

# **IBM PROJECT FINAL DOCUMENT**

**TEAM ID:PNT2022TMID05148**

**Team leader: ARIHARAN S**

**Team Member :ARUN J**

**Team Member:ARUN KUMAR K**

**Team Member:ASEELA A**

# Project Report

## ➤ INTRODUCTION

Project Overview

Purpose

## ➤ LITERATURE SURVEY

Existing problem

References

Problem Statement Definition

## ➤ IDEATION & PROPOSED SOLUTION

Empathy Map Canvas

Ideation & Brainstorming

Proposed Solution

Problem Solution fit

## ➤ REQUIREMENT ANALYSIS

Functional requirement

Non-Functional requirements

## ➤ PROJECT DESIGN

Data Flow Diagrams

Solution & Technical Architecture

➤ **PROJECT PLANNING & SCHEDULING**

Sprint Planning & Estimation

Sprint Delivery Schedule

➤ **CODING & SOLUTIONING (Explain the features added in the project along with code)**

Feature 1

Feature 2

Database Schema (if Applicable)

➤ **ADVANTAGES & DISADVANTAGES**

➤ **CONCLUSION**

➤ **FUTURE SCOPE**

➤ **APPENDIX**

Source

code

Live page

URL

## **INTRODUCTION**

### **Project Overview**

NewsTracker is a fullstack web application which allows users to register along with their favourite topics, upon login the app displays the news based on the user's interest. The news displayed in the app is based on the Newscatcher API and Cricbuzz API from Rapid API site. A news-sharing app wants to help users

find relevant and important news easily every day and also provide explicitly news from that users locality/region which may of help to the user.

## **Purpose**

Enabling users to view news from anywhere at anytime. It also helps to reduce the time to get information about a specific topic. Also enables a person to get an updated news which may help Business people to make business related decisions quickly and correctly.

## **LITERATURE SURVEY**

### **Existing problem**

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. Sometimes may show irrelevant and updated news.

### **References**

An Approach to News Event Detection and Tracking Based on Stream of Online News,

Authors : Yajie Qi, Li Zhou, Huayou Si, Jian Wan, Ting Jin.

About the paper :

Once an event occurs, usually there are a large number of online news to be released. How to quickly and

accurately detect the hot events from the huge amount of online news is the focus and hotspot.

## Exploring Mobile News Reading Interactions for News App Personalisation

Authors: Marios Constantinides, John Dowell, David Johnson, Sylvain Malacria

About the paper :

As news is increasingly accessed on smartphones and tablets, the need for personalising news app interactions is apparent. We report a series of three studies addressing key issues in the development of adaptive news app interfaces.

## Android News App

Authors: Brijesh Joshi,  
Nehal Patel. About the  
paper:

As world's technology is rapidly growing, we have 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 these facilities.

Research on Topic Detection and Tracking for Online News Text Authors: Guixian Xu, Yueting Meng, Zhan Chen, Xiaoyu Qiu, Changzhi Wang, Haishen Yao.

About the paper :

With the rapid development of the Internet, the amount of data has grown exponentially.

On the one hand, the accumulation of big data provides the basic support for artificial intelligence.

On the other hand, in the face of such huge data information, how to extract the knowledge of interest from it has become a matter of general concern.

