

Project Report

TEAM ID :PNT2022TMID25237

PROJECT :NEWS TRACKER

1. INTRODUCTION

1.1 Project Overview

1.2 Purpose

2. LITERATURE SURVEY

2.1 Existing problem

2.2 References

2.3 Problem Statement Definition

3. IDEATION & PROPOSED SOLUTION

3.1 Empathy Map Canvas

3.2 Ideation & Brainstorming

3.3 Proposed Solution

3.4 Problem Solution fit

4. REQUIREMENT ANALYSIS

4.1 Functional requirement

4.2 Non-Functional requirements

5. PROJECT DESIGN

5.1 Data Flow Diagrams

5.2 Solution & Technical Architecture

5.3 User Stories

6. PROJECT PLANNING & SCHEDULING

6.1 Sprint Planning & Estimation

6.2 Sprint Delivery Schedule

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

7.1 Feature 1

7.2 Feature 2

7.3 Database Schema (if Applicable)

8. ADVANTAGES & DISADVANTAGES

9. Testing

10. CONCLUSION

11. FUTURE SCOPE 12. APPENDIX

Source Code

Live Page Url

1. INTRODUCTION

1.1 Project Overview

NewsTracker is a fullstack web application which allows users to register along with their favourite topics, upon login the app displays the news based on the user's interest. The news displayed in the app is based on the Newcatcher API and Cricbuzz API from Rapid API site. A news-sharing app wants to help users find relevant and important news easily every day and also provide explicitly news from that users locality/region which may be of help to the user.

1.2 Purpose

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

2. LITERATURE SURVEY

2.1 Existing problem

Physical newspapers are old fashioned in this digital era. They cost money to buy, can easily be damaged, limited amount of information, not flexible to modifications, poor quality. Sometimes may show irrelevant and updated news.

2.2 References

S. no	Paper title	Author	Published date	Implementation	Resource link
1	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)
2	Topic Detection and Tracking in News Articles	Sagar Patel, Sanket Suthar, Sandip Patel, Neha Patel	March, 2015	1. Pre-processing 2. Tokenization 3. Stemming/Lemmatization 4. Vector Space Model 5. Topic tracking	(PDF) Topic Detection and Tracking in News Articles (researchgate.net)
3	An End-to-end Weaklysupervised News Aggregation Framework	Xijin Tang, Xiaohui Huang	June, 2022	The framework combines Snorkel based weakly Supervised classification, 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 classification of	An End-to-end Weakly-supervised News Aggregation Framework Request PDF (researchgate.net)

				<p>unlabeled news with no manual tagging. Subsequently, we utilize LDA to generate the topics and</p>	
				<p>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.</p>	
4	Exploring mobile news reading interactions for news app personalisation	Marios Constantinides ,John Dowell , David Johson, Sylvain Malacria	August, 2015	<ol style="list-style-type: none"> 1. Identification of newsreader types 2. Interaction logging and classification study 3. Deployment and data collection 4. Predicting News reader types 5. Adaptive UI 	(PDF) Exploring mobile news reading interactions for news app personalisation (researchgate.net)

5	Innovative Application For News Tracker	Dr.C.K.Gomathy, Dr.V.Geetha, Peddireddy Abhiram, Marios Constantinides	September, 2020	This paper aimed at developing an online news management system that is of information to either a college. Online news management system provides a simple interface for maintenance of college information. The creation management of accurate, up-to-date information regarding to college. The main objective for developing this project is provide all the functionality related to latest news and it tracks.	(PDF) The Innovative Application for News Management System (researchgate.net)
---	---	---	-----------------	---	--

2.3 Problem Statement Definition

Newspaper contains limited , non user/reader specific, Location specific news . There are multiple news-sharing apps available which can be used by a single user and are often spammed with notifications. There is also a lot of unwanted news which gets shared. So it may take a lot of time for the user to find the news he/she likes. A news-sharing app wants to help users find relevant and important news easily every day and also provide explicitly news from that users locality/region which may of help to the user.

3. IDEATION & PROPOSED SOLUTION


3.1 Empathy Map Canvas



3.2 Ideation & Brainstorming

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

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



Brainstorm & idea prioritization

Use this template in your own brainstorming sessions so your team can unleash their imagination and start shaping concepts even if you're not sitting in the same room.

