NEWS TRACKER APPLICATION

A PROJECT REPORT

Submitted By

TAMILARASU V
JINI J TRACY
MITHRA P
JESHEENA A

In partial fulfillment for the award of the degree

Of

BACHELOR OF TECHNOLOGY

In

INFORMATION TECHNOLOGY

GOVERNMENT COLLEGE OF TECHNOLOGY

COIMBATORE-641013

November 2022

BONAFIDE CERTIFICATE

Certified that IBM project report **"NEWS TRACKER APPLICATION"** is bonafide work of **"TAMILARASU V,JINI J TRACY,MITHRA P,JASHEENA A"** who carried out this project work under my supervision.

SIGNATURE

Dr.R.DEVI

FACULTY ADVISOR
Professor
Dept of Information Technology,
Government college of technology
Coimbatore-641013

SIGNATURE

Mr.Gladson oliver

MENTOR
Assistant Professor
Dept of Information Technology,
Government college of technology
Coimbatore-641013

TABLE OF CONTENTS

TABLE OF FIGURES

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 :

The purpose of newspapers should be to inform people with diverse up-to-date content, which they may have otherwise not been able to discover on their own. At least that is the idealized version of a commercial newspaper. As a matter of fact, newspapers have evolved to serve the interests of the masses and know exactly how to deliver people the content they would like to see. So, newspaper and magazine publishers have a hard time keeping up with the pace. Change is needed and publishers must embrace mobile.

2. LITERATURE SURVEY

2.1 Existing system

Literature Survey:

(1)News Keyword Extraction for Topic Tracking

This paper presents a keyword extraction technique that can be used for tracking topics over time. In our work, keywords are a set of significant words in an article that gives high-level description of its contents to readers. Identifying keywords from a large amount of on-line news data is very useful in that it can produce a short summary of news articles. As on-line text documents rapidly increase in size with the growth of WWW, keyword extraction has become a basis of several text mining applications such as search engine, text categorization, summarization, and topic detection. Manual keyword extraction is an extremely difficult and time consuming task; in fact, it is almost impossible to extract keywords manually in case of news articles published in a single day due to their volume. For a rapid use of keywords, we need to establish an automated process that extracts keywords from news articles. We propose an unsupervised keyword extraction technique that includes several variants of the conventional TF-IDF model with reasonable heuristics.

(2)Breaking News Detection and Tracking in Twitter

Twitter has been used as one of the communication channels for spreading breaking news. We propose a method to collect, group, rank and track breaking news in Twitter. Since short length messages make similarity comparison difficult, we boost scores on proper nouns to improve the grouping results. Each group is ranked based on popularity and reliability factors. Current detection method is limited to facts part of messages. We developed an application called "Hotstream" based on the proposed method. Users can

discover breaking news from the Twitter timeline. Each story is provided with the information of message originator, story development and activity chart. This provides a convenient way for people to follow breaking news and stay informed with real-time updates.

(3)Learning approaches for detecting and tracking news events
The authors extend existing supervised-learning and unsupervised-clustering algorithms to allow document classification based on the information content and temporal aspects of news events. They've adapted several IR and machine learning techniques for effective event detection and tracking. The article discusses our research using manually segmented documents.

(4)Using Cloud Computing Capabilities On The Example Of Implementing A News Application-Function
The possibilities of cloud computing technologies are considered on the example of the application implementation, which is a function that receives a news feed through the NewsApi service. The cloud computing model FaaS (Function as a Service), the Microsoft Azure cloud platform and the Azure Functions solution are used for implementation.

(5)Explaining the News Feed Algorithm: An Analysis of the "News Feed FYI" Blog
Facebook uses algorithmic curation---automated selection and ranking of content---to present a personalized News Feed to each user for consumption. However, the News Feed user interface provides little information to help users understand how the ranking algorithm works. We analyzed the company's "News Feed FYI' blog series to better understand the degree to which Facebook employs "how'and "why' explanations of its News Feed algorithm. These types of explanations have been used in other recommendation and intelligent systems as a means of promoting user understanding and acceptance. Our findings show that the "News Feed" FYI' blog posts focus more on explanations that justify why the algorithm works the way it does, and less on explanations that describe how the system works. These findings suggest that the "News Feed" FYI' series would be more helpful for increasing users' confidence in the system, but not improving their trust in the system.

(6)Android News App
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.

(7)Self-Hosted Kubernetes: Deploying Docker Containers Locally With Minikube
Containerization is a cutting-edge DevOps technology which unifies the IT operations and Development domains. In recent times, virtualization using Virtual Machines has become an overkill for its large overhead on systems. As a lightweight alternative,

containerization offers containers that constitute a package of an application along with all its dependencies that is required for it to execute. Containerization platforms help in building containers from images. Docker is a widely popular containerization platform. Containerization Orchestration tools manage these containers. Kubernetes is the front runner of the emerging market of container orchestration tools. These software work together seamlessly in order to successfully implement containerization both locally and on the cloud. In this paper, we aim to deploy the container orchestration tool Kubernetes on a local system with a Docker sample container. The purpose of this is to ensure that all the configurations and management needed for a Docker container is set successfully on the local system before it is deployed onto the cloud or on the premise. The on-premise deployment use case is very important in domains such as finance and healthcare where organizations hesitate to upload confidential information on to the cloud for security reasons but still require scaling of their applications

(8)Research on Topic Detection and Tracking for Online News Texts

With the rapid development of the Internet, the amount of data has grown exponentially. On the one hand, the accumulation of big data provides the basic support for artificial intelligence. On the other hand, in the face of such huge data information, how to extract the knowledge of interest from it has become a matter of general concern. Topic tracking can help people to explore the process of topic development from the huge and complex network texts information. By effectively organizing large scale news documents, a method for the evolution of news topics over time is proposed in this paper to realize the tracking and evolution of topics in the news text set. First, the LDA (latent Dirichlet allocation) model is used to extract topics from news texts and the Gibbs Sampling method is used to speculate parameters. The topic mining using the Kmeans method is compared to highlight the advantages of using LDA for topic discovery. Second, the improved single-pass algorithm is used to track news topics. The JS (JensenShannon) divergence is used to measure the topic similarity, and the time decay function is introduced to improve the similarity between topics with the similar time. Finally, the strength of the news topic and the content change of the topic in different time windows are analyzed. The experiments show that the proposed method can effectively detect and track the topic and clearly reflect the trend of topic evolution

(9) A Cloud-based Framework for COVID-19 Media Classification, Information Extraction, and Trends Analysis

The coronavirus COVID-19 pandemic has become the center of concern worldwide and hence the focus of media attention. Checking the coronavirus-related news and updates has become a daily routine of everyone. Hence, news processing and analytics become key solutions to harvest the real value of this massive amount of news. This conscious growth of published news about COVID-19 makes it hard for a variety of audiences to navigate through, analyze, and select the most important news (e.g., relevant information about the pandemic, its evolution, the vital precautions, and the necessary interventions). This can be realized using current and emerging technologies including Cloud computing, Artificial Intelligence (AI) and Deep Learning (DL). In this paper, we propose a framework to analyze the massive amount of public Covid-19 media reports over the Cloud. This framework encompasses four modules, including text preprocessing, deep learning, and machine learning-based news information extraction,

and recommendation. We conducted experiments to evaluate three modules of our framework and the results we have obtained prove that combining derived information from the news reports provides the policymakers, health authorities, and the public, a complete picture of the way this virus is proliferating. Analyzing this data swiftly is a powerful tool to provide imperative answers to questions that are relevant to public health.

(10) A real-time news personalized push notifier using topic modeling and social scoring for enhanced reader engagement
Push Notification (PN) and Personalized Push Notifications (PPN) are key contemporary topics in mobile app industry today. Push notifications provide a viable content recommendation channel which complements in-app recommendation in mobile apps. There are existing algorithms for in-app content recommendation, however, the PN based recommendation systems are still under research. In this paper, we present "Clairvoyant-Push" - a novel Personalized Push Notification system based on user segmentation and social scoring. User segmentation is done by using the Latent Dirichlet Allocation (LDA) based topic modeling. Moreover, social scoring is used to assign score to each articles to filter out the quality news content for each segments. We have deployed and tested our proposed system using A/B testing framework. The results show an average of 89% lift in opening rate compared to the control group. Further, the results indicate that our system is outperforming with an opening rate of 1012% compared to the industry standard personalised push opening rate of 6-8%.

