

NEWS TRACKER APPLICATION

A PROJECT REPORT

Submitted By

KEERTHANA R	210819205025
NAGESWARI S	210819205031
LIDIYA JONES RAJ S	210819205028
MADLIN JENCY D	210819205029

In partial fulfillment for the award of the degree

Of

BACHELOR OF TECHNOLOGY

In

INFORMATION TECHNOLOGY



KINGS ENGINEERING COLLEGE, IRUNGATTUKOTAI

ANNA UNIVERSITY:CHENNAI 600025

November 2022

BONAFIDE CERTIFICATE

Certified that IBM project report **“NEWS TRACKER APPLICATION”** is bonafide work of **“KEERTHANA R, NAGESWARI S , LIDIYA JONES RAJ S, MADLIN JENCY D”** who carried out this project work under my supervision.

SIGNATURE

Dr.G.MANIKANDAN

HEAD OF THE DEPARTMENT

Professor

Dept of Information Technology,
Kings Engineering College,
Irungattukottai,
Chennai-602 117

SIGNATURE

Mrs. K. SARANYA

MENTOR

Assistant Professor

Dept of Information Technology,
Kings Engineering College,
Irungattukottai,
Chennai-602 117.

ACKNOWLEDGEMENT

We thank God for his blessings and also for giving us good knowledge and strength in enabling us to finish our project. Our deep gratitude goes to our founder late **Dr. D. SELVARAJ, M.A., M.Phil.**, for his patronage in the completion of our project. We like to take this opportunity to thank our honourable chairperson **Dr.S. NALINI SELVARAJ, M.COM., MPhil., Ph.D.** and honourable director, **MR.S. AMIRTHARAJ, M.Tech., M.B.A** for their support given to us to finish our project successfully. We wish to express our sincere thanks to our beloved principal. **Dr.T. JOHN ORAL BASKAR., M.E., Ph.D** for his kind encouragement and his interest towards us.

We are extremely grateful and thanks to our professor **Dr.G.MANIKANDAN**, head of the Information Technology department, Kings Engineering College, for his valuable suggestion, guidance and encouragement. We wish to express our sense of gratitude to our project Mentor **Mrs. K. SARANYA**, Assistant Professor of Information Technology Department, Kings Engineering College and our project Evaluator **Mr. B. MUTHAZAGHAN**, Associate Professor of Information Technology Department, Kings Engineering College, with their guidance and direction made our project a grand success. Also we thank our industrial Mentor **Mrs. SAI PRIYA**, for tuning this project as remarkable one. We Also Thank our College Experimental Project Based Learning SPOC **Dr. Mrs. M. PARAMESHWARI** for managing this initiative in correct direction. We express our sincere thanks to our parents, friends and staff members, who have helped and encouraged us during the entire course of completing this project work successfully.

TABLE OF CONTENTS

CHAPTER	TITLE	PAGE NO
I.	INTRODUCTION 1.1 Project overview 1.2 purpose	06
II.	LITERATURE SURVEY 2.1 Existing System 2.2 References 2.3 Problem Statement Definition	06
III	IDEATION & PRPOSED SOLUTION 3.1 Empathy Map Canvas 3.2 Ideation & Brain storming 3.3 Proposed Solution 3.4 Problem solution fit	07 08 09 10
IV.	REQUIREMENT ANALYSIS 4.1 Functional Requirements 4.2 Non-Functional Requirements	11
V.	PROJECT DESIGN 5.1 Data Flow Diagrams 5.2 Solution & Technical Architecture 5.3 User Stories	12 12 13
VI.	PROJECT PLANNING & SCHEDULING 6.1 Sprint Planning & Estimation 6.2 Sprint Delivery Schedule 6.3 Reports from JIRA	14 14 15
. VII.	CODING & SOLUTIONING 7.1 Feature 1 7.2 Feature 2	18

VIII.	TESTING 8.1 Test Cases 8.2 User Acceptance Testing	18 19
I X.	RESULTS 9.1 Performance Metrics 9.2 Output	20 21
X.	ADVANTAGES & DISADVANTAGES	25
XI.	CONCLUSION	25
XII.	FUTURE SCOPE	25
XIII	APPENDIX <i>Source Code</i> <i>GitHub link</i>	26

TABLE OF FIGURES

SNO	FIGURES	PAGE NO
01	Data Flow & Technical Architecture	12
02	Sprint 1	15
03	Sprint 2	15
04	Sprint 3	16
05	Sprint 4	17
06	Login	20
07	Login Sucessful	21
08	Registration	22
09	Registration Sucessful	22
10	Interface 1	23
11	Interface 2	24

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

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. And **Some of the news apps exaggerating the simple news to be trend in market and they showing inappropriate ads and contents frequently**

2.2 References

- 1.<https://flask.palletsprojects.com/en/2.1.x/tutorial/>
- 2.<https://www.ibm.com/docs/en/db2-for-zos/11?topic=zos-db2-basics-tutorial-working-db2-Interactively>
- 3.<https://www.ibm.com/in-en/products/watson-assistant/docs-resources>
- 4.<https://www.ibm.com/blogs/cloud-archive/2019/03/video-containerization-explained/>
- 5.<https://developer.ibm.com/components/cloud-foundry/tutorials>

2.3 Problem Statement Definition

In today's world people have lots of work to do but there is no time to schedule everything they want. Reading newspaper is not possible for everyone in this urgent world. Using of this app user can easily access and read their interested news by selecting their interested category and get quick feeds for the day. There are multiple news-sharing apps used by a single user and are often spammed with notifications. There also a lot of fake news which gets shared. A news sharing app wants to help users find relevant and important news easily every day and also understand explicitly that the news is not fake but from proper sources. This app will help user to get their interested news by their choice of selecting categories

commodities, sports, technology, education etc. In this app the user can register using their persona details and interacts with the application and save their time and they can segregate the news by their interest.