- 10 minutes to prepare
- 1 hour to collaborate
- 2-8 people recommended

Before you collaborate

A little bit of preparation goes a long way with this session. Here's what you need to do to get going.

10 minutes

A

Team gathering

Define who should participate in the session and send an invite. Share relevant information or pre-work ahead.

B

Set the goal

Think about the problem you'll be focusing on solving in the brainstorming session.

C

Learn how to use the facilitation tools

Use the Facilitation Superpowers to run a happy and productive session.

[Open article](#)

1

Define your problem statement

What problem are you trying to solve? Frame your problem as a How Might We statement. This will be the focus of your brainstorm.

5 minutes

PROBLEM

Newspaper contains limited, non user/reader specific, Location specific news. There are multiple news-sharing apps available which can be used by a single user and are often spammed with notifications. There is also a lot of unwanted news which gets shared. So it may take a lot of time for the user to find the news he/she likes. A news-sharing app wants to help users find relevant and important news easily every day and also provide explicitly news from that users locality/region which may be of help to the user.

Key rules of brainstorming

To run a smooth and productive session

Stay in topic.

Encourage wild ideas.

Defer judgment.

Listen to others.

Go for volume.

If possible, be visual.

Step-2: Brainstorm, Idea Listing and Grouping

2

Brainstorm

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

10 minutes

Jayashakthi Vishnu

Backend LogicAuthenticationTime CalculationMind MappingCompetitive AnalysisPlan of Action

Ranjith

UIUI designUser ExperienceProject ChallengesArticle FiltersPersonalization

Mithun

Easy AccessibleTime DurationCustomer SatisfactionLanguage PreferenceGood IdeasRandomness

Akilan

Attractive UXTemperature ReadingBookmarksMinimal ColoursOptimised RequestResponsive Frontend

3

Group ideas

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

20 minutes

Backend

Handle Multiple RequestFetch News With Regular TimesGather interesting topics from frontend and store it in DBAuthentication and email verification of usersException Handling in the RequestConvert data to standard format for frontend useProcess the data from API callReduce the number of request rem frontend to backend

Frontend

Responsive UI with cool UXMinimalistic Clean UIShow news based on user's interestBookmark the news for users

API Call

TIP

Add customizable tags to sticky notes to make it easier to find, browse, or group, and categorize important ideas as themes within your mural.

