PROJECT REPORT

1. INTRODUCTION

Android provides simple application structure and requires Java and Mark-up languages knowledge to work with. Such as, an discrete movement delivers a solitary screen for a user interface and a service whole completes work in the contextual. We can work on different module separately and can combine at the end, we can also add future modules easily afterwards. API (Application Programming Interface) which is an intermediate interface between different applications. It provides automation, immediacy, adaption and personalization. News API provides us the source of news articles from many different sources at one place and updates it. To expand the sources old fashioned Admin panel can be used where writers will fill the gap of API. In 2014, a design language has been created by google named Material Design which is based on “cards” uses grind based layouts, responsive animation, padding and depth effects like shadow to create an responsive, attractive and easy user interface. With the use of different libraries and material design it is possible to use attractive UI.

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 120+ newspapers from 50+ countries. The main focus of this application is to connect news articles from all around the world and deliver it to user as fast as possible in best visualize way.

1.2 Purpose

The purpose of the project is to develop an application which will display news articles and videos verified and approved at backend .User can browse news articles and videos through various categories. App will enable user to view news details, images, and video .Users can also bookmark any news article/video for future use. The application will have three type of user's at backend one will be admin manager with having full access to backend features including user and content management. Whereas the other two users will be reporters and editors will have limited access to backend features. The reporters will be creating news articles and videos on the other hand editors will have responsibility to approve and publish the articles and videos added by reporters. Both users will be added to the system by admin.

2.LITERATURE SURVEY

Here, we will take a look at all the previous solutions, attempts and implementations tothe news tracker application or anything that is atleast vaguely related to it. Existing Solutions NewsBreak is a popular website to read ongoing and past news via the internet browsers. The website works by aggregating news from various sources and presents them in alikeable manner for the users to read it. The website also offers the ability for users to sign up to the so said website and record their progress, manage profiles, no.of news read, bookmark news, commenting on news ends and so on.

2.1 Existing problem

- News Apps Are more convenient to read.One Can easily relax, Reclined over a deck chair And reading a Newspaper.This kind of reading is not possible with internet news.
- Newspaper can be carried and read at only location of the world.

There is no precondition of availability of internet connectivity.

- Reading newspaper doesnot require any equipment such as computer.
- It is important to note that internet capabilities are improving very fast,and with that

The internet news is moving toward overcoming its current disadvantage over

Newspaper.

2.2 References

S.No	Paper Title	Author(s)	Month /Year	Method/Implementation technique(s)	Resource Link
1	Exploring mobile news reading interactions for news app personalization	Marios Constantinide s,John Dowell, DavidJohson, Sylvain Malacria	August, 2015	1. Identification of news reader types 2. Interaction logging and classification study 3. Deployment and data collection 4. Predicting News reader types 5. Adaptive UI	(PDF) Exploring mobile news reading interactions for newsapp personalisation (researchgate.net)
2	Detection and Tracking in NewsArticles	Sagar Patel, Sanket Suthar, SandipPatel,Neha Patel	March, 2015	1. Pre-processing 2. Tokenization 3. Stemming/Lemmization 4. Vector Space Model 5. Topic tracking	(PDF) Topic Detection and Tracking in News Articles (researchgate.net)

3	Following the Fed with a NewsTracker	Michael William McCracken	January, 2012	The paper is not a technical paper but is essentially a statistical paper on how should one conclude whether the data have comein stronger, weaker or as expected. Thisis based on the CitiGroup U.S Economic Surprise Index.	(PDF) Following theFed with a News Tracker (researchgate.net)

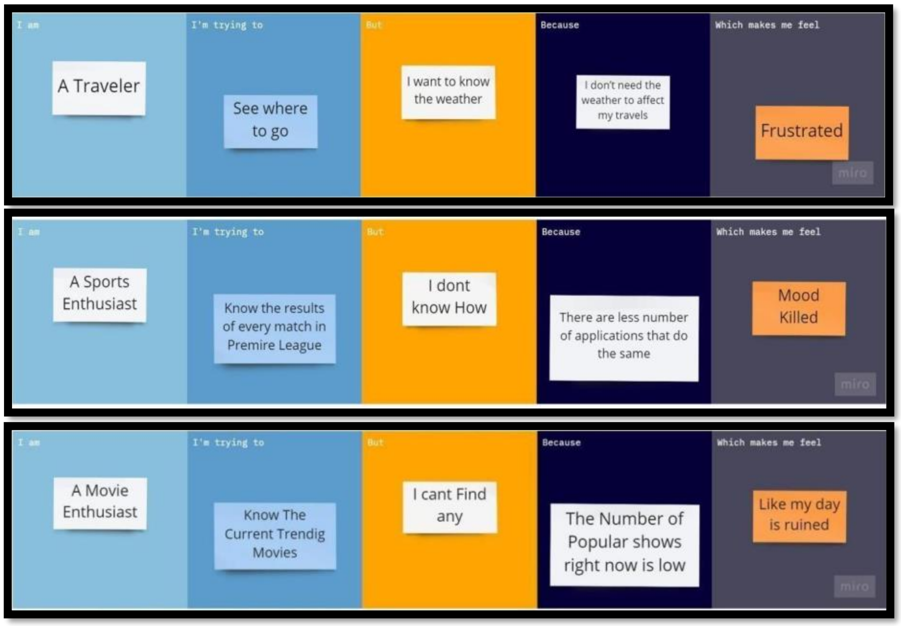
2.3 Problem Statement Definition

PROBLEM:

We encounter several events of local and national significance every day, and we rely on NEWS to inform us about them. However, we recently found there isn't a single platform that does both. As a result, we identified a social need and created a platform that offers such news in the form of brief films that are taken from the most reliable sources worldwide.

SOLUTION:

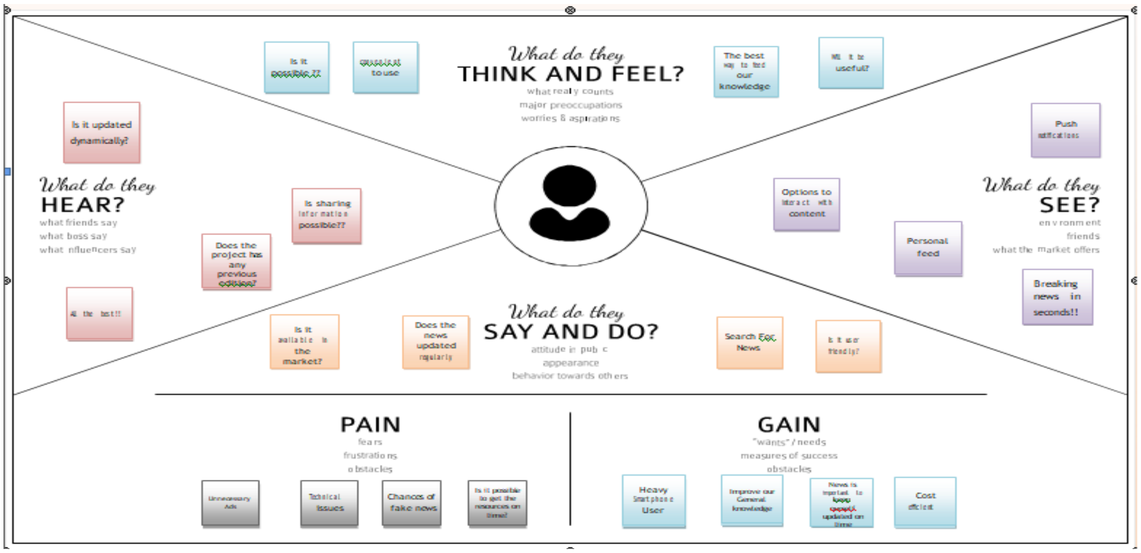
We deliver news in the form of quick videos. Distributing news from a reliable, trustworthy source everywhere, preventing the spread of false information. We offer local news in locations where there was already a news channel. One platform to read all of his/her favorite local news, dependence on search engines and social media. We added multiple languages to our platform so that locals could readily understand it.



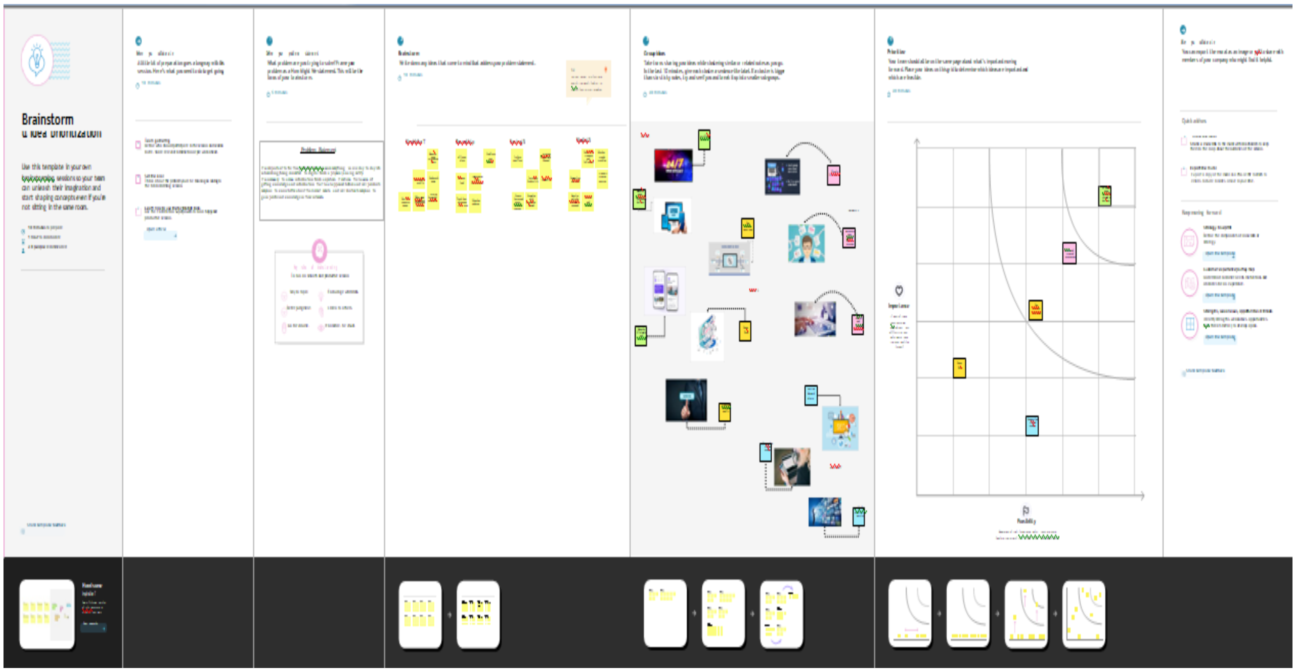
I am	I'm trying to	But	Because	Which makes me feel
A Traveler	See where to go	I want to know the weather	I don't need the weather to affect my travels	Frustrated
A Sports Enthusiast	Know the results of every match in Premire League	I dont know How	There are less number of applications that do the same	Mood Killed
A Movie Enthusiast	Know The Current Trendig Movies	I cant Find any	The Number of Popular shows right now is low	Like my day is ruined

3. IDEATION & PROPOSED SOLUTION

3.1 Empathy Map Canvas



3.2 Ideation & Brainstorming



Brainstorm
& idea prioritization

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.

