

# Insight-stream: Navigate the News Landscape

Insight-stream 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 Insight-stream to embark on an informative journey and experience the future of news consumption.

## Description:

- Welcome to the cutting-edge frontier of news exploration with Insight-stream! Our revolutionary web application is meticulously crafted to transcend the boundaries of traditional news consumption, catering to the diverse interests of both avid news enthusiasts and seasoned information professionals. With an emphasis on an intuitive user interface and a robust feature set, Insight-stream is poised to redefine the entire news discovery and consumption process.
- Designed with a commitment to user-friendly aesthetics, Insight-stream immerses users in an unparalleled journalistic adventure. Navigate seamlessly through a vast expanse of news categories with features such as dynamic search, effortlessly bringing you the latest and most relevant stories from around the world.
- From those seeking the latest headlines to seasoned news connoisseurs, Insight-stream embraces a diverse audience, fostering a dynamic community united by a shared passion for staying informed. Our vision is to reshape how users interact with news, presenting a platform that not only delivers breaking stories but also encourages collaboration and sharing within the vibrant news community.
- Embark on this informative journey with us, where innovation seamlessly intertwines with journalistic

tradition. Every click within Insight-stream propels you closer to a realm of global happenings and perspectives. Join us and experience the evolution of news consumption, where each feature is meticulously crafted to offer a glimpse into the future of staying informed.

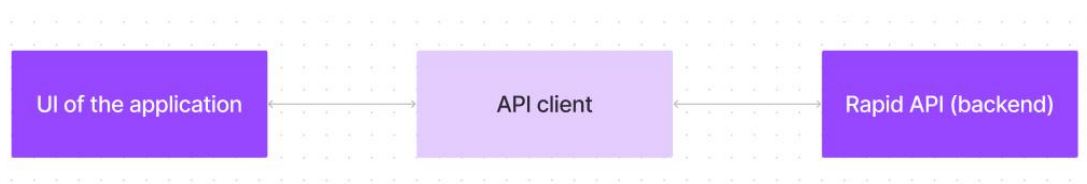
- Elevate your news exploration with Insight-stream, where every headline becomes a gateway to a world of information waiting to be discovered and understood.

Trust Insight-stream to be your reliable companion on the journey of staying connected with the pulse of the world.

### 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 Insight-stream, 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. Insight-stream feels...different. Intrigued, you tap a category and dive in, ready to explore the future of staying informed.

### Technical Architecture:



The user experience starts with the Insight-stream 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 Insight-stream. 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 Insight-stream might integrate external data feeds or functionalities through Rapid API, enriching the user experience without building everything from scratch.

## Project Objective

The primary objective of Insight-stream 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 Insight-stream:

- **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 front-end 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/>