Reference:

□ News Keyword Extraction for Topic Tracking (https://ieeexplore.ieee.org/document/4624203)

□ Breaking News Detection and Tracking in Twitter (https://ieeexplore.ieee.org/abstract/document/5616930)

□ Learning approaches for detecting and tracking news events (https://ieeexplore.ieee.org/abstract/document/784083)

 USING CLOUD COMPUTING CAPABILITIES ON THE EXAMPLE OF IMPLEMENTING A NEWS APPLICATION-FUNCTION (https://elib.psu.by/handle/123456789/31517)

 Explaining the News Feed Algorithm: An Analysis of the "News Feed FYI"

Blog (https://dl.acm.org/doi/abs/10.1145/3027063.3053114)

 Android News App (https://www.ripublication.com/ijaer18/ijaerv13n11_78.pdf)

 SELF-HOSTED KUBERNETES: DEPLOYING DOCKER CONTAINERS LOCALLY WITH MINIKUBE (https://ieeexplore.ieee.org/abstract/document/9170208)

 Research on Topic Detection and Tracking for Online News Texts (https://ieeexplore.ieee.org/document/8703401)

 A Cloud-based Framework for COVID-19 Media Classification, Information Extraction, and Trends Analysis (https://ieeexplore.ieee.org/document/9658709)

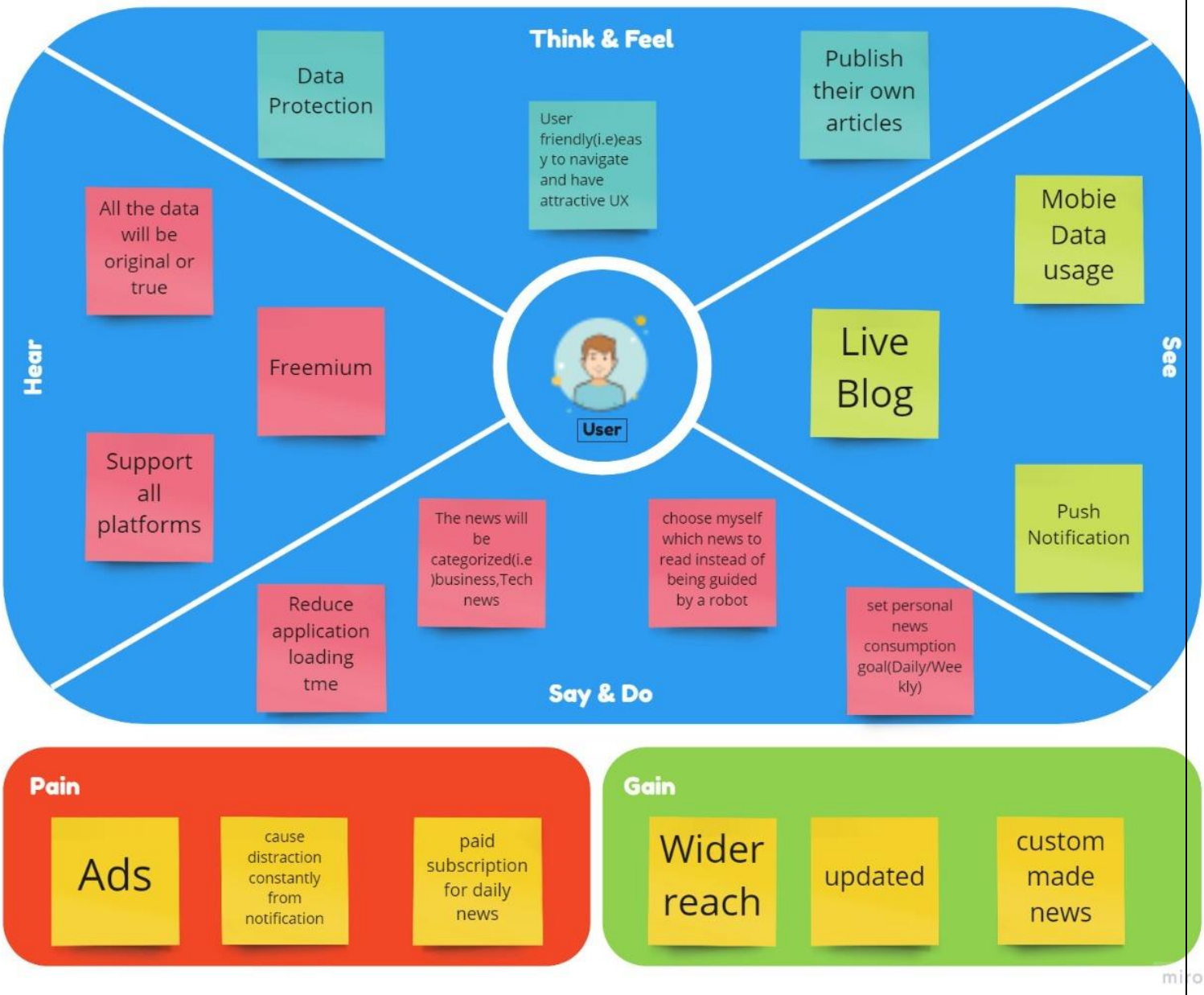
 A real-time news personalized push notifier using topic modeling and social scoring for enhanced reader engagement (https://ieeexplore.ieee.org/document/7364120

2.2 Problem Statement Definition

NEWS TRACKER APPLICATION Users expect updates to be immediately available and accessible via multiple devices, and find relevant and important news easily every day because there are multiple news sharing apps used by a single user and are often spammed with notifications which cause users to miss important events across the Globe

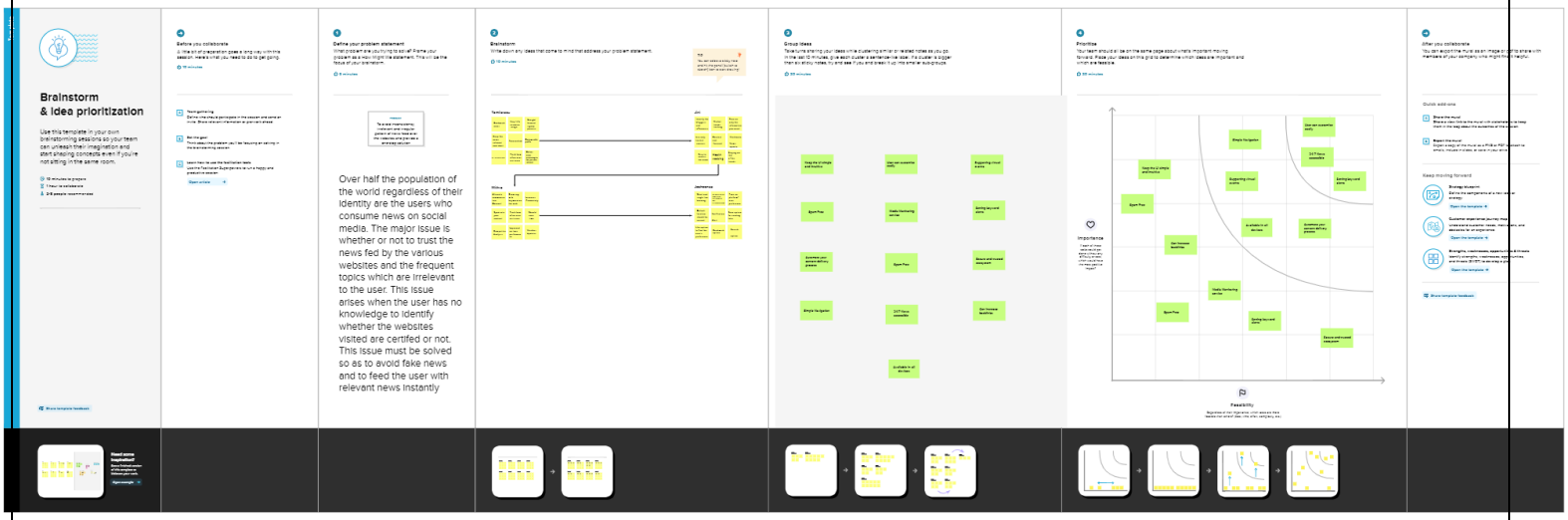
3. IDEATION & PROPOSED SOLUTION

3.1 Empathy Map Canvas



z

3.2 ation & Brainstorming

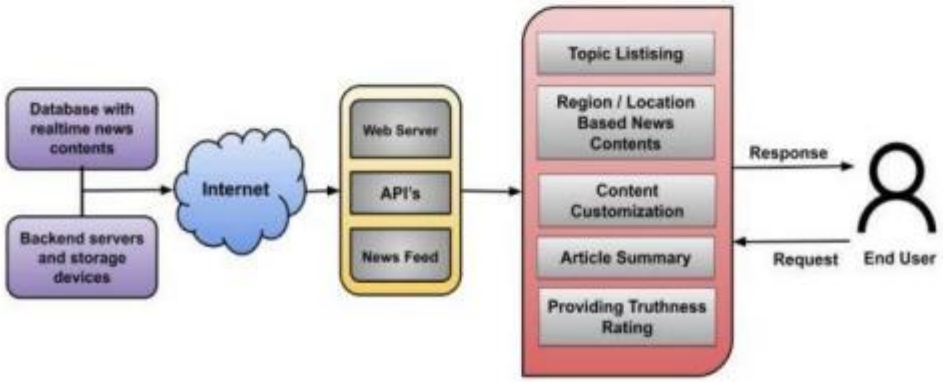


Solution Architecture:

Solution architecture is a complex process – with many sub-processes – that bridges the gap between business problems and technology solutions. Its goals are to:

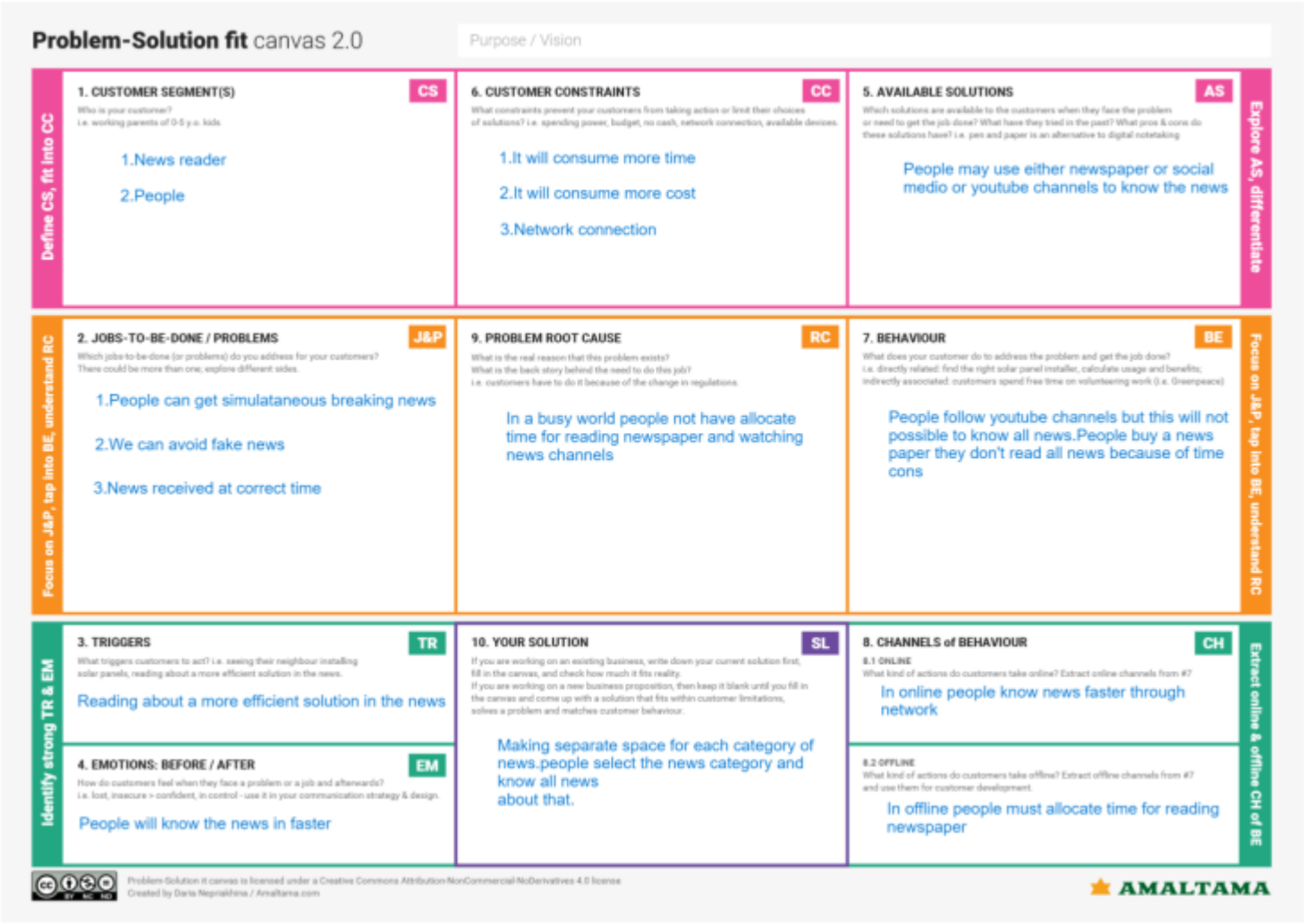
- Find the best tech solution to solve existing business problems.
- Describe the structure, characteristics, behavior, and other aspects of the software to project stakeholders.
- Define features, development phases, and solution requirements.
- Provide specifications according to which the solution is defined, managed, and delivered.

Example - Solution Architecture Diagram:



3.3 Problem Solution fit

Template:



Project Design Phase-II

Solution Requirements (Functional & Non-functional)

Date	08 October 2022
Team ID	PNT2022TMID07030
Project Name	Project – News Tracker Application
Maximum Marks	4 Marks

Functional Requirements:

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 entering username and password Login through Login through email
FR-4	User interaction	Done through user interface between client and server View the related news by subscripted or requested page

4.

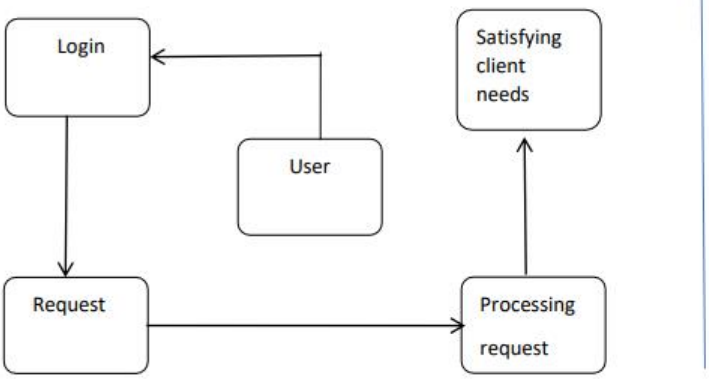
5. PROJECT DESIGN

5.1 Data Flow Diagrams

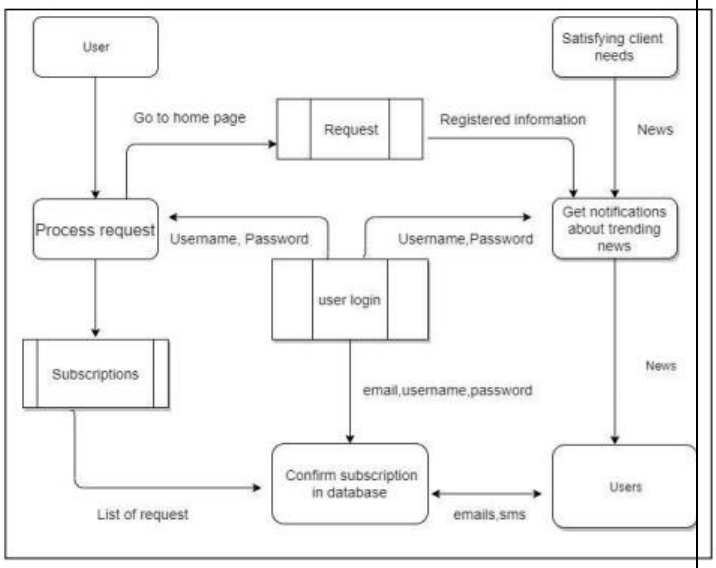
Data Flow Diagrams:

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

Example:



Example: DFD Level 0 (Industry Standard)



Solution & Technical Architecture

Example - Solution Architecture Diagram: NEWS TRACKER APPLICATION

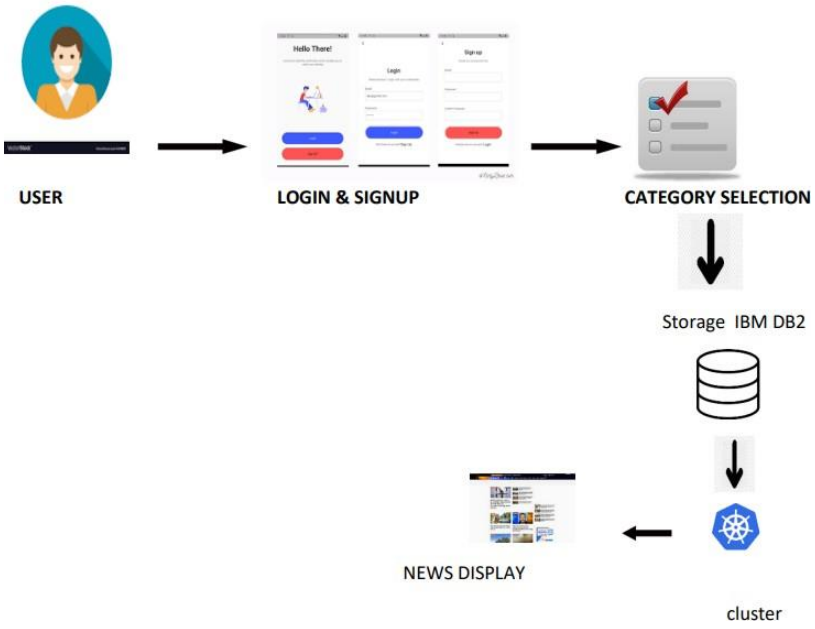


FIG 1: Data flow and Technical Architecture

- The user interacts with the application.

Registers by giving the details.

- Integrate the application with news APIs and store the data in the database.
- The database will have all the details and the user can search the news by using a search bar.

5.3 User Stories

Functional Requirement (Epic)	User Story Number	User Story / Task	Acceptance criteria	Priority	Release
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 my dashboard	High	Sprint-2
Home Page	USN-4	As a user I can view the headlines of the news that interest me	I can read the news elaborately by clicking on the headlines	High	Sprint-2
	USN-6	As a user I can search a newsI want	I can read the news elaborately	High	Sprint-4
Chat assistance	USN-6	As a user I can ask my queries about the application with the help of the chat assistance	I can contact the customer care executive if needed	Medium	Sprint-3

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	USN-1	As a user, I can register for theapplication by entering my email, password, and confirming my password.	10	High	TAMILARASU
Sprint-1		USN-2	As a user, I will receive confirmation emailonce I have registered for the application	10	High	JINI J TRACY
Sprint-1	Login	USN-3	As a user, I can log into the application by entering email & password.	15	High	MITHRA
Sprint-2	Input Necessary Details	USN-4	As a user, I can search the news in the application	15	High	JESHEENA
Sprint-2	Data Pre-processing	USN-5	The application searches for news related to the entered details.	15	High	TAMILARASU
Sprint-3	Searching of news	USN-6	As a user, I can search for the accurate news what I want	20	High	JINI J TRACY
Sprint-3		USN-7	As a user, I can get accurate news in the application	5	Medium	MITHRA
Sprint-4	Review	USN-8	As a user, I can give feedback of the application.	20	High	JESHEENA