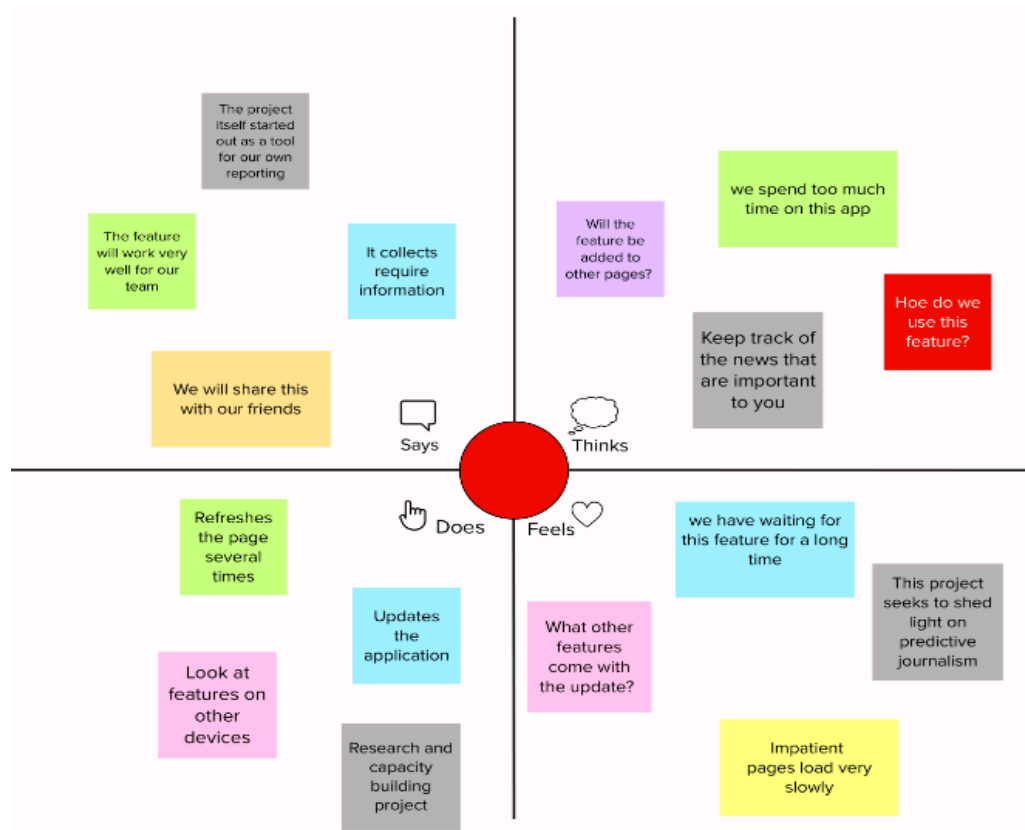
### **Problem Statement Definition**

Newspaper contains limited, non user/reader specific, Location specific news.

There are multiple news-sharing apps available which can be used by a single user and are often spammed with notifications. There is also a lot of unwanted news which gets shared.

So it may take a lot of time for the user to find the news he/she likes. A news-sharing app wants to help users find relevant and important news easily every day and also provide explicitly news from that user's locality/region which may be of help to the user.

## IDEATION & PROPOSED SOLUTION



## 3.1 Empathy Map Canvas

### Ideation & Brainstorming

Brainstorming provides a free and open environment that encourages everyone within a team to participate in the creative thinking process that leads to problem solving. Prioritizing volume over value, out-of-the-box ideas are welcome and built upon, and all participants are encouraged to collaborate, helping each other develop a rich amount of creative solutions.

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

The screenshot displays a digital workspace for brainstorming and idea prioritization, organized into four main vertical panels.

- Panel 1 (Left):** Titled "Brainstorm & idea prioritization", it includes a lightbulb icon and instructions: "Use this template in your own brainstorming sessions so your team can unleash their imagination and start shaping concepts even if you're not sitting in the same room." It also lists duration (10 minutes to prepare, 1 hour to collaborate) and a recommendation (3-8 people).
- Panel 2:** Titled "Before you collaborate", it provides a tip: "A little bit of preparation goes a long way with this session. Here's what you need to do to get going." It includes three numbered steps: 1. Team gathering, 2. Set the goal, and 3. Learn how to use the facilitation tools. A "Open article" link is provided.
- Panel 3:** Titled "Define your problem statement", it asks "What problem are you trying to solve? Frame your problem as a how might we statement. This will be the focus of your brainstorming." It features a "PROBLEM" box with the text: "To avoid inconsistency, irrelevant and irregular pattern of news feed and tell us what market news you're interested in and get a quick peek for the day." Below this is a "Key rules of brainstorming" section with icons for staying on topic, encouraging wild ideas, deferring judgment, listening to others, going for volume, and being visual.
- Panel 4 (Right):** Titled "Brainstorm", it prompts "Write down any ideas that come to mind that address your problem statement." It shows four columns of ideas generated by participants: Sophia Anjelica J, Tarunika V, Kavi Priyaa, and Karishma P. Each column contains a grid of ideas in yellow boxes.

**Sophia Anjelica J's ideas:**

- Sign UP
- Choice Based Result
- Accurate results
- Rapid API
- Finding genuine news
- Different language options
- User friendly
- Regular updates
- IBM Cloud

**Tarunika V's ideas:**

- Keep the news relevant and short
- All in one place
- It saves user time
- Identify the bloggers and influencers
- Browsing with the keyword on the web
- Weather updates
- Delivery news according to person interest
- Use only trusted sources
- Increases Productivity

**Kavi Priyaa's ideas:**

- Bookmark news
- Market trends tracking
- Personalized channel
- Short and single line heading
- Search option
- Notification alert
- Competitive analysis
- Health Tracking
- Easy to access the news

**Karishma P's ideas:**

- Regional Reading
- Faster data fetching
- 24/7 availability
- Reduced team notification
- Simple UI
- Flask Backend
- Less Distracting
- Better Security and Backup
- No ads



## Step-2: Brainstorm, Idea Listing and Grouping

### 3 Group Ideas

Take turns sharing your ideas while clustering similar or related notes as you go. Once all sticky notes have been grouped, give each cluster a sentence-like label. If a cluster is bigger than six sticky notes, try and see if you can break it up into smaller sub-groups.

