PROJECT REPORT ON NEWS TRACKERAPPLICATION

|  |  |
| --- | --- |
| Date | 15 November 2022 |
| Team ID | PNT2022TMID31583 |
| Project Name | News Tracker Application |

**TEAM MEMBERS:**

Team Lead : Hari Prasath S Team Member 1 : Jegadeeswaran R Team Member 2 : Danusu B Team Member 3 : Mohan Kumar S

**CONTENTS**

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

Digital news continues to evolve, encouraged by a various innovations in recent time, from ground breaking new technologies like virtual reality and automated reporting to experiments on social platforms that have altered campaign coverage. Topic detection and tracking is challenging topic in information retrieval technology that can be used in the text mining. In topic detection we finding the most important topics in a collection of news articles. Our approach combines a variety of learning techniques. Topic detection is an unsupervised task and topic tracking is supervised task. To identify the serious news, we identify the clusters that fall into same category.

The corpus considered the news from a large number of internet news sites from across the world like Times of India and CNN, and of various subscription news wires. Thus the collection of different news from different source have same events. Newspapers normally receive the news from various news agencies with very few changes. Thus the corpus of news articles contains the same events written by different journalists which must be eliminated from the collection.

* 1. **Purpose**

The main purpose of project is to identify interesting events happens in the world. Analysts are continuously trying to identify latest news and story from a very large resources of information that arrives daily. So for journalist it is easy to understand and identify actual events. The objective is to decompose topics is to extract the events which never seen before and combine them which represents same news stories. The scope of the project is text in news articles obtained from the various newspaper websites. Text mining utilizes techniques from the field of data mining, combines methodologies from various other areas such as categorization, information retrieval, clustering, summarization, information extraction, computational linguistics, concept linkage and topic tracking.

The main objectives of the project are as follows:

* It tracks all the information of Types, Media, and Latest Posts etc.
* Provides the searching facilities based on various factors. Such as News, Advertisement, Contents.
* To increase efficiency of managing the News.
* Integrating of all records of contents.
* Shows the information and description of the News.
* The transactions are executed in off-line mode, hence online data for News, types capture and modification is not possible.
  1. **Existing Problem**

## There are multiple news-sharing apps used by a single user and are often spammed with notifications. There is also a lot of fake news which gets shared. A news-sharing app wants to help users find relevant and important news easily every day and also understand explicitly that the news is not fake but from proper sources. Start with an audience-first focus. Instead of thinking about what your organization produces (news articles) and how to fit that into an app, start from scratch and consider your audience. Who are they, and what problems do they face? Use your imagination, do some role playing with colleagues, and actually talk to some real people. In the existing model, there’s no algorithm to find the nearest donor during emergency cases. The user needs to register for the type of plasma needed and has to wait for the donor to arrive in case of unavailability of that type of plasma in the bank.

* 1. **References**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **S.No** | **Title** | **Publication**  **details** | **Methodology/**  **algorithms** | **Merits** | **Demerits** |
| 1. | Clustering-Based Online News Topic Detection and Tracking Through Hierarchical Bayesian Nonparametric Models | Wentao Fan, Zhiyan Guo, Nizar Bouguila,  Wenjuan Hou  SIGIR '21: Proceedings of the 44th International ACM SIGIR Conference on Research andDevelopment in Information | Dividing articles into topically coherent units, discovering their topic could be quite valuable in many applications where people need efficient access to large quantities of information. | Has used the  Agglomerative  hierarchical  clustering using average distance measure for topic  Detection | It cannot add Topic Tracking module within our approach |
| 2. | Detection and Tracking  For Online News Texts | Guixian Xu, Yueting Meng, Zhan Chen  , Xiaoyu Qiu  August 2015  DOI:10.1145/2785830.2785860  Conference: the 17th International Conference | The topic mining using the K-means method is compared to highlight the advantages of using LDA for topic discovery. Second, the improved single-pass algorithm is used to track news topics. | Then the improved Single-Pass algorithm is used for topic tracking. | The position factor is not considered, which will be the direction of the next work  . |
| 3. | Tracking Aspects in News Documents | Maoto Inoue; Takao Miura2020,vol.2021, pp. 1-6,  DOI:10.1109/ICCS49678.2020.9277004 | An extract some story-lines as aspects of topics by probabilistic likelihood. | Usage of cloud server data base | It is not determine the number of daily topics. |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 4. | Exploring mobile news reading interactions for news app personalization | MariosConstantinides  John Dowell  David Johnson  Sylvain Malacria | It classified each user as a particular news reader type corresponding with the five factors and making the simplifying assumption that the factors are equally weighted. | Explored the feasibility of recognizing patterns of news reading interactions and evaluated three adaptive interface designs for different news reader types | 1. Incomplete adaptive mobile news framework providing automatic personalization of news apps |
| 5. | Clustering News Articles for Topic Detection | Vaidehi Patel, Arpita Patel  MAY 2018 | IRE Journals | Volume 1 Issue 11 | ISSN: 2456-8880 | The motivation for research in TDT is to provide a core technology for a system that would monitor news and alert an analyst to new and interesting events occurring in the world. | It is used for  Agglomerative  hierarchical  clustering using  average distance  measure for topic detection | It would not be applying system for some particular  domain |
| 6. | Exploring mobile news reading interactions for news app personalisation | Marios Constantinides,John Dowell,David Johnson,Sylvain Malacria | Set was of the classified each user as a particular news reader type corresponding with the five factors and making the simplifying assumption that the factors are equally weighted. | Explored the feasibility of recognising patterns of news reading interactions and evaluated three adaptive interface designs for different news reader types. | Incomplete adaptive mobile news framework providing automatic personalization of news apps. |

* 1. **Problem Statement Definition**

## As world's technology is rapidly growing, we has fast connection and network to instantly connect to other person. Day to day use of mobile, tablets and laptop is increasing, most of the people already have this facilities. In this fast and information oriented world we need to stay updated with every incidents and news too. The main focus of this application is to connect news articles from all around the world and deliver it to user as fast as possible in best visualize way.

## The rapid development of the network makes the number of web pages is more and more huge. In order to get the news they need, people usually browse a large number of web pages, this is not only a waste of time, but also can't be able to get the latest and complete news certainly. The users or people now does not need the outdated news or unwanted news, they need more than ever for that access the latest news to be adaptive and personalized.



|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Problem**  **Statement (PS)** | **I am**  **(Customer)** | **I’m trying to** | **But** | **Because** | **Which makes me feel** |
| PS-1 | Journalist/Article writer | Gather information | I am unable to get the right news | There are vast area of news are available | Annoyed |
| PS-2 | Student | Read current affairs to clear exams | It is hard to go through all papers | News is not brief | Exhausted |

1. **IDEATION AND PROPOSED SOLUTION**
   1. **Empathy Map Canvas**

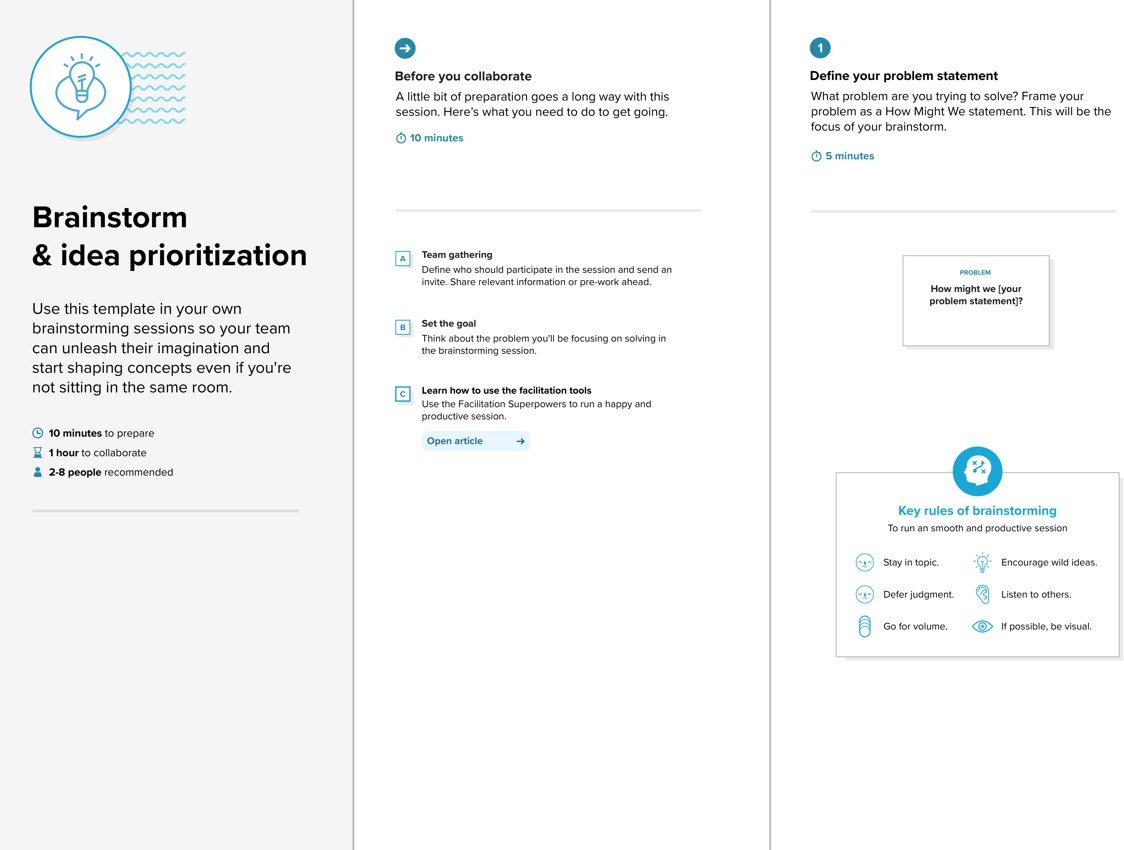
An empathy map is a simple, easy-to-digest visual that captures knowledge about a user’s behavior’s and attitudes. It is a useful tool to helps teams better understand their users. Creating an effective solution requires understanding the true problem and the person who is experiencing it. The exercise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

****

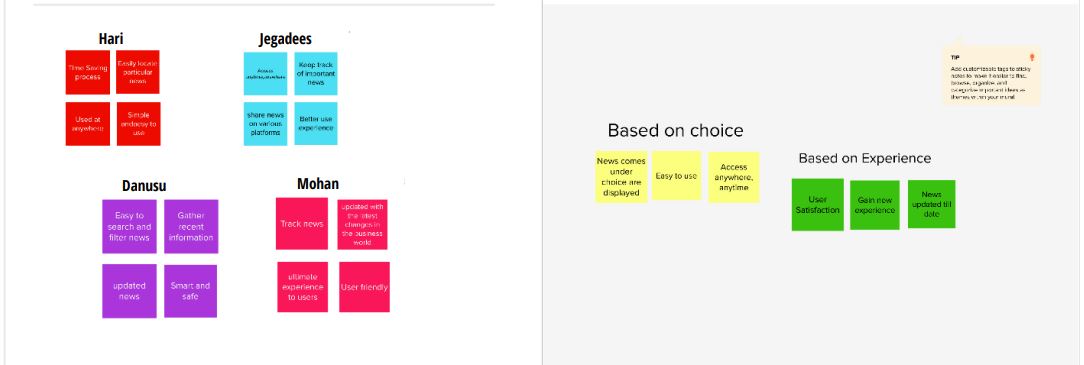
* 1. **Ideation And Brainstorming**

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.

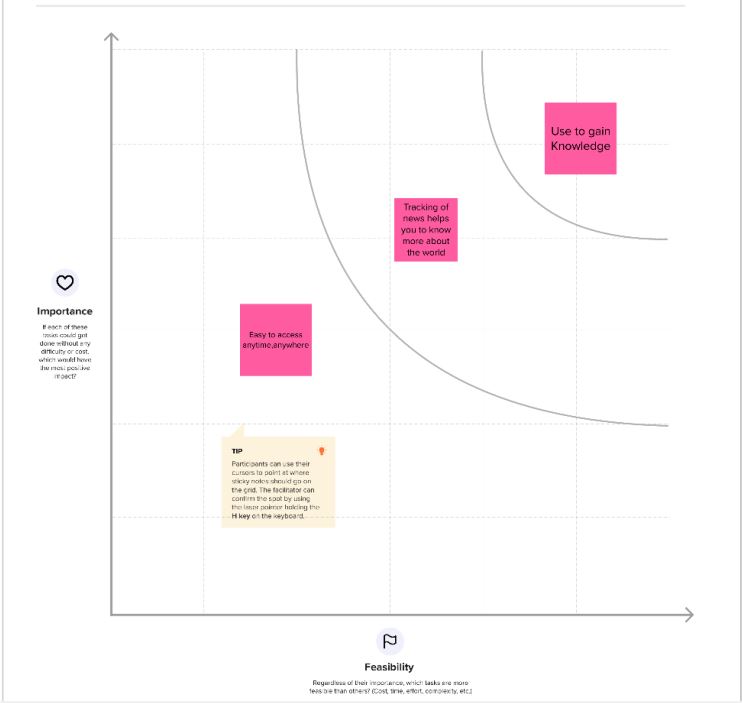
**Step-1: Team Gathering, Collaboration and Select the Problem Statement**



**Step-2: Brainstorm, Idea Listing and Grouping**



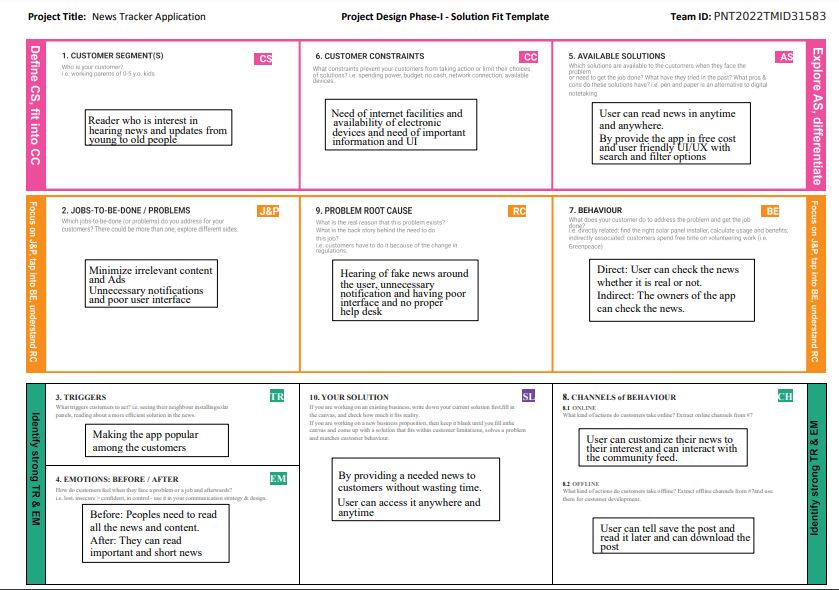
**Step-3: Idea Prioritization**



* 1. **Proposed Solution**

|  |  |  |
| --- | --- | --- |
| **S.No.** | **Parameter** | **Description** |
| 1. | Problem Statement (Problem to be solved) | The news headlines should be easy and instant  in time.The aim is to make the reader read the required news anytime and anywehere without wasting time on the unnecessary news. |
| 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 | 1. A user can read news only from their interested fields rather than reading all the news and interest based filtering. 2. 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) | 1. Add advertisements to the application, so that we can get revenue from those advertisement-sponsored organizations. More advertisements   may irritate the user.   1. Add premium subscription, users who   subscribe for premium won’ t get advertisements and get the content more faster without any popups. |
| 6. | Scalability of the Solution | As it was an application-based project, good ideation and execution can result in the development of an flawless application. If the users like our application, they may recommend and share it with the others, which will grow our application insights. |

