PROJECT REPORT ON

NEWS TRACKER APPLICATION

CLOUD APPLICATION DEVELOPMENT

TEAM ID : PNT2022TMID47959

SUBMITTED BY

KEERTHAN S - 912219104018

RAMYA D - 912219104026

SHOBANA M - 912219104032

SWETHA K - 912219104035

TABLE OF CONTENTS

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 &TechnicalArchitecture
    5.3.User Stories

6. Project Planning & Scheduling
    6.1.Sprint Planning & Estimation
    6.2.Sprint Delivery Schedule
    6.3.Reports From Jira

7. Coding & Solutioning (explain The Features Added In The Project Along With Code)
    7.1.Feature 1
    7.2.Feature 2

8. Testing 8.1.Test Cases 8.2.User Acceptance Testing
9. Results 9.1.Performance Metrics
10. Advantages & Disadvantages
11. Conclusion
12. Future Scope
13. Appendix Source Code

1.INTRODUCTION

1.1 Project Overview

NewsTracker is an application designed to enhance and optimize the way a user interacts with news stories. This is achieved primarily using a search engine that connects to NewsAPI which allows the user to narrow-down the content of their results.Additional search parameters such as Polarity and Subjectivity use Natural Language Processing through Python's NLTK and Textblob libraries to parse through the html data given by the URL returned by NewsAPI, which will order a user's search results by the degree to which an article is objective or positive, with the corresponding score listed for each article.Other features available to users include the ability to save stories that they want to refer back to later, save multiple search queries for easy-access through an accordian drop-down, and the ability to set one of said queries as a user's default so that their headline feed will be base its results off that query default so that the main headline page will show results based off of the user's default search query.

1.2 Purpose

The ability to run an advanced search query to display a specific type of story and store it permanently in a user's account could be beneficial for anyone wanting to keep up with current events and/or investment opportunities.An investor is trying to assess the pros and cons of owning stock in a particular company. Using the subjectivity feature to identify degrees of subjectivity, they may choose to filter out articles that are marked as highly subjective, which allows them to save researching time and become better informed. Someone wanting to become more politically informed about a particular issue may use the subjectivity feature to identify news sources and types of stories that may not be worth reading due to the degree to which the author's tone skews from being objective. A day/swing trader may want to use the polarity feature to evaluate the overall market sentiment of aparticular stock, crypto, or geo-political region. An individual has been feeling that the amount of negative news they've been consuming has taken a toll on their mental health, and want to expose themselves to more positive and uplifting news may use the polarity feature to filter out negative stories.Although there are many different use cases that type of application might provide to its users, as described above, I also think it is important to mention that current NLP Sentiment Analysis capabilities aren't perfect, and a user should not expect to get accurate results or insights 100% of the time.

2. Literature Survey

2.1 Existing problem

Read online articles without any annoying contents .But there are multiple news-sharing apps used by a single user and are often spammed with notifications. There is also numerous fake news which gets shared. A news-sharing app wants to help users find relevant and important news easily every day and understand explicitly that the news is not fake but from proper sources. Spamming of messages usually leads to clearing of the content without viewing thus probably leading the user to lose access to important information. Businesses must publish irrelevant news because younger generations prefer news with more fun instead of reliable news. Read online articles without any advertisements because Advertisements in the apps might irritate the user while reading the news. Apps generate income through subscriptions and advertisements. Read precise contents because News apps want to increase the time that user spends on their app so that they can show ads and generate revenue. Avoid irrelevant news, Irrelevant news makes the user stop viewing the news thus losing access to credible news. Users don't want to spend time reading the entire content. Users only check their phones and go through these notifications when they are free or during a particular time of the day when they are pass. One app not accommodating both the regional and international news might create a requirement for different apps. Businesses have to publish irrelevant news because younger generations prefer news with more fun instead of reliable news. News apps are trying to be like social media apps.

2.2 References

Author	Martijn Kleppe and Marco Otte
Title	Analysing and understanding news consumption patterns by tracking online user
Year	2017

Understanding users' online behaviour is of growing interest to academic researchers in a variety of fields. The Newstracker is a custom-built system that collects Web activities of specified and authenticated users, cleans the data by removing non-relevant data, extracts the associated content, and stores this as a new data set to be used for analysis. To create the Newstracker, we applied two out of several data mining techniques: Web usage mining and Web content mining. While designing the Newstracker, we encountered several issues particularly during the pre-processing phase. First, caching is cited as an issue for reliably collecting user Internet use. When using a cached Web page, a browser will not request the page or elements of the page and thus not produce the HTTP request needed for the proxy server to log activity. The basis of the Newstracker was a local proxy. Even though applying a proxy to monitor Web behaviour is a relatively easy and a low-cost solution, it does ask some technical knowledge of

the respondent. A second methodological challenge is setting up the whitelist of websites that need to be monitored. A third and possibly most important methodological challenge is to find respondents who are willing to have their online behaviour being monitored. In the data analyses phase, we expect the future analyses of the scraped content will be a challenge. Especially researchers who are monitoring the behaviour of large amounts of respondents during longer period will gather large amounts of scraped content.

Author	Swit Phuvipadawat, Tsuyoshi Murata
Title	Breaking News Detection and Tracking in Twitter
Year	2010

Twitter is a social networking service that allows users to share information, which is described by Twitter as “What’s happening?” in a form of short texts (140 characters). Keywords-Twitter, Topic Detection and Tracking, Real-time text-mining, Information Retrieval. Breaking news is defined by Wiktionary as “news that has either just happened or is currently happening. Breaking news may contain incomplete information, factual error or poor editing because of rush.” With this definition Twitter can fit the needs of breaking news delivery. However, news posted in Twitter requires an effort to discover it. Firstly, users often have problems of deciding which users to follow. That is, to find users with interesting tweets. Secondly, users need to read through status updates and follow links to obtain further information. There are two important elements in a message: emotions and facts. The inclusion of emotions in the message makes news delivered in Twitter, different from news delivered by professional journalists. In this paper, we present a method to collect, group, rank and track breaking news. Tasks are divided into two stages: story finding and story development. In this paper, we focus on facts part of messages. Emotions are left for our future works. Tasks are presented in three steps: sampling, indexing and grouping. We use the Stanford Named Entity Recognizer (NER) for the classification of proper nouns. NER provides a general implementation of linear chain Conditional Random Field (CRF) sequence models, coupled with well-engineered feature extractors for Named Entity Recognition. In this work, we discussed the characteristics of breaking news in Twitter and presented a method to collect, group, rank and track breaking news from Twitter. To improve the similarity comparison for short-length messages, we emphasized proper nouns. There are several factors to rank the news story. In this experiment we use reliability, popularity and behave for the ranking factors. With these counterparts, we can understand the user’s perception to news stories, impacts to the mass audience and the pattern in which the information spreads

Author	

	Milos Krstajic´, Mohammad Najm-Araghi, Florian Mansmann and Daniel A Keim
Title	Story Tracker: Incremental visual text analytics of news story development
Year	2013

.

Understanding temporal development of unstructured and semi-structured text data streams is becoming increasingly important in many application areas, such as journalism, politics, or business intelligence. Keywords are News stream analysis, topic evolution, dynamic visualization, text analytics. Research on topic development, visualization, and analysis of temporal dynamics in information streams has strong connections to the fields of text mining, information visualization, and time-series analysis. A well-known initiative in text mining was topic detection and tracking (TDT),1 which investigated methods for discovering events in broadcast news streams. The news articles are collected as XML data, as described by Krstajic´ et al. Each timestamped data item contains meta data, such as named entities or tags. Entities identify people or organizations that are mentioned in the document, while tags categorize the news (e.g., earthquake, sports). Very often, news stories have braided nature, that is, documents that belong to different clusters can be still highly related. Furthermore, news stories may split into two different topics at some point. The visualization in our system can be seen as like parallel coordinates, which is a well-known technique for visualizing highdimensional data. The global overview visualization helps in identifying the major news stories over a long period of time, and it is enriched with the interaction techniques to filter and rerank stories on various user-adjustable criteria in order to provide a clutter-free display.

2.3 Problem Statement Definition

Jack is a businessman . User user charaecteristics Who needs to get the information about News . User need Because he interested to know current affairs and want to update himself .

NEWS TRACKER APPLICATION

I am	Regular news reader in online news tracker applications
I'm trying to	Read online articles without any annoying contents
But	● There are multiple news-sharing apps used by a single user and are often spammed with notifications. ● There is also numerous fake news which gets shared. ● A news-sharing app wants to help users find relevant and important news easily every day and understand explicitly that the news is not fake but from proper sources.
Because	● News apps are trying to be like social media apps ● News apps want to increase the time that user spends on their app so that they can show ads and generate revenue ● To increase the user screen time, news apps make users encounter eye-catching news rather than credible ones ● Apps generate income through subscriptions.
Which makes me feel	● Fake news hurts individuals and society as it persuades consumers to accept false

beliefs that are shared to forward specific agendas.

● Identifying relevant news from excessive amounts of information on social media requires substantial time, energy, and mental efforts

● Constant news updates and pop-ups of breaking news in social media may increase the feeling of news overload.

“Real-time Customer Problem Statement”

I am	I'm trying to	But	Because	Which makes me feel
online news articles reader	Read online articles without any annoying contents	spamming of messages usually leads to clearing of the content without viewing thus probably leading the user to lose access to important information.	Business must publish irrelevant news because younger generations prefer news with more fun instead of reliable news.	irritated
	Read online articles without any advertisements	Advertisements in the app might irritate the user while reading the news.	Apps generate income through subscriptions and advertisements	Frustrated.
	Read precise contents	News apps want to increase the time that user spends on their app so that they can show ads and generate revenue	Users don’t want to spend time reading the entire content. They need short and crisp news	Annoying
	Avoid irrelevant news	Irrelevant news makes	Businesses	Exhausted

		the user stop viewing the news thus losing access to credible news	must publish irrelevant news because younger generations prefer news with more fun instead of reliable news.	