3. IDEATION & PROPOSED SOLUTION

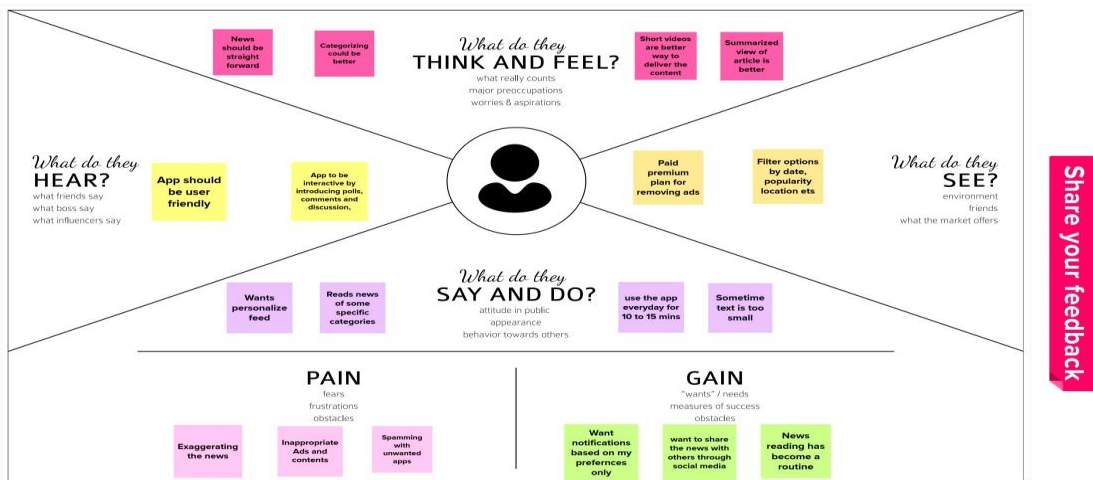
3.1 Empathy Map Canvas

Edit this template
Right-click to unlock

Empathy Map Canvas

Gain insight and understanding on solving customer problems.

1 Build empathy and keep your focus on the user by putting yourself in their shoes.



3.2 Ideation & Brainstorming

Proposed Solution

S. No.	Parameter	Description
1.	Problem Statement (Problem to be solved)	As our lives are very busy these days, we often feel we need more than 24 hrs. a day to cope up with everything we have in our schedule. Well, that's not possible but reducing the time by changing the conventional method of reading news can help. Just tell us what market news you're interested in and get a quick peek for the day. Only read what you feel is relevant and save your time. This app helps you to query for all information about Indices, Commodities, Currencies, Future Rates, Bonds, etc.... as on official websites.
2.	Idea / Solution description	Our idea is we help our users to give their suggested news with their prescribed timing. We give 4 options to every users to choose their suggested news topics like Indices, Commodities, Currencies, Future Rates. If users want short videos for their suggested news we will provide that option also, and our motive is "Only read what you feel is relevant and save your time."
3.	Novelty / Uniqueness	Our uniqueness is all users are same no user is superior than other users we treat all users has same. After many discussion we conclude in our application no premium plans or subscription our entire application is a premium so we didn't want any other plans
4.	Social Impact / Customer Satisfaction	After launching of this application many peoples burden will reduce of this news reading. This will create huge benefit for customers to at what they want to read the news or watching the news in their scheduled time.
5.	Scalability Of Solution	Our application response the user request with in few seconds. Our application response many number of request at a time. So this will ensure our application will scalable.

3.3 Problem Solution fit

Problem-Solution bit canvas 2.0

Purpose / Vision	
1. CUSTOMER SEGMENT(S) CS <p>The person who reads newspaper on daily basis is our customer. He is a person who has news reading on daily basis.</p>	6. CUSTOMER CONSTRAINTS <p>spending power, budget, no cash, network connection, available devices</p>
2. JOBS-TO-BE-DONE/ PROBLEMS <p>As our lives are very busy these days, we often feel we need more than 24 hrs. a day to cope up with everything we have in our schedule. Well, that's not possible but reducing the time by changing the conventional method of reading news can help. Just tell us what market news you're interested in and get a quick peek for the day. Only read what you feel is relevant and save your time. This app helps you to query for all information about Indices, Commodities, Currencies, Future Rates, Bonds, etc. ... as on official websites.</p>	9. PROBLEM ROOT CAUSE <p>The cause of the problem is there are so much of unwanted information are wandered around the world so we spent most of the unwanted news. We acknowledge the problem as soon as possible.</p>
3. TRIGGERS TR <p>If customer see their neighbors installing a very effective news app, searching more about efficient news app.</p>	10. YOUR SOLUTION <p>Our idea is we help our users to give their suggested news with their prescribed timing. We give 4 options to every users to choose their suggested news topics like Indices, Commodities, Currencies, Future Rates. If users want short videos for their suggested news we will provide that option also, and our motive is "Only read what you feel is relevant and save your time."</p>
4. EMOTIONS: BEFORE / AFTER EM <p>Before: They have force to read all the uninterested news. After: which news they want to read they can.</p>	8. CHANNELS of BEHAVIOUR CH <p>ONLINE They should have uninterrupted network connection. OFFLINE They should download their interested news for later use.</p>
5. AVAILABLE SOLUTIONS AS <p>Many paid subscription plans for news app for reading the news.</p>	7. BEHAVIOUR <p>After launching of this application many people burden will reduce of this news reading. This will create huge benefit for customers to at what they want to read the news or watching the news in their scheduled time.</p>

4. REQUIREMENT ANALYSIS

4.1 Functional requirement

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 install the app from Google play store or from the website
FR-2	User Registration	Registration through Gmail
FR-3	User Confirmation	Confirmation via Email Id
FR-4	User Login	User should login the app with the user's name and password credentials

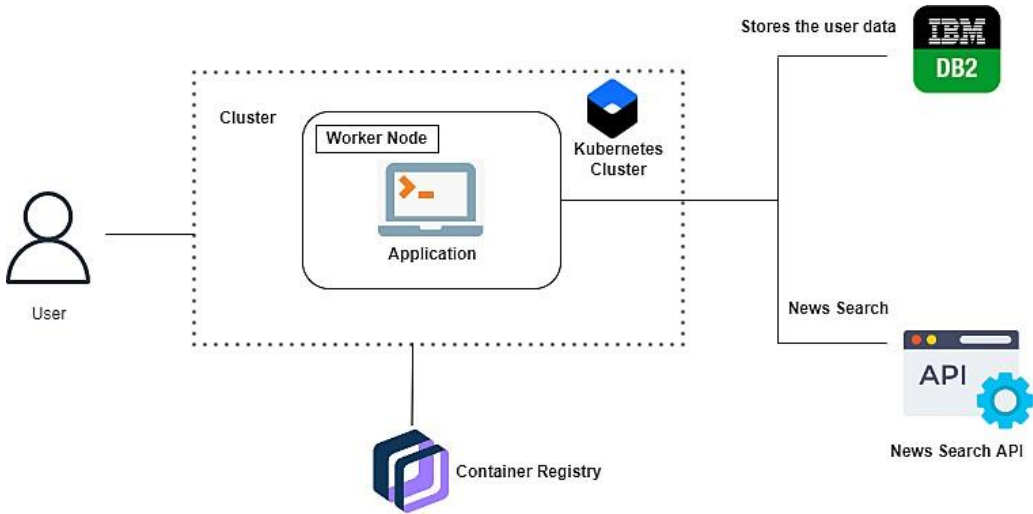
4.2 Non-Functional requirements

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 user guide given in the app.
NFR-2	Security	It is extremely secured app. Fake news cannot be shared.
NFR-3	Performance	Performance of the app is very accurate
NFR-4	Availability	More sub categories are available