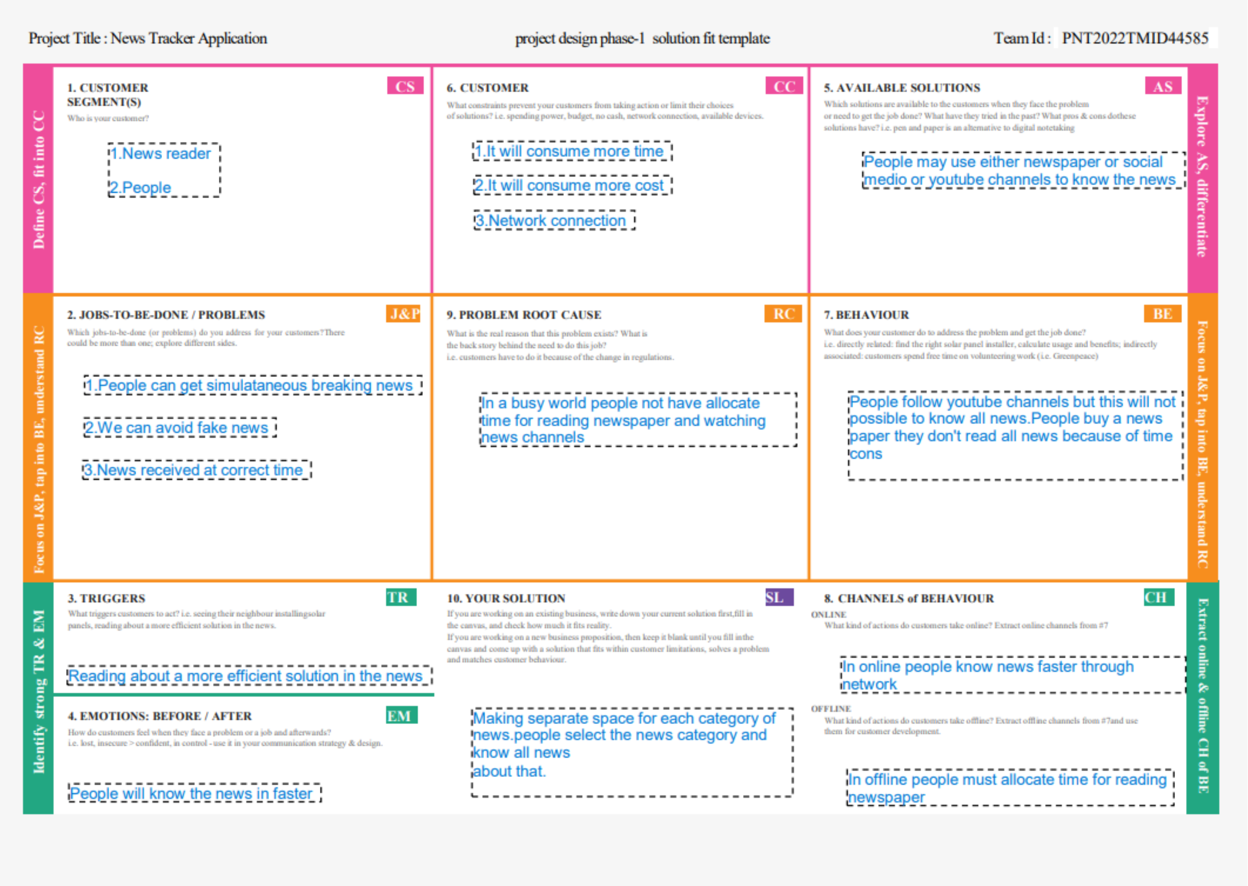
Problem Statement

3.3 Proposed Solution

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	1. A user can read news only from their interested fields rather than reading all the news. 2. 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)	1. Add advertisements to the application, so that we can get revenue from those advertisement-sponsored organizations. More advertisements may irritate the user. 2. 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.

3.4 Problem Solution fit



Project Title : News Tracker Application

project design phase-1 solution fit template

Team Id : PNT2022TMID44585

Define CS, fit into CC

1. CUSTOMER SEGMENT(S) CS

Who is your customer?

1.News reader

2.People

6. CUSTOMER CC

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.It will consume more time

2.It will consume more cost

3.Network connection

5. AVAILABLE SOLUTIONS AS

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 dothese solutions have? i.e. pen and paper is an alternative to digital notetaking

People may use either newspaper or social medio or youtube channels to know the news

Explore AS, differentiate

Focus on J&P, tap into BE, understand RC

2. JOBS-TO-BE-DONE / PROBLEMS J&P

Which jobs-to-be-done (or problems) do you address for your customers?There could be more than one; explore different sides.

1.People can get simulataneous breaking news

2.We can avoid fake news

3.News received at correct time

9. PROBLEM ROOT CAUSE RC

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.

In a busy world people not have allocate time for reading newspaper and watching news channels

7. BEHAVIOUR BE

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)

People follow youtube channels but this will not possible to know all news.People buy a news paper they don't read all news because of time cons

Focus on J&P, tap into BE, understand RC

Identify strong TR & EM

3. TRIGGERS TR

What triggers customers to act? i.e. seeing their neighbour installingsolar panels, reading about a more efficient solution in the news.

Reading about a more efficient solution in the news

4. EMOTIONS: BEFORE / AFTER EM

How do customers feel when they face a problem or a job and afterwards? i.e. lost, insecure > confident, in control - use it in your communication strategy & design.

People will know the news in faster

10. YOUR SOLUTION SL

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 inthe canvas and come up with a solution that fits within customer limitations, solves a problem and matches customer behaviour.

Making separate space for each category of news.people select the news category and know all news about that.

8. CHANNELS of BEHAVIOUR CH

ONLINE

What kind of actions do customers take online? Extract online channels from #7

In online people know news faster through network

OFFLINE

What kind of actions do customers take offline? Extract offline channels from #7and use them for customer development.

In offline people must allocate time for reading newspaper

Extract online & offline CH of BE

4.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 Registration	Registration through online application Registration through Gmail Registration through website

FR-2	User Confirmation	Confirmation via Email Confirmation via OTP
FR-3	User login	Login through browser directly by enteringusername and password Login through Login through email
FR-4	User interaction	Done through user interface between clientand server View the related news by subscripted or requested page

Non-Functional requirements

FR No.	**Non-Functional Requirement**	**Description**
NFR-1	**Usability**	End users can receive push updates fornew content on a site by subscribing to the site’s news feed
NFR-2	**Security**	How well are the system and its data protected against attacks
NFR-3	**Reliability**	How often does the system experience critical failures? How much time does it take to fix the issue when it arises ?And how is user availability time compared to downtime?

NFR-4	**Performance**	Performance is the core non-functional requirements no system can do without.Itdefines how fast a software system or a particular piece of it responds to certain users actions under a certain workload. Inmost cases, this metric explains how longa user must wait before the target operation happens (the page renders, a transaction is processed, etc.) given the overall number of users at the moment. But it’s not always like that. Performance requirements may describe background processes invisible to users, e.g. backup. But let’s focus on user-centric performance.
NFR-5	**Availability**	Availability describes how likely the system is accessible to a user at a given point in time. While it can be expressed as an expected percentage of successful requests, you may also define it as a percentage of time the system is accessible for operation during some timeperiod. For instance, the system may be available 98 percent of the time during a month. Availability is perhaps the most business-critical requirement, but todefine it, you also must have estimations for reliability and maintainability.

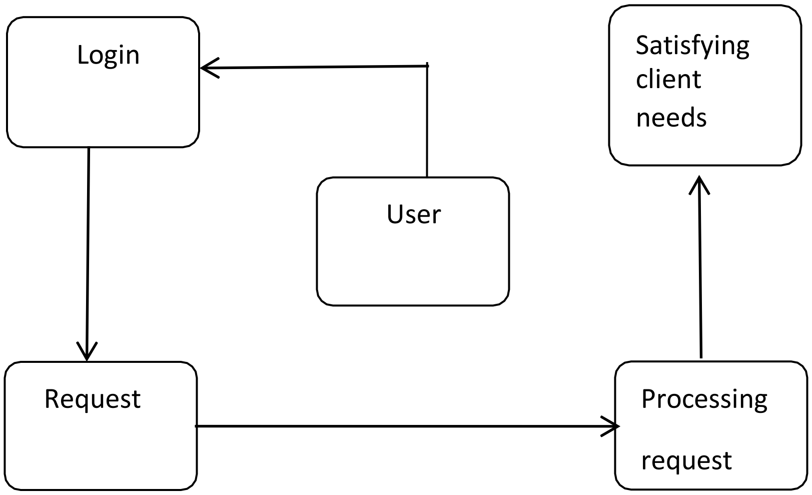
NFR -6	**Scalability**	Scalability assesses the highest workloads under which the system will still meet the performance requirements. There are two ways to enable your system scale as the workloads get higher: horizontal and vertical scaling.

