

## Introduction

### Project Title: InsightStream

#### Team Members:

- **Aafrin Mofica F** – Team Leader

Email id- aafrinmofica@gmail.com

- **Sridevi S**– Frontend Developer

Email id- s.sridevikpm@gmail.com

- **Nathiya E**– UI/UX Designer

Email id- srinathiya72@gmail.com

- **Jasmine pushpa K** – Frontend Developer

Email id-Kumarbommi1980@gmail.com

## Project Overview

### Purpose:

InsightStream is a dynamic news web application that enhances how users explore and consume news. It provides an intuitive and responsive interface, offering access to a wide range of news categories sourced from APIs.

### Features:

- ✓ **News from API Sources** – Fetch real-time news from global sources using APIs.
- ✓ **Visual News Exploration** – Show breaking stories with curated image galleries.
- ✓ **Intuitive UI** – Seamless navigation with a modern and responsive design.
- ✓ **Advanced Search** – Locate specific news articles using keywords or categories.
- ✓ **Newsletter Subscription** – Stay updated with curated news delivered via email.

## Architecture

### Component Structure:

- **NavbarComponent.jsx** – Provides navigation across different sections.
- **Hero.jsx** – Displays trending news and includes a search bar.
- **HomeArticles.jsx** – Lists news articles based on categories.
- **TopStories.jsx** – Highlights major trending stories.
- **NewsLetter.jsx** – Allows users to subscribe to news updates.
- **Footer.jsx** – Displays relevant links and copyright information.

### State Management:

Utilizes **React Context API** for managing global states such as user preferences and news categories.

Uses **useState** and **useEffect** hooks for handling local state within components.

### Routing:

Implemented using React Router Dom to manage navigation between different pages:

- **/** – Home Page (Displays latest news articles)
- **/top-stories** – Displays trending news stories
- **/category/:id** – Shows news related to a selected category
- **/search** – Displays search results

### Setup Instructions

## Prerequisites:

Ensure the following are installed:

- ✓ **Node.js & npm** - Download [here](#)
- ✓ **React.js** - Installed via `npx create-react-app`
- ✓ **Git** - Download [Git](#)
- ✓ **IDE** - Recommended: VS Code