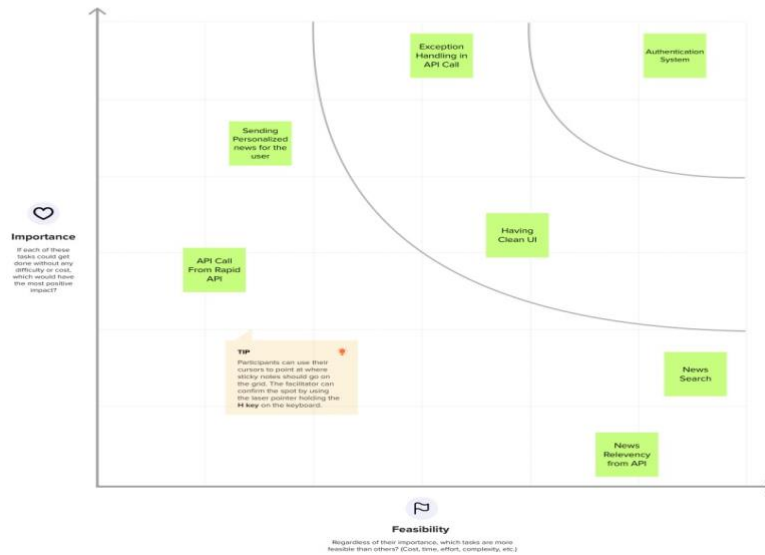
Step-3: Idea Prioritization

3

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



3.3 Proposed Solution

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

3.4 Problem Solution fit

Define CS, fit into CC	1. CUSTOMER SEGMENT(S) <small>Who is your customer? i.e. working parents of 0-5 y.o. kids</small> Common People	6. CUSTOMER CONSTRAINTS <small>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</small> Network Connectivity	5. AVAILABLE SOLUTIONS <small>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 newsreading</small> News apps with lot of advertisements and many irrelevant news	Explore AS, differentiate
	2. JOBS-TO-BE-DONE / PROBLEMS <small>Which jobs-to-be-done (or problems) do you address for your customers? There could be more than one; explore different sides</small> Personalized news for the users, Ad free user interface	9. PROBLEM ROOT CAUSE <small>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</small> Even though there are many news apps over there, most of them are full of ads and irrelevant news.	7. BEHAVIOUR <small>What does your customer do to address the problem and get the job done? Directly related: find the right solar panel installer; calculate usage and benefits; indirectly associated: customers spend free time on volunteering work (i.e. Greenpeace)</small> News channels and newspapers	
Focus on J&P, fit into BE, understand RC	3. TRIGGERS <small>What triggers customers to act? i.e. seeing their neighbour installing solar panels, reading about a more efficient solution in the news</small> Traditional newspaper makes them to receive news at a time delay and most of other online news apps are full of ads.	10. YOUR SOLUTION <small>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.</small> <ul style="list-style-type: none"> Get the user's favourite topics Fetch the news from rapid api Display the news based on user topics 	8. CHANNELS of BEHAVIOUR 8.1 ONLINE <small>What kind of actions do customers take online? Extract online channels from #7</small> View news from the apps like google news, inshort	Identify strong TR & EM
	4. EMOTIONS: BEFORE / AFTER <small>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.</small> Irritated, Difficult > Satisfied, Easy		8.2 OFFLINE <small>What kind of actions do customers take offline? Extract offline channels from #7 and use them for customer development.</small> View news from the newspaper	

4. REQUIREMENT ANALYSIS

4.1 Functional requirement

FR No.	Functional Requirement (Epic)	Sub Requirement (Story / Sub-Task)
FR-1	User Registration	Registration through Email
FR-2	User Confirmation	Confirmation via Email using Sendgrid
FR-3	User Preferred Topics	Collecting user's favourite topics
FR-4	Collecting News	Collecting news from rapid API at regular Intervals
FR-5	Bookmark	Bookmarking Favourite news for the user
FR-6	Login	Logging in the user using the registered email and password
FR-7	News Feed	Showing the news fetched from the API to the user

