InsightStream: **News App** (React Application)

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.

In today's digital age, staying informed is crucial. With numerous news sources available, it's hard to find relevant info. Our team proposes developing a News App to address this issue. A web-based application that provides personalized news feeds from various sources, aggregated in one platform. Users can customize their feed based on interests and location. Relevant articles are recommended using modern algorithms. Improving user engagement and retention is the goal,

The News App utilizes modern web development technologies like Node.js, React.js, and RESTful APIs to build the app. A scalable, efficient, and user-centric news aggregator is created, providing a personalized news experience. Users can access news from multiple sources in one place, with features like breaking news notifications and social sharing options. The app is designed to be user-friendly and easy to navigate. Making it easy for users to stay informed about current events. The app's goal is to provide a one-stop solution for news consumption. Enhancing the overall user experience is the ultimate goal.

The News App aims To revolutionize the way we consume news by providing a platform that. Aggregates news from multiple sources. Users can stay informed about current events without searching multiple sites. The app's personalized recommendations engine suggests relevant articles based on user interests, location, and reading history. The app's goal is to provide a seamless and intuitive news consumption experience. By achieving this goal, the News App aims to become the go to news source for users.

Scenario Based Intro:

Suppose you're rushing home after work, phone clutched in your hand. Today's been a whirlwind, and you have no idea what's happening in the world. Suddenly, you remember InsightStream , the innovative app you downloaded that promised to revolutionize your news experience. Intrigued, you open the app. Images flash across the screen – breaking headlines, in-depth articles, diverse categories. This isn't your typical news feed. InsightStream feels...different. Intrigued, you tap a category and dive in, ready to explore the future of staying informed.

Imagine stepping off the subway after a long day at work, your phone buzzing with notifications. You want to catch up on the latest headlines, but scrolling through multiple apps and websites feels overwhelming. That's when you remember InsightStream, the intuitive news app you recently installed. With just a tap, you're welcomed by a clean, visually engaging interface filled with breaking news, trending topics, and personalized recommendations based on your interests. Instead of sifting through cluttered feeds, the news you care about is presented seamlessly in one place.

As you browse, InsightStream's smart algorithms work in the background, curating a feed that aligns with your reading habits and location. Whether it's global affairs, technology, sports, or entertainment, every article is carefully aggregated from trusted sources, ensuring accuracy and relevance. You swipe through stories effortlessly, saving articles for later and sharing insightful pieces with friends. A real-time notification alerts you to a major event unfolding—you're always in the know without the hassle of constant searching.

Just before heading home, you check out the "Trending Now" section and dive into an exclusive analysis on a topic that's been making headlines all day. The app's smooth navigation, engaging visuals, and advanced search features make it more than just a news aggregator—it's a personalized, intelligent companion for staying informed. With InsightStream, news consumption becomes an effortless, enriching experience, ensuring you're always connected to the world in a way that suits your lifestyle.
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 InsightStream:

✓ 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.

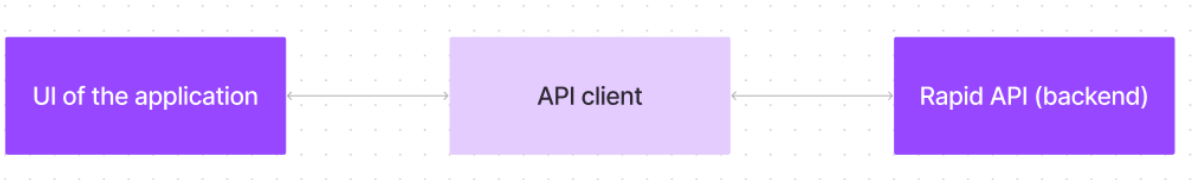
✓ Intuitive Design:

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.

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.

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.

Install Node.js and npm on your development machine, as they are required to run JavaScript on the server-side.

- Download: https://nodejs.org/en/download/
- Installation instructions: https://nodejs.org/en/download/package-manager/

✔ **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.

Install React.js, a JavaScript library for building user interfaces.

- Create a new React app:

```
npx create-react-app my-react-app
```

Replace my-react-app with your preferred project name.