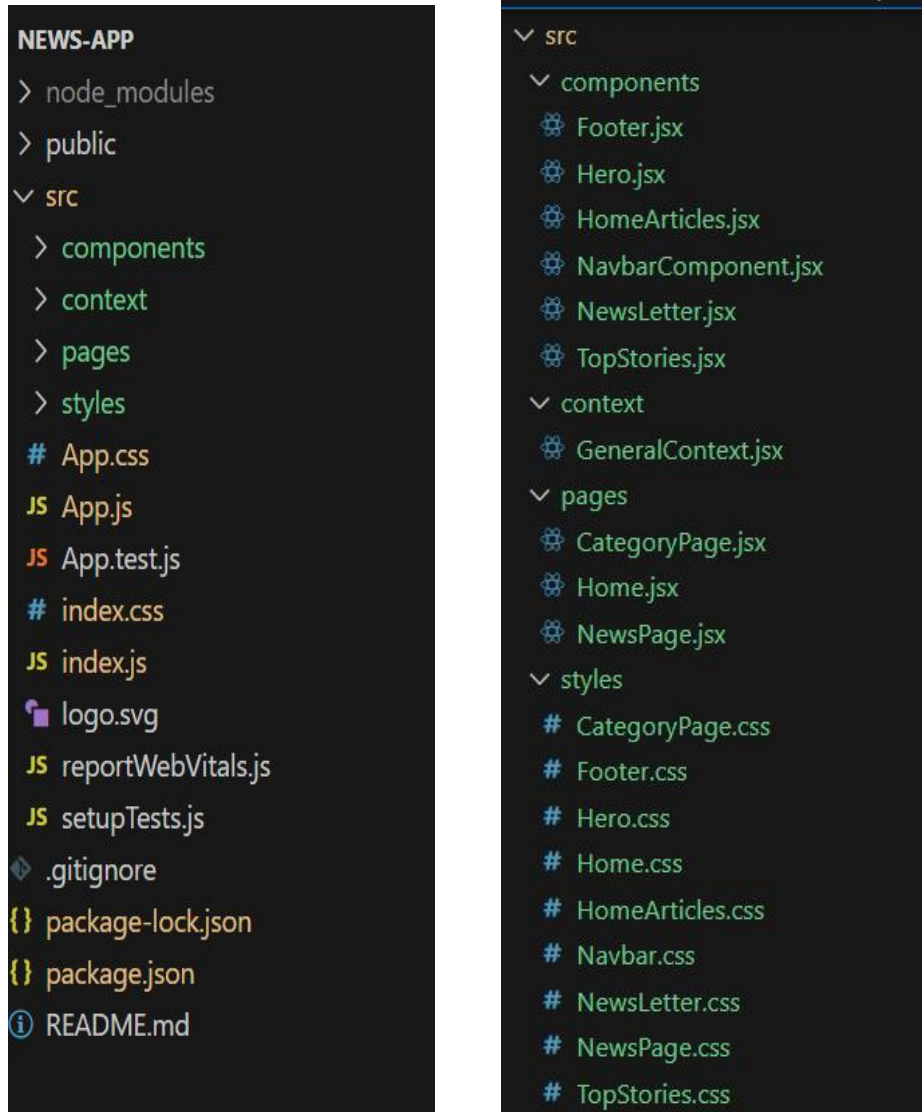
### **To clone and run the Application project from GitHub:**

Follow below steps

- **Install Dependencies:** Navigate into the cloned repository directory and install libraries: `cd news, 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 have successfully installed and set up the application on your local machine.

# 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 Development

- Project Development
- Setup the Routing paths
- Setup the clear routing paths to access various files in the application.

## Example:

```
<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

```
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);
  }
}
```

- Fetching news by search/category

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

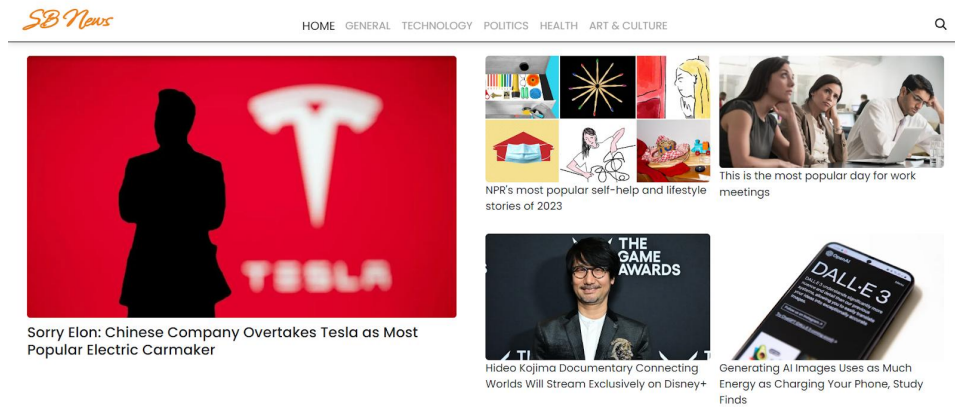
const {id} = useParams();
useEffect(() => {
  if (id){
    fetchNews(id)
  }
}, [window.location.pathname])

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);
  }
}
```