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	20	6 Days	24 Oct 2022	29 Oct 2022	20	29Oct2022
Sprint-2	20	6 Days	31 Oct 2022	05 Nov 2022	20	05 Nov 2022
Sprint-3	20	6 Days	07 Nov 2022	12 Nov 2022	20	12 NOV 2022
Sprint-4	20	6 Days	14 Nov 2022	19 Nov 2022	20	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)

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

6.3 Reports from JIRA

Sprit-1

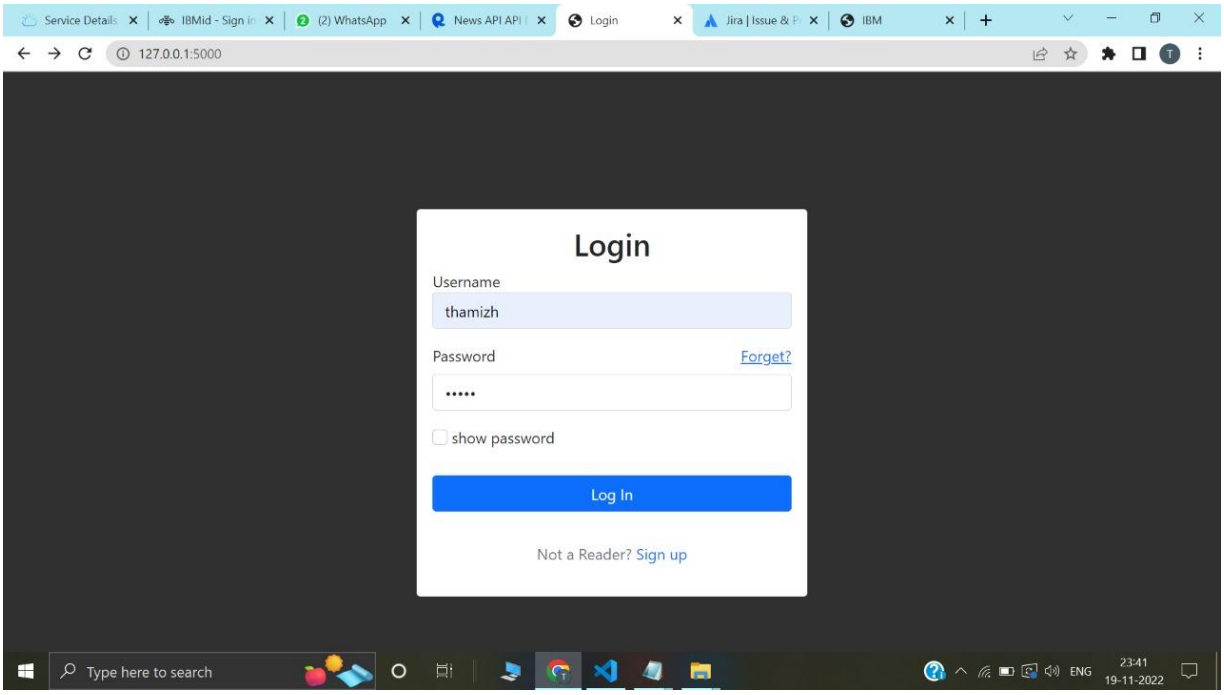


FIG 3: Sprint 1

Sprint-2

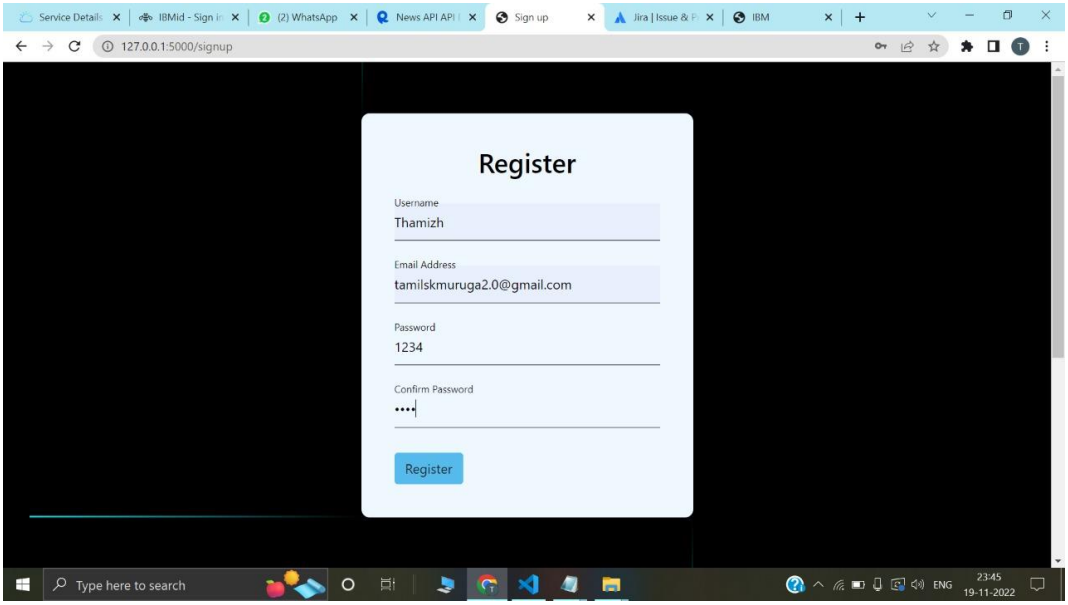


FIG 4:Sprint 2

Sprint-3

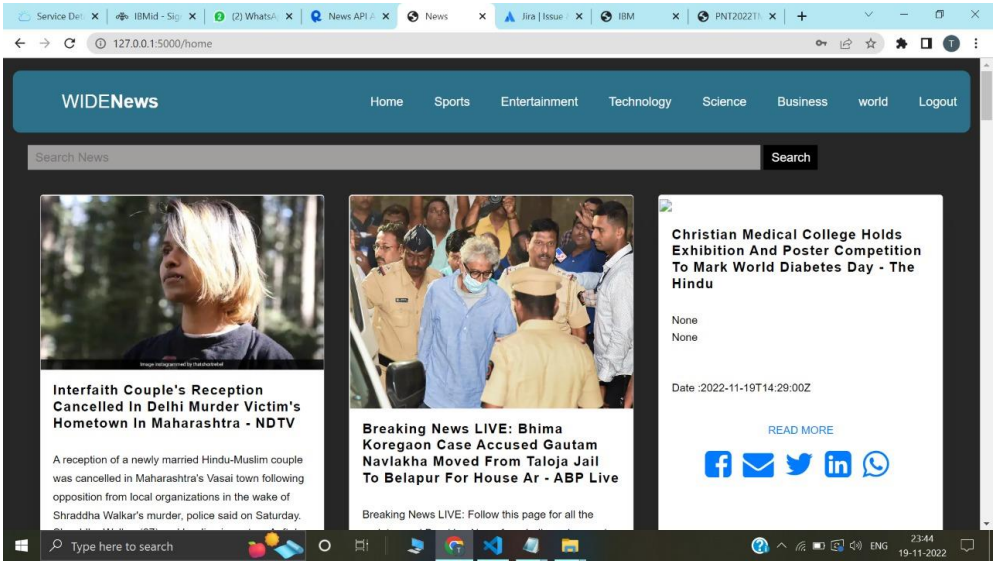


FIG 5: Sprint 3

Sprint 4

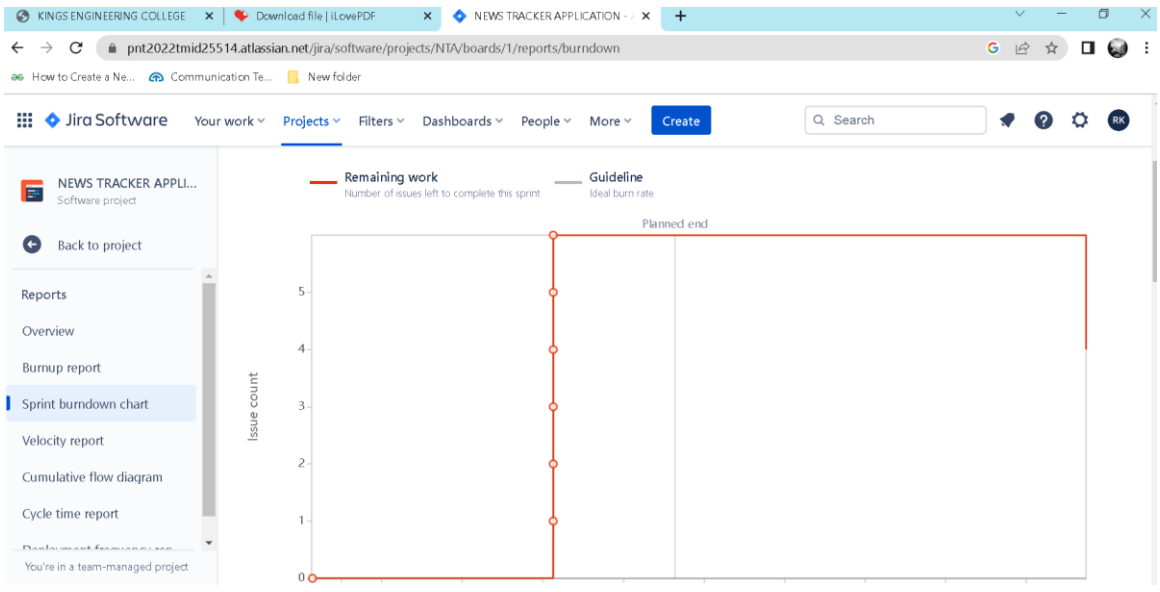


FIG 6: Sprint 4

7 CODING & SOLUTIONING

7.1 Feature 1

User easy interface for using the application

7.2 Feature 2

Easy acesss

8 TESTING

8.1 User Acceptance Testing

				Date	16-Nov-22							
				Team ID	PNT2022TMID25514							
				Project Name	Project - News Tracker Application							
				Maximum Marks	4 marks							
	Feature Type	Component	Test Scenario	Pre-Requisite	Steps To Execute	Test Data	Expected Result	Actual Result	Status	Comments	TC for Automation(Y/N)	BUG ID	Executed By
	Functional	Home Page	Verify user is able to see the Login/Signup p when user clicked on account LOGIN button		1.Enter URL and click go 2.Click on Account Login dropdown button 3.Verify login/Singup popup displayed or not	https://shopsnear.com/	Login/Signup popup should display	Working as expected	Pass				MADLIN JENCY
	UI	Home Page	Verify the UI elements in Login/Signup popup		1.Enter URL and click go 2.Click on Account Login dropdown button 3.Verify login/Singup with below UI elements: a.email box b.password box c.Login button d.New customer? CreatEe aRegister link e.Last password? Recovery password link	https://shopsnear.com/	Application should show below UI elements: a.email text box b.password text box c.Login button with orange colour d.New customer? Create account link e.Last password? Recovery password link	Working as expected	Fail	Steps are not clear to follow		BUG-1234	LIDIYA JONES RAJ
