

Project Documentation

InsightStream: Navigate the News Landscape

Introduction

- **Project Title:** InsightStream: Navigate the News Landscape
- **Team ID:**
- **Team Leader:** Saraswathi S (sivakumarpoomnaganapathi@gmail.com)
- **Team Members:**
 - Pavithra R(pavithrarajkumar09@gmail.com)
 - Poornima V(Poornimav2407@gmail.com)
 - Sarjana P(iamsarjana.2007@gmail.com)

Project Overview

- **Purpose:**

The purpose of this project is to build a modern and user-friendly news application that enables users to stay updated with current events, while delivering accurate and relevant content in a seamless and engaging manner. It focuses on offering a fast, responsive, and intuitive interface for an enhanced user experience, and by utilizing the latest web technologies, the project ensures timely access to information anytime and anywhere.

- **Features:**

- Personalized News Feed
- Real-time Notifications
- Bookmark & Save Articles
- Multimedia Support
- Share Art

Architecture

- **Frontend:** React.js with Bootstrap and Material UI to create a responsive, interactive, and user-friendly interface for reading and navigating news.
- **Backend:** Node.js with Express.js to manage server-side logic, handle API requests, and serve news content efficiently.

- **Database:** MongoDB to store user profiles, preferences, saved articles, feedback, and other application data.

Setup Instructions

- **Prerequisites:**
 - Node.js
 - MongoDB 1
 - Git – React.js
 - Express.js – Mongoose – Visual Studio Code

- **Installation Steps:**

1. Clone the Repository: - Open a terminal or command prompt and navigate to the directory where you want to clone the repository. - Run the command: `git clone https://github.com/[username]/insight-stream.git` - Replace [username] with the actual GitHub username or organization name.
 2. Install Dependencies: - Navigate to the cloned repository directory: `cd insight-stream` - Run the command: `npm install` - This will install all the required dependencies listed in the package.json file.
 3. Configure Environment Variables: - Create a .env file in the root directory of the project. - Add the following environment variables: - `REACT_APP_API_KEY`: The API key for the News API - `REACT_APP_BASE_URL`: The base URL for the API requests - Replace the placeholder values with your actual API key and base URL.
 4. Start the Development Server: - Run the command: `npm start` - This will start the development server, and you can access the application at `http://localhost:3000`
- Environment Variables:

- `REACT_APP_API_KEY`: The API key for the News API

- `REACT_APP_BASE_URL`: The base URL for the API requests

Troubleshooting:

- If you encounter any issues during the installation process, try deleting the `node_modules` directory and running `npm install` again.
- If you encounter any issues with the development server, try checking the console logs for errors or restarting the server.

Folder Structure

Client:

The React application is organized into the following folders:

- components: Reusable UI components used throughout the application
- Header.js: The header component with navigation and search bar
- Footer.js: The footer component - NewsCard.js: The individual news article components
- SearchBar.js: The search bar component
- pages: Components that represent individual pages or routes
- Home.js: The home page component that displays the news feed
- Category.js: The category page component that displays news articles for a specific category
- SearchResults.js: The search results page component that displays news articles matching the search query
- assets: Static assets used in the application
- images: Folder for image assets
- fonts: Folder for font assets - utils: Utility functions and custom hooks used in the project
- api.js: A utility function for making API requests
- helpers.js: A utility function for formatting dates and other helper functions
- context: Context API files for state management
- NewsContext.js: The context API file for managing news articles state Utilities:
- Helper Functions:
- formatDate: A function for formatting dates

- truncateText: A function for truncating text
- Custom Hooks:
 - useNews: A custom hook for fetching news articles
 - useSearch: A custom hook for handling search queries