20 minutes

**UI/UX**

- Interactive UI
- Lightweight
- Reduced Spam notifications
- Add to favourites
- Engaging Cloud App

**Maintainability**

- Less complex code
- Write maintainable code
- Maintain Database

**Backend**

- Flask backend
- User data privacy
- Rapid API
- IBM DB

**App Features**

- No ads
- Accurate results
- Different language options
- Choice based result
- Different language options
- Finding genuine news

### 4 Prioritize

Your team should all be on the same page about what's important moving forward. Place your ideas on this grid to determine which ideas are important and which are feasible.

20 minutes

**Importance**

**Feasibility**

Remember: Let our team continue to work on what we can do. The backend can handle the data by using the most trusted sources. We keep on the backend.

### 5 After you collaborate

You can export the final result as an image or pdf to share with members of your company who might find it helpful.

Quick add-ons

- Show the team**  
Share a view link to the board with permissions to keep track of the board and the updates of the board.
- Export the board**  
Export a copy of the board as a PNG or PDF to attach to emails, include it in slides, or save to your drive.

Keep moving forward

- Strategy blueprint**  
Define the components of a new idea or strategy.  
[Open the template](#)
- Customer experience journey map**  
Visualize customer needs, expectations, and preferences for an experience.  
[Open the template](#)
- Strengths, weaknesses, opportunities & threats**  
Identify strengths, weaknesses, opportunities, and threats (SWOT) to develop a plan.  
[Open the template](#)

[Show template features](#)

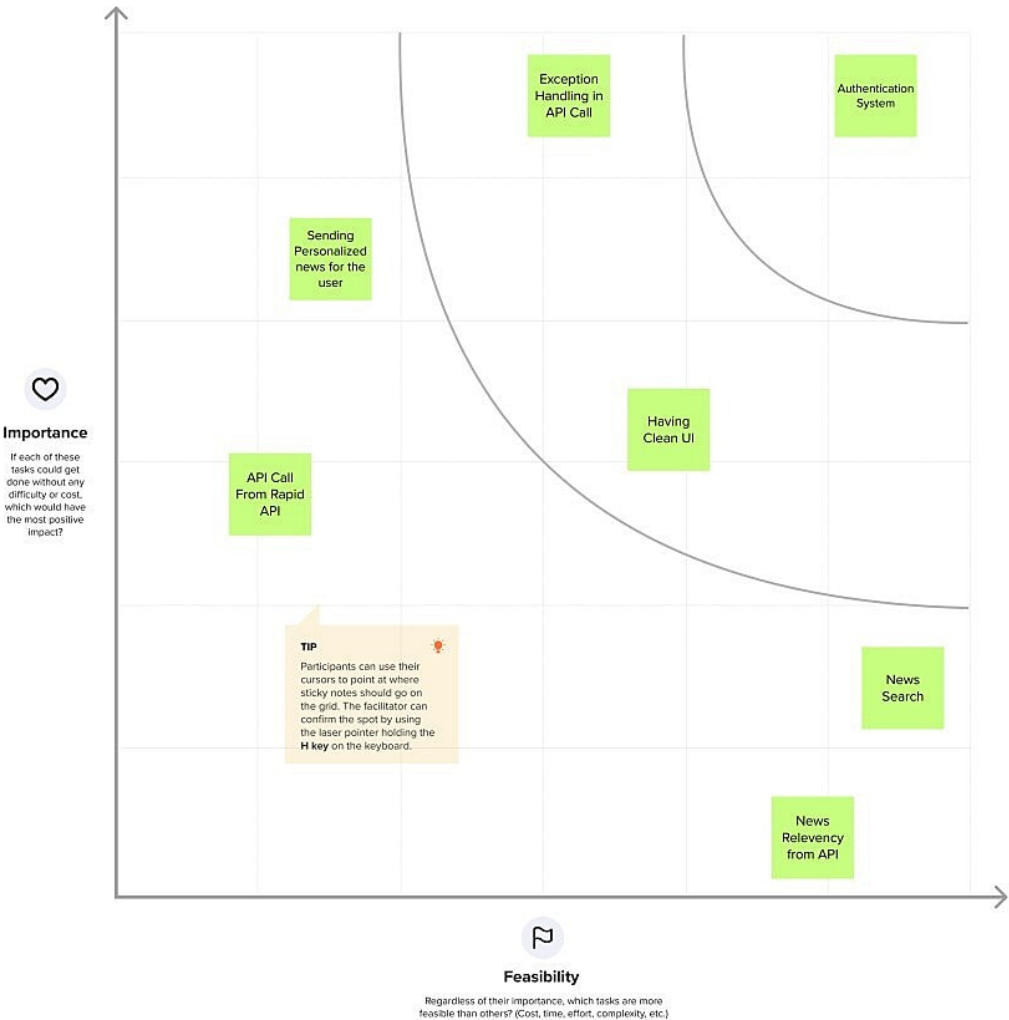
### **Step-3: Idea Prioritization**

4

### Prioritize

Your team should all be on the same page about what's important moving forward. Place your ideas on this grid to determine which ideas are important and which are feasible.