5. PROJECT DESIGN

5.1 Data Flow Diagrams



Solution & Technical Architecture

Example - Solution Architecture Diagram: NEWS TRACKER APPLICATION

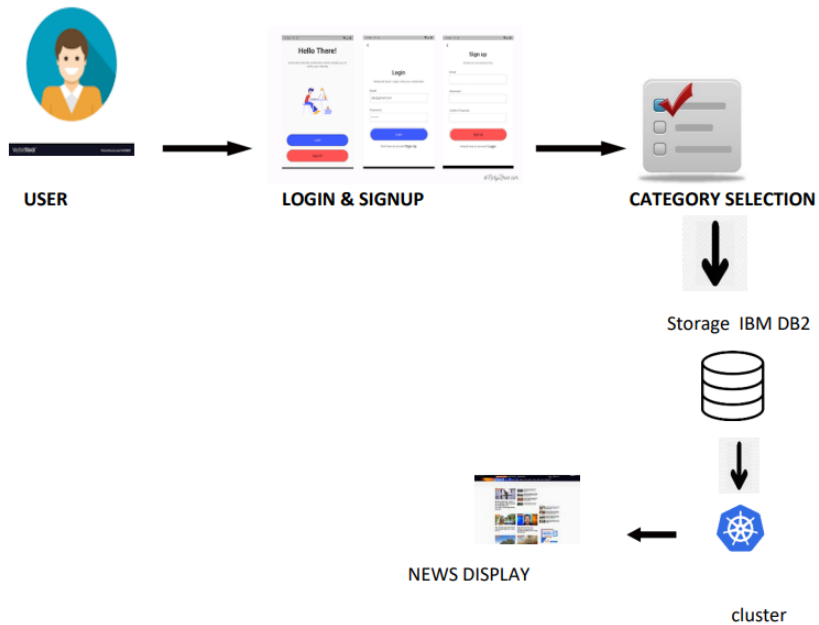


FIG 1: Data flow and Technical Architecture

- 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

User Type	Functional Requirement (Epic)	User Story Number	User Story / Task	Acceptance criteria	Priority	Release
Customer (Web 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 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 news I want	I can read the news elaborately	High	Sprint-4
Customer Care	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

Sprint	Functional Requirement (Epic)	User Story Number	User Story / Task	Priority	Team Members
Sprint-1	User Installation	USN-1	User installation in both website and google play	High	NAGESWAR I
		USN-2	How user interface should be	Low	
Sprint-2	User Register & Login	USN-3	How user registration done through mail	High	MADLIN JENCY
		USN-4	How app should give confirmation via OTP	Low	
		USN-5	How User login done	High	
Sprint-3	User search based	USN-6	How User interest-based News will appear on app Search the news based on the user prescribe topic.	High	LIDIYA JONES RAJ
		USN-7	Showing the Category- wise news for interested domain	Low	
Sprint-4	User Information	USN-8	How users can include their information in preferred topic so that they could be recommended by the application.	High	KEERTHAN A

6.2 Sprint Delivery Schedule

Sprint	Total Story Points	Duration	Sprint Start Date	Sprint End Date (Planned)	Story Points Completed (as on Planned End Date)
Sprint -1	1	3 Days	24 Oct 2022	26 Oct 2022	20
Sprint -2	1	3 Days	31 Oct 2022	02 Nov 2022	20
Sprint -3	1	3 Days	07 Nov 2022	09 Nov 2022	20
Sprint -4	1	3 Days	14 Nov 2022	16 Nov 2022	20

