Project report

IBM

News Tracker Application

TEAM ID: [PNT2022TMID32908]

IBM-Project-37913-1660364659

Project By

JUHAIF AHAMED.H - 820419104027

BALAJI M   - 820419104012
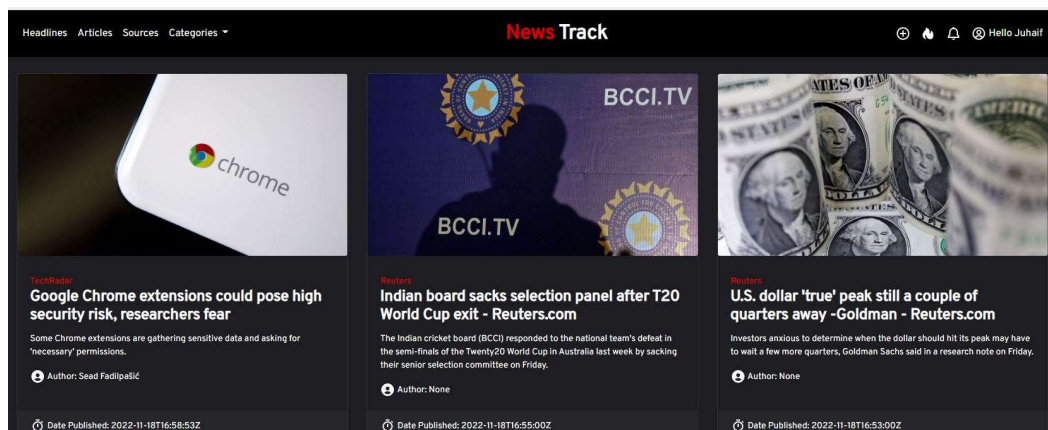
HARISH B   - 820419104024

DINESH KUMAR U - 820419104020

7th semester

Department of Computer Science and
Engineering
Anjalai Ammal Mahalingam
Engineering college

1. **INTRODUCTION**
   1. Project Overview
   2. Purpose
2. **LITERATURE SURVEY**
   1. Existing problem
   2. References
   3. Problem Statement Definition
3. **IDEATION & PROPOSED SOLUTION**
   1. Empathy Map Canvas
   2. Ideation & Brainstorming
   3. Proposed Solution
   4. Problem Solution fit
4. **REQUIREMENT ANALYSIS**
   1. Functional requirement
   2. Non-Functional requirements
5. **PROJECT DESIGN**
   1. Data Flow Diagrams
   2. Solution & Technical Architecture
   3. User Stories
6. **PROJECT PLANNING & SCHEDULING**
   1. Sprint Planning & Estimation
   2. Sprint Delivery Schedule
   3. Reports from JIRA
7. **CODING & SOLUTIONING (Explain the features added in the project along with code)**
   1. Feature 1
   2. Feature 2
8. **TESTING**
   1. Test Cases
   2. User Acceptance Testing
9. **ADVANTAGES & DISADVANTAGES**
10. **CONCLUSION**
11. **FUTURE SCOPE**
12. **APPENDIX**
    Source Code
    GitHub & Project Demo Link

1. Introduction

1.1 Project Overview

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.

1.2 Purpose

Newspapers are one of the most popular and most needed commodities in our daily life. In today's busy world, reading newspapers has become one of the traditional ways of getting news. News is produced every minute and distributed via television, radio and the Internet, so the news updated the next morning is already outdated. So newspaper and magazine publishers have a hard time keeping up with the pace. Change is needed and publishers must embrace mobile.

2. LITRATURE SURVEY

2.1 Existing problem

We've collected all the relevant negative factors of lack of news tracking statistics, along with results from studies and reports that have analyzed the issue.This guide will give you an unbiased look at why the media reports negative news. We'll provide you with an informed and educated overview of the subject in general.

2.2 Reference

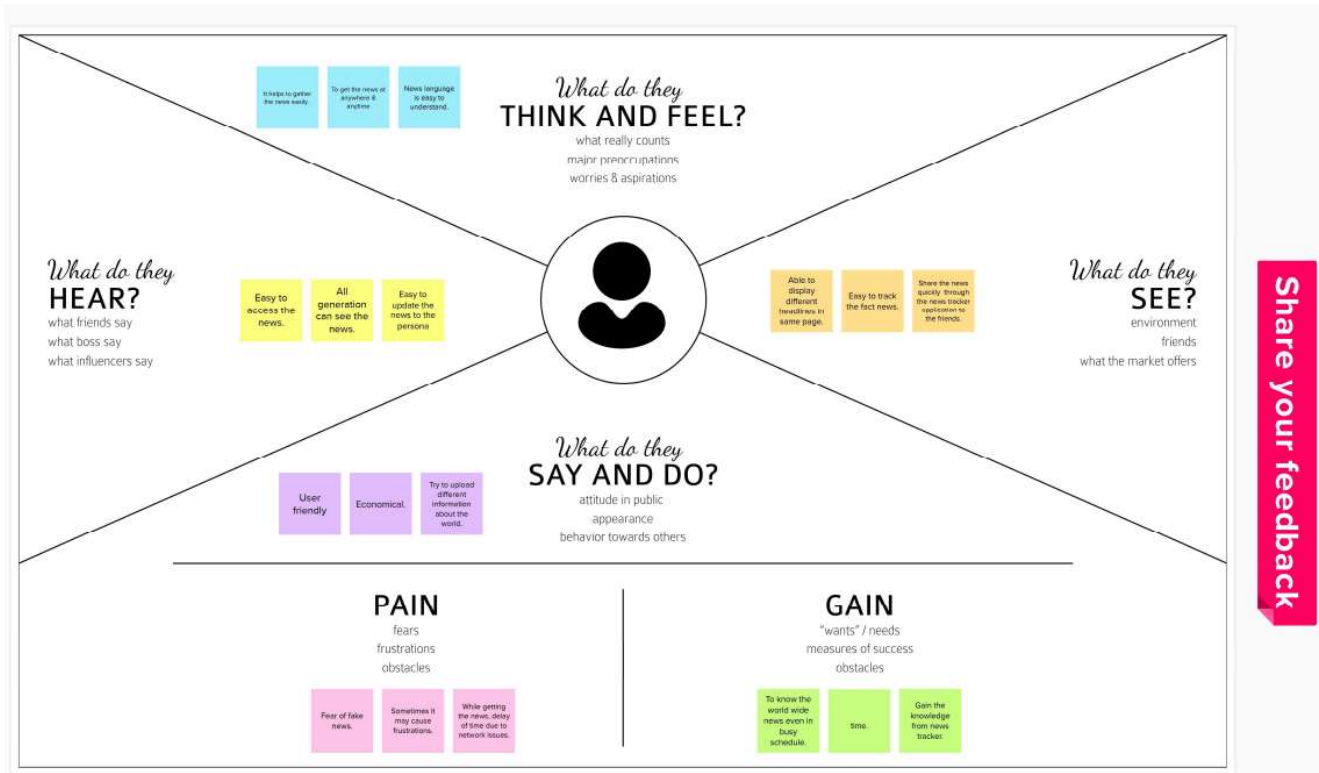| S.No	Title	Author(s)	Month /Year	Method/Implementa tion technique(s)	Resource Link
1	Exploring mobile news reading interactions for news app personalisation	Marios Constantinides, John Dowell, David Johson, Sylvain Malacria	August, 2015	1. Identifcation of news reader types 2. Interaction logging and classifcation study 3. Deployment and data collection 4. Predicting News reader types 5. Adaptive UI	(PDF) Exploring mobile news reading interactions for news app personalisation (researchgate.net)
2	Detection and Tracking in News Articles	Sagar Patel, Sanket Suthar, Sandip Patel, Neha Patel	March, 2015	1. Pre-processing 2. Tokenization 3. Stemming/L emmization 4. Vector Space Model 5. Topic tracking	(PDF) Topic Detection and Tracking in News Articles (researchgate.net)
3	Following the Fed with a News Tracker	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 come in stronger, weaker or as expected. This is based on the CitiGroup U.S Economic Surprise Index.	(PDF) Following the Fed with a News Tracker (researchgate.net)