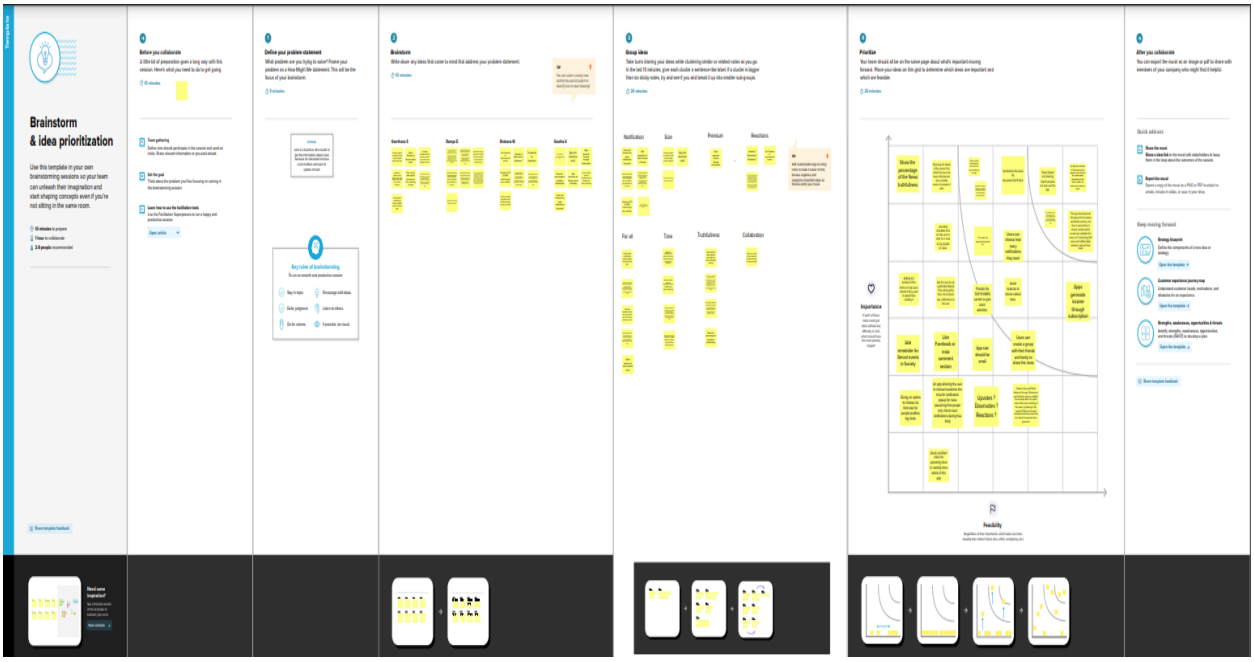
3. IDEATION & PROPOSED SOLUTION

3.1 Empathy Map Canvas

An empathy map is a simple, easy-to-digest visual that captures knowledge about a user's behaviours and attitudes. It is a useful tool to helps teams better understand their users. Creating an effective solution requires understanding the true problem and the person who is experiencing it. The exercise of creating the map helps participants consider things from the user's perspective along with his or her goals and challenges.

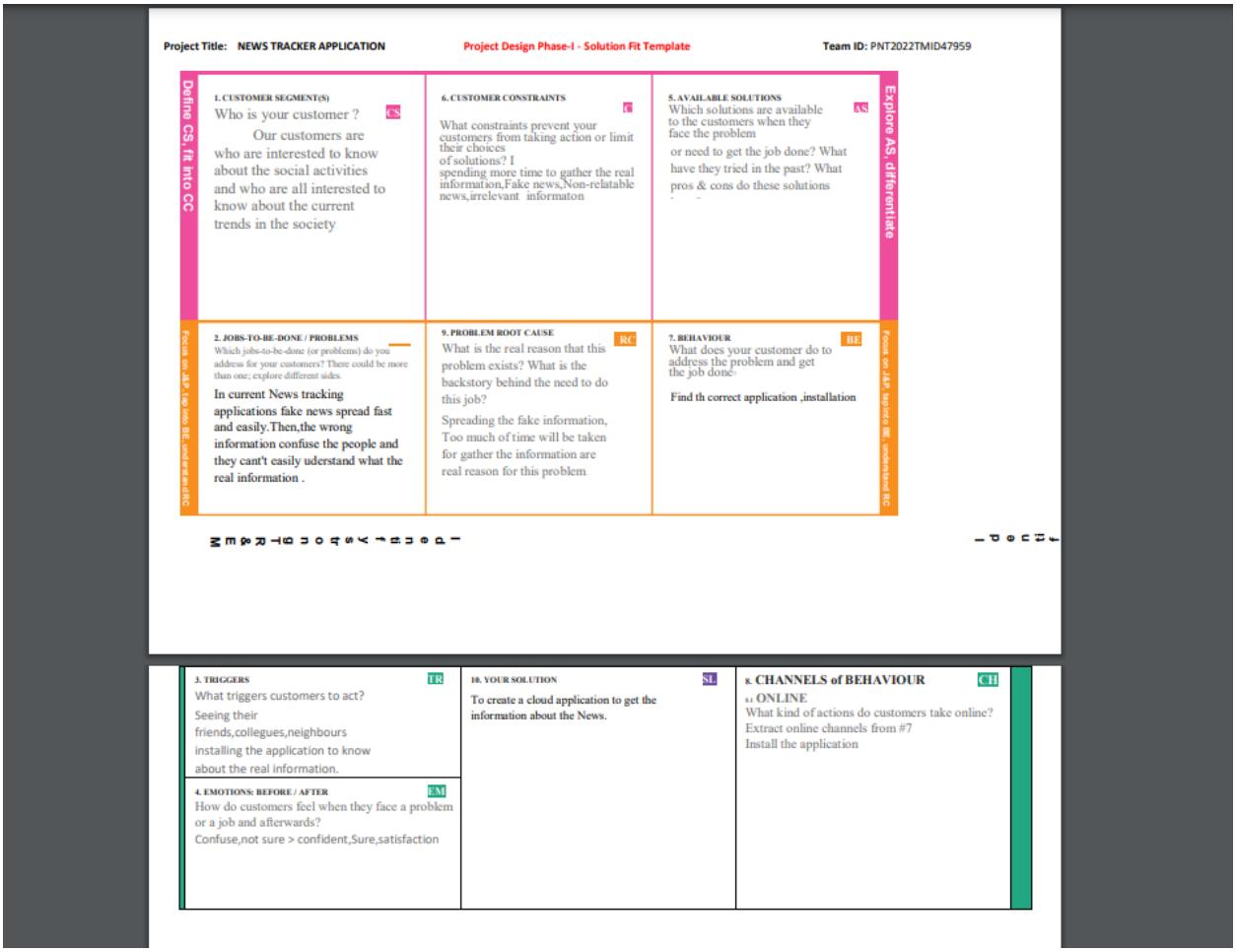
3.2 Ideation & Brainstorming

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, outof-the-box ideas are welcome and built upon, and all participants are encouraged to collaborate, helping each other develop a rich number of creative solutions.



3.4 Problem Solution Fit

The Problem-Solution Fit simply means that you have found a problem with your customer and that the solution you have realized for it actually solves the customer's problem .



4. Requirement Analysis

4.1 Functional Requirement

Functional requirements are product features or functions that developers must implement to enable users to accomplish their tasks. So, it is important to make them clear both for the development team and the stakeholders. Generally, functional requirements describe system behaviour under specific conditions.

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 mail account or mobile number
FR-2	User Confirmation	Confirmation via Email Confirmation via OTP
FR-3	User Region	Selects user's current region or country and further local locations.
FR-4	Language Selection	Selecting up language on which the articles to viewed
FR-5	Topic/Category Selection	Selecting topic/categories like technology, science, politics, moves, health care, space, games, etc…
FR-6	News Source Selection	Choosing up of sources from where the news articles to be displayed like msn, times now, the Hindu, CNBC news, etc
FR- 7	Notification preferences	Decide on which the notification can be popped up.

4.2 Non-functional Requirements

Non-functional Requirements define system attributes such as security, reliability, performance, maintainability, scalability, and usability. They serve as constraints or restrictions on the design of the system across the different backlogs.

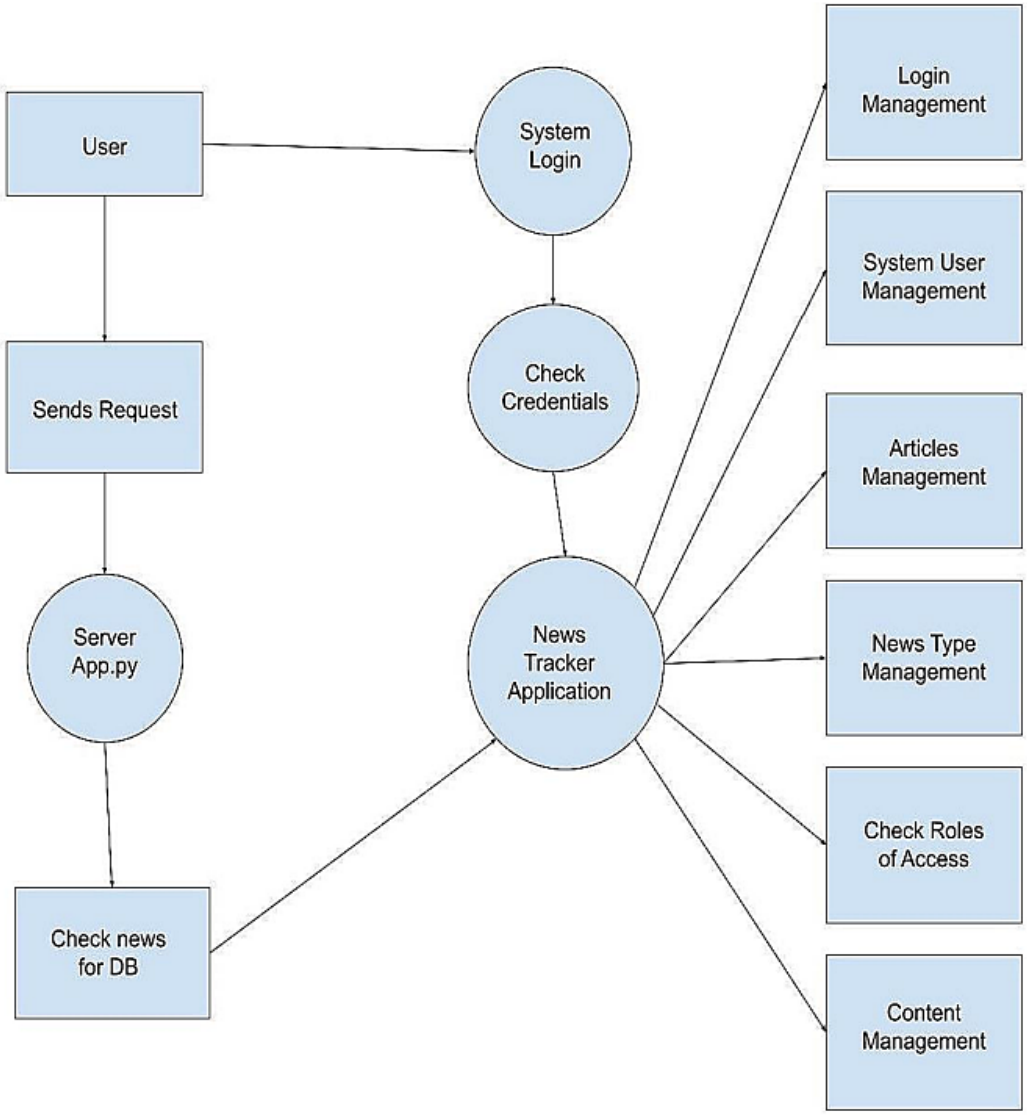
Following are the non-functional requirements of the proposed solution.

