Navigate the News App

**(React Application)**

|  |  |  |
| --- | --- | --- |
| S. No | TEAM MEMBERS | MAIL ID |
| 1. | Monisha R – TEAM LEADER | [monisharosekumar6718@gmail.com](mailto:monisharosekumar6718@gmail.com) |
| 2. | Vaishnavi R | [vaishnavir112005@gmail.com](mailto:vaishnavir112005@gmail.com) |
| 3. | Rohini S | [rs0969890@gmail.com](mailto:rs0969890@gmail.com) |
| 4. | Yamini K | [Kannanyamini1519@gmail.com](mailto:Kannanyamini1519@gmail.com) |

**Team members:**

# INDEX

|  |  |
| --- | --- |
| S. No | CONTENT |
| 1. | Introduction |
| 2. | Technical Architecture |
| 3. | Project goals and Objectives |
| 4. | Features of News App |
| 5. | Pre-Requisites |
| 6. | Project Structure |
| 7. | Project Flow |
| 8. | API Integration |
| 9. | Project Execution |
| 10. | Conclusion and future enhancement |

# Introduction:

InsightStream is a revolutionary web application designed to redefine how people discover and consume news. It offers an intuitive interface, dynamic search, and a vast range of news categories for all types of users. Join InsightStream to embark on an informative journey and experience the future of news consumption.

# Technical Architecture:

The user experience starts with the InsightStream web application's UI, likely built with a framework like React or Vue.js for a smooth, single-page experience. This UI interacts with an API client specifically designed for InsightStream. This client handles communication with the backend, but with a twist: it leverages Rapid API, a platform providing access to various external APIs. This suggests InsightStream might integrate external data feeds or functionalities through Rapid API, enriching the user experience without building everything from scratch.

# Project Goals and Objectives:

The primary objective of InsightStream is to establish a user-friendly platform tailored for individuals who are passionate about staying informed, exploring diverse news topics, and accessing the latest updates.

Our key goals include:

✔ **User-Friendly Experience:** Develop an interface that is intuitive and easy to navigate, ensuring users can effortlessly access, save, and share their preferred news articles.

✔ **Comprehensive News Management:** Provide robust features for organizing and managing news content, incorporating advanced search options for a personalized news experience.

✔ **Technology Stack:** Employ cutting-edge web development technologies, such as React.js, to ensure an efficient and enjoyable user interface.

# Features of News App:

✔ **News from API Sources:**

Access a vast library of global news spanning various categories and interests, ensuring a well-rounded coverage of current affairs.

✔ **Visual News Exploration:**

Discover breaking stories and explore different news categories through curated image galleries, enhancing the visual appeal of news discovery.

✔ **Intuitive Design:**

Navigate the application seamlessly with a clean, modern interface designed for optimal user experience and clarity in information presentation.

✔ **Advanced Search Feature:**

Easily access news articles on specific topics through a powerful search feature, providing users with tailored news content based on their interests.

# PRE-REQUISITES:

Here are the key prerequisites for developing a frontend application using React.js:

✔ Node.js and npm:

* Node.js is a powerful JavaScript runtime environment that allows you to run JavaScript code on the local environment.
* It provides a scalable and efficient platform for building network applications.
* Download: <https://nodejs.org/en/download/>

✔ **React.js**:

React.js is a popular JavaScript library for building user interfaces. It enables developers to create interactive and reusable UI components, making it easier to build dynamic and responsive web applications.

* Start the development server:

nmp start

This command launches the development server, and you can access your React app at [http://localhost:3000](http://localhost:3000/) in your web browser.

✔ **HTML, CSS, and JavaScript**:

* Basic knowledge of HTML for creating the structure of your app.
* CSS is used for styling.
* JavaScript for client-side interactivity is essential.

✔ **Version Control**:

Use Git for version control, enabling collaboration and tracking changes throughout the development process. Platforms like GitHub or Bitbucket can host your repository.

