PROJECT DOCUMENTATION

InsightStream: Navigate the News Landscape

**1. Introduction**

**Project Title:** InsightStream: Navigate the News Landscape  
**Team ID:** NM2025TMID30075  
**Team Leader:** RAMYA M & [202400137@sigc.edu](mailto:202400137@sigc.edu)

**Team Members:**

* RAJESWARI R & [202400605@sigc.edu](mailto:202400605@sigc.edu)
* RISHIKA T & [202400504@sigc.edu](mailto:202400504@sigc.edu)
* YOGESWARI K & [202400239@sigc.edu](mailto:202400239@sigc.edu)

**2. Project Overview**

**Purpose:**  
InsightStream is a modern news aggregator web application designed to help users efficiently discover and consume diverse news content from various reliable sources. It centralizes multiple news feeds, enhances accessibility, and promotes media literacy for a broad user base.

**Features:**

* Real-time news aggregation from multiple sources (external APIs)
* Advanced search and category filtering
* Personalized bookmarks and reading history
* Secure user authentication and profiles
* Responsive, accessible UI (desktop/mobile)
* Admin control panel for content moderation
* Feedback and reporting system

**3. Architecture**

* **Frontend:** React.js with Bootstrap and Material UI for dynamic, responsive interfaces
* **Backend:** Node.js and Express.js handling API requests, authentication, and business logic
* **Database:** MongoDB for storing user data, preferences, bookmarks, and news metadata
* **State Management:** Context API (global state), React hooks (local state)
* **External APIs:** Integration with news APIs (e.g., NewsAPI, RapidAPI)

**4. Setup Instructions**

**Prerequisites:**

* Node.js
* MongoDB
* Git
* React.js, Express.js
* News API keys
* Visual Studio Code

**Installation Steps:**

shell

git clone [repository-url]

cd client

npm install

cd ../server

npm install

Set up .env with API keys for news sources.

**5. Folder Structure**

text

InsightStream/

|-- client/

| |-- components/

| |-- pages/

| |-- assets/

| |-- utils/

|-- server/

| |-- routes/

| |-- models/

| |-- controllers/

**6. Running the Application**

**Frontend:**

shell

cd client

npm start

**Backend:**

shell

cd server

npm start

**Access:** Visit [http://localhost:3000](http://localhost:3000/) in the browser.

**7. API Documentation**

**User:**

* POST /api/user/register
* POST /api/user/login

**News:**

* GET /api/news/fetch
* GET /api/news/:id

**Bookmarks & History:**

* POST /api/bookmark
* GET /api/history

**Admin:**

* POST /api/admin/moderate

**8. Authentication**

* Mechanism: JWT-based authentication secures user sessions.
* Protection: Middleware guards all private routes and restricts admin functions.

