**Project Documentation**

### Project Title: InsightStream: Navigate the News Landscape

**1. Introduction**

\* **Project Title:** insight stream: Navigate the news landscape

\***Team ID**: NM2025TMID40697

\***Team Leader**: Rama lakshmi M

& ramalakshmiramalakshmi7025@gmail.com

\* **Team Members:**

1.saranya B

2.sadhana ER

3.oviya K

4.vaidheeswari S

2. **Project Overview**

- \*\*Purpose:\*\* SB News is a modern news streaming application that aggregates and displays real-time news articles from various categories using the NewsAPI. It provides users with an intuitive interface to browse top stories, search for specific topics, and view detailed articles.

- \*\*Features:\*\*

- Hero section showcasing featured news articles

- Top stories sidebar with trending headlines

- Categorized news sections (e.g., Business, Technology, Politics)

- Search functionality for custom queries

- Category pages with pagination for browsing articles

- Newsletter subscription form for user engagement

- Detailed news article pages

- Responsive design with hover effects and smooth transitions

**3. Architecture**

- \*\*Frontend:\*\* React.js with Bootstrap and React-Bootstrap for styling and responsive layouts, React Router for navigation, Axios for API requests, and React Icons for UI elements. Custom CSS handles specific styling like gradients and animations.

- \*\*Backend:\*\* None (client-side only; data fetched directly from third-party NewsAPI).

- \*\*Database:\*\* None (data sourced from NewsAPI; no local storage implemented).

**4. Setup Instructions**

- \*\*Prerequisites:\*\*

- Node.js

- Git

- Visual Studio Code (or any code editor)

- \*\*Installation Steps:\*\*

# Clone the repository

git clone <repository-url>

# Install dependencies

cdsb-news

npm install

**5. Folder Structure**

SB-News/

|-- src/

| |-- components/ # Reusable UI components

| | |-- Hero.jsx

| | |-- TopStories.jsx

| | |-- HomeArticles.jsx

| | |-- NewsLetter.jsx

| | |-- NavbarComponent.jsx

| | |-- Footer.jsx

| |-- pages/ # Main application pages

| | |-- Home.jsx

| | |-- CategoryPage.jsx

| | |-- NewsPage.jsx

| |-- context/ # Context providers for state management

| | |-- GeneralContext.jsx

| |-- styles/ # CSS files for styling

| | |-- App.css

| | |-- index.css

| | |-- Navbar.css

| | |-- Hero.css

| | |-- TopStories.css

| | |-- HomeArticles.css

| | |-- NewsLetter.css

| | |-- CategoryPage.css

| | |-- NewsPage.css

| | |-- Footer.css

| | |-- Home.css

| |-- App.js # Main app entry with routing

| |-- index.js # React DOM rendering

| |-- reportWebVitals.js # Performance reporting

| |-- App.test.js # Unit tests

|-- .gitignore

|-- package.json

|-- README.md (optional)

**6. Running the Application**

- \*\*Frontend:\*\*

cdsb-news

npm start

- \*\*Backend:\*\* N/A (no backend required)

- \*\*Access:\*\* Visit http://localhost:3000

**7. API Documentation**

- \*\*Third-Party API:\*\* NewsAPI (https://newsapi.org/)

- Endpoints Used:

- /v2/everything?q=<query>&apiKey=<key> (for searches and categories like politics)

- /v2/top-headlines?category=<category>&country=us&apiKey=<key> (for top headlines in categories like business, technology)

- \*\*Note:\*\* Replace <key> with your NewsAPI key in the code or via environment variables (e.g., REACT\_APP\_API\_KEY in .env).

**8. Authentication**

- None implemented

(public news app; no user login required).

**9. User Interface**

- \*\*Home Page:\*\* Features Hero section, Top Stories, categorized articles (Business, Technology, Politics), and Newsletter subscription.

- \*\*Category Page:\*\* Displays articles for a specific category with breadcrumb navigation and pagination.