	Functional	Home page	Verify user is log into application with Valid credentials		1.Enter URL(https://shopsnear.com/) and click go 2.Click on Account Login dropdown button 3.Enter Valid username/email in Email box 4.Enter valid password in password text box 5.Click on login button	Username: chalsm@gmail.com password: Testing123	User should navigate to user account homepage						KEERTHANA
	Functional	Login page	Verify user is log into application with InValid credentials		1.Enter URL(https://shopsnear.com/) and click go 2.Click on Account Login dropdown button 3.Enter Invalid username/email in Email text box 4.Enter valid password in password text box 5.Click on login button	Username: chalsm@gmail password: Testing123	Application should show 'Incorrect email or password ' validation message.						NAGESWARI
	Functional	Login page	Verify user is log into application with InValid credentials		1.Enter URL(https://shopsnear.com/) and click go 2.Click on Account Login dropdown button 3.Enter Valid username/email in Email text box 4.Enter Invalid password in password text box 5.Click on login button	Username: chalsm@gmail.com password: Testing12367866786767 876	Application should show 'Incorrect email or password ' validation message.						KEERTHANA
	Functional	Login page	Verify user is log into application with InValid credentials		1.Enter URL(https://shopsnear.com/) and click go 2.Click on Account Login dropdown button 3.Enter InValid username/email in Email text box 4.Enter Invalid password in password text box 5.Click on login button	Username: chalsm password: Testing12367866786767 876	Application should show 'Incorrect email or password ' validation message.						LIDIYA JONES RAJ

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	10	4	2	3	20
Duplicate	1	0	3	0	4
External	2	3	0	1	6
Fixed	11	2	4	20	37
Not Reproduced	0	0	1	0	1
Skipped	0	0	1	1	2
Won't Fix	0	5	2	1	8
Totals	24	14	13	26	77

3. Test Case Analysis

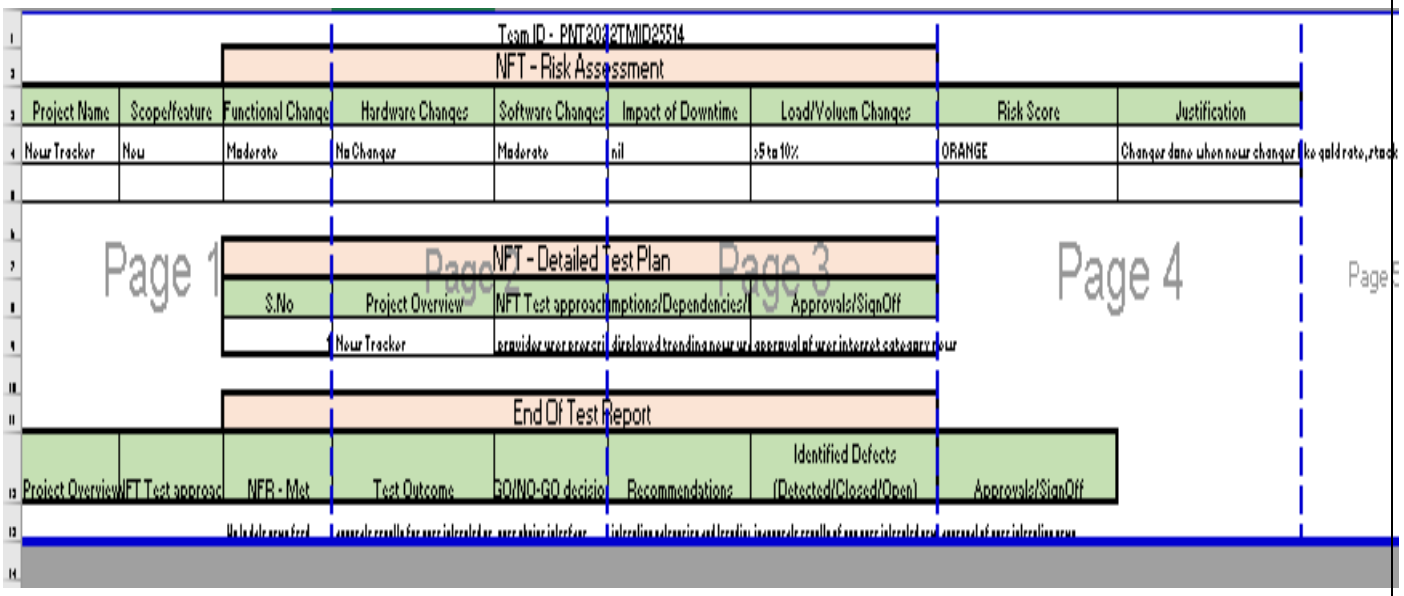
This report shows the number of test cases that have passed, failed, and untested

Section	Total Cases	Not Tested	Fail	Pass
Print Engine	7	0	0	7
Client Application	51	0	0	51
Security	2	0	0	2
Outsource Shipping	3	0	0	3

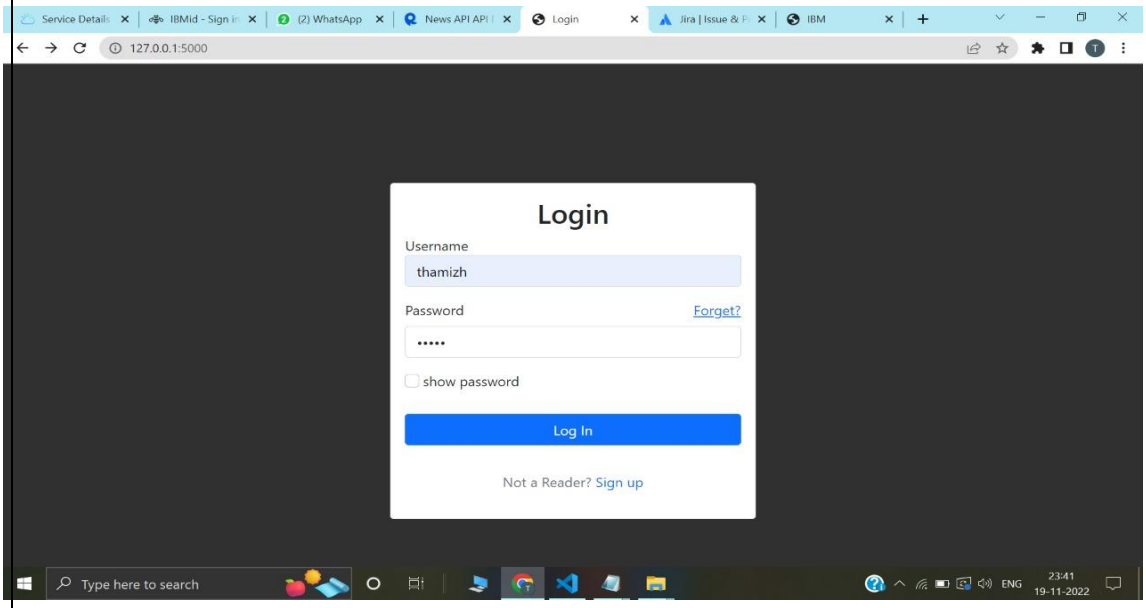
Exception Reporting	9	0	0	9
Final Report Output	4	0	0	4
Version Control	2	0	0	2