20 minutes



ProposedSolution

S.No.	Parameter	Description
1.	Problem Statement (Problem to be solved)	Most people don't like to carry a newspaper with them. Some people want them to be updated only in the area they are interested in
2.	Idea / Solution description	An application needs to be developed in which users can read news whenever they want and they will be able to customize their area of interest. So that they will be notified, if any new news is updated in their interested areas.
3.	Novelty / Uniqueness	A user can read news only from their interested fields rather than reading all the news. This application provides users with a trusted and secured ecosystem. News shared through the application is original and spam free.
4.	Social Impact / Customer Satisfaction	This application encourages its users to provide feedback. Based on that feedback, developments were made eventually.
5.	Business Model (Revenue Model)	Add advertisements to the application, so that we can get revenue from those advertisement-sponsored organizations. More advertisements may irritate the user. Add premium subscription, users who subscribe for premium won't get advertisements.
6.	Scalability of the Solution	As it was an application-based project, correct ideation and execution can develop an application with no bugs and errors, so that the user might like our application and some might suggest and share it to their surroundings, resulting in an increase in our application insights.

## Problem Solution Fit

<p><b>1. CUSTOMER SEGMENT (S)</b> <b>CS</b></p> <p>General Public from different segments of the society particularly the working class of the society (age group: above 20 yrs)</p> <p><b>Define CS, fit into CC</b></p>	<p><b>6. CUSTOMER CONSTRAINTS</b> <b>CC</b></p> <p>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.</p> <ul style="list-style-type: none"> <li>No choices</li> <li>No reliable sources</li> <li>No budget</li> <li>Tight schedule</li> </ul>	<p><b>5. AVAILABLE SOLUTIONS</b> <b>AS</b></p> <ul style="list-style-type: none"> <li>Newspaper</li> <li>News apps with no user personalization and news summaries</li> <li>Apps like Google news, Flipboard, etc...</li> <li>News apps without search and filter option</li> <li>News apps with just single source of information retrieval</li> </ul> <p><b>Explore AS, differentiate</b></p>
<p><b>2. JOBS-TO-BE-DONE / PROBLEMS</b> <b>J&amp;P</b></p> <p>Which jobs-to-be-done (or problems) do you address for your customers? There could be more than one; explore different sides.</p> <ul style="list-style-type: none"> <li>Summarization of News</li> <li>Aggregation of News from multiple sources</li> <li>Personalization of News Feed</li> <li>Providing Reliable News</li> </ul> <p><b>Focus on J&amp;P, tap into BE, understand RC</b></p>	<p><b>9. PROBLEM ROOT CAUSE</b> <b>RC</b></p> <p>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.</p> <p>People are busy with their tight schedule and hence have no time read news and also have trust issues with the sources.</p>	<p><b>7. BEHAVIOUR</b> <b>BE</b></p> <p>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 (e.g. Greenpeace).</p> <ul style="list-style-type: none"> <li>Enquire with people who are well-informed</li> <li>Look out for sources online</li> <li>Consume news via social media</li> </ul> <p><b>Focus on J&amp;P, tap into BE, understand RC</b></p>
<p><b>3. TRIGGERS</b> <b>TR</b></p> <p>What triggers customers to act? i.e. seeing their neighbour installing solar panels, reading about a more efficient solution in the news.</p> <p>Seeing people around them being well informed of all the current affairs</p> <p><b>Identify strong TR &amp; EM</b></p>	<p><b>10. YOUR SOLUTION</b> <b>SL</b></p> <p>If you are working on an existing business, write down your current solution first, fit in the canvas, and check how much it fits reality. If you are venturing on a new business proposition, then keep it blank until you fit in the canvas and come up with a solution that fits within customer limitations, solves a problem and matches customer behaviour.</p> <p>A simple web application to summarize and aggregate the news from multiple reliable sources and as per the user's preferences.</p>	<p><b>8. CHANNELS OF BEHAVIOUR</b> <b>CH</b></p> <p><b>8.1 ONLINE</b> What kind of actions do customers take online? Extract online channels from #7</p> <ul style="list-style-type: none"> <li>Look out for sources online</li> <li>Consume news via social media</li> </ul> <p><b>8.2 OFFLINE</b> What kind of actions do customers take offline? Extract offline channels from #7 and use them for customer development.</p> <ul style="list-style-type: none"> <li>Enquire with people who are well-informed</li> </ul> <p><b>Extract online &amp; offline CH of BE</b></p>
<p><b>4. EMOTIONS: BEFORE / AFTER</b> <b>EM</b></p> <p>How do customers feel when they face a problem or a job and afterwards? i.e. lost, insecure &gt; confident, in control &gt; relief in your communication strategy &amp; design.</p> <ul style="list-style-type: none"> <li>Before: Frustrated, Lost, Less confident</li> <li>After: Well-informed, Knowledgeable, Happy</li> </ul>		

