**InsightStream: Navigating the News - Project Documentation**

**1. Introduction**

**Project Title: InsightStream: Navigating the News**

**Team Members: Mail ID**

* **MADHAVAN D** - Project Leader madhavandmaddy2005@gmail.com
* **VISHVA ST** – Member vishva25112004@gmail.com
* **GANGAI AMARAN S** – Member agangai54@gmail.com
* **PARANDHAMAN C** – Member paranthaman161104@gmail.com
* **DHARISANAM N** – Member ndharisanam@gmail.com

**2. Project Overview**

**Purpose**

InsightStream is an AI-powered news aggregation platform designed to provide real-time, personalized, and credible news insights to users. The platform leverages AI and machine learning to analyze and curate news articles based on user preferences and trends.

**Features (Frontend)**

* **Personalized News Feed** – AI-driven recommendations based on user interactions.
* **Real-time Search & Filters** – Advanced filtering by category, region, and keyword.
* **News Summarization** – AI-generated article summaries.
* **Sentiment Analysis** – Highlights positive, neutral, or negative tones in news articles.
* **Bookmarks & Reading List** – Save articles for later reading.
* **Multi-Language Support** – Accessibility for a global audience.
* **Dark Mode & Custom Themes** – Enhanced readability and UI customization.
* **Social Sharing** – Share news articles directly on social media.

**Component Structure**

* **App.js** – Root component managing routes and global state.
* **Layout.js** – Manages header, sidebar, and main content structure.
* **NewsFeed.js** – Displays news articles dynamically.
* **NewsCard.js** – Reusable component for displaying a single article.
* **SearchBar.js** – Handles search input and queries.
* **SentimentIndicator.js** – Displays sentiment analysis results.
* **BookmarkPage.js** – Lists saved articles for later reading.

**State Management**

* **Redux Toolkit** is used for global state management:
* newsSlice.js – Manages fetched news articles.
* userPreferencesSlice.js – Stores user preferences.
* bookmarksSlice.js – Handles bookmarked articles.
* authSlice.js – Manages authentication.
* **useState and useContext** for handling UI-related states.

**Routing (React Router)**

* path: '/' – Home Page
* path: '/news' – News Feed
* path: '/search' – Search Results
* path: '/bookmarks' – Bookmarked Articles
* path: '\*' – 404 Not Found

**4. Setup Instructions**

**Prerequisites**

* Node.js (v16 or later)
* npm or yarn
* Git
* Code Editor (VS Code recommended)

**Installation Guide**

# Clone the repository

git clone https://github.com/your-username/InsightStream.git

cd InsightStream

# Install dependencies

npm install # or yarn install

# Set up environment variables

cp .env.example .env

Configure .env with your API key and backend URL:

REACT\_APP\_API\_KEY=your\_api\_key\_here

REACT\_APP\_API\_BASE\_URL=https://newsapi.org/v2

Run the development server:

npm start # or yarn start

**5. Folder Structure**

InsightStream/

├── public/ # Static assets

├── src/

│ ├── components/ # Reusable UI components

│ ├── pages/ # Page components

│ ├── redux/ # Redux slices and store

│ ├── utils/ # Helper functions and custom hooks

│ ├── styles/ # Global styles and themes

│ ├── App.js # Main app component

│ ├── index.js # App entry point

├── .env # Environment variables

├── package.json # Project dependencies

└── README.md # Project documentation

**6. Running the Application**

**Frontend**

npm start # or yarn start



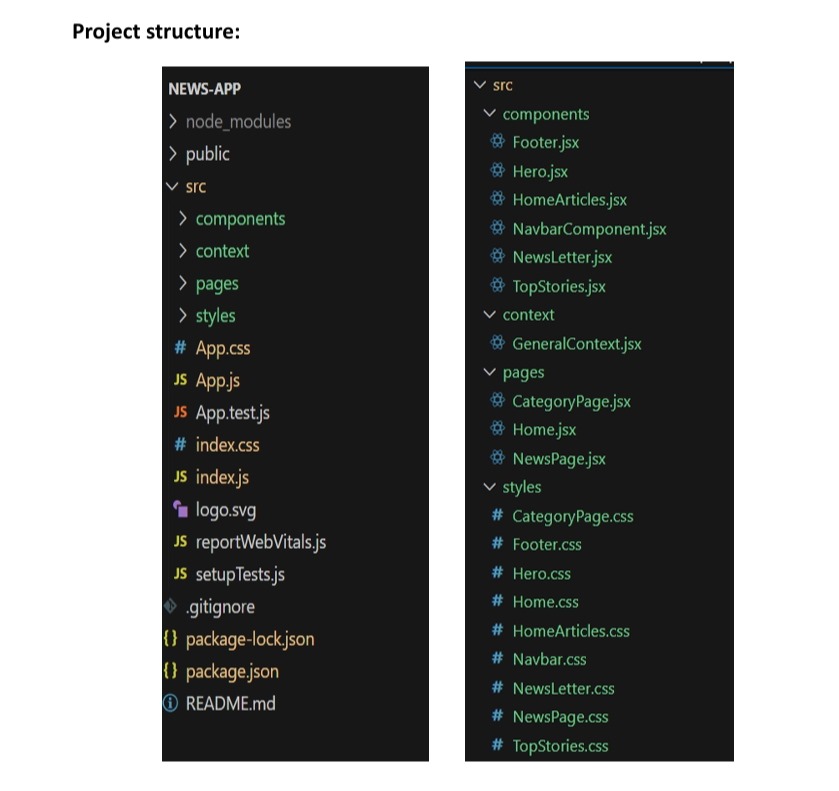
**7. Component Documentation**

**Key Components**

* **NewsCard.js** – Displays individual news articles.
* **NewsFeed.js** – Fetches and renders a list of articles.
* **SearchBar.js** – Handles search functionality.
* **BookmarkPage.js** – Displays saved articles.

**Reusable Components**

* **Button.js** – Customizable button component.
* **Loader.js** – Displays a loading spinner.
* **ErrorBoundary.js** – Catches and displays errors.



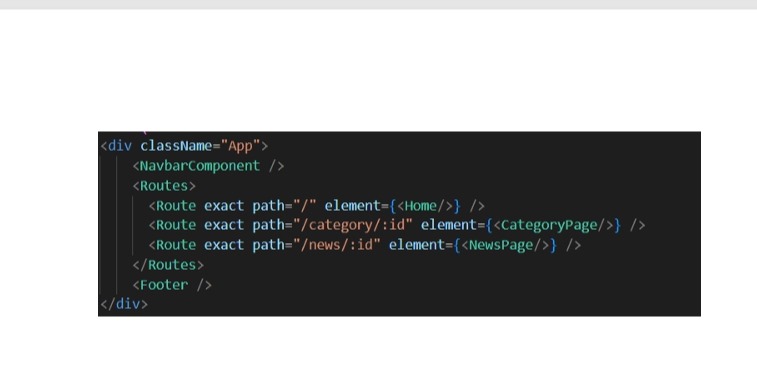
**8. State Management**

**Global State (Redux Toolkit)**

* Centralized store for news articles, user preferences, and authentication.
* Utilizes createSlice() for modular state management.

**Local State**

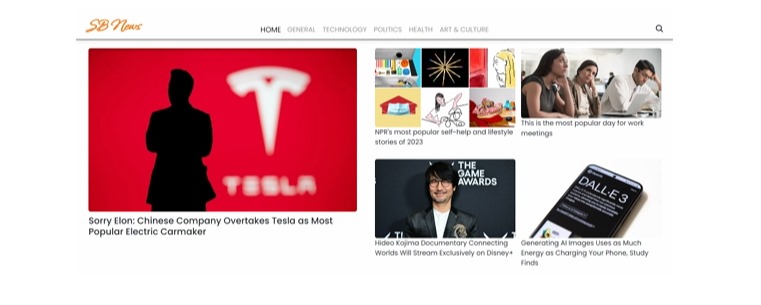
* useState for UI interactions like toggles and dropdowns.