Running the Application

- **Frontend:**

1. Open terminal → cd client
2. Run npm start
3. Server runs on <http://localhost:3000>

- **Backend:**

1. Open terminal → cd server
2. Run npm start
3. Backend API runs at <http://localhost:5000>

- **Access:**

1. Open a browser
2. Visit <http://localhost:3000>
3. Use InsightStream with backend connected

API Documentation

- **User:**

-/api/user/register – Register a new user

-/api/user/login – User login

- **Projects:**

-/api/projects/create – Create a new project

-/api/projects/:id – Get project by ID

- **Applications:**

-/api/apply – Apply to a project

- **Chats:**

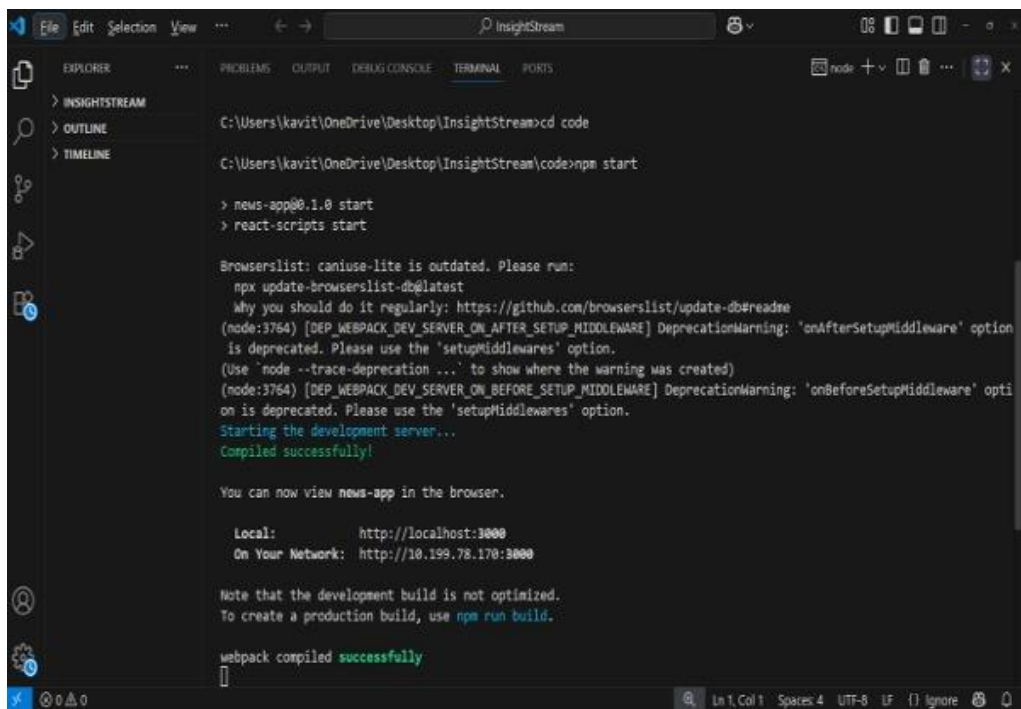
-/api/chat/send – Send a chat message

-/api/chat/:userId – Get chats by user ID

Authentication

- JWT-based authentication for secure login
- Middleware used to protect private routes

