SB News... (React Application)

Introduction:

🌐 Welcome to the cutting-edge frontier of news exploration with SB News! 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, SB News is poised to redefine the entire news discovery and consumption process.

🗞️Designed with a commitment to user-friendly aesthetics, SB News 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, SB News 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 SB News 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 SB News, where every headline becomes a gateway to a world of information waiting to be discovered and understood. Trust SB News to be your reliable companion on the journey of staying connected with the pulse of the world. 🌐

Project Goals and Objectives:

The primary objective of SB News 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 SB News:

✓ **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.

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 clone and run the Application project from GitHub:

Follow below steps:

✓ **Get the code:**

• Download the code from the drive link given below:

Install Dependencies:

• Navigate into the cloned repository directory and install libraries:

```
cd news-app-react
npm install
```

✓ **Start the Development Server**:

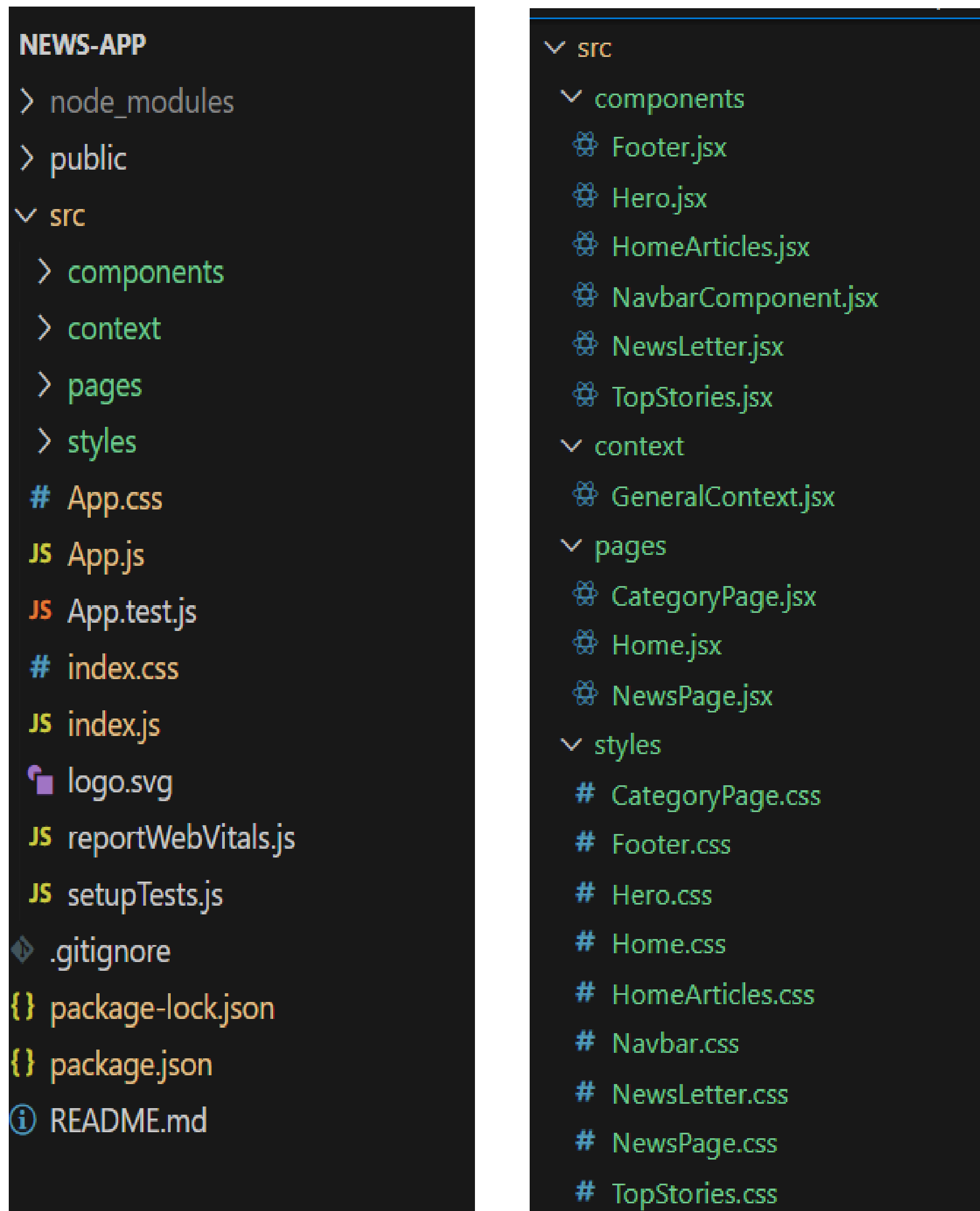• To start the development server, execute the following command:

```
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:

```
NEWS-APP
> node_modules
> public
∨ src
  > components
  > context
  > pages
  > styles
  # App.css
  JS App.js
  JS App.test.js
  # index.css
  JS index.js
  🔖 logo.svg
  JS reportWebVitals.js
  JS setupTests.js
  ◈ .gitignore
  {} package-lock.json
  {} package.json
  ⓘ README.md
```

```
∨ src
  ∨ components
    ⚛ Footer.jsx
    ⚛ Hero.jsx
    ⚛ HomeArticles.jsx
    ⚛ NavbarComponent.jsx
    ⚛ NewsLetter.jsx
    ⚛ TopStories.jsx
  ∨ context
    ⚛ GeneralContext.jsx
  ∨ pages
    ⚛ CategoryPage.jsx
    ⚛ Home.jsx
    ⚛ NewsPage.jsx
  ∨ styles
    # CategoryPage.css
    # Footer.css
    # Hero.css
    # Home.css
    # HomeArticles.css
    # Navbar.css
    # NewsLetter.css
    # NewsPage.css
    # TopStories.css
```

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/11ndRXhrfk3zqlY-dtOfi35ary_xzgI7c/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**:

    1. 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.

    Ex:

```
<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);
    }
  }
```
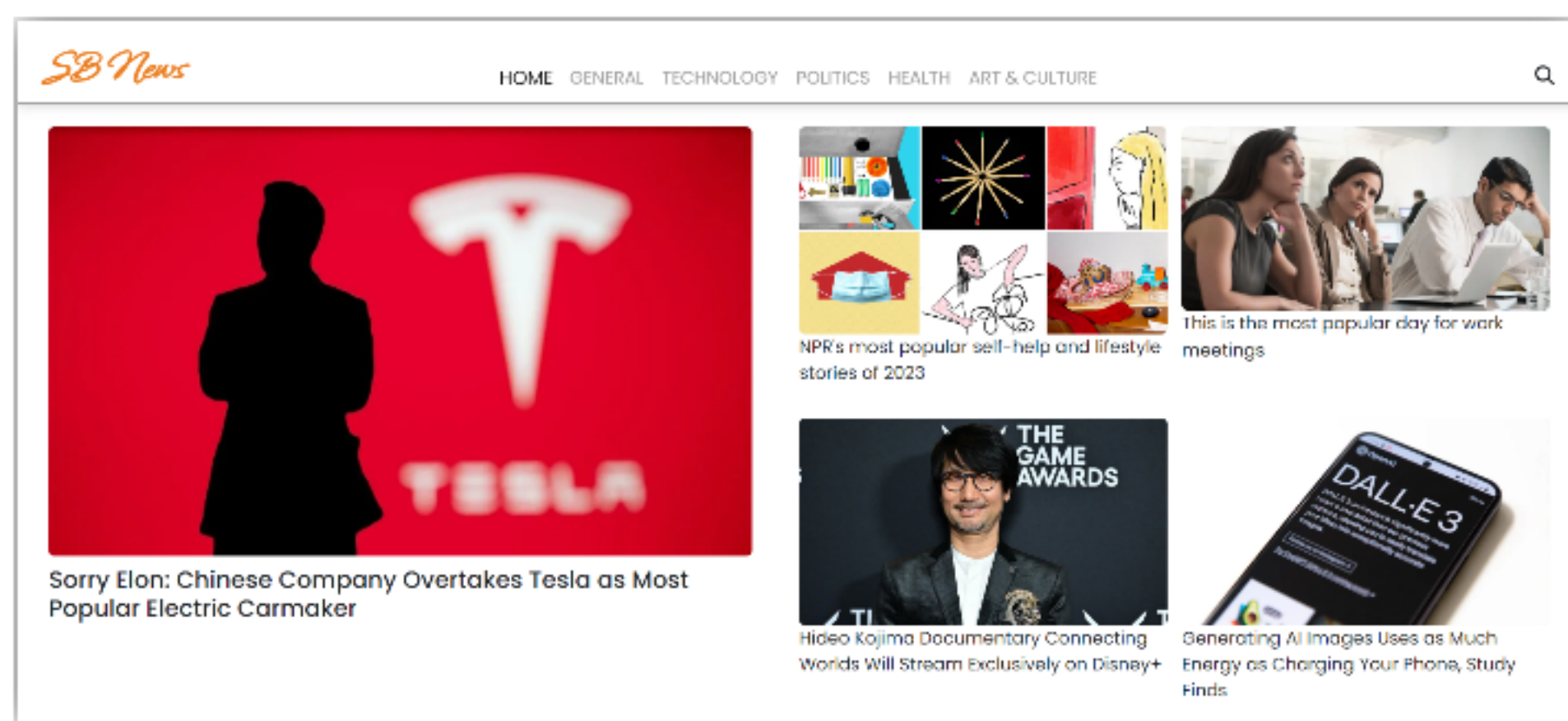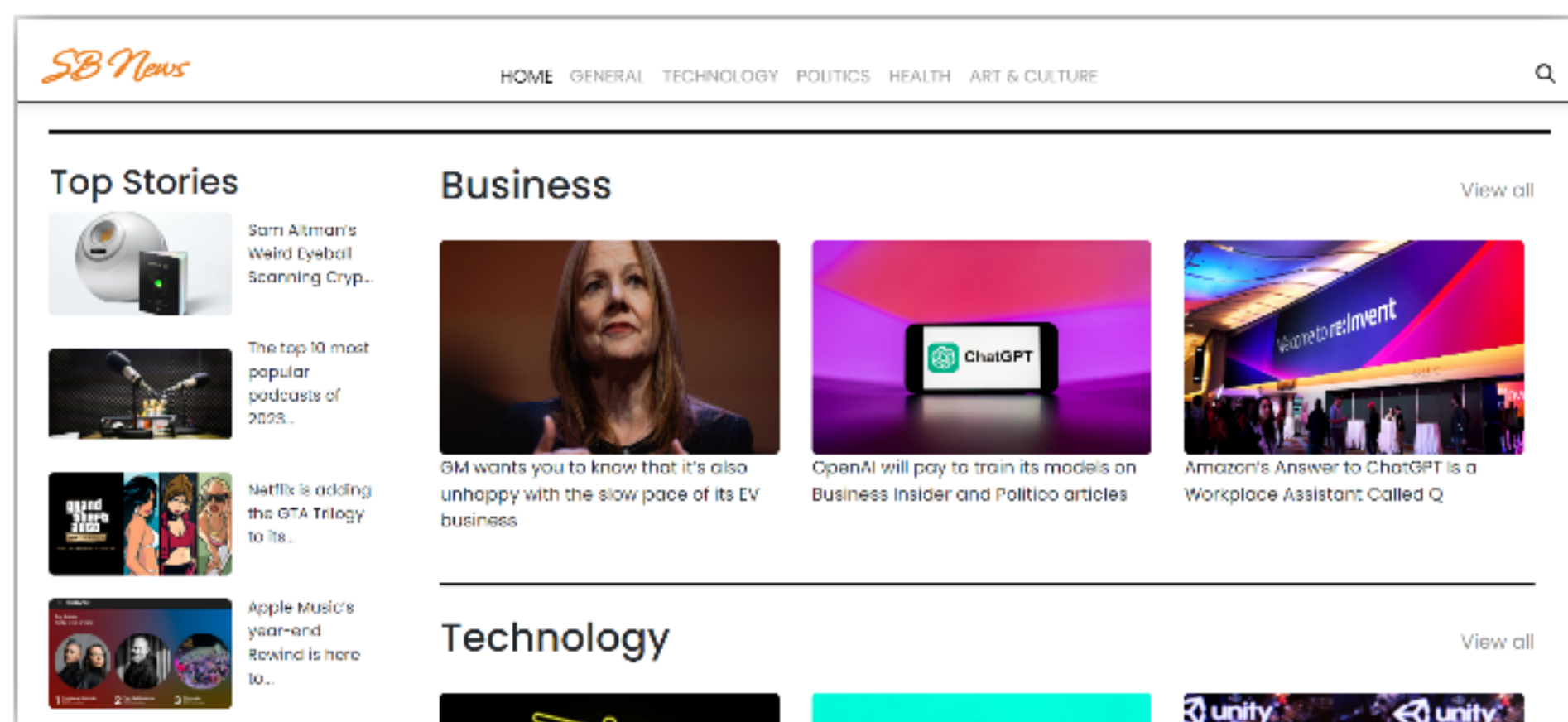