* + Git: Download and installation instructions can be found at: <https://git-scm.com/downloads>

✔ **Development Environment**:

Choose a code editor or Integrated Development Environment (IDE) that suits your preferences such as Visual Studio Code.

* Visual Studio Code: Download from <https://code.visualstudio.com/download>

## Install Dependencies:

* Navigate into the cloned repository directory and install libraries:

cd news-app-react

npm install

## ✔ Start the Development Server:

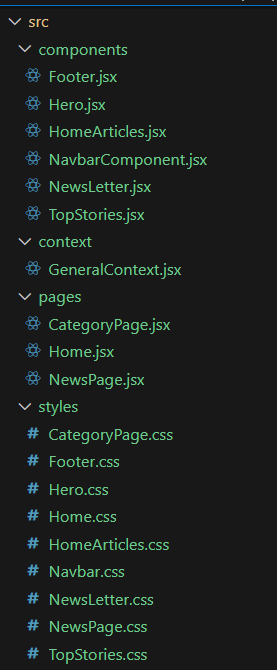
* + To start the development server, execute the following command:

npm start

## Access the App:

* + Open your web browser and navigate to [http://localhost:3000](http://localhost:3000/).
  + You should see the applications homepage, indicating that the installation and setup were successful.

# Project structure:

****

In this project, we’ve split the files into 4 major folders, *Components, Context, Pages and Styles.* In the pages folder, we store the files that acts as pages at different URLs in the application. It returns the small components in the application. The context Api will be coded in the context folder. All the styling CSS files will be stored in the styles folder.

# Project Flow:

Before starting to work on this project, let’s see the demo.

Demo link: <https://drive.google.com/file/d/1i8y09FiMk7QM0akH3my10OBWqXH-8dNh/view?usp=sharing>

**Milestone 1: Project setup and configuration.**

## Installation of required tools:

To build InsightStream, we'll need a developer's toolkit. We'll use React.js for the interactive interface, React Router Dom for seamless navigation, and Axios to fetch news data. For visual design, we'll choose either Bootstrap or Tailwind CSS for pre-built styles and icons.

Open the project folder to install necessary tools. In this project, we use:

* + React Js
  + React Router Dom
  + React Icons
  + Bootstrap/tailwind CSS
  + Axios
* For further reference, use the following resources
  + <https://react.dev/learn/installation>
  + <https://react-bootstrap-v4.netlify.app/getting-started/introduction/>
  + <https://axios-http.com/docs/intro>
  + <https://reactrouter.com/en/main/start/tutorial>

**Milestone 2: Project Development**

* Setup the Routing paths

Setup the clear routing paths to access various files in the application.



* Develop the Navbar and Hero components
* Code the popular categories components and fetch the categories from ***newsapi***.
* Also, add the trending news in the home page.
* Additionally, we can add the component to subscribe for the newsletter and the footer.
* Now, develop the category page to display various news articles under the different categories.

## Important Code snips:

**Fetching Top/Trending news**

With the API request, we fetch the trending news articles.



The code snippet shows a function written in Python called fetchTopNews that fetches news articles from an API. Here's a breakdown of the code:

*Async function fetchTopNews:*

The code defines an asynchronous function named fetchTopNews. An asynchronous function is used to handle asynchronous operations, such as making API requests that take time to complete.

***try...catch block:***

* + The try...catch block is used to handle the API request.
  + The try block contains the code that attempts to fetch data from the API using axios.get.
  + axios is an external Python library for making HTTP requests. If you don't already use Axios in your project, you'll need to install it using a package manager like pip.
  + The .get method makes a GET request to the specified URL.

*API URL:*

The URL used in the API request is

'[https://newsapi.org/v2/everything?q=popular&apiKey=37306aca596542f0a840297](https://newsapi.org/v2/everything?q=popular&apiKey=37306aca596542f0a8402978de3d4224) [8de3d4224](https://newsapi.org/v2/everything?q=popular&apiKey=37306aca596542f0a8402978de3d4224)'.

