

# InsightStream: Navigate the News Landscape

## ➤ **project Objective**

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.

## **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.
- **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.**
- Development Environment: Choose a code editor or Integrated Development Environment (IDE) that suits your preferences, such as Visual Studio Code, Sublime Text, or 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. 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
- ① 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 folde

## ➤ **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**

## ➤ **Project Development**

- **Setup the Routing paths**

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

```

<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>

```

Ex:

- 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



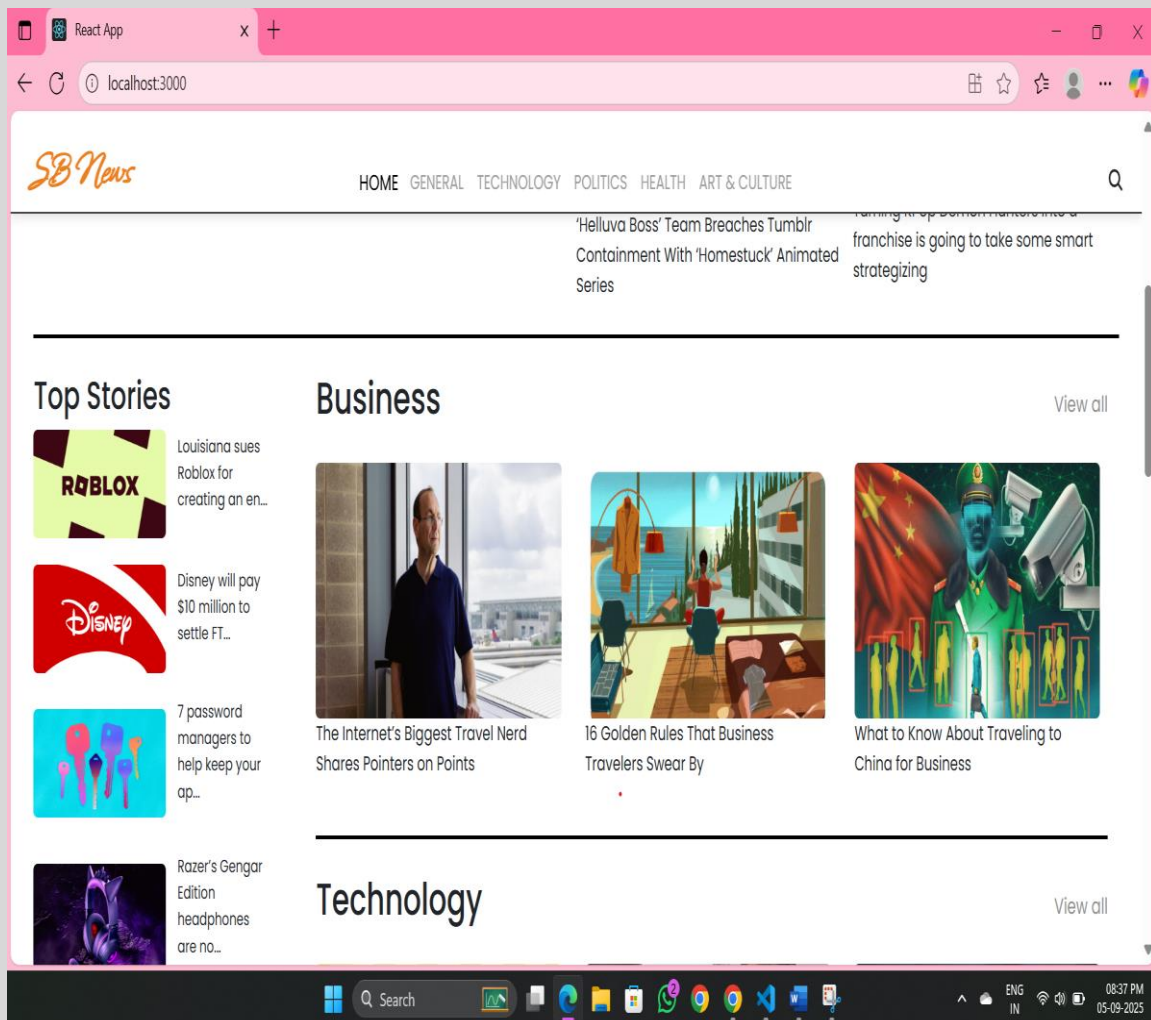
```
const [categoryNews, setCategoryNews] = useState([]);

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





## WIRED Tests Dozens of Air Purifiers a Year. Here's What We Look For (2025)



## Apple May Take AirPods Pro 3 Beyond Audio



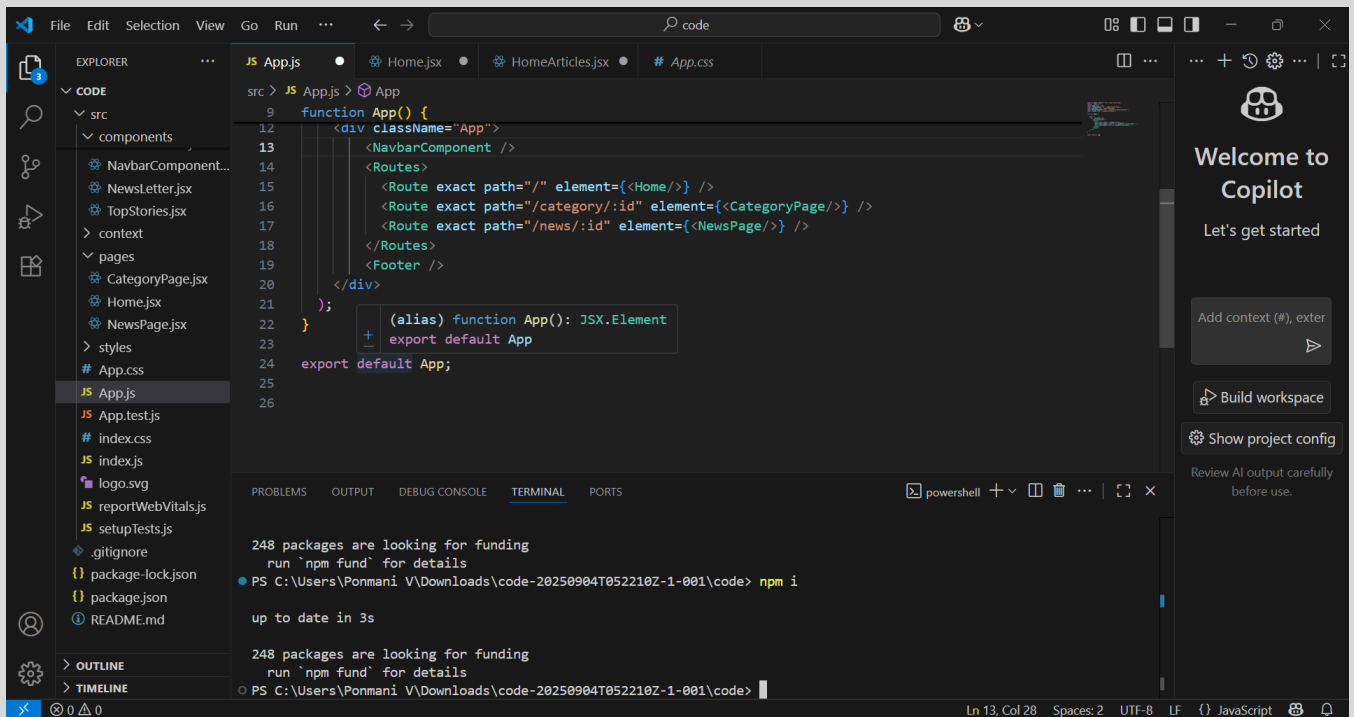
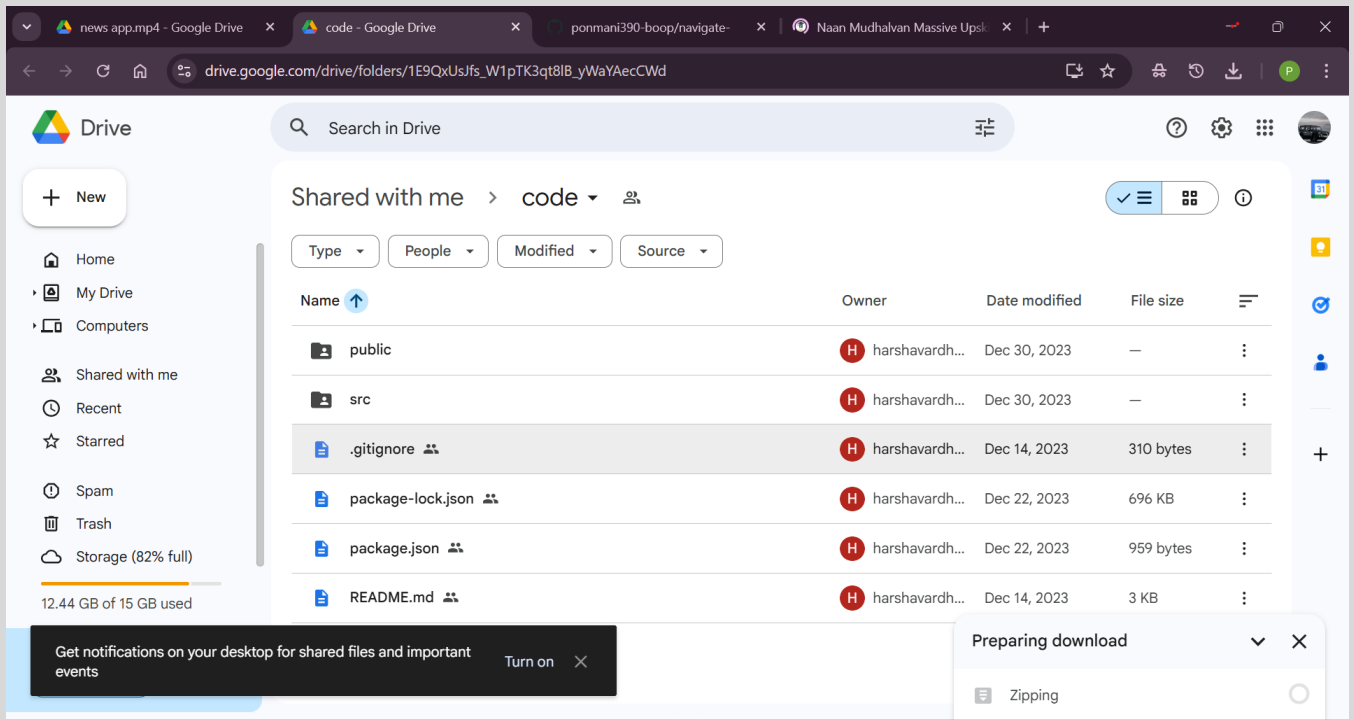
## 'Helluva Boss' Team Breaches Tumblr Containment With 'Homestuck' Animated Series