5. PROJECT DESIGN

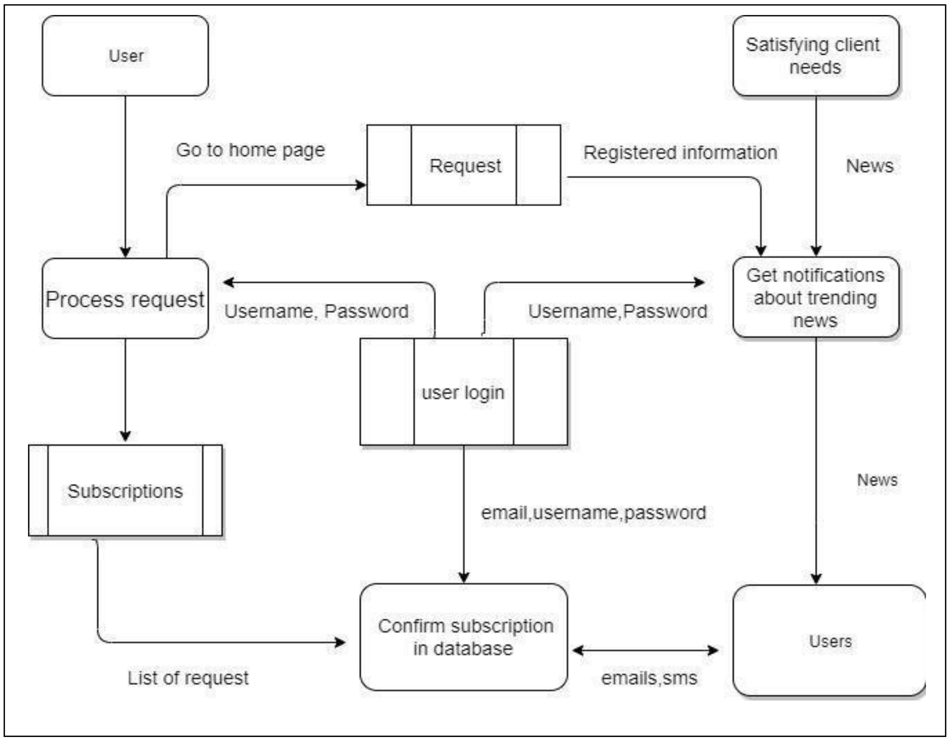
5.1 Data Flow Diagrams

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 enter and leaves the system, what changes the information, and where data is stored.

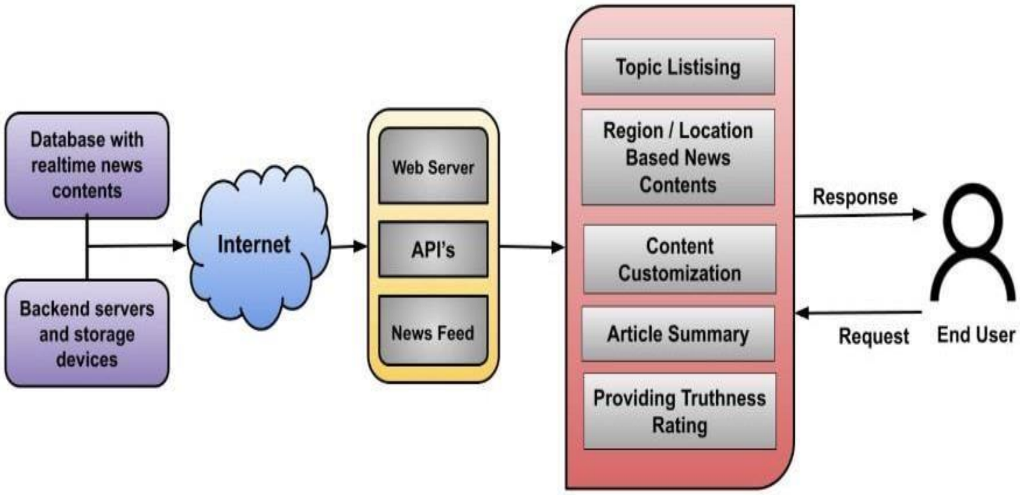
Example: (Simplified)



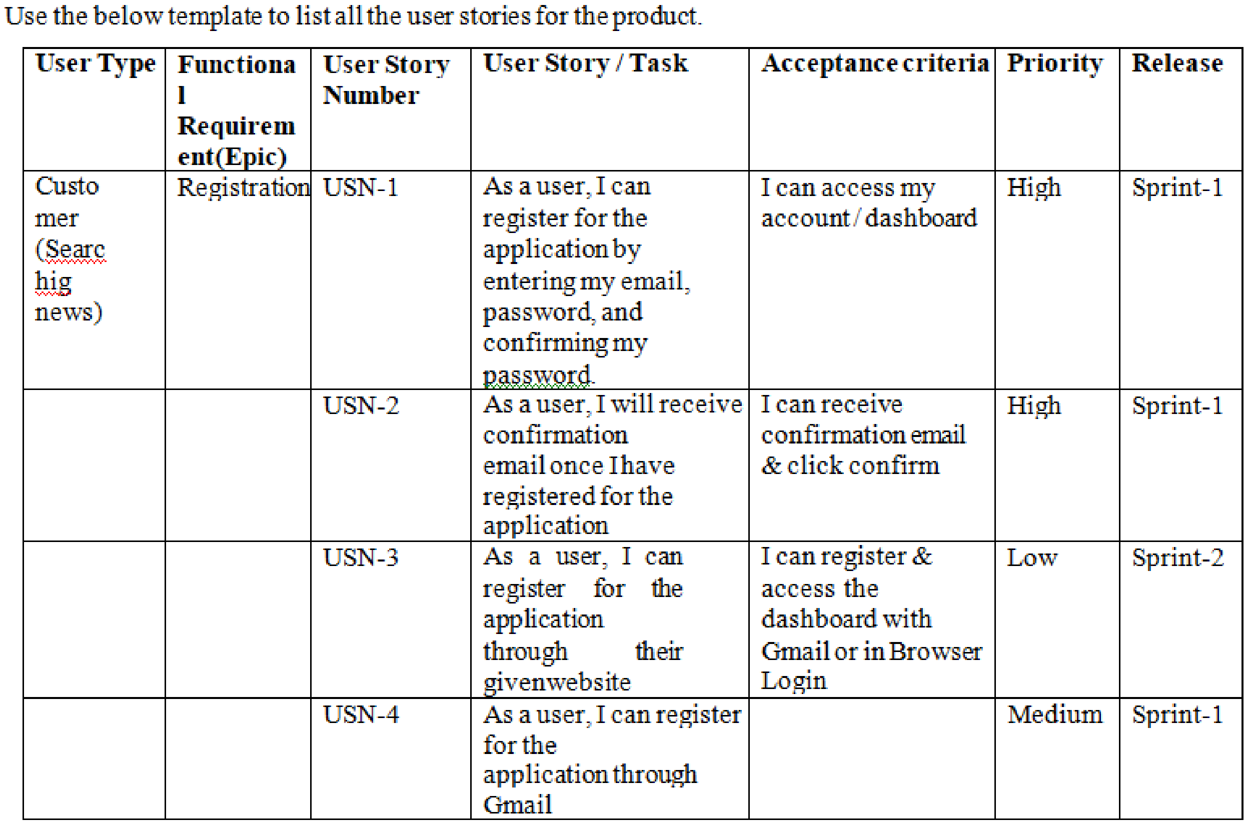
Example: DFD Level 0 (Industry Standard)



Solution & Technical Architecture

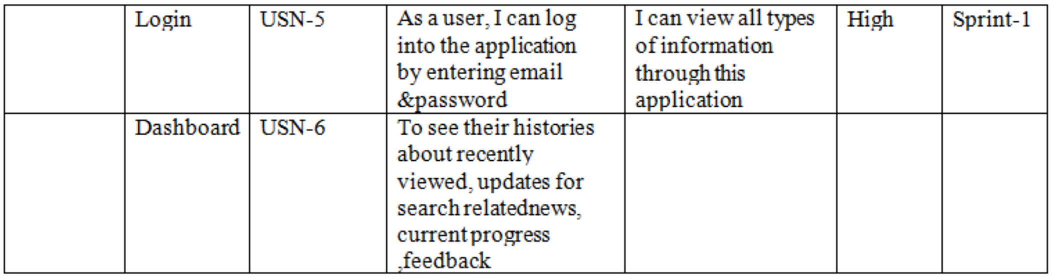

Database with realtime news contents
Backend servers and storage devices
Internet
Web Server
API's
News Feed
Topic Listising
Region / Location Based News Contents
Content Customization
Article Summary
Providing Truthness Rating
Response
Request
End User

5.3 User Stories

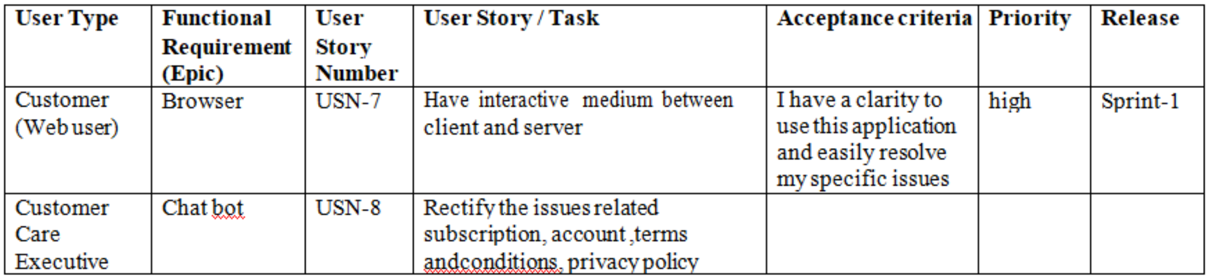


Use the below template to list all the user stories for the product.