**9. User Interface**

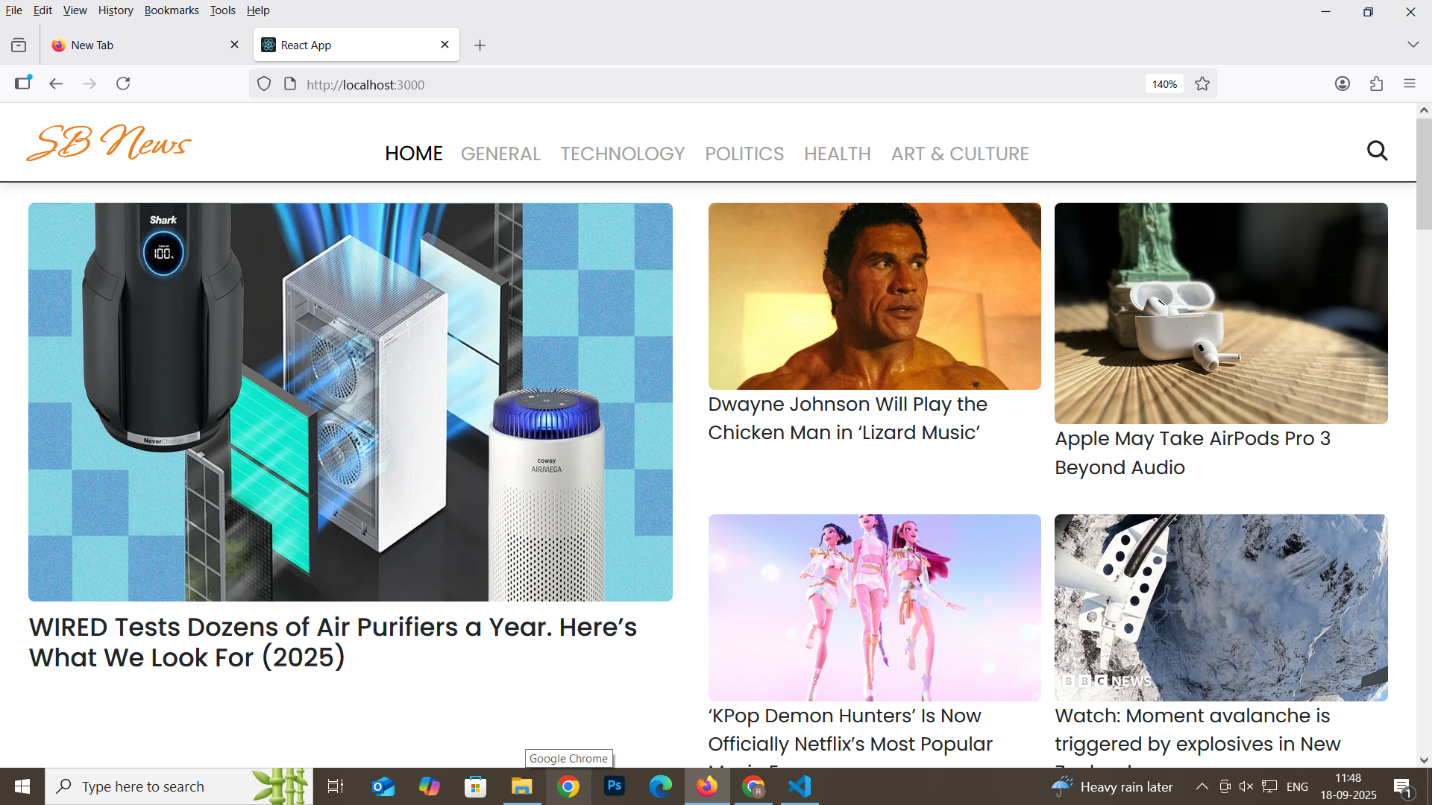
* Landing Page (trending news summary)
* News Categories & Search
* Article Detail Page
* User Bookmarks & History
* Admin Dashboard

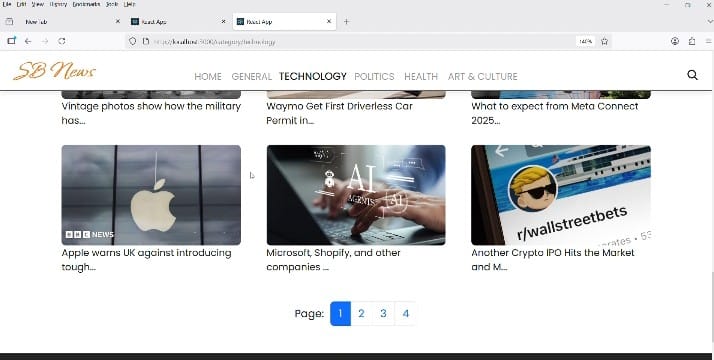
**10. Testing**

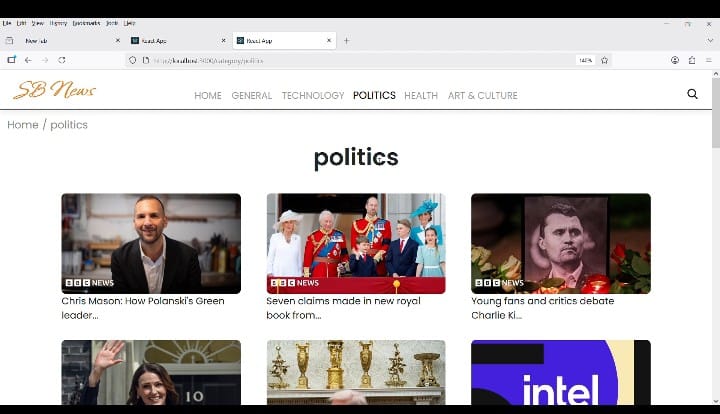
* Manual Testing: Test major flows and edge cases at each development milestone.
* Tools Used: Postman (API testing), Chrome Dev Tools (UI/UX, frontend console errors).