4	An End-to-end Weakly-supervised News Aggregation Framework	Xijin Tang, Xiaohui Huang	June, 2022	The framework combines Snorkel-based weakly-supervised classifcation, Latent Dirichlet Allocation (LDA) topic modeling, and topic signal detection model to classify and aggregate unlabeled news texts and ultimately generate visualized results containing news categories, news topics, and temporal topic relationships. This paper uses constructed knowledge thesaurus and the Snorkel method to weakly supervise the classifcation of unlabeled news with no manual tagging. Subsequently, we utilize LDA to generate the topics and obtain the signal value of each topic based on the topic signal detection function. Finally, we establish the temporal topic relationships and get the visualized results of news aggregation.	An End-to-end Weakly-supervised News Aggregation Framework | Request PDF (researchgate.net

2.3 Problem Statement Definition

Problem Statement (PS)	I am (Customer)	I'm trying to	But	Because	Which makes me feel
PS-1	Common People	To get the news anywhere and anytime	It Shows fake and irrelevant news	It Didn't able to access correct NEWS Platform Resources	Tension
PS-2	Businessman	To collect world wide business information	It didn't able to access correct news	It didn't able to filter and notify important Bussiness NEWS	Frustrated
PS-3	Local People	To access local news instantly	It didn't access the local news	Because of that most of the native peoples didn't share the information in news platform app	Unawareness
PS-4	Foreign People	To access local news as well as global news in their language	But This option is not available	They didn't understand the news from that platform	Confussed
PS-5	Common People	The News article phases are very easily understandable by using often usable words	But Many News headlines and articles are used rare verbal phases	They didn't understand the news clearly from that news app	Confussed

3.IDEATION & PROPOSED SOLUTION

3.1   Empathy Map Canvas

3.2 Ideation And Brain Storming

2

Brainstorm

Write down any ideas that come to mind that address your problem statement.

🕐 10 minutes

Balaji

- Important NEWS can be notified
- Translate the NEWS into their Language
- Save later option for later reading
- Recommends news by their location
- Option that to read all contents on the screen due to display over the news
- Personal Preference NEWS can be notified

Juhaif Ahamed

- Chatbot
- Feedback
- Comment and Share NEWS to others
- User can customise their topics according to their daily interest
- Viewing , hearing enhancements-integrated videos with NEWS
- Mobile app accessible

Dinesh Kumar

- News in 60 words
- Detection of fake news
- Read Offline
- Deliverance of NEWS at right time
- Finding of genuine NEWS
- Personalization option for Reading-light mode,dark mode,Font size

Harish

- Fetches short videos with NEWS articles
- History Option for viewed news.
- Gathered the Trusted NEWS Platforms from all around the world
- Manage Trusted NEWS Platform
- Flash NEWS
- Integrate with Social Media Post into the NEWS article

Juhaif Ahamed

- Once a user has opted in to receive push notifications, they can receive updates for breaking news and essential stories, immediately, operating as a 24-hour news service directly to a user's device.
- Reduce spam notification
- User interface is intractive and Good Looking
- Collection - Favorites, bookmarked pages
- Top stories across all your favorite topics and sources in one place.
- News presented through Interactive and Engaging layouts

3

Group ideas

Take turns sharing your ideas while clustering similar or related notes as you go. In the last 10 minutes, give each cluster a sentence-like label. If a cluster is bigger than six sticky notes, try and see if you and break it up into smaller sub-groups.

🕐 20 minutes

CUSTOMIZATION

- Personalization option for Reading-light mode,dark mode,Font size
- User interface is intractive and Good Looking
- Translate the NEWS into their Language
- Feedback
- Comment and Share NEWS to others
- Save later option for later reading
- History Option for viewed news.
- Collection - Favorites, bookmarked pages
- News presented through Interactive and Engaging layouts

INNOVATION IDEAS

- Chatbot
- News in 60 words
- Detection of fake news
- Recommends news by their location
- Fetches short videos with NEWS articles
- Reduce spam notification
- Integrate with Social Media Post into the NEWS article
- Option that to read all contents on the screen due to display over the news
- Viewing , hearing enhancements-integrated videos with NEWS
- Even public can upload their native news. If their own posted news is trusted and true so they news will promoted on that local area

NECCESSARY IDEAS

- Important NEWS can be notified
- User can customise their topics according to their daily interest
- Gathered the trusted NEWS Platforms from all around the world
- Manage Trusted NEWS Platform
- Staying on top of the trends
- Top stories across all your favorite topics and sources in one place.
- Finding of genuine NEWS
- Read Offline
- Personalization option for Reading-light mode,dark mode,Font size
- Once a user has opted in to receive push notifications, they can receive updates for breaking news and essential stories, immediately, operating as a 24-hour news service directly to a user's device.

News Tracker Application

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

If each of these tasks could get done without any difficulty or cost, which would have the most positive impact?

Feasibility

Regardless of their importance, which tasks are more feasible than others? (Cost, time, effort, complexity, etc.)

TIP

Participants can use their cursors to point at where sticky notes should go on the grid. The facilitator can
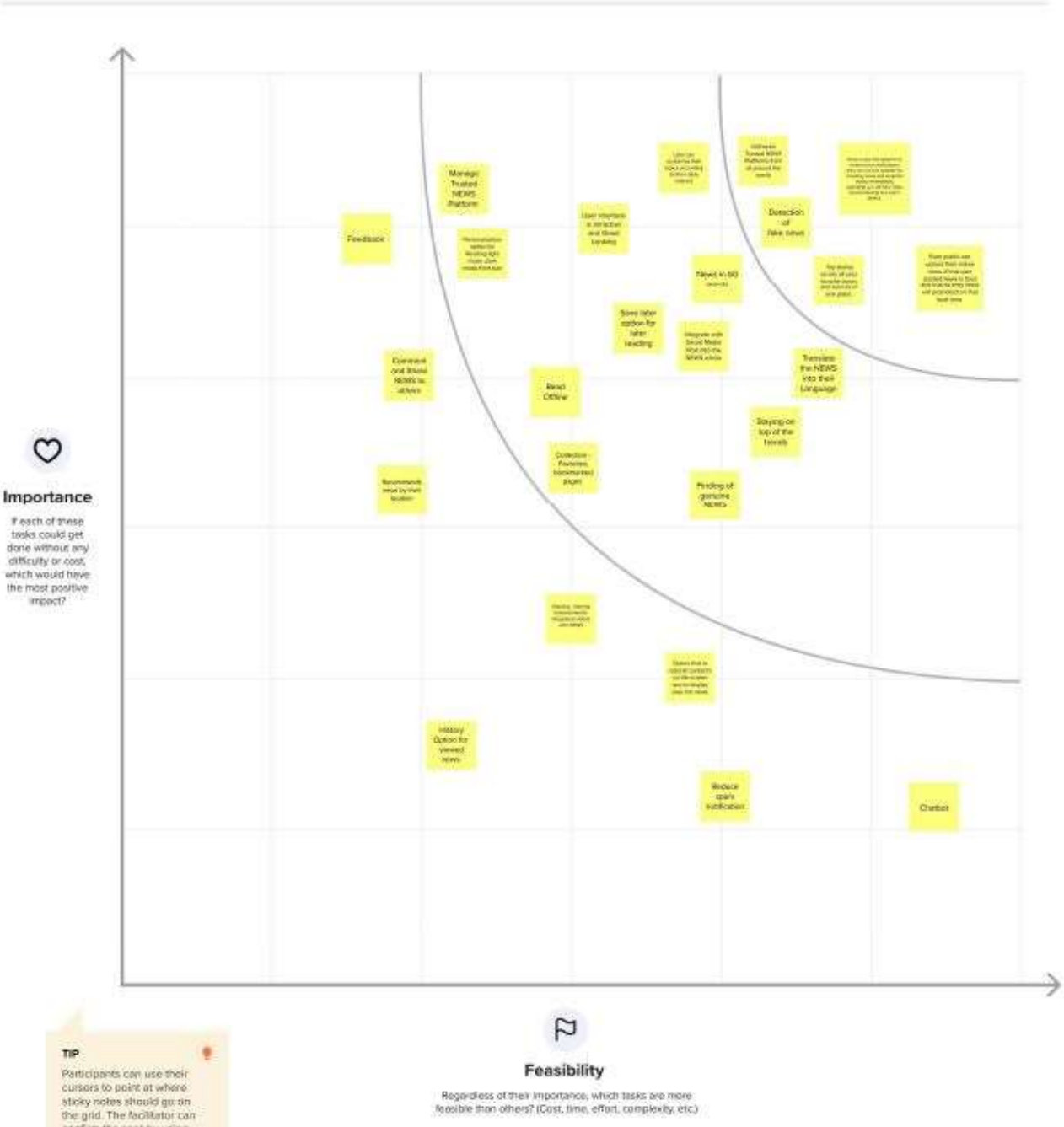
3.3 Proposed Solution:

