Insight Stream - News Application

Introduction

Project Title: Insight Stream

Team Members:

Team Leader : PRATHESH G

Email ID ; abi0402003@gmail

Team member : SUBASH

Email ID : subashswex28@gmail.com

Team member : SANTHOSH S

Email ID: Snila9443@mail.com

Team member : VARUN RAJ R

Email ID : varunraj9176@gmail.com

2. Project Overview

Purpose

Insight Stream is a modern news application designed to deliver categorized news to users in an intuitive and engaging manner. The primary goal is to keep users informed with the latest news across multiple categories while providing a seamless user experience.

Features

- Categorized news display (e.g., Technology, Sports, Business, Entertainment, etc.)
- Responsive and user-friendly UI
- Home page displaying trending and latest news
- About page with details about Insight Stream
- Subscribe page for users to sign up for personalized news updates

3. Architecture

Component Structure

- **App.jsx** - Root component managing routes and layout
- **Header.jsx** - Contains navigation and branding
- **Footer.jsx** - Displays copyright and links
- **HomePage.jsx** - Displays top and trending news
- **CategoryPage.jsx** - Shows news based on user-selected category
- **ArticleCard.jsx** - Reusable component to display individual news articles
- **AboutPage.jsx** - Information about Insight Stream
- **SubscribePage.jsx** - Allows users to subscribe for updates

State Management

- **Context API** is used for global state management, storing news data, user preferences, and subscriptions.
- Local state management within individual components for UI interactions.

Routing

- **React Router** is used to handle navigation between pages:

  - / - Home Page
  - /about - About Page
  - /subscribe - Subscription Page
  - /category/:category Name - Dynamic route for news categories

4. Setup Instructions

Prerequisites

Ensure you have the following installed:

- Node.js (latest LTS version recommended)
- Git download
- npm or yarn

Installation

- Clone the repository:
- Git clone https://github.com/Ashu1090/insightstream.git
- Navigate into the project directory:
- Cdinsightstream
- Install dependencies:
- npm install
- Start the development server:
- npm run dev

5. Folder Structure

```
insightstream/
│  ── src/
│      ── components/   # Reusable components (Header.jsx, Footer.jsx, ArticleCard.jsx)
│      ── pages/        # Page components (HomePage.jsx, AboutPage.jsx, SubscribePage.jsx)
│      ── assets/       # Images, icons, and styles
│      ── utils/        # Utility functions and custom hooks
│      ── context/      # Context API files for global state
│  ── public/           # Static files
│  ── package.json      # Project dependencies and scripts
│  ── README.md         # Project documentation
```

6. Running the Application

To start the frontend server locally, run:

npm run dev

This will launch the application at http://localhost:5173/.

7. Component Documentation

Key Components

- **ArticleCard.jsx**: Displays a single news article with title, image, and description.
- **CategoryList.jsx**: Renders a list of available news categories.
- **SubscribeForm.jsx**: Handles user subscription input and submission.

Reusable Components

- **Button.jsx**: Styled button used across the app.
- **Loader.jsx**: Displays a loading spinner while fetching data.

8. State Management

Global State

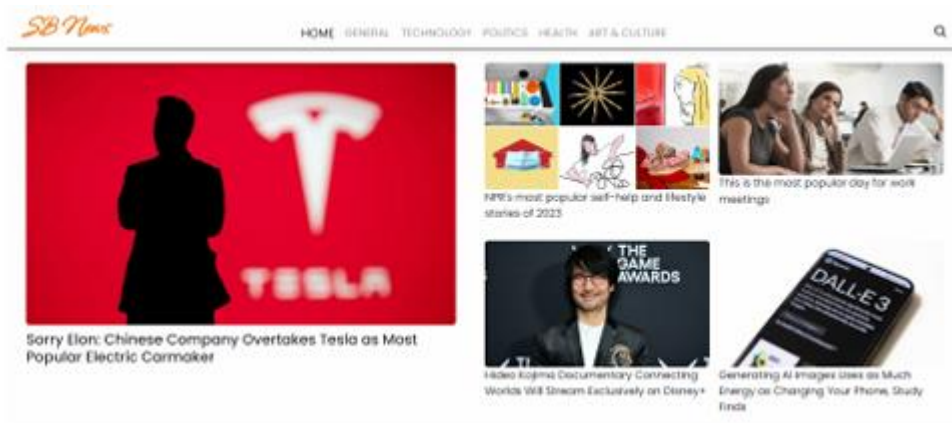- Managed via **Context API** to store and share news data and user preferences.