User Interface



```
File Edit Selection View ... InsightStream
EXPLORER PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
> INSIGHTSTREAM
> OUTLINE
> TIMELINE

C:\Users\kavit\OneDrive\Desktop\InsightStream>cd code
C:\Users\kavit\OneDrive\Desktop\InsightStream\code>npm start

> news-app@0.1.0 start
> react-scripts start

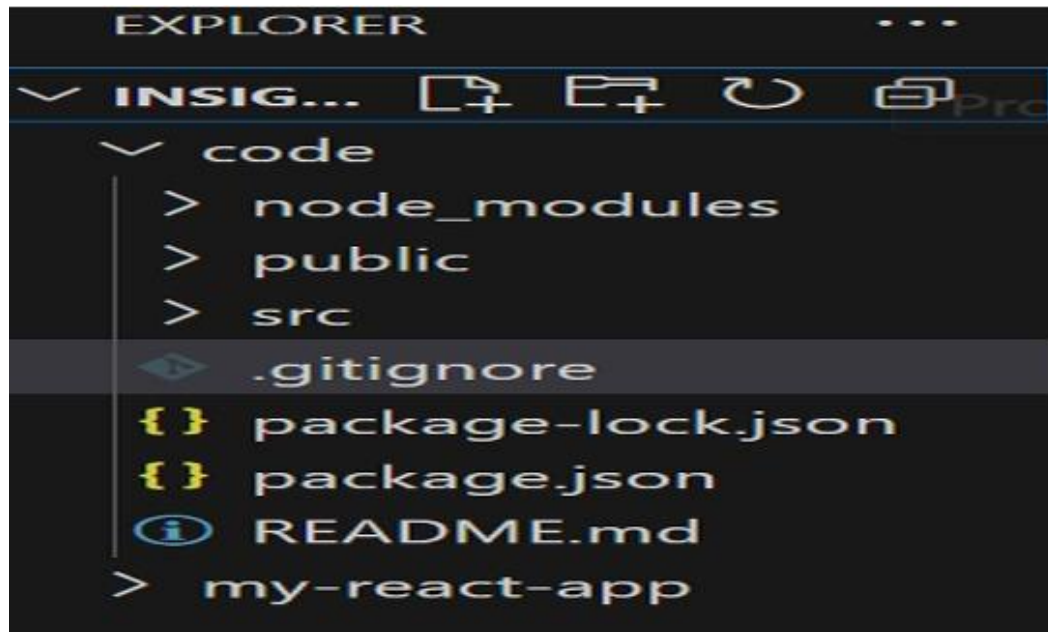
Browserslist: caniuse-lite is outdated. Please run:
  npx update-browserslist-db@latest
  why you should do it regularly: https://github.com/browserslist/update-db#readme
(node:3764) [DEP_WEBPACK_DEV_SERVER_ON_AFTER_SETUP_MIDDLEWARE] DeprecationWarning: 'onAfterSetupMiddleware' option
is deprecated. Please use the 'setupMiddlewares' option.
(Use 'node --trace-deprecation ...' to show where the warning was created)
(node:3764) [DEP_WEBPACK_DEV_SERVER_ON_BEFORE_SETUP_MIDDLEWARE] DeprecationWarning: 'onBeforeSetupMiddleware' opti
on is deprecated. Please use the 'setupMiddlewares' option.
Starting the development server...
Compiled successfully!

You can now view news-app in the browser.

  Local:            http://localhost:3000
  On Your Network:  http://10.199.78.170:3000

Note that the development build is not optimized.
To create a production build, use npm run build.

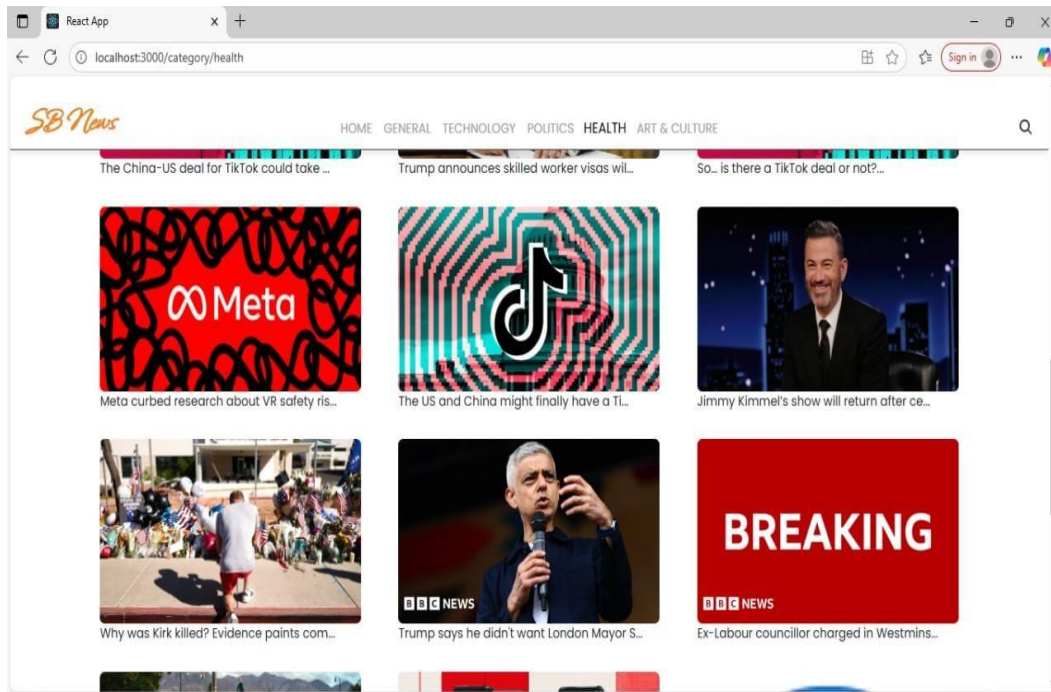
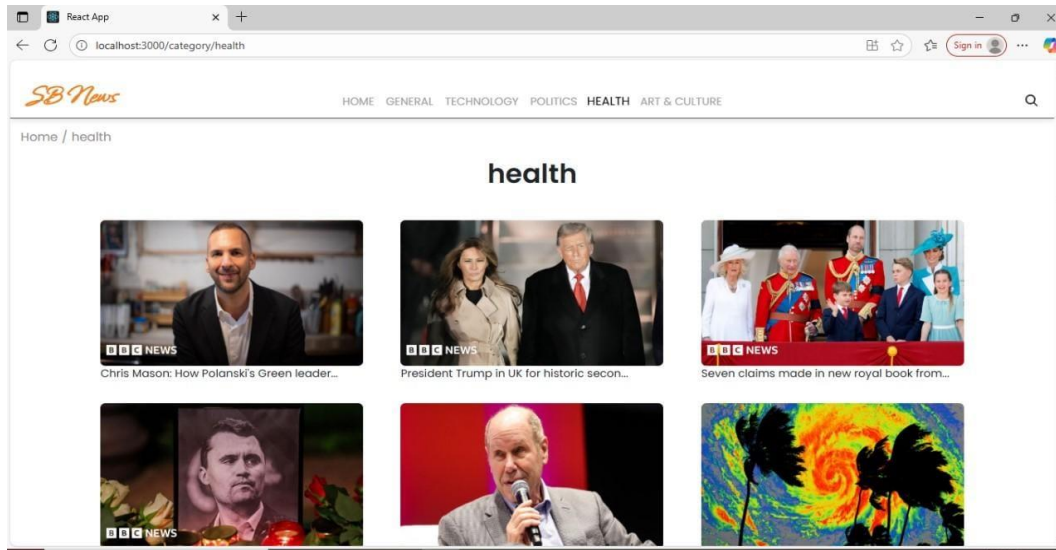
webpack compiled successfully
```

