

NAAN MUDHALVAN PROJECT

UNIVERSITY OF MADRAS



COLLEGE NAME: AGURCHAND MANMULL JAIN COLLEGE

COLLEGE CODE : 1301

SUBJECT: FRONTEND DEVELOPMENT WITH REACT.JS

TOPIC: INSIGHTSTREAM (NEWS APP)

TEAM	ROLE	UNM ID	EMAIL ID
GOKUL RAJ V(222205208)	Team Leader	unm130122G109	rajgokul436@gmail.com
JEEVA R(222205227)	Team Member	unm130122G108	jeevaraja9790@gmail.com
THAMIZH SELVAN R(222205316)	Team Member	unm130122G107	thamizhselvan985@gmail.com
PRASANTH R(222205271)	Team Member	unm130122G106	prasanthshamini123@gmail.com

Project Documentation

Introduction

Project Title: Insightstream

Team Members:

- GOKUL RAJ V
- JEEVA A
- THAMIZH SELVAN R
- PRASANTH R

Project Overview

Purpose

The News App is a web application designed to provide users with real-time news updates from various sources. It allows users to explore trending news, search for specific topics, and navigate through different categories seamlessly..

Features

- Fetches and displays the latest news articles.
- Category-wise news segregation (e.g., Sports, Technology, Business, etc.).
- Search functionality to find articles based on keywords.
- Responsive design for optimal viewing on different devices.
- Interactive UI using React and Bootstrap.

Architecture

Component Structure

/code

├── public/ # Static assets (logos, manifest, favicon, robots.txt, index.html)

├── src/

| ├── components/ # Reusable UI components

- | | | — **NavbarComponent.jsx**
- | | | — **Footer.jsx**
- | | | — **NewsLetter.jsx**
- | | | — **HomeArticles.jsx**
- | | | — **TopStories.jsx**
- | | | — **Hero.jsx**
- | | — **context/** **# Context API for global state management**
- | | | — **GeneralContext.jsx**
- | | — **pages/** **# Route-based pages**
- | | | — **NewsPage.jsx**
- | | | — **CategoryPage.jsx**
- | | | — **Home.jsx**
- | | — **styles/** **# CSS files for styling components and pages**
- | | — **App.js** **# Root component**
- | | — **index.js** **# Entry point for the React app**
- | | — **reportWebVitals.js** **# Performance monitoring**
- | | — **setupTests.js** **# Test setup**
- | — **package.json** **# Project dependencies and scripts**
- | — **package-lock.json** **# Dependency lock file**
- | — **.gitignore** **# Files to ignore in Git**
- | — **README.md** **# Project documentation**

State Management

The application utilizes React's built-in state management along with Context API to store and manage global state for:

- News articles

- Search results
- Category filters

Routing

React Router is used for navigation between different pages, ensuring a smooth user experience.

Setup Instructions

Prerequisites

- Node.js (version 14.x or higher)
- npm package manager

Installation

1. Clone the repository:
2. `git clone`
3. <https://github.com/Gokul1903/gokul.git>
4. Navigate to the project directory:
5. `cd Insightstream_NaanMudhalvan`
6. Install dependencies:
7. `npm install`

Folder Structure

Client

- `src/`: Contains the source code.
- `components/`: Reusable UI components (e.g., Navbar, NewsCard, Loader).
- `pages/`: Different page components (e.g., Homepage, NewsList, NewsDetails).
- `services/`: API service functions for fetching news data.
- `app/`: Contains the main App component and routing logic..

Running the Application

To start the application locally:

```
npm start
```

This will launch the application at <http://localhost:3000>.

Component Documentation

Key Components

- **Navbar:** Displays navigation links for news categories.
- **Homepage:** Shows trending news articles.
- **NewsList:** Lists all news articles with a brief description.
- **NewsDetails:** Provides full details of a selected news article.
- **SearchBar:** Allows users to search for specific news articles.
- **Loader:** Displays a loading indicator during data fetching.

Reusable Components

- **NewsCard:** Displays a preview of each news article.
- **Button:** Custom-styled button component.

State Management

Global State

Managed using Redux Toolkit, encompassing:

- **News:** State slice for storing news articles.
- **Categories:** State slice for managing different news categories.
- **User Preferences:** State slice for storing user preferences like dark mode, language, etc.

Local State

Handled within components using the `useState` hook for UI-specific states, such as:

- **Search Input:** State for storing user search queries.
- **Modal Visibility:** State for handling modal open/close actions.
- **Dropdown Selection:** State for managing selected filters or categories.

User Interface

The application is styled using Bootstrap and React-Bootstrap for a clean and responsive UI.

Styling

CSS Frameworks/Libraries

- **Tailwind CSS** – Utility-first framework for fast and responsive styling.

Theming

- Implement dark mode and light mode using Tailwind CSS (dark: variant) or Material-UI's theme provider for a customizable user experience.

Testing

Testing Strategy

- Jest and React Testing Library are used for unit and integration testing.
- Code coverage is monitored using Jest's built-in tools.

Code Coverage

- **State Management:** Uses **Redux Toolkit** for global state (news, categories, user preferences) and **useState** for local UI states (modals, search input).
- **Component Structure:** Organized with **React functional components**, API calls using **Redux Thunks**, and reusable UI components for better maintainability.

Screenshots or Demo

Screenshot

<https://drive.google.com/drive/folders/1yfNcThVnfWrWlU8bPlqvtdjaxPSGT4u8>

Live Demo

<https://drive.google.com/file/d/1hqwgKjpbC9rNbrY1q2doozPzx6JEMvfd/view?usp=drivesdk>

Known Issues

- API rate limits might cause occasional delays in fetching news.
- Some categories may not always have articles available.
- Pagination could be improved for better user experience.

Future Enhancements

- Dark mode toggle feature.
- WebSocket-based real-time news updates.

- Improved UI animations.
- User authentication for personalized news feeds.