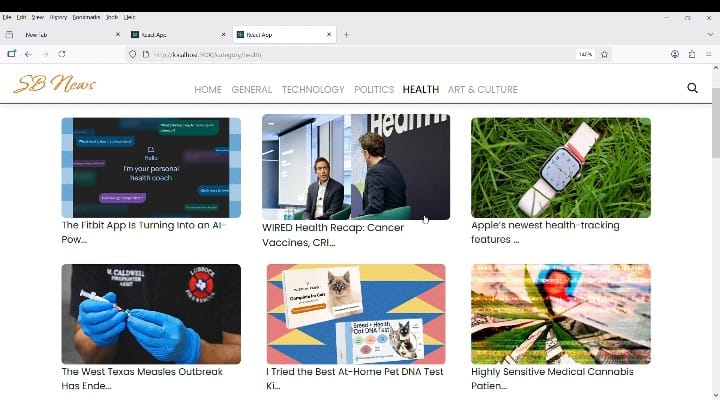
**11. Screenshots or Demo link**

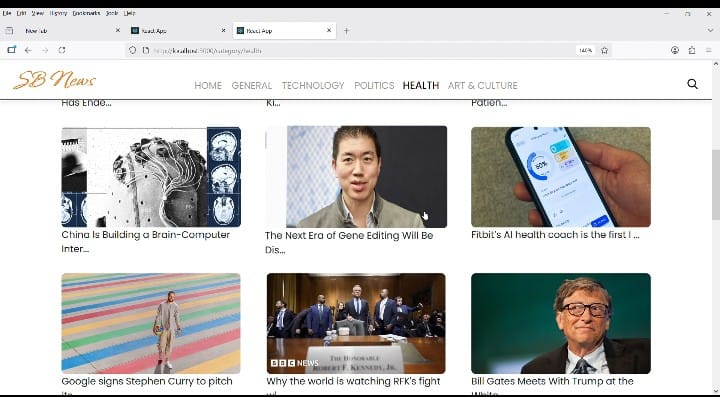
Live Demo: [**https://drive.google.com/file/d/1ZZawgkWd17rd68x\_U7LmvGqcOJVoAOzc/view?usp=drivesdk**](https://drive.google.com/file/d/1ZZawgkWd17rd68x_U7LmvGqcOJVoAOzc/view?usp=drivesdk)

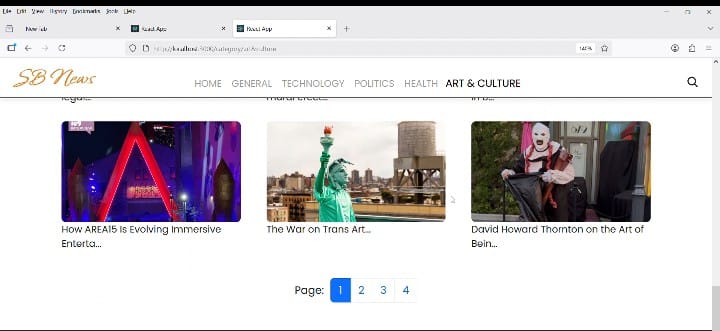


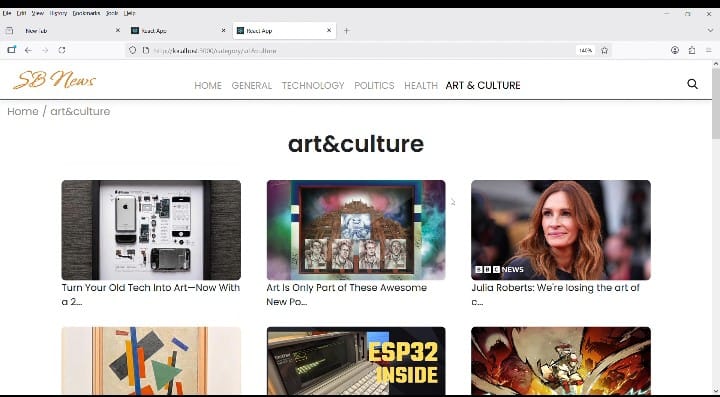












**12. Known Issues**

* Limited API quota (free plans)
* Some design elements may not render identically on all browsers
* Bookmark feature requires optimization for large user data sets

**13. Future Enhancements**

* Push notifications for breaking news
* In-app social sharing options
* AI-based news recommendation engine
* Mobile app version
* More granular filtering (by region, language)