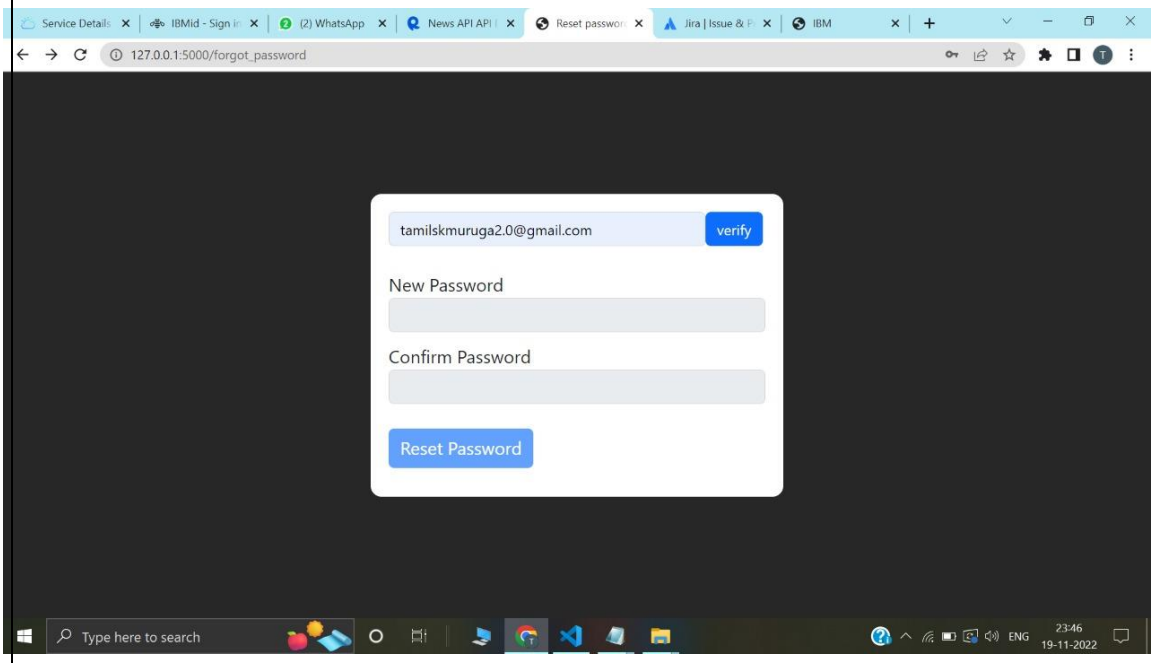
9. RESULTS

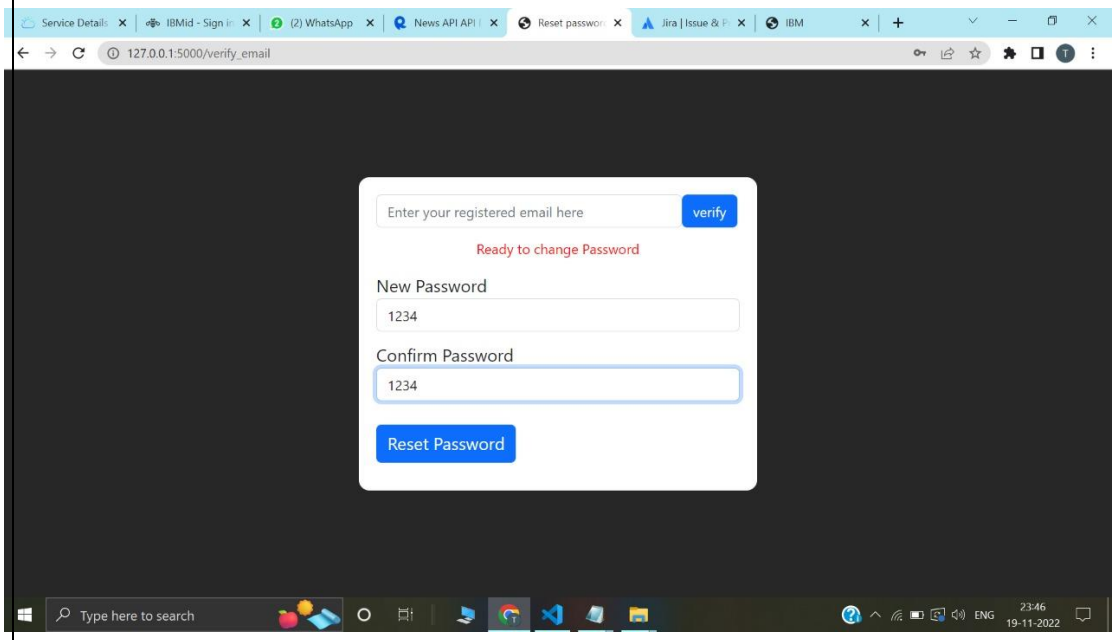
9.1 Performance Metrics

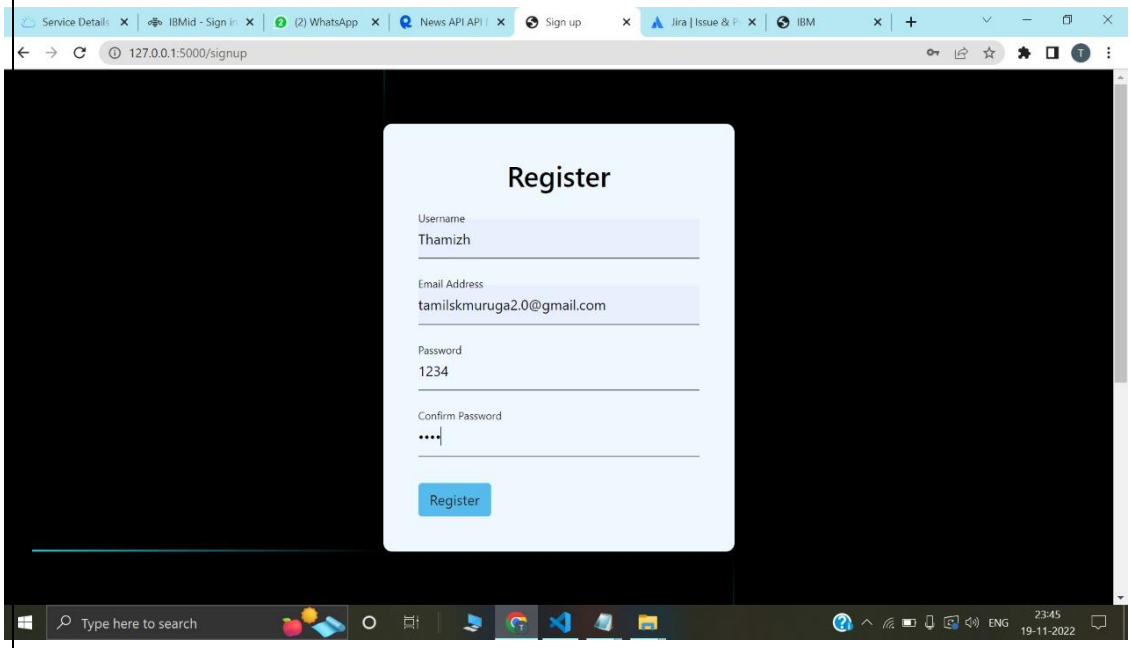


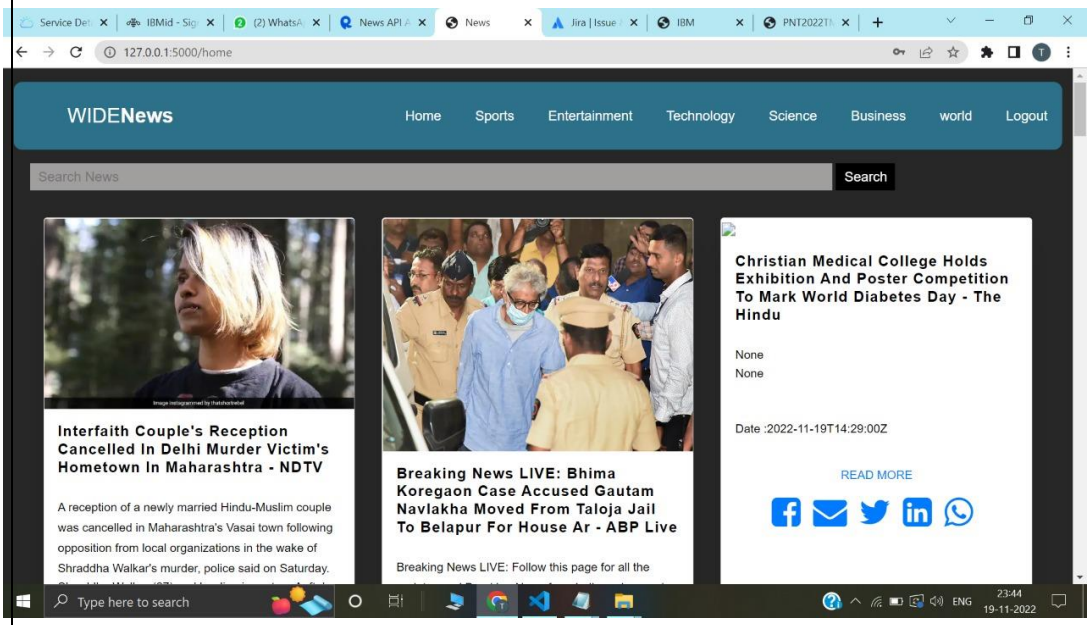
OUTPUT


127.0.0.1:5000
Login
Username
thamizh
Password
Forget?
show password
Log In
Not a Reader? Sign up
Type here to search
ENG
23:41
19-11-2022


127.0.0.1:5000/forgot_password
tamilskmuruga2.0@gmail.com
verify
New Password
Confirm Password
Reset Password
Type here to search
ENG
23:46
19-11-2022


127.0.0.1:5000/verify_email
Enter your registered email here
verify
Ready to change Password
New Password
1234
Confirm Password
1234
Reset Password
Type here to search
ENG
23:46
19-11-2022


127.0.0.1:5000/signup
Register
Username
Thamizh
Email Address
tamilskmuruga2.0@gmail.com
Password
1234
Confirm Password
Register
Type here to search
ENG
23:45
19-11-2022



10.ADVANTAGES & DISADVANTAGES

ADVANTAGES

1. All trending news will be displayed in a orderly manner.
2. We can costomize the topic which we are interested and make them display
3. The user experience are quit simple and friendly

DISADVANTAGES

1. Same news which is published in different websites can be displayed.
2. Delay response of news api can make the news collapse.
3. It does not showing contact page and about page

11. CONCLUSION

Thus the projected using IBM CLOUD is tested ,verified and executed successfully.