## Installation Steps:

1. Clone the repository:

```
git clone <repository-link>
```

```
cd news-app-react
```

2. Install dependencies:

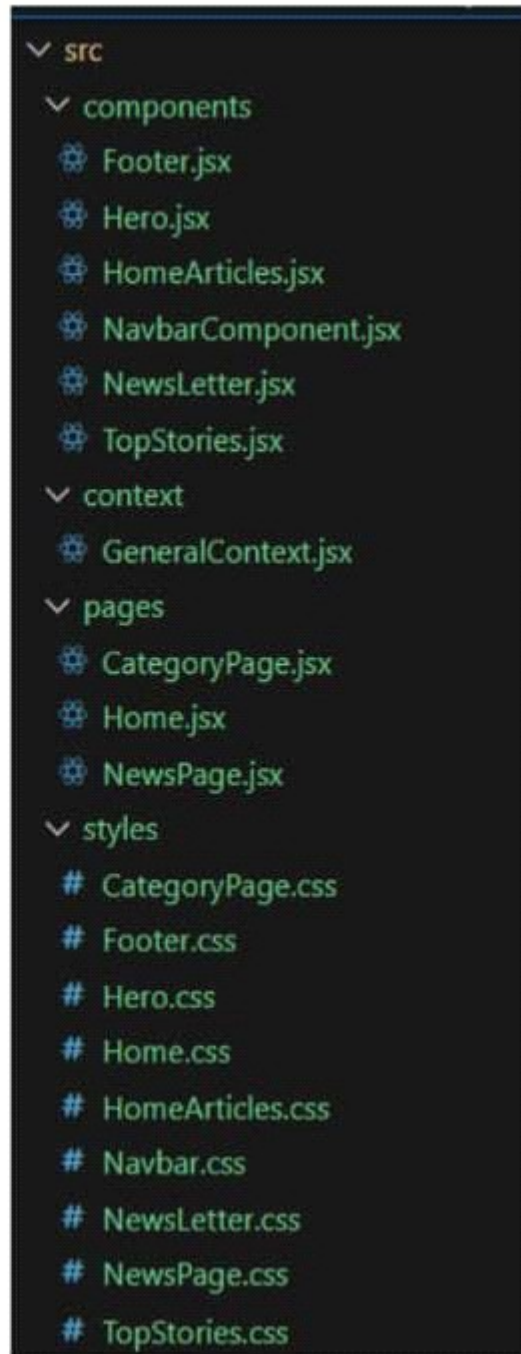
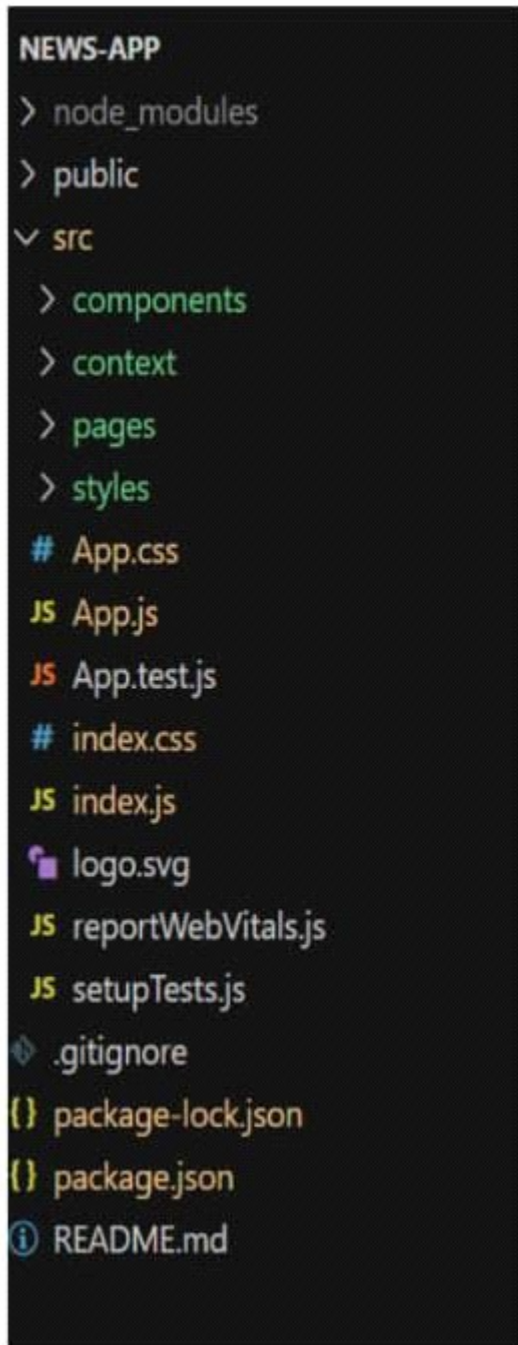
```
npm install
```

3. Start the development server:

```
npm start
```

4. Open `http://localhost:3000` in your browser to view the app.

## Folder 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.

## Client (API Handling)

The client/ folder is responsible for interacting with APIs to fetch news dynamically.

- **apiClient.js** – This file centralizes API requests, making it easier to manage endpoints and handle errors.

```
19
20 const fetchNews = async (id) => {
21   try {
22     const response = await axios.get(`https://newsapi.org/v2/everything?q=${id}&apiKey=285c70d635ee4fb79d690bb470406det`);
23     setCategoryNews(response.data.articles);
24   } catch (error) {
25     console.error(error);
26   }
27 }
28
29 const [activePage, setActivePage] = useState(1);
```

## Utilities (Helper Functions & Custom Hooks)

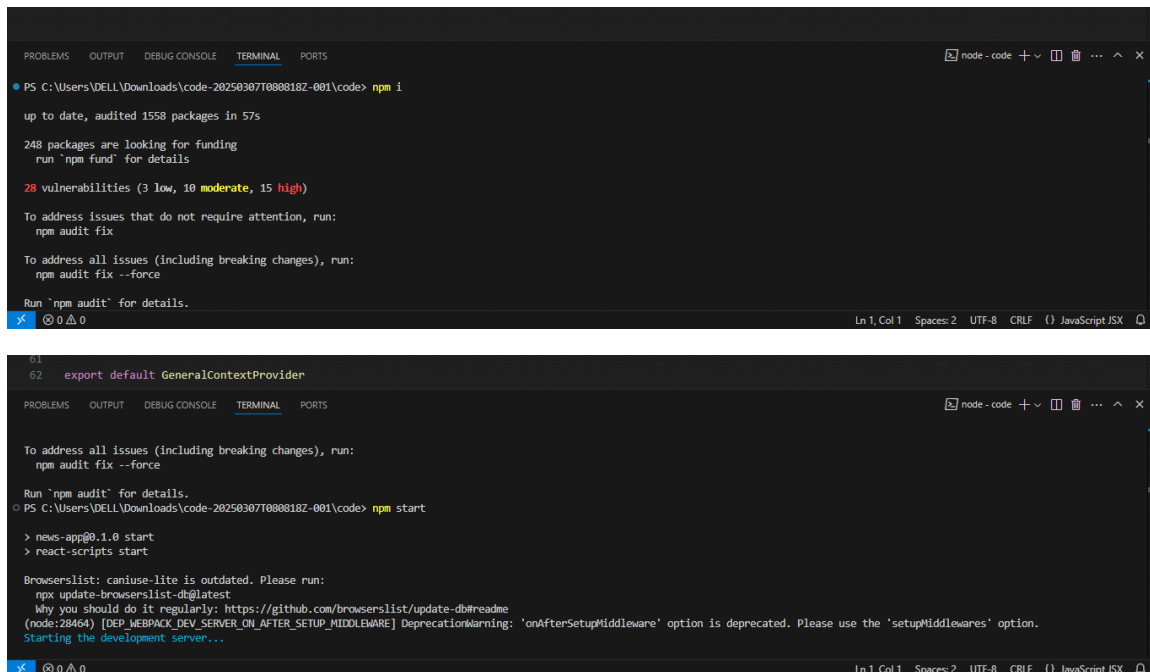
The utilities/ folder contains helper functions and custom hooks to make the codebase cleaner.