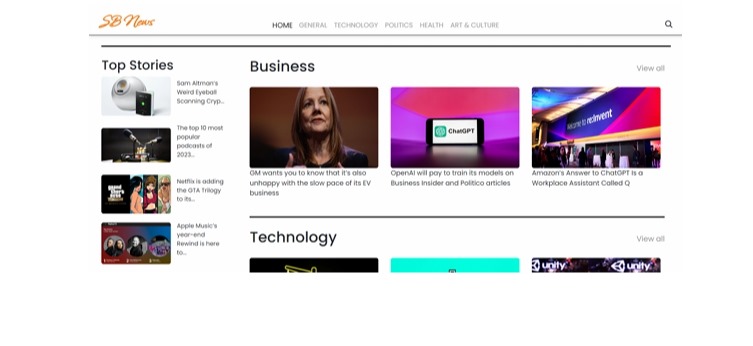
**9. User Interface**

Include screenshots of:

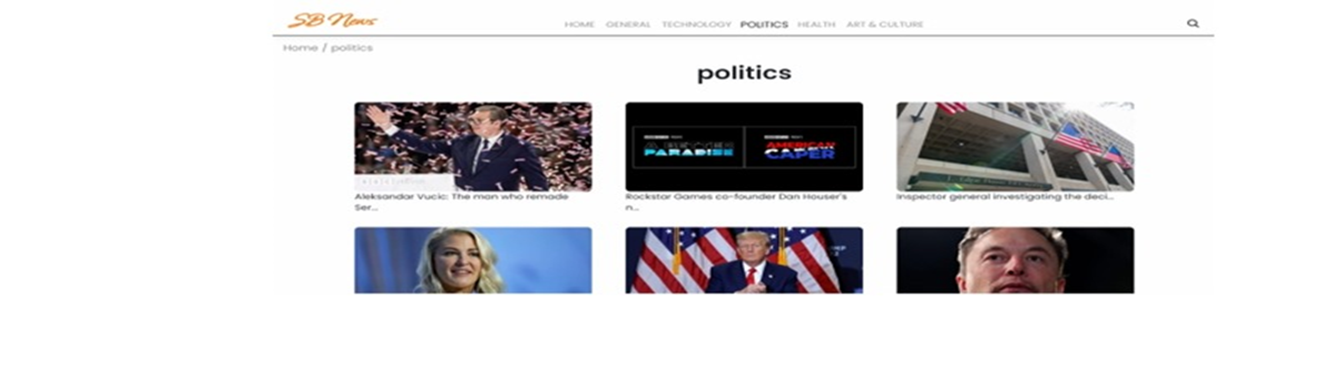
* Home Page



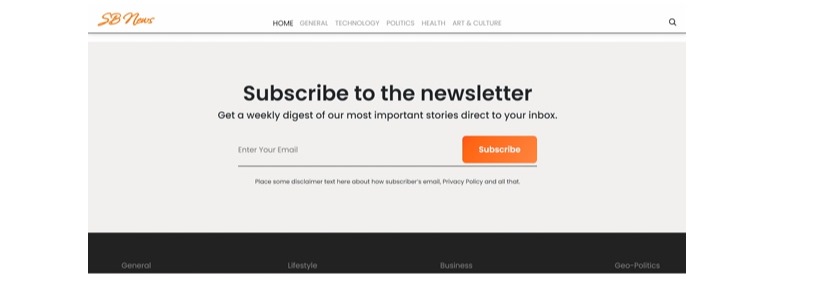
• News Feed



• Search Results



• Bookmark Page



**10. Styling**

**CSS Frameworks/Libraries**

* Tailwind CSS for responsive and modern styling.
* Styled-Components for dynamic styling.

**Theming**

* Dark mode toggle with persisted user settings.
* Customizable color themes.

**11. Testing**

**Testing Strategy**

* **Jest & React Testing Library** for unit tests.
* **Cypress** for end-to-end (E2E) testing.

**Code Coverage**

Run tests with:

npm test # or yarn test

**12. Demolink**

https://drive.google.com/file/d/1MryQrI\_Qq73MY7cfwysNP5Zi96X4ZC\_X/view?usp=sharing

**13. Known Issues**

* Some API endpoints may have rate limits.
* UI responsiveness improvements needed for smaller screens.
* Optimize search performance for large datasets.

**14. Future Enhancements**

* Implement user authentication (OAuth/Firebase/Auth0).
* Add AI-driven news topic summarization.
* Expand language support and regional filtering.
* Improve accessibility with screen reader support.