12. FUTURE SCOPE

In future, we planned to implement this project in large scale with extra features which will be helpful and used by all the people.

APPENDIX

13.1. Source Code

App,py

```
from flask import Flask,render_template,request,url_for,redirect,flash,session
import hashlib
import ibm_db
import ibm_boto3
from ibm_botocore.client import Config, ClientError


import requests

app=Flask(__name__)
app.secret_key = "abc"



#cloud
connection = ibm_db.connect("DATABASE=bludb;HOSTNAME=815fa4db-dc03-4c70-869a-
a9cc13f33084.bs2io90l08kqb1od8lcg.databases.appdomain.cloud;PORT=30367;SECURITY=SSL;SSLServerCertificat
e=DigiCertGlobalRootCA.crt;UID=ltk79172;PWD=lyUb0CFg1tQd1MDu;","")

news_api_key="f71565537970409cbe006afcbac30c33"

@app.route("/")
def index():
    return  render_template("index.html")

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


@app.route("/register",methods=['GET','POST'])
def register():
    error = None
    if request.method=='POST':
```

```
        username= request.form['username']
        email=request.form['email']
        password=request.form['password']
        cpass=request.form['cpassword']
        if password != cpass:
            error = "Password and Confirm password should be same!!"
        else:
            hashed_password=hashlib.sha256(password.encode()).hexdigest()

            sql="SELECT * FROM newstracker WHERE username=?"
            stat =ibm_db.prepare(connection,sql)
            ibm_db.bind_param(stat,1,username)
            ibm_db.execute(stat)
            res=ibm_db.fetch_assoc(stat)
            if res:
                error="Username is already exits user different username"

            else:

                sql1="INSERT INTO newstracker VALUES (?,?,?)"
                pre_stat =ibm_db.prepare(connection,sql1)
                ibm_db.bind_param(pre_stat,1,username)
                ibm_db.bind_param(pre_stat,2,email)
                ibm_db.bind_param(pre_stat,3,hashed_password)
                ibm_db.execute(pre_stat)

                flash('Account created successfully')
                return redirect(url_for('index'))
    return render_template('signup.html', error = error)


@app.route('/login',methods=["GET","POST"])
def login():
    error = None
    if request.method=="POST":
        username=request.form['username']
        password=request.form['password']
        hashed=hashlib.sha256(password.encode()).hexdigest()


        sql="SELECT * FROM newstracker WHERE username=?"
        con =ibm_db.prepare(connection,sql)
        ibm_db.bind_param(con,1,username)
        ibm_db.execute(con)
        res=ibm_db.fetch_assoc(con)

        if res:
            if hashed == res['HASHED_PASSWORD']:
```

```
            session['username'] = request.form['username']
            return redirect(url_for('home'))
        else:
            error = "Login Failed!!"
            return render_template('index.html',error=error)
    else:
        error = "Login Failed!!"
        return render_template('index.html',error=error)



@app.route('/home')
def home():
    error=None
    if 'username' in session:
        username = session['username']
        main_url="https://newsapi.org/v2/top-headlines?country=in&apiKey="+news_api_key
        news=requests.get(main_url).json()
        articles=news["articles"]
        news_articles_title=[]
        for a in articles:
            news_articles_title.append(a["title"])

        news_articles_description=[]
        for a in articles:
            news_articles_description.append(a["description"])

        news_articles_url=[]
        for a in articles:
            news_articles_url.append(a["url"])

        news_articles_urlToImage=[]
        for a in articles:
            news_articles_urlToImage.append(a["urlToImage"])

        news_articles_publishedAt=[]
        for a in articles:
            news_articles_publishedAt.append(a["publishedAt"])

        news_articles_content=[]
        for a in articles:
            news_articles_content.append(a["content"])


        return render_template("home.html",user=username,articles=articles)
    else:
        error="Pleasr Login to access"
```

```
        return render_template('index.html',error=error)

@app.route('/sports')
def sports():
    error=None
    if 'username' in session:
        username = session['username']
        main_url="https://newsapi.org/v2/top-headlines?country=in&category=sports&apiKey="+news_api_key
        news=requests.get(main_url).json()
        articles=news["articles"]
        return render_template("home.html",user=username,articles=articles)
    else:
        error="Pleasr Login to access"
        return render_template('index.html',error=error)

@app.route('/entertainment')
def entertainment():
    error=None
    if 'username' in session:
        username = session['username']
        main_url="https://newsapi.org/v2/top-headlines?country=in&category=entertainment&apiKey="+news_api_key
        news=requests.get(main_url).json()
        articles=news["articles"]
        return render_template("home.html",user=username,articles=articles)
    else:
        error="Pleasr Login to access"
        return render_template('index.html',error=error)


@app.route('/technology')
def technology():
    error=None
    if 'username' in session:
        username = session['username']
        main_url="https://newsapi.org/v2/top-headlines?country=in&category=technology&apiKey="+news_api_key
        news=requests.get(main_url).json()
        articles=news["articles"]
        return render_template("home.html",user=username,articles=articles)
    else:
        error="Pleasr Login to access"
        return render_template('index.html',error=error)

@app.route('/science')
def science():
    error=None
    if 'username' in session:
        username = session['username']
        main_url="https://newsapi.org/v2/top-headlines?country=in&category=science&apiKey="+news_api_key
```

```
        news=requests.get(main_url).json()
        articles=news["articles"]
        return render_template("home.html",user=username,articles=articles)
    else:
        error="Pleasr Login to access"
        return render_template('index.html',error=error)

@app.route('/business')
def business():
    error=None
    if 'username' in session:
        username = session['username']
        main_url="https://newsapi.org/v2/top-headlines?country=in&category=business&apiKey="+news_api_key
        news=requests.get(main_url).json()
        articles=news["articles"]
        return render_template("home.html",user=username,articles=articles)
    else:
        error="Please Login to access"
        return render_template('index.html',error=error)

@app.route('/world')
def world():
    error=None
    if 'username' in session:
        username = session['username']
        main_url="https://newsapi.org/v2/top-headlines?sources=bbc-news&apiKey="+news_api_key
        news=requests.get(main_url).json()
        articles=news["articles"]
        return render_template("home.html",user=username,articles=articles)
    else:
        error="Pleasr Login to access"
        return render_template('index.html',error=error)

@app.route("/search",methods=["GET","POST"])
def search():
    if 'username' in session:
        username = session['username']
        if request.method=="POST":
            search=request.form["search"]
            news_data = requests.get(f'https://newsapi.org/v2/everything?q={search}&apiKey='+news_api_key).json()
            articles=news_data['articles']
            return render_template("home.html",user=username,articles=articles)



@app.route('/logout')
def logout():
  session.pop('username', None)
```

```
  return redirect(url_for('index'))


@app.route('/forgot_password')
def forgot_password():
   return render_template("resetpassword.html")

@app.route("/verify_email",methods=["GET","POST"])
def verify_email():
   error=None
   if request.method=="POST":
      email=request.form["email"]
      sql="SELECT * FROM newstracker WHERE email=?"
      stat =ibm_db.prepare(connection,sql)
      ibm_db.bind_param(stat,1,email)
      ibm_db.execute(stat)
      res=ibm_db.fetch_assoc(stat)


      if res:
         return render_template("resetpassword.html",result="200",email=email)
      else:
         error="No such email is available"
         return render_template("resetpassword.html",result="404",error=error)


@app.route("/update_password",methods=["GET",'POST'])
def update_password():
   error=None
   if request.method=="POST":
       password=request.form["newp"]
       cpass=request.form["cp"]
       email=request.form["email"]
       if cpass==password:

         hashed=hashlib.sha256(password.encode()).hexdigest()
         sql="UPDATE newstracker SET password=? WHERE email=?"
         stat =ibm_db.prepare(connection,sql)
         ibm_db.bind_param(stat,1,hashed)
         ibm_db.bind_param(stat,2,email)
         ibm_db.execute(stat)

         error="Password reset successful"
         return render_template("index.html",error=error)
       else:
         error="Password and confirm password must be same"
         return render_template("resetpassword.html",error=error)
```

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

index.html

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Login</title>
    <link rel="stylesheet" href="{{url_for('static',filename='/css/index.css')}}">
    <link rel='stylesheet' href='https://cdnjs.cloudflare.com/ajax/libs/twitter-bootstrap/4.1.3/css/bootstrap.min.css'>
    <link rel='stylesheet' href='https://fonts.googleapis.com/css?family=Muli'><link rel="stylesheet" href="./style.css">
    <link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/font-awesome/4.7.0/css/font-awesome.min.css">
    <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.0.2/dist/css/bootstrap.min.css" rel="stylesheet"
integrity="sha384-EVSTQN3/azprG1Anm3QDgpJLIm9Nao0Yz1ztcQTwFspd3yD65VohhpuuCOmLASjC"
crossorigin="anonymous">

</head>
<body>


<div class="container">
    <br><br><br> <br>
    <div class="row p-5" id="row">
        <div class="col-md-5 mx-auto">