* 1. **Problem Solution fit**

****

1. **REQUIREMENT ANALYSIS**
   1. **Functional requirement**

Following are the functional requirements of the proposed solution:

|  |  |  |
| --- | --- | --- |
| **FR No.** | **Functional Requirement (Epic)** | **Sub Requirement (Story / Sub-Task)** |
| FR-1 | User Installation | User can install the app from the Google play store or from the website |
| FR-2 | User Registration | Registration through Form Registration through Gmail |
| FR-3 | User Confirmation | Confirmation via email |
| FR-4 | User Login | User should log in the app with the user's name and password |

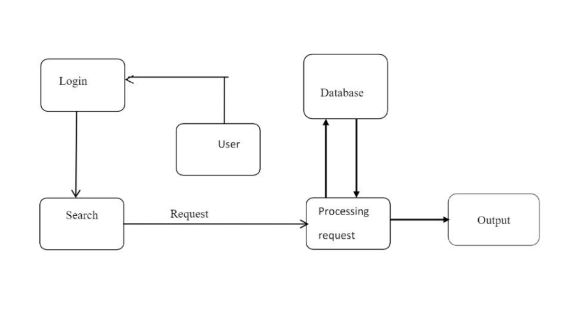
* 1. **Non-Functional requirements**

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

|  |  |  |
| --- | --- | --- |
| **FR No.** | **Non-Functional Requirement** | **Description** |
| NFR-1 | **Usability** | Everyone can understand the process of using the app easily by the commands given in the app |
| NFR-2 | **Security** | It is a more secured app. No fake news can be shared.. |
| NFR-3 | **Reliability** | The system needs to be reliable enough and needs to function without any technical failures. |
| NFR-4 | **Performance** | The system needs to be as fast as possible regardless  of the number of integrations and the traffic on the website increases. |

|  |  |  |
| --- | --- | --- |
| NFR-5 | **Availability** | The system needs to be accessible to a user at anygiven point of time. |
| NFR-6 | **Scalability** | The system needs to be scalable enough to support  a large number of users at the same time whilemaintaining the optimal performance. |

1. **PROJECT DESIGN**
   1. **Data Flow Diagram**



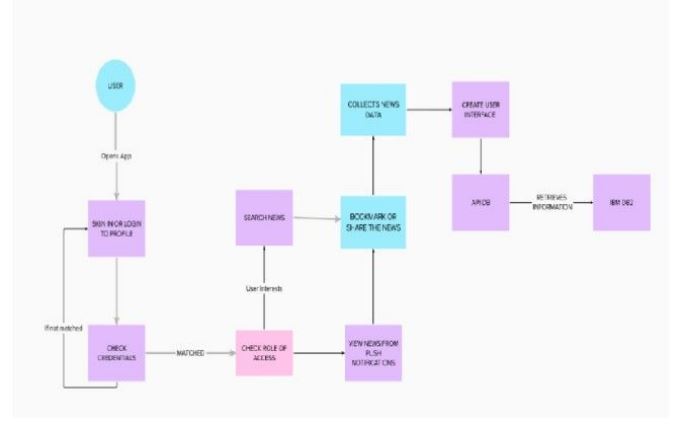
***Fig: Data Flow Diagram***

* 1. **Solution & Technical Architecture**

1. **Solution Architecture Diagram**

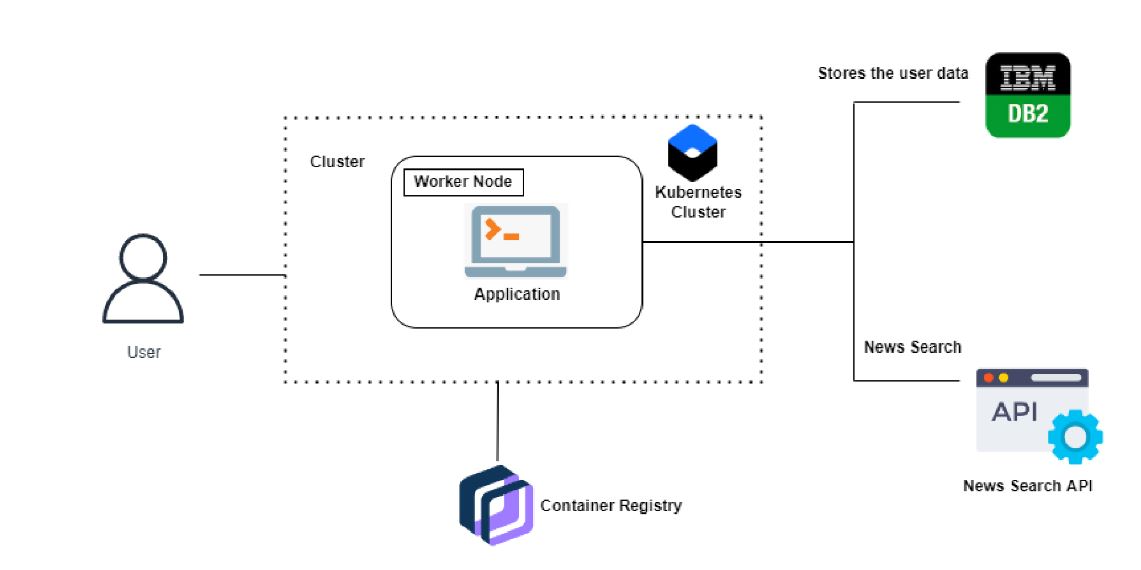
Solution architecture is a complex process – with many sub-processes – that bridges the gap between business problems and technology solutions. Its goals are to:

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



***Fig: Architecture and data flow of the plasma donor application***

1. **Technical Architecture**

****

***Fig: Technical Architecture***

**Table 1: Application Characteristics**

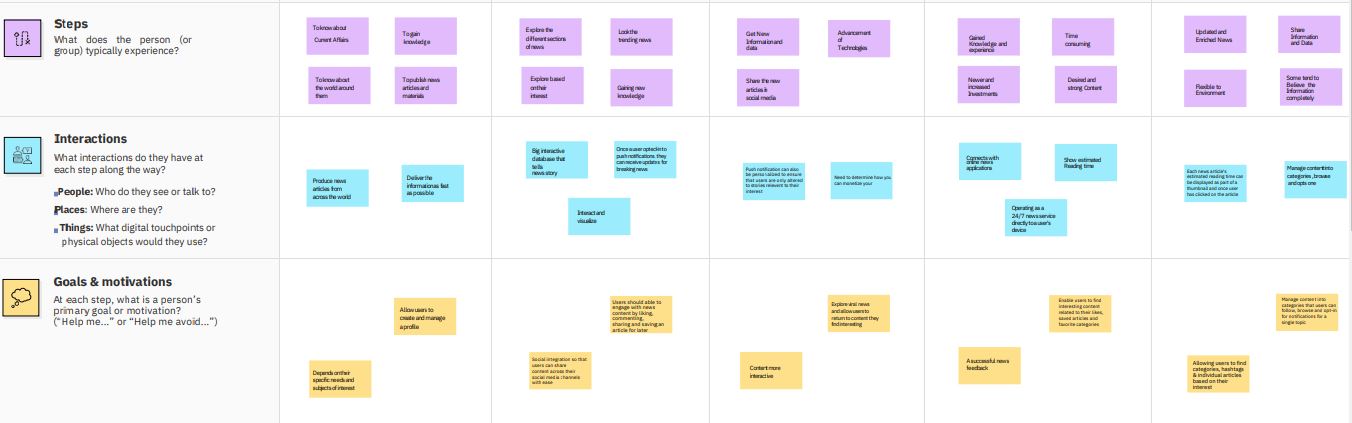
|  |  |  |  |
| --- | --- | --- | --- |
| **S. No** | **Characteristics** | **Description** | **Technology** |
| 1. | Open-Source Frameworks | Python – flask is an open-source framework used to develop the  application. | Python - Flask |
| 2. | Security Implementations | Container registry and Kubernetes Cluster are used for  encryption of data. | Container registry and Kubernetes Cluster |
| 3. | Scalable Architecture | Kubernetes Cluster allow containers to run across multiple  machines and environments. | Kubernetes Cluster |
| 4. | Availability | Kubernetes Cluster provides all time  availability. | Kubernetes Cluster |
| 5. | Performance | Docker improves the application performance. | Docker |

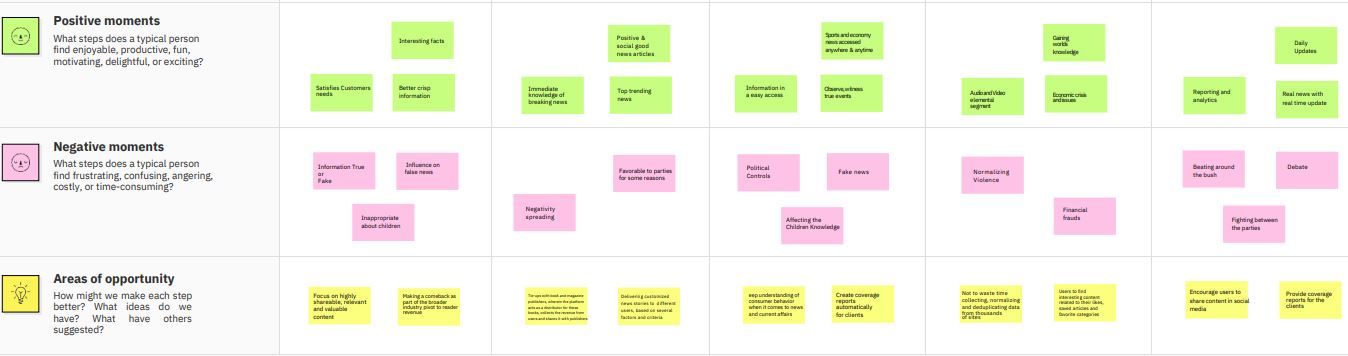
**Table 2: Components and Technologies**

|  |  |  |  |
| --- | --- | --- | --- |
| **S. No** | **Component** | **Description** | **Technology** |
| 1. | User Interface | The interaction between the user and application e.g., Web UI, Mobile App,Chatbot etc. | HTML, CSS, JavaScript / Bootstrapetc. |
| 2. | Application Logic-1 | Framework used for design the application. | Python, Python - Flask |
| 3. | Application Logic-2 | Accessing the cloud and storing the details of theusers both  donors and patients. | IBM Cloud, IBM DB2 |
| 4. | Application Logic-3 | Docker is an open-source platform for building,deploying,  and managing containerized applications. | Docker |
| 5. | Database | Data Type, Configurations etc. | SQL. |
| 6. | Cloud Database | Database Service on Cloud | IBM Cloud, IBM DB2 etc. |
| 7. | File Storage | File storage requirements | IBM Block Storage or Other Storage Service or Local Files ystem |
| 8. | External API-1 | They make it easier for  developers to store,manage and deploy container images. | Container Registry |
| 9. | Infrastructure (Server / Cloud) | Application Deployment on Local System / Cloud | Local, Cloud Foundry, Kubernetes, etc. |

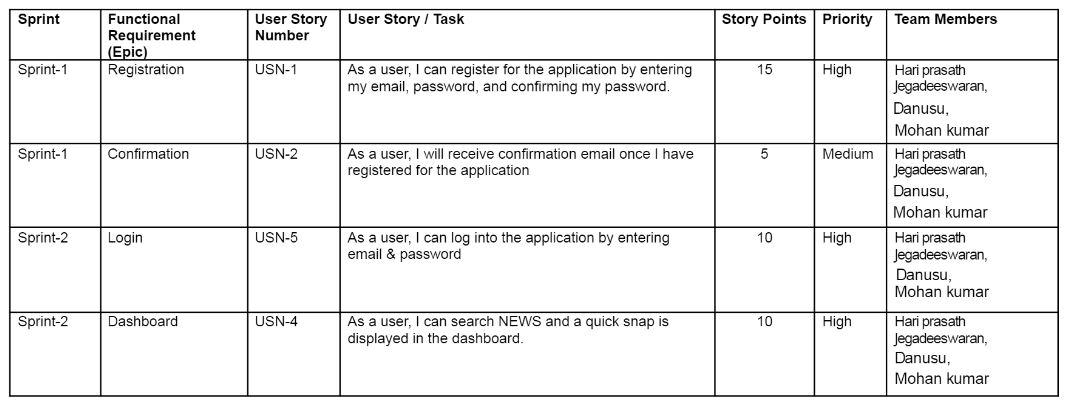
* 1. **User Stories**

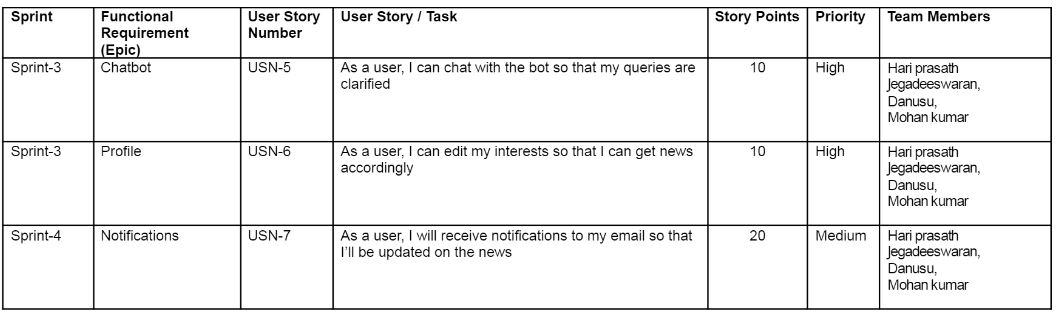
A user story is an informal, general explanation of a software feature written from the perspective of the end user or customer. The purpose of a user story is to articulate how a piece of work will deliver a particular value back to the customer. Note that "customers" don't have to be external end users in the traditional sense, they can also be internal customers or colleagues within your organization who depend on your team. User stories are a few sentences in simple language that outline the desired outcome. Stories fit neatly into agile frameworks like [scrum](https://www.atlassian.com/agile/scrum) and [kanban](https://www.atlassian.com/agile/kanban). Kanban teams pull user stories into their backlog and run them through their workflow. It’s this work on user stories that help scrum teams get better at [estimation](https://www.atlassian.com/agile/project-management/estimation) and sprint planning, leading to more accurate forecasting and greater agility.