Proposed Solution Template:

Project team shall fill the following information in proposed solution template.

S.No.	Parameter	Description
1.	Problem Statement (Problem to be solved)	User needs a way to get a interested marketing in a quick peek of the day. So that we give relevant news where the user more interested on that day and save their time.
2.	Idea / Solution description	The app should include all of the trustworthy sources from across the world and then specify in each article which source has validated this news because I'm assuming that customers can't discern the difference between real and fake news.
3.	Novelty / Uniqueness	• Saving and downloading the interested articles • Access Both Local and International News instantly • Even Public can Upload their native news, if that user posted news is trust and true their news will promoted on that local area category • Translate the NEWS articles in their native language • User can access and customize their topics by their Interest then that News should be notify 24/7
4.	Social Impact / Customer Satisfaction	When news feed are customized/personalized to the user, the time spend to learn about the day will be minimized and will be completely productive. .For customer Satisfaction ,we Track all kind of news .it will be more Useful to get knowledge in Today's World.
5.	Business Model (Revenue Model)	Our First and most important Business is to deliver a good content or good news to the people apart from that We are going to promote advertisement about good products.

3.4 Problem Solution Fit

1. CUSTOMER SEGMENT(S) CS

Who is your customer?
i.e. working parents of 0-5 y.o. kids

News tracker app can be used by all age groups there is no age limitation
This app will be helpful to school students to older age groups in knowing the proper and real news updates regularly

6. CUSTOMER CONSTRAINTS 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.

News tracker app can be used offline
It will consume More time
It is budget friendly to use for all age groups

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

People may use either newspaper or social media or youtube channels to know the news. The internet and the intermingling of social media with important worldwide events has made it almost impossible to live under a rock. But finding a reliable one-stop shop to engage with your news can be somewhat of a challenge.

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.

Personalization is not flexible and reliable for user. The problems and pains of the user are 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 news feed 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.

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.

The main root cause of NEWS traker application is to gather the NEWS from Non authorized NEWS Medium or website that cause to spread fake news to people

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)

while designing this app,we have already pre-built the "chat with us" option (or)"Feedback"option.so this helps our customer to report their problem with us easily,so that we get sorted out the problem easily.

3. TRIGGERS TR

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

Desire to update their knowledge by knowing about what goes on around them,at the Local,National and International Levels.By seeing about their intrest like polities,sports etc.

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

User can tensed to know the complete information of the particular news and can get boredom to get the old issues of printed newspaper so they prefer news website to read an more ellorate news and get old news in their fingertip

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

This app helps you to query for allinformation about Indices Commodities, Currencies, FutureRates, Bonds,ete as on official websites.
It helps the user to read thenews in their own national or international language.
we create a scalable, responsive and user-friendly newstracker application.
It fetches different news sources all around the world so user can select different countries news in which they are interested.

8. CHANNELS of BEHAVIOUR CH

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

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

Online:The users searching more information online by means of many websites which has to be provided in the app .
Offline:The user downloads many NEWS articles whose information has to be stores in our app that can be access by user

4.REQUIREMENT ANALYSIS
4.1 Functional requirement

Functional Requirements:

Following are the functional requirements of the proposed solution.

| FR No.	Functional Requirement (Epic)	Sub Requirement (Story / Sub-Task)
FR-1	User Installation	User can Browse the app from the website
FR-2	User Registration	Registration through Form Registration through Gmail
FR-3	User Confirmation	Confirmation via Email Confirmation via OTP
FR-4	User Login	User should login the app with the user's name and password

4.2 Non-Functional requirements

Non-functional Requirements:

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

FR No.	Non-Functional Requirement	Description
NFR-1	Usability	Everyone can understand the process of using the app easily by the commands given in the app.
NFR-2	Security	This can be access only by the authorized API. so Security can be improved and No fake news can be shared.
NFR-3	Performance	The updation of trending news occurs without any interruption. So, it performance is good
NFR-4	Availability	This application will be available to the all the user who are using this application

5 PROJECT DESIGN

5.1 Data Flow Diagrams
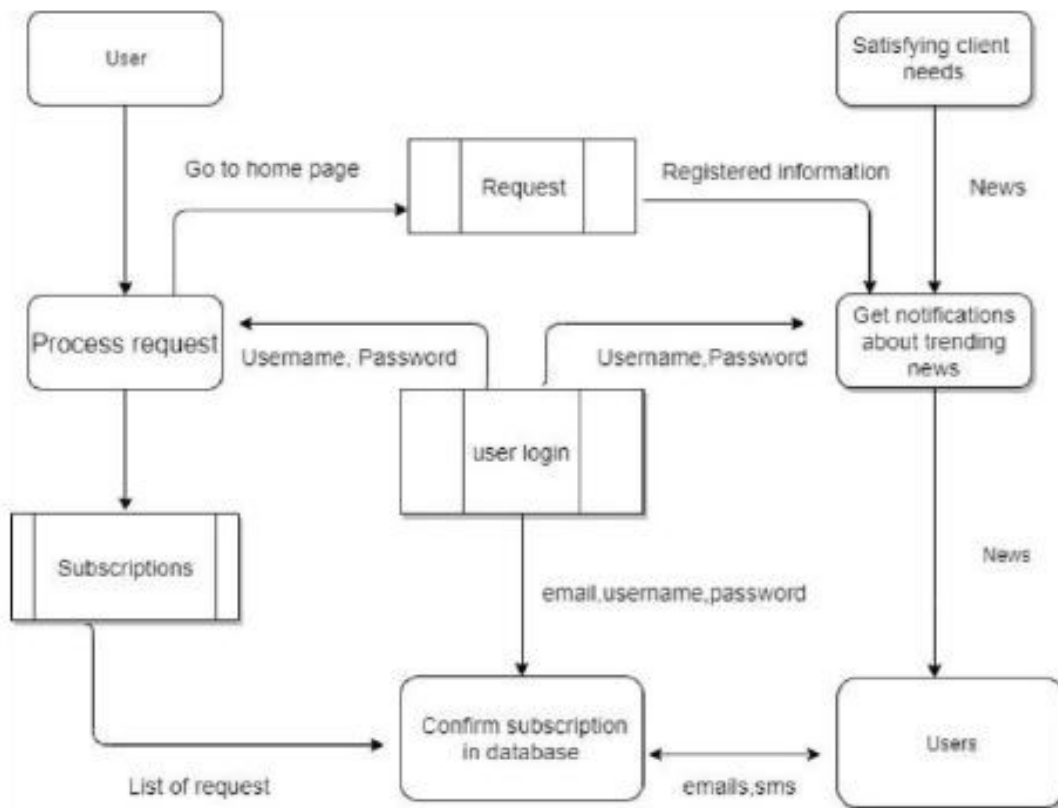
5.2 Solution &Technical Architecture

Table 1: Components & Technologies:

S.No	Component	Description	Technology
1.	User Interface	The user can interact with the application toknow about the trending news	HTML,CSS, JavaScript/ Angular Js/ ReactJs etc.
2.	Application Logic-1	The application contains this resource gives you basic understanding of Flask	Flask
3.	Application Logic-2	The application contains the news sub-division like geographical news, economicnews and society news	IBM Watson STT service
4.	Application Logic-3	The user can view the growth of the economy in industry through graph	IBM Watson Assistant
5.	Database	Updation of trending news are stored in the MySQL database	MySQL, NoSQL, etc.
6.	Cloud Database	With the use of cloud, media coverage issue cannot be occurred	IBM DB2, IBM Cloudant etc.

Table 2: Application Characteristics:

S.No	Characteristics	Description	Technology
1.	Open-Source Frameworks	Flask is flexible and doesn't require touse any particular project or code layout used in this application	Python-Flask
2.	Security Implementations	This can be access only by the journalist. So, It is a high Security	Container Registry, Kubernetes Cluster.
3.	Scalable Architecture	News Tracker is a socio-economic access because helps to know aboutthe daily activity of the world	Container Registry, Kubernetes Cluster.
4.	Availability	This application will be available to the all the user who are using this application	Container Registry, Kubernetes Cluster.
5.	Performance	The updation of trending news occurs without any interruption. So,it performance is good	Kubernetes Cluster.

5.3 User Stories

User Stories

User Type	Functional Requirement (Epic)	User Story Number	User Story / Task	Acceptance criteria	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
	Login	USN-3	As a user, I can log into the application by entering email & password	I can access the Dashboard and the application.	High	Sprint-1
Customer (Web user)	Upload Photo	USN-4	As a user, I can upload the food photo.	I can get the nutrition details.	High	Sprint-1
Administrator	User details	USN-5	As a user, I can fill the Details.	I can get whether the scanned food is suitable or not.	High	Sprint-2
	Push notification	USN-6	As a user, I will search the food items.	I can get the notification, related to my search.	High	Sprint-3
	Shown the nutrition details	USN-7	As a user, I can scan the food.	I can get the nutrition details of the scanned food.	High	Sprint-4
	Receipe shown	USN-8	As a user,I want to get the receipe for the scanned food.	I can get the receipe about the food.	Low	Sprint-4

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

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	Registration(Admin and Customer)	USN-1	As a user, I can register for the application by entering my email, password, and confirming my password.	3	High	Juhaif Ahamed
Sprint-1		USN-2	As a user, I will receive confirmation email once I have registered for the application	5	High	Dinesh Kumar
Sprint-1		USN-3	As a user, I can register for the application through Facebook	1	Low	Harish
Sprint-1		USN-4	As a user, I can register for the application through Gmail	2	Medium	Balaji
Sprint-1	Login (Admin and Customer)	USN-5	As a user, I can log into the application by entering email & password	1	High	Juhaif Ahamed
Sprint-2	Dashboard(Admin and Customer)	USN-6	As a user I should be able to navigate and access all the features hassle free	5	High	Juhaif,Balaji
Sprint-2	Layout	USN-7	As a user I should be able to access the portal with different devices with the same comfort	3	High	Dinesh,Harish
Sprint-3	Data Store,Retrieval and Authentication	USN-8	Get Data from API and store as JSON in DB2	5	High	Balaji,Dinesh
Sprint-3		USN-9	Get bin data from API and store in DFS	3	High	Juhaif,Harish
Sprint-3	Local News Dashboard	USN-10	Create a Option of post and authorize the news by User's location	2	High	Juhaif,Balaji
Sprint-4	User Segregation	USN-11	As a CC executive I should be able to	3	Low	Harish

Sprint	Functional Requirement (Epic)	User Story Number	User Story / Task	Story Points	Priority	Team Members
	and data access		uniquely identify the customer and offer help			
Sprint-4	Change code	USN-12	As a administrator I should be able to modify code according to the future requirements.	2	Medium	Dinesh
Sprint-4	Monitor the system And Testing	USN-13	As a administrator I should be able to monitor the cloud system and fix errors before customer.	1	High	Balaji
Sprint-4	Depolyment with Docker	USN-14	As a User,I will deploy the entire Application using Docker.	2	Medium	Juhaif
Sprint-4	Orchest with Kubernates	USN-15	As a User,I will allocate the server nodes and balance the work loads in server.	2	Medium	Juhaif

6.2 Sprint Delivery Schedule

Project Tracker, Velocity & Burndown 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	12	6 Days	24 Oct 2022	29 Oct 2022	12	29 Oct 2022
Sprint-2	8	6 Days	31 Oct 2022	05 Nov 2022	8	05 Nov 2022
Sprint-3	10	6 Days	07 Nov 2022	12 Nov 2022	10	12 Nov 2022
Sprint-4	10	6 Days	14 Nov 2022	19 Nov 2022	10	19 Nov 2022

Velocity:

Imagine we have a 10-day sprint duration, and the velocity of the team is 20 (points per sprint). Let's calculate the team's average velocity (AV) per iteration unit (story points per day)
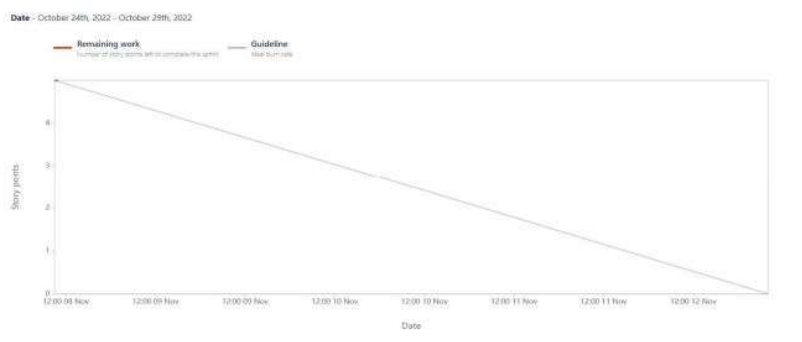
Average Velocity of Sprint-1 = 12/6 =2.0
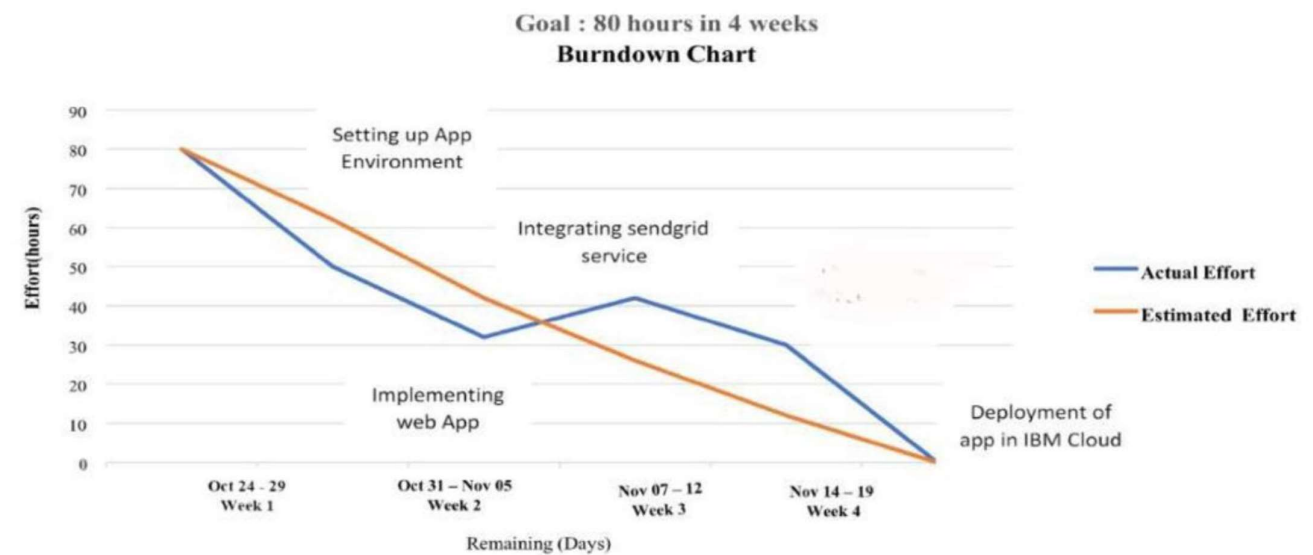
Average Velocity of Sprint-2 = 8/6 =1.3