## REQUIREMENT ANALYSIS

### a. Functional requirement

FR No.	Functional Requirement (Epic)	Sub Requirement (Story / Sub-Task)
FR-1	User Registration	Registration through Email
FR-2	User Confirmation	Confirmation via Email using Sendgrid
FR-3	User Preferred Topics	Collecting user's favourite topics
FR-4	Collecting News	Collecting news from rapid API at regular

		Intervals
FR-5	Bookmark	Bookmarking Favourite news for the user
FR-6	Login	Logging in the user using the registered email and password
FR-7	News Feed	Showing the news fetched from the API to the user

## b. Non-Functional requirements

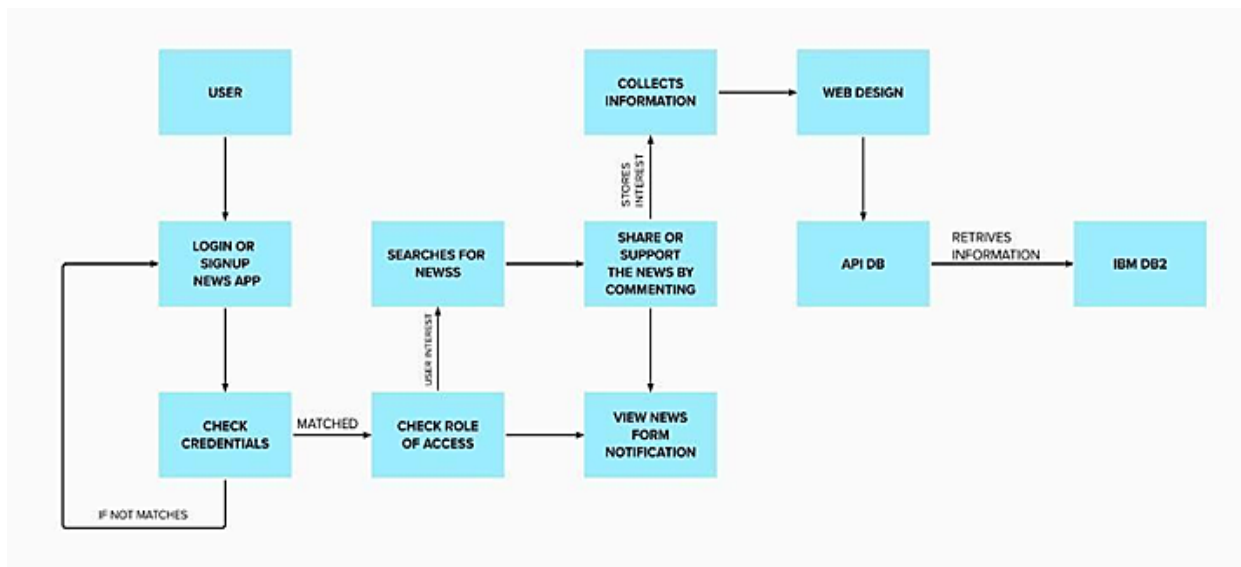
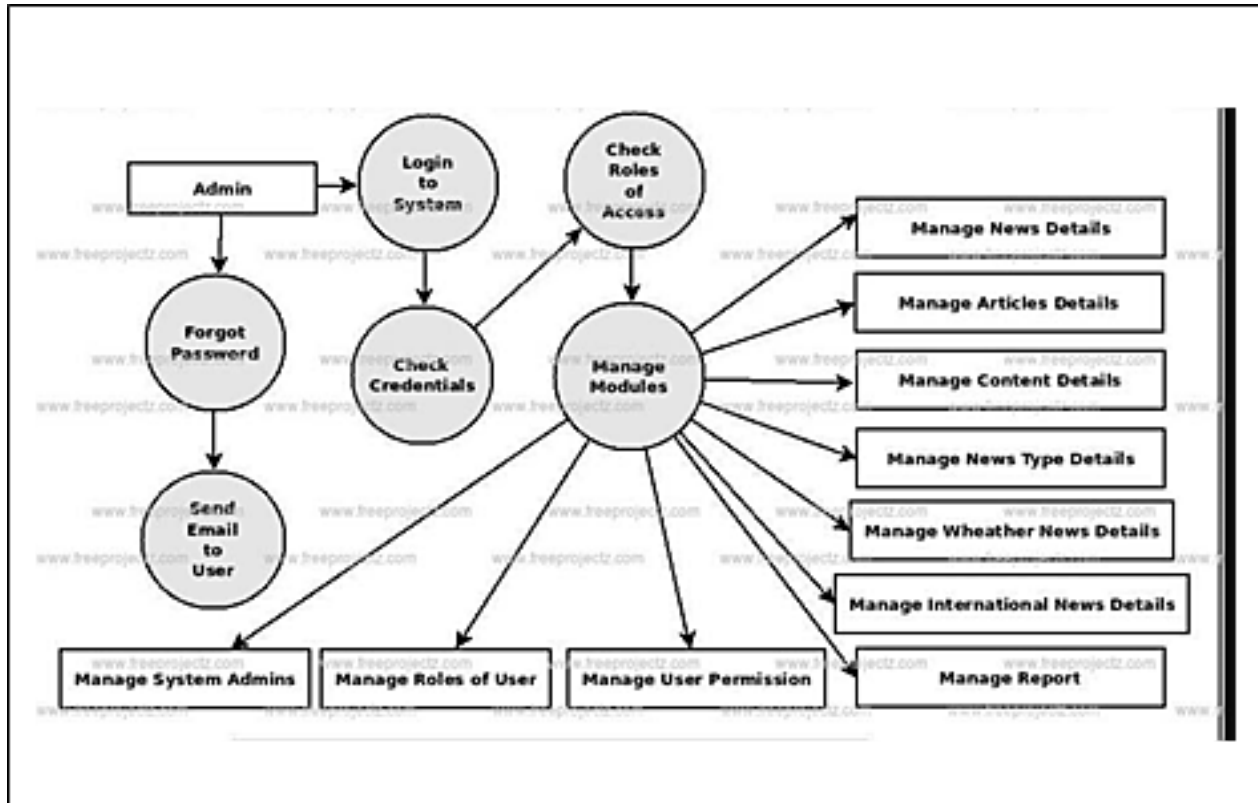
FR No.	Non-Functional Requirement	Description
NFR-1	<b>Usability</b>	The app should be able to be used by all people
NFR-2	<b>Security</b>	The app should authenticate the legitimate users and should restrict bot attacks
NFR-3	<b>Reliability</b>	The app should show the news which are non fake and spam free
NFR-4	<b>Performance</b>	The app should be accessible in older devices too
NFR-5	<b>Availability</b>	The app should be available to all devices
NFR-6	<b>Scalability</b>	The app should handle multiple users and be designed in such a way for future upgrade