****

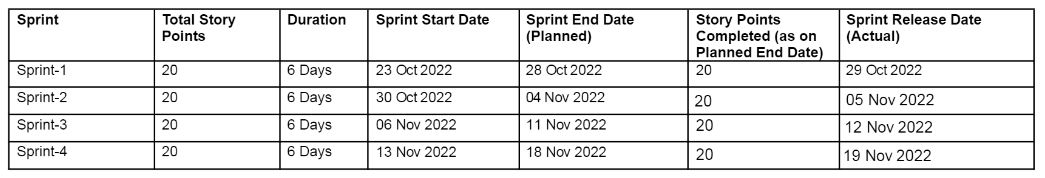
****

1. **PROJECT PLANNING AND SCHEDULING**
   1. **Sprint Planning And Estimation**

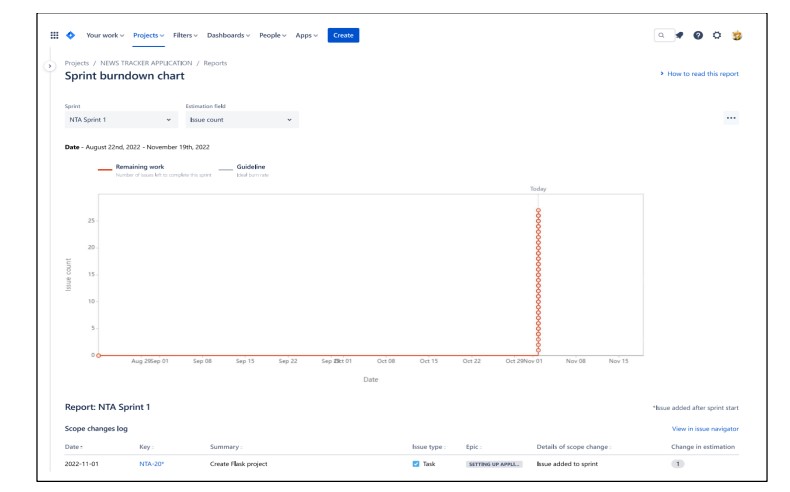
****

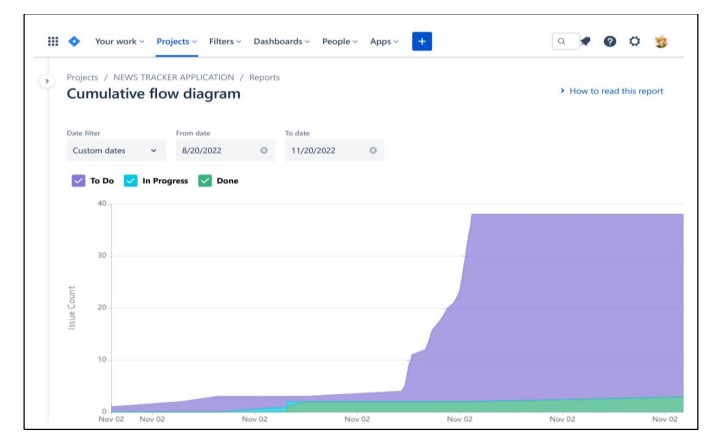
****

* 1. **Sprint Delivery Schedule**

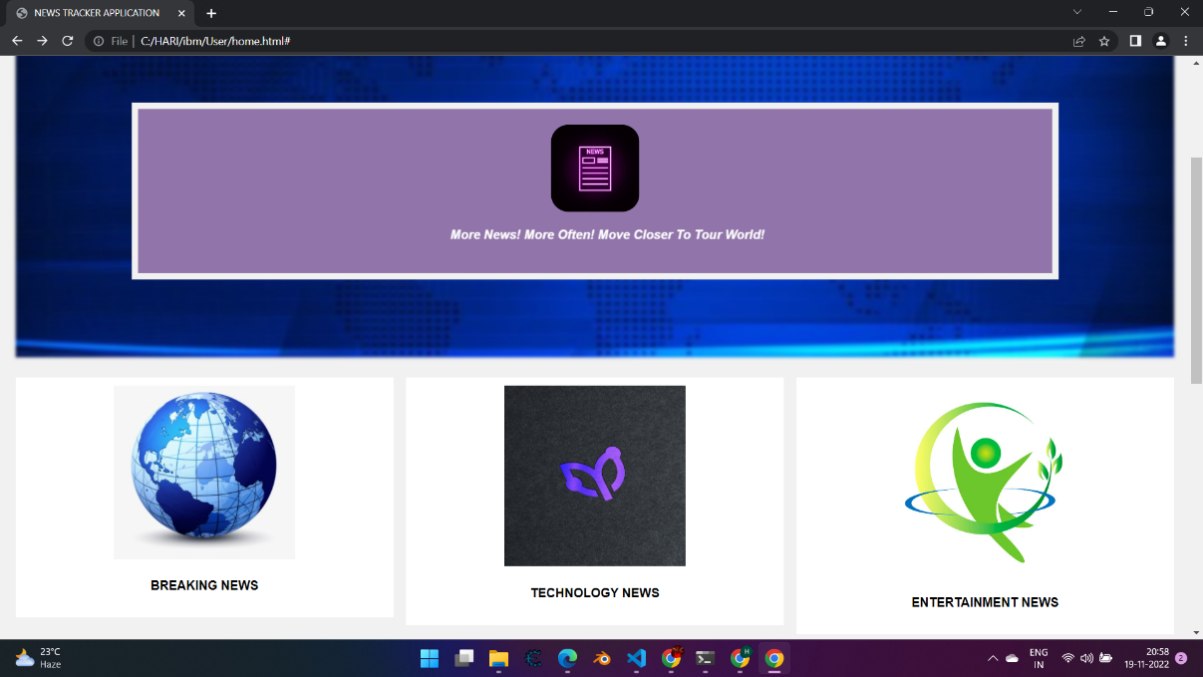
****

* 1. **Reports from JIRA**

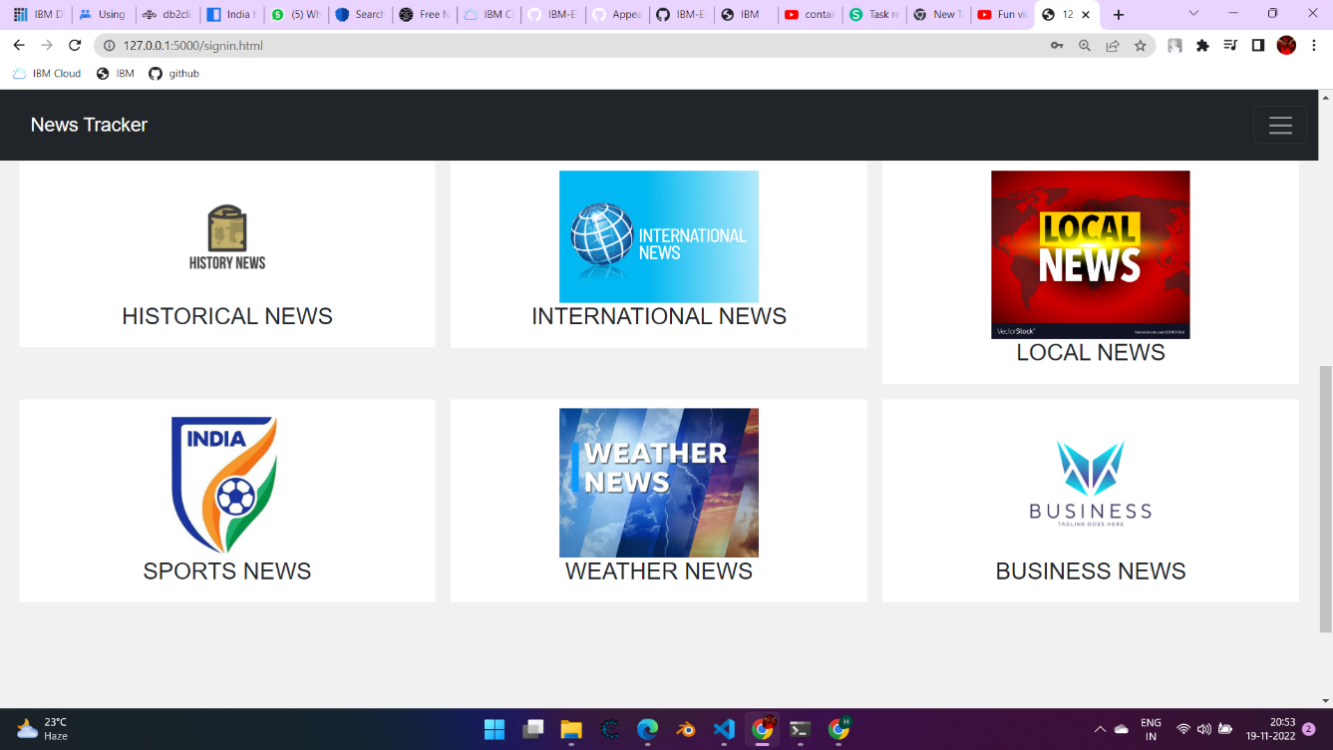




1. **CODING AND SOLUTIONING**
   1. **Feature 1**

****

* 1. **Feature 2**

****

* 1. **Database Schema IBM Db2**
* A hybrid ANSI-compliant data virtualization tool for accessing, querying and summarizing data across the enterprise which:
* Provides a massively parallel processing (MPP) architecture Exploits Hive, HBase and Apache Spark concurrently for best-in-class analytic capabilities
* Requires only a single database connection or query to connect disparate sources such as HDFS, RDMS, NoSQL databases, object stores andWeb HDFS
* Provides low latency support for ad-hoc and complex queries,high performance, and federation capabilities
* Understands dialects from other vendors and various productsfrom Oracle, IBM® Db2® and IBM Netezza®
* Enables advanced row and column security

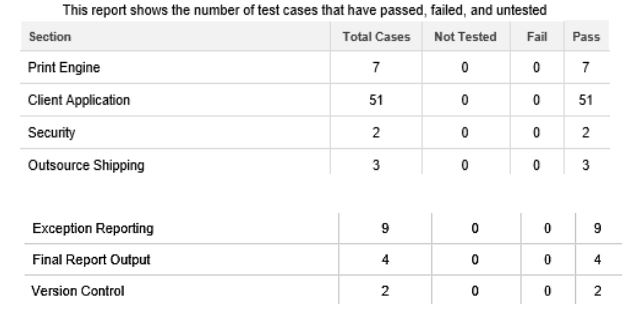
**KUBERNATES-**

* Kubernetes — also known as “k8s” or “kube” — is a container orchestration platform for scheduling and automating the deployment, management, and scaling of containerized applications.
* Kubernetes was first developed by engineers at Google before being open sourced in 2014. It is a descendant of Borg, a container orchestrationplatform used internally at Google. Kubernetes is Greek
* for *helmsman* or *pilot*, hence the helm in the [Kubernetes logo](https://github.com/cncf/artwork/tree/master/projects/kubernetes) (link residesoutside IBM).
* Today, Kubernetes and the broader container ecosystem are maturing into a general-purpose computing platform and ecosystem that rivals

— if notsurpasses — virtual machines (VMs) as the basic building blocks of modern cloud infrastructure and applications.

* This ecosystem enables organizations to deliver a high- productivity [Platform-as-a-Service (PaaS)](https://www.ibm.com/in-en/cloud/learn/paas) that addresses multipleinfrastructure- related and operations-related tasks and issues
* surrounding [cloud-native](https://www.ibm.com/in-en/cloud/learn/cloud-native) development so that development teams can focussolely on coding and innovation.

1. **TESTING**
   1. **Test Cases**

****

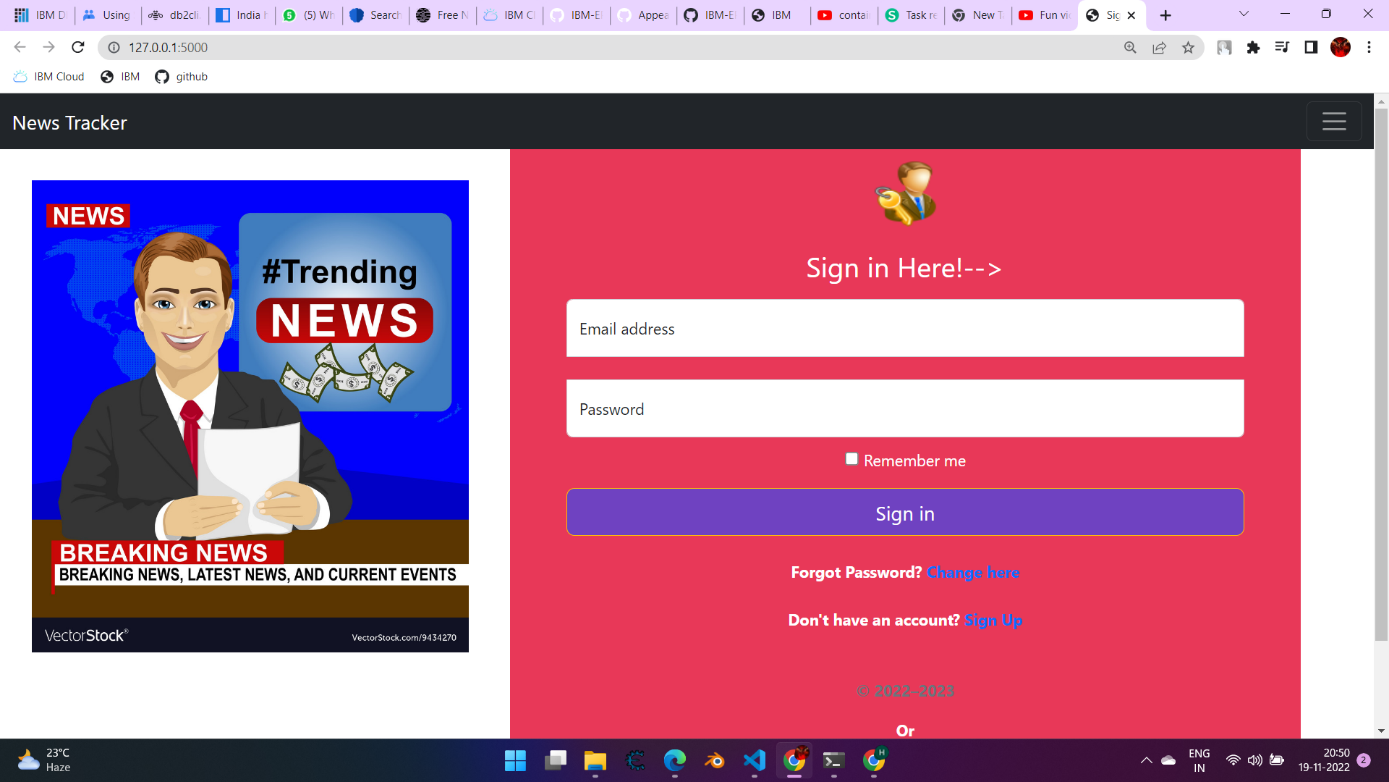
* 1. **User Acceptance Testing**

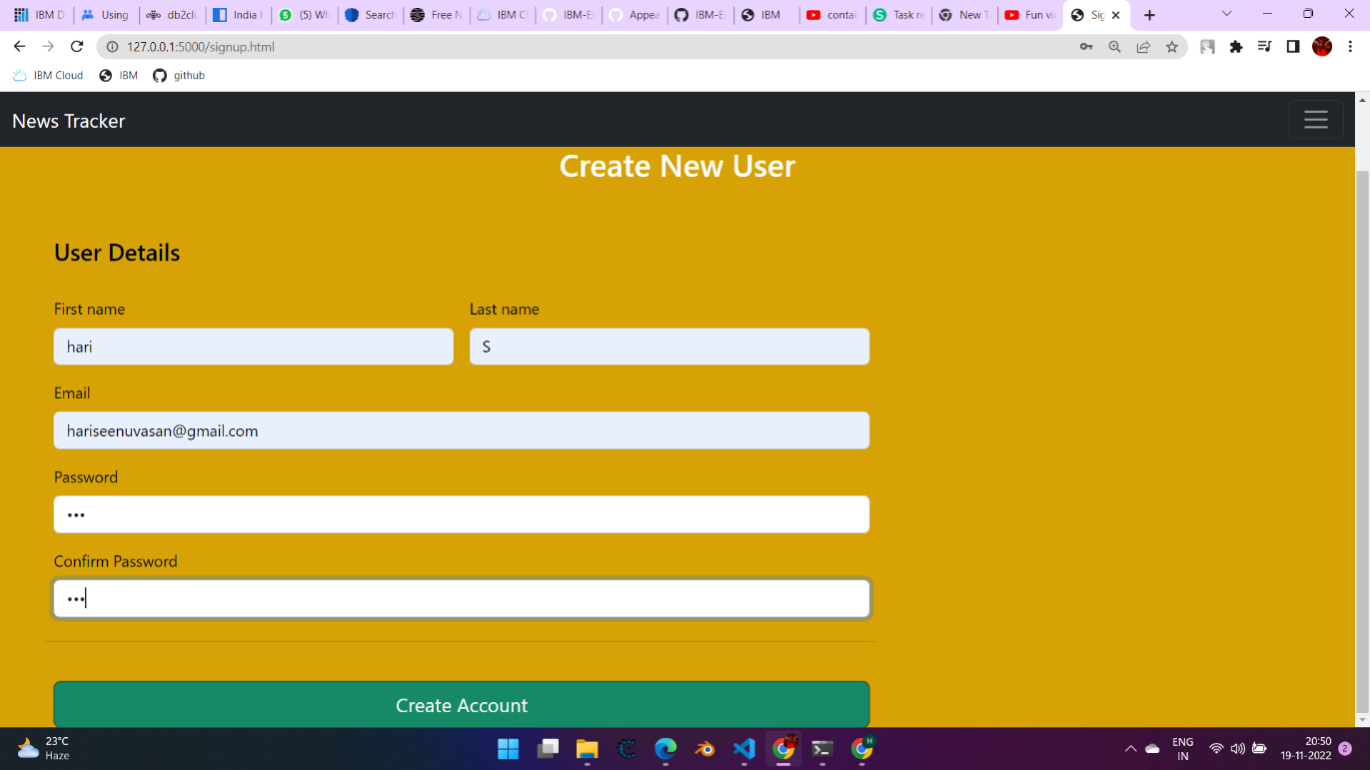
Acceptance Testing is a level of the software testing where a system is tested for acceptability. The purpose of this test is to evaluate the system’s compliance with the business requirements and assess whether it is acceptable for delivery.