This is likely a specific API endpoint that returns popular news articles. You might need to replace this URL with the actual endpoint you want to use depending on the API you're using. Replace '37306aca596542f0a8402978de3d4224' with a placeholder instructing users to replace it with their own API key.

***Error Handling (catch block):***

The catch block handles any errors that might occur during the API request. If there's an error, it's logged to the console using console.error(error).

***Setting State (then block not shown):***

The .then method (not shown in the code snippet) is likely used to process the fetched data after a successful API request.

In this case, it likely updates a state variable named topNews (based on the function name fetchTopNews) with the fetched news articles. This state variable might be used to display the news articles in a user interface.

## Fetching news by search/category

With the specific category or search keyword, we use API request to fetch all the news articles related to that.



The code snippet shows a function called get\_news\_articles that fetches news articles from a news API. Here's a breakdown of the code:

***Imports:***

The code starts by importing the requests library. The requests library is a popular Python library for making HTTP requests. If you don't already have it installed in your project, you can install it using pip install requests.

***API Key:***

The line API\_KEY = 'YOUR\_API\_KEY' defines a variable named API\_KEY and assigns it a placeholder value 'YOUR\_API\_KEY'. You should replace this with a placeholder instructing users to replace it with their own API key obtained from the news API provider they want to use.

***Function Definition (get\_news\_articles):***

The code defines a function named get\_news\_articles that takes two parameters:

* + query: This parameter is likely a string representing the search query for news articles.
  + source: This parameter is likely a string representing the news source (e.g., 'bbc-news', 'cnn').

***Building the API Request URL:***

The URL includes the following parts:

* + - Base URL: <https://newsapi.org/v2/everything>
    - Query parameters:
    - q: This parameter is set to the query argument passed to the function.
    - apiKey: This parameter is set to the API\_KEY variable, which should contain the user's API key.

***Making the API Request (requests.get):***

The line response = requests.get(url) sends a GET request to the API URL constructed earlier. The requests.get function from the requests library is used to make the HTTP request. The response from the API is stored in the response variable.

***Error Handling (try...except block):***

* + The try...except block is used to handle potential errors during the API request.
  + The try block contains the code that attempts to fetch data from the API using requests.get(url).
  + The except block handles any exceptions that might occur during the request, such as network errors or invalid API responses. In this case, it prints an error message to the console using print(f'Error fetching news articles: {e}')

## API Integration:

### **Fetching Trending News**

axios.get(`https://newsapi.org/v2/everything?q=popular&apiKey=YOUR\_API\_KEY`);

### **Fetching Category-based News**

axios.get(`https://newsapi.org/v2/ axios.get(`https://newsapi.org/v2/everything?q={category}&apiKey=YOUR\_API\_KEY`);

everything?q={category}&apiKey=YOUR\_API\_KEY`); axios.get(`https://newsapi.org/v2/everything?q={category}&apiKey=YOUR\_API\_KEY`);

### **Fetching News by Search Query**

axios.get(`https://newsapi.org/v2/everything?q={searchTerm}&apiKey=YOUR\_API\_KEY`);

