **7. CODING & SOLUTIONING**

**(Explain the features added in the project along with code)**

## 7.1 Feature 1:

```
NEWS Tracker Application > app.py > index
 1  from flask import Flask,render_template,session,redirect,url_for,request
 2  import ibm_db
 3  import re
 4  import hashlib
 5  from newsapi import NewsApiClient
 6  import os
 7  app = Flask(__name__)
 8  app.secret_key = os.urandom(16)
 9
10
11 try:
12     conn = ibm_db.connect(os.getenv('KEY'),'', '')
13 except Exception as err:
14     print("Exception occurs->", err)
15
16
17
18 @app.route('/')
19 def index():
20     if not session:
21         return render_template('login.html')
22     return redirect(url_for('home'))
23
24 @app.route('/login')
25 def login():
26     if not session or not session['login_status']:
27         return render_template('login.html')
28     return redirect(url_for('home'))
29
30
31 @app.route('/registration')
32 def registration():
33     return render_template('registration.html')
34
35 @app.route('/home')
36 def home():
37     api_key = os.getenv('NEWS_API')
38
39     newsapi = NewsApiClient(api_key=api_key)
40
41     top_headlines = newsapi.get_top_headlines(sources = "bbc-news")
42     all_articles = newsapi.get_everything(sources = "bbc-news")
43
44     t_articles = top_headlines['articles']
45     a_articles = all_articles['articles']
46
47     news = []
48     desc = []
49     img = []
50     p_date = []
51     url = []
52
53     for i in range (len(t_articles)):
54         main_article = t_articles[i]
55
56         news.append(main_article['title'])
57         desc.append(main_article['description'])
58         img.append(main_article['urlToImage'])
59         p_date.append(main_article['publishedAt'])
```

```

61     news_all = []
62     desc_all = []
63     img_all = []
64     p_date_all = []
65     url_all = []
66
67     for j in range(len(a_articles)):
68         main_all_articles = a_articles[j]
69
70         news_all.append(main_all_articles['title'])
71         desc_all.append(main_all_articles['description'])
72         img_all.append(main_all_articles['urlToImage'])
73         p_date_all.append(main_all_articles['publishedAt'])
74         url_all.append(main_all_articles['url'])
75
76     all = zip(news_all,desc_all,img_all,p_date_all,url_all)
77
78
79     return render_template('home.html',contents=contents,all = all)
80
81 @app.route('/do_register',methods=['GET','POST'])
82 def do_register():
83     if request.method == 'POST':
84         email = request.form['email']
85         #-----
86         # password hashing
87         password = request.form['pwd']
88         cnf_password = request.form['pwd-repeat']
89         if password != cnf_password:

```

```

88             cnf_password = request.form['pwd-repeat']
89             if password != cnf_password:
90                 msg = "Password Doesn't Match"
91                 return msg
92
93             password = bytes(password,'utf-8')
94             password = hashlib.sha256(password).hexdigest()
95
96             # case 1: check if user does exists already
97             sql = "SELECT * FROM users WHERE username =?"
98             stmt = ibm_db.prepare(conn, sql)
99             ibm_db.bind_param(stmt,1,email)
100            ibm_db.execute(stmt)
101            acc = ibm_db.fetch_assoc(stmt)
102            if acc:
103                msg = "Account already Exists, Please login "
104                return msg
105
106            # case 2: validate the input if it matches the required pattern
107            if not re.match(r"^\S+@\S+\.\S+$", email):
108                msg = "Please Enter Valid Email Address "
109                return msg
110
111            insert_sql = "INSERT INTO users VALUES (?,?)"
112            prep_stmt = ibm_db.prepare(conn, insert_sql)
113            ibm_db.bind_param(prep_stmt, 1, email)
114            ibm_db.bind_param(prep_stmt, 2, password)
115            ibm_db.execute(prep_stmt)
116