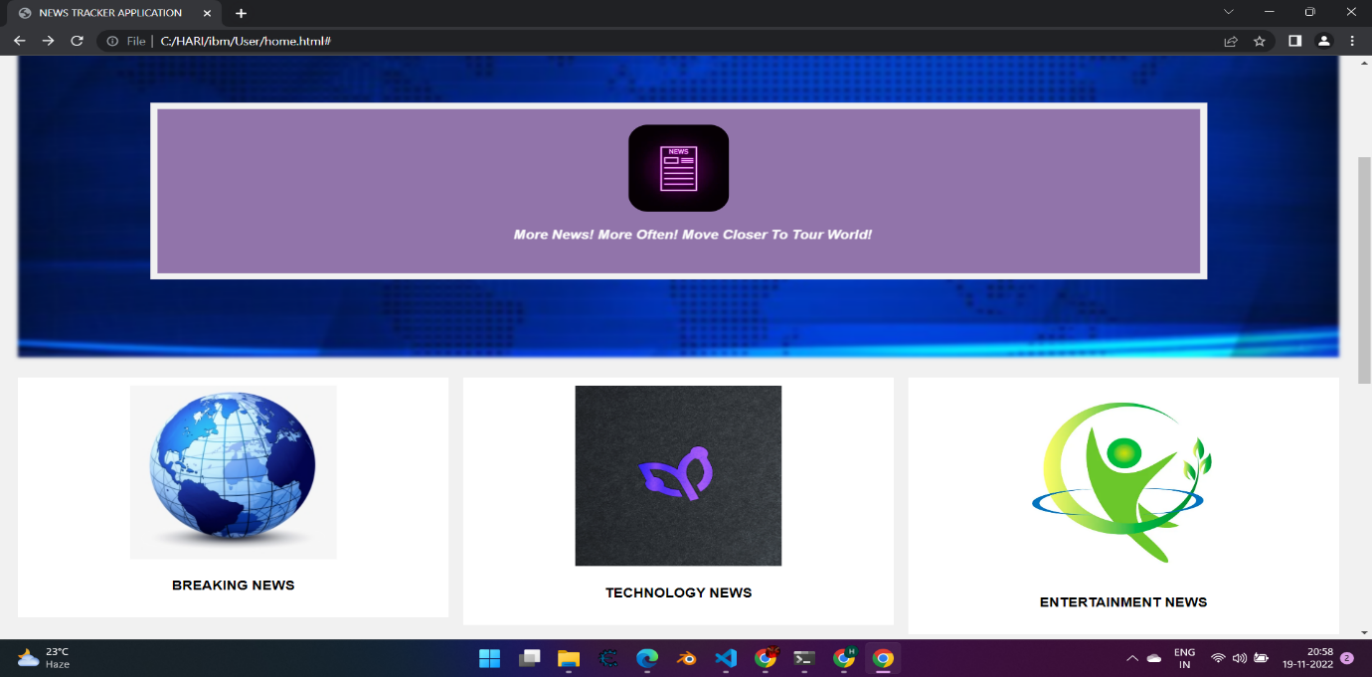
Formal testing with respect to user needs, requirements, and business processes conducted to determine whether or not a system satisfies the acceptance criteria and to enable the user, customers or other authorized entity to determine whether or not to accept the system.

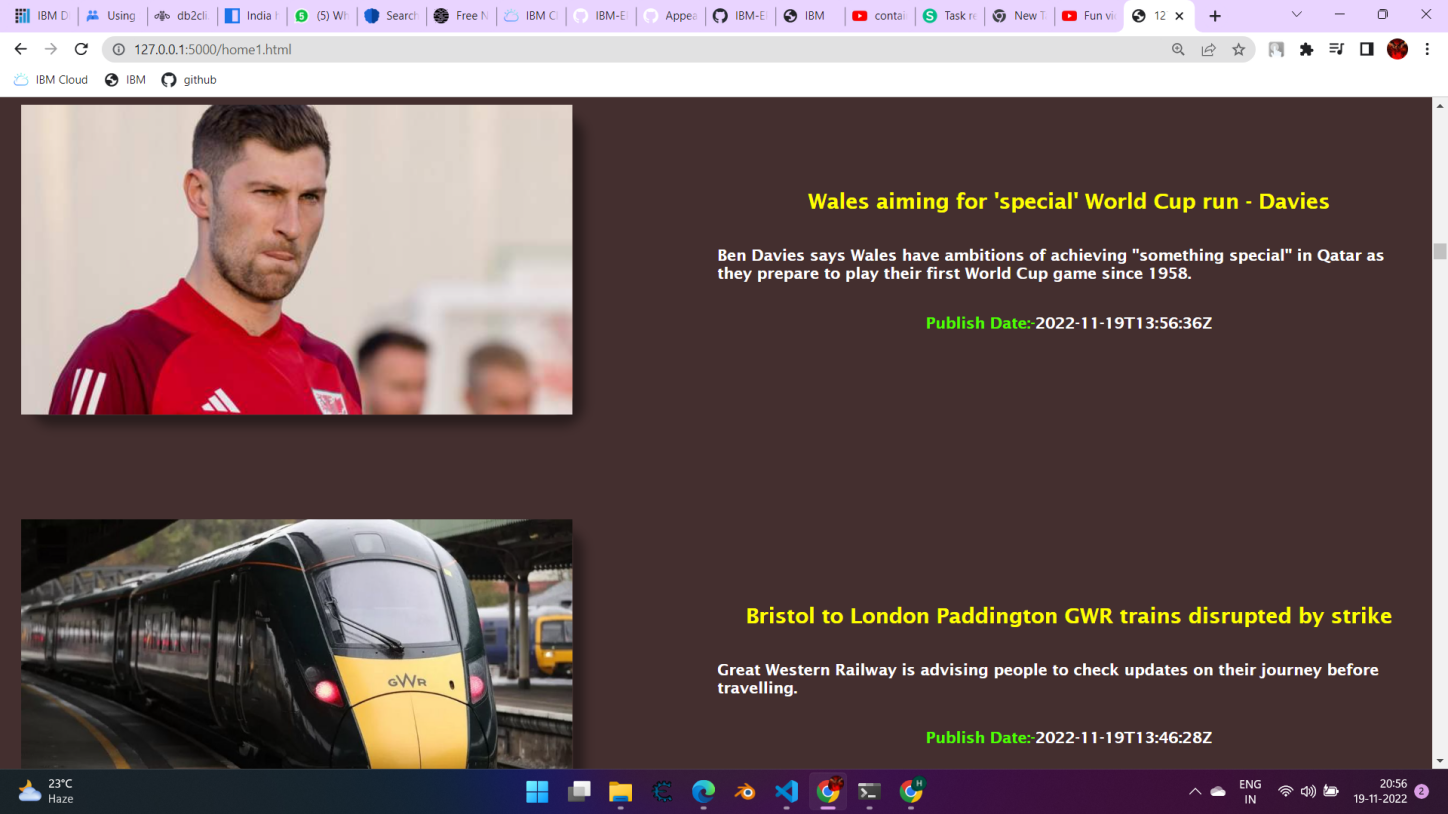
In this webpage, the customer’s acceptance is been monitored and it is been put into usage.

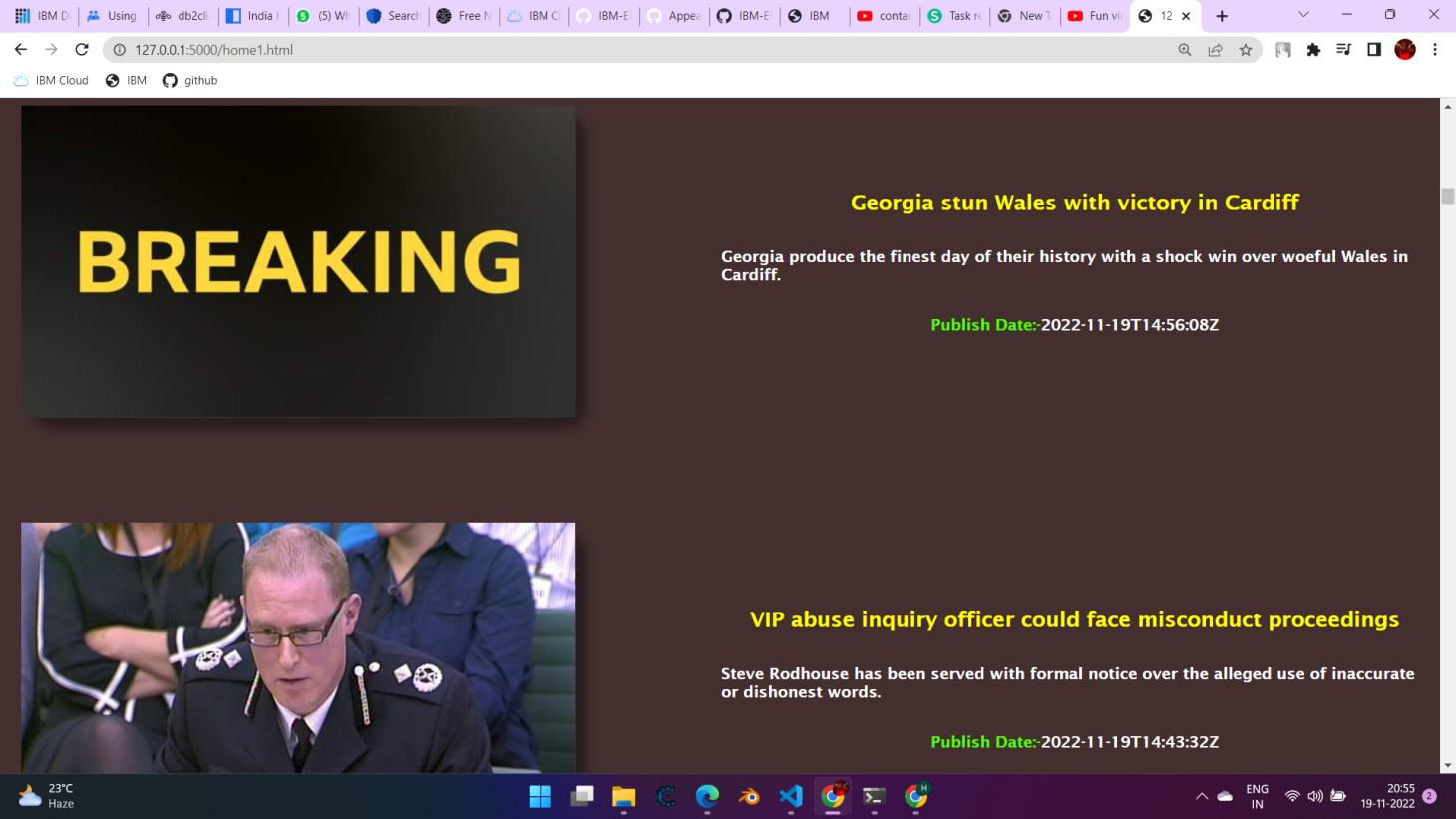
1. **RESULTS**
   1. **Performance Metrics**





****





1. **ADVANTAGES AND DISADVANTAGES**

**ADVANTAGES**

❖ In this app news is already categorization.

❖ Easily accessible and portable.

❖ Better user experience.

❖ Apps help you convert visitors to loyal readers.

❖ Minute by minute updates of news.

❖ This app helps you to get local news updates instantly.

❖ To explore and discover trending news and topics.

❖ News feeds for you based on your interest.

**DISADVANTAGES**

❖ Some apps demand premium subscription from user.

❖ Occurrence of Advertisement disturb the user

.

❖ Sometimes the news gives brief information.

❖ Prevalence of fake and uncertain news can confuse the user lead to

Misconception.

❖ Fake news may mislead the readers.

1. **CONCLUSION**

We explored the feasibility of recognising patterns of news reading interactions and evaluated three adaptive interface designs for different news reader types.We show that from their interaction log, a specific user can be recognised as one of three kinds. The reader types emerging from the online survey are well defined and distinct. The evaluation of the three variant interfaces suggests that different news reader types need different user interfaces. We have demonstrated a method for monitoring users’ news reading behaviour and inferring news reader type from it. In the future we will further explore the design of adaptive interfaces, in order to be in a position to demonstrate a complete adaptive mobile news framework providing automatic personalisation of news apps.

**FUTURE SCOPE**

In the future we will further explore the design of adaptive interfaces, in order to be in a position to demonstrate a complete adaptive mobile news framework providing automatic personalisation of news apps.