User Type	Functional Requirement(Epic)	User Story Number	User Story / Task	Acceptance criteria	Priority	Release
Customer (Searchig news)	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 their givenwebsite	I can register & access the dashboard with Gmail or in Browser 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	I can view all types of information through this application	High	Sprint-1
	Dashboard	USN-6	To see their histories about recently viewed, updates for search relatednews, current progress ,feedback			

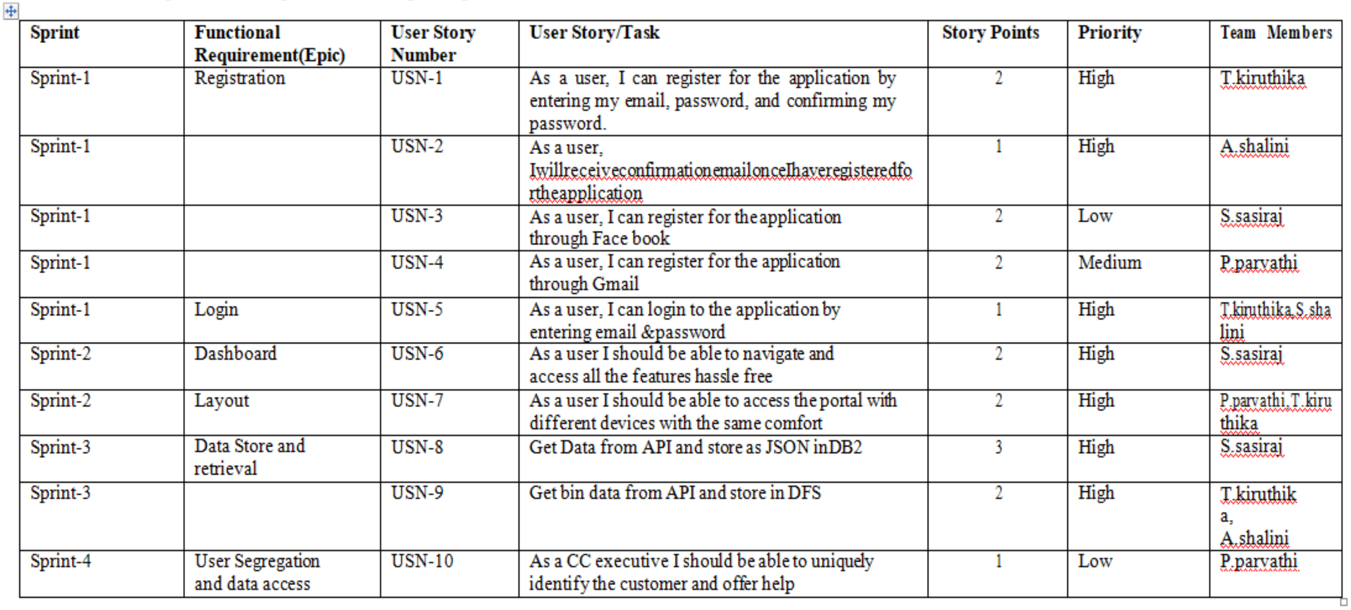


User Type	Functional Requirement (Epic)	User Story Number	User Story / Task	Acceptance criteria	Priority	Release
Customer (Web user)	Browser	USN-7	Have interactive medium between client and server	I have a clarity to use this application and easily resolve my specific issues	high	Sprint-1
Customer Care Executive	Chat bot	USN-8	Rectify the issues related subscription, account ,terms andconditions, privacy policy			

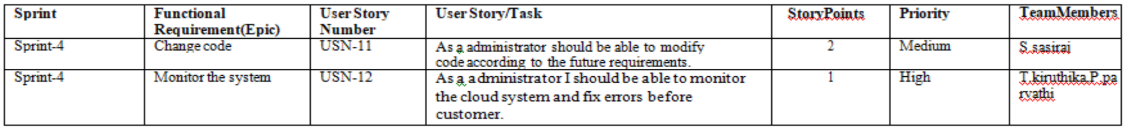
6. PROJECT PLANNING & SCHEDULING

6.1 Sprint Planning & Estimation

Use the below template to create product back log and sprint schedule

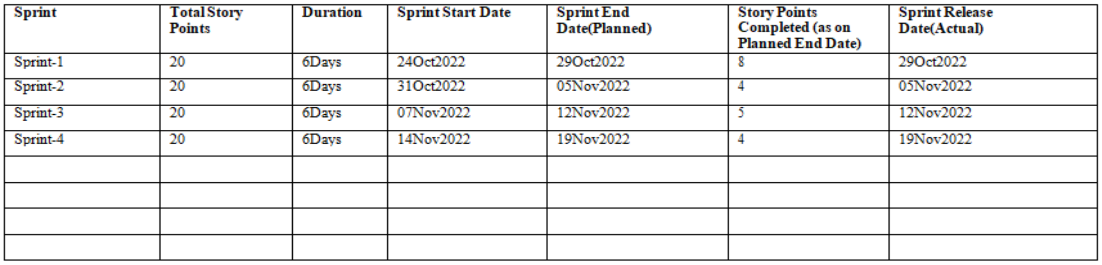


Sprint	Functional Requirement(Epic)	User Story Number	User Story/Task	Story Points	Priority	Team Members
Sprint-1	Registration	USN-1	As a user, I can register for the application by entering my email, password, and confirming my password.	2	High	T.kiruthika
Sprint-1		USN-2	As a user, Iwillreceiveconfirmationemailoncelhaveregisteredfortheapplication	1	High	A.shalini
Sprint-1		USN-3	As a user, I can register for the application through Face book	2	Low	S.sasiraj
Sprint-1		USN-4	As a user, I can register for the application through Gmail	2	Medium	P.parvathi
Sprint-1	Login	USN-5	As a user, I can login to the application by entering email &password	1	High	T.kiruthika,S.shalini
Sprint-2	Dashboard	USN-6	As a user I should be able to navigate and access all the features hassle free	2	High	S.sasiraj
Sprint-2	Layout	USN-7	As a user I should be able to access the portal with different devices with the same comfort	2	High	P.parvathi,T.kiruthika
Sprint-3	Data Store and retrieval	USN-8	Get Data from API and store as JSON inDB2	3	High	S.sasiraj
Sprint-3		USN-9	Get bin data from API and store in DFS	2	High	T.kiruthika, A.shalini
Sprint-4	User Segregation and data access	USN-10	As a CC executive I should be able to uniquely identify the customer and offer help	1	Low	P.parvathi



Sprint	Functional Requirement(Epic)	User Story Number	User Story/Task	StoryPoints	Priority	TeamMembers
Sprint-4	Change code	USN-11	As a administrator should be able to modify code according to the future requirements.	2	Medium	S.sasirai
Sprint-4	Monitor the system	USN-12	As a administrator I should be able to monitor the cloud system and fix errors before customer.	1	High	T.kiruthika,P.parvathi

Project Tracker,Velocity & Burn down Chart: (4 Marks)



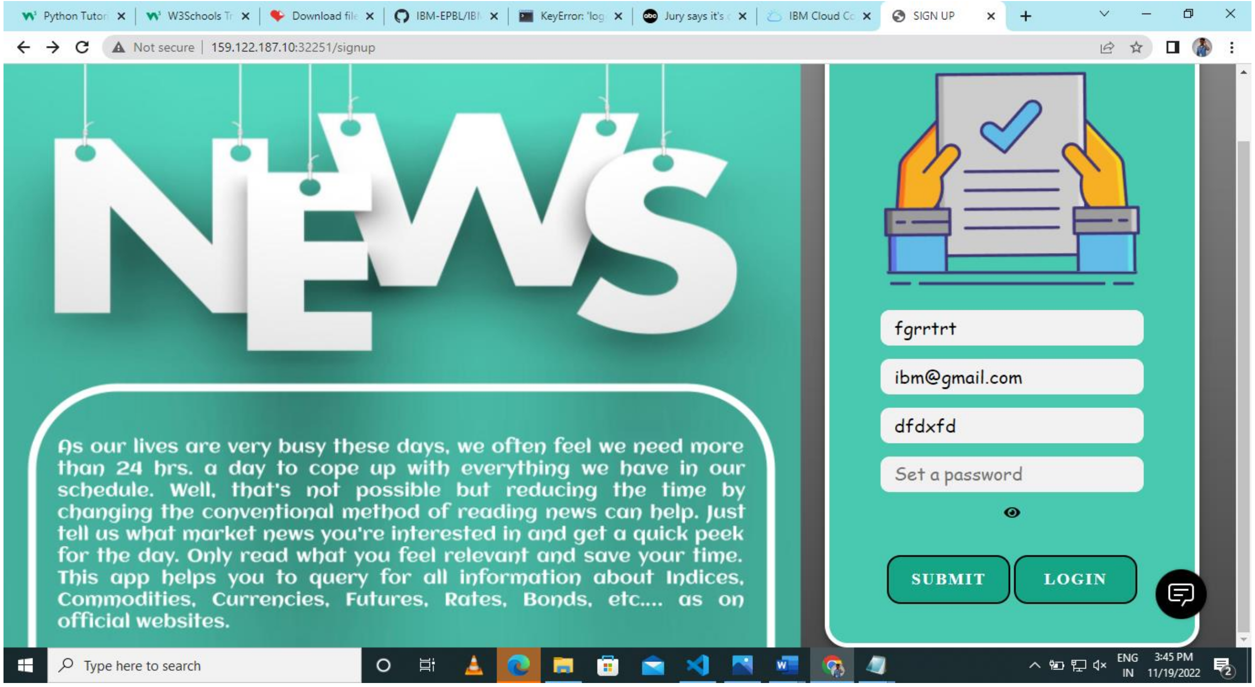
Sprint	Total Story Points	Duration	Sprint Start Date	Sprint End Date(Planned)	Story Points Completed (as on Planned End Date)	Sprint Release Date(Actual)
Sprint-1	20	6Days	24Oct2022	29Oct2022	8	29Oct2022
Sprint-2	20	6Days	31Oct2022	05Nov2022	4	05Nov2022
Sprint-3	20	6Days	07Nov2022	12Nov2022	5	12Nov2022
Sprint-4	20	6Days	14Nov2022	19Nov2022	4	19Nov2022

6.2 Sprint Delivery Schedule

TITLE	DESCRIPTION	DATE
Literature Survey & Information Gathering	Literature survey on the selected project & gathering information by referring the, technical papers, research publications etc.	16 OCTOBER 2022
Prepare Empathy Map	Prepare Empathy Map Canvas to capture the user Pains & Gains, Prepare listof problem statements	16 OCTOBER 2022
Ideation	List the by organizing the brainstorming session and prioritize the top 3 ideas based on the feasibility & importance.	16 OCTOBER 2022
Proposed Solution	Prepare the proposed solution document, which includes the novelty, feasibility of idea, business model, social impact, scalability of solution, etc.	16 OCTOBER 2022
Problem Solution Fit	Prepare problem - solution fit document.	21 OCTOBER 2022
Solution Architecture	Prepare solution architecture document.	21 OCTOBER 2022

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

7.1 Feature 1



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 relevant and save your time. This app helps you to query for all information about Indices, Commodities, Currencies, Futures, Rates, Bonds, etc.... as on official websites.