Average Velocity of Sprint-3 = 10/6 =1.6

Average Velocity of Sprint-4 = 10/6 =1.6

Burndown Chart:

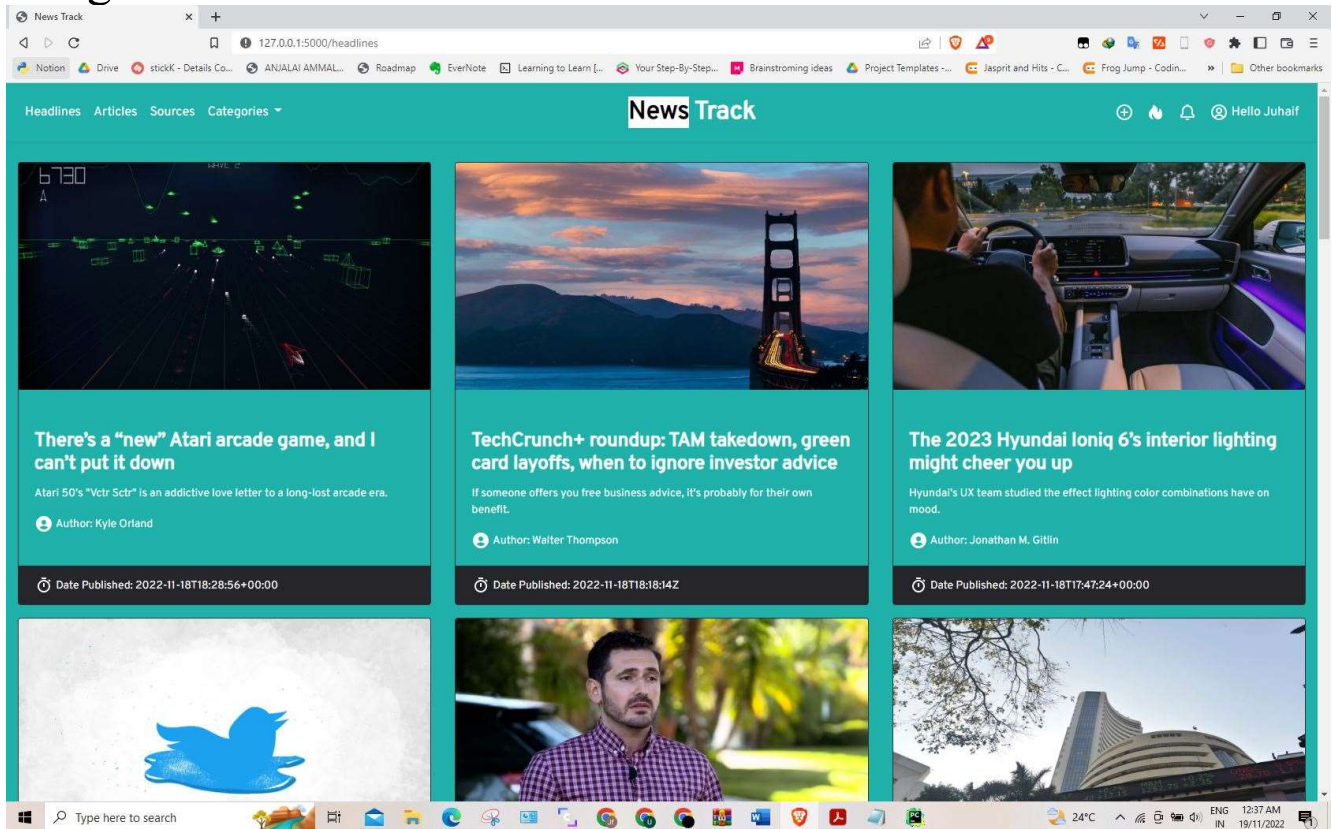# 6.3 Reports from JIRA

Goal : 80 hours in 4 weeks
Burndown Chart

Setting up App Environment

Integrating sendgrid service

Implementing web App

Deployment of app in IBM Cloud

Actual Effort

Estimated Effort

Effort(hours)

Remaining (Days)

Oct 24 - 29 Week 1

Oct 31 – Nov 05 Week 2

Nov 07 – 12 Week 3

Nov 14 – 19 Week 4

7.CODING & SOLUTION

7.1 Feature 1

Access All Local and International News Around the word by categories

News Tracker Application

Screenshot 1 — Business Category

News Track — Headlines · Articles · Sources · Categories — Hello Juhaif

127.0.0.1:5000/category/business

Oil prices slide 3% as supply fears recede - Reuters.com
Oil dropped by more than $2 a barrel on Friday, on track for a second weekly decline, due to concern about weakened demand in China and further increases to U.S. interest rates.
Author: None
Date Published: 2022-11-18T18:39:00Z

Gravitas: FDA approves life-changing therapy for Diabetes - WION
The FDA has approved a life-changing therapy for Diabetes. It says an antibody called Teplizumab can delay the onset of Type 1 Diabetes. Priyanka Sharma repo...
Author: None
Date Published: 2022-11-18T18:33:58Z

Indonesia's economy could benefit from higher commodity prices, says economist - CNBC International TV
Anthony Nafte of CLSA says Indonesia's commodity-driven economy could do "reasonably well" as a disruption of coal supply from Russia would lead to China buy...
Author: None
Date Published: 2022-11-18T18:30:02Z

https://www.youtube.com/watch?v=J4mm0o4e-LE

Screenshot 2 — Sports Category

News Track — Headlines · Articles · Sources · Categories — Hello Juhaif

127.0.0.1:5000/category/sports

Titans offensive coordinator Todd Downing charged with DUI, speeding hours after win over Packers - CBS Sports
Downing was pulled over in Williamson County in Tennessee on Friday morning
Author:
Date Published: 2022-11-18T18:33:00Z

India vs New Zealand Series Preview | All Eyes on Hardik Pandya | Cricket Canvas - Hindustan Times
Crashing out of the 2022 T20 WC Semi-finals, both India and New Zealand look to regroup before the upcoming ODI WC and the next T20 WC. India steps in with a...
Author: None
Date Published: 2022-11-18T18:22:54Z