1. **APPENDIX**
   1. **Source Code**

**Front End**

**Base.html**

<!DOCTYPE html>

<html lang="en">

<head>

{% block head %}

<meta charset="utf-8">

<meta name="viewport" content="width=device-width , initial-scale=1">

<!--Bootstrap CSS-->

<link href="https://cdn.jsdelivr.net/npm/bootstrap@5.2.1/dist/css/bootstrap.min.css" rel="stylesheet" integrity="sha384-iYQeCzEYFbKjA/T2uDLTpkwGzCiq6soy8tYaI1GyVh/UjpbCx/TYkiZhlZB6+fzT" crossorigin="anonymous">

<link rel="stylesheet" type="text/css" href="../static/sheets/styles1.css">

<link rel="stylesheet" type="text/css" href="../static/sheets/style2.css">

<!--Google Fonts-->

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

<link href="https://fonts.googleapis.com/css2?family=Noto+Sans+JP:wght@300&display=swap" rel="stylesheet">

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

<link href="https://fonts.googleapis.com/css2?family=Nunito+Sans:ital@1&display=swap" rel="stylesheet">

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

<link href="https://fonts.googleapis.com/css2?family=Oxygen:wght@300&display=swap" rel="stylesheet">

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

<link href="https://fonts.googleapis.com/css2?family=Playfair+Display:wght@500&display=swap" rel="stylesheet">

{% endblock %}

</head>

<body>

<nav class="navbar navbar-dark bg-dark fixed-top">

<div class="container-fluid">

<a class="navbar-brand" href="#">News Tracker</a>

<button class="navbar-toggler" type="button" data-bs-toggle="offcanvas" data-bs-target="#offcanvasDarkNavbar" aria-controls="offcanvasDarkNavbar">

<span class="navbar-toggler-icon"></span>

</button>

<div class="offcanvas offcanvas-end text-bg-dark" tabindex="-1" id="offcanvasDarkNavbar" aria-labelledby="offcanvasDarkNavbarLabel">

<div class="offcanvas-header">

<h5 class="offcanvas-title" id="offcanvasDarkNavbarLabel">Profile</h5>

<button type="button" class="btn-close btn-close-white" data-bs-dismiss="offcanvas" aria-label="Close"></button>

</div>

<div class="offcanvas-body">

<ul class="navbar-nav justify-content-end flex-grow-1 pe-3">

<li class="nav-item">

<a class="nav-link active" aria-current="page" href="home.html">Home</a>

</li>

<li class="nav-item">

<a class="nav-link" href="base.html">Fetch News</a>

</li>

<li class="nav-item">

<a class="nav-link" href="about.html">About Us</a>

</li>

<li class="nav-item">

<a class="nav-link" href="signin.html">Sign In</a>

</li>

<li class="nav-item">

<a class="nav-link" href="signup.html">Sign Up</a>

</li>

</div>

</div>

</div>

</nav>

<div class="header">

<div class="bg-image"></div><div class="bg-text">

<img src="https://newstracker1.s3.us-south.cloud-object-storage.appdomain.cloud/img1.jpg" alt="Avatar" class="avatar">

<p><marquee><b><i>More News! More Often! Move Closer To Tour World!</i></b></marquee></p>

</div>

<div class="row">

<div class="column nature">

<div class="content">

<a href="/home1.html"><img src="https://newstracker1.s3.us-south.cloud-object-storage.appdomain.cloud/Earth.png" alt="breaking news" style="width:50%">

<h4>BREAKING NEWS</h4></a>

</div>

</div>

<div class="column nature">

<div class="content">

<img src="https://newstracker1.s3.us-south.cloud-object-storage.appdomain.cloud/Tech.png" alt="Lights" style="width:50%">

<h4>TECHNOLOGY NEWS</h4>

</div>

</div>

<div class="column nature">

<div class="content">

<img src="https://newstracker1.s3.us-south.cloud-object-storage.appdomain.cloud/Lifestyle.jpg" alt="Nature" style="width:50%">

<h4>ENTERTAINMENT NEWS</h4>

<p></p>

</div>

</div>

<div class="column cars">

<div class="content">

<img src="https://newstracker1.s3.us-south.cloud-object-storage.appdomain.cloud/history.png" alt="Car" style="width:50%">

<h4>HISTORICAL NEWS</h4></a>

</div>

</div>

<div class="column cars">

<div class="content">

<img src="https://newstracker1.s3.us-south.cloud-object-storage.appdomain.cloud/international.jpg" alt="Car" style="width:50%">

<h4>INTERNATIONAL NEWS</h4>

</div>

</div>

<div class="column cars">

<div class="content">

<img src="https://newstracker1.s3.us-south.cloud-object-storage.appdomain.cloud/local.jpg" alt="Car" style="width:50%">

<h4>LOCAL NEWS</h4>

</div>

</div>

<div class="column people">

<div class="content">

<img src="https://newstracker1.s3.us-south.cloud-object-storage.appdomain.cloud/sprots.jpg" alt="Car" style="width:50%">

<h4>SPORTS NEWS</h4>

</div>

</div>

<div class="column people">

<div class="content">

<img src="https://newstracker1.s3.us-south.cloud-object-storage.appdomain.cloud/weather.webp" alt="Car" style="width:50%">

<h4>WEATHER NEWS</h>

</div>

</div>

<div class="column people">

<div class="content">

<img src="https://newstracker1.s3.us-south.cloud-object-storage.appdomain.cloud/Business.jpg" alt="Car" style="width:50%">

<h4>BUSINESS NEWS</h4>

</div>

</div>

</div>

</div>

<script src="../static/sheets/Script grid.js"></script>

<br></div>

<div class="footer">

<h2>NEWS TRACKER APPLICATION</h2>

</div>

</body>

</html>

**about.html**

{% extends "base.html" %}

{% block head %}

<title>About Page</title>

{% endblock %}

{% block body %}

<h1>Welcome to About Page</h1>

<p> This is a paragraph</p>

{% endblock %}

**change pass.html**

<!doctype html>

<html lang="en">

<head>

<meta charset="utf-8">

<meta name="viewport" content="width=device-width, initial-scale=1">

<meta name="description" content="">

<meta name="author" content="Mark Otto, Jacob Thornton, and Bootstrap contributors">

<meta name="generator" content="Hugo 0.84.0">

<title>Sign In</title>

<link rel="canonical" href="https://getbootstrap.com/docs/5.0/examples/sign-in/">

<link href="https://getbootstrap.com/docs/5.0/assets/css/docs.css" rel="stylesheet">

<!-- Bootstrap core CSS -->

<link href="https://cdn.jsdelivr.net/npm/bootstrap@5.2.1/dist/css/bootstrap.min.css" rel="stylesheet" integrity="sha384-iYQeCzEYFbKjA/T2uDLTpkwGzCiq6soy8tYaI1GyVh/UjpbCx/TYkiZhlZB6+fzT" crossorigin="anonymous">

<!-- Favicons -->

<link rel="apple-touch-icon" href="/docs/5.0/assets/img/favicons/apple-touch-icon.png" sizes="180x180">

<link rel="icon" href="/docs/5.0/assets/img/favicons/favicon-32x32.png" sizes="32x32" type="image/png">

<link rel="icon" href="/docs/5.0/assets/img/favicons/favicon-16x16.png" sizes="16x16" type="image/png">

<link rel="manifest" href="/docs/5.0/assets/img/favicons/manifest.json">

<link rel="mask-icon" href="/docs/5.0/assets/img/favicons/safari-pinned-tab.svg" color="#7952b3">

<link rel="icon" href="/docs/5.0/assets/img/favicons/favicon.ico">

<meta name="theme-color" content="#7952b3">

<style>

.bd-placeholder-img {

font-size: 1.125rem;

text-anchor: middle;

-webkit-user-select: none;

-moz-user-select: none;

user-select: none;

}

@media (min-width: 768px) {

.bd-placeholder-img-lg {

font-size: 3.5rem;

}

}

</style>

<!-- Custom styles for this template -->

<link href="../static/sheets/sign in.css" rel="stylesheet">

<link href="../static/sheets/colors.css" rel="stylesheet">

</head>

<body class="text-center">

<nav class="navbar navbar-dark bg-dark fixed-top">

<div class="container-fluid">

<a class="navbar-brand" href="#">News Tracker</a>

<button class="navbar-toggler" type="button" data-bs-toggle="offcanvas" data-bs-target="#offcanvasDarkNavbar" aria-controls="offcanvasDarkNavbar">

<span class="navbar-toggler-icon"></span>

</button>

<div class="offcanvas offcanvas-end text-bg-dark" tabindex="-1" id="offcanvasDarkNavbar" aria-labelledby="offcanvasDarkNavbarLabel">

<div class="offcanvas-header">

<h5 class="offcanvas-title" id="offcanvasDarkNavbarLabel">Profile</h5>

<button type="button" class="btn-close btn-close-white" data-bs-dismiss="offcanvas" aria-label="Close"></button>

</div>

<div class="offcanvas-body">

<ul class="navbar-nav justify-content-end flex-grow-1 pe-3">

<li class="nav-item">

<a class="nav-link active" aria-current="page" href="home.html">Home</a>

</li>

<li class="nav-item">

<a class="nav-link" href="base.html">Fetch News</a>

</li>

<li class="nav-item">

<a class="nav-link" href="about.html">About Us</a>

</li>

<li class="nav-item">

<a class="nav-link" href="signin.html">Sign In</a>

</li>

<li class="nav-item">

<a class="nav-link" href="signup.html">Sign Up</a>

</li>

</div>

</div>

</div>

</nav>

<main class="form-signin ">

<form action = "{{ url\_for("get\_User1")}}" method="POST">

<img class="mb-4" src="static/Images/house-user-solid.svg" alt="" width="72" height="57">

<h1 class="h3 mb-3 fw-normal white">Change Password</h1>

<div class="form-floating">

<input type="email" class="form-control" id="floatingInput" name = "uname" placeholder="name@example.com">

<label for="floatingPassword">Password</label>

</div>

<br>

<div class="form-floating">

<input type="password" class="form-control" id="floatingPassword" name = "upwd" placeholder="Password">

<label for="floatingPassword">Confirm Password</label>

</div>

<button class="w-100 btn btn-lg bd-green-500 btn-outline-warning" type="submit">Change</button><br><br>

</form>

</main>

<script src="https://cdn.jsdelivr.net/npm/bootstrap@5.2.1/dist/js/bootstrap.bundle.min.js" integrity="sha384-u1OknCvxWvY5kfmNBILK2hRnQC3Pr17a+RTT6rIHI7NnikvbZlHgTPOOmMi466C8" crossorigin="anonymous"></script>

<script src="https://cdn.jsdelivr.net/npm/@docsearch/js@3"></script>

<script src="https://cdn.jsdelivr.net/npm/@stackblitz/sdk@1/bundles/sdk.umd.js"></script>

<script src="/docs/5.2/assets/js/docs.min.js"></script>

<script>

document.querySelectorAll('.btn-edit').forEach(btn => {

btn.addEventListener('click', event => {

const htmlSnippet = event.target.closest('.bd-code-snippet').querySelector('.bd-example').innerHTML

const classes = Array.from(event.target.closest('.bd-code-snippet').querySelector('.bd-example').classList).join(' ')

const jsSnippet = event.target.closest('.bd-code-snippet').querySelector('.btn-edit').getAttribute('data-sb-js-snippet')

StackBlitzSDK.openBootstrapSnippet(htmlSnippet, jsSnippet, classes)

})

})

StackBlitzSDK.openBootstrapSnippet = (htmlSnippet, jsSnippet, classes) => {

const markup = `<!doctype html>

<html lang="en">

<head>

<meta charset="utf-8">

<meta name="viewport" content="width=device-width, initial-scale=1">

<link href="https:\/\/cdn.jsdelivr.net\/npm\/bootstrap@5.2.1\/dist\/css\/bootstrap.min.css" rel="stylesheet">

<link href="https://getbootstrap.com/docs/5.2/assets/css/docs.css" rel="stylesheet">

<title>Bootstrap Example</title>

<${'script'} src="https:\/\/cdn\.jsdelivr\.net\/npm\/bootstrap@5\.2\.1\/dist\/js\/bootstrap\.bundle\.min\.js"></${'script'}>

</head>

<body class="p-3 m-0 border-0 ${classes}">

<!-- Example Code -->

${htmlSnippet.replace(/^/gm, ' ')}

<!-- End Example Code -->

</body>

</html>`

const project = {

files: {

'index.html': markup,

'index.js': jsSnippetContent

},

title: 'Bootstrap Example',

description: `Official example from ${window.location.href}`,

template: jsSnippet ? 'javascript' : 'html',

tags: ['bootstrap']

}

StackBlitzSDK.openProject(project, { openFile: 'index.html' })

}

</script>

</body>

</html>

**home.html**

<!doctype html>

<html lang="en">

<head>

<meta charset="utf-8">

<meta name="viewport" content="width=device-width, initial-scale=1">

<meta name="description" content="">

<meta name="author" content="Mark Otto, Jacob Thornton, and Bootstrap contributors">

<meta name="generator" content="Hugo 0.84.0">

<title>Sign In</title>

<link rel="canonical" href="https://getbootstrap.com/docs/5.0/examples/sign-in/">

<link href="https://getbootstrap.com/docs/5.0/assets/css/docs.css" rel="stylesheet">

<!-- Bootstrap core CSS -->

<link href="https://cdn.jsdelivr.net/npm/bootstrap@5.2.1/dist/css/bootstrap.min.css" rel="stylesheet" integrity="sha384-iYQeCzEYFbKjA/T2uDLTpkwGzCiq6soy8tYaI1GyVh/UjpbCx/TYkiZhlZB6+fzT" crossorigin="anonymous">

<!-- Favicons -->

<link rel="apple-touch-icon" href="/docs/5.0/assets/img/favicons/apple-touch-icon.png" sizes="180x180">

<link rel="icon" href="/docs/5.0/assets/img/favicons/favicon-32x32.png" sizes="32x32" type="image/png">

<link rel="icon" href="/docs/5.0/assets/img/favicons/favicon-16x16.png" sizes="16x16" type="image/png">

<link rel="manifest" href="/docs/5.0/assets/img/favicons/manifest.json">

<link rel="mask-icon" href="/docs/5.0/assets/img/favicons/safari-pinned-tab.svg" color="#7952b3">

<link rel="icon" href="/docs/5.0/assets/img/favicons/favicon.ico">

<meta name="theme-color" content="#7952b3">

<style>

.bd-placeholder-img {

font-size: 1.125rem;

text-anchor: middle;

-webkit-user-select: none;

-moz-user-select: none;

user-select: none;

}

@media (min-width: 768px) {

.bd-placeholder-img-lg {

font-size: 3.5rem;

}

}

</style>

<!-- Custom styles for this template -->

<link href="signin.css" rel="stylesheet">

<link href="colors.css" rel="stylesheet">

</head>

<body class="text-center">

<nav class="navbar navbar-dark bg-dark fixed-top">

<div class="container-fluid">

<a class="navbar-brand" href="#">About Me</a>

<button class="navbar-toggler" type="button" data-bs-toggle="offcanvas" data-bs-target="#offcanvasDarkNavbar" aria-controls="offcanvasDarkNavbar">

<span class="navbar-toggler-icon"></span>

</button>

<div class="offcanvas offcanvas-end text-bg-dark" tabindex="-1" id="offcanvasDarkNavbar" aria-labelledby="offcanvasDarkNavbarLabel">

<div class="offcanvas-header">

<h5 class="offcanvas-title" id="offcanvasDarkNavbarLabel">Profile</h5>

<button type="button" class="btn-close btn-close-white" data-bs-dismiss="offcanvas" aria-label="Close"></button>

</div>

<div class="offcanvas-body">

<ul class="navbar-nav justify-content-end flex-grow-1 pe-3">

<li class="nav-item">

<a class="nav-link active" aria-current="page" href="home.html">Home</a>

</li>

<li class="nav-item">

<a class="nav-link" href="base.html">Main</a>

</li>

<li class="nav-item">

<a class="nav-link" href="about.html">About</a>

</li>

<li class="nav-item">

<a class="nav-link" href="signin.html">Sign In</a>

</li>

<li class="nav-item">

<a class="nav-link" href="signup.html">Sign Up</a>

</li>

</div>

</div>

</div>

</nav>

<main class="form-signin">

<form action="{{ url\_for("login")}}">

<div class="form-floating">

<h1 class="pink div2">Sorry for the inconvience caused.</h1><br>

</div>

<br>

<div class="form-floating">

<h3 class="red div2">You have entered the wrong credentials.</h3><br>

</div>

<button class="w-100 btn btn-lg bd-purple-500 btn-outline-warning center" type="submit">Return To Home Page</button><br><br>

</form>

</main>

<script src="https://cdn.jsdelivr.net/npm/bootstrap@5.2.1/dist/js/bootstrap.bundle.min.js" integrity="sha384-u1OknCvxWvY5kfmNBILK2hRnQC3Pr17a+RTT6rIHI7NnikvbZlHgTPOOmMi466C8" crossorigin="anonymous"></script>

<script src="https://cdn.jsdelivr.net/npm/@docsearch/js@3"></script>

<script src="https://cdn.jsdelivr.net/npm/@stackblitz/sdk@1/bundles/sdk.umd.js"></script>

<script src="/docs/5.2/assets/js/docs.min.js"></script>

<script>

document.querySelectorAll('.btn-edit').forEach(btn => {

btn.addEventListener('click', event => {

const htmlSnippet = event.target.closest('.bd-code-snippet').querySelector('.bd-example').innerHTML

const classes = Array.from(event.target.closest('.bd-code-snippet').querySelector('.bd-example').classList).join(' ')

const jsSnippet = event.target.closest('.bd-code-snippet').querySelector('.btn-edit').getAttribute('data-sb-js-snippet')