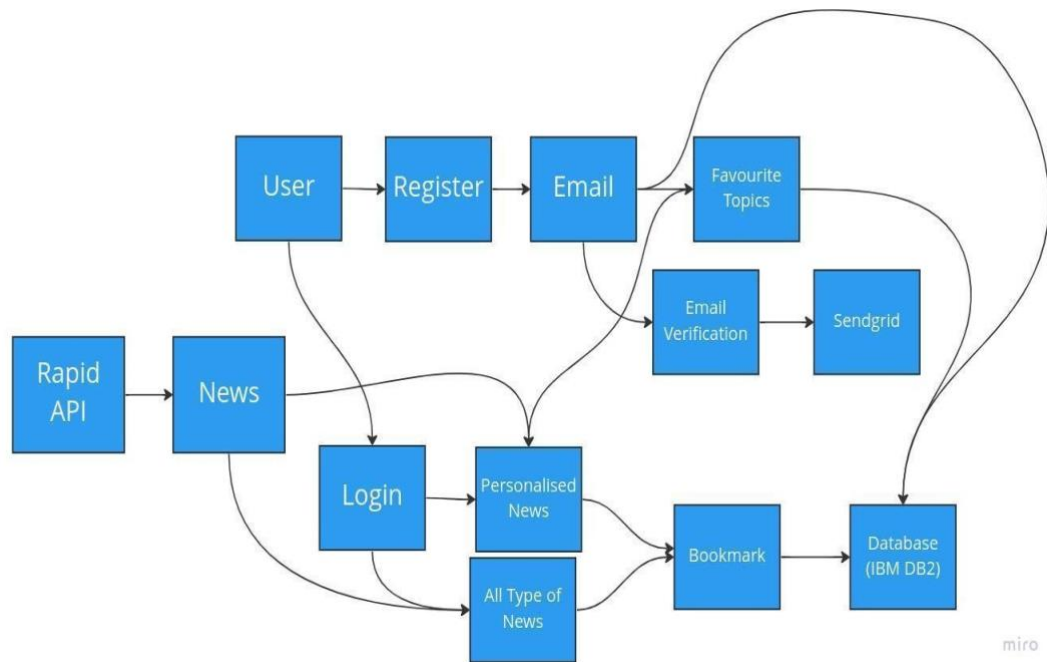
4.2 Non-Functional requirements

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

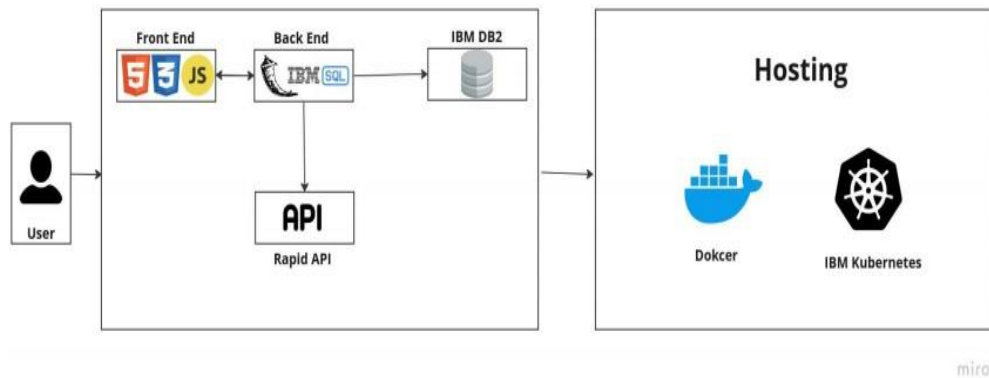
5. PROJECT DESIGN

5.1 Data Flow Diagrams

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



5.2 Solution & Technical Architecture



5.3 User Stories

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

6. PROJECT PLANNING & SCHEDULING

6.1 Sprint Planning & Estimation

Sprint	Functional Requirement (Epic)	User Story Number	User Story / Task	Story Points	Priority	Team Members
Sprint-1	Registration	USN-1	As a user, I can register for the application by entering my email, password, and confirming my password.	10	High	Roshini,Nandhini, Dhanushya, Ramya
Sprint-1	Login	USN-2	As a user, I will receive confirmation email once I have registered for the application	5	High	Roshini,Nandhini, Dhanushya, Ramya
Sprint-1	Email Verification	USN-3	As a user I can verify my email using the link sent to my mailid	5	High	Roshini,Nandhini, Dhanushya, Ramya
Sprint-2	API Fetch	USN-4	Fetch News from Rapid API at regular interval	10	High	Roshini,Nandhini, Dhanushya, Ramya
Sprint-2	REST Endpoints for backend	USN-5	Creating endpoints at the backend inorder to interact with frontend	10	Medium	Roshini,Nandhini, Dhanushya, Ramya
Sprint-3	Designing Frontend	USN-6	Create a minimalisting design in figma to create frontend	2	Medium	Roshini,Nandhini, Dhanushya, Ramya
Sprint-3	Creating Frontend	USN-7	Create the frontend webpage using the design	10	Low	Roshini,Nandhini, Dhanushya, Ramya
Sprint-3	Connect frontend and backend	USN-8	Connect the frontend and backend and complete the application	8	High	Roshini,Nandhini, Dhanushya, Ramya
Sprint-4	Testing	USN-9	Testing the application before final release	10	High	Roshini,Nandhini, Dhanushya, Ramya
Sprint-4	Deployment	USN-10	Deployment of the application	10	High	Roshini,Nandhini, Dhanushya, Ramya

6.2 Sprint Delivery Schedule

[illegible]

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

7.1 Feature 1

Verification email Sender

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

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

7.2 Feature 2

Cookie Checker

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

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

7.3 Database Schema (if Applicable)

User Table

Table definition

USER

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

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

Bookmarks Table:

Table definition

BOOKMARK

Approximate -1 rows (4.00 MB)
Updated on

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

8. ADVANTAGES & DISADVANTAGES

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

9. TESTING • TEST CASE

Test case ID	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
Login Page_TC_OO1	Functional	Home Page	Verify user is able to see the Login/Signup page when user land on the page		1.Enter URL and click go 2.Verify login/Signup page displayed or not		Login/Signup page should display	Working as expected	Pass		N		
Login Page_TC_OO2	UI	Login page	Verify the UI elements in Login/Signup page		1.Enter URL and click go 3.Verify login/Signup page with below UI		Application should show below UI elements: a.email text box	Working as expected	Pass		N		

					elements: a.email text box b.password text box c.Login button d.New customer? SignUp button e.News topics buttons		b.password text box c.Login button d.New customer? Sign UP button. e.Favourite topics card in signup section.					
Login Page_TC_003	Functional	Login page	Verify user is able to log into application with Valid credentials		1.Enter URL and click go 2.Enter the credentials in the text box. 3.If the user is verified you will be redirected to the home page.	User name: test@gmail.com password: Testin g123	User should be redirected towards the home page.	Working as expected	P a s s		N	