- Navigate to the project directory:

```
cd my-react-app
```

- Running the React App:

  With the React app created, you can now start the development server and see your React application in action.

- Start the development server:

  ```
  npm start
  ```

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

✔ **HTML, CSS, and JavaScript**: Basic knowledge of HTML for creating the structure of your app, CSS for styling, and 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, Sublime Text, or WebStorm.

- Visual Studio Code:
  Download from https://code.visualstudio.com/download
- Sublime Text:
  Download from https://www.sublimetext.com/download
- WebStorm:
  Download from https://www.jetbrains.com/webstorm/download

To install and run the Application project from google drive:

Follow below steps:

✔ **Get the code:**

- Download the code from the drive link given below:

https://drive.google.com/drive/folders/1tDoSwd-1I3HsPJ9_92MnZTUtteeda-hL?usp=sharing

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:

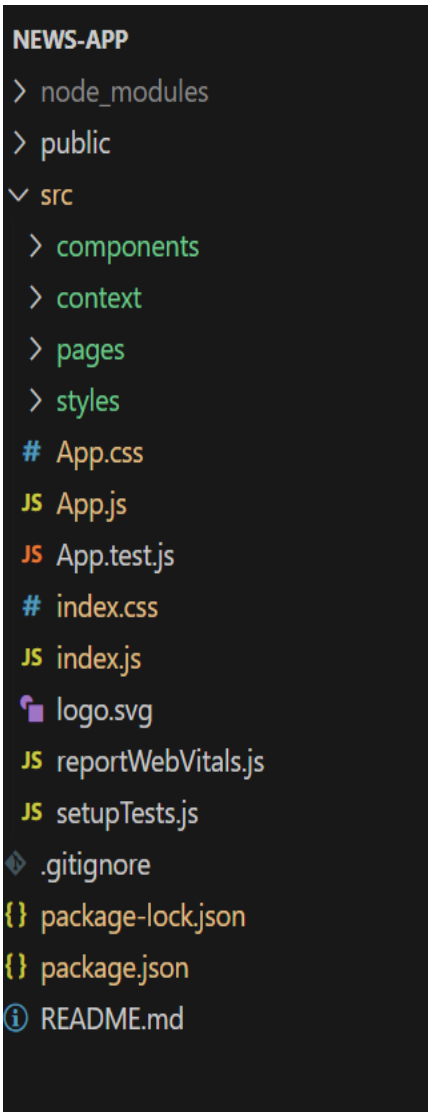
```
npm start
```

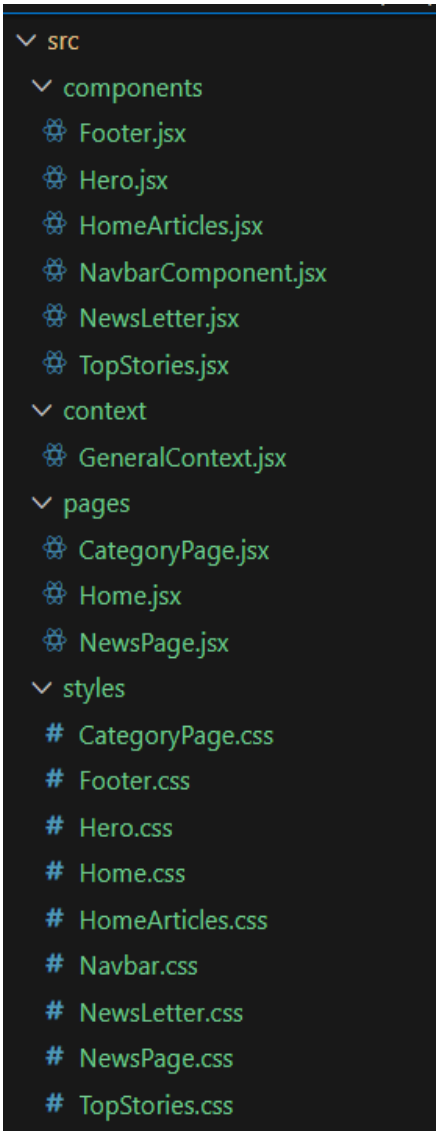
Access the App:

- Open your web browser and navigate to http://localhost:3000.
- You should see the applications homepage, indicating that the installation and setup were successful.