StackBlitzSDK.openBootstrapSnippet(htmlSnippet, jsSnippet, classes)

})

})

StackBlitzSDK.openBootstrapSnippet = (htmlSnippet, jsSnippet, classes) => {

const markup = `<!doctype html>

<html lang="en">

<head>

<meta charset="utf-8">

<meta name="viewport" content="width=device-width, initial-scale=1">

<link href="https:\/\/cdn.jsdelivr.net\/npm\/bootstrap@5.2.1\/dist\/css\/bootstrap.min.css" rel="stylesheet">

<link href="https://getbootstrap.com/docs/5.2/assets/css/docs.css" rel="stylesheet">

<title>Bootstrap Example</title>

<${'script'} src="https:\/\/cdn\.jsdelivr\.net\/npm\/bootstrap@5\.2\.1\/dist\/js\/bootstrap\.bundle\.min\.js"></${'script'}>

</head>

<body class="p-3 m-0 border-0 ${classes}">

<!-- Example Code -->

${htmlSnippet.replace(/^/gm, ' ')}

<!-- End Example Code -->

</body>

</html>`

const project = {

files: {

'index.html': markup,

'index.js': jsSnippetContent

},

title: 'Bootstrap Example',

description: `Official example from ${window.location.href}`,

template: jsSnippet ? 'javascript' : 'html',

tags: ['bootstrap']

}

StackBlitzSDK.openProject(project, { openFile: 'index.html' })

}

</script>

</body>

</html>

**error.html**

<!doctype html>

<html lang="en">

<head>

<meta charset="utf-8">

<meta name="viewport" content="width=device-width, initial-scale=1">

<meta name="description" content="">

<meta name="author" content="Mark Otto, Jacob Thornton, and Bootstrap contributors">

<meta name="generator" content="Hugo 0.84.0">

<title>Sign In</title>

<link rel="canonical" href="https://getbootstrap.com/docs/5.0/examples/sign-in/">

<link href="https://getbootstrap.com/docs/5.0/assets/css/docs.css" rel="stylesheet">

<!-- Bootstrap core CSS -->

<link href="https://cdn.jsdelivr.net/npm/bootstrap@5.2.1/dist/css/bootstrap.min.css" rel="stylesheet" integrity="sha384-iYQeCzEYFbKjA/T2uDLTpkwGzCiq6soy8tYaI1GyVh/UjpbCx/TYkiZhlZB6+fzT" crossorigin="anonymous">

<!-- Favicons -->

<link rel="apple-touch-icon" href="/docs/5.0/assets/img/favicons/apple-touch-icon.png" sizes="180x180">

<link rel="icon" href="/docs/5.0/assets/img/favicons/favicon-32x32.png" sizes="32x32" type="image/png">

<link rel="icon" href="/docs/5.0/assets/img/favicons/favicon-16x16.png" sizes="16x16" type="image/png">

<link rel="manifest" href="/docs/5.0/assets/img/favicons/manifest.json">

<link rel="mask-icon" href="/docs/5.0/assets/img/favicons/safari-pinned-tab.svg" color="#7952b3">

<link rel="icon" href="/docs/5.0/assets/img/favicons/favicon.ico">

<meta name="theme-color" content="#7952b3">

<style>

.bd-placeholder-img {

font-size: 1.125rem;

text-anchor: middle;

-webkit-user-select: none;

-moz-user-select: none;

user-select: none;

}

@media (min-width: 768px) {

.bd-placeholder-img-lg {

font-size: 3.5rem;

}

}

</style>

<!-- Custom styles for this template -->

<link href="signin.css" rel="stylesheet">

<link href="colors.css" rel="stylesheet">

</head>

<body class="text-center">

<nav class="navbar navbar-dark bg-dark fixed-top">

<div class="container-fluid">

<a class="navbar-brand" href="#">About Me</a>

<button class="navbar-toggler" type="button" data-bs-toggle="offcanvas" data-bs-target="#offcanvasDarkNavbar" aria-controls="offcanvasDarkNavbar">

<span class="navbar-toggler-icon"></span>

</button>

<div class="offcanvas offcanvas-end text-bg-dark" tabindex="-1" id="offcanvasDarkNavbar" aria-labelledby="offcanvasDarkNavbarLabel">

<div class="offcanvas-header">

<h5 class="offcanvas-title" id="offcanvasDarkNavbarLabel">Profile</h5>

<button type="button" class="btn-close btn-close-white" data-bs-dismiss="offcanvas" aria-label="Close"></button>

</div>

<div class="offcanvas-body">

<ul class="navbar-nav justify-content-end flex-grow-1 pe-3">

<li class="nav-item">

<a class="nav-link active" aria-current="page" href="home.html">Home</a>

</li>

<li class="nav-item">

<a class="nav-link" href="base.html">Main</a>

</li>

<li class="nav-item">

<a class="nav-link" href="about.html">About</a>

</li>

<li class="nav-item">

<a class="nav-link" href="signin.html">Sign In</a>

</li>

<li class="nav-item">

<a class="nav-link" href="signup.html">Sign Up</a>

</li>

</div>

</div>

</div>

</nav>

<main class="form-signin">

<form action="{{ url\_for("login")}}">

<div class="form-floating">

<h1 class="pink div2">Sorry for the inconvience caused.</h1><br>

</div>

<br>

<div class="form-floating">

<h3 class="red div2">You have entered the wrong credentials.</h3><br>

</div>

<button class="w-100 btn btn-lg bd-purple-500 btn-outline-warning center" type="submit">Return To Home Page</button><br><br>

</form>

</main>

<script src="https://cdn.jsdelivr.net/npm/bootstrap@5.2.1/dist/js/bootstrap.bundle.min.js" integrity="sha384-u1OknCvxWvY5kfmNBILK2hRnQC3Pr17a+RTT6rIHI7NnikvbZlHgTPOOmMi466C8" crossorigin="anonymous"></script>

<script src="https://cdn.jsdelivr.net/npm/@docsearch/js@3"></script>

<script src="https://cdn.jsdelivr.net/npm/@stackblitz/sdk@1/bundles/sdk.umd.js"></script>

<script src="/docs/5.2/assets/js/docs.min.js"></script>

<script>

document.querySelectorAll('.btn-edit').forEach(btn => {

btn.addEventListener('click', event => {

const htmlSnippet = event.target.closest('.bd-code-snippet').querySelector('.bd-example').innerHTML

const classes = Array.from(event.target.closest('.bd-code-snippet').querySelector('.bd-example').classList).join(' ')

const jsSnippet = event.target.closest('.bd-code-snippet').querySelector('.btn-edit').getAttribute('data-sb-js-snippet')

StackBlitzSDK.openBootstrapSnippet(htmlSnippet, jsSnippet, classes)

})

})

StackBlitzSDK.openBootstrapSnippet = (htmlSnippet, jsSnippet, classes) => {

const markup = `<!doctype html>

<html lang="en">

<head>

<meta charset="utf-8">

<meta name="viewport" content="width=device-width, initial-scale=1">

<link href="https:\/\/cdn.jsdelivr.net\/npm\/bootstrap@5.2.1\/dist\/css\/bootstrap.min.css" rel="stylesheet">

<link href="https://getbootstrap.com/docs/5.2/assets/css/docs.css" rel="stylesheet">

<title>Bootstrap Example</title>

<${'script'} src="https:\/\/cdn\.jsdelivr\.net\/npm\/bootstrap@5\.2\.1\/dist\/js\/bootstrap\.bundle\.min\.js"></${'script'}>

</head>

<body class="p-3 m-0 border-0 ${classes}">

<!-- Example Code -->

${htmlSnippet.replace(/^/gm, ' ')}

<!-- End Example Code -->

</body>

</html>`

const project = {

files: {

'index.html': markup,

'index.js': jsSnippetContent

},

title: 'Bootstrap Example',

description: `Official example from ${window.location.href}`,

template: jsSnippet ? 'javascript' : 'html',

tags: ['bootstrap']

}

StackBlitzSDK.openProject(project, { openFile: 'index.html' })

}

</script>

</body>

</html>

**for pass.html**

<!doctype html>

<html lang="en">

<head>

<meta charset="utf-8">

<meta name="viewport" content="width=device-width, initial-scale=1">

<meta name="description" content="">

<meta name="author" content="Mark Otto, Jacob Thornton, and Bootstrap contributors">

<meta name="generator" content="Hugo 0.84.0">

<title>Sign In</title>

<link rel="canonical" href="https://getbootstrap.com/docs/5.0/examples/sign-in/">

<link href="https://getbootstrap.com/docs/5.0/assets/css/docs.css" rel="stylesheet">

<!-- Bootstrap core CSS -->

<link href="https://cdn.jsdelivr.net/npm/bootstrap@5.2.1/dist/css/bootstrap.min.css" rel="stylesheet" integrity="sha384-iYQeCzEYFbKjA/T2uDLTpkwGzCiq6soy8tYaI1GyVh/UjpbCx/TYkiZhlZB6+fzT" crossorigin="anonymous">

<!-- Favicons -->

<link rel="apple-touch-icon" href="/docs/5.0/assets/img/favicons/apple-touch-icon.png" sizes="180x180">

<link rel="icon" href="/docs/5.0/assets/img/favicons/favicon-32x32.png" sizes="32x32" type="image/png">

<link rel="icon" href="/docs/5.0/assets/img/favicons/favicon-16x16.png" sizes="16x16" type="image/png">

<link rel="manifest" href="/docs/5.0/assets/img/favicons/manifest.json">

<link rel="mask-icon" href="/docs/5.0/assets/img/favicons/safari-pinned-tab.svg" color="#7952b3">

<link rel="icon" href="/docs/5.0/assets/img/favicons/favicon.ico">

<meta name="theme-color" content="#7952b3">

<style>

.bd-placeholder-img {

font-size: 1.125rem;

text-anchor: middle;

-webkit-user-select: none;

-moz-user-select: none;

user-select: none;

}

@media (min-width: 768px) {

.bd-placeholder-img-lg {

font-size: 3.5rem;

}

}

</style>

<!-- Custom styles for this template -->

<link href="../static/sheets/sign in.css" rel="stylesheet">

<link href="../static/sheets/colors.css" rel="stylesheet">

</head>

<body class="text-center">

<nav class="navbar navbar-dark bg-dark fixed-top">

<div class="container-fluid">

<a class="navbar-brand" href="#">News Tracker</a>

<button class="navbar-toggler" type="button" data-bs-toggle="offcanvas" data-bs-target="#offcanvasDarkNavbar" aria-controls="offcanvasDarkNavbar">

<span class="navbar-toggler-icon"></span>

</button>

<div class="offcanvas offcanvas-end text-bg-dark" tabindex="-1" id="offcanvasDarkNavbar" aria-labelledby="offcanvasDarkNavbarLabel">

<div class="offcanvas-header">

<h5 class="offcanvas-title" id="offcanvasDarkNavbarLabel">Profile</h5>

<button type="button" class="btn-close btn-close-white" data-bs-dismiss="offcanvas" aria-label="Close"></button>

</div>

<div class="offcanvas-body">

<ul class="navbar-nav justify-content-end flex-grow-1 pe-3">

<li class="nav-item">

<a class="nav-link active" aria-current="page" href="home.html">Home</a>

</li>

<li class="nav-item">

<a class="nav-link" href="base.html">Fetch News</a>

</li>

<li class="nav-item">

<a class="nav-link" href="about.html">About Us</a>

</li>

<li class="nav-item">

<a class="nav-link" href="signin.html">Sign In</a>

</li>

<li class="nav-item">

<a class="nav-link" href="signup.html">Sign Up</a>

</li>

</div>

</div>

</div>

</nav>

<main class="form-signin ">

<form action = "{{ url\_for("get\_User")}}" method="POST">

<img class="mb-4" src="static/Images/house-user-solid.svg" alt="" width="72" height="57">

<h1 class="h3 mb-3 fw-normal white">Change your password</h1>

<div class="form-floating">

<input type="email" class="form-control" id="floatingInput" name = "uname" placeholder="name@example.com">

<label for="floatingInput">Email address</label>

</div>

<br>

<button class="w-100 btn btn-lg btn-warning btn-outline-warning" type="submit"><a href="static/templates/changepass.html">Change Password</a></button><br><br>

</form>

</main>

<script src="https://cdn.jsdelivr.net/npm/bootstrap@5.2.1/dist/js/bootstrap.bundle.min.js" integrity="sha384-u1OknCvxWvY5kfmNBILK2hRnQC3Pr17a+RTT6rIHI7NnikvbZlHgTPOOmMi466C8" crossorigin="anonymous"></script>

<script src="https://cdn.jsdelivr.net/npm/@docsearch/js@3"></script>

<script src="https://cdn.jsdelivr.net/npm/@stackblitz/sdk@1/bundles/sdk.umd.js"></script>

<script src="/docs/5.2/assets/js/docs.min.js"></script>

<script>

document.querySelectorAll('.btn-edit').forEach(btn => {

btn.addEventListener('click', event => {

const htmlSnippet = event.target.closest('.bd-code-snippet').querySelector('.bd-example').innerHTML

const classes = Array.from(event.target.closest('.bd-code-snippet').querySelector('.bd-example').classList).join(' ')

const jsSnippet = event.target.closest('.bd-code-snippet').querySelector('.btn-edit').getAttribute('data-sb-js-snippet')

StackBlitzSDK.openBootstrapSnippet(htmlSnippet, jsSnippet, classes)

})

})

StackBlitzSDK.openBootstrapSnippet = (htmlSnippet, jsSnippet, classes) => {

const markup = `<!doctype html>

<html lang="en">

<head>

<meta charset="utf-8">

<meta name="viewport" content="width=device-width, initial-scale=1">

<link href="https:\/\/cdn.jsdelivr.net\/npm\/bootstrap@5.2.1\/dist\/css\/bootstrap.min.css" rel="stylesheet">

<link href="https://getbootstrap.com/docs/5.2/assets/css/docs.css" rel="stylesheet">

<title>Bootstrap Example</title>

<${'script'}

src="https:\/\/cdn\.jsdelivr\.net\/npm\/bootstrap@5\.2\.1\/dist\/js\/bootstrap\.bundle\.min\.js"></${'script'}>

</head>

<body class="p-3 m-0 border-0 ${classes}">

<!-- Example Code -->

${htmlSnippet.replace(/^/gm, ' ')}

<!-- End Example Code -->

</body>

</html>`

const project = {

files: {

'index.html': markup,

'index.js': jsSnippetContent

},

title: 'Bootstrap Example',

description: `Official example from ${window.location.href}`,

template: jsSnippet ? 'javascript' : 'html',

tags: ['bootstrap']

}

StackBlitzSDK.openProject(project, { openFile: 'index.html' })

}

</script>

</body>

</html>

**home1.html**

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

<meta http-equiv="X-UA-Compatible" content="IE=edge">

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

<link rel="stylesheet" href="{{url\_for('static',filename='../static/sheets/style3.css')}}">

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

<link rel="icon" href="{{url\_for('static', filename='img/logo.png')}}">

<link href="https://fonts.googleapis.com/icon?family=Material+Icons" rel="stylesheet">

<title>{% block title %}{% endblock %}</title>

</head>

<body>

<nav>

<a href="/" class="logo"><marquee><h2>News Tracker</h2></a></marquee>

<ul class= "menu">

</ul>

</nav>

<section id="main">

<div class="banner"><img src="https://t4.ftcdn.net/jpg/03/37/92/49/360\_F\_337924964\_fF6I0iZkv9lvoIEn71f0OsgawArIPPDu.jpg">