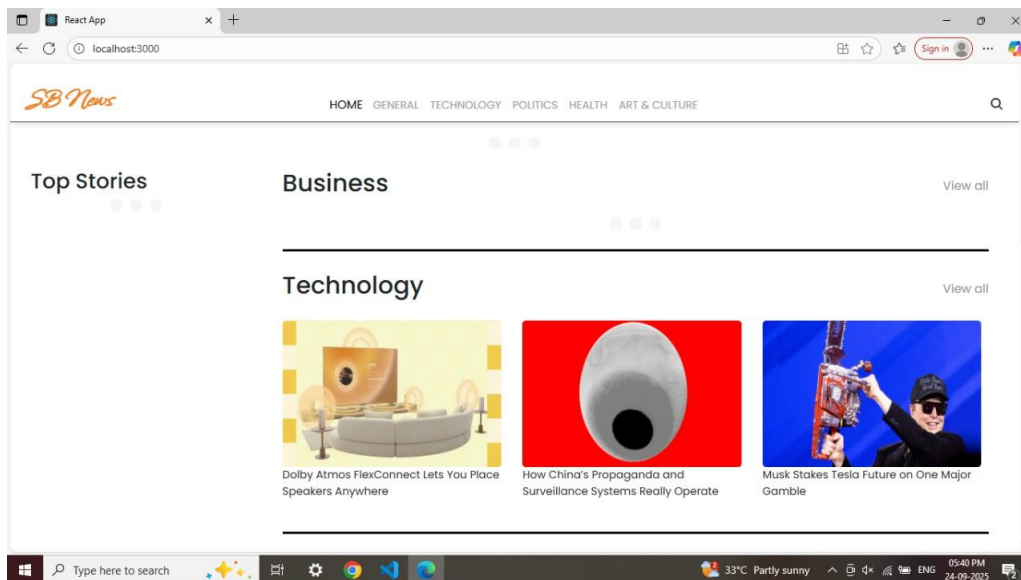
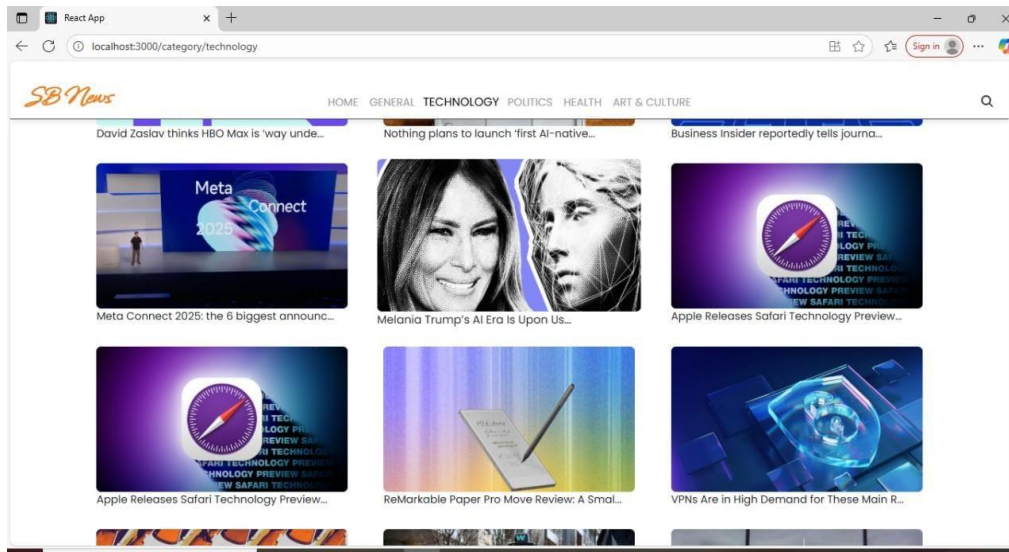