NFR No.	Non-Functional Requirement	Description
NFR-1	**Usability**	Easy to use instead of reading by own read a loud option helps the user to read the contents.
NFR-2	**Security**	Securely verified at every stage before publishing of the contents
NFR-3	**Reliability**	Provides accurate new articles and news contents
NFR-4	**Performance**	Contents are as soon as accessible by everyone when the publisher publishes it
NFR-5	**Availability**	Only available to the user if and only if the news is identified as true
NFR-6	**Scalability**	Easy to extend the already customized news categories simply by editing the existing one.

5. Project Design

5.1 Data Flow Diagram

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

● Solution architecture, term used in information technology with various definitions such as "A description of a discrete and focused business operation or activity and how IS/IT supports that operation".

● Solution architecture provides the ground for software development projects by tailoring IT solutions to specific business needs and defining their functional requirements and stages of implementation. It is comprised of many subprocesses that draw guidance from various enterprise architecture viewpoints.

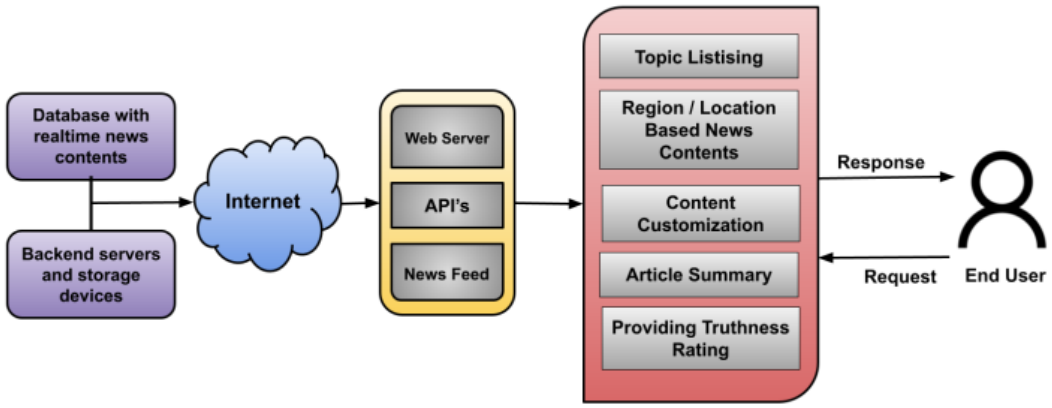
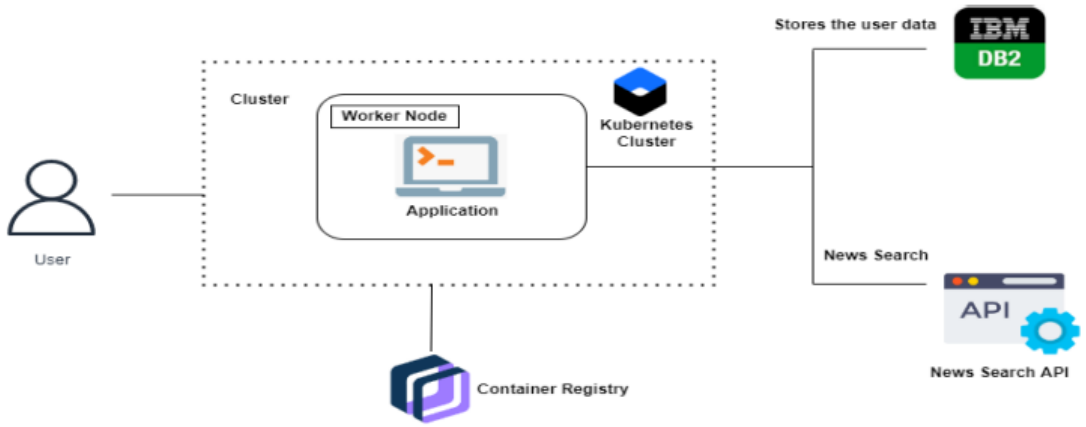


Figure: News Tracker Application – Solution Architecture

Technology architecture deals with the deployment of application components on technology components. A standard set of predefined technology components is provided to represent servers, network, workstations, among other examples



6. Project Planning & Scheduling

6.1 Sprint Planning & Estimation

Sprint	*Total Story Points*	*Duration*	*Sprint Start Date*	*Sprint End Date (Planned)*	*Sprint Release Date (Actual)*
Sprint-1	20	6 Days	24 Oct 2022	29 Oct 2022	29 Oct 2022
Sprint-2	20	6 Days	31 Oct 2022	05 Nov 2022	05 Nov 2022
Sprint-3	20	6 Days	07 Nov 2022	12 Nov 2022	12 Nov 2022
Sprint-4	20	6 Days	14 Nov 2022	19 Nov 2022	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)

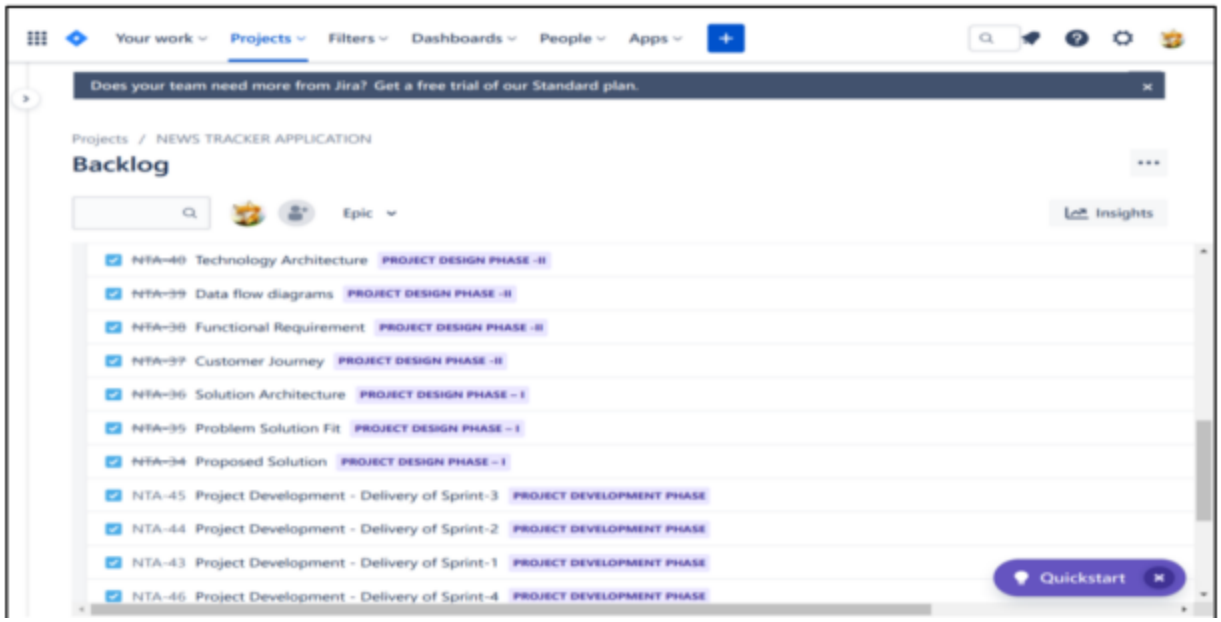
Burn-down Chart:
A burn down chart is a graphical representation of work left to do versus time. It is often used in agile software development methodologies such as Scrum. However, burn down charts can be applied to any project containing measurable progress over time.