- \*\*News Page:\*\* Detailed view of a single article with image, content, source, and link to full article.

- \*\*Navbar:\*\* Includes logo, category links, and search bar.

- \*\*Footer:\*\* Category links and copyright information.

**10. Testing**

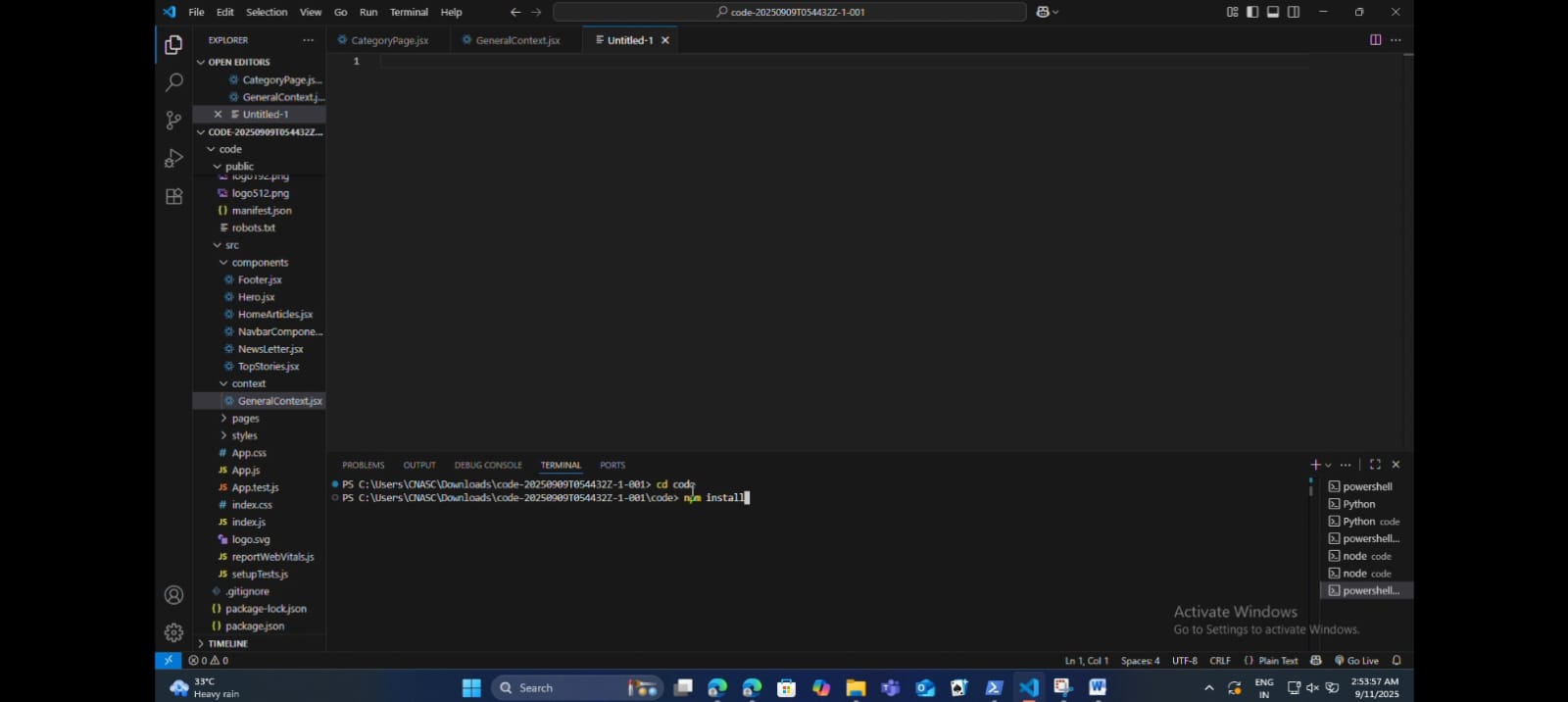
- \*\*Manual Testing:\*\* Browser testing for UI responsiveness and API fetches.