# Project Execution:

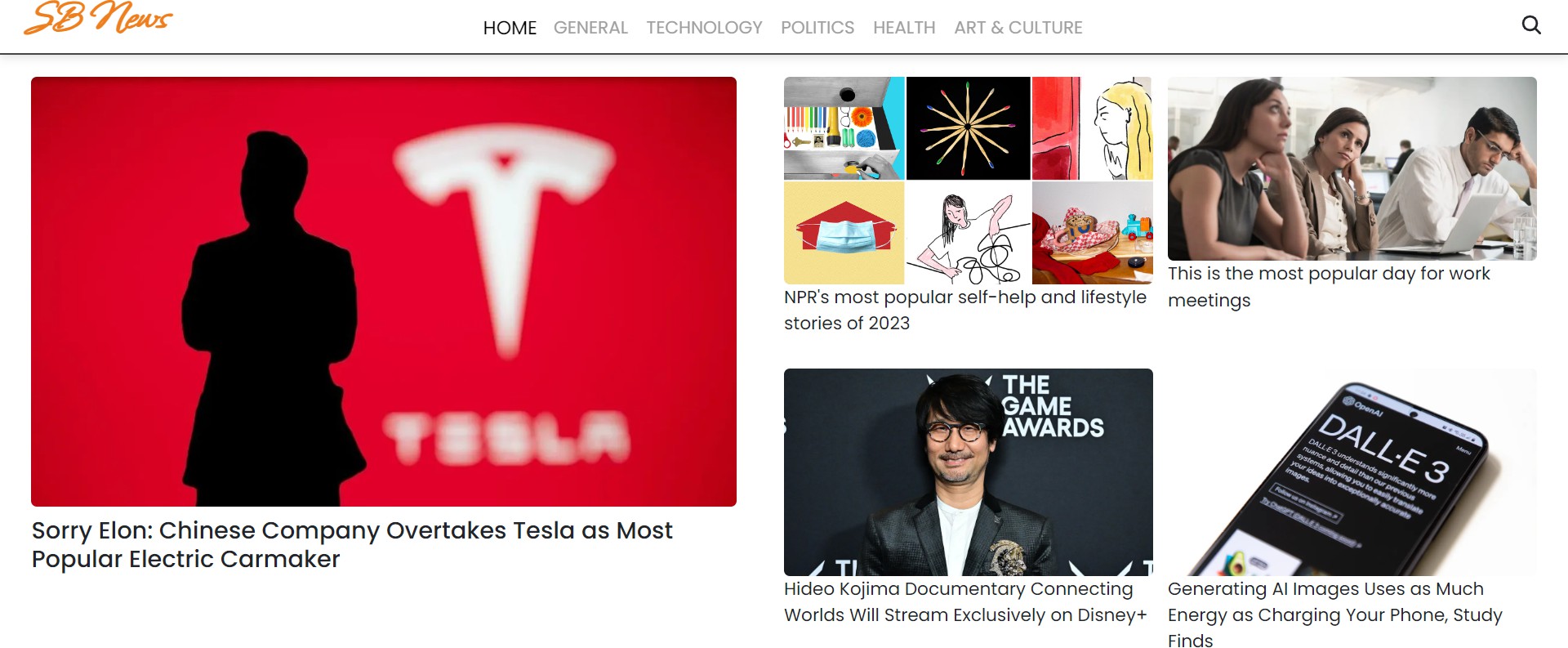
After completing the code, run the react application by using the command “npm start” or “npm run dev” if you are using vite.js

Here are some of the screenshots of the application.