```

## 7.2 Feature 2 :

```
NEWS Tracker Application > templates > home.html > html > body > nav > ul.menu > li.nav-items
1  <!DOCTYPE html>
2  <html lang="en">
3  <head>
4      <meta charset="UTF-8">
5      <meta http-equiv="X-UA-Compatible" content="IE=edge">
6      <meta name="viewport" content="width=device-width, initial-scale=1.0">
7      <link rel="stylesheet" href="{{url_for('static', filename='css/style.css')}}">
8      <link rel="preconnect" href="https://fonts.googleapis.com">
9      <link rel="icon" href="{{url_for('static', filename='img/logo.png')}}">
10     <link href="https://fonts.googleapis.com/icon?family=Material+Icons" rel="stylesheet">
11     <title>News Website</title>
12 </head>
13 <body>
14     <nav>
15         <a href="/" class="logo"><h2>News Tracker Application</h2></a>
16         <ul class="menu">
17             <li class="nav-items"><a href="/">Home</a></li>
18             <li class="nav-items"><a href="#news">News</a></li>
19             <li class="nav-items"><a href="/>Contact</a></li>
20             <li class="nav-items"><a href="#">Logout</a></li>
21         </ul>
22     </nav>
23 <section id="main">
24     <div class="banner">
25     </div>
26     <div class = "head">TODAY'S TOP HEADLINES</div>
27     {% for title,desc,img,p_date,url in contents %}
28     <div class="wrapper" id="news">
29         <div class="item"><a href="{{url}}><a href="{{url}}>
31             <div class="content"><a href="{{url}}><h2>{{title}}</h2></a></div>
32             <div class="content"><a href="{{url}}><p>{{desc}}</p></a></div>
33             <div class="content"><p style="color: #rgb(77, 255, 0);>Publish Date:- </p><p>{{p_date}}</p></div>
34         </div>
35     </div>
36     {% endfor %}
37
38     <div class = "head">TODAY'S NEWS</div>
39     {% for title,desc,img,p_date,url in all %}
40     <div class="wrapper">
41         <div class="item"><a href="{{url}}>
43             <div class="content"><a href="{{url}}><h2>{{title}}</h2></a></div>
44             <div class="content"><p>{{desc}}</p></div>
45             <div class="content"><p style="color: #rgb(77, 255, 0);>Publish Date:- </p><p> {{p_date}} </p></div>
46         </div>
47     </div>
48     {% endfor %}
49 </section>
50 </body>
51 </html>
```

```
NEWS Tracker Application > templates > login.html > html > body > form > div.mainContainer > p.register > a
1   <!DOCTYPE html>
2   <html lang="en">
3     <head>
4       <meta charset="UTF-8">
5       <meta http-equiv="X-UA-Compatible" content="IE=edge">
6       <meta name="viewport" content="width=device-width, initial-scale=1.0">
7       <title>Login page in HTML</title>
8     </head>
9     <body>
10       <h1>LOGIN</h1>
11       <form action="{{url_for('do_login')}}" method="post">
12         <!-- Headings for the form -->
13         <div class="headingsContainer">
14           <h3>Sign in</h3>
15           <p>Sign in with your username and password</p>
16         </div>
17
18         <!-- Main container for all inputs -->
19         <div class="mainContainer">
20           <!-- Username -->
21           <label for="username">Your username</label>
22           <input type="text" placeholder="Enter Username" name="username" required>
23
24           <br><br>
25
26           <!-- Password -->
27           <label for="pswrd">Your password</label>
28           <input type="password" placeholder="Enter Password" name="pswrd" required>
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
```

```
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
```

```
NEWS Tracker Application > templates > registration.html > form
1   <form action="{{url_for('do_register')}}" method="post">
2     <div>
3       <h1>Register</h1>
4       <p>Kindly fill in this form to register.</p>
5       <label for="email"><b>Email</b></label>
6       <input type="text" placeholder="Enter Email" name="email" id="email" required/>
7       <label for="pwd"><b>Password</b></label>
8       <input type="password" placeholder="Enter Password" name="pwd" id="pwd" required/>
9       <label for="pwd-repeat"><b>Repeat Password</b></label>
10      <input
11        type="password"
12        placeholder="Repeat Password"
13        name="pwd-repeat"
14        id="pwd-repeat"
15        required
16      />
17
18      <button type="submit">Register</button>
19    </div>
20    <!-- wrapping the text inside the p tag with a tag for routing to login page url-->
21    <!-- the # must ideally be replaced by the login page url -->
22    <div>
23      <p>Already have an account? <a href="#">Log in</a>.</p>
24    </div>
25  </form>
26
```

## **8. TESTING**

## 8.1 Test Cases :

Test case	Feature	Components	Test scenario	Expected result	Actual result	Status	Comment	Bug
Sign in	Functional	Login page	Identify whether User can see sign in option	Its visible	yes	Pass	Successful	No
Sign up	Functional	Login page	Identify whether User can see sign up option	Its visible	yes	Pass	Successful	No
Forgot password	Functional	Login page	Identify whether user can see forgot password	Its available	yes	Pass	Successful	No
Get news	Functional	Home page	Identify whether user can able to see news	Its available	yes	Pass	Successful	No
Various news	Functional	Dashboard	Identify whether the user can able to see various news	Its available	yes	Pass	Successful	No

## 8.2 User Acceptance Testing :

### Defect Analysis:

This report shows the number of resolved or closed bug at each severity level, & how they were resolved .