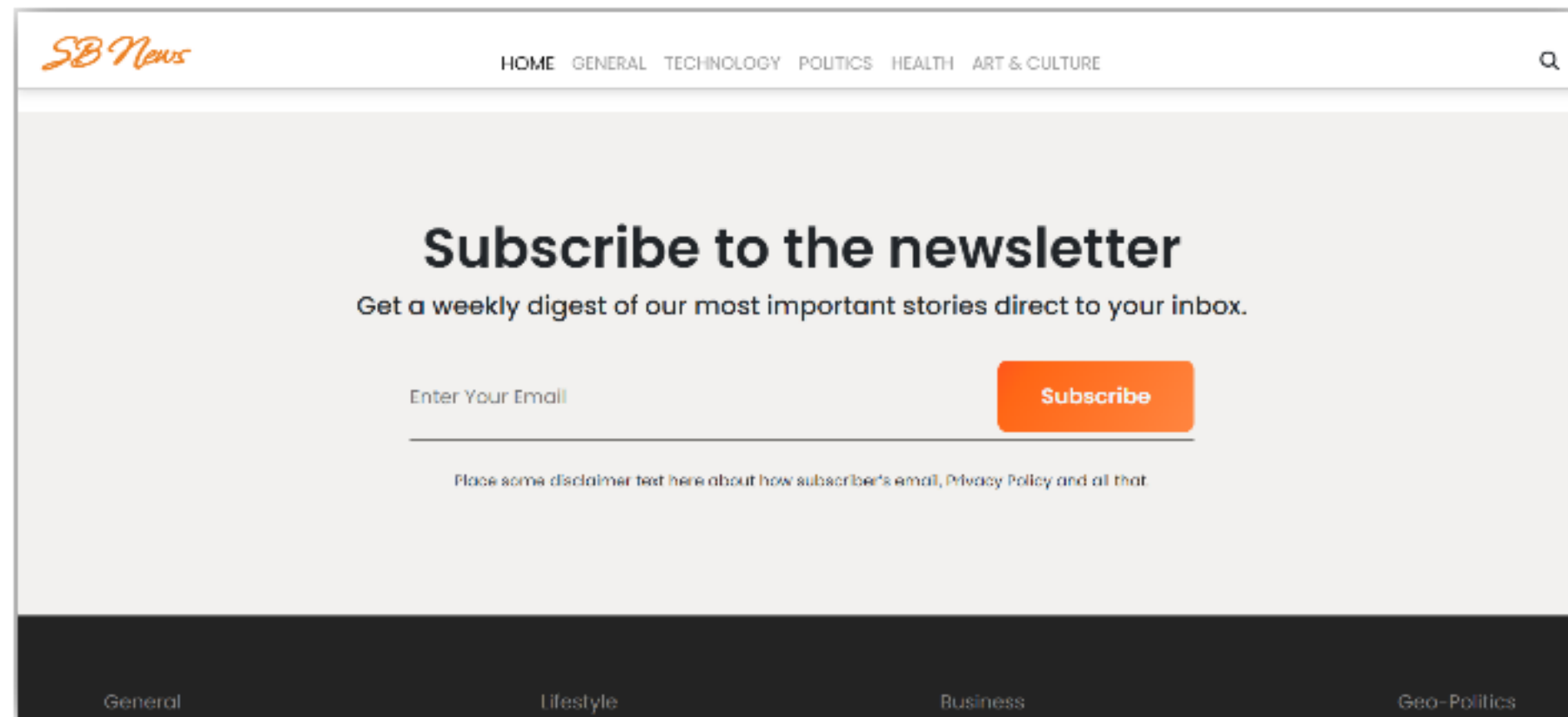
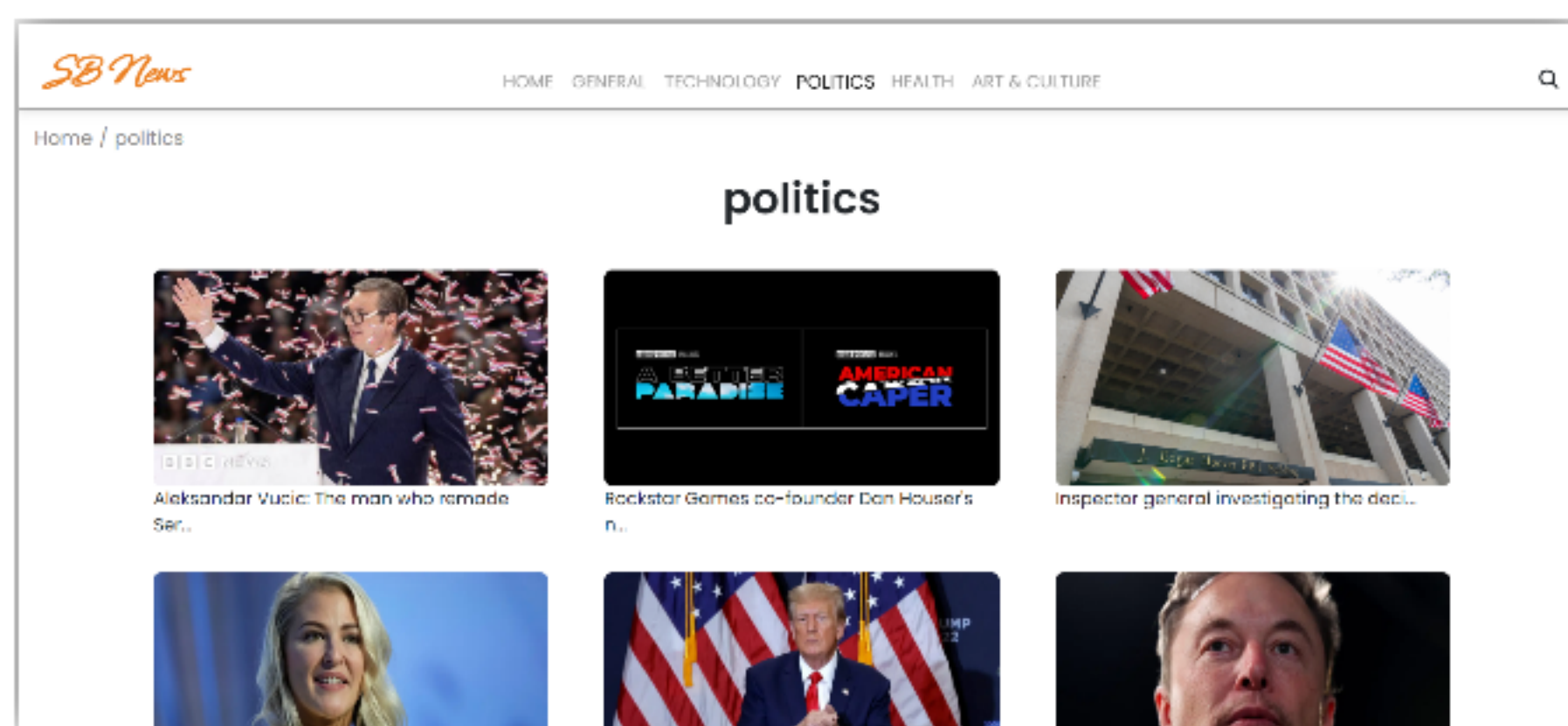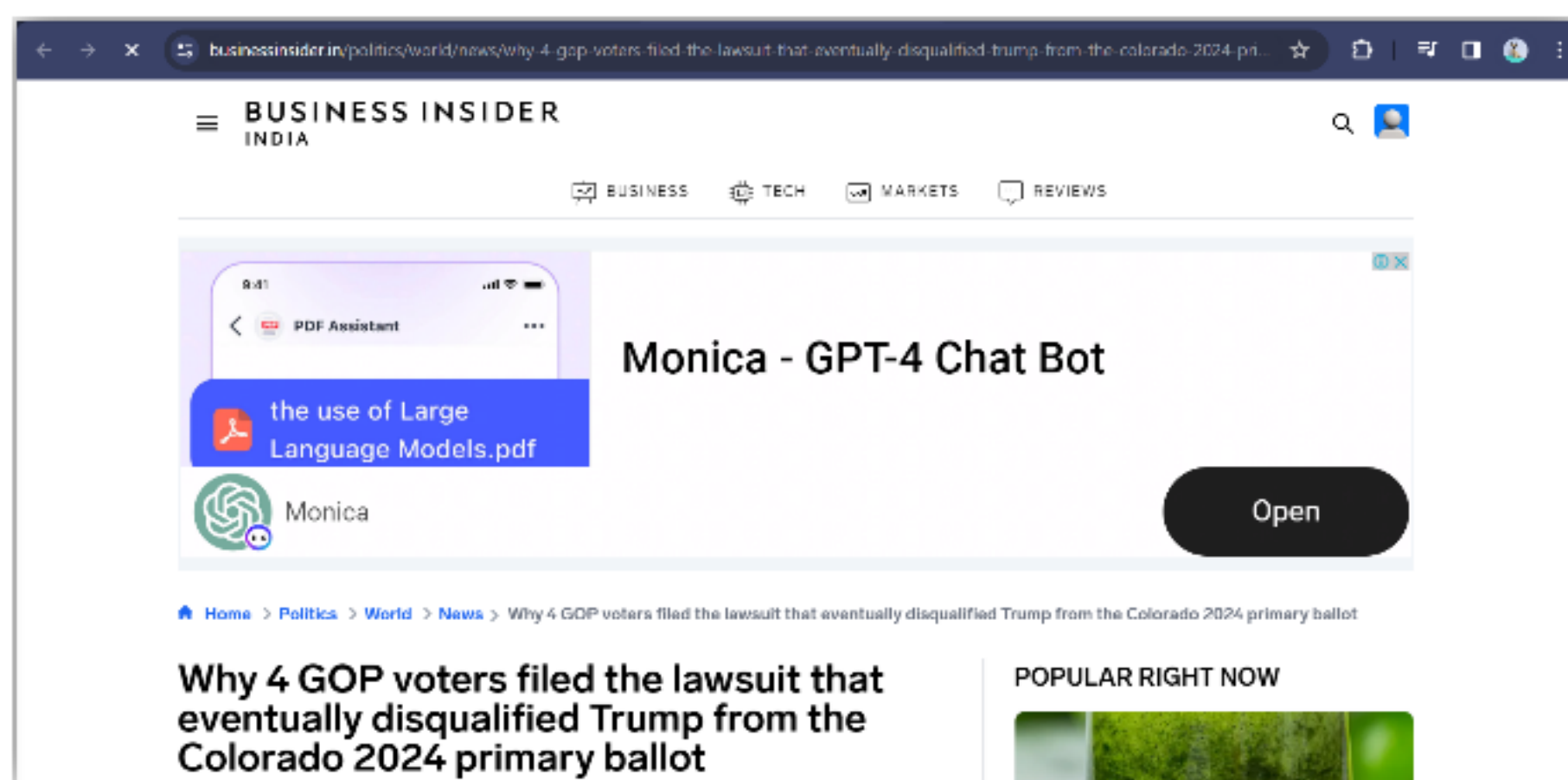## User Interface snips:

► Hero components

► Popular categories

► Newsletter

► Category/Search result page

► Redirected page

*** Happy coding!! ***