</div>

<div class = "head">TODAY'S TOP HEADLINES</div>

{% for title,desc,img,p\_date,url in contents %}

<div class="wrapper">

<div class="item"><a href="{{url}}"><img src="{{img}}"></a></div>

<div class = "c-box">

<div class="content"><a href="{{url}}"><h2>{{title}}</h2></a></div>

<div class="content"><a href="{{url}}"><p>{{desc}}</p></a></div>

<div class="content"><p style="color:rgb(77, 255, 0);">Publish Date:- </p><p>{{p\_date}}</p></div>

</div>

</div>

{% endfor %}

<div class = "head">TODAY'S NEWS</div>

{% for title,desc,img,p\_date,url in all %}

<div class="wrapper">

<div class="item"><a href="{{url}}"><img src="{{img}}"></a></div>

<div class = "c-box">

<div class="content"><a href="{{url}}"><h2>{{title}}</h2></a></div>

<div class="content"><p>{{desc}}</p></div>

<div class="content"><p style="color:rgb(77, 255, 0);">Publish Date:- </p><p> {{p\_date}} </p></div>

</div>

</div>

{% endfor %}

</section>

<footer>

<div class="head-git"><h2>For IBM Nalayathiran 2022</a></h2></div>

<div class="head-f"><h2> Copyright &copy; 2022</h2></div>

</footer>

</body>

</html>

**signin.html**

<!doctype html>

<html lang="en">

<head>

<meta charset="utf-8">

<meta name="viewport" content="width=device-width, initial-scale=1">

<meta name="description" content="">

<meta name="author" content="Mark Otto, Jacob Thornton, and Bootstrap contributors">

<meta name="generator" content="Hugo 0.84.0">

<title>Sign In</title>

<link rel="canonical" href="https://getbootstrap.com/docs/5.0/examples/sign-in/" >

<link href="https://getbootstrap.com/docs/5.0/assets/css/docs.css" rel="stylesheet">

<!-- Bootstrap core CSS -->

<link href="https://cdn.jsdelivr.net/npm/bootstrap@5.2.1/dist/css/bootstrap.min.css"

rel="stylesheet" integrity="sha384-

iYQeCzEYFbKjA/T2uDLTpkwGzCiq6soy8tYaI1GyVh/UjpbCx/TYkiZhlZB6+fzT" crossorigin="anonymous">

<!-- Favicons -->

<link rel="apple-touch-icon" href="/docs/5.0/assets/img/favicons/apple-touch-icon.png" sizes="180x180">

<link rel="icon" href="/docs/5.0/assets/img/favicons/favicon-32x32.png" sizes="32x32" type="image/png">

<link rel="icon" href="/docs/5.0/assets/img/favicons/favicon-16x16.png" sizes="16x16" type="image/png">

<link rel="manifest" href="/docs/5.0/assets/img/favicons/manifest.json">

<link rel="mask-icon" href="/docs/5.0/assets/img/favicons/safari-pinned-tab.svg" color="#7952b3">

<link rel="icon" href="/docs/5.0/assets/img/favicons/favicon.ico">

<meta name="theme-color" content="#7952b3">

<style>

.bd-placeholder-img {

font-size: 1.125rem;

text-anchor: middle;

-webkit-user-select: none;

-moz-user-select: none;

user-select: none;

}

@media (min-width: 300px) {

.bd-placeholder-img-lg {

font-size: 3.5rem;

}

}

</style>

<!-- Custom styles for this template -->

<link href="../static/sheets/sign in.css" rel="stylesheet">

<link href="../static/sheets/colors.css" rel="stylesheet">

</head>

<body class="text-center">

<nav class="navbar navbar-dark bg-dark fixed-top">

<div class="container-fluid">

<a class="navbar-brand" href="#">News Tracker</a>

<button class="navbar-toggler" type="button" data-bs-toggle="offcanvas" data-bs-target="#offcanvasDarkNavbar" aria-controls="offcanvasDarkNavbar">

<span class="navbar-toggler-icon"></span>

</button>

<div class="offcanvas offcanvas-end text-bg-dark" tabindex="-1" id="offcanvasDarkNavbar" aria-labelledby="offcanvasDarkNavbarLabel">

<div class="offcanvas-header">

<h5 class="offcanvas-title" id="offcanvasDarkNavbarLabel">Profile</h5>

<button type="button" class="btn-close btn-close-white" data-bs-dismiss="offcanvas" aria-label="Close"></button>

</div>

<div class="offcanvas-body">

<ul class="navbar-nav justify-content-end flex-grow-1 pe-3">

<li class="nav-item">

<a class="nav-link active" aria-current="page" href="home.html">Home</a>

</li>

<li class="nav-item">

<a class="nav-link" href="base.html">Fetch News</a>

</li>

<li class="nav-item">

<a class="nav-link" href="about.html">About Us</a>

</li>

<li class="nav-item">

<a class="nav-link" href="signin.html" >Sign In</a>

</li>

<li class="nav-item">

<a class="nav-link" href="signup.html">Sign Up</a>

</li>

</div>

</div>

</div>

</nav>

<div class="row g-0 d-flex align-items-center">

<div class="col-lg-4 d-lg-flex">

<img src="https://newstracker1.s3.us-south.cloud-object-storage.appdomain.cloud/images1.jpg" alt="News"

class="w-100 rounded-t-5 rounded-tr-lg-0 rounded-bl-lg-5" />

</div>

<main class="form-signin ">

<form action = "{{ url\_for("getUser")}}" method="POST">

<center><img class="mb-4" src="https://newstracker1.s3.us-south.cloud-object-storage.appdomain.cloud/login-icon-3058-Windows.ico" alt="" width="72" height="57" >

<h1 class="h3 mb-3 fw-normal white" >Sign in Here!--></h1></center>

<div class="form-floating">

<input type="email" class="form-control" id="floatingInput" name = "uname" placeholder="name@example.com">

<label for="floatingInput">Email address</label>

</div>

<br>

<div class="form-floating">

<input type="password" class="form-control" id="floatingPassword" name = "upwd" placeholder="Password">

<label for="floatingPassword">Password</label>

</div>

<div class="checkbox mb-3">

<label class="white">

<input type="checkbox" value="remember-me"> Remember me

</label>

</div>

<button class="w-100 btn btn-lg bd-purple-500 btn-outline-warning" type="submit">Sign in</button><br><br>

<p class="white"><b>Forgot Password? <a href="forpass.html">Change here</b></a></p>

<p class="white"><b>Don't have an account? <a href="signup.html">Sign Up</b></a></p>

<p class="mt-5 mb-3 text-muted"><b>&copy; 2022–2023</b></p>

<p class="white"><b>Or</b></p>

<p class="white"><b>Sign in with <img src="static/images/golo..png" alt=" " width="40" height="40"></b></p>

</form>

</main>

</div>

<script src="https://cdn.jsdelivr.net/npm/bootstrap@5.2.1/dist/js/bootstrap.bundle.min.js" integrity="sha384-u1OknCvxWvY5kfmNBILK2hRnQC3Pr17a+RTT6rIHI7NnikvbZlHgTPOOmMi466C8" crossorigin="anonymous"></script>

<script src="https://cdn.jsdelivr.net/npm/@docsearch/js@3"></script>

<script src="https://cdn.jsdelivr.net/npm/@stackblitz/sdk@1/bundles/sdk.umd.js"></script>

<script src="/docs/5.2/assets/js/docs.min.js"></script>

<script>

document.querySelectorAll('.btn-edit').forEach(btn => {

btn.addEventListener('click', event => {

const htmlSnippet = event.target.closest('.bd-code-snippet').querySelector('.bd-example').innerHTML

const classes = Array.from(event.target.closest('.bd-code-snippet').querySelector('.bd-example').classList).join(' ')

const jsSnippet = event.target.closest('.bd-code-snippet').querySelector('.btn-edit').getAttribute('data-sb-js-snippet')

StackBlitzSDK.openBootstrapSnippet(htmlSnippet, jsSnippet, classes)

})

})

StackBlitzSDK.openBootstrapSnippet = (htmlSnippet, jsSnippet, classes) => {

const markup = `<!doctype html>

<html lang="en">

<head>

<meta charset="utf-8">

<meta name="viewport" content="width=device-width, initial-scale=1">

<link href="https:\/\/cdn.jsdelivr.net\/npm\/bootstrap@5.2.1\/dist\/css\/bootstrap.min.css" rel="stylesheet">

<link href="https://getbootstrap.com/docs/5.2/assets/css/docs.css" rel="stylesheet">

<title>Bootstrap Example</title>

<${'script'} src="https:\/\/cdn\.jsdelivr\.net\/npm\/bootstrap@5\.2\.1\/dist\/js\/bootstrap\.bundle\.min\.js"></${'script'}>

</head>

<body class="p-3 m-0 border-0 ${classes}">

<!-- Example Code -->

${htmlSnippet.replace(/^/gm, ' ')}

<!-- End Example Code -->

</body>

</html>`

const project = {

files: {

'index.html': markup,

'index.js': jsSnippetContent

},

title: 'Bootstrap Example',

description: `Official example from ${window.location.href}`,

template: jsSnippet ? 'javascript' : 'html',

tags: ['bootstrap']

}

StackBlitzSDK.openProject(project, { openFile: 'index.html' })

}

</script>

</body>

</html>

**signup.html**

<!doctype html>

<html lang="en">

<head>

<meta charset="utf-8">

<meta name="viewport" content="width=device-width, initial-scale=1">

<meta name="description" content="">

<meta name="author" content="Mark Otto, Jacob Thornton, and Bootstrap contributors">

<meta name="generator" content="Hugo 0.84.0">

<title>Sign Up</title>

<link rel="stylesheet" type="text/css" href="../static/sheets/sign in.css">

<link rel="canonical" href="https://getbootstrap.com/docs/5.0/examples/checkout/">

<link href="https://cdn.jsdelivr.net/npm/bootstrap@5.1.3/dist/css/bootstrap.min.css" rel="stylesheet" integrity="sha384-1BmE4kWBq78iYhFldvKuhfTAU6auU8tT94WrHftjDbrCEXSU1oBoqyl2QvZ6jIW3" crossorigin="anonymous">

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

<link href="https://fonts.googleapis.com/css2?family=Noto+Sans+JP:wght@300&display=swap" rel="stylesheet">

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

<link href="https://fonts.googleapis.com/css2?family=Nunito+Sans:ital@1&display=swap" rel="stylesheet">

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

<link href="https://fonts.googleapis.com/css2?family=Oxygen:wght@300&display=swap" rel="stylesheet">

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

<link href="https://fonts.googleapis.com/css2?family=Playfair+Display:wght@500&display=swap" rel="stylesheet">

<!-- Bootstrap core CSS -->

<link href="https://cdn.jsdelivr.net/npm/bootstrap@5.2.1/dist/css/bootstrap.min.css" rel="stylesheet" integrity="sha384-iYQeCzEYFbKjA/T2uDLTpkwGzCiq6soy8tYaI1GyVh/UjpbCx/TYkiZhlZB6+fzT" crossorigin="anonymous">

<!-- Favicons -->

<link rel="apple-touch-icon" href="/docs/5.0/assets/img/favicons/apple-touch-icon.png" sizes="180x180">

<link rel="icon" href="/docs/5.0/assets/img/favicons/favicon-32x32.png" sizes="32x32" type="image/png">

<link rel="icon" href="/docs/5.0/assets/img/favicons/favicon-16x16.png" sizes="16x16" type="image/png">

<link rel="manifest" href="/docs/5.0/assets/img/favicons/manifest.json">

<link rel="mask-icon" href="/docs/5.0/assets/img/favicons/safari-pinned-tab.svg" color="#7952b3">

<link rel="icon" href="/docs/5.0/assets/img/favicons/favicon.ico">

<meta name="theme-color" content="#7952b3">

<style>

table{

text-align: center;

border:3px solid black;

border-collapse: collapse;

width: 100%;

height: 90px;

}

th{

border-collapse: collapse;

border: 2px solid black;

font-family: 'Oxygen', serif;

font-weight: 35px;

font-size: 25px;

height: 45px;

width: 60px;

color: black;

}

td{

border-collapse: collapse;

border: 2px solid white;

width: 60px;

height: 45px;

font-size:20px;

font-weight: 67px;

font-family: 'Oxygen', serif;

background-color: white;

color: black;

}

.tabrow

{

background-color: #00ff80;

}

.tabdat{

background-color: black;

color: white;

font-weight: bold;

}

</style>

<link rel="stylesheet" href="../static/sheets/colors.css">

<!-- Custom styles for this template -->

</head>

<body class="bd-yellow-600">

<nav class="navbar navbar-dark bg-dark fixed-top">

<div class="container-fluid">

<a class="navbar-brand" href="#">News Tracker</a>

<button class="navbar-toggler" type="button" data-bs-toggle="offcanvas" data-bs-target="#offcanvasDarkNavbar" aria-controls="offcanvasDarkNavbar">

<span class="navbar-toggler-icon"></span>

</button>

<div class="offcanvas offcanvas-end text-bg-dark" tabindex="-1" id="offcanvasDarkNavbar" aria-labelledby="offcanvasDarkNavbarLabel">

<div class="offcanvas-header">

<h5 class="offcanvas-title" id="offcanvasDarkNavbarLabel">Profile</h5>

<button type="button" class="btn-close btn-close-white" data-bs-dismiss="offcanvas" aria-label="Close"></button>

</div>

<div class="offcanvas-body">

<ul class="navbar-nav justify-content-end flex-grow-1 pe-3">

<li class="nav-item">

<li class="nav-item">

<a class="nav-link" href="signin.html">Sign In</a>

</li>

<li class="nav-item">

<a class="nav-link" href="signup.html">Sign Up</a>

</li>

</div>

</div>

</div>

</nav>

<div class="container div2">

<main>

<div class="py-5 text-center">

<h2 class="text-white div2">Create New User</h2>

</div>

<div class="col-md-7 col-lg-8">

<h4 class="mb-3 text-black">User Details</h4>

<form action = "{{ url\_for("storedUser")}}" method="POST">

<div class="row g-3">

<!--<div class="col-sm-6">-->

<!--<label for="firstName" class="form-label text-white">Account Number</label>-->

<!--<input type="text" class="form-control" id="acc\_no" name="acc\_no" placeholder="" value="" required>-->

<!--</div>-->

<div class="row g-3">

<div class="col-sm-6">

<label for="firstName" class="form-label text-black">First name</label>

<input type="text" class="form-control" id="firstName" name = "fname" placeholder="" value="" required>

<div class="invalid-feedback">

Valid first name is required.

</div>

</div>

<div class="col-sm-6">

<label for="lastName" class="form-label text-black">Last name</label>

<input type="text" class="form-control" id="lastName" name = "lname" placeholder="" value="" required>

<div class="invalid-feedback">

Valid last name is required.