Ireland chasing southern-hemisphere hat-trick against Australia - The Guardian
Ireland will complete a clean sweep of southern-hemisphere giants in a calendar year if they beat Australia in Dublin
Author: Michael Aylwin
Date Published: 2022-11-18T18:07:00Z

7.2 Feature 2

Trusted News Sources only accessed by using NewsAPI and We can View some Trending Articles from Popular Websites like as Verge,Forbes etc
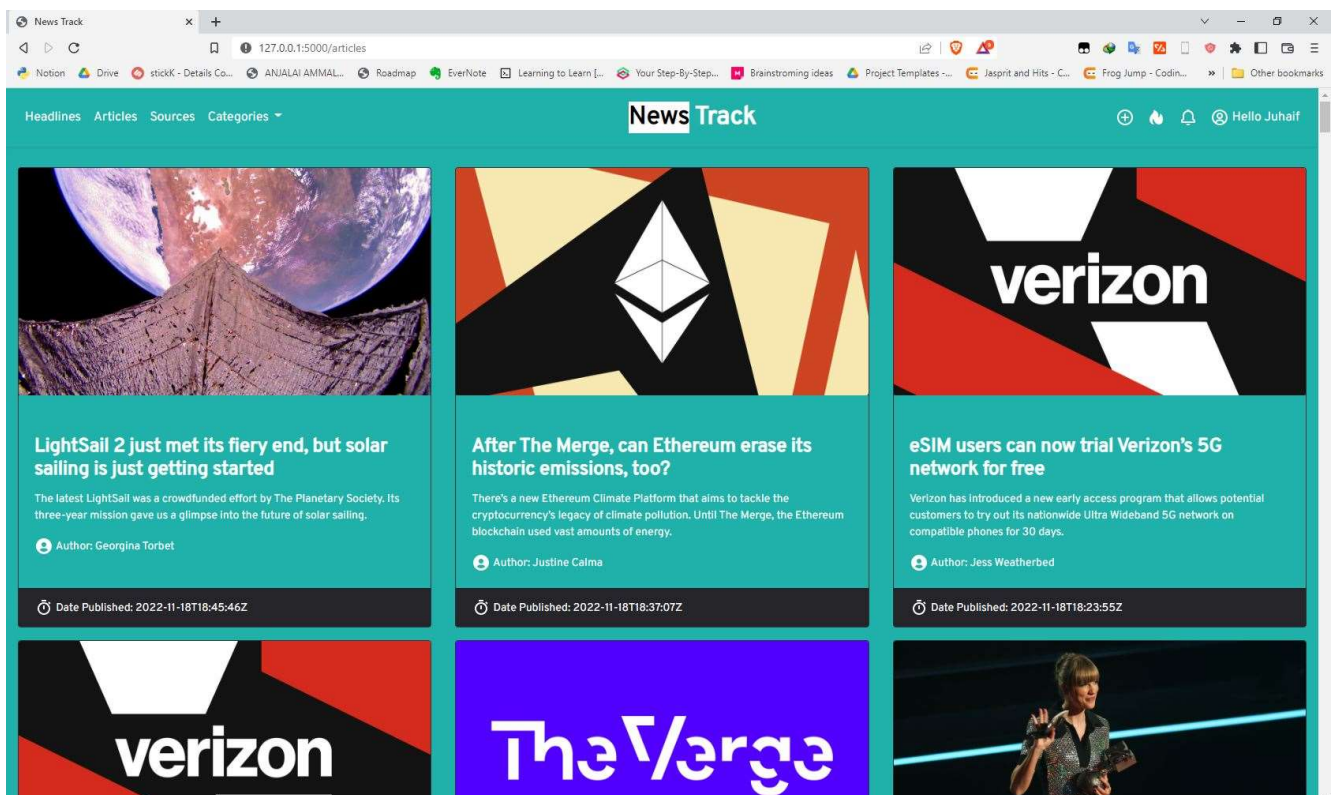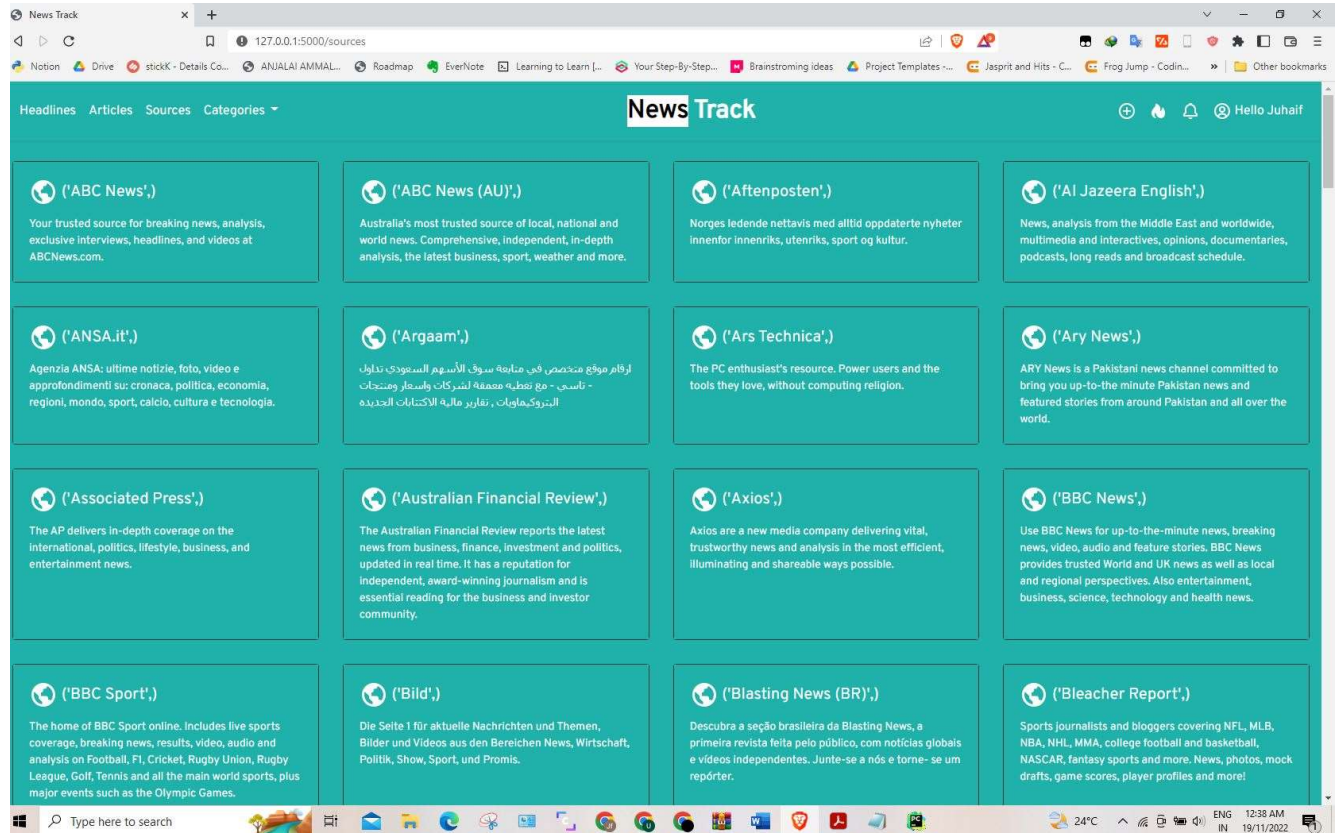
8. TESTING

8.1 TEST CASE

Date	18-Nov-22												
Team ID	PNT2022TMID32908												
Project Name			News Tracker Application										
Maximum Marks			4 marks										