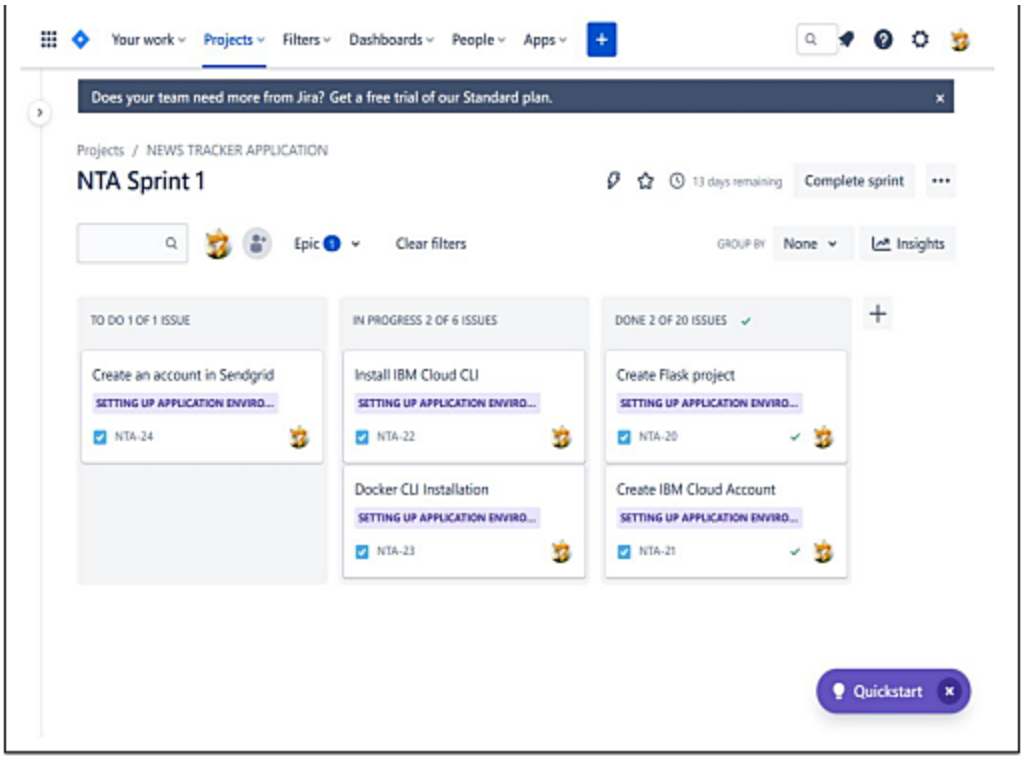
6.2 Sprint Delivery Schedule

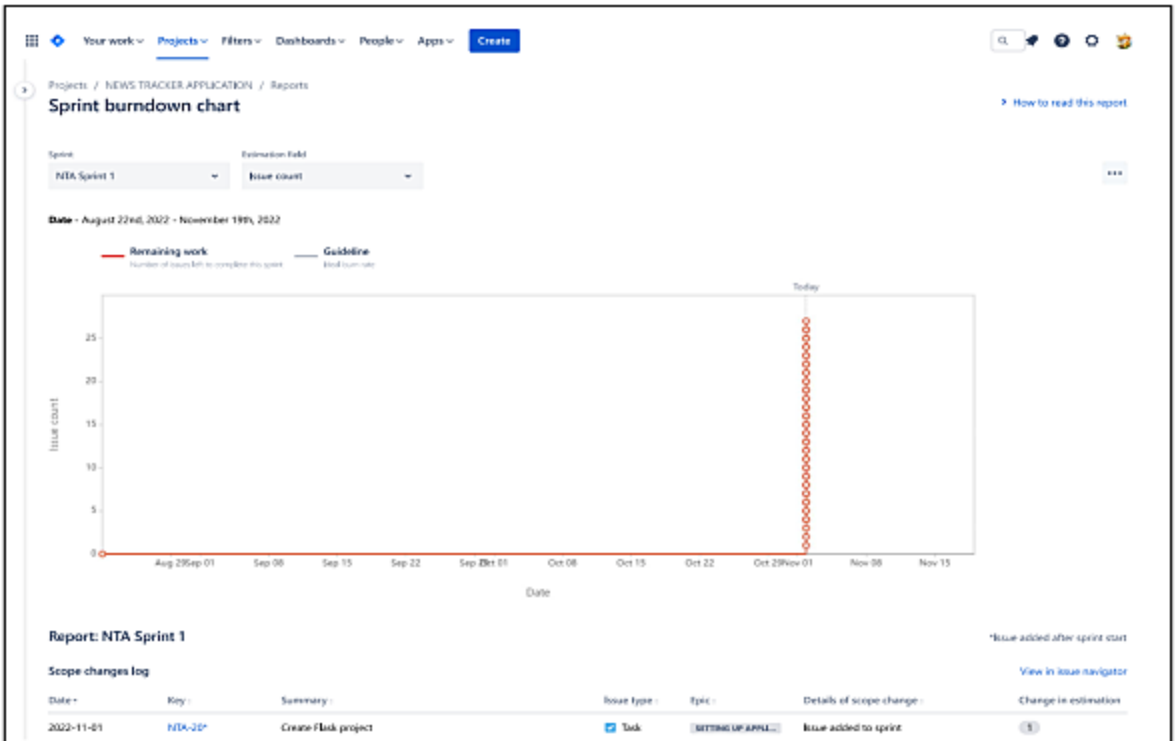
Sprint	*Functional Requirement (Epic)*	*User Story Number*	*User Story / Task*	*Story Points*	*Priority*
Sprint-1	Registration	USN-1	As a user, I can register for the application by entering my email, password, and confirming my password.	2	High
		USN-2	As a user, I will receive confirmation email once I have registered for the application.	1	High
Sprint-2		USN-3	As a user, I can register for the application through Facebook	2	Low
		USN-4	As a user, I can register for the application through Gmail	2	Medium
	Login	USN-5	As a user, I can log into the application by entering email and password	1	High

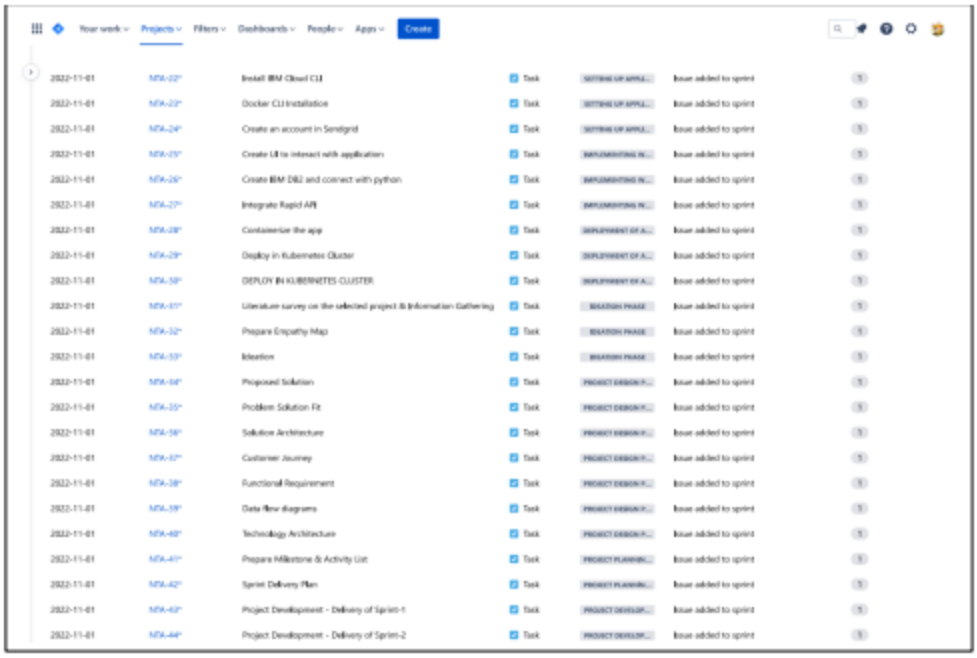
Sprint-3	Dashboard	USN-6	As a user I should be able to navigate and access all the features hassle free	2	High
	Layout	USN-7	As a user I should be able to access the portal with different devices with the same comfort	2	Medium
	User Segregation and data	USN-8	As a CC executive I should be able to identify uniquely.	1	
Sprint-4		USN-9	As an administrator I should be able to modify code according to the future requirements.	2	High
		USN-10	As an administrator I should be able to monitor the cloud system and fix errors before customer.	2	Medium