</div>

</div>

<div class="col-12">

<label for="floatingInput" class="form-label text-black">Email</label>

<input type="email" class="form-control" id="email" name = "mail" placeholder="you@example.com">

</div>

<div class="col-12">

<label for="pass" class="form-label text-black">Password</label>

<input type="password" class="form-control" id="pass" name = "npwd" placeholder="Enter a Valid Password" required>

</div>

<div class="col-12">

<label for="confpass" class="form-label text-black">Confirm Password</label>

<input type="password" class="form-control" id="confpass" name="cpwd" placeholder="Enter a Valid Password" required>

</div>

<hr class="my-4">

<a href="#" alt="didnt connect"><button class="w-100 btn bd-teal-700 btn-outline-dark btn-lg" type="submit">Create Account</button></a>

</form>

</div>

</div>

</main>

<script src="https://cdn.jsdelivr.net/npm/bootstrap@5.2.1/dist/js/bootstrap.bundle.min.js" integrity="sha384-u1OknCvxWvY5kfmNBILK2hRnQC3Pr17a+RTT6rIHI7NnikvbZlHgTPOOmMi466C8" crossorigin="anonymous"></script>

<script src="https://cdn.jsdelivr.net/npm/@docsearch/js@3"></script>

<script src="https://cdn.jsdelivr.net/npm/@stackblitz/sdk@1/bundles/sdk.umd.js"></script>

<script src="/docs/5.2/assets/js/docs.min.js"></script>

<script>

document.querySelectorAll('.btn-edit').forEach(btn => {

btn.addEventListener('click', event => {

const htmlSnippet = event.target.closest('.bd-code-snippet').querySelector('.bd-example').innerHTML

const classes = Array.from(event.target.closest('.bd-code-snippet').querySelector('.bd-example').classList).join(' ')

const jsSnippet = event.target.closest('.bd-code-snippet').querySelector('.btn-edit').getAttribute('data-sb-js-snippet')

StackBlitzSDK.openBootstrapSnippet(htmlSnippet, jsSnippet, classes)

})

})

StackBlitzSDK.openBootstrapSnippet = (htmlSnippet, jsSnippet, classes) => {

const markup = `<!doctype html>

<html lang="en">

<head>

<meta charset="utf-8">

<meta name="viewport" content="width=device-width, initial-scale=1">

<link href="https:\/\/cdn.jsdelivr.net\/npm\/bootstrap@5.2.1\/dist\/css\/bootstrap.min.css" rel="stylesheet">

<link href="https://getbootstrap.com/docs/5.2/assets/css/docs.css" rel="stylesheet">

<title>Bootstrap Example</title>

<${'script'} src="https:\/\/cdn\.jsdelivr\.net\/npm\/bootstrap@5\.2\.1\/dist\/js\/bootstrap\.bundle\.min\.js"></${'script'}>

</head>

<body class="p-3 m-0 border-0 ${classes}">

<!-- Example Code -->

${htmlSnippet.replace(/^/gm, ' ')}

<!-- End Example Code -->

</body>

</html>`

const project = {

files: {

'index.html': markup,

'index.js': jsSnippetContent

},

title: 'Bootstrap Example',

description: `Official example from ${window.location.href}`,

template: jsSnippet ? 'javascript' : 'html',

tags: ['bootstrap']

}

StackBlitzSDK.openProject(project, { openFile: 'index.html' })

}

</script>

</body>

</html>

**Back End**

**app.py**

from glob import escape

import ibm\_db

import re

from newsapi import NewsApiClient

import pycountry

from flask import Flask, redirect, render\_template, request, session, url\_for

app = Flask(\_\_name\_\_)

hostname = 'b0aebb68-94fa-46ec-a1fc-1c999edb6187.c3n41cmd0nqnrk39u98g.databases.appdomain.cloud'

uid = 'qbc77613'

pwd = 'Ih7n42kjzJRpuViC'

driver = "{IBM DB2 ODBC DRIVER}"

db\_name = 'Bludb'

port = '31249'

protocol = 'TCPIP'

cert = "certi.crt"

dsn = (

"DATABASE ={0};"

"HOSTNAME ={1};"

"PORT ={2};"

"UID ={3};"

"SECURITY=SSL;"

"PROTOCOL={4};"

"PWD ={6};"

).format(db\_name, hostname, port, uid, protocol, cert, pwd)

connection = ibm\_db.connect(dsn, "", "")

app.secret\_key = 'a'

@app.route('/')

def login():

return render\_template('signin.html')

@app.route('/signin.html',methods = ['POST'])

def getUser():

if request.method == 'POST':

user = request.form['uname']

password = request.form['upwd']

sql = "SELECT \* FROM CUSTOMER1 where Email = ?"

stmt = ibm\_db.prepare(connection, sql)

email = user

# Explicitly bind parameters

ibm\_db.bind\_param(stmt, 1,user)

ibm\_db.execute(stmt)

dictionary = ibm\_db.fetch\_assoc(stmt)

pwd = dictionary["PASSWORD"]

if password != pwd:

return render\_template('error.html')

return render\_template('base.html')

@app.route('/signup.html')

def putUser():

return render\_template('signup.html')

@app.route('/signup.html',methods = ['POST'])

def storedUser():

if request.method == 'POST':

fname = request.form['fname']

lname = request.form['lname']

mail = request.form['mail']

npwd = request.form['npwd']

cpwd = request.form['cpwd']

print("inside checking")

if len(fname) == 0 or len(lname) == 0 or len(mail) == 0 or len(npwd)== 0 or len(cpwd) == 0:

msg = "Form is not filled completely!!"

print(msg)

return render\_template('signup.html', msg=msg)

elif npwd != cpwd:

msg = "Password is not matched"

print(msg)

return render\_template('signup.html', msg=msg)

elif not re.match(r'[^@]+@[^@]+\.[^@]+', mail):

msg = 'Invalid email'

print(msg)

return render\_template('signup.html', msg=msg)

elif not re.match(r'[A-Za-z]+', fname):

msg = "Enter valid Firstname"

print(msg)

return render\_template('signup.html', msg=msg)

elif not re.match(r'[A-Za-z]+', lname):

msg = "Enter valid LastName"

print(msg)

return render\_template('signup.html', msg=msg)

sql = "select \* from CUSTOMER1 where email = ?"

stmt = ibm\_db.prepare(conn, sql)

ibm\_db.bind\_param(stmt, 3, mail)

ibm\_db.execute(stmt)

account = ibm\_db.fetch\_assoc(stmt)

print(account)

if account:

msg = 'Acccount already exists'

else:

mail = mail

sql = "INSERT INTO CUSTOMER1 (FirstName,LastName,Email,password,confirmpassword) VALUES(?,?,?,?,?);"

stmt = ibm\_db.prepare(connection, sql)

ibm\_db.bind\_param(stmt, 1, fname)

ibm\_db.bind\_param(stmt, 2, lname)

ibm\_db.bind\_param(stmt, 3, mail)

ibm\_db.bind\_param(stmt, 4, npwd)

ibm\_db.bind\_param(stmt, 5, cpwd)

ibm\_db.execute(stmt)

print("successs")

msg = "succesfully signed up"

return render\_template('signin.html', msg=msg, name=name)

else:

return render\_template('signup.html')

@app.route('/forpass.html')

def get\_User():

return render\_template('forpass.html')

@app.route('/change pass.html')

def get\_User1():

return render\_template('change pass.html')

@app.route('/home1.html',methods = ['POST','GET'])

def home():

api\_key = 'f9ea965541334519819e9b3275eb54f8'

newsapi = NewsApiClient(api\_key=api\_key)

top\_headlines = newsapi.get\_top\_headlines(sources = "bbc-news")

all\_articles = newsapi.get\_everything(sources = "bbc-news")

t\_articles = top\_headlines['articles']

a\_articles = all\_articles['articles']

news = []

desc = []

img = []

p\_date = []

url = []

for i in range (len(t\_articles)):

main\_article = t\_articles[i]

news.append(main\_article['title'])

desc.append(main\_article['description'])

img.append(main\_article['urlToImage'])

p\_date.append(main\_article['publishedAt'])

url.append(main\_article['url'])

contents = zip( news,desc,img,p\_date,url)

news\_all = []

desc\_all = []

img\_all = []

p\_date\_all = []

url\_all = []

for j in range(len(a\_articles)):

main\_all\_articles = a\_articles[j]

news\_all.append(main\_all\_articles['title'])

desc\_all.append(main\_all\_articles['description'])

img\_all.append(main\_all\_articles['urlToImage'])

p\_date\_all.append(main\_all\_articles['publishedAt'])

url\_all.append(main\_article['url'])

all = zip( news\_all,desc\_all,img\_all,p\_date\_all,url\_all)

return render\_template('home1.html',contents=contents,all = all)

if \_\_name\_\_ == '\_\_main\_\_':

app.run(debug=True)

**app2.py**

from flask import Flask,render\_template

from newsapi import NewsApiClient

import pycountry

app = Flask(\_\_name\_\_)

@app.route("/")

def home():

api\_key = 'f9ea965541334519819e9b3275eb54f8'

newsapi = NewsApiClient(api\_key=api\_key)

top\_headlines = newsapi.get\_top\_headlines(sources = "bbc-news")

all\_articles = newsapi.get\_everything(sources = "bbc-news")

t\_articles = top\_headlines['articles']

a\_articles = all\_articles['articles']

news = []

desc = []

img = []

p\_date = []

url = []

for i in range (len(t\_articles)):

main\_article = t\_articles[i]

news.append(main\_article['title'])

desc.append(main\_article['description'])

img.append(main\_article['urlToImage'])

p\_date.append(main\_article['publishedAt'])

url.append(main\_article['url'])

contents = zip( news,desc,img,p\_date,url)

news\_all = []

desc\_all = []

img\_all = []

p\_date\_all = []

url\_all = []

for j in range(len(a\_articles)):

main\_all\_articles = a\_articles[j]

news\_all.append(main\_all\_articles['title'])

desc\_all.append(main\_all\_articles['description'])

img\_all.append(main\_all\_articles['urlToImage'])

p\_date\_all.append(main\_all\_articles['publishedAt'])

url\_all.append(main\_article['url'])

all = zip( news\_all,desc\_all,img\_all,p\_date\_all,url\_all)

return render\_template('home1.html',contents=contents,all = all)

if \_\_name\_\_ == '\_\_main\_\_':

app.run(debug=True)

**checkEmail.py**

from flask import request

from flask\_restful import Resource

from utils.dbQuery import selectQuery

class CheckEmail(Resource):

def post(self):

data=request.json

email=str(data["email"])

res=selectQuery("SELECT email FROM USER WHERE EMAIL=?",(email,)) print(res)

if(not res):

return {"status":True},200

return {"status":False},400

**getnews.py**

from random import random

from flask\_restful import Resource

from utils.cookieChecker import token\_required

from utils.dbQuery import selectQuery

from utils.apiFetch import apiData

import random

from utils.testData import retArray

class Personal(Resource):

@token\_required

def get(email, self):

topicsArr = ["sport", "tech", "world", "finance", "politics", "business",

"economics", "entertainment", "beauty", "travel", "music", "food",

"science", "cricket"]

fav = selectQuery("SELECT favourites from user where email=?",

(email,))[

'FAVOURITES']

fav = fav.split(',')

for x in fav:

topicsArr.remove(x)

retArr = []

favList = []

# favList = retArray

nonFavList = []

for x in fav:

try:

data = apiData(x)

except:

data=None

if (data is not None):

for y in data:

favList.append(y)

try:

random.shuffle(favList)

favList = sorted(favList, key=lambda k: k['date'], reverse=True)

except:

favList=[]

for x in topicsArr:

try:

data = apiData(x)

except:

data=None

if(data is not None):

for y in data:

nonFavList.append(y)

try:

random.shuffle(nonFavList)

nonFavList = sorted(nonFavList, key=lambda k: k['date'], reverse=True)

except:

nonFavList=[]

retArr = favList+nonFavList

return {"data": retArr}, 200

class News(Resource):

@token\_required

def get(email,self,topic):

topicsArr = ["headline","sport", "tech", "world", "finance", "politics",

"business",

"economics", "entertainment", "beauty", "travel", "music", "food",

"science", "cricket"]

if(topic not in topicsArr):

return {"status":"Not a valid topic"},404

try:

retArr=apiData(topic)

random.shuffle(retArr)

retArr=sorted(retArr,key=lambda k:k['date'],reverse=True)

except:

retArr=[]

return {"data":retArr},200

**login.py**

from datetime import datetime

from flask\_restful import Resource

from flask import request,after\_this\_request

from utils.password import checkPassword

from utils.dbQuery import \*

from dateparser import parse

from utils.tokener import generate\_confirmation\_token

from utils.emailSender import emailSender

import app

import jwt

class Login(Resource):

def post(self):

data=request.json

email=data["email"]

password=data["password"]

queryRes=selectQuery("SELECT \* from user where email=?",(email,))

res=checkPassword(password,queryRes["PASSWORD"])

if(not res):

return {"status":"Wrong credentials"},400

if(not queryRes["VERIFIED"]):

lastTime=parse(queryRes["RESEND\_TIME"])

currTime=datetime.now()

diff=currTime-lastTime

diff=diff.total\_seconds()

if(diff>3600):

token=generate\_confirmation\_token(email)

emailSender(email,token)

insertQuery("UPDATE user set RESEND\_TIME=? where

email=?",(datetime.now(),email))

return {"status":"Not verified"},400

@after\_this\_request

def cookieSender(response):

access\_token=jwt.encode(

{"email":email},app.app.config['SECRET\_KEY']

)

response.set\_cookie("access\_token",str(access\_token),httponly=True)

response.set\_cookie("email",str(email),httponly=True)

return response

return{"status":"Successfully Logged in"},200

**register.py**

from datetime import datetime

from flask import request,after\_this\_request

from flask\_restful import Resource

from utils.dbQuery import insertQuery

from utils.emailSender import newEmailSender

from utils.password import genHash

class Register(Resource):

def post(self):

req=request.json

name=req['name']

email=req['email']

password=req['password']

favourite=req['favourite']

fav=','.join(favourite)

if(name=='' or email=='' or password=='' or fav==''):

return {"status":"Missing data"},404

password=genHash(password)

t=(name,email,password,fav,datetime.now())

res=insertQuery('INSERT INTO user

(name,email,password,favourites,resend\_time) values (?,?,?,?,?)',t)

if(not res):

return {"status":"Error while registering"},400

@after\_this\_request

def emailer(response):

newEmailSender(email)

return response

return {"status":"Successfully registered"},200

**verify.py**

from flask import request

from flask\_restful import Resource

from utils.tokener import confirm\_token

from utils.dbQuery import insertQuery

class Verify(Resource):

def post(self):

data=request.json

token=data["token"]

email=confirm\_token(token)

if(not email):

return {"status":"Couldn't verify"},400

res=insertQuery("UPDATE user set verified=True where

email=?",(email,))

if(not res):

return {"status":"Couldn't Update"},400

return {"status":"verified successfully"},200

**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

EXPOSE 5000

CMD ["python","app.py"]

**Deployment.xaml**

apiVersion: apps/v1

kind: Deployment

metadata:

name: flask-node-deployment

spec:

replicas: 1

selector:

matchLabels:

app: flasknode

template:

metadata:

labels:

app: flasknode

spec:

containers:

- name: ibm-peta

image: registry.in.bluemix.net/flask-node/app

imagePullPolicy: Always

ports:

- containerPort: 5000

**sendmail.py**

import smtplib

import sendgrid as sg

import os

from sendgrid.helpers.mail import Mail, Email, To, Content

SUBJECT = "expense tracker"

s = smtplib.SMTP('smtp.gmail.com', 587)

def sendmail(TEXT,email):

print("sorry we cant process your candidature")

s = smtplib.SMTP('smtp.gmail.com', 587)

s.starttls()

s.login("harshil.s2019@kgkite.ac.in", "bdvph9670q#@rshil123")

message = 'Subject: {}\n\n{}'.format(SUBJECT, TEXT)

s.sendmail("harshil.s2019@kgkite.ac.in", email, message)

s.quit()

def sendgridmail(user,TEXT):

from\_email = Email("mohankumar.s2019@kgkite.ac.in")

to\_email = To(user)

subject = "Sending with SendGrid is Fun"

content = Content("text/plain",TEXT)

mail = Mail(from\_email, to\_email, subject, content)

<<<<<<< HEAD

mail\_json = mail.get()

=======

mail\_json = mail.get()

>>>>>>> 38886577501b0f0a93c8cf5545721bee8df32307

response = sg.client.mail.send.post(request\_body=mail\_json)

print(response.status\_code)

print(response.headers)

* 1. **GitHub And Project Demo Link**

1. **GitHub**

https://github.com/IBM-EPBL/IBM-Project-46848-1660793142

1. **Project Demo Link**

https://user-images.githubusercontent.com/59636301/202842406-c474ebb1-46f8-4800-b69d-f1abd211a14e.mp4