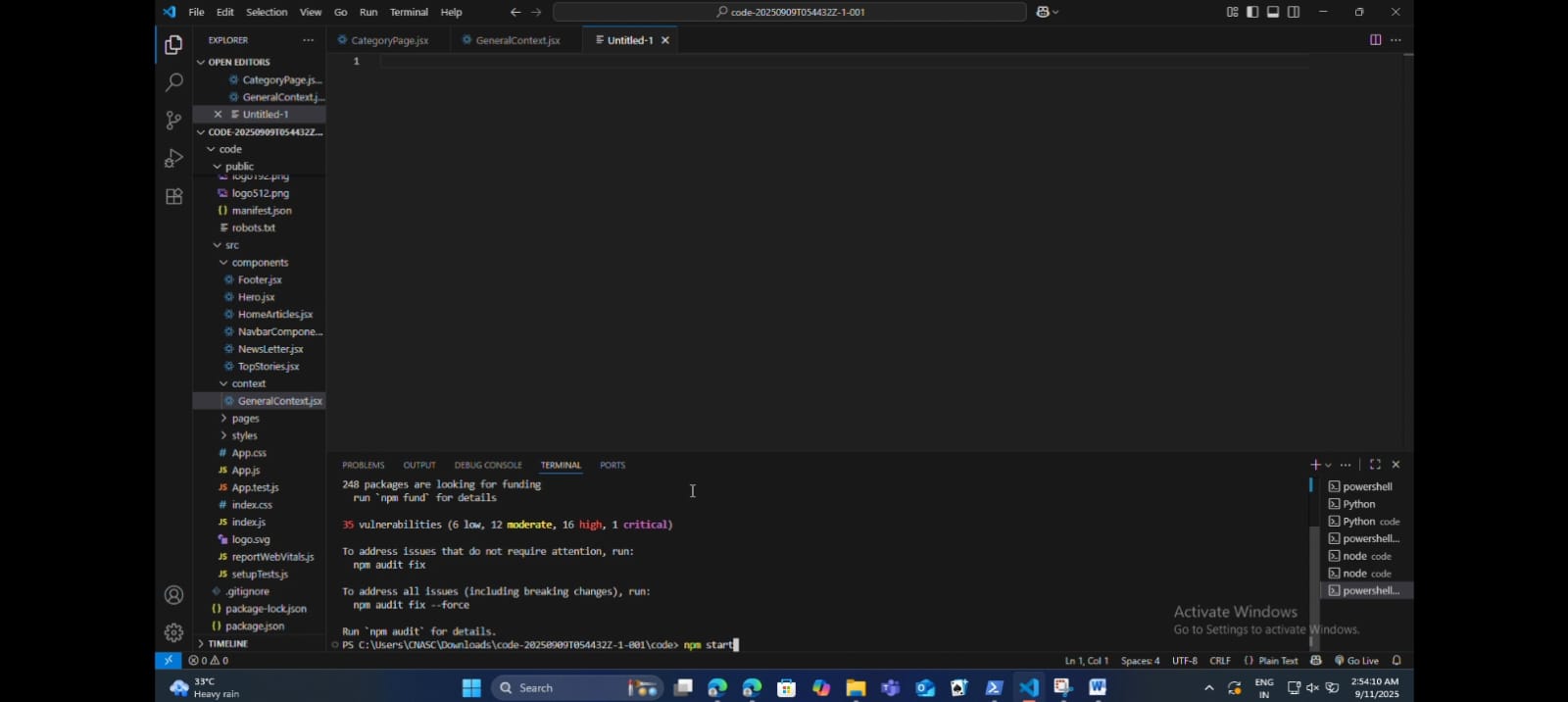
- \*\*Unit Testing:\*\* Basic tests in App.test.js using @testing-library/react.