You have successfully installed and set up the application on your local machine. You can now proceed with further customization, development, and testing as needed.

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. The components folder stores all the files, that 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:

Project demo:

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

Use the code in:

https://drive.google.com/drive/folders/1tDoSwd-1I3HsPJ9_92MnZTUtteeda-hL?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.

```jsx
<div className="App">
    <NavbarComponent />
    <Routes>
      <Route exact path="/" element={<Home/>} />
      <Route exact path="/category/:id" element={<CategoryPage/>} />
      <Route exact path="/news/:id" element={<NewsPage/>} />
    </Routes>
    <Footer />
</div>
```

- 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.

```jsx
const fetchTopNews = async () => {
  try {
    const response = await axios.get("https://newsapi.org/v2/everything?q=popular&apiKey=37306aca596542f0a8402978de3d4224");
    setTopNews(response.data.articles);
  } catch (error) {
    console.error(error);
  }
}
```

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=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.

```
const [categoryNews, setCategoryNews] = useState([]);

const {id} = useParams();
  useEffect(() => {
    if (id){
      fetchNews(id)
    }
    }, [window.location.pathname])

    const fetchNews = async (id) => {
      try {
        const response = await axios.get(`https://newsapi.org/v2/everything?q=${id}&apiKey=37306aca596542f0a8402978de3d4224`);
        setCategoryNews(response.data.articles);
      } catch (error) {
        console.error(error);
      }
    }
```

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 line url = f'https://newsapi.org/v2/everything?q={query}&apiKey={API_KEY}' constructs the URL for the API request using a formatted string literal (f-string).

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):

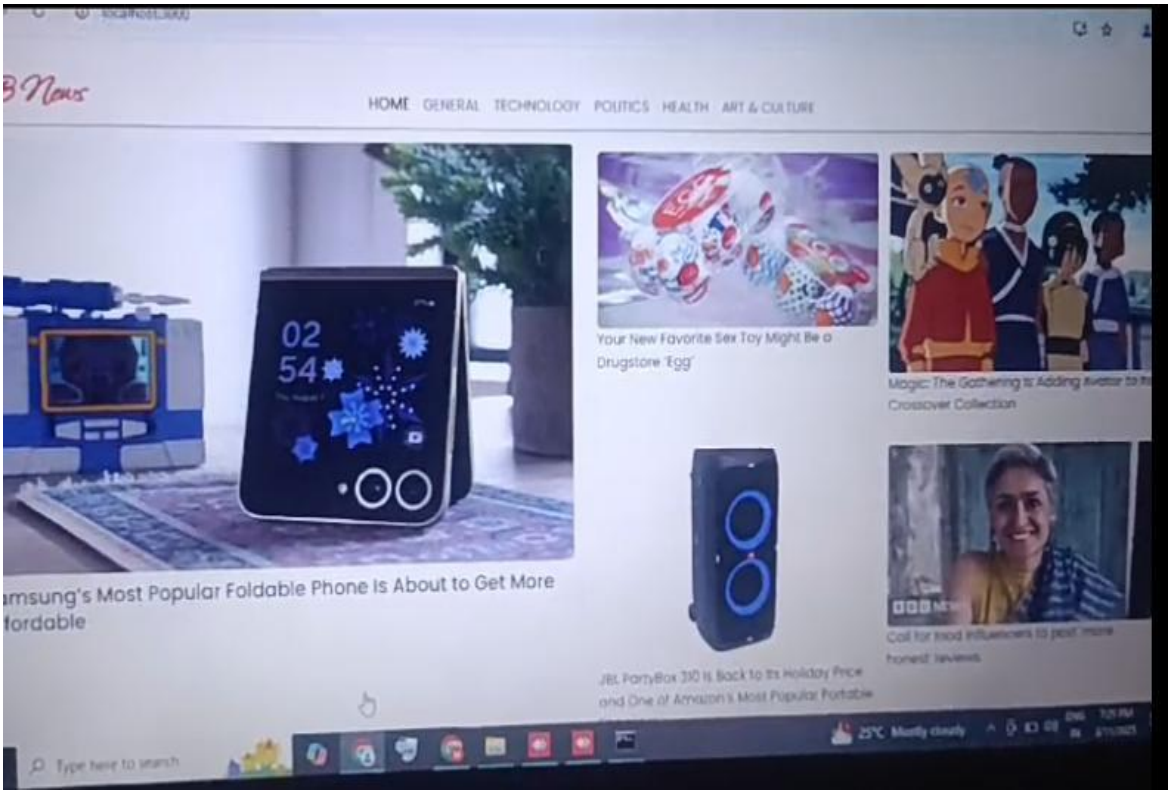
- The try...except block is used to handle potential errors during the API request.
- The try block contains the code that attempts to fetch data from the API using requests.get(url).
- The except block handles any exceptions that might occur during the request, such as network errors or invalid API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