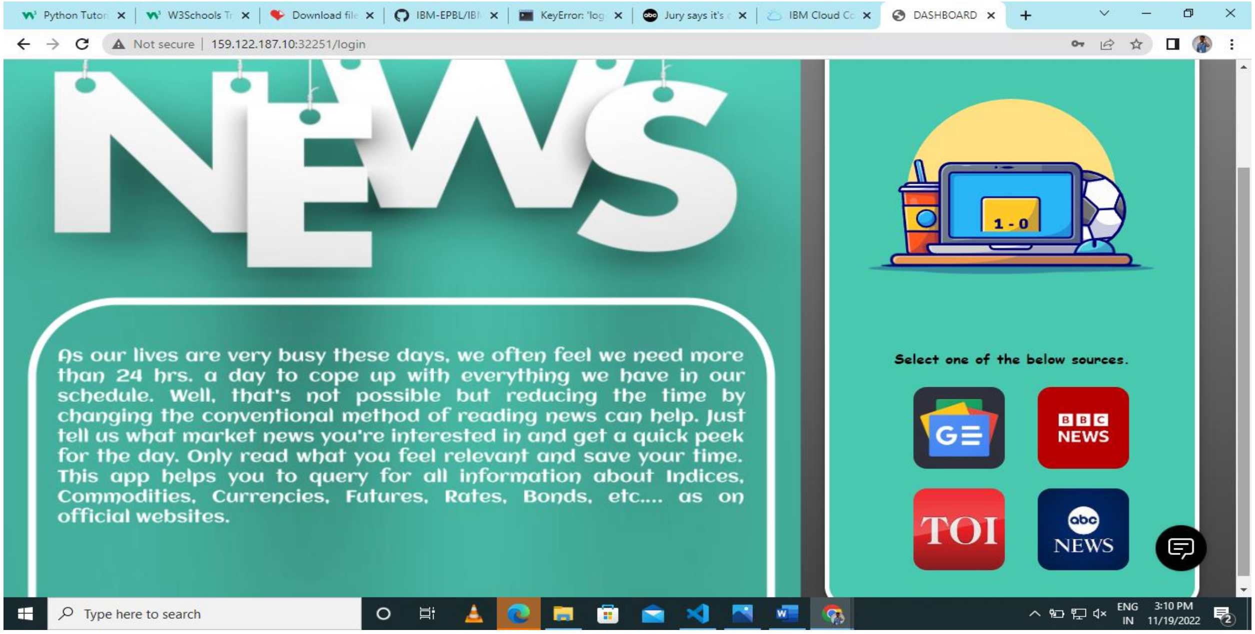
fgrrtrt

ibm@gmail.com

dfdxfd

Set a password

7.2 Feature 2



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 relevant and save your time. This app helps you to query for all information about Indices, Commodities, Currencies, Futures, Rates, Bonds, etc.... as on official websites.

Select one of the below sources.

8. TESTING

8.1 Test Cases

Software testing is the process of everything and verifying that a software products or application does what it is supposed to do. The benefits of testing include preventing bugs, reducing development costs and improving performance.

8.2 User Acceptance Testing

1. Purpose of Document

The purpose of this document is to briefly explain the test coverage and open issues of the News Tracker Application 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	***Severity 1***	***Severity 2***	***Severity 3***	***Severity 4***	***Subtotal***
By Design	8	3	4	2	17
Duplicate	2	1	0	6	9
External	6	3	0	1	16
Fixed	5	1	4	10	20
Not Reproduced	1	2	0	1	4
Skipped	0	0	9	3	12
Won't Fix	2	3	1	1	7
Totals	24	13	18	24	79

3.Test Case Analysis

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

Section	*Total Cases*	*Not Tested*	*Fail*	*Pass*
Print Engine	8	0	0	8
Client Application	23	0	0	23
Security	3	0	0	3
Outsource Shipping	2	0	0	2
Exception Reporting	6	0	0	6
Final Report Output	6	0	0	6
Version Control	1	0	0	1

9. RESULTS

Performance Metrics

NFT - Risk Assessment	
Project Name	News Tracker Application
Scope/feature	New
Functional Changes	Low
Hardware Changes	No Changes
Software Changes	Moderate
Impact of Downtime	No Impact of Downtime

Load/Voluem Changes	>5 to 10%
Risk Score	ORANGE
Justification	As we have seen the chnages

NFT - Detailed Test Plan	
Project Overview	News Tracker Application
NFT Test approach	Locust
Assumptions/Dependencies/Risks	No Risk Factors
Approvals/SignOff	Approved

End Of Test Report	
Project Overview	News Tracker Application
NFT Test approach	Locust
NFR - Met	Nil
Test Outcome	Working good
GO/NO-GO decision	None
Recommendations	None
Identified Defects(Detected/Closed/Open)	No Defects Identified
Approvals/SignOff	Approved

10.ADVANTAGES & DISADVANTAGES

Advantage

The main focus of this application is to connect news articles from all around the world.

Deliver it to user as fast as possible in best visualize way.

One of the main benefits of this news app is that day can be used in case of emergencies**.**

Disadvantage

Require data/ wifi to get online.

Companies not making as much money due to free reading for audiences.

News spreads quicker online - people find out news before they should.

Lose money - can't get people to pay for digital.

Older audiences may not access digital platforms.

Costly to maintain.

11.CONCLUSION

When developing a news app, it is important to include all the crucial features that make it engaging and easy to use. So, here are the top features of a great news app.

However, it is also necessary to research your competitors to identify other useful features that can be included in your future or existing apps.

12. FUTURE SCOPE

A mobile news app has simple goals: Capture and retain reader attention, and repeat the process, several times a day. Pretty straightforward. But not that simple in the real world. For a news provider, the smartphone screen is the the most challenging environment ever seen. There, chances are that a legacy media or a pure-player will find itself in direct competition, not only with the usual players in its field, but also with Facebook, Snapchat, Instagram and scores of gaming applications.

13. APPENDIX

13.1 Source Code

Dashboard.html

```
<html>

  <head>

    <title>DASHBOARD</title>

    <meta name="viewport" content="width=device-width, initial-scale=1.0">

    <meta charset="utf-8">

    <link rel="stylesheet" type="text/css" href="{{ url_for('static', filename='dashboard.css') }}">

    <link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/font-awesome/4.7.0/css/font-awesome.min.css">

    <script src="https://unpkg.com/@lottiefiles/lottie-player@latest/dist/lottie-player.js"></script>

    <link rel="stylesheet" href="https://use.fontawesome.com/releases/v5.15.1/css/all.css" integrity="sha384-vp86vTRFVJgpjF9jiIGPEEqYqlDwgyBgEF109VFjmqGmIY/Y
  4HV4d3Gp2irVfcrp" crossorigin="anonymous">

    <script>

   window.watsonAssistantChatOptions =
   {
    integrationID: "78eea96f-f7bf-4a38-bd83-3eac507aacb7", //
  The ID of this integration.
    region: "au-syd", // The region your integration is hosted in.
    serviceInstanceID: "a57bbab3-1d2e-4714-ae9b-
  c3c2c507ef41", // The ID of your service instance.
    onLoad: function(instance) { instance.render(); }
   };

   setTimeout(function()
   {
    const t=document.createElement('script');
    t.src="https://web-
  chat.global.assistant.watson.appdomain.cloud/versions/" +
  (window.watsonAssistantChatOptions.clientVersion || 'latest') +
  "/WatsonAssistantChatEntry.js";
    document.head.appendChild(t);
   });

  </script>

</head>

<body class="body">
```

```
<div class="on-screen">

 <div class="form">

  <form action="/news" method="POST">

   <h4>Welcome,{{username}}</h4>

   <lottie-player
 src="https://assets7.lottiefiles.com/packages/lf20_d8E2BRvHrr.j
 son"  speed="1"  style="justify-content: center;" loop
 autoplay></lottie-player>

   <center><h4>Select one of the below sources.</h4></center>



   <button type="submit"name="newsresource"
 value="google"><img name="newsresource"
 value="abc"src="https://ibm-news-tracker.s3.jp-tok.cloud-
 object-storage.appdomain.cloud/google.jpg" /></button>

   <button type="submit"name="newsresource"
 value="bbc"><img name="newsresource"
 value="abc"src="https://ibm-news-tracker.s3.jp-tok.cloud-
 object-storage.appdomain.cloud/bbc.jpg" /></button><br><br>

   <button type="submit"name="newsresource"
 value="toi"><img name="newsresource"
 value="abc"src="https://ibm-news-tracker.s3.jp-tok.cloud-
 object-storage.appdomain.cloud/toi.jpg" /></button>
```

```
            <button type="submit"name="newsresource"
          value="abc"><img name="newsresource"
          value="abc"src="https://ibm-news-tracker.s3.jp-tok.cloud-
          object-storage.appdomain.cloud/abc.jpg" /></button>


            <p class="message"></p>


          </form>


         </div>


        </div>
        </body>
    </html>
```