## PROJECT DESIGN

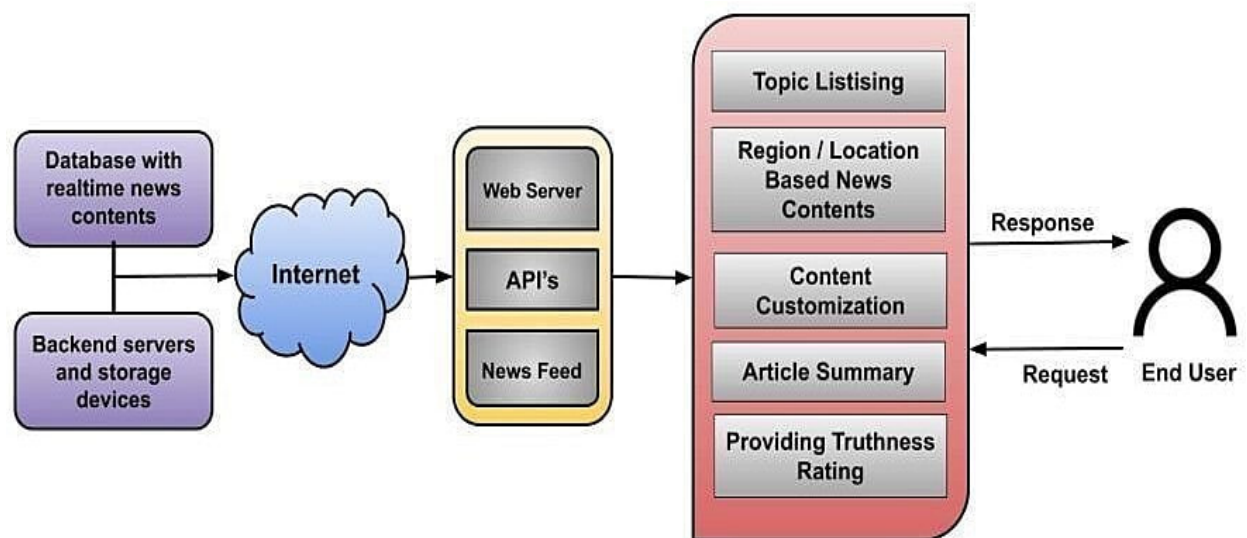
### Data Flow Diagrams

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





## FUNCTIONS:

User Type	Functional Requirement(Epic)	User Story Number	User Story / Task	Acceptance criteria	Priority	Release
Normal 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 / <a href="#">dashboard</a>	High	Sprint-1
	Email Verification	USN-2	As a user, I will receive confirmation email once I have registered for the application	I can receive confirmation email & click confirm	High	Sprint-1
	Favourite topics	USN-3	As a user, I can choose <a href="#">my</a> favourite topics	I can see all <a href="#">my</a> preferred news under <a href="#">for you</a> category	Medium	Sprint-2
	Login	USN-4	As a user, I can login with the email and password into the app	I can access to all the news	High	Sprint-1
	Dashboard	USN-5	As a user, I can see all the news under specific tab	I can view all the news	Medium	Sprint-3
	Bookmark	USN-6	As a user, I can bookmark my favourite news topics	I can later view my bookmarked news	Low	Sprint-2
	Access	USN-7	As a user, I can access the site anywhere and everywhere	I can view the site by typing the URL of the site	High	Sprint-4

## PROJECT PLANNING & SCHEDULING

### Sprint Planning& Estimation

Sprint	Functional Requirement (Epic)	User Story Number	User Story/ Task	StoryPoints	Priority	TeamMembers
Sprint-1	Registration	USN-1	As a user,I can register for the application by entering my email, password, and confirming my password.	10	High	Ariharan S Arun J Arun Kumar K Aseela A
Sprint-1	Login	USN-2	As a user, I will receive confirmationemail once I have registered for the application	5	High	Ariharan S Arun J Arun Kumar K Aseela A
Sprint-1	Email Verification	USN-3	As a user I can verify my email using the link sentto my mailid	5	High	Ariharan S Arun J Arun Kumar K Aseela A
Sprint-2	API Fetch	USN-4	Fetch News fromRapid API at regular interval	10	High	Ariharan S Arun J Arun Kumar K Aseela A
Sprint-2	REST Endpoints for backend	USN-5	Creating endpoints at the backend inorder to interact with frontend	10	Medium	Ariharan S Arun J Arun Kumar K Aseela A

Sprint-3	Designing Frontend	USN-6	Create a minimalisting design in figma to create frontend	2	Medium	Ariharan S Arun J Arun Kumar K Aseela A