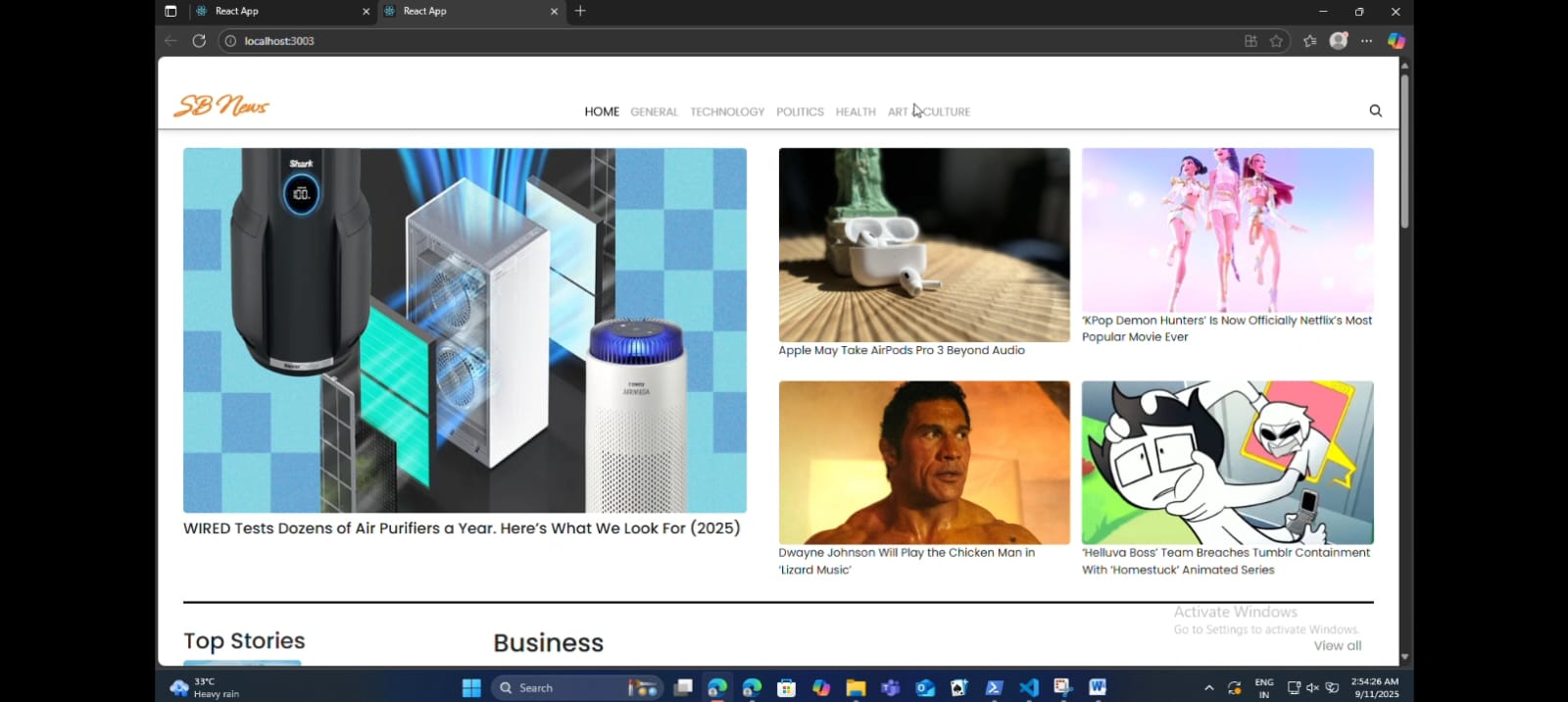
- \*\*Tools:\*\* Chrome Dev Tools, React Developer Tools.