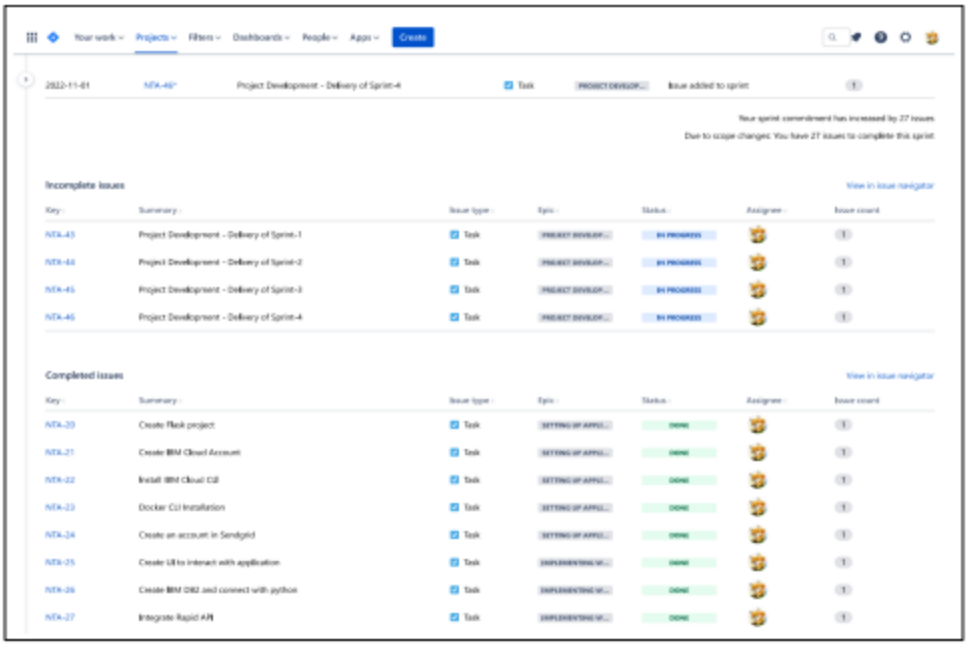
6.3 Reports from JIRA

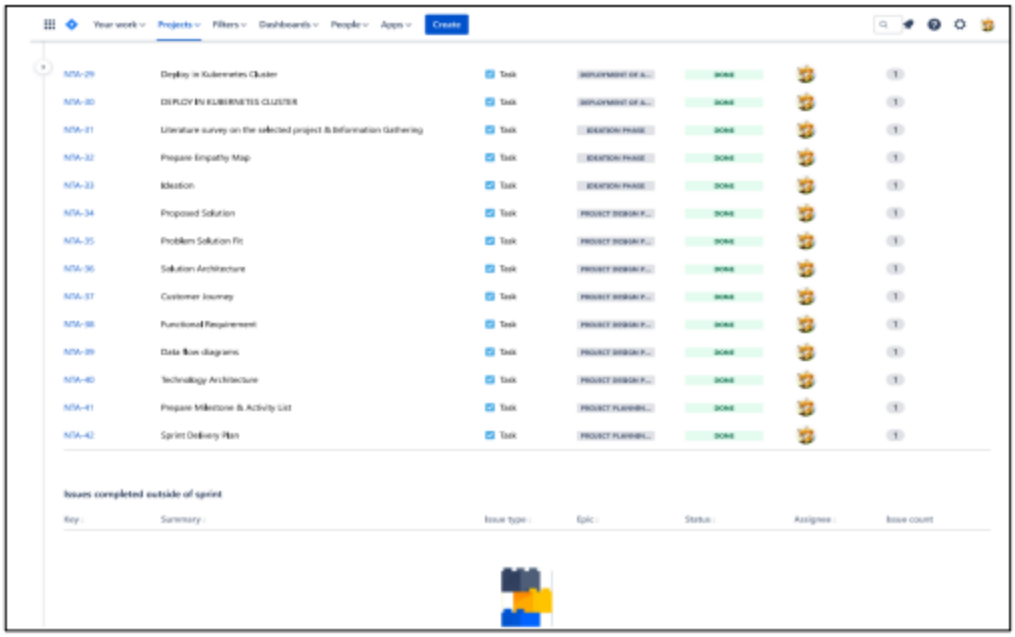

Your work
Projects
Filters
Dashboards
People
Apps
Does your team need more from Jira? Get a free trial of our Standard plan.
Projects / NEWS TRACKER APPLICATION
Backlog
Epic
Insights
NTA-40 Technology Architecture PROJECT DESIGN PHASE -II
NTA-39 Data flow diagrams PROJECT DESIGN PHASE -II
NTA-38 Functional Requirement PROJECT DESIGN PHASE -II
NTA-37 Customer Journey PROJECT DESIGN PHASE -II
NTA-36 Solution Architecture PROJECT DESIGN PHASE – I
NTA-35 Problem Solution Fit PROJECT DESIGN PHASE – I
NTA-34 Proposed Solution PROJECT DESIGN PHASE – I
NTA-45 Project Development - Delivery of Sprint-3 PROJECT DEVELOPMENT PHASE
NTA-44 Project Development - Delivery of Sprint-2 PROJECT DEVELOPMENT PHASE
NTA-43 Project Development - Delivery of Sprint-1 PROJECT DEVELOPMENT PHASE
NTA-46 Project Development - Delivery of Sprint-4 PROJECT DEVELOPMENT PHASE
Quickstart

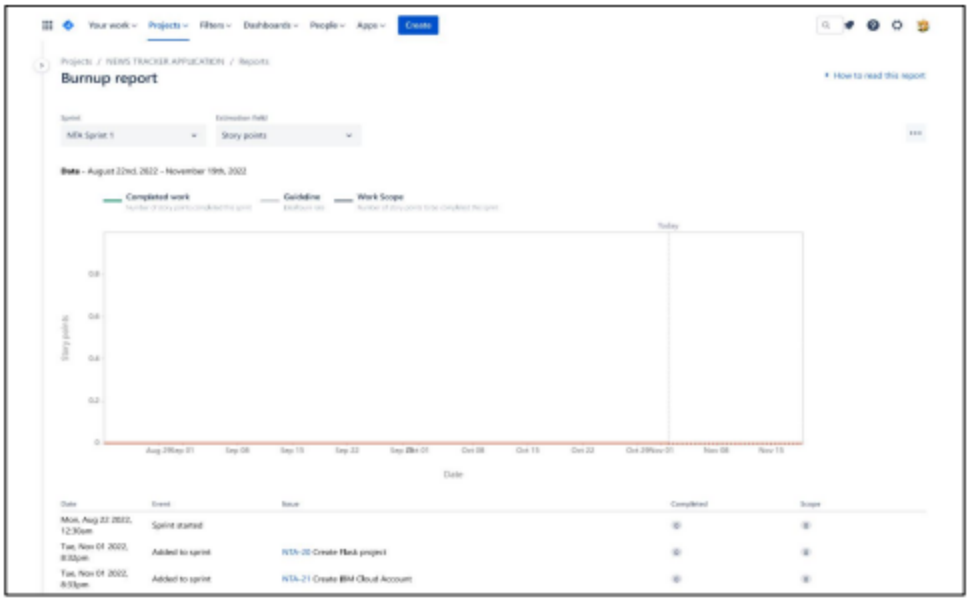

Your work
Projects
Filters
Dashboards
People
Apps
Does your team need more from Jira? Get a free trial of our Standard plan.
Projects / NEWS TRACKER APPLICATION
NTA Sprint 1
13 days remaining
Complete sprint
Epic
Clear filters
GROUP BY
None
Insights
TO DO 1 OF 1 ISSUE
IN PROGRESS 2 OF 6 ISSUES
DONE 2 OF 20 ISSUES
Create an account in Sendgrid
SETTING UP APPLICATION ENVIRO...
NTA-24
Install IBM Cloud CLI
NTA-22
Create Flask project
NTA-20
Docker CLI Installation
NTA-23
Create IBM Cloud Account
NTA-21
Quickstart

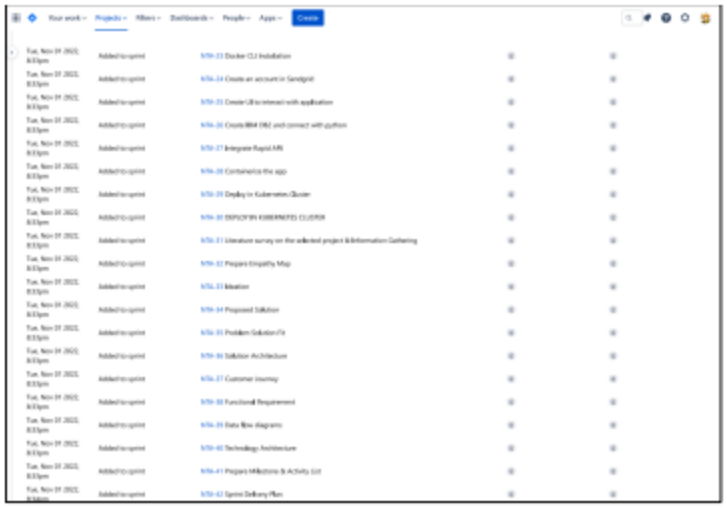

Your work
Projects
Filters
Dashboards
People
Apps
Create
Projects / NEWS TRACKER APPLICATION / Reports
Sprint burndown chart
How to read this report
Remaining work
Guideline
Today
Issue count
Date
Report: NTA Sprint 1
Scope changes log
View in issue navigator

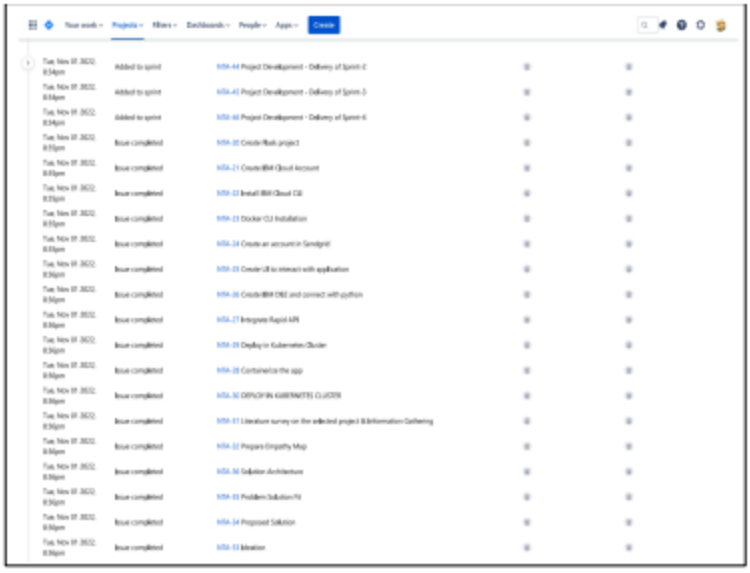

Your work
Projects
Filters
Dashboards
People
Apps
Create
2022-11-01 NTA-22* Install IBM Cloud CLI Task Issue added to sprint
2022-11-01 NTA-23* Docker CLI Installation Task Issue added to sprint
2022-11-01 NTA-24* Create an account in Sendgrid Task Issue added to sprint
2022-11-01 NTA-25* Create UI to interact with application Task Issue added to sprint
2022-11-01 NTA-26* Create IBM DB2 and connect with python Task Issue added to sprint
2022-11-01 NTA-27* Integrate Rapid API Task Issue added to sprint
2022-11-01 NTA-28* Containerize the app Task Issue added to sprint
2022-11-01 NTA-29* Deploy in Kubernetes Cluster Task Issue added to sprint
2022-11-01 NTA-30* DEPLOY IN KUBERNETES CLUSTER Task Issue added to sprint
2022-11-01 NTA-31* Literature survey on the selected project & Information Gathering Task Issue added to sprint
2022-11-01 NTA-32* Prepare Empathy Map Task Issue added to sprint
2022-11-01 NTA-33* Ideation Task Issue added to sprint
2022-11-01 NTA-34* Proposed Solution Task Issue added to sprint
2022-11-01 NTA-35* Problem Solution Fit Task Issue added to sprint
2022-11-01 NTA-36* Solution Architecture Task Issue added to sprint
2022-11-01 NTA-37* Customer Journey Task Issue added to sprint
2022-11-01 NTA-38* Functional Requirement Task Issue added to sprint
2022-11-01 NTA-39* Data flow diagrams Task Issue added to sprint
2022-11-01 NTA-40* Technology Architecture Task Issue added to sprint
2022-11-01 NTA-41* Prepare Milestone & Activity List Task Issue added to sprint
2022-11-01 NTA-42* Sprint Delivery Plan Task Issue added to sprint
2022-11-01 NTA-43* Project Development - Delivery of Sprint-1 Task Issue added to sprint
2022-11-01 NTA-44* Project Development - Delivery of Sprint-2 Task Issue added to sprint