Resolution	Severity 1	Severity 2	Severity 3	Severity 4	Sub total
By Design	11	5	2	3	21
Duplicate	1	0	3	0	4
External	3	2	0	1	6
Fixed	15	5	4	21	45
Not reproduced	0	0	1	0	1
Skipped	0	0	1	1	2
Won't Fix	0	5	2	1	8
Totals	30	17	13	27	87

### TEST CASE ANALYSIS:

This report shows the number of test case that have passed , failed & untested.

Section	Total cases	Not Tested	Fail	Pass
Print Engine	8	0	0	8
Client Application	41	0	0	42
Security	4	0	0	4
Outsource Shipping	5	0	0	5
Exception Reporting	3	0	0	9
Final Report Output	4	0	0	4
Version Control	2	0	0	2

## **9. RESULTS**

## 9.1 Performance Metrics :

### Registration Page

#### Register

Kindly fill in this form to register.

Email	Enter Email	Password	Enter Password	Repeat Password	Repeat Password	Register
-------	-------------	----------	----------------	-----------------	-----------------	----------

Already have an account? [Log in](#).

### Login Page :

# LOGIN

#### Sign in

Sign in with your username and password

Your username

Your password

Remember me

[Forgot Password?](#)

Not a member? [Register here!](#)

## Dash Board :



## NEWS Page :



# **10. ADVANTAGES & DISADVANTAGES**

## **ADVANTAGES:**

- The user is updated with everything going around his city.
- Easy to use.
- The news can be viewed only by registered users.
- This system for Android users.
- News Broadcast before media.
- Useful in any area need a recent updates about their surroundings.
- More secure.
- Anyone can broadcast and receive news.

## **DISADVANTAGES :**

- It requires an active internet connection.
- These applications Fetch the news from single source.
- Few applications publish articles only from one particular country & the language supported is only English.
- Foreigners who have difficulty reading English or who cannot understand English will not be aware of the current situations in the local country or around the world.
- It is difficult for the users to find required news updates in different languages from different sources in a single web application.

# **11. CONCLUSION**

## **CONCLUSION:**

We did research in different news applications and came up with a solution. It is an open source application, anyone from all over the world can access our application without any subscription charges or any user details. The amount of time and complexity required for manually searching and reading will be reduced and also security of newsfeed is maintained. The news reports can be obtained regularly and also monitored by the Admin. Here correctness and validity of broadcasted news information is verified. The system provides efficiency of News feed. The user will get updated about news that happened in the area which is of their interest. Thus, it helps to deliver fastest news before media.

## **12. FUTURE SCOPE**

## **FUTURE SCOPE :**

In future a database will be added to the application which will store the user information and stats can be known easily by adding a small counter. Only Admin can see the count about the number of users who accessed our application. Pagination can also be added to our application which will load the news in different pages. The application that is developed is a simple and user friendly. By using advanced CSS styles and different front-end technologies interface of the application can be developed more according to user requirements. 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 improved in a more efficient way on full articles. Data quality check needed.

## **13. APPENDIX**

## Source Code :

### login.html

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Login page in HTML</title>
</head>
<body>
    <h1>LOGIN</h1>
    <form action="{{url_for('do_login')}}" method="post">
        <!-- Headings for the form -->
        <div class="headingsContainer">
            <h3>Sign in</h3>
            <p>Sign in with your username and password</p>
        </div>
        <!-- Main container for all inputs -->
        <div class="mainContainer">
            <!-- Username -->
            <label for="username">Your username</label>
            <input type="text" placeholder="Enter Username" name="username" required>
            <br><br>
            <!-- Password -->
            <label for="pswrd">Your password</label>
            <input type="password" placeholder="Enter Password" name="pswrd" required>
            <!-- sub container for the checkbox and forgot password link -->
            <div class="subcontainer">
                <label>
```

```

<input type="checkbox" checked="checked" name="remember"> Remember
me
</label>
<p class="forgotpsd"> <a href="#">Forgot Password?</a></p>
</div>

<!-- Submit button -->
<button type="submit">Login</button>

<!-- Sign up link -->
<p class="register">Not a member? <a href="/registration">Register
here!</a></p>