## Hero components

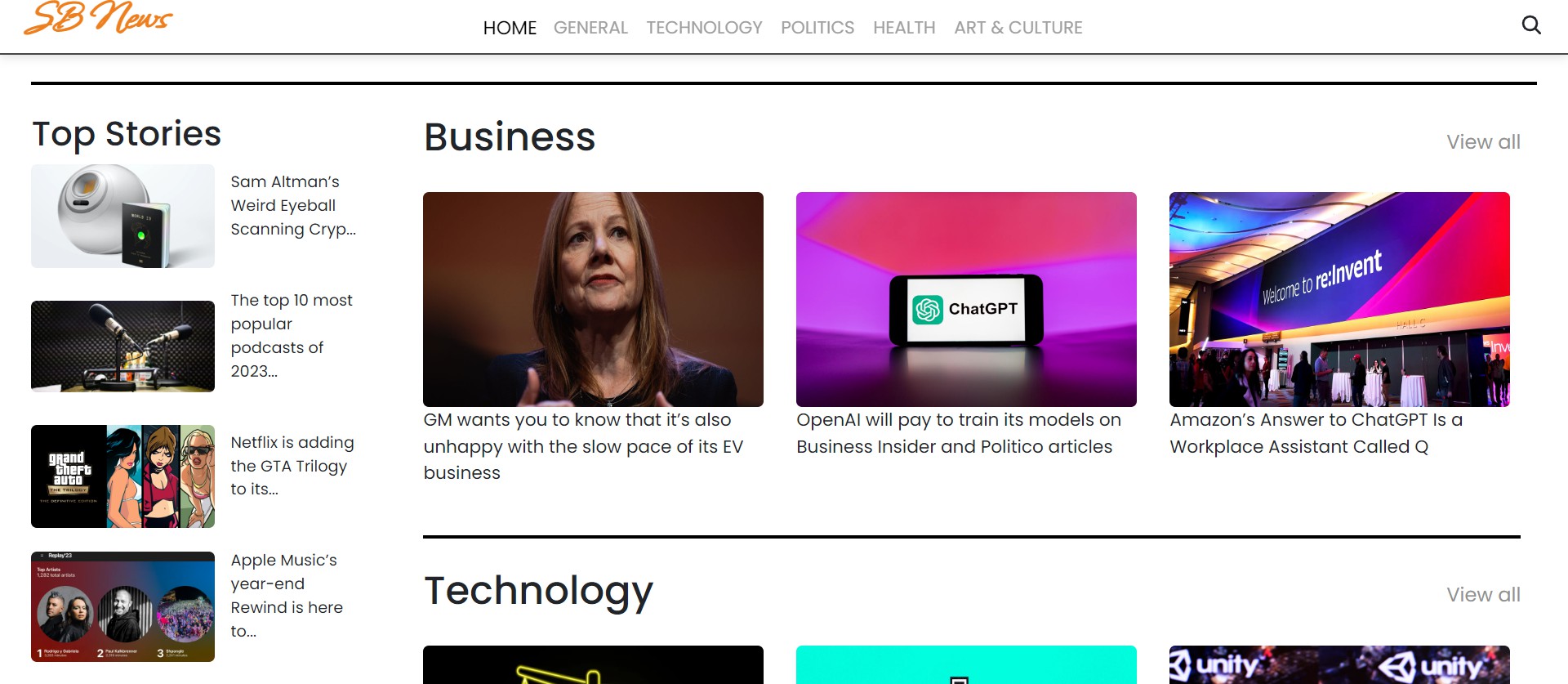
In the hero component, the trending news articles are displayed. It is to highlight them.

Apart from that, the search bar is also available to search for various articles and categories.