Forgot.html

```
      <html>


        <head>


         <title>FORGOT USERNAME/PASSWORD</title>


         <meta name="viewport" content="width=device-width, initial-
          scale=1.0">


         <meta charset="utf-8">


         <link rel="stylesheet" type="text/css" href="{{ url_for('static',
filename='pages.css') }}">
<link rel="stylesheet"
href="https://cdnjs.cloudflare.com/ajax/libs/font-
awesome/4.7.0/css/font-awesome.min.css">


<script src="https://unpkg.com/@lottiefiles/lottie-
player@latest/dist/lottie-player.js"></script>


<link rel="stylesheet"
href="https://use.fontawesome.com/releases/v5.15.1/css/all.css"
integrity="sha384-
vp86vTRFVJgpjF9jiIGPEEqYqlDwgyBgEF109VFjmqGmIY/Y
4HV4d3Gp2irVfcrp" crossorigin="anonymous">


   <script>


 window.watsonAssistantChatOptions =
 {
   integrationID: "78eea96f-f7bf-4a38-bd83-3eac507aacb7", //
The ID of this integration.
   region: "au-syd", // The region your integration is hosted in.
   serviceInstanceID: "a57bbab3-1d2e-4714-ae9b-
c3c2c507ef41", // The ID of your service instance.
   onLoad: function(instance) { instance.render(); }
 };


 setTimeout(function()
 {
   const t=document.createElement('script');
   t.src="https://web-
chat.global.assistant.watson.appdomain.cloud/versions/" +
 (window.watsonAssistantChatOptions.clientVersion || 'latest') +
 "/WatsonAssistantChatEntry.js";
   document.head.appendChild(t);
  });

 </script>

</head>

<body class="body">

<div class="on-screen">

 <div class="form">

  <form action="/recover" method="POST">

   <lottie-player
 src="https://assets6.lottiefiles.com/private_files/lf30_GjhcdO.jso
 n"  speed="1"  style="justify-content: center;" loop
 autoplay></lottie-player><br>

   <h4>Forgot your username or password?<br>Don't worry,
 enter your registered email address here, your login credentials
 will be sent to your registered email address.</h4><br>

   <input name="email" type="email" placeholder="&#xf007;
 example@mail.com" required /><br><br>

   <h4>{{msg}}</h4><br>
```

```
        <button style="width:140px" type="submit">SUBMIT</button>

        <button style="width:140px" type="button" onclick="window.location.href='/'">LOGIN</button>

        <p class="message"></p>

      </form>

    </div>

  </div>

  </body>

</html>
```

Login.html

```
   <html>

    <head>

     <title>LOGIN</title>

     <meta name="viewport" content="width=device-width, initial-scale=1.0">
```

```
<meta charset="utf-8">

<link rel="stylesheet" type="text/css" href="{{ url_for('static',
filename='pages.css') }}">

<link rel="stylesheet"
href="https://cdnjs.cloudflare.com/ajax/libs/font-
awesome/4.7.0/css/font-awesome.min.css">

<script src="https://unpkg.com/@lottiefiles/lottie-
player@latest/dist/lottie-player.js"></script>

<link rel="stylesheet"
href="https://use.fontawesome.com/releases/v5.15.1/css/all.css"
integrity="sha384-
vp86vTRFVJgpjF9jiIGPEEqYqlDwgyBgEF109VFjmqGmIY/Y
4HV4d3Gp2irVfcrp" crossorigin="anonymous">

<script>

  window.watsonAssistantChatOptions =
  {
    integrationID: "78eea96f-f7bf-4a38-bd83-3eac507aacb7", //
The ID of this integration.
    region: "au-syd", // The region your integration is hosted in.
    serviceInstanceID: "a57bbab3-1d2e-4714-ae9b-
c3c2c507ef41", // The ID of your service instance.
    onLoad: function(instance) { instance.render(); }
  };

  setTimeout(function()
   {
     const t=document.createElement('script');
     t.src="https://web-chat.global.assistant.watson.appdomain.cloud/versions/" + (window.watsonAssistantChatOptions.clientVersion || 'latest') + "/WatsonAssistantChatEntry.js";
     document.head.appendChild(t);
   });

 </script>

</head>

<body class="body">

<div class="on-screen">

 <div class="form">

   <form action="/login" method="POST">

     <lottie-player src="https://assets2.lottiefiles.com/packages/lf20_ky03n5aXvs.json"  speed="1"  style="justify-content: center;" loop autoplay></lottie-player><br>

     <button style="width:290px" type="button" onclick="window.location.href='/signup'">CREATE AN ACCOUNT</button><br>
```

```
    <button style="width:290px" type="button"
  onclick="window.location.href='/forgot'">FORGOT
  USERNAME/PASSCODE</button><br>

    <input name="username" type="text" placeholder="&#xf007;
  Username" required />

    <input name="password" type="password" id="password"
  placeholder="&#xf023;  Password" required/><br>

    <i class="fas fa-eye" onclick="show()"></i><br>

    <h4>{{msg}}</h4><br>

    <button style="width:135px"
  type="submit">LOGIN</button>

    <p class="message"></p>

   </form>

  </div>

</div>

<script>

  function show()
  {
      var password = document.getElementById("password");
      var icon = document.querySelector(".fas")
      if(password.type === "password")
      {
       password.type = "text";
      }
      else
      {
       password.type = "password";
      }
     };

    </script>

    </body>

</html>
```

Newaccount.html

```
   <html>

    <head>

     <title>NEW ACCOUNT</title>

     <meta name="viewport" content="width=device-width, initial-scale=1.0">
```

```
<meta charset="utf-8">

<link rel="stylesheet" type="text/css" href="{{ url_for('static',
filename='pages.css') }}">
<link rel="stylesheet"
href="https://cdnjs.cloudflare.com/ajax/libs/font-
awesome/4.7.0/css/font-awesome.min.css">

<script src="https://unpkg.com/@lottiefiles/lottie-
player@latest/dist/lottie-player.js"></script>

<link rel="stylesheet"
href="https://use.fontawesome.com/releases/v5.15.1/css/all.css"
integrity="sha384-
vp86vTRFVJgpjF9jiIGPEEqYqlDwgyBgEF109VFjmqGmIY/Y
4HV4d3Gp2irVfcrp" crossorigin="anonymous">

    <script>

  window.watsonAssistantChatOptions =
  {
    integrationID: "78eea96f-f7bf-4a38-bd83-3eac507aacb7", //
The ID of this integration.
    region: "au-syd", // The region your integration is hosted in.
    serviceInstanceID: "a57bbab3-1d2e-4714-ae9b-
c3c2c507ef41", // The ID of your service instance.
    onLoad: function(instance) { instance.render(); }
  };

  setTimeout(function()
    {
      const t=document.createElement('script');
      t.src="https://web-chat.global.assistant.watson.appdomain.cloud/versions/" + (window.watsonAssistantChatOptions.clientVersion || 'latest') + "/WatsonAssistantChatEntry.js";
      document.head.appendChild(t);
    });

  </script>

</head>

<body class="body">

<div class="on-screen">

  <div class="form">

    <form action="/login" method="POST">

      <lottie-player src="https://assets2.lottiefiles.com/private_files/lf30_3ghvm6sn.json"  speed="1"  style="justify-content: center;" loop autoplay></lottie-player><br>

      <center><h4>Your account has been successfully registered and the login credentials were shared with you to the registered email address.</h4><br><center>
      <input name="username" type="text" placeholder="&#xf007; Username" required />

      <input name="password" type="password" id="password" placeholder="&#xf023;  Password" required/><br>

      <i class="fas fa-eye" onclick="show()"></i><br><br>

      <button type="submit">LOGIN</button>

      <p class="message"></p>

    </form>

  </div>

</div>

<script>

  function show()
  {
    var password = document.getElementById("password");
    var icon = document.querySelector(".fas")
    if(password.type === "password")
    {
      password.type = "text";
    }
    else
```

```
           {
             password.type = "password";
           }
         };

       </script>

       </body>

</html>
```

News.html

```
    <html>

       <head>

          <meta charset="UTF-8">

          <title>CAD-NEWSTRACKER</title>

          <link rel="stylesheet" type="text/css" href="{{ url_for('static',
        filename='news.css') }}">

          <link rel="preconnect" href="https://fonts.googleapis.com">

          <link rel="preconnect" href="https://fonts.gstatic.com"
        crossorigin>
```

```
 <script src="https://unpkg.com/@lottiefiles/lottie-
player@latest/dist/lottie-player.js"></script>

  <link
href="https://fonts.googleapis.com/css2?family=Cormorant+Gar
amond:ital,wght@1,700&display=swap" rel="stylesheet">

  <link
href="https://fonts.googleapis.com/css2?family=Crimson+Text
&display=swap" rel="stylesheet">

    <script>

 window.watsonAssistantChatOptions =
 {
   integrationID: "78eea96f-f7bf-4a38-bd83-3eac507aacb7", //
The ID of this integration.
   region: "au-syd", // The region your integration is hosted in.
   serviceInstanceID: "a57bbab3-1d2e-4714-ae9b-
c3c2c507ef41", // The ID of your service instance.
   onLoad: function(instance) { instance.render(); }
 };

 setTimeout(function()
 {
   const t=document.createElement('script');
   t.src="https://web-
chat.global.assistant.watson.appdomain.cloud/versions/" +
(window.watsonAssistantChatOptions.clientVersion || 'latest') +
"/WatsonAssistantChatEntry.js";
   document.head.appendChild(t);
  });
</script>
</head>
<body>
<div class="form">
    <form>
        <br><br>
        <button style="margin-right:-2%;" type="button">CAD-NEWSTRACKER</button>
        <button style="margin-left:40%;" type="button" onclick="window.location.href='/logout'">LOGOUT</button>
        <button type="button" onclick="window.location.href='/dashboard'">DASHBOARD</button>
    </form>
 </div>
 <br><br><br><br><br><br>
```

```
            {% for news, author,publishedAt, desc,img,content,url in context %}

               <br>

               <img style="width: 100%; height: 10px; align-content: center;"src="https://ibm-news-tracker.s3.jp-tok.cloud-object-storage.appdomain.cloud/the_line.png"><br>

               <center><h1>{{news}}</h1></center><br><br>

              <img style="height: 500px; width: 100%" src="{{img}}" alt=""><br>

              <h2>Published Time: {{publishedAt}}</h2>

              <h2>Author: {{author}}</h2><br>

              <h3>Description: {{desc}}</h3><br>

              <h3>Content: {{content}}<a href={{url}} target="blank">Read More...</a></h3><br>

            {% endfor %}

        </body>

</html>
```