```
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Install the latest PowerShell for new features and improvements! https://aka.ms/PSWindows

PS C:\Users\Ponmani V> set-ExecutionPolicy Unrestricted
set-ExecutionPolicy : Access to the registry key 'HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\PowerShell\1\ShellIds\Microsoft.PowerShell' is denied. To change the execution policy for the default (LocalMachine) scope, start Windows PowerShell with the "Run as administrator" option. To change the execution policy for the current user, run "Set-ExecutionPolicy -Scope CurrentUser".
At line:1 char:1
+ set-ExecutionPolicy Unrestricted
+ ~~~~~
+ CategoryInfo          : PermissionDenied: (:) [Set-ExecutionPolicy], UnauthorizedAccessException
+ FullyQualifiedErrorId : System.UnauthorizedAccessException,Microsoft.PowerShell.Commands.SetExecutionPolicyCommand

PS C:\Users\Ponmani V>
```



```
File Edit Selection View Go Run ... code
src > components > HomeArticles.jsx > HomeArticles > politicsNews.map() callback
1 import React, { useContext } from 'react'
2 import '../styles/HomeArticles.css'
3 import { GeneralContext } from '../context/GeneralContext'
4 import { useNavigate } from 'react-router-dom';
5 import { Spinner } from 'react-bootstrap';
6
7 const HomeArticles = () => {
8
9     const navigate = useNavigate();
10
11     const {businessNews, technologyNews, politicsNews} = useContext(GeneralContext)
12
13     return (
14
15         <div className='home-articles-container'>
16
17             <div className='home-articles-body'>
18
19                 <div className="home-articles-head">
20                     <h2>Business</h2>
21                     <p onClick={() => navigate('/category/business')} >View all</p>
22                 </div>
23
24                 {businessNews.length > 0 ?
25
26                     <div className="home-articles">
27
28                         {businessNews.map((news, index) =>{
29                             return index < 3 && (
```

Ln 99, Col 20 Spaces: 2 UTF-8 CRLF {} JavaScript JSX

28°C Mostly cloudy Search 09:59 PM 05-09-2025