Sprint-3	Creating Frontend	USN-7	Create the frontend webpage using the design	10	Low	Ariharan S Arun J Arun Kumar K Aseela A
Sprint-3	Connect frontend and backend	USN-8	Connect the frontend and backend and complete the application	8	High	Ariharan S Arun J Arun Kumar K Aseela A
Sprint-4	Testing	USN-9	Testing the application before final release	10	High	Ariharan S Arun J Arun Kumar K Aseela A
Sprint-4	Deployment	USN-10	Deployment of the application	10	High	Ariharan S Arun J Arun Kumar K Aseela A

## Sprint Delivery Schedule

Milestone List	24-Oct	25-Oct	26-Oct	27-Oct	28-Oct	29-Oct	30-Oct	31-Oct	1-Nov	2-Nov	3-Nov	4-Nov	5-Nov	6-Nov	7-Nov	8-Nov	9-Nov	10-Nov	11-Nov	12-Nov	13-Nov	14-Nov	15-Nov	16-Nov	17-Nov	18-Nov	19-Nov
<b>Sprint 1 - Registration and Sign in</b>																											
Design Sign Up & Sign in Page	1 Day																										
Email Auth		2 Days																									
DB2 Database Design			1 Day																								
Email and Password Sign in				1 Day																							
Email Confirmation on user account creation					1 Day																						
<b>Sprint 2 - API Fetching and Backend Endpoints</b>																											
Fetch data from Rapid API					2 Days																						
Flask-REST API coding							2 Days																				
Create time function for fetch from API using threading								1 Day																			
Test backend									2 Days																		
<b>Sprint 3 - UI and UX design and Connecting frontend and backend</b>																											
Design main Welcome Page										3 Days																	
News Card Design																2 Days											
Explore Designs and Saved Design																	1 Day										
Bookmarks design																		1 Day									
Connecting frontend and backend																			1 Day								
UI responsiveness																			2 Days								
<b>Sprint 4 - Deployment, Testing and Integrations</b>																											
Deploying the App on cloud																						3 Days					
Implementing Loggers																								2 Days			
Final Testing																									2 Days		

## CODING & SOLUTIONING (Explain the features added in the project along with code)

### Feature 1

#### Verification email Sender

The below code is for the verification of email.