Signup.html

```
<html>

  <head>

    <title>SIGN UP</title>

    <meta name="viewport" content="width=device-width, initial-scale=1.0" />

    <meta charset="utf-8" />

    <link rel="stylesheet" type="text/css" href="{{ url_for('static', filename='pages.css') }}" />

    <link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/font-awesome/4.7.0/css/font-awesome.min.css"/>

    <script src="https://unpkg.com/@lottiefiles/lottie-player@latest/dist/lottie-player.js"></script>

    <link rel="stylesheet" href="https://use.fontawesome.com/releases/v5.15.1/css/all.css" integrity="sha384-vp86vTRFVJgpjF9jiIGPEEqYqlDwgyBgEF109VFjmqGmIY/Y4HV4d3Gp2irVfcrp" crossorigin="anonymous">

        <script>
  window.watsonAssistantChatOptions =
  {
    integrationID: "78eea96f-f7bf-4a38-bd83-3eac507aacb7", // The ID of this integration.
    region: "au-syd", // The region your integration is hosted in.
    serviceInstanceID: "a57bbab3-1d2e-4714-ae9b-c3c2c507ef41", // The ID of your service instance.
    onLoad: function(instance) { instance.render(); }
  };

  setTimeout(function()
  {
    const t=document.createElement('script');
    t.src="https://web-chat.global.assistant.watson.appdomain.cloud/versions/" + (window.watsonAssistantChatOptions.clientVersion || 'latest') + "/WatsonAssistantChatEntry.js";
    document.head.appendChild(t);
  });

</script>

</head>

<body class="body">

  <div class="on-screen">

    <div class="form">
    <form action="/register" method="POST">

      <lottie-player style="height: 300px;"
src="https://assets5.lottiefiles.com/packages/lf20_6oomsuir.json
" speed="1" style="justify-content: center" loop autoplay
></lottie-player>

      <input type="text" name="fullname" placeholder="Full
Name" required/>

      <input type="email" name="email" placeholder="Email
address" required/>

      <input type="text" name="username"
placeholder="Username" required/>

      <input type="password" name="password" id="password"
placeholder="Set a password" required/><br>

      <i class="fas fa-eye" onclick="show()"></i><br>

      <h4>{{msg}}</h4><br>

      <button style="width:135px"
type="submit">SUBMIT</button>

      <button style="width:135px" type="button"
onclick="window.location.href='/'">LOGIN</button>

    </form>
```

```
       </div>

      </div>

     </body>

     <script>

      function show()
      {
       var password = document.getElementById("password");
       var icon = document.querySelector(".fas");
       if (password.type === "password")
       {
        password.type = "text";
       }
       else
       {
        password.type = "password";
       }
      }

     </script>

</html>
```

Dashboard.css

```
@import url(https://fonts.googleapis.com/css?family=Roboto:300);

.body
{
  background-image: url('https://ibm-news-tracker.s3.jp-tok.cloud-object-storage.appdomain.cloud/login-signup-backgroud-image.png');
  background-repeat: no-repeat;
  background-size: cover;
}

h4
{
  font-family: Pacifico, cursive;
}

.on-screen
{
  width: 300px;
  padding: 2% 0 0;
  margin: auto;
}

.form
{
  position: absolute;
  border-radius: 25px;
  right:  40px;
  border: 5px solid white;
  z-index: 1;
  background: #48c9b0;
  max-width: 360px;
  margin: 0 auto 80px;
  padding: 20px;
  text-align: center;
  box-shadow: 0 0 20px 0 rgba(0, 0, 0, 0.2), 0 5px 5px 0 rgba(0, 0,
    0, 0.24);
}

.form button
{
  border:0;
  background:transparent;
}

.form img
{
  margin-left: 20px;
  widthwidth: 100px;
  height:100px;
  border-radius: 15px;
}

.form button:hover,.form button:focus
{
  transform: translateY(-4px);
```

```
}

.form button:active
{
  transform: translateY(2px);
}

body
{
  background: #76b852; /* fallback for old browsers */
  background: -webkit-linear-gradient(right, #76b852, #8DC26F);
  background: -moz-linear-gradient(right, #76b852, #8DC26F);
  background: -o-linear-gradient(right, #76b852, #8DC26F);
  background: linear-gradient(to left, #76b852, #8DC26F);
  font-family: "Roboto", sans-serif;
  -webkit-font-smoothing: antialiased;
  -moz-osx-font-smoothing: grayscale;
}

@media(max-width: 1000px)
{
  .body
  {
    background: #48c9b0;
    background-image: none;
    background-repeat: no-repeat;
    background-size: cover;
  }
```

```
h4
{
  font-family: Merriweather, serif;
  font-size: 14px;
}

.on-screen
{
  width: 100%;
  padding: 0 0 0;
  margin: auto;
}

.form
{
  position: absolute;
  border-radius: 0px;
  right:  0px;
  border: 0px solid white;
  z-index: 1;
  background: #48c9b0;
  max-width: 100%;
  margin: 0 auto 0px;
  padding: 20px;
  text-align: center;
  box-shadow: 0 0 0px 0 rgba(0, 0, 0, 0), 0 0px 5px 0 rgba(0, 0, 0, 0);
```

```
       }

       .form button
       {
         border:0;
         background:transparent;
       }

      .container
      {
         max-width: 100%;
       }

       @media(max-width: 459px)
       {
           .container .content .category
         {
           flex-direction: column;
         }

}
```

News.css

```
     h1
     {
       font-family: 'Merriweather', sans-serif;
       font-size: 40px;
```

```
}

h2
{
  font-family: 'Crimson Text', serif;
  font-size: 15px;
}

h3
{
  font-family: 'Cormorant Garamond', serif;
  font-size: 30px;
}

.form
{
  position: fixed;
  border-radius: 0px 0px 50px 50px;
  top: 0px;
  z-index: 0;
  background: #48c9b0;
  width: 100%;
  height:100px;
  padding: 0px;
  text-align: center;
  margin-left:-8px;
}
```

```
.form button
{
  font-family: 'Merriweather', serif;
  font-size: 18px;
  font-weight: bold;
  letter-spacing: .1em;
  outline: 0;
  background: #17a588;
  width: 250px;
  border: 5;
  border-radius:15px;
  margin: 0px 0px 8px;
  padding: 15px;
  color: #FFFFFF;
  -webkit-transition: all 0.3 ease;
  transition: all 0.3 ease;
  cursor: pointer;
  transition: all 0.2s;
}

.form button:hover,.form button:focus
{
  background: #148f77;
  box-shadow: 0 5px 10px rgba(0, 0, 0, 0.2);
  transform: translateY(-4px);
}

.form button:active
```

```
{
  transform: translateY(2px);
  box-shadow: 0 2.5px 5px rgba(0, 0, 0, 0.2);
}

@media(max-width: 1000px)
{
  .form
  {
    position: fixed;
    border-radius: 0px 0px 50px 50px;
    top: 0px;
    z-index: 0;
    background: #48c9b0;
    width: 100%;
    height:250px;
    padding: 0px;
    text-align: center;
    margin-left:-8px;
  }

  .form button
  {
    font-family: 'Merriweather', serif;
    font-size: 25px;
    font-weight: bold;
    letter-spacing: .1em;
    outline: 0;
```

```
      background: #17a588;
      width: 250px;
      height: 90px;
      border: 5;
      border-radius:15px;
      margin: 0px 0px 8px;
      padding: 15px;
      cursorcolor: #FFFFFF;
      -webkit-transition: all 0.3 ease;
      transition: all 0.3 ease;
      cursor: pointer;
      transition: all 0.2s;
    }

    .container
    {
      max-width: 100%;
    }

    @media(max-width: 459px)
    {
    .container .content .category
      {
        flex-direction: column;
      }
}
```