Test case ID	Feature Type	Component	Test Scenario	Pre-Requisite	Steps To Execute	Test Data	Expected Result	Actual Result	Status	Commnets	TC for Automation(Y/N)	BUG ID	Executed By
1	Functional	Login Page	Verify user is able to Login into the Application		1) Open the News tracker application. 2) Login with user Credentials 3) Verify logged in to user account	Email: jfad@gmail.com Password: 1234	Login Successful	Working as expected	Pass		N		Juhaif Ahamed
2	Functional	Signup Page	Verify user is able to Signup in the Application		1) Open the news tracker 2) Enter the Details and Create a new User 3) Verify if user is created and inserted into DB Table	Email: jfadad@gmail.com Password: 1234	Account Created Successfully	Working as expected	Pass		N		Juhaif Ahamed
3	Functional	Dashboard page	Verify if all the user details are stored in Database		1) Open the News tracker application. 2) Enter the Details and Create a new User 3) Verify if user is created and inserted into DB Table	Username: jfadad@gmail.com password: 1234	User should navigate to user account homepage	Working as expected	Pass				Balaji
4	Functional	Login page	Verify user is able to log into application with InValid credentials		1.Enter URL and click go 2.Click on Sign IN button 3.Enter InValid username/email in Email text box 4.Enter valid password in password text box 5.Click on login button	Username:balaji@gmail.com password: 592001	Application should show 'Incorrect email or password ' validation message.	Working as expected	Pass				Balaji
5	Functional	Login page	Verify user is able to log into application with InValid credentials		1.Enter URL and click go 2.Click on Sign IN button 3.Enter InValid username/email in Email text box 4.Enter valid password in password text box 5.Click on login button	Username:harish@gmail.com password:32002	Application should show 'Incorrect email or password ' validation message.	Working as expected	Pass				Harish

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	3	1	2	1	7
Duplicate	1	0	0	0	1
External	1	0	0	1	2
Fixed	2	1	1	1	5
Not Reproduced	0	0	0	0	1
Skipped	0	0	0	0	0
Won't Fix	0	0	0	0	0
Totals	7	2	3	3	16

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
Login Page	4	0	0	4
Registration Page	1	0	0	1
Home Page	2	0	0	2

9. ADVANTAGES:
1. Viewers can get their news straight off their smartphone or tablet computer.
2. News is at their fingertips in an instant. An online newspaper can be read more elaborate than a printed newspaper.
3.You can read the Popular News Articles too very easily at the click of the mouse.
4.Access the News By Categories

DISADVANTAGES:
1.It can limited by time.
2.It may rely too heavily on personalities, emotions, opinions… not facts.
3.It can shortchange complex stories or avoid them altogether.

10. CONCLUSION
The Motivation and scope behind this project are to connect people through this application and provide a medium to share their views on the topic/news/information. Then, People with the same interest can interact with each other. However, they can even share more information on the topic. This app while cross-checks the redundancy in the information along with the false and misleading information,which later results in panic in the people.

11. FUTURE SCOPE:

Location feature with automation can be implemented which means as user move from one city to other local news will change as per it. Offline Reading can be improve will more efficient way on full articles. Data quality check needed. If API can't reach to certain article source it gives null value which can cause problem in JSON parsing .

12. APPENDIX

SOURCE CODE:
main.py:

```
from app import app


if __name__ == "__main__":
    app.run(debug=True)
```

__init__.py:

```
from flask import Flask

app = Flask(__name__)

from app import views
```

views.py:

```
from app import app
import ibm_db
from flask import session, flash, redirect, render_template, request, url_for
from .request import (businessArticles, entArticles, get_news_source,
            healthArticles, publishedArticles, randomArticles,
            scienceArticles, sportArticles, techArticles,
            topHeadlines)

conn = ibm_db.connect(
    'DATABASE=bludb;'
    'HOSTNAME=19af6446-6171-4641-8aba-
9dcff8e1b6ff.c1ogj3sd0tgtu0lqde00.databases.appdomain.cloud;'
    'PORT=30699;'
    'SECURITY=SSL;'
    'SSLServerCertificate=DigiCertGlobalRootCA.crt;'
    'UID=vdw12720;'
    'PWD=2C3yBJCDvrFURLPQ;', '', ''
    )
```

```python
    global account

    @app.route('/', methods=['GET', 'POST'])
    def login():
        if request.method == 'POST':
            # getting user data
            email = request.form.get('email')
            password = request.form.get('password')
            sql_check_query = "SELECT * FROM user WHERE email = ?"
            stmt = ibm_db.prepare(conn, sql_check_query)
            ibm_db.bind_param(stmt, 1, email)
            ibm_db.execute(stmt)
            account = ibm_db.fetch_assoc(stmt)
            print(account)
            if account:
                # email id exists
                # checking if the password is correct
                if not account['PASSWORD'] == password:
                    flash('Invalid password', category='error')

                else:
                    # user entered the correct password
                    # redirecting the user to the dashboard
                    session['user_id'] = account['EMAIL']
                    return redirect(url_for('home'))

            else:
                # email id does not exist in the database
                flash('Email invalid... Try Again', category='error')

            return render_template('auth/login.html')

        return render_template('auth/login.html')
        # return render_template('login.html')

    @app.route('/register', methods=['GET', 'POST'])
    def register():
        if request.method == 'POST':
            # getting user data
            email = request.form.get('email')
            password = request.form.get('password')
            # checking: user already exists or not
            sql_check_query = "SELECT * FROM user WHERE email = ?"
            stmt = ibm_db.prepare(conn, sql_check_query)
            ibm_db.bind_param(stmt, 1, email)
            ibm_db.execute(stmt)

            account = ibm_db.fetch_assoc(stmt)
            # email id does not exist in the database
            if not account:
                # inserting the data into the database
                sql_insert_query = "INSERT INTO user VALUES (?, ?)"
                stmt = ibm_db.prepare(conn, sql_insert_query)
                ibm_db.bind_param(stmt, 1, email)
                ibm_db.bind_param(stmt, 2, password)
                ibm_db.execute(stmt)

                # user data inserted into the database
                # redirecting to login page
```

```python
            flash('User created successfully! Please Login', category='success')
            return redirect('/')

        else:
            flash('Email id already exists! Try another one', category='error')

        return render_template('auth/register.html')

    return render_template('auth/register.html')
    # return render_template('register.html')

@app.route('/home', methods=['GET', 'POST'])
def home():
    articles = publishedArticles()

    todays_news = articles[0]
    top_headlines = articles[1]
    business_articles = articles[2]
    tech_articles = articles[3]
    entertainment_articles = articles[4]
    science_articles = articles[5]
    sport_articles = articles[6]
    health_articles = articles[7]

    return  render_template('home.html',
                    todays_news = todays_news,
                    top_headlines = top_headlines,
                    business_articles = business_articles,
                    tech_articles = tech_articles,
                    entertainment_articles = entertainment_articles,
                    science_articles = science_articles,
                    sport_articles = sport_articles,
                    health_articles = health_articles,
                    account = session['user_id'])

@app.route('/headlines')
def headlines():
    headlines = topHeadlines()

    return  render_template('headlines.html', headlines = headlines, account = session['user_id'])

@app.route('/articles')
def articles():
    random = randomArticles()

    return  render_template('articles.html', random = random, account = session['user_id'])

@app.route('/sources')
def sources():
    newsSource = get_news_source()

    return  render_template('sources.html', newsSource = newsSource, account = session['user_id'])

@app.route('/category/business')
def business():
    sources = businessArticles()

    return  render_template('business.html', sources = sources, account = session['user_id'])
```

```python
@app.route('/category/tech')
def tech():
    sources = techArticles()

    return  render_template('tech.html', sources = sources, account = session['user_id'])

@app.route('/category/entertainment')
def entertainment():
    sources = entArticles()

    return  render_template('entertainment.html', sources = sources, account = session['user_id'])

@app.route('/category/science')
def science():
    sources = scienceArticles()

    return  render_template('science.html', sources = sources, account = session['user_id'])

@app.route('/category/sports')
def sports():
    sources = sportArticles()

    return  render_template('sport.html', sources = sources, account = session['user_id'])

@app.route('/category/health')
def health():
    sources = healthArticles()

    return  render_template('health.html', sources = sources, account = session['user_id'])
```