6.3 Reports from JIRA

Sprint-1

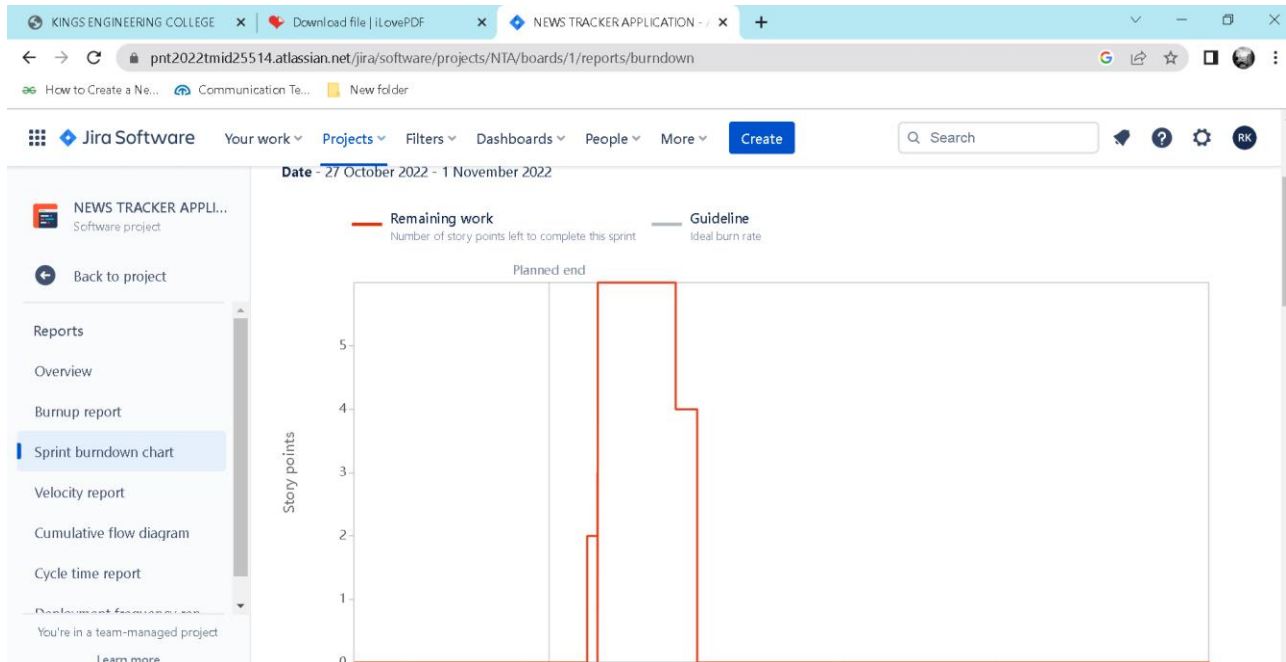


FIG 3: Sprint 1

Sprint-2

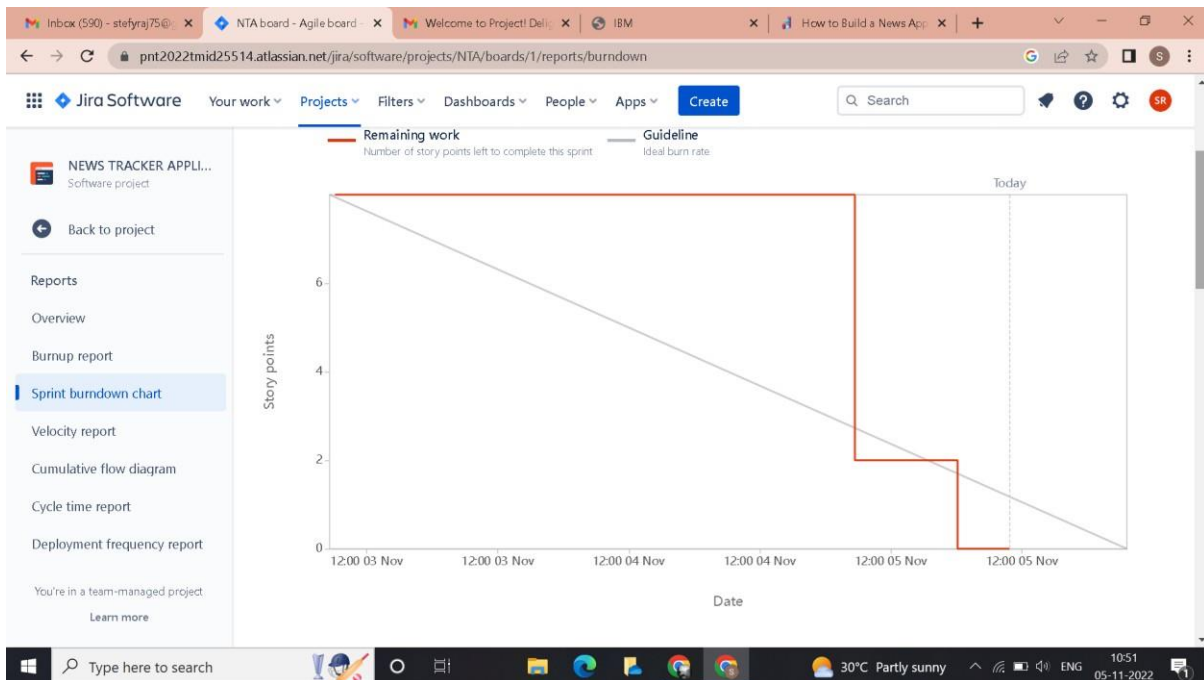


FIG 4: Sprint 2

Sprint-3

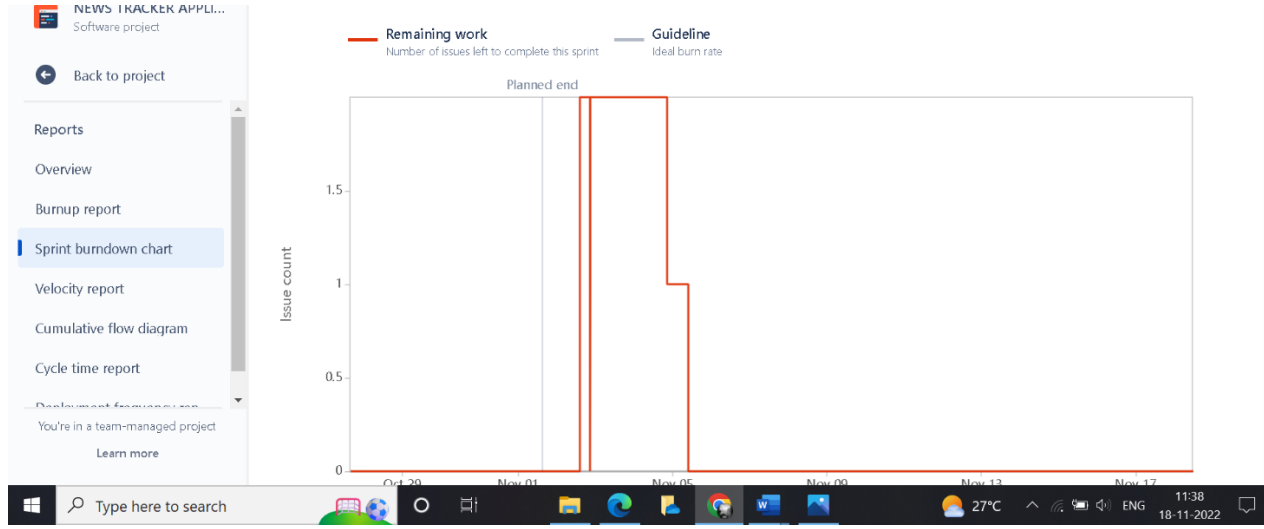


FIG 5: Sprint 3

Sprint-4

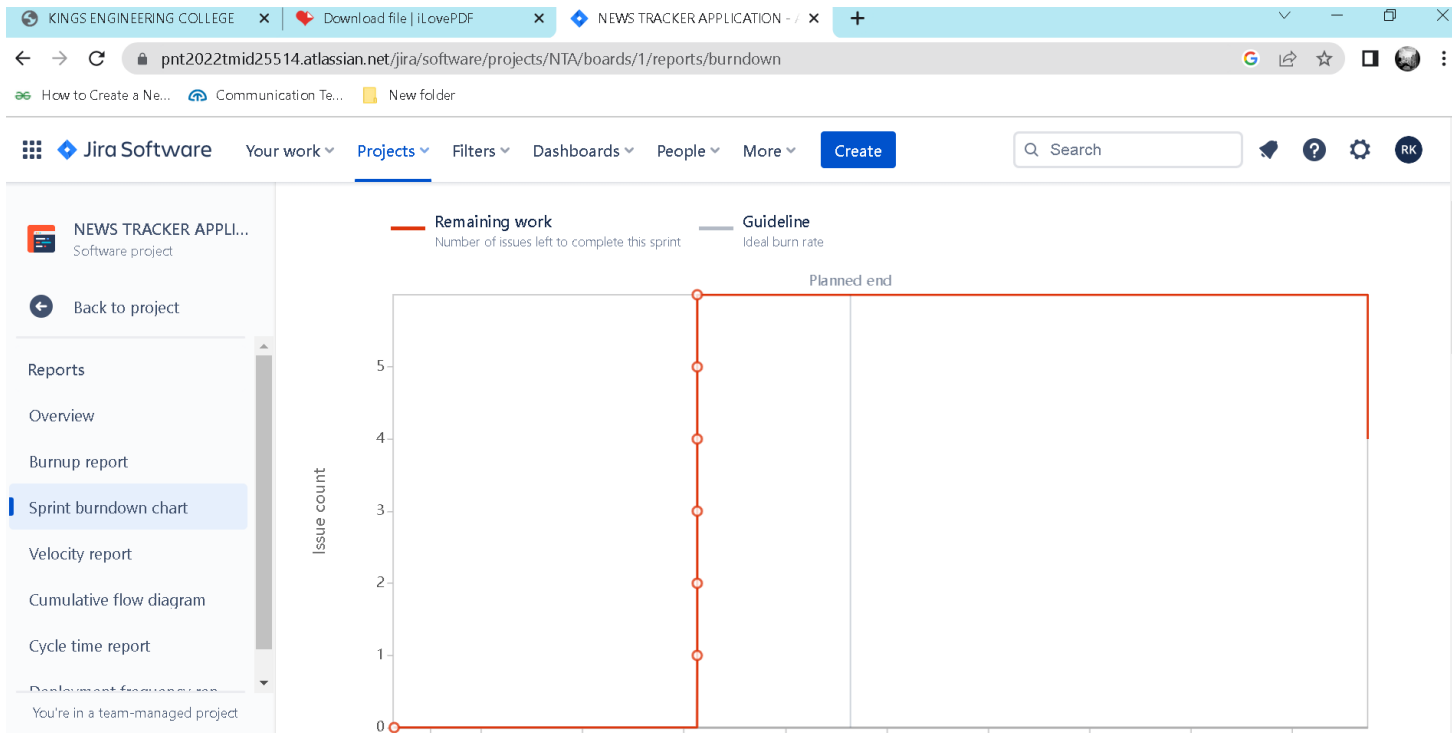


FIG 6: Sprint 4

**Acceptance Testing
UAT Execution & Report Submission**

Date	16 November 2022
Team ID	PNT2022TMID25514
Project Name	Project – News Tracker Application
Maximum Marks	4 Marks

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

1	Team ID - PNT2022TMD25514								
2	NFT - Risk Assessment								
3	Project Name	Scope/feature	Functional Change	Hardware Changes	Software Changes	Impact of Downtime	Load/Volum Changes	Risk Score	Justification
4	New Tracker	New	Moderate	No Change	Moderate	nil	>5 to 10%	ORANGE	Change done when new change is added, track
5									
6	NFT - Detailed Test Plan								
7	S.No								
8	Project Overview								
9	NFT Test approach								
10	Assumptions/Dependencies								
11	Approvals/SignOff								
12	New Tracker								
13	End Of Test Report								
14	Project Overview								
15	NFT Test approach								
16	NFR - Met								
17	Test Outcome								
18	GO/NO-GO decision								
19	Recommendations								
20	Identified Defects								
21	(Detected/Closed/Open)								
22	Approvals/SignOff								
23	No data was found								
24	No data was found								