Testing

- Manual testing at each milestone
- Tools: Postman, Chrome DevTools

Screenshots





Known Issues

- **Loading Delays:** Some news articles or multimedia content may take longer to load during peak traffic periods or with slow internet connections, affecting the user experience.
- **Offline Reading Limitations:** The offline reading feature may not function properly for all types of content, such as videos or dynamically loaded articles, limiting its usability.
- **Notification Delays:** Real-time notifications may experience slight delays under certain network conditions or server load, which could result in users receiving breaking news alerts later than expected.
- **Search Accuracy:** The search functionality may occasionally return less relevant results for niche topics or newly published articles, making it harder for users to find specific news items.
- **UI/UX Inconsistencies:** Minor user interface inconsistencies may appear on different screen sizes, devices, or browsers, affecting layout and readability.
- **User Profile Sync Issues:** Changes to user preferences, saved articles, or bookmarks may not sync instantly across multiple devices in some cases.
- **Backend Limitations:** Under heavy load, the backend server may experience slower response times, impacting the speed of content delivery and chat features.

Future Enhancements

- **AI-Powered Personalization:** Implement machine learning algorithms to provide more accurate and dynamic content recommendations based on user behavior and preferences.
- **Multi-Language Support:** Add support for multiple languages to cater to a wider, global audience.
- **Advanced Search & Filters:** Enhance search functionality with filters for categories, date ranges, sources, and trending topics.
- **Push Notifications & Alerts:** Improve notification system to include personalized alerts for breaking news and favorite topics.
- **Offline Mode Expansion:** Extend offline reading to include multimedia content such as videos and interactive infographics.
- **Dark Mode & Accessibility:** Introduce customizable themes and accessibility features for better readability and user comfort.
- **Social Media Integration:** Enable seamless sharing of news articles across multiple social media platforms.
- **Analytics Dashboard:** Provide users and admins with insights into reading habits, popular content, and engagement metrics.