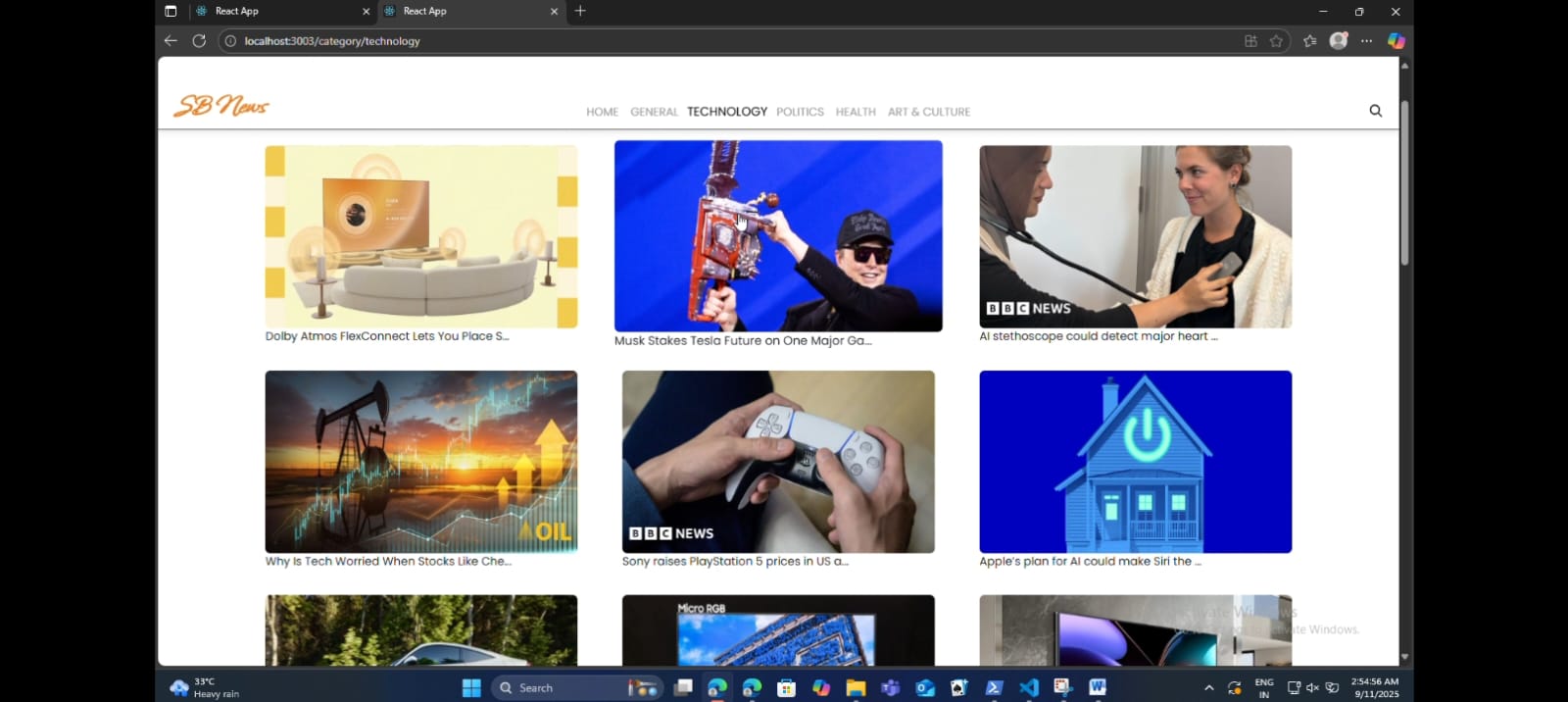
**11. Screenshots or Demo**

- (Placeholder - Add screenshots of Home Page, Category Page, etc., or link to a demo deployment on Vercel/Netlify.)









**12. Known Issues**

- API rate limiting from NewsAPI (free tier has daily limits).

- No offline support or caching for news articles.

- Search may return broad results; refinement needed for exact matches.

**13. Future Enhancements**

- Integrate user authentication for personalized news feeds.

- Add backend (Node.js/Express) for storing user preferences or custom newsletters.

- Implement infinite scrolling instead of pagination.

- Add dark mode toggle.

- Integrate more APIs (e.g., for weather or stocks) for expanded categories