9.2 Outputs

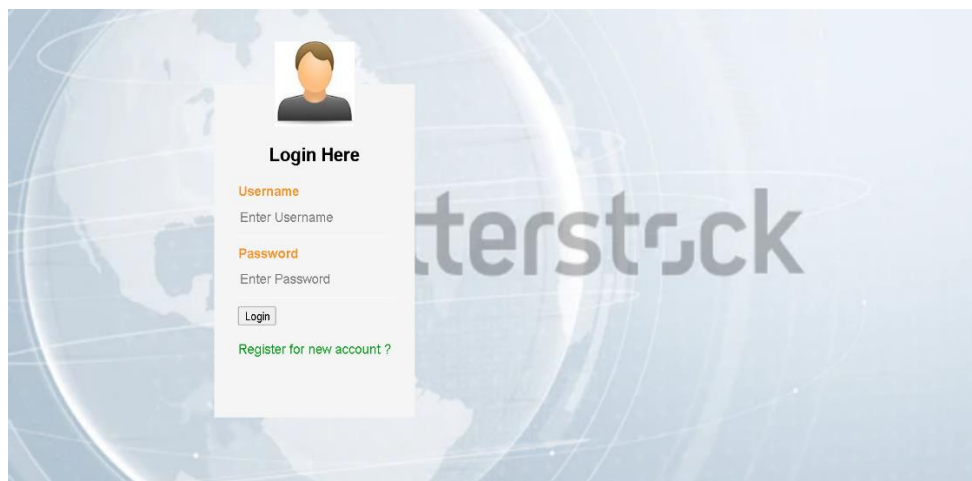


FIG 7:Login

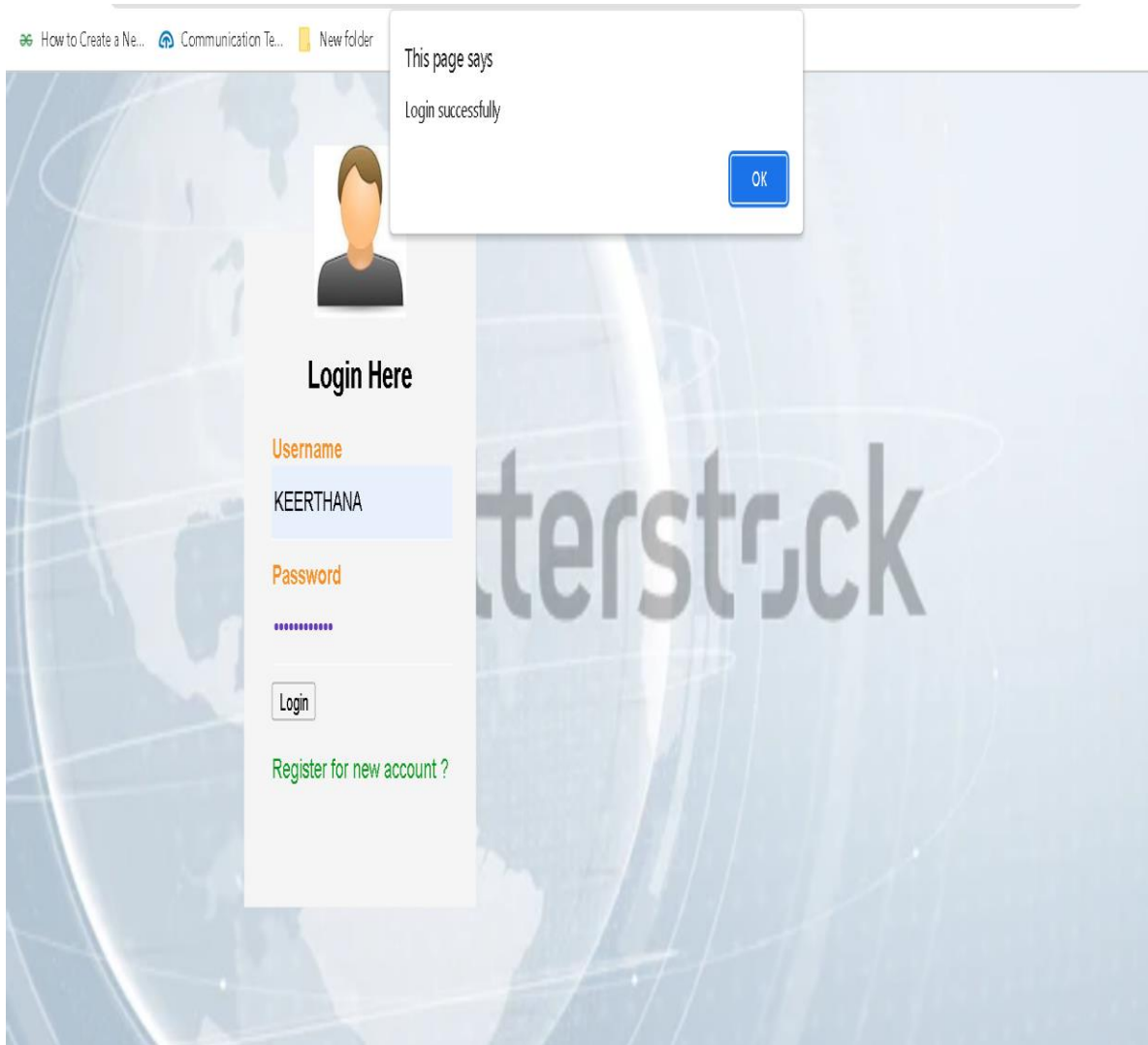


FIG 8: Login Sucessful

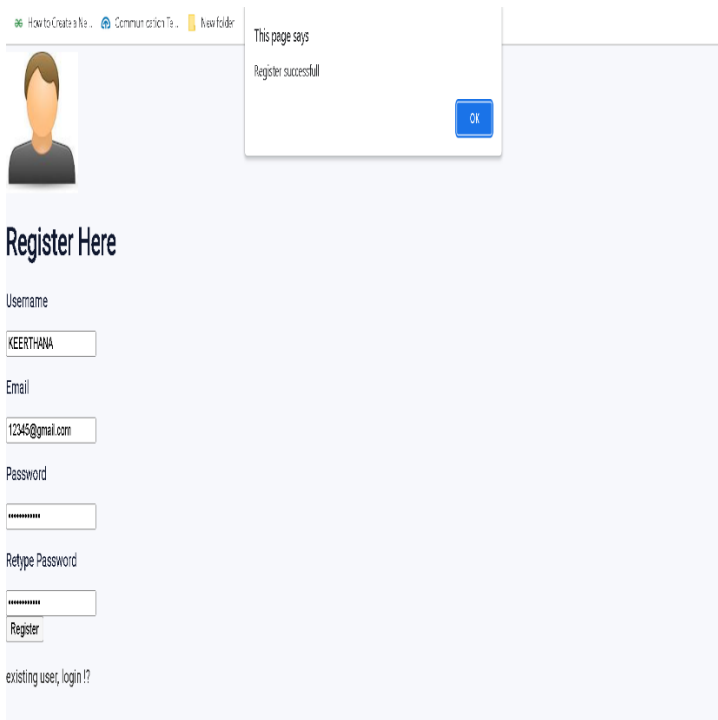


FIG 9: Registration

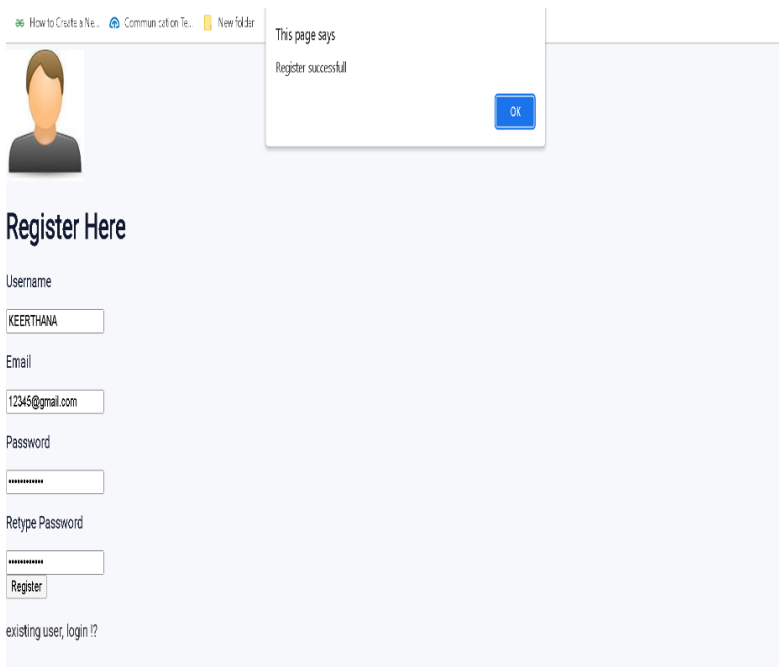


FIG 10:Registration Successful