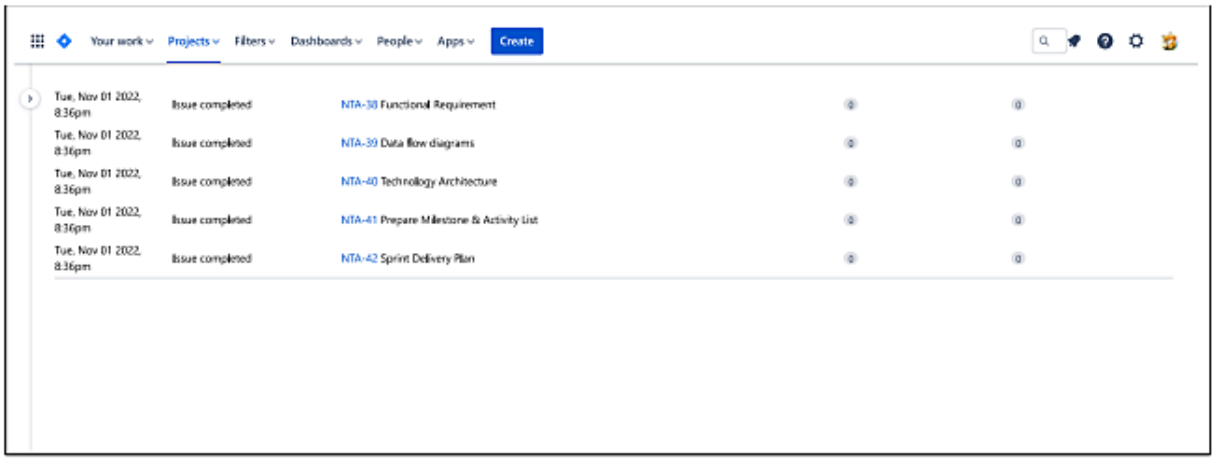

Your work
Projects
Filters
Dashboards
People
Apps
Create
2022-11-01 NTA-46* Project Development - Delivery of Sprint-4 Task Issue added to sprint
Your sprint commitment has increased by 27 issues.
Due to scope changes. You have 27 issues to complete this sprint.
Incomplete issues
View in issue navigator
Key
Summary
Issue type
Epic
Status
Assignee
Issue count
NTA-43 Project Development - Delivery of Sprint-1 Task IN PROGRESS
NTA-44 Project Development - Delivery of Sprint-2 Task IN PROGRESS
NTA-45 Project Development - Delivery of Sprint-3 Task IN PROGRESS
NTA-46 Project Development - Delivery of Sprint-4 Task IN PROGRESS
Completed issues
View in issue navigator
Key
Summary
Issue type
Epic
Status
Assignee
Issue count
NTA-20 Create Flask project Task DONE
NTA-21 Create IBM Cloud Account Task DONE
NTA-22 Install IBM Cloud CLI Task DONE
NTA-23 Docker CLI Installation Task DONE
NTA-24 Create an account in Sendgrid Task DONE
NTA-25 Create UI to interact with application Task DONE
NTA-26 Create IBM DB2 and connect with python Task DONE
NTA-27 Integrate Rapid API Task DONE

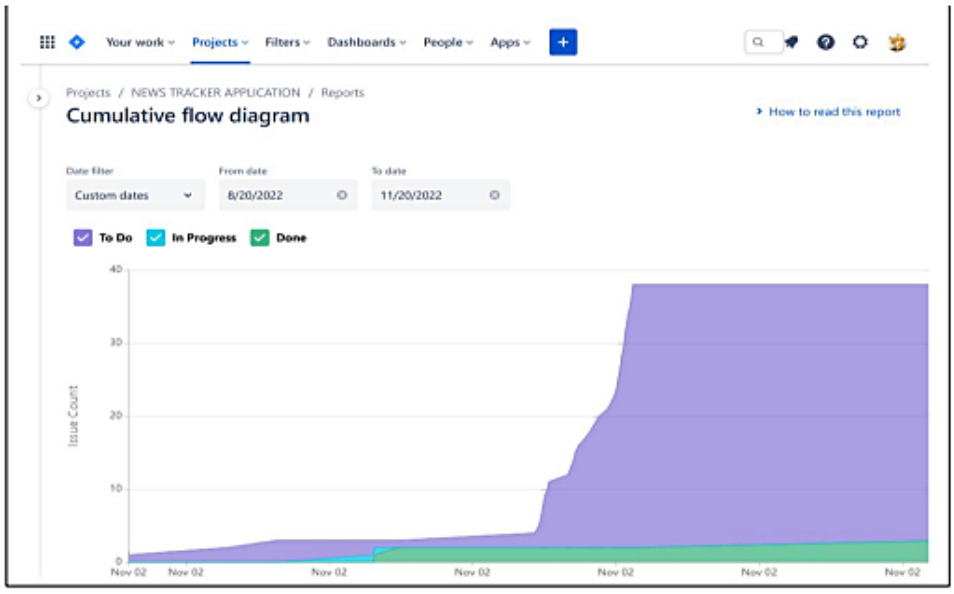

Your work
Projects
Filters
Dashboards
People
Apps
Create
NTA-29 Deploy in Kubernetes Cluster Task DONE
NTA-30 DEPLOY IN KUBERNETES CLUSTER Task DONE
NTA-31 Literature survey on the selected project & Information Gathering Task DONE
NTA-32 Prepare Empathy Map Task DONE
NTA-33 Ideation Task DONE
NTA-34 Proposed Solution Task DONE
NTA-35 Problem Solution Fit Task DONE
NTA-36 Solution Architecture Task DONE
NTA-37 Customer Journey Task DONE
NTA-38 Functional Requirement Task DONE
NTA-39 Data flow diagrams Task DONE
NTA-40 Technology Architecture Task DONE
NTA-41 Prepare Milestone & Activity List Task DONE
NTA-42 Sprint Delivery Plan Task DONE
Issues completed outside of sprint
Key
Summary
Issue type
Epic
Status
Assignee
Issue count

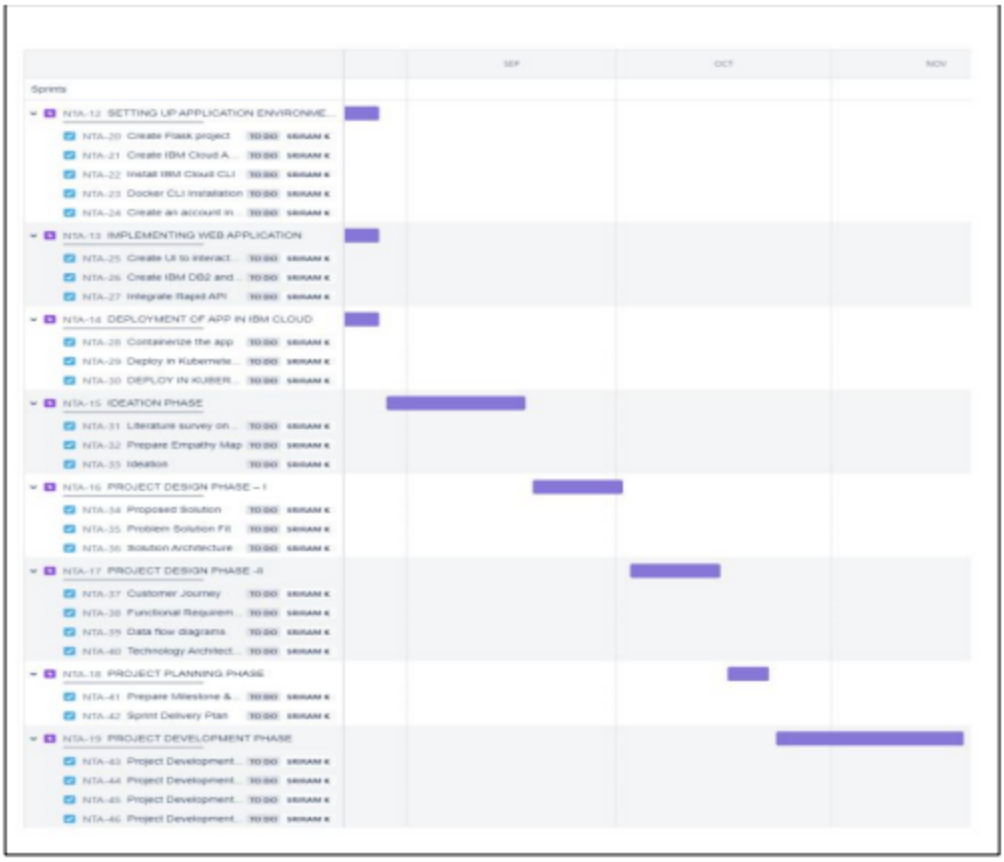

Your work
Projects
Filters
Dashboards
People
Apps
Create
Projects / NEWS TRACKER APPLICATION / Reports
Burnup report
How to read this report
NTA Sprint 1
Story points
Date - August 22nd, 2022 - November 19th, 2022
Completed work
Guideline
Work Scope
Today
Story points
Date
Date
Event
Issue
Completed
Scope
Mon, Aug 22 2022, 12:30pm
Sprint started
Tue, Nov 01 2022, 8:00pm
Added to sprint
NTA-20 Create Flask project
Tue, Nov 01 2022, 8:03pm
Added to sprint
NTA-21 Create IBM Cloud Account


Your work
Projects
Filters
Dashboards
People
Apps
Create


Your work
Projects
Filters
Dashboards
People
Apps
Create


Your work
Projects
Filters
Dashboards
People
Apps
Create
Tue, Nov 01 2022, 8:36pm
Issue completed
NTA-38 Functional Requirement
Tue, Nov 01 2022, 8:36pm
Issue completed
NTA-39 Data flow diagrams
Tue, Nov 01 2022, 8:36pm
Issue completed
NTA-40 Technology Architecture
Tue, Nov 01 2022, 8:36pm
Issue completed
NTA-41 Prepare Milestone & Activity List
Tue, Nov 01 2022, 8:36pm
Issue completed
NTA-42 Sprint Delivery Plan


Your work
Projects
Filters
Dashboards
People
Apps
Projects / NEWS TRACKER APPLICATION / Reports
Cumulative flow diagram
How to read this report
Date filter
Custom dates
From date
8/20/2022
To date
11/20/2022
To Do
In Progress
Done
Issue Count
40
30
20
10
0
Nov 02
Nov 02
Nov 02
Nov 02
Nov 02
Nov 02
Nov 02



7. Coding & Solutioning (explain The Features Added In The Project Along With Code)

7.1 Feature 1

● After go to the news tracker application, the user enters their account details.If the user already have the account then they enter the user name and password.

● Next, they select the any on of the source and go to the news portal.

● If the user haven't the account then they will select the create account option and enter their details.

7.2 Features 2

● After go to the news portal, the user read the news from google news, BBC, etc.

