**Frontend Development with React.js**

**Project Documentation format**

1. **Introduction**
   * Project Title: Insight Stream - Navigate the News Landscape
   * Team ID : NM2025TMID39988  
       
       
     Team Leader : HARSHINI E  
     [sac2427bca5777@ssacollegechennai.com](mailto:sac2427bca5777@ssacollegechennai.com)
   * Role: linking and uploading code  
       
     Team Member : HARINI M  
     [sac2427bca5213@ssacollegechennai.com](mailto:sac2427bca5213@ssacollegechennai.com)
   * Role: project document  
       
     Team Member : JASRA J  
     sac2427bca5454@ssacollegechennai.com
   * Role:demo video  
     Team Member : JENCY JAYA G  
     [sac2427bca5345@ssacollegechennai.com](mailto:sac2427bca5345@ssacollegechennai.com)
   * Role: project document
2. Project Overview  
     
   Purpose: The purpose of InsightStream is to provide users with a clean and intuitive platform to explore and navigate through the latest news articles across various categories. The application aims to simplify news discovery and help users stay informed with curated articles.  
     
   Features:  
   - Categorized News Sections: Business, Technology, Politics, Health, and Art & Culture.  
   - Top Stories: Displays trending news articles.  
   - Search Functionality: Allows users to search for specific topics.  
   - Newsletter Subscription: Users can subscribe for weekly news digests.  
   - Responsive UI: Mobile-friendly layout using Bootstrap.  
   - External Links: Each news article links to its original source.
   * **Purpose**: Briefly describe the purpose and goals of the project.
   * **Features**: Highlight the key features and functionalities of the frontend.
3. State Management  
     
   Global State:  
   - Managed using React Context API (GeneralContext).  
   - Provides news data to components across pages.  
     
   Local State:  
   - useState hooks handle component-level states like loading and error handling.
   * **Component Structure**: Outline the structure of major React components and how they interact.
   * State Management  
       
     Global State:  
     - Managed using React Context API (GeneralContext).  
     - Provides news data to components across pages.  
       
     Local State:  
     - useState hooks handle component-level states like loading and error handling.
   * **Routing**: Explain the routing structure if using react-router or another routing library.
4. Setup Instructions  
     
   Prerequisites:  
   - Node.js (v16 or above)  
   - npm (v8 or above)  
     
   Installation:  
   1. Clone the repository.  
   2. Navigate to the project folder.  
   3. Run `npm install` to install dependencies.  
   4. Run `npm start` to launch the development server.  
     
   Environment Variables:  
   - Add your News API key in GeneralContext.jsx (65d48d4cfd19490eb3cba17ee2f1379c).
   * **Prerequisites**: List software dependencies (e.g., Node.js).
   * **Installation**: Provide a step-by-step guide to clone the repository, install dependencies, and configure environment variables.
5. Folder Structure  
     
   Client:  
   - components/: Navbar, Hero, Footer, TopStories, Newsletter, HomeArticles.  
   - context/: GeneralContext.jsx for global state and API calls.  
   - pages/: Home.jsx, CategoryPage.jsx, NewsPage.jsx.  
   - styles/: CSS files for each component/page for modular styling.  
     
   Utilities:  
   - Axios is used for fetching data from the News API.  
   - React Bootstrap and React Icons for UI and icons.
   * **Client**: Describe the organization of the React application, including folders like components, pages, assets, etc.
   * **Utilities**: Explain any helper functions, utility classes, or custom hooks used in the project.
6. Running the Application  
     
   To run the application locally:  
   1. Open terminal in the project root.  
   2. Run `npm install`.  
   3. Run `npm start`.  
   4. Access the app at http://localhost:3000.
   * Provide commands to start the frontend server locally.
     + **Frontend**: npm start in the client directory.
7. Component Documentation  
     
   Key Components:  
   - NavbarComponent: Provides navigation.  
   - Hero: Displays introduction.  
   - TopStories: Shows trending articles.  
   - HomeArticles: Displays news by category.  
   - Newsletter: Handles email subscription.  
   - Footer: Displays footer information.  
     
   Reusable Components:  
   - Spinner (React Bootstrap): Displays loading states.  
   - Article Cards: Used across categories.
   * **Key Components**: Document major components, their purpose, and any props they receive.
   * **Reusable Components**: Detail any reusable components and their configurations.
8. State Management  
     
   Global State:  
   - Managed using React Context API (GeneralContext).  
   - Provides news data to components across pages.  
     
   Local State:  
   - useState hooks handle component-level states like loading and error handling.
   * **Global State**: Describe global state management and how state flows across the application.
   * **Local State**: Explain the handling of local states within components.
9. User Interface  
     
   The UI is designed with simplicity and readability in mind. Bootstrap provides responsive layout. React Icons enhance navigation. Key UI elements:  
   - Navbar for navigation.  
   - Search bar for quick access to news.  
   - Card layout for articles.  
   - Newsletter subscription box.  
   - Footer for credits and links.
   * Provide screenshots or GIFs showcasing different UI features, such as pages, forms, or interactions.
10. Styling  
      
    CSS Frameworks/Libraries:  
    - Bootstrap for responsive design.  
    - Custom CSS files for each component.  
      
    Theming:  
    - Consistent color scheme across components.  
    - Modular CSS organization.

* **CSS Frameworks/Libraries**: Describe any CSS frameworks, libraries, or pre-processors (e.g., Sass, Styled-Components) used.
* **Theming**: Explain if theming or custom design systems are implemented.

1. Testing  
     
   Testing Strategy:  
   - Basic testing with Jest and React Testing Library.  
   - Unit tests for components.  
   - Integration tests for API data rendering.  
     
   Code Coverage:  
   - Coverage ensured by writing unit tests for main components like Navbar, HomeArticles.

* Testing  
    
  Testing Strategy:  
  - Basic testing with Jest and React Testing Library.  
  - Unit tests for components.  
  - Integration tests for API data rendering.  
    
  Code Coverage:  
  - Coverage ensured by writing unit tests for main components like Navbar, HomeArticles.
* **Code Coverage**: Explain any tools or techniques used for ensuring adequate test coverage.

1. Screenshots or Demo  
     
   Demo Link: <https://drive.google.com/file/d/1b8SCuhk59Z0Kadp-1Vb3vbcLtsHMQz5Y/view?usp=drivesdk>
2. Known Issues  
     
   - Free News API may have restrictions on localhost calls.  
   - Some categories may return limited articles.  
   - Occasional CORS errors depending on environment.

* Document any known bugs or issues that users or developers should be aware of.

1. Future Enhancements  
     
   - Add user authentication for personalized feeds.  
   - Implement dark mode for better UX.  
   - Add bookmarking functionality for saving articles.  
   - Expand to support multiple languages.

* Outline potential future features or improvements, such as new components, animations, or enhanced styling.