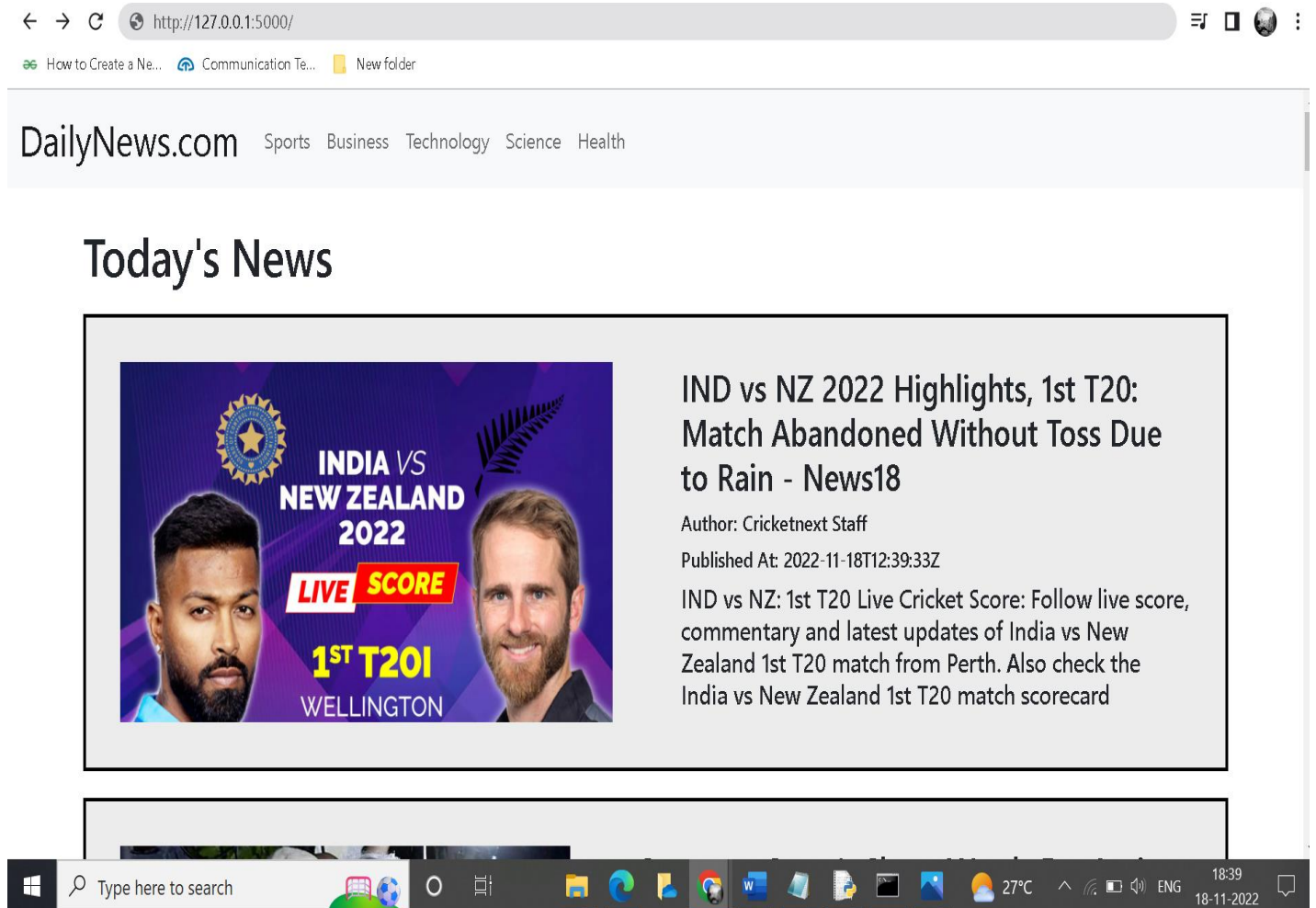


FIG 11: interface 1

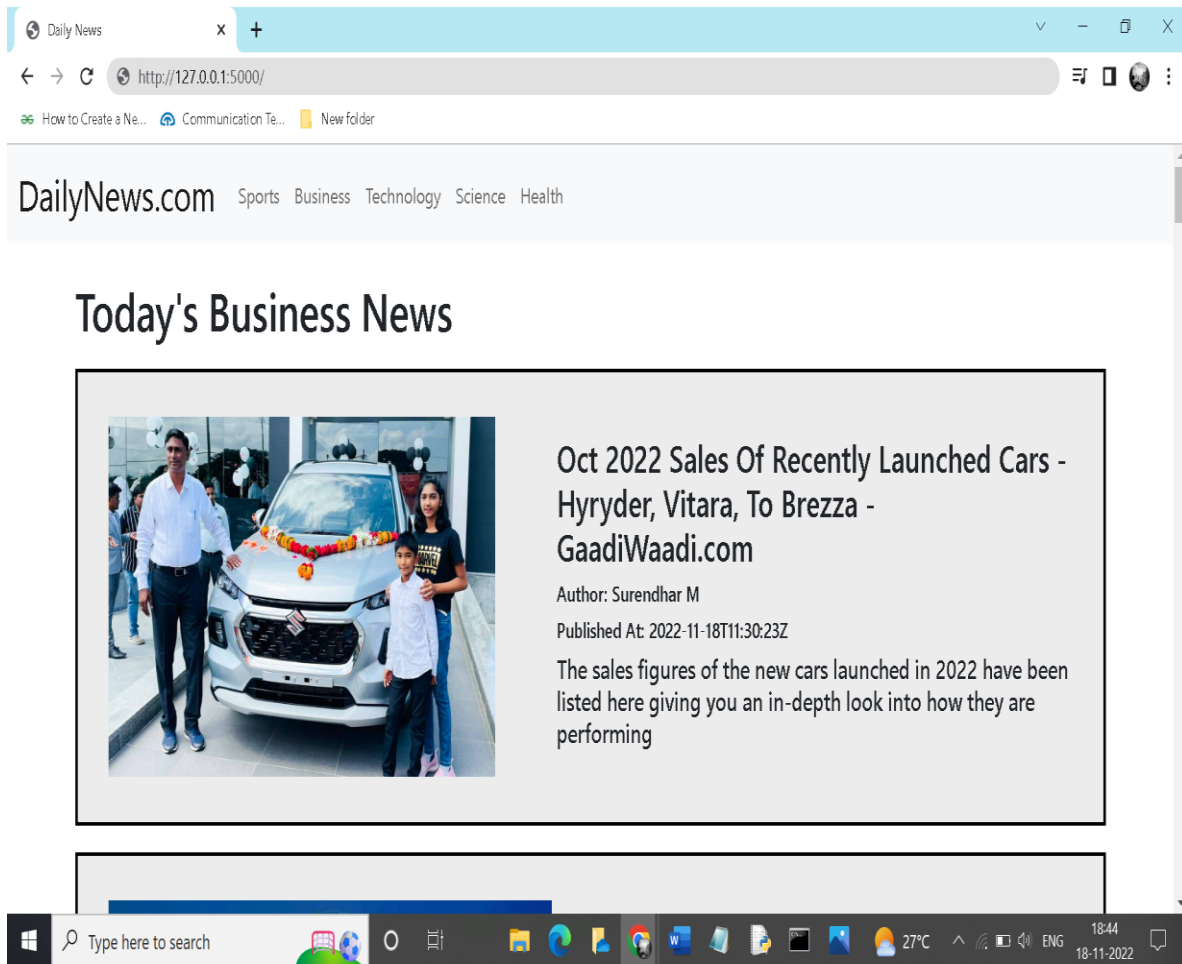


FIG 12: Category wise news

10.ADVANTAGES & DISADVANTAGES

ADVANTAGES

1. All trending news will be displayed in a orderly manner.
2. We can costumize 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.