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

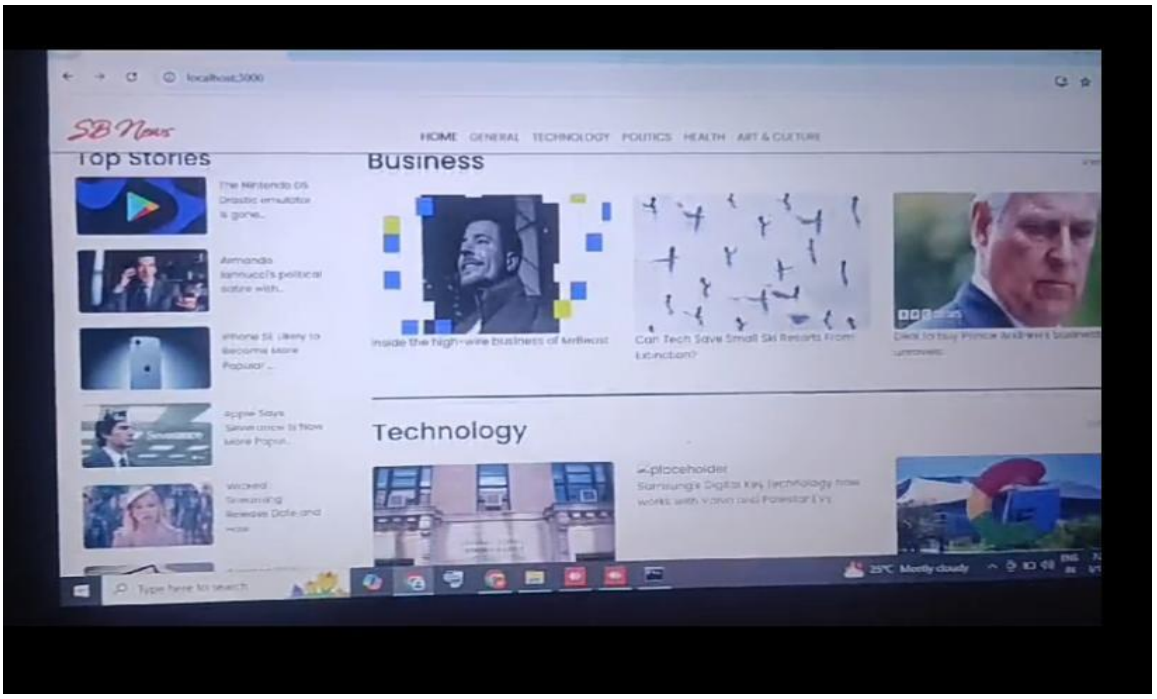
Here are some of the screenshots of the application.

? **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.


HOME GENERAL TECHNOLOGY POLITICS HEALTH ART & CULTURE
02
54
msung's Most Popular Foldable Phone Is About to Get More
fordable
Your New Favorite Sex Toy Might Be a
Drugstore 'Egg'
Crossover Collection
Call for food influencers to post more
honest reviews
JBL PartyBox 310 Is Back to Its Holiday Price
and One of Amazon's Most Popular Portable
Type here to search
25°C Mostly cloudy