● If the user want to logout the news application then click the logout button otherwise Go to the dashboard.

8. TESTING

8.1 Test Cases

Test case ID	Feature Type	Component	Test Scenario	Pre-Requisite	Steps To Execute	Test Data	Expected Result	Actual Result	Status	Commnets	Executed By
Home Page-1	Functional	Home Page	Open Home Screen of the application	NIL	1.Enter URL (https://newstracker.ml) and click go	https://newstracker.ml	Login/Signup popup should display	Working as expected	Pass	NIL	KEERTHANA S
Login Page-1	Functional	Login Page	Verify the UI elements in Login/Signup popup	User should have existing username and password	1.Enter URL(https://newstracker.ml) and click go 2.Enter Valid username in text box 3.Enter valid password in password text box 4.Click on login button	Username: keerthana password: Keeru@123	Application should show below UI elements: a.email text box b.password text box c.Login button d.New customer? Create account link e.Show button Forgot accout?	Working as expected	Pass	Nill	RAMYA D
Login Page-2	Functional	Login Page	Verify the UI elements in Login/Signup popup	User should not have existing username and password	1.Enter URL(https://newstracker.ml) and click go 2.Enter Valid username in text box 3.Enter valid password in password text box 4.Click on login button	Username: casdfgg@gmail password: d32rfwe	Application should show 'Incorrect email or password ' validation message.	Working as expected	Pass	Nill	SHOBANA M
Signup Page-1	Functional	Signup page	Register a user and store the data into database and send login credentials to them via mail address provided by the user	Should have a valid mail address	1.Enter URL(https://newstracker.ml) and click go 2.Click create an account button 3.Enter full name, email, username and password	Any valid data	Successful creation of acction and sharing credentials via mail	Working as expected	Pass	Nill	SWETHA K

Signup Page-2	Functional	Signup page	Register a user, where the email and username already exist.	Nil	1. Enter URL(https://newstracker.ml) and click go 2. Click create an account button 3. Enter full name, email, username and password	Any valid data	Shows Error: email or username alreadt exist	Working as expected	Pass	Nill	KEERTHANA S
Forgot username or password-1	Functional	Forgot page	Enter the already registered mail id and get back the username and password to the same mail.	Should have registered mail id	1. Enter URL(https://newstracker.ml) and click go 2. Click forgot username/password button 3. Enter the mail id	Enter valid mail id	Receive username and password to the provided mail address	Working as expected	Pass	Nill	RAMYA D
Forgot username or password-2	Functional	Forgot page	Enter maill id which is not registered.	Any mail id	1. Enter URL(https://newstracker.ml) and click go 2. Click forgot username/password button 3. Enter the mail id	Enter invalid mail id	Shows Error: email id not found	Working as expected	Pass	Nill	SHOBANA M
Dashboard	Functional	Dashboard	Redirect user to dashboard page on providing valid credentials	Valid credentials	1. Enter URL(https://newstracker.ml) and click go 2. Enter Valid username in text box 3. Enter valid password in password text box 4.Click on login button	Valid credentials	Redirects to dashboard, shows user's full on the top screen	Working as expected	Pass	Nill	SWETHA K
News-1	Functional	News Page	Redirect user to news page on clicking any one the listed news sources	NIL	1.Click any one the listed news sources	Clicking BBC NEWS	Redirects to BBC NEWS page, show articles from the api	Working as expected	Pass	Nill	RAMYA D
News-2	Functional	News Page	Click dashboard button	NIL	1.Click dashboard button	Click dashboard button	Redirects to dashboard page,if the user is logged in	Working as expected	Pass	Nill	SHOABANA M
News-3	Functional	News Page	Click logout button	NIL	1.Click logout button	Click logout button	Redirects to login screen and shows messages as logged out successfully	Working as expected	Pass	Nill	SWETHA K

8.2 User Acceptance Testing

1. Purpose of Document

The purpose of this document is to briefly explain the test coverageand open issues of the News Tracker Application project at the time of the release to User Acceptance Testing (UAT).

2. Defect Analysis

This reportshows the number of resolved or closed bugs at each severity level, and how they were resolved

Resolution	*Severity 1*	*Severity 2*	*Severity 3*	*Severity 4*	*Subtotal*
By Design	8	3	4	2	17
Duplicate	2	1	0	6	9
External	6	3	0	1	16

Fixed	5	1	4	10	20
Not Reproduced	1	2	0	1	4
Skipped	0	0	9	3	12
Won't Fix	2	3	1	1	7
Totals	24	13	18	24	79

Section	*Total Cases*	*Not Tested*	*Fail*	*Pass*
Print Engine	8	0	0	8
Client Application	23	0	0	23
Security	3	0	0	3
Outsource Shipping	2	0	0	2

Exception Reporting	6	0	0	6
Final Report Output	6	0	0	6
Version Control	1	0	0	1

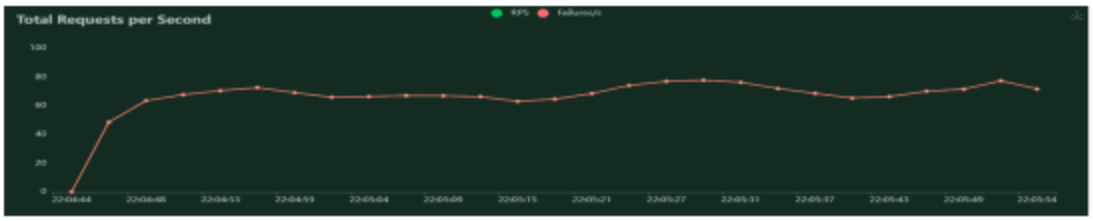
9. RESULTS

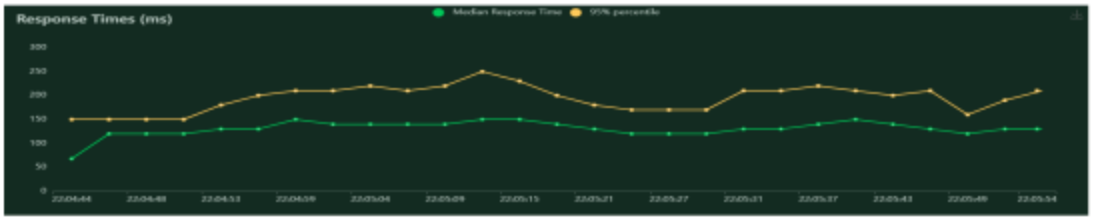
9.1 Performance Metrics

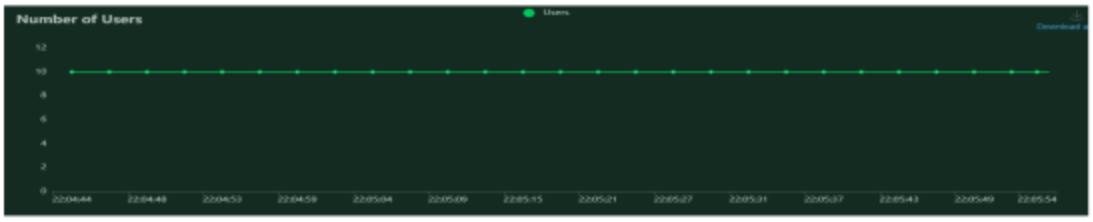
NFT - Risk Assessment	
Project Name	News Tracker Application
Scope/feature	New
Functional Changes	Low
Hardware Changes	No Changes
Software Changes	Moderate
Impact of Downtime	No Impact of Downtime
Load/Voluem Changes	>5 to 10%
Risk Score	ORANGE
Justification	As we have seen the chnages

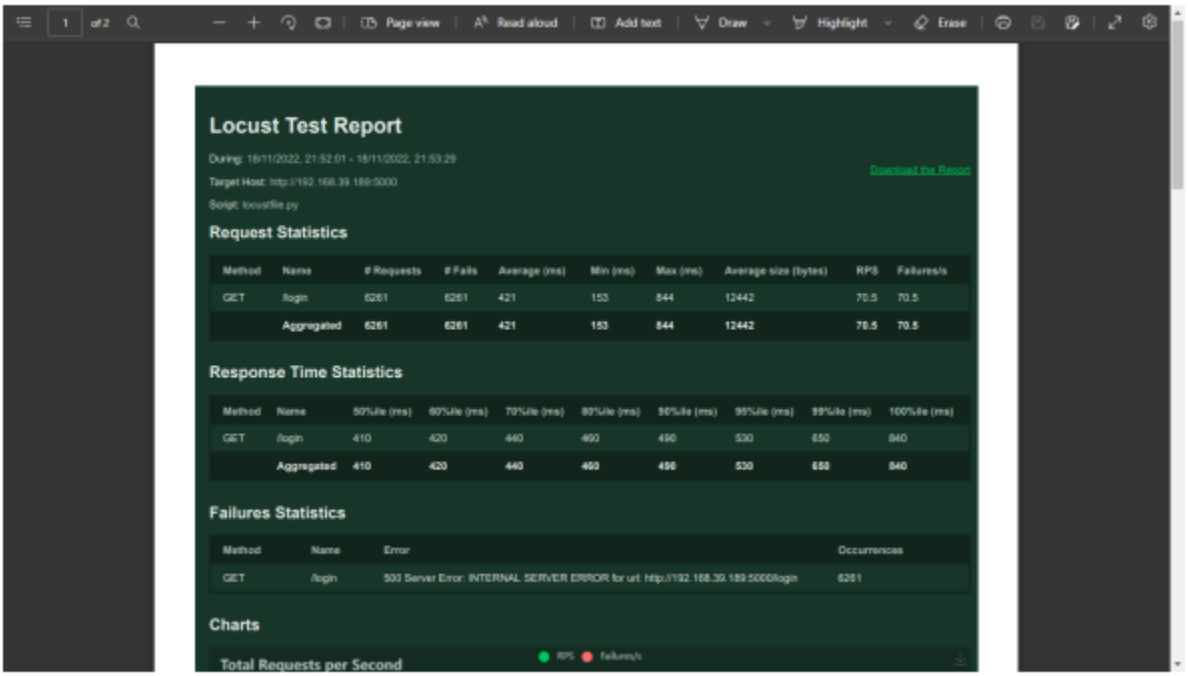
NFT - Detailed Test Plan	
Project Overview	News Tracker Application
NFT Test approach	Locust
Assumptions/Dependencies/Risks	No Risk Factors
Approvals/SignOff	Approved

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









10. ADVANTAGES & DISADVANTAGES

This news tracker application is easy to readability. Hardcopy newspapers can most times appear threatening unless you have a knack for gulping bulky literary contents. From the front page to the rear there's hardly any space, everywhere is littered with loads of texts in tiny fonts just to conserve space but online, every news is well arranged and spaced. The texts are written in lovely and attractive readable fonts. The columns are carefully arranged in tabs and you can choose whichever you find interesting and consume away. This application has portability and mobility. While a mobile hand-held phone or tab you can access breaking news on-the-go without having to stress yourself holding high volume papers and probably making your neighbour uncomfortable in the cab with your arms widespread. You don't need in inconvenience your seat mate when reading daily Nigerian news from Hynaija, Punch Nigeria, Vanguard Nigeria, Legit.ng (Formerly Nauj.com) or Nairaland etc. The convenience of reading is a huge plus. When you're surfing the web for news, you are entitled to news from as many news online outlet as your patience or time can permit. Different source of news are available to you all in one small digital device. You could even read intercontinental and global news right at the laid-back comfort of the settee in your living room.