request.py:

```python
from app.models import Articles
from app.models import Sources
from newsapi import NewsApiClient
from app.config import Config
import urllib.request,json

api_key=None
base_url=None
base_url_for_everything=None
base_url_top_headlines=None
base_source_list=None

all_types_articles = []

def news_sorter(all_articles):

    source = []
    title = []
    desc = []
    author = []
    img = []
    p_date = []
    url = []

    for i in range(len(all_articles)):

        article = all_articles[i]
```

```python
        source.append(article['source'])
        title.append(article['title'])
        desc.append(article['description'])
        author.append(article['author'])
        img.append(article['urlToImage'])
        p_date.append(article['publishedAt'])
        url.append(article['url'])

        article_object = Articles(source, title, desc, author, img, p_date, url)
        all_types_articles.append(article_object)
        contents = zip(source, title, desc, author, img, p_date, url)

    return  contents

def publishedArticles():
    newsapi = NewsApiClient(api_key= Config.API_KEY)
    news= []

    #todays news

    get_articles = newsapi.get_everything(sources= 'cnn, reuters, cnbc, the-verge, gizmodo, the-next-web, techradar,
recode, ars-technica')
    all_articles = get_articles['articles']

    output = news_sorter(all_articles)
    news.append(output)

    #top headlines

    get_articles = newsapi.get_top_headlines(sources= 'cnn, reuters, cnbc, techcrunch, the-verge, gizmodo, the-next-
web, techradar, recode, ars-technica')
    all_articles = get_articles['articles']

    output = news_sorter(all_articles)
    news.append(output)

    #business articles

    get_articles = newsapi.get_top_headlines(category='business')
    all_articles = get_articles['articles']

    output = news_sorter(all_articles)
    news.append(output)

    #tech articles

    get_articles = newsapi.get_top_headlines(category='technology')
    all_articles = get_articles['articles']

    output = news_sorter(all_articles)
    news.append(output)

    #entertainment_articles

    get_articles = newsapi.get_top_headlines(category='entertainment')
    all_articles = get_articles['articles']

    output = news_sorter(all_articles)
    news.append(output)
```

```python
    #science_articles

    science_articles = newsapi.get_top_headlines(category='science')
    all_articles = science_articles['articles']

    output = news_sorter(all_articles)
    news.append(output)

    #sport articles

    sport_articles = newsapi.get_top_headlines(category='sports')
    all_articles = sport_articles['articles']

    output = news_sorter(all_articles)
    news.append(output)

    #health articles

    get_articles = newsapi.get_top_headlines(category='health')
    all_articles = get_articles['articles']

    output = news_sorter(all_articles)
    news.append(output)

    return news

def topHeadlines():
    newsapi = NewsApiClient(api_key= Config.API_KEY)
    top_headlines = newsapi.get_top_headlines(sources= 'cnn, reuters, cnbc, techcrunch, the-verge, gizmodo, the-
next-web, techradar, recode, ars-technica')
    all_headlines = top_headlines['articles']

    return  news_sorter(all_headlines)

def randomArticles():
    newsapi = NewsApiClient(api_key= Config.API_KEY)
    random_articles = newsapi.get_everything(sources= 'the-verge, gizmodo, the-next-web, recode, ars-technica')
    all_articles = random_articles['articles']

    return  news_sorter(all_articles)

def businessArticles():
    newsapi = NewsApiClient(api_key= Config.API_KEY)
    business_articles = newsapi.get_top_headlines(category='business')
    all_articles = business_articles['articles']

    return  news_sorter(all_articles)

def techArticles():
    newsapi = NewsApiClient(api_key= Config.API_KEY)
    tech_articles = newsapi.get_top_headlines(category='technology')
    all_articles = tech_articles['articles']

    return  news_sorter(all_articles)

def entArticles():
    newsapi = NewsApiClient(api_key= Config.API_KEY)
    ent_articles = newsapi.get_top_headlines(category='entertainment')
```

```python
    all_articles = ent_articles['articles']

    return  news_sorter(all_articles)

def scienceArticles():
    newsapi = NewsApiClient(api_key= Config.API_KEY)
    science_articles = newsapi.get_top_headlines(category='science')
    all_articles = science_articles['articles']

    return  news_sorter(all_articles)

def sportArticles():
    newsapi = NewsApiClient(api_key= Config.API_KEY)
    sport_articles = newsapi.get_top_headlines(category='sports')
    all_articles = sport_articles['articles']

    return  news_sorter(all_articles)

def healthArticles():
    newsapi = NewsApiClient(api_key= Config.API_KEY)
    health_articles = newsapi.get_top_headlines(category='health')
    all_articles = health_articles['articles']

    return  news_sorter(all_articles)

def get_news_source():
    '''
    Function that gets the json response to our url request
    '''
    get_news_source_url = 'https://newsapi.org/v2/sources?apiKey=' + Config.API_KEY
    with urllib.request.urlopen(get_news_source_url) as url:
        get_news_source_data = url.read()
        get_news_source_response = json.loads(get_news_source_data)
        news_source_results = None

        if get_news_source_response['sources']:
            news_source_results_list = get_news_source_response['sources']
            news_source_results = process_sources(news_source_results_list)

    return news_source_results

def process_sources(source_list):
    '''
    function that process the news articles and transform them to a list of objects
    '''
    news_source_result = []
    for news_source_item in source_list:
        name = news_source_item.get('name')
        description = news_source_item.get('description')
        url = news_source_item.get('url')

        if name:
            news_source_object = Sources(name, description,url)
            news_source_result.append(news_source_object)
    return news_source_result
```

models.py:

```python
class Sources:
    def __init__(self, name, description, url):
        self.name=name,
        self.description=description
        self.url=url


class Articles:
    '''Define article model'''
    def __init__(self, source, author, title, description, url, urlToImage, publishedAt):
        self.source = source
        self.author = author
        self.title = title
        self.description = description
        self.url = url
        self.urlToImage = urlToImage
        self.publishedAt = publishedAt
```

config.py:
```python
class Config:

    NEWS_BASE_URL_SOURCES = 'https://newsapi.org/v2/top-headlines/sources?apiKey={}'
    NEWS_BASE_EVERYTHING_URL = 'https://newsapi.org/v2/everything?domains={}&apiKey={}'
    NEWS_BASE_HEADLINES_URL = 'https://newsapi.org/v2/top-headlines?country=us&apiKey={}'
    NEWS_BASE_SOURCE = 'https://newsapi.org/v2/top-headlines?sources={}&apiKey={}'
    API_KEY = " f1a683b7df544ace8de3d9ce54790eb1"

class ProdConfig(Config):
    pass

class DevConfig(Config):
    DEBUG = True


config_options= {
    'development': DevConfig,
    'production': ProdConfig
}
```

GitHub Link:

https://github.com/IBM-EPBL/IBM-Project-37913-1660364659

Project Demo Link

https://drive.google.com/file/d/1xszyGxWAdPILN2vtRcWiwjM067opd9Do/view?usp=sharing