13. APPENDIX

13.1.Source

Code

Index.html

```
<!DOCTYPE html>
<head>
<title>NEWSIK APP</title>
<link rel="stylesheet" type="text/css" href="indexstyle.css">
<script type="text/javascript" src="js.js"></script>
<body>
<div class="box">

<h1>Login Here</h1>
<form name="myform" onsubmit="return vfun()">
<p>Username</p>
<input type="text" name="uname" placeholder="Enter Username">
<p>Password</p>
<input type="password" name="upswd" placeholder="Enter Password">
<div id="errorBox"></div>
<button class="submit" formaction="https://daily-news-ps.herokuapp.com/">Login</button>
<br><br>
<a href="register.html">Register for new account ?</a>
</form>
</div>
</body>
</head>
</html>
```

Indexstyle.css

```
body{
  margin: 0;
  padding: 0;
  background: url(back.jpg)no-repeat center center fixed;
  -webkit-background-size: cover;
  -moz-background-size: cover;
  -o-background-size: cover;
  font-family: sans-serif;
}
```

```
.box{
```

```
  background: #f5f5f5;
  color: red;
  top: 50%;
  left: 30%;
  position: absolute;
  transform: translate(-50%,-50%);
  box-sizing: border-box;
  padding: 70px 30px;
}
```

```

.user{
  width: 100px;
  height: 100px;
  position: absolute;
  top: -50px;
  left: calc(50% - 50px);
}

h1{
  margin: 0;
  padding: 0 0 20px;
  text-align: center;
  font-size: 22px;
  color: black;
}

p{
  color: #f49126;
  margin: 0;
  padding: 0;
  font-weight: bold;
}

input{
  width: 100%;
  margin-bottom: 10px;
}

input[type="text"], input[type="password"], input[type="email"]
{
  border: none;
  border-bottom: 1px solid #fff;
  background: transparent;
  outline: none;
  height: 40px;
  color: #673ab7;
  font-size: 16px;
}

input[type="submit"]
{
  border: none;
  outline: none;
  height: 40px;
  background: #2196f3;
  color: #fff;
  font-size: 18px;
  border-radius: 20px;
}

```

```
input[type="submit"]:hover
{
    cursor: pointer;
    background: #0097a7;
}
```

```
a
{
    text-decoration: none;
    font-size: 16px;
    line-height: 20px;
    color: #069818;
}
```

```
a:hover
{
    color: red;
}
```

Register.html

```
<!DOCTYPE html>
<head>
<title>Register Form Design</title>
<link rel="stylesheet" type="text/css" href="style.css">
<script type="text/javascript" src="js.js"></script>

<body>

<div class="box">


<h1>Register Here</h1>

<form name="myform2" onsubmit="return vfun1()">

<p>Username</p>
<input type="text" name="uname1" placeholder="Enter Username">

<p>Email</p>
<input type="Email" name="email1" placeholder="Enter email id">

<p>Password</p>
<input type="password" name="upswd1" placeholder="Enter Password">

<p>Retype Password</p>
<input type="password" name="upswd2" placeholder="Re-Enter Password">

<div id="errorBox"></div>
<input type="submit" name="" value="Register">
```

```

<br><br>
<a href="index.html">existing user, login !?</a>
</form>

</div>

</body>
</head>
</html>

```

Js.js

//form 1

```

function vfun(){
var uname=document.forms["myform"]["uname"].value;
var upswd=document.forms["myform"]["upswd"].value;

if(uname==null || uname==""){
document.getElementById("errorBox").innerHTML =
"enter the user name";
return false;
}

if(upswd==null || upswd==""){
document.getElementById("errorBox").innerHTML =
"enter the password";
return false;
}

if (uname != " " && upswd != " "){
alert("Login successfully");
}
}

```

app.py

```

from flask import Flask,render_template, request, jsonify, session, redirect
from flask_mysqldb import MySQL
from utils import category, find
from flask import cli

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

#mysql
app.config['MYSQL_HOST'] = 'localhost'
app.config['MYSQL_USER'] = 'root'
app.config['MYSQL_PASSWORD'] = "
app.config['MYSQL_DB'] = 'flask'

mysql = MySQL(app)

@app.route('/')

```

```

def index():
    return render_template('index.html')

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

@app.route("/log_res", methods = ['POST', 'GET'])
def log_res():
    if request.method == 'GET':
        msg = "Invalid Credentials!!"
        return render_template("register.html", msg = msg)

    if request.method == 'POST':
        try:
            email = request.form['email']
            password = request.form['password']
            cursor = mysql.connection.cursor()
            sql = "SELECT * FROM USER WHERE EMAIL=%s AND PASSWORD=%s",(email, password)
            cursor.execute(sql)
            data = cursor.fetchone()
            dbemail = data[1]
            dbpassword = data[2]

            if password == dbpassword:
                session['status']="ok"
                return render_template("index.html")
            else:
                msg = "Invalid Credentials!!"
                return render_template("register.html", msg = msg)

        except TypeError as e:
            msg = "Invalid Credentials!!"
            return render_template("register.html", msg = msg)

@app.route('/logout')
def logout():
    if 'status' in session:
        session.pop('status',None)
        msg = "logout successful!!"
        return render_template("register.html", msg = msg)
    else:
        msg = "User already logged out!!"
        return render_template("register.html", msg = msg)

@app.route('/home')
def getVariable():
    if 'status' in session:
        return render_template('index.html')
    else:
        msg = "Please Login!!"
        return render_template("register.html", msg = msg)

```

```
if __name__ == '__main__':  
    app.run(port='90', debug=True)
```

GitHub :

<https://github.com/IBM-EPBL/IBM-Project-26566-1660029745>