Disadvantage of news tracker application is Reading Garbage news daily. With the rise of news blogs all over the web space, most bloggers can just sit at home and write something out of the blues just to gain traffic for their blog. The legitimacy of news sources online is questionable. The online world is now filled up with falsity, misinterpretation of

information, defamation of public characters, unguided and unfounded news. Generally, the public consume garbage information and end up misguided and mislead. The veracity of news being churned out online daily is a major bane of the 21stcentury information age. The upsurge of digital news all over the country is gradually making print journalism staff who eck a living out of the hustle to be laid off as even the newspapers outfits themselves are now going digital making half of their staff go dispensable.If you'll be publishing online, you don't need a large office space, cleaners, typists, tons of editors, receptionists and every other relevant staff involved in print media.

11. CONCLUSION

In our project work, an attempt has been made to develop a News or information-based website. We develop this project that helps the people and make them aware so that they can know any news. To establish this website, we use various methodologies. To develop this project, we have faced many problems but we hardly tried to develop this project. Our supervisor helps us by giving his valuable opinion, decision and time.

12. FUTURE SCOPE

As for our future development, the following can be done:

● We will manage news reporting system.

● We can make video conference system.

● We update our database

● We can make Searching System more flexible.

● Sensibility level could add be added.

13. APPENDIX

Source Code

Dashboard.css

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

.body
{
  background-image: url('https://ibm-news-tracker.s3.jp-tok.cloud-object-
storage.appdomain.cloud/login-signup-backgroud-image.png');
```

```
  background-repeat: no-repeat;
  background-size: cover;
}

h4
{
  font-family: Pacifico, cursive;
}

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

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

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

.form img
{
  margin-left: 20px;
  widthwidth: 100px;
```

```
  height:100px;
  border-radius: 15px;
}

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

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

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

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

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

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

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

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

.container
{
  max-width: 100%;
}

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

```
}
```

news.css

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

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

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

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

.form button
{
  font-family: 'Merriweather', serif;
  font-size: 18px;
  font-weight: bold;
```

```
  letter-spacing: .1em;
  outline: 0;
  background: #17a588;
  width: 250px;
  border: 5;
  border-radius:15px;
  margin: 0px 0px 8px;
  padding: 15px;
  color: #FFFFFF;
  -webkit-transition: all 0.3 ease;
  transition: all 0.3 ease;
  cursor: pointer;
  transition: all 0.2s;
}

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

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

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

```
  margin-left:-8px;
 }

 .form button
 {
  font-family: 'Merriweather', serif;
  font-size: 25px;
  font-weight: bold;
  letter-spacing: .1em;
  outline: 0;
  background: #17a588;
  width: 250px;
  height: 90px;
  border: 5;
  border-radius:15px;
  margin: 0px 0px 8px;
  padding: 15px;
  cursorcolor: #FFFFFF;
  -webkit-transition: all 0.3 ease;
  transition: all 0.3 ease;
  cursor: pointer;
  transition: all 0.2s;
 }

 .container
 {
  max-width: 100%;
 }

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

pages.css

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

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

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

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

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

```
  font-size: 20px;
border-radius:10px;
}

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

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

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

.form .message
{
  margin: 6px 6px;
  color: #808080;
```

```
  font-size: 11px;
  text-align: center;
  font-weight: bold;
  font-style: normal;
}
.form .message a
{
  color: #FFFFFF;
  text-decoration: none;
  font-size: 13px;
}

.form .register-form
{
  display: none;
}

.container
{
  position: relative;
  z-index: 1;
  max-width: 300px;
  margin: 0 auto;
}

.container:before, .container:after
{
  content: "";
  display: block;
  clear: both;
}

.container .info
{
  margin: 50px auto;
  text-align: center;
}

.container .info h1
{
  margin: 0 0 15px;
```

```
  padding: 0;
  font-size: 36px;
  font-weight: 300;
  color: #1a1a1a;
}

.container .info span
{
  color: #4d4d4d;
  font-size: 12px;
}

.container .info span a
{
  color: #000000;
  text-decoration: none;
}

.container .info span .fa
{
  color: #EF3B3A;
}

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

dashboard.html

```
<html>

  <head>
```

```
    <title>DASHBOARD</title>

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

    <meta charset="utf-8">

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

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

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

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

        <script>

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

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

    </script>

  </head>
```

```
 <body class="body">

 <div class="on-screen">

  <div class="form">

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

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

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

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


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

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

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

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

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

    </form>

  </div>

 </div>

 </body>
```

```
</html>
```

forgot.html

```
<html>

  <head>

    <title>FORGOT</title>

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

    <meta charset="utf-8">

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

    <link href='https://fonts.googleapis.com/css?family=Titillium+Web:400,300,600' rel='stylesheet' type='text/css'>

    <link href='https://fonts.googleapis.com/css?family=Titillium+Web:400,300,600' rel='stylesheet' type='text/css'>

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

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

  </head>

  <body class="body">

  <div class="on-screen">

    <div class="form">

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

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

        <p style="font-family: 'Pacifico', cursive;">Forgot your username or password?<br>Don't worry, enter your registered email address here, your login credentials will be sent to your registered email address.</p><br>

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

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

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

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

      </form>

    </div>

  </div>

  </body>

</html>
```

login.html

```
<html>

  <head>

    <title>LOGIN</title>

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

    <meta charset="utf-8">

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

```
awesome/4.7.0/css/font-awesome.min.css">

   <link href='https://fonts.googleapis.com/css?family=Titillium+Web:400,300,600'
rel='stylesheet' type='text/css'>

   <link href='https://fonts.googleapis.com/css?family=Titillium+Web:400,300,600'
rel='stylesheet' type='text/css'>

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

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

  </head>

  <body class="body">

  <div class="on-screen">

    <div class="form">

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

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

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

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

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

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

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

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

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

    </form>

  </div>

 </div>

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

 </body>

</html>
```

newaccount.html

```
<html>

 <head>

  <title>LOGIN</title>

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

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

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

    <link href='https://fonts.googleapis.com/css?family=Titillium+Web:400,300,600' rel='stylesheet' type='text/css'>

    <link href='https://fonts.googleapis.com/css?family=Titillium+Web:400,300,600' rel='stylesheet' type='text/css'>

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

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

  </head>

  <body class="body">

  <div class="on-screen">

    <div class="form">

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

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

        <p style="font-family: 'Pacifico', cursive;">Your account has been successfully registered and the login credentials were shared with you to the registered email address.</p><br>

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

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

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

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

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

    </form>

  </div>

 </div>

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

 </body>

</html>
```

news.html

```
<html>

  <head>

    <meta charset="UTF-8">

    <title>CAD-NEWSTRACKER</title>

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

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

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

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

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

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

        <script>

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

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

   </script>

   </head>

   <body>
```

```
   <div class="form">

      <form>

         <br><br>

         <button style="margin-right:-2%;" type="button">CAD-NEWSTRACKER</button>

         <button style="margin-left:40%;" type="button"
onclick="window.location.href='/logout'">LOGOUT</button>

         <button type="button"
onclick="window.location.href='/dashboard'">DASHBOARD</button>

      </form>

                                    </div>

    <br><br><br><br><br><br><br>

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

                                    <br>

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

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

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

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

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

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

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

```
      {% endfor %}

   </body>

</html>
```

signup.html

```
<html>

  <head>

     <title>SIGN UP</title>

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

     <meta charset="utf-8" />

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

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

     <link href="https://fonts.googleapis.com/css?family=Titillium+Web:400,300,600" rel="stylesheet" type="text/css"/>

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

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

   </head>

   <body class="body">

     <div class="on-screen">

       <div class="form">
```

```
        <form action="/register" method="POST">

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

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

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

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

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

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

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

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

        </form>

      </div>

    </div>

  </body>

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

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

   </script>

 </html>
```

app.py

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

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

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

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

```
@app.route('/login',methods=['GET','POST'])
def login():
    global userid
    msg=' '
    if request.method=='POST':
        username = request.form['username']
        password = request.form['password']
        sql = "SELECT * FROM NTAAC WHERE username = ? AND password = ?"
        stmt = ibm_db.prepare(conn,sql)
        ibm_db.bind_param(stmt,1,username)
        ibm_db.bind_param(stmt,2,password)
        ibm_db.execute(stmt)
        account = ibm_db.fetch_assoc(stmt)
        print(account)
        if account:
            return render_template('dashboard.html')
        else:
            return render_template('login.html')

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

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

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

```
        else:
            return render_template('forgot.html')

if __name__ == '__main__':
    app.run(host='0.0.0.0', port=88, debug=True, threaded=True)
```

deployment.yaml

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

Dockerfile

```
FROM python:3.6
WORKDIR /app
ADD . /app
COPY requirements.txt /app
RUN python3 -m pip install -r requirements.txt
RUN python3 -m pip install ibm_db
RUN python3 -m pip install newsapi-python
RUN python3 -m pip install flask_mail
RUN python3 -m pip install flask
EXPOSE 5000
CMD ["python","app.py"]
```

Requirements

flask

flask_mail

newsapi-python

ibm_db

service.yaml

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