Pages.html

```
@import url(https://fonts.googleapis.com/css?family=Roboto:300);

.body
{
  background-image: url('https://ibm-news-tracker.s3.jp-tok.cloud-object-storage.appdomain.cloud/login-signup-backgroud-image.png');
  background-repeat: no-repeat;
  background-size: cover;
}

h1
{
  font-family: 'Merriweather', sans-serif;
  font-size: 15px;
}

h4
{
  font-family: Pacifico, cursive;
}

.on-screen
{
  width: 300px;
  padding: 2% 0 0;
  margin: auto;
```

```
}

.form
{
  position: absolute;
  border-radius: 25px;
  right:  40px;
  border: 5px solid white;
  z-index: 1;
  background: #48c9b0;
  max-width: 360px;
  margin: 0 auto 80px;
  padding: 20px;
  text-align: center;
  box-shadow: 0 0 20px 0 rgba(0, 0, 0, 0.2), 0 5px 5px 0 rgba(0, 0,
    0, 0.24);
}

.form input
{
  font-family: FontAwesome, "Pacifico", cursive;
  outline: 0;
  background: #f2f2f2;
  width: 80%;
  height: 40px;
  border: 0;
  margin: 0 0 15px;
  padding: 15px;
  box-sizing: border-box;
  font-size: 20px;
  border-radius:10px;
}

.form button
{
  font-family: 'Merriweather', serif;
  font-size: 18px;
  font-weight: bold;
  letter-spacing: .1em;
  outline: 0;
  background: #17a588;
  width: 50%;
  border: 5;
  border-radius:15px;
  margin: 0px 0px 8px;
  padding: 15px;
  color: #FFFFFF;
  -webkit-transition: all 0.3 ease;
  transition: all 0.3 ease;
  cursor: pointer;
  transition: all 0.2s;
}

.form button:hover,.form button:focus
{
  background: #148f77;
  box-shadow: 0 5px 10px rgba(0, 0, 0, 0.2);
  transform: translateY(-4px);
}

.form button:active
{
  transform: translateY(2px);
  box-shadow: 0 2.5px 5px rgba(0, 0, 0, 0.2);
}


body
{
  background: #76b852; /* fallback for old browsers */
  background: -webkit-linear-gradient(right, #76b852, #8DC26F);
  background: -moz-linear-gradient(right, #76b852, #8DC26F);
  background: -o-linear-gradient(right, #76b852, #8DC26F);
  background: linear-gradient(to left, #76b852, #8DC26F);
  font-family: "Roboto", sans-serif;
  -webkit-font-smoothing: antialiased;
  -moz-osx-font-smoothing: grayscale;
}

@media(max-width: 1000px)
{
  .body
  {
    background: #48c9b0;
    background-image: none;
    background-repeat: no-repeat;
    background-size: cover;
  }

  .on-screen
  {
    width: 100%;
    padding: 0 0 0;
    margin: auto;
  }

  .form
  {
    position: absolute;
    border-radius: 0px;
    right:  0px;
    border: 0px solid white;
    z-index: 1;
    background: #48c9b0;
    max-width: 100%;
    margin: 0 auto 0px;
    padding: 20px;
    text-align: center;
    box-shadow: 0 0 0px 0 rgba(0, 0, 0, 0), 0 0px 5px 0 rgba(0, 0, 0, 0);
  }
```

```
.form input
{
  font-family: FontAwesome,'Merriweather', serif;
  outline: 0;
  background: #f2f2f2;
  width: 280px;
  height: 50px;
  border: 0;
  margin: 0 0 15px;
  padding: 15px;
  box-sizing: border-box;
  font-size: 20px;
  border-radius:10px;
}

h4
{
  font-family: Merriweather, serif;
  font-size: 14px;
}

.form button
{
  font-family: 'Merriweather', serif;
  font-size: 18px;
  font-weight: bold;
  letter-spacing: .1em;
  outline: 0;
```

```
        background: #17a588;
        width: 280px;
        border: 5;
        border-radius:15px;
        margin: 0px 0px 8px;
        padding: 15px;
        color: #FFFFFF;
        -webkit-transition: all 0.3 ease;
        transition: all 0.3 ease;
        cursor: pointer;
        transition: all 0.2s;
       }

      .container
      {
        max-width: 100%;
       }

       @media(max-width: 459px)
       {
       .container .content .category
         {
           flex-direction: column;
         }
}
```

App.py

```
from flask import Flask,render_template,request,redirect,url_for,session
from flask_mail import Mail, Message
from newsapi import NewsApiClient
import ibm_db
import re
app=Flask(__name__)
mail = Mail(app)
app.config['MAIL_SERVER']='smtp.gmail.com'
app.config['MAIL_PORT'] = 465
app.config['MAIL_USERNAME'] = '********@gmail.com'
app.config['MAIL_PASSWORD'] = '*******'
app.config['MAIL_USE_TLS'] = False
app.config['MAIL_USE_SSL'] = True
mail = Mail(app)
app.secret_key = 'a'
conn=ibm_db.connect("DATABASE=bludb;HOSTNAME=ba99a9e6-d59e-4883-8fc0-d6a8c9f7a08f.c1ogj3sd0tgtu0lqde00.databases.appdomain.cloud;PORT=31321;SECURITY=SSL;SSLServerCertificate=DigiCertGlobalRootCA.crt;UID=*******;PWD=**********",' ',' ')
global newsresource


@app.route('/')
def home():
    return render_template('login.html')
```

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

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

@app.route('/dashboard')
def dashboard():
        if(session['loggedin']==True):
                return
    render_template('dashboard.html',username=session['fullname'])
        return render_template('login.html')

@app.route('/login',methods=['GET','POST'])
def login():
        global userid
        msg=' '
        if request.method=='POST':
                username = request.form['username']
                password = request.form['password']
                sql = "SELECT * FROM NTAAC 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)
                account = ibm_db.fetch_assoc(stmt)
        if account:
            session['loggedin'] = True
            session['id'] = account['USERNAME']
            userid=  account['USERNAME']
            session['username'] = account['USERNAME']
            session['fullname']=account['FULLNAME']
            session['email'] = account['EMAIL']
            msg = 'Logged in successfully !'
            return
  render_template('dashboard.html',username=session['fullname'])
        else:
            msg = 'Incorrect username / password !'
        return render_template('login.html',msg=msg)

@app.route('/register',methods=['GET','POST'])
def register():
    if request.method=='POST':
        fullname = request.form['fullname']
        username = request.form['username']
        email = request.form['email']
        password = request.form['password']
        queryone = "SELECT * FROM NTAAC WHERE
  username = ? "
        stmtone = ibm_db.prepare(conn,queryone)
        ibm_db.bind_param(stmtone,1,username)
        ibm_db.execute(stmtone)
        userexist = ibm_db.fetch_assoc(stmtone)
        querytwo = "SELECT * FROM NTAAC WHERE email
  = ? "
```

```
        stmtwo = ibm_db.prepare(conn,querytwo)
        ibm_db.bind_param(stmtwo,1,email)
        ibm_db.execute(stmtwo)
        emailexist = ibm_db.fetch_assoc(stmtwo)
        if emailexist:
            msg="Email id already exist!"
        elif userexist:
            msg="Username id already exist!"
        else:
            insert_sql="INSERT INTO NTAAC VALUES(?, ?, ?, ?)"
            prep_stmt=ibm_db.prepare(conn,insert_sql)
            ibm_db.bind_param(prep_stmt,1,fullname)
            ibm_db.bind_param(prep_stmt,2,email)
            ibm_db.bind_param(prep_stmt,3,username)
            ibm_db.bind_param(prep_stmt,4,password)
            ibm_db.execute(prep_stmt)
            msg = Message('CAD-NEWSTRACKER Account Created Successfully',sender ='cad.newstracker@gmail.com',recipients = [email])
            msg.body = ("Welcome to CAD-NEWSTRACKER, your account was successfully registered with us."+"\n\n\n"+"Your login credentials are:"+"\n\n"+"Username: "+username+"\n"+"Password: "+password+"\n\n\n"+"Happy Reading........"+"\n\n\n\n"+"With regard,"+"\n"+"CAD-NEWSTRACKER")
            mail.send(msg)
            return render_template('newaccount.html')
        return render_template('signup.html',msg=msg)
```

```
@app.route('/recover',methods=['GET','POST'])
def recover():
    if request.method=='POST':
        email = request.form['email']
        query = "SELECT * FROM NTAAC WHERE email = ?
  "
        stmt = ibm_db.prepare(conn,query)
        ibm_db.bind_param(stmt,1,email)
        ibm_db.execute(stmt)
        emailexist = ibm_db.fetch_assoc(stmt)
        if emailexist:
            queryone=("SELECT * FROM NTAAC WHERE
  EMAIL =?")
            stmtone = ibm_db.prepare(conn,queryone)
            ibm_db.bind_param(stmtone,1,email)
            ibm_db.execute(stmtone)
            credentials = ibm_db.fetch_assoc(stmtone)
            username=str(credentials['USERNAME'])
            password=str(credentials['PASSWORD'])
            msg = Message('CAD-NEWSTRACKER Login
  Credentials',sender ='cad.newstracker@gmail.com',recipients =
  [email])
            msg.body = ("Request for sending your login
  credentials was completed successfully."+"\n\n\n"+"Your login
  credentials are:"+"\n\n"+"Username:
  "+username+"\n"+"Password: "+password+"\n\n\n"+"Happy
  Reading........"+"\n\n\n\n"+"With regard,"+"\n"+"CAD-
  NEWSTRACKER")
            mail.send(msg)
            msg="Login credentials sent to your mail
  succcessfully....."
```

```
            email='   '
            return render_template('login.html',msg=msg)
        else:
            msg="Email id not found."
            return render_template('forgot.html',msg=msg)

@app.route('/news',methods=['GET','POST'])
def news():
    newsapi = NewsApiClient(api_key="*************")
    if request.method == 'POST':
        if(request.form['newsresource']=="google"):
            newsresource="google-news-in"
            msg="GOOGLE NEWS"
        elif(request.form['newsresource']=="bbc"):
            newsresource="bbc-news"
            msg="BBC NEWS"
        elif(request.form['newsresource']=="toi"):
            newsresource="the-times-of-india"
            msg="Times of India"
        elif(request.form['newsresource']=="abc"):
            newsresource="abc-news"
            msg="ABC NEWS"
    topheadlines =
  newsapi.get_top_headlines(sources=newsresource)
    articles = topheadlines['articles']
    news = []
    author = []
    publishedAt=[]
```

```
     desc=[]
     img = []
     content=[]
     url=[]
     for i in range(len(articles)):
         myarticles = articles[i]
         news.append(myarticles['title'])
         author.append(myarticles['author'])
         publishedAt.append(myarticles['publishedAt'])
         desc.append(myarticles['description'])
         img.append(myarticles['urlToImage'])
         content.append(myarticles['content'])
         url.append(myarticles['url'])
         mylist = zip(news, author,publishedAt,
   desc,img,content,url)
     return render_template('news.html', context = mylist)

 @app.route('/logout')
 def logout():
     session['loggedin']=False
     session.pop('id', None)
     session.pop('username', None)
     msg="Logged out successfully......"
     return render_template('login.html',msg=msg)

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

deployment.yaml

```
apiVersion: apps/v1
 kind: Deployment
 metadata:
   name: cad-newstracker-deployment
 spec:
   replicas: 1
   selector:
     matchLabels:
       app: newstracker
   template:
     metadata:
       labels:
         app: newstracker
     spec:
       containers:
       - name: newstracker
         image: us.icr.io/newstracker/cad
         imagePullPolicy: Always
         ports:
   - containerPort: 5000
```

Service.yaml

```
 apiVersion: v1
  kind: Service
  metadata:
    name: cad-newstracker-deployment
  spec:
    ports:
    - port: 5000
      targetPort: 5000
    selector:
app: newstracker
```

GitHub & Project Demo Link

Github
https://github.com/IBM-EPBL/IBM-Project-36997-1660299482

Project demo link
https://youtu.be/UAo3nHATujk