</div>

</form>
</body>
</html>

```

## app.py

```

from flask import Flask, render_template, session, redirect, url_for, request
import ibm_db
import re
import hashlib
from newsapi import NewsApiClient
import os
app = Flask(__name__)
app.secret_key = os.urandom(16)
try:
    conn = ibm_db.connect(os.getenv('KEY'), "", "")
except Exception as err:
    print("Exception occurs->", err)
@app.route('/')

```

```
def index():
    if not session:
        return render_template('login.html')
    return redirect(url_for('home'))

@app.route('/login')
def login():
    if not session or not session['login_status']:
        return render_template('login.html')
    return redirect(url_for('home'))

@app.route('/registration')
def registration():
    return render_template('registration.html')

@app.route('/home')
def home():
    api_key = os.getenv('NEWS_API')

    newsapi = NewsApiClient(api_key=api_key)

    top_headlines = newsapi.get_top_headlines(sources = "bbc-news")
    all_articles = newsapi.get_everything(sources = "bbc-news")

    t_articles = top_headlines['articles']
    a_articles = all_articles['articles']

    news = []
    desc = []
    img = []
    p_date = []
    url = []

    for i in range (len(t_articles)):
        main_article = t_articles[i]
        news.append(main_article['title'])
        desc.append(main_article['description'])
```

```

        img.append(main_article['urlToImage'])
        p_date.append(main_article['publishedAt'])
        url.append(main_article['url'])

    contents = zip( news,desc,img,p_date,url)

    news_all = []
    desc_all = []
    img_all = []
    p_date_all = []
    url_all = []

    for j in range(len(a_articles)):
        main_all_articles = a_articles[j]

        news_all.append(main_all_articles['title'])
        desc_all.append(main_all_articles['description'])
        img_all.append(main_all_articles['urlToImage'])
        p_date_all.append(main_all_articles['publishedAt'])
        url_all.append(main_all_articles['url'])

    all = zip( news_all,desc_all,img_all,p_date_all,url_all)

    return render_template('home.html',contents=contents,all = all)

@app.route('/do_register',methods=['GET','POST'])
def do_register():
    if request.method == 'POST':
        email = request.form['email']
        #-----
        # password hashing
        password = request.form['pwd']
        cnf_password = request.form['pwd-repeat']
        if password != cnf_password:
            msg = "Password Doesn't Match"

```

```
return msg

password = bytes(password,'utf-8')
password = hashlib.sha256(password).hexdigest()

# case 1: check if user does exists already
sql = "SELECT * FROM users WHERE username =?"
stmt = ibm_db.prepare(conn, sql)
ibm_db.bind_param(stmt,1,email)
ibm_db.execute(stmt)
acc = ibm_db.fetch_assoc(stmt)
if acc:
    msg = "Account already Exists, Please login "
    return msg

# case 2: validate the input if it matches the required pattern
if not re.match(r"^\S+@\S+\.\S+$", email):
    msg = "Please Enter Valid Email Address "
    return msg

insert_sql = "INSERT INTO users VALUES (?,?)"
prep_stmt = ibm_db.prepare(conn, insert_sql)
ibm_db.bind_param(prep_stmt, 1, email)
ibm_db.bind_param(prep_stmt, 2, password)
ibm_db.execute(prep_stmt)

return redirect(url_for('login'))
return redirect(url_for('registration'))
@app.route('/do_login',methods=['GET','POST'])
def do_login():
    if request.method == 'POST':
        username = request.form['username']
        password = request.form['pswrd']
        # salt the password
        password = bytes(password,'utf-8')
        password = hashlib.sha256(password).hexdigest()
```

```

#query the db
sql = "SELECT * FROM users WHERE username =? AND password=?"
stmt = ibm_db.prepare(conn,sql)
ibm_db.bind_param(stmt,1,username)
ibm_db.bind_param(stmt,2,password)
ibm_db.execute(stmt)
acc = ibm_db.fetch_assoc(stmt)
if not acc:
    return "<h1> User Not exists</h1>"
if acc:
    session['login_status'] = True
    session['username'] = username
    session['user_id'] = username.split('@')[0]
    return redirect(url_for('home'))
return redirect('registration.html')
@app.route('/logout')
def logout():
    session.pop('login_status',None)
    session.pop('user_id',None)
    session.pop('username',None)
    return redirect(url_for('login'))
if __name__ == '__main__':
    app.run(debug=True)

```

## **GitHub & Project Demo Link:**

### **GitHub Link :**

<https://github.com/IBM-EPBL/IBM-Project-32003-1660207331>

### **Project Demo Link:**

<https://github.com/IBM-EPBL/IBM-Project-32003-1660207331/tree/main/Demo%20Link>