            <div class="card card-body">
                <h2 class="text-center">Login</h2>
                        {% with messages = get_flashed_messages() %}
                         {% if messages %}

                                {% for message in messages %}
                                    <p style="text-align: center;color: green;">{{ message }} </p>
                                {% endfor %}

                         {% endif %}
                        {% endwith %}
                        {% if error %}
                         <p style="text-align: center;color: red;">{{ error }}</p>
                         {% endif %}
                <form id="submitForm" action="/login" method="POST" >
```

```
                <div class="form-group required">
                    <lSabel for="username">Username </lSabel>
                    <input type="text" class="form-control text-lowercase" id="username" required="" name="username"
value="">
                </div>
                <div class="form-group required">
                    <label class="d-flex flex-row align-items-center" for="password">Password
                        <a class="ml-auto border-link small-xl" href="{{url_for('forgot_password')}}">Forget?</a></label>
                    <input type="password" class="form-control" required="" id="password" name="password" value="">
                </div>

                <div style="margin-left:20px ;">
                    <input type="checkbox" class="form-check-input" onclick="show()" >
                    <label class="form-check-label">show password</label>
                </div>
                 <br>
                <div class="form-group pt-1">
                    <button class="btn btn-primary btn-block" type="submit">Log In</button>
                </div>
            </form>
            <p class="small-xl pt-3 text-center">
                <span class="text-muted">Not a Reader?</span>

                 <a href="{{url_for('signup')}}" style="text-decoration:none;">Sign up</a>
            </p>
        </div>
    </div>
</div>
</div>



</body>
<script>
    function show(){
        var a=document.getElementById("password");

        if(a.type === "password"){
            a.type="text";
        }
        else{
            a.type="password";
        }
    }
</script>
</html>
```

Signup.html

```
<!DOCTYPE html>
<html lang="en">
<head>
   <meta charset="UTF-8">
   <meta http-equiv="X-UA-Compatible" content="IE=edge">
   <meta name="viewport" content="width=device-width, initial-scale=1.0">
   <title>Sign up</title>
   <link rel="stylesheet" href="{{url_for('static',filename='/css/signup.css')}}">
   <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.0.2/dist/css/bootstrap.min.css" rel="stylesheet"
integrity="sha384-EVSTQN3/azprG1Anm3QDgpJLIm9Nao0Yz1ztcQTwFspd3yD65VohhpuuCOmLASjC"
crossorigin="anonymous">

</head>
<body>
 <div class="container">
   <div class="register">
      <h2>Register</h2>
      <span></span>
      <span></span>
      <span></span>
      <span></span>
      {% if error %}
        <p><strong>Error:</strong> {{ error }}
       {% endif %}
      <form action="/register" method="POST">
       <div class="user-box">
         <input type="text" name="username" required="">
         <label>Username</label>
       </div>
       <div class="user-box">
         <input type="text" name="email" required="">
         <label>Email Address</label>
       </div>
       <div class="user-box" id="pass_f" >
         <input type="text" name="password" id="password" required=""  >
         <label>Password</label>
       </div>
       <div class="user-box">
         <input type="password" name="cpassword" id="cpassword" required="">
         <label>Confirm Password</label>
       </div>
        <button class="btn" type="submit"  id="submit" >Register</button>
      </form>
    </div>
 </div>
```

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

Home.html

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta http-equiv="X-UA-Compatible" content="IE=edge">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <link rel="stylesheet" href="https://cdn.jsdelivr.net/npm/bootstrap@4.0.0/dist/css/bootstrap.min.css"
integrity="sha384-Gn5384xqQ1aoWXA+058RXPxPg6fy4IWvTNh0E263XmFcJlSAwiGgFAW/dAiS6JXm"
crossorigin="anonymous">
  <title>News</title>
  <link rel="stylesheet" href="{{url_for('static',filename='css/home.css')}}">
  <link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/font-awesome/4.7.0/css/font-awesome.min.css">
  <link rel="stylesheet" href="../static/css/home.css">
</head>
<body>

    <nav>
      <input id="nav-toggle" type="checkbox">
      <!--<div class="logo">world<strong>News</strong></div>-->
      <ul class="links">
        <li><a href="{{url_for('home')}}">Home</a></li>
        <li><a href="{{url_for('sports')}}">Sports </a></li>
        <li><a href="{{url_for('entertainment')}}">Entertainment  </a></li>
        <li><a href="{{url_for('technology')}}">Technology </a></li>
        <li><a href="{{url_for('science')}}">Science </a></li>
        <li><a href="{{url_for('business')}}">Business </a></li>
        <li><a href="{{url_for('world')}}">world </a></li>

        <li><a href="{{url_for('logout')}}">Logout</a></li>

      </ul>
      <label for="nav-toggle" class="icon-burger">
        <div class="line"></div>
        <div class="line"></div>
        <div class="line"></div>
      </label>
    </nav>

    <br><br> <br>


    <div class="main">
```

```
    <br>
    <form action="/search" method="POST">
      <input type="text" name="search" class="search" placeholder="Search News">
      <input type="submit" name="submit" class="submit" value="Search">
    </form>

    <br><br>

    <ul class="cards">
      {%for article in articles%}

        <li class="cards_item">
          <div class="card">
            <div class="card_image "><img class="card-img-top" src="{{article.urlToImage}}"></div>
            <div class="card_content">
              <h2 class="card_title">{{article.title}}</h2>
              <br>
              <B class="card_text">{{article.description}}</B>
              <p class="card_text">{{article.content}}</p>
              <br>
              <p class="card_text">Date :{{article.publishedAt}}</p>
              <button class="btn card_btn"><a href="{{article.url}}">Read More</a></button>
                 <div class="share text-center"  style="text-align:cenetr ;">
                    <a href="https://www.facebook.com/sharer/sharer.php?u={{article.url}}" ><i class="fa fa-facebook-
square"  style="font-size:40px;padding-right: 10px;"></i></a>
                    <a href="mailto:?subject={{article.title}} &body={{article.url}}"><i class="fa fa-envelope"
style="font-size:40px;padding-right: 10px;"></i></a>
                    <a href="https://twitter.com/share?url={{article.url}}&text={{article.title}}" ><i class="fa fa-twitter"
style="font-size:40px;padding-right: 10px;"></i></a>
                    <a
href="https://www.linkedin.com/shareArticle?url={{article.urlToImage}}&title={{article.title}}&summary={{article.d
escription}}&source={{article.url}}"><i class="fa fa-linkedin-square" style="font-size:40px;padding-right:
10px;"></i></a>
                    <a href="whatsapp://send?text={{article.url}}" data-action="share/whatsapp/share"><i class="fa fa-
whatsapp" style="font-size:40px;padding-right: 10px;"></i></a>
                 </div>
            </div>
          </div>
        </li>

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

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

Resetpassword.html

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.2.2/dist/css/bootstrap.min.css" rel="stylesheet"
integrity="sha384-Zenh87qX5JnK2Jl0vWa8Ck2rdkQ2Bzep5IDxbcnCeuOxjzrPF/et3URy9Bv1WTRi"
crossorigin="anonymous">

    <title>Reset password</title>
</head>

<style>
    *{
        margin: 0;
        padding: 0;
        box-sizing: border-box;
    }
    body{
      height: 100vh;
      display: flex;
      align-items: center;
      justify-content: center;
      background-color: #272727;

    }
    .forgot-pass{
        width: 460px;
        height: min-content;
        padding: 20px;
        border-radius: 12px;
        background: #ffffff;
    }
    .forgot-pass h1{
        font-size: 36px;
        margin-bottom: 25px;

    }
    .forgot-pass form{
        font-size: 20px;

    }
    .forgot-pass form .form-group{
        margin-bottom: 12px;
```

```
    }
    .forgot-pass form input[type="submit"]{
        font-size: 20px;
        margin-top: 15px;
    }
</style>
<body>
    <div class="forgot-pass">
        {% if error %}
        <p style="text-align: center;color: red;">{{ error }}</p>
        {% endif %}
        <form action="/verify_email" method="POST">
            <div class="row">

                <div class="col-8">
                    <div class="form-group">
                        <table width="170%" >
                            <tr>

                                <td> <input type="email" id="email" name="email"   required class="form-control "
placeholder="Enter your registered email here"></td>
                                <td><input type="hidden" name="" id="result" value="{{result}}">
                                    <button class="btn btn-primary btn-block" type="submit">verify </button></td>

                            </tr>


                        </table>
                    </div>

                </div>




            </div>
        </form>

        <p id="message" style="color: red;text-align: center;"></p>
            <form action="/update_password" method="POST">
                <div class="form-group">
                    <label for="">New Password</label>
                    <input type="text" id="newp" name="newp" class="form-control" disabled >
                </div>

                <div class="form-group">
                    <label for="">Confirm Password</label>
                    <input type="text" id="cp" class="form-control" name="cp" disabled>
```

```
                <input type="hidden" name="email" value="{{email}}">
              </div>

              <div class="form-group">
                <input type="submit" name="" id="reset" class="btn btn-primary" value="Reset Password">
              </div>
            </form>
    </div>

</body>

<script>

   var result=document.getElementById("result").value
   console.log(result)
   if(result =="200"){

      document.getElementById("newp").disabled=false
       document.getElementById("cp").disabled=false
       document.getElementById("reset").disabled=false
       document.getElementById("message").innerHTML="Ready to change Password"

   }
   else{

      document.getElementById("newp").disabled=true
      document.getElementById("cp").disabled=true
      document.getElementById("reset").disabled=true

   }
</script>
</html>
```