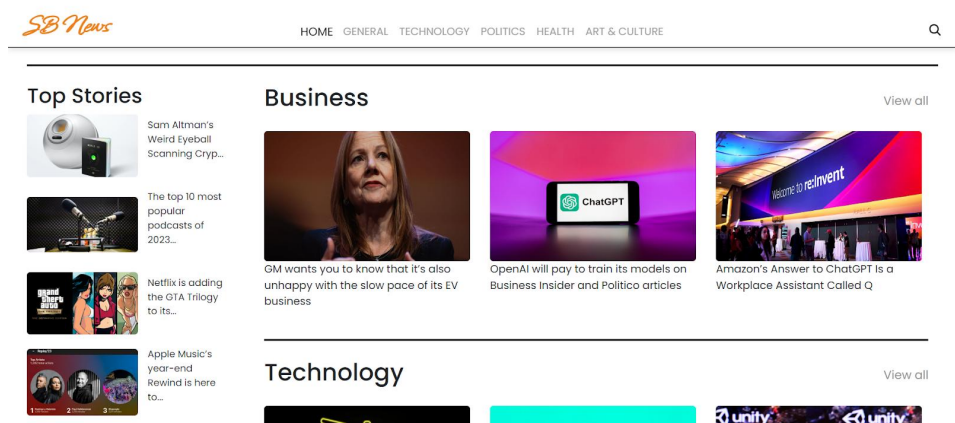


# Project Implementation & Execution

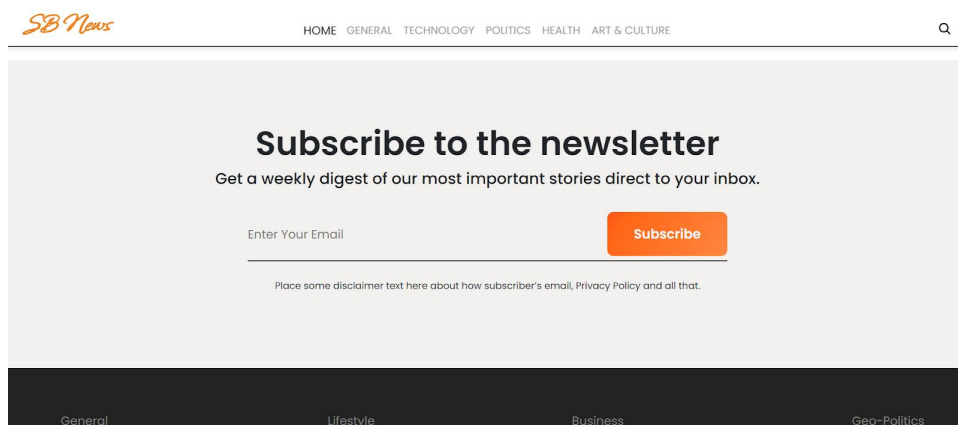
- Hero components



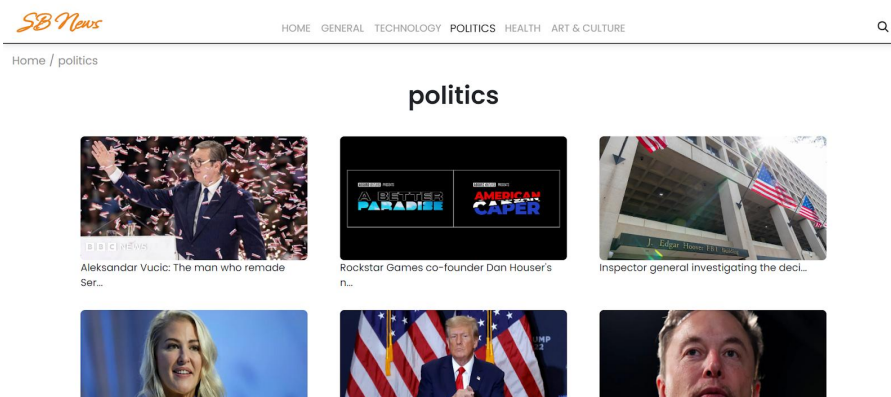
- Popular categories



- Newsletter



- Category/Search result page



- Redirected page