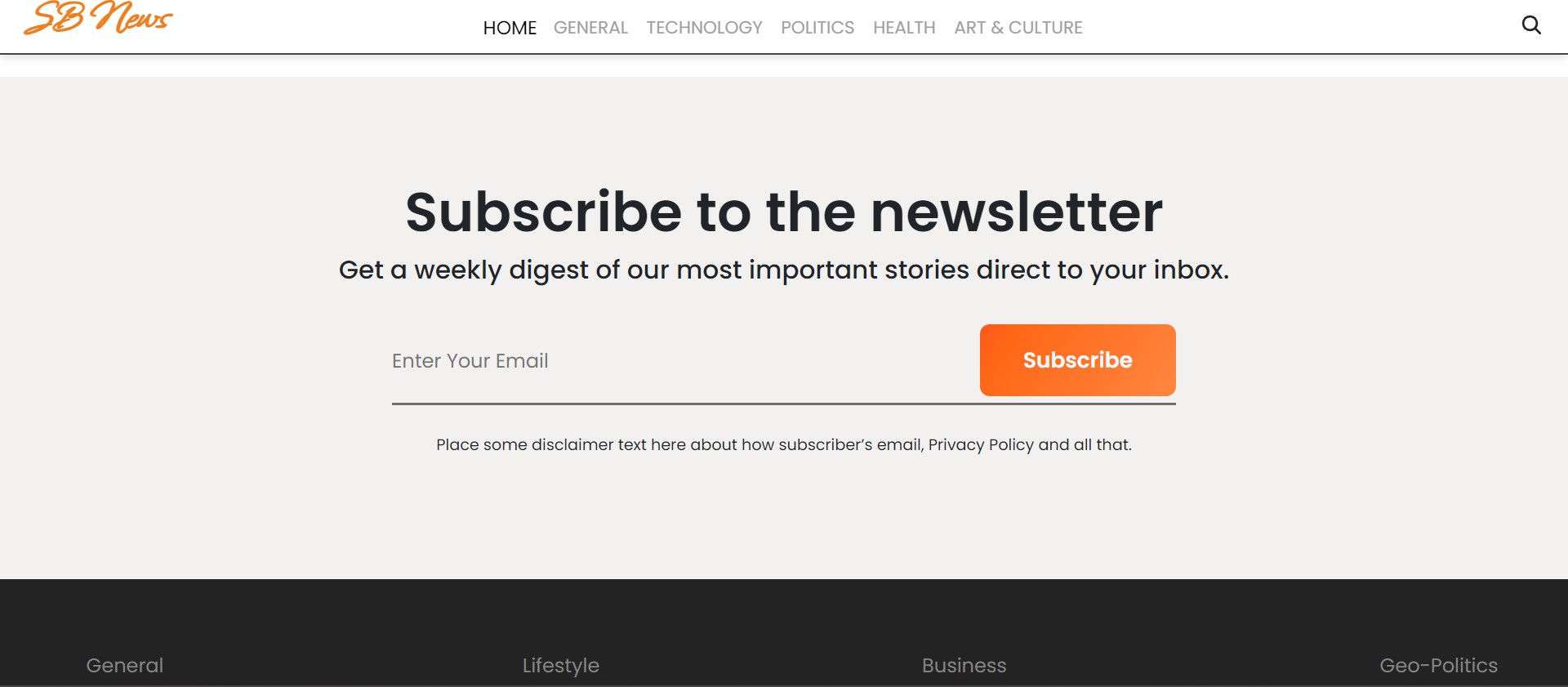
## Popular categories

In the hero component, the trending news articles are displayed. It is to highlight them. Apart from that, the search bar is also available to search for various articles and categories.



* **Newsletter**

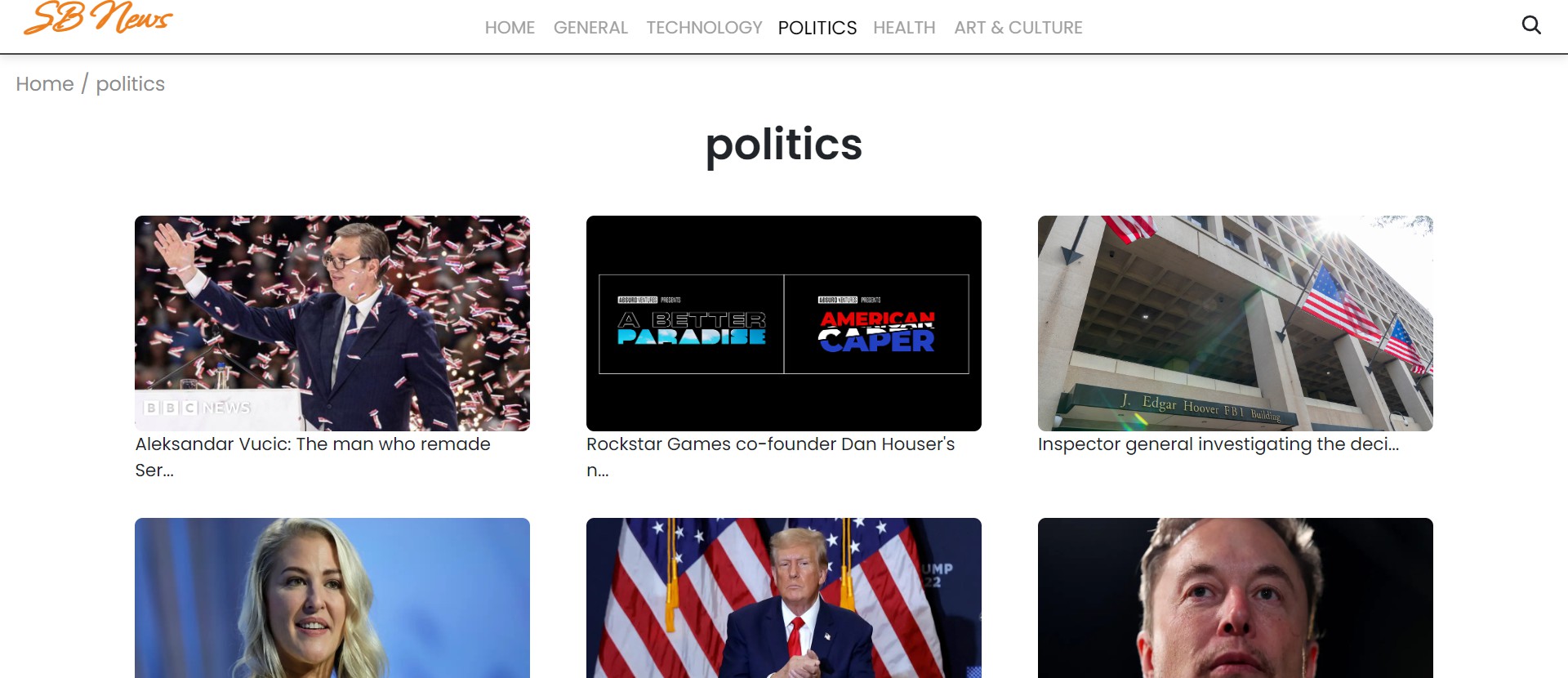
Staying informed is key! This section would act as a magnet for users who want to stay up-to-date on the latest news. A brief signup form with an email field would be presented, along with a clear call to action button like "Subscribe Now" or "Get Daily News Updates." With a simple click, users can join the InsightStream community and receive curated news delivered straight to their inbox