### Helper Functions

- **formatDate.js** – Formats dates for displaying news timestamps.
- **truncateText.js** – Shortens long news titles or descriptions.

### Running the Application:

To start the frontend server locally, use:



```
PS C:\Users\DELL\Downloads\code-20250307T080818Z-001\code> npm i

up to date, audited 1558 packages in 57s

248 packages are looking for funding
  run `npm fund` for details

28 vulnerabilities (3 low, 10 moderate, 15 high)

To address issues that do not require attention, run:
  npm audit fix

To address all issues (including breaking changes), run:
  npm audit fix --force

Run `npm audit` for details.

61
62 export default GeneralContextProvider

To address all issues (including breaking changes), run:
  npm audit fix --force

Run `npm audit` for details.
PS C:\Users\DELL\Downloads\code-20250307T080818Z-001\code> npm start

> news-app@0.1.0 start
> react-scripts start

Browserslist: caniuse-lite is outdated. Please run:
  npx update-browserslist-db@latest
  why you should do it regularly: https://github.com/browserslist/update-db#readme
(node:28464) [DEP_WEBPACK_DEV_SERVER_ON_AFTER_SETUP_MIDDLEWARE] DeprecationWarning: 'onAfterSetupMiddleware' option is deprecated. Please use the 'setupMiddlewares' option.
Starting the development server...
```

1. npm i
2. npm start

## Component Documentation

### Key Components:

- ✓ **NavbarComponent.jsx** – Navigational header.
- ✓ **Hero.jsx** – Displays trending news and search bar.
- ✓ **HomeArticles.jsx** – Shows category-based news articles.
- ✓ **TopStories.jsx** – Highlights major global stories.
- ✓ **NewsLetter.jsx** – Allows users to subscribe for updates.
- ✓ **Footer.jsx** – Displays relevant links and credits.

### Reusable Components:

- ✓ **ArticleCard.jsx** – Reusable card component for displaying news articles.

- ✓ **Button.jsx** – Customizable button component.

## State Management

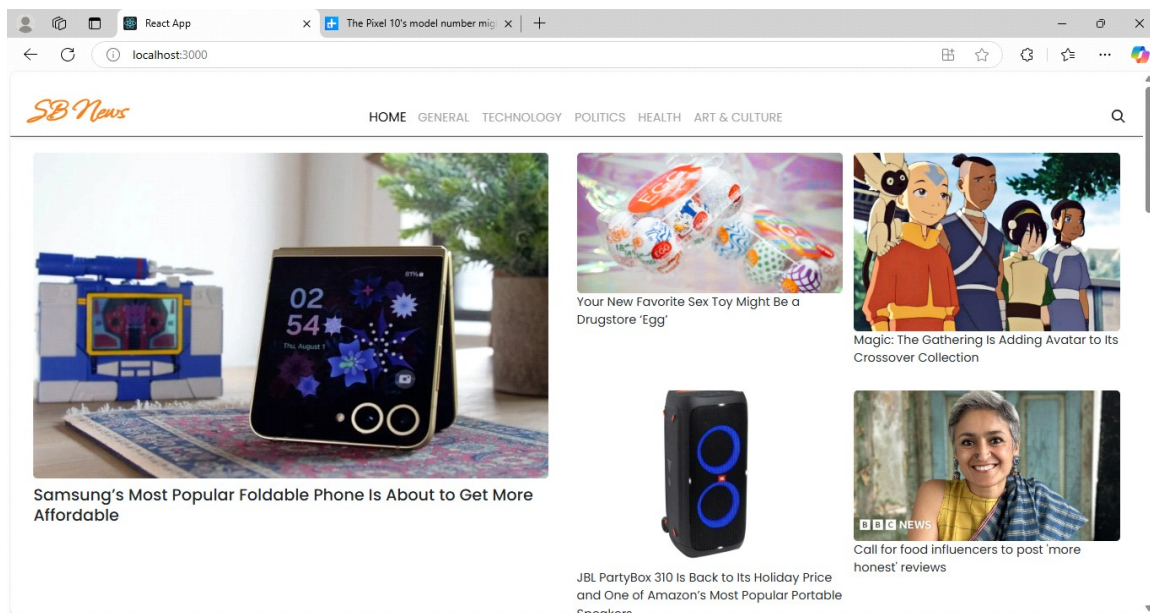
### Global State:

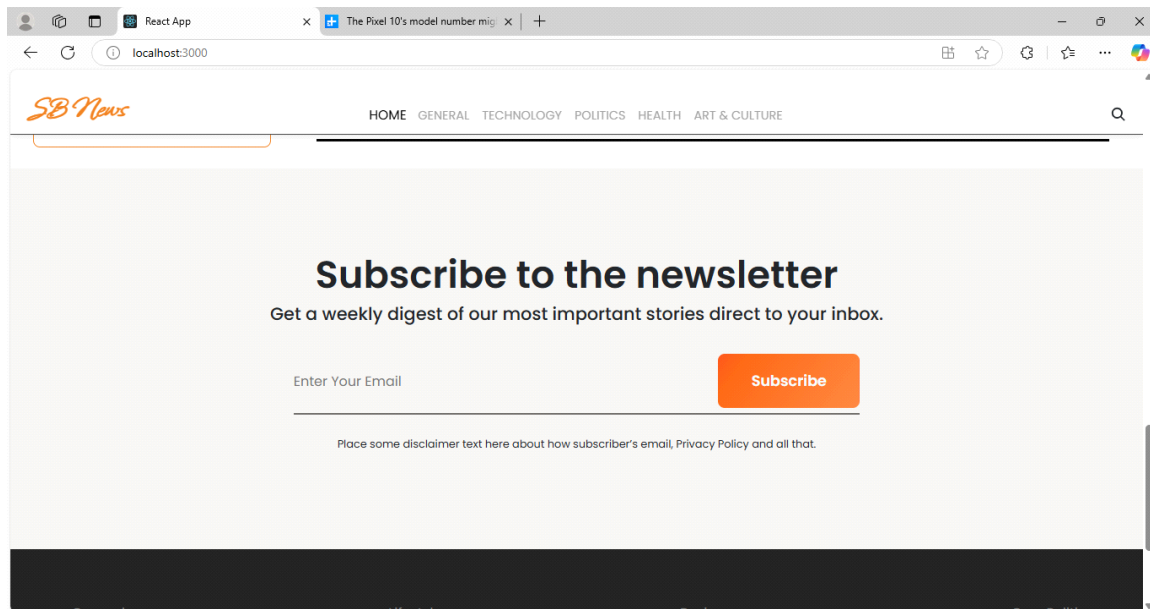
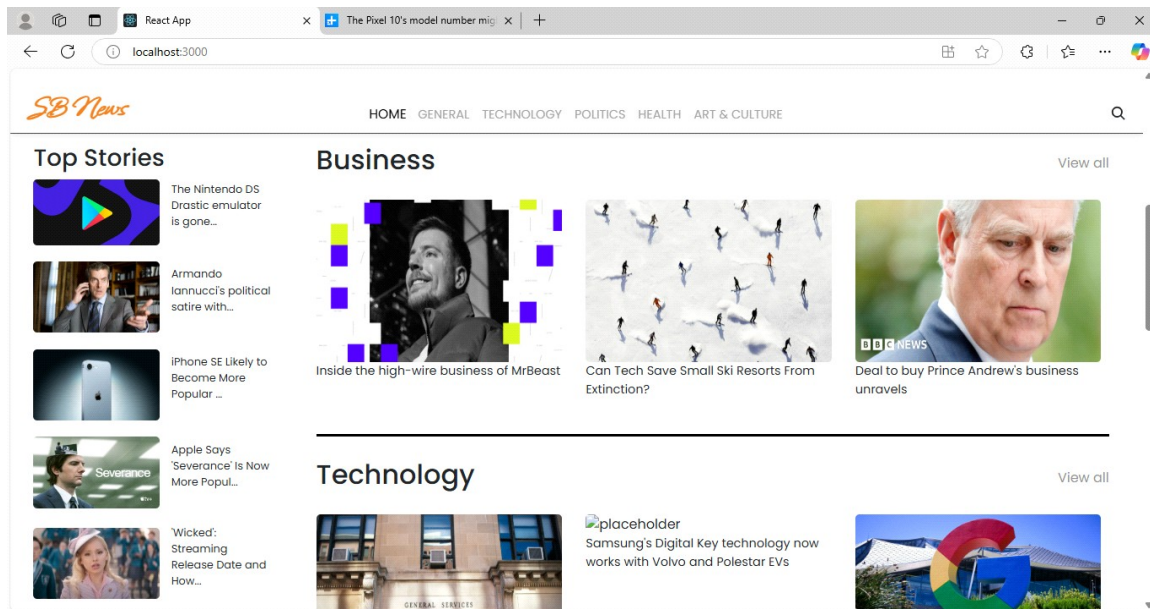
- ✓ Context API is used to store and share news data, user preferences, and search queries across components.