Local State

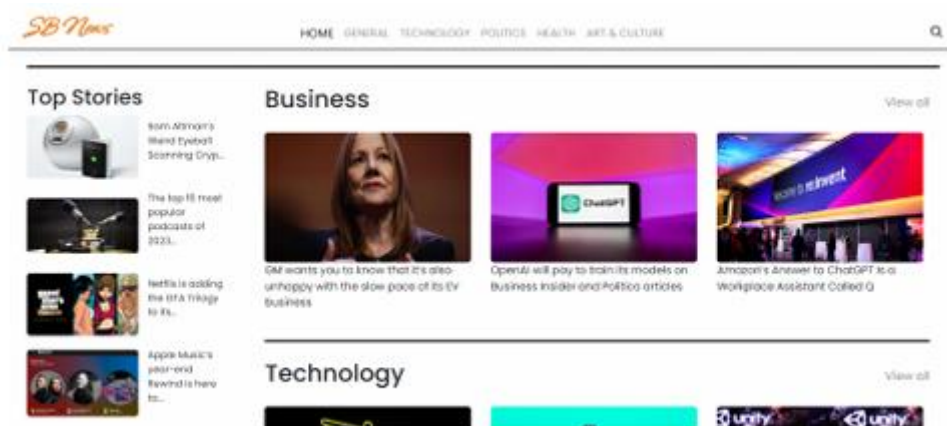- Used within components to handle UI interactions such as toggles and input fields.

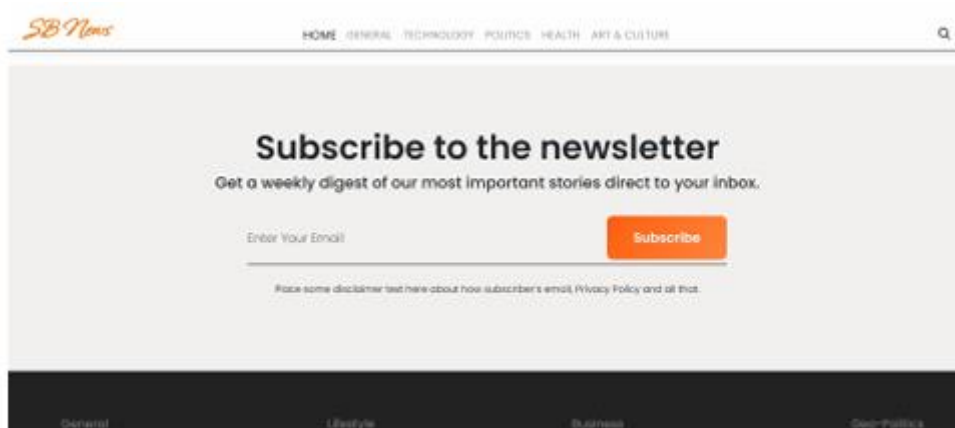## 9. User Interface

Screenshots or GIFs showcasing:

Homepage with trending news

Popular Categories

Subscription form interaction

10. Styling

CSS Frameworks/Libraries

- **Styled Components** for modular and dynamic styling.
- **CSS Modules** for scoped styles in individual components.

Theming

- Dark mode support (if implemented in future versions)

11. Testing

Testing Strategy

- **Jest & React Testing Library** for unit and integration tests.
- **Cypress** (future enhancement) for end-to-end testing.

Code Coverage

- Code coverage reports generated using Jest.

12. GitHub Link

https://github.com/2226060/News-app-SWTID1741256887149395

13. Known Issues

- Performance optimizations needed for large news datasets.
- Improve mobile responsiveness for small-screen devices.

14. Future Enhancements

- Implement dark mode for better accessibility.
- Add a search functionality for articles.
- Improve animation and transitions for a smoother user experience.