As it also ensure a way of security Email verification helps ensure that your contact list is accurate and error free, that the email addresses you have are active, and that they belong to the people you want to reach.

```

1 def emailSender(email, token):
2     configuration = sib_api_v3_sdk.Configuration()
3     configuration.api_key['api-key'] = app.data['mail_api_key']
4     api_instance = sib_api_v3_sdk.TransactionalEmailsApi(
5         sib_api_v3_sdk.ApiClient(configuration))
6     now = datetime.now()
7     dt_string = now.strftime("%d/%m/%Y %H:%M:%S")
8     msg = {}
9     msg['Subject'] = "Verfiy your NewsTracker Account"
10    msg['From'] = {"name": "News Tracker Dev Team",
11                  "email": "verify@news
12    tracker.com"}
13    msg['To'] = [{"email": email}]
14    msg['Text']=f'Please click this <a href="http://127.0.0.1:5500/frontend/pages/verify.html?token={token}">link</a> to
15    verify your account'
16    html = f"""
17    <html>
18    <head></head>
19    <body>
20    <p>👋, for joining NewsTracker 🙌</p>
21    <br>
22    <p>Hurray🎉, you just registerd
23    at NewsTracker<br><br>
24    Please click the following link
25    to verify your account:<br>
26    <a href="http://127.0.0.1:5500/frontend/pages/verify.html?token={token}">Click Here to Verify 🎉</a>
27    </p>
28    <br>
29    <p>⚠️Note: This link expires within one hour from the time sent</p>
30    <br><br>
31    <p>Regrads,<br></p>
32    <p><a href="https://localhost:5000">NewsTracker Dev Team</a></p>
33    <br><br>
34    <p>Email sent at {dt_string}</p>
35    </body>
36    </html>
37    """
38    send_smtp_email = sib_api_v3_sdk.SendSmtpEmail(
39        to=msg['To'], html_content=html,
40        sender=msg['From'], subject=msg['Subject'], text_content=msg['Text'])
41    try:
42        api_response = api_instance.send_transac_email(send_smtp_email)
43        print(api_response)
44    except ApiException as e:
45        print("Exception when calling SMTPApi→send_transac_email: %s\n" % e)

```

The above function is used to send the verification code to the desired email.

## Feature 2

### Cookie Checker

```
1 def token_required(f):
2     @wraps(f)
3     def decorated(*args, **kwargs):
4         token = request.cookies.get("access_token")
5         try:
6             data = jwt.decode(token, app.app.config['SECRET_KEY'], algorithms=['HS256'])
7             ip=request.headers.get("ip")
8             cookieIp=data['ip']
9             if(ip!=cookieIp):
10                 resp={"status":"not logged in"}
11                 @after_this_request
12                 def deleter(response):
13                     response.delete_cookie("access_token",path="/")
14                     response.delete_cookie("email",path="/")
15                     return response
16                 return resp,401
17         except:
18             resp = {"status":"not logged in"}
19             @after_this_request
20             def deleter(response):
21                 response.delete_cookie("access_token",path="/")
22                 response.delete_cookie("email",path="/")
23                 return response
24             return resp, 401
25         return f(data['email'],*args, **kwargs)
26     return decorated
```

This code is used to check the cookie from the client side and checks whether the user is signed in or not.

# Database Schema:

## (ifApplicable) User Table

Table definition

:

×

USER

Approximate 2 rows (4.03 MB)  
Updated on 2022-10-17 15:43:28

Name	Data type	Nullable	Length	Scale	
ID	INTEGER	N		0	👁
NAME	VARCHAR	N	255	0	👁
EMAIL	VARCHAR	N	255	0	👁
PASSWORD	VARCHAR	N	255	0	👁
FAVOURITES	CLOB	Y	1048576	0	👁
BOOKMARKS	CLOB	Y	1048576	0	👁
VERIFIED	BOOLEAN	Y	1	0	👁
RESEND_TIME	VARCHAR	N	255	0	👁

## Bookmarks Table:

### Table definition

BOOKMARK

Approximate -1 rows (4.00 MB)  
Updated on

Name	Data type	Nullable	Length	Scale	
ID	INTEGER	N		0	👁
DATA	CLOB	N	1048576	0	👁

## ADVANTAGES & DISADVANTAGES

1. This app can be accessed anywhere and anytime, So that the user can view the news
2. Its ad free
3. The news is only based on the API
4. It may contain some unwanted content but we don't have control over it
5. The user can bookmark their favourite news.



## **CONCLUSION**

Thus we have developed a full stack application based on the plans and within the given time. We have tested the application in both desktop and mobile and it worked well, Overall it was a great experience.

## **FUTURE SCOPE**

In future we may integrate our own news API instead of third party APIs and may develop a mobile native application so that it can be used in both android and ios.

## **DEMO LINK:**

- [LIVE WEBSITE LINK](#)
- [DEMO VIDEO LINK](#)