Login Page _TC_ OO4	Function al	Lo gin pa ge	Verify user is able to log into applicati on with Invalid credentia ls		1.Enter URL and click go 2.Click on button 3.Enter Valid userna	Use rname: test@ gmail. com passw ord: Testin g6	Applicati on should show 'Incorrec t email or passwor d '	W or ki ng as ex pe ct ed	P a s s		N		
------------------------------	----------------	-----------------------	--	--	---	--	---	---	------------------	--	---	--	--

					me/em ail in Email text box 4.Enter Invalid passwor d in passwor d text box 5.Click on login button		validatio n message .						
Hom ePag e_TC_ _001	Fuction al	Ho me pa ge	Verify user is able to see the news page		1. Login and go to the site		Applicati on should show the news	W or ki ng as ex pe ct ed	P a s s		N		
Hom ePag e_TC_ _002	Function al	Ho me pa ge	Verify user is able to view news under specific categories		1. Go tothe site 2. Click on topics button		Applicati on should show news releated to that topic	W or ki ng as ex pe ct ed	P a s s		N		

HomePage_TC_003	Functional	Homepage	Verify user is able to view recommended news		1. Go to the site 2. Click on recommended tab		Application should show the news based on user's interest	Working as expected	Pass		N		
-----------------	------------	----------	--	--	--	--	---	---------------------	------	--	---	--	--

HomePage_TC_004	Functional	Homepage	Verify user is able to bookmark a product		1. Go to the site 2. Click on bookmark icon in the news card.		Application should show saving icon and the news is bookmarked	Working as expected	Pass		N		
HomePage_TC_005	Functional	Homepage	Verify user is able to navigate to the news article page		1. Go to the site 2. Click on any news card		Application should navigate to the article page where the actual article is displayed.	Working as expected	Pass		N		
BookmarkPage_TC_001	Functional	Bookmark page	Verify user is able to view the bookmarks		1. Go to the site 2. Click on bookmarks tab under the user icon		Application should view all the user bookmarks	Working as expected	Pass		N		

Book mark Page _TC_ 002	Function al	Bo ok ma rk pa ge	Verify user is able to delete a bookamar k		1. Go tothe bookam arks page. 2. Clickon the bookma rk icon in the page		Applicati on should show the deleting icon and the bookma rk should be	W or ki ng as ex pe ct ed	p a s s		N		
							deleted.						
News Page _TC_ 001	Function al	Ne ws pa ge	Verify user is able to view the news		1. Go tothe site 2. Clickon a news card		Applicati on should show the news article	W or ki ng as ex pe ct ed	p a s s		N		
News Page _TC_ 002	Function al	Ne ws pa ge	Verify user is able to view page not loading text		1. Go tothe site 2. Clickon a news card		Applicati on should show page not loading text	W or ki ng as ex pe ct ed	p a s s		N		
News Page _TC_ 003	Function al	Ne ws pa ge	Verify user is able to view the news article by click on page not loading text		1. Go tothe site 2. Clickon a news card 3. Click on page not loading text.		Applicat oin should redirect to the actual article page	W or ki ng as ex pe ct ed	p a s s		N		

Logout_TC_001	Functional	Logout	Verify user is able to logout of the application		1. Go to the site 2. Click on user icon 3. Click on logout		Application should logout the user.	Working as expected	Pass		N	
---------------	------------	--------	--	--	--	--	-------------------------------------	---------------------	------	--	---	--

- USER ACCEPTANCE TESTING:**

1. 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	0	0	0	0	0
Duplicate	0	0	2	0	2
External	0	0	0	0	0
Fixed	0	0	0	3	0
Not Reproduced	0	0	0	1	1
Skipped	0	0	0	0	0
Won't Fix	0	0	1	0	1
Totals	0	0	3	4	4

2. 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	4	0	0	4
Home	5	0	0	5
Bookmark	2	0	0	2
News	3	0	0	3
Logout	1	0	0	1

- PERFORMANCE TESTING:**