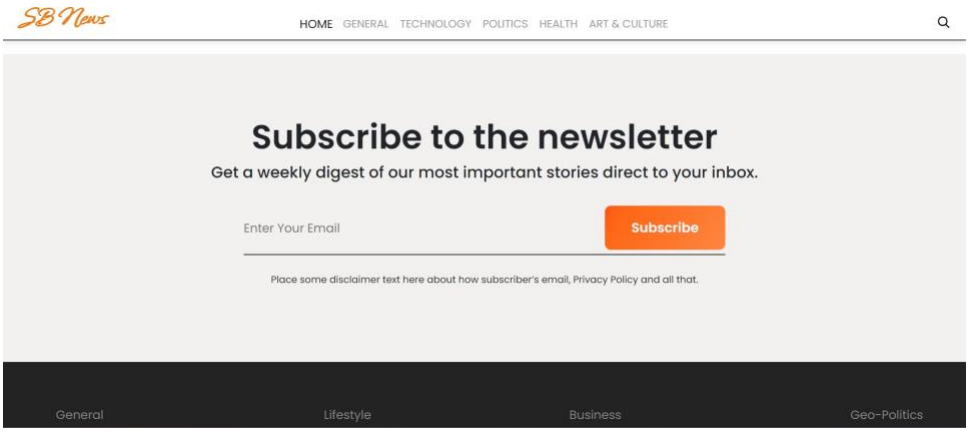
? **Popular categories**

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.

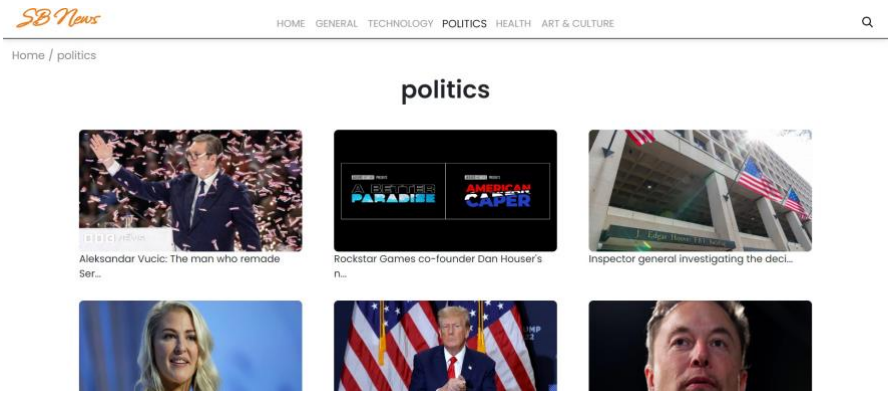


? **Category/Search result page**

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

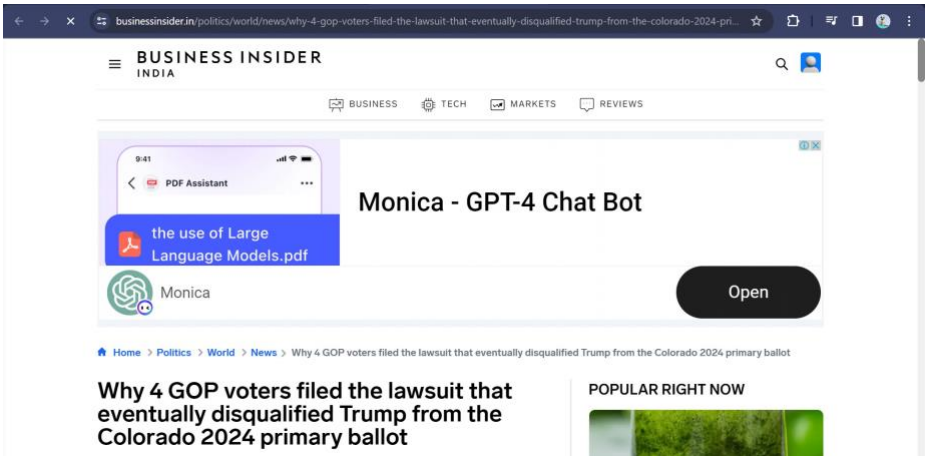
available, an image to give you a quick glimpse into the story. To further refine your exploration, filters or sorting options might be available. Imagine narrowing

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/drive/folders/1rP6vWfK90czFxst_Y4Y1E3y-U_-q16eR