**INSIGHTSTREAM : NAVIGATE THE NEWS LANDSCAPE**

**1. Introduction**

Project Title: InsightStream2

Team Members:

[Adhavan.S.R] –[ Project Coordinator]

[Jagadeeswaran.B] – [Research Manager]

[Nivesh.T] – [Designer]

[Devraj.S] – [Developer]

**2. Project Overview**

**Purpose:**

InsightStream2 is a React.js-based news aggregator frontend that fetches and displays categorized news

articles from the NewsAPI. The goal is to provide users with a seamless browsing experience for news by

category.

**Features:**

Category-based news browsing

Paginated article lists

Responsive UI

Article previews with images

Navigation between categories and home

**3. Architecture**

**Component Structure** **:**

**App:** Root component, handles routing.

**CategoryPage:** Displays news articles for a selected category.

**HomePage:** Main landing page with category selection.

**ArticleCard:** Reusable component for displaying individual articles.

**Navbar:** Navigation bar for site-wide navigation.

**State Management:**

Local state via React useState and useEffect hooks.

No global state management library; state is managed within components.

**Routing:**

Uses react-router-dom for client-side routing.

**4. Setup Instructions**

**Prerequisites:**

Node.js (v16+ recommended)

npm

**Installation:**

**1.** Clone the repository:

git clone https://github.com/yourusername/InsightStream2.git

**2.** Navigate to the project directory:

cd InsightStream2-main/code

**3.** Install dependencies:

npm install

**4.** Configure environment variables (if needed):

Add your NewsAPI key in the appropriate file or .env.

**5.** Folder Structure

**Client:**

src/components/ – Reusable UI components (e.g., Navbar, ArticleCard)

src/pages/ – Page components (e.g., HomePage, CategoryPage)

src/assets/ – Images, icons, and other static assets

src/styles/ – CSS files for styling

**Utilities:**

Helper functions for API calls (e.g., fetchNews)

Custom hooks (if any) for data fetching or state logic

**6.** Running the Application

To start the frontend server locally:

npm start

Run this command inside the code directory.

**7.** Component Documentation

**Key Components:**

CategoryPage

**Purpose:** Displays news articles for a selected category.

**Props:** Uses useParams for category ID; no direct props.

**Features:** Fetches news, paginates results, displays articles.

ArticleCard

**Purpose:** Renders individual article preview.

**Props:** article (object with title, image, url, etc.)

**Reusable Components:**

ArticleCard – Used in lists to display articles.

Navbar – Used across pages for navigation.

**8. State Management**

**Global State:**

Not implemented; all state is local to components.

**Local State:**

**CategoryPage:**

categoryNews – Stores fetched articles.

activePage – Tracks current pagination page.

**9. User Interface**

Screenshots or GIFs can be added here to showcase:

Home page with category selection

Category page with paginated articles

Article preview cards

**10. Styling**

CSS Frameworks/Libraries:

React Bootstrap for UI components (e.g., Pagination, Spinner)

Custom CSS in src/styles/

Theming:

No custom theming system implemented.

**11. Testing**

Testing Strategy:

Unit tests for components using Jest and React Testing Library.

Integration tests for page rendering and API calls.

**Code Coverage:**

Use jest --coverage to check coverage.

**12.** Screenshots or Demo

Add screenshots here or link to a live demo:

Home Page

Category Page

**13.** Known Issues

API rate limits may affect news loading.

Some articles may lack images, causing broken previews.

Pagination may not handle edge cases with few articles.

**14.** Future Enhancements

Add user authentication and personalized feeds.

Improve error handling and loading states.

Enhance UI with animations and transitions.

Implement global state management (e.g., Redux).

Add dark mode and theming