* **Category/Search result page**

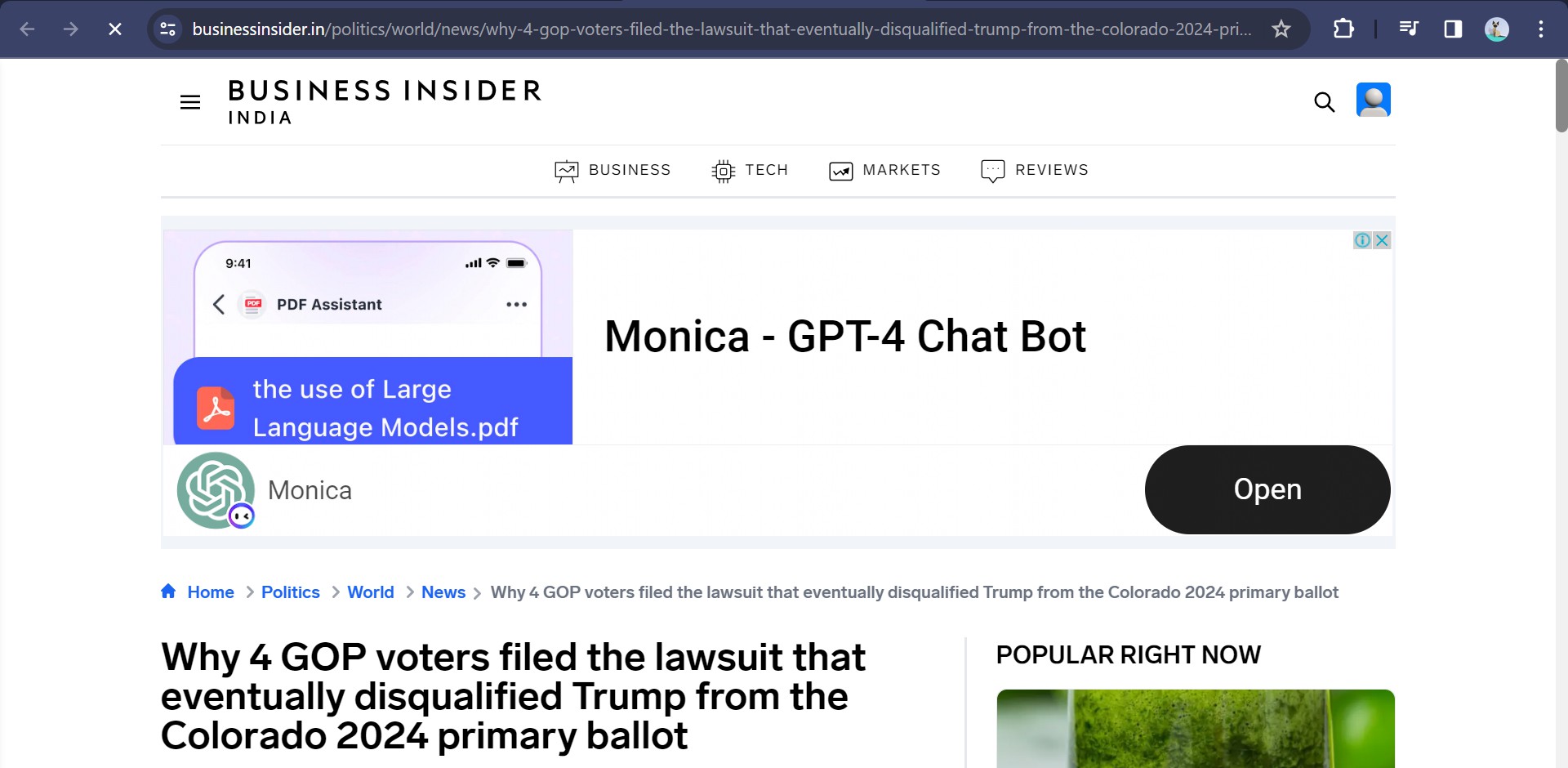
Finding the news you crave is effortless with InsightStream. This page displays a neatly organized list of articles matching your chosen category or specific search query. Each entry would provide a clear headline, a concise summary, and if available, an image to give you a quick glimpse into the story. To further refine your exploration, filters or sorting options might be available. Imagine narrowing down results by date, source, keyword, or other relevant criteria to pinpoint exactly