### Local State:

- ✓ `useState` & `useEffect` hooks are used to manage component-specific states, like search input and filtered news lists.

## User Interface





## Styling

- ✓ **CSS Framework:** Bootstrap/Tailwind CSS for responsive design.
- ✓ **Theming:** Written in CSS Modules to maintain scoped styling.

## Testing



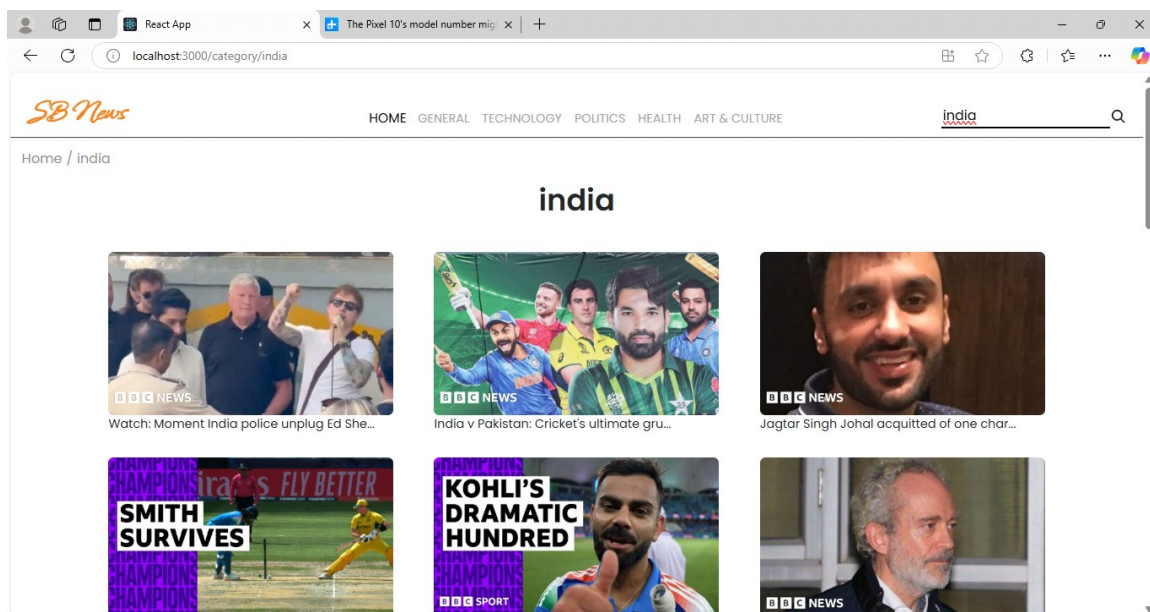
## Testing Strategy:

- ✓ Unit testing with Jest and React Testing Library.
- ✓ API response testing using Mock Service Worker (MSW).

## Code Coverage:

- ✓ Uses Jest Coverage to track test effectiveness.

## Screenshots or Demo





**Project Demo Link:** [https://drive.google.com/file/d/1VUrJe6gG2S-5rubpNPWuKXZavXy2Fmdl/view?usp=drive\\_link](https://drive.google.com/file/d/1VUrJe6gG2S-5rubpNPWuKXZavXy2Fmdl/view?usp=drive_link)

## Known Issues

- ✓ API rate limits may affect news fetching.
- ✓ UI optimizations required for mobile responsiveness.

## Future Enhancements

- ✓ Implement Dark Mode.
- ✓ Add User Authentication for personalized news.
- ✓ Integrate Push Notifications for breaking news alerts.