what you're looking for.



## Redirected Article page

This is where you dive deep! The article page proudly displays the complete news story, retrieved directly from the original source. To keep you engaged and exploring related topics, the page might also suggest additional articles based on the current story. These suggestions can open doors to a world of interconnected information, allowing you to become a well-rounded news connoisseur.



**Project Demo link:**

<https://drive.google.com/file/d/1i8y09FiMk7QM0akH3my10OBWqXH-8dNh/view?usp=sharing>

## Conclusion:

InsightStream is a **comprehensive and modern** news application that provides users with an **intuitive, efficient, and engaging** way to stay updated on current events.

With its **real-time updates, seamless navigation, and advanced search features**, the application ensures that users have access to **reliable and well-organized** news articles at their fingertips.

The project is designed to be scalable and adaptable, making it an excellent choice for future expansions and improvements.

This news application not only offers an aesthetically pleasing and user-friendly experience but also ensures **high performance, accessibility, and efficiency** through the use of **cutting-edge web technologies**.

Its well-structured UI, API integration, and feature-rich environment make it a **powerful platform** for news consumption.

### **Future Enhancements:**

✔ **User Authentication & Profiles** – Implementing user logins for personalized experiences.

✔ **AI-based News Recommendations** – Providing smart suggestions based on user reading patterns.

✔ **Dark Mode** – Enhancing user accessibility and experience with a night-friendly theme.

✔ **Multi-language Support** – Expanding the application’s reach to a global audience.

✔ **Offline Mode** – Allowing users to save articles for reading without an internet connection.

✔ **Push Notifications** – Alerting users with breaking news and category-based updates.

✔ **Community Engagement Features** – Enabling comments, likes, and discussions on news articles.

✔ **Integration with Social Media** – Allowing easy sharing of articles on platforms like Twitter and Facebook.

These enhancements will further elevate InsightStream’s functionality, making it a **cutting-edge, user-centric news platform** that adapts to modern needs and expectations.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

*Thank You!!!*