- Project Documentation

Project Title

Insight Stream: Navigate the News Landscape

---

1. Introduction

Project Title: Insight Stream – Navigate the News Landscape

Team ID: NM2025TMID48372

Team Leader: S.Pooja sri & poojasri10072006@gmail.com
Role: Coding and development
Team Members:
N.Priya & priyanarayananpn5@gmail.com
Role: Document creater
K.Priyadharshini & priyadharshinikumaresan2706@gmail.com
Role: Demo video
P.Priyadharshini & pp692819@gmail.com
Role: Document
S.Priyadharshini & subaramani111973@gmail.com
Role: Document

---

2. Project Overview

Purpose:
Insight Stream is a news aggregation and analysis platform designed to help users easily navigate the overwhelming digital news landscape. It collects news from trusted sources, categorizes content, provides summaries, and applies sentiment analysis to deliver insightful perspectives.

Features:

Real-time news aggregation from multiple APIs

Smart categorization (Politics, Sports, Tech, Health, etc.)

AI-driven summarization for quick reading

Sentiment analysis and trend visualization

Personalized recommendation engine

User dashboard with bookmarks and history

---

3. Architecture

Frontend: React.js with TailwindCSS / Material UI

Backend: Node.js + Express.js (handles API requests, news data processing)

Database: MongoDB (stores users, preferences, bookmarks, and aggregated news)

External APIs: News API, Google News, or other RSS feeds for news sources

AI/ML: Natural Language Processing (NLP) for summarization & sentiment analysis

---

4. Setup Instructions

Prerequisites:

Node.js

MongoDB

Git

React.js

Express.js

Visual Studio Code

Installation Steps:

```
# Clone the repository
git clone <repo-link>

# Install client dependencies
cd client
npm install

# Install server dependencies
cd ../server
npm install
```

---

5. Folder Structure

```
InsightStream/
|-- client/       # React frontend
|   └── components/
|   └── pages/
|-- server/       # Node.js backend
|   └── routes/
|   └── models/
|   └── controllers/
|-- database/      # MongoDB collections
```

---

6. Running the Application

Frontend:

```
cd client
npm start
```

Backend:

```
cd server
npm start
```

Access: Visit http://localhost:3000

---

7. API Documentation

User:

POST /api/user/register

POST /api/user/login

News:

GET /api/news/latest

GET /api/news/category/:type

GET /api/news/trends

User Dashboard:

POST /api/bookmark

GET /api/bookmarks/:userId

---

8. Authentication

JWT-based authentication for secure login

Middleware protects private routes

---

9. User Interface

Landing Page

News Dashboard (categories, trending, sentiment)

Personalized Feed Page

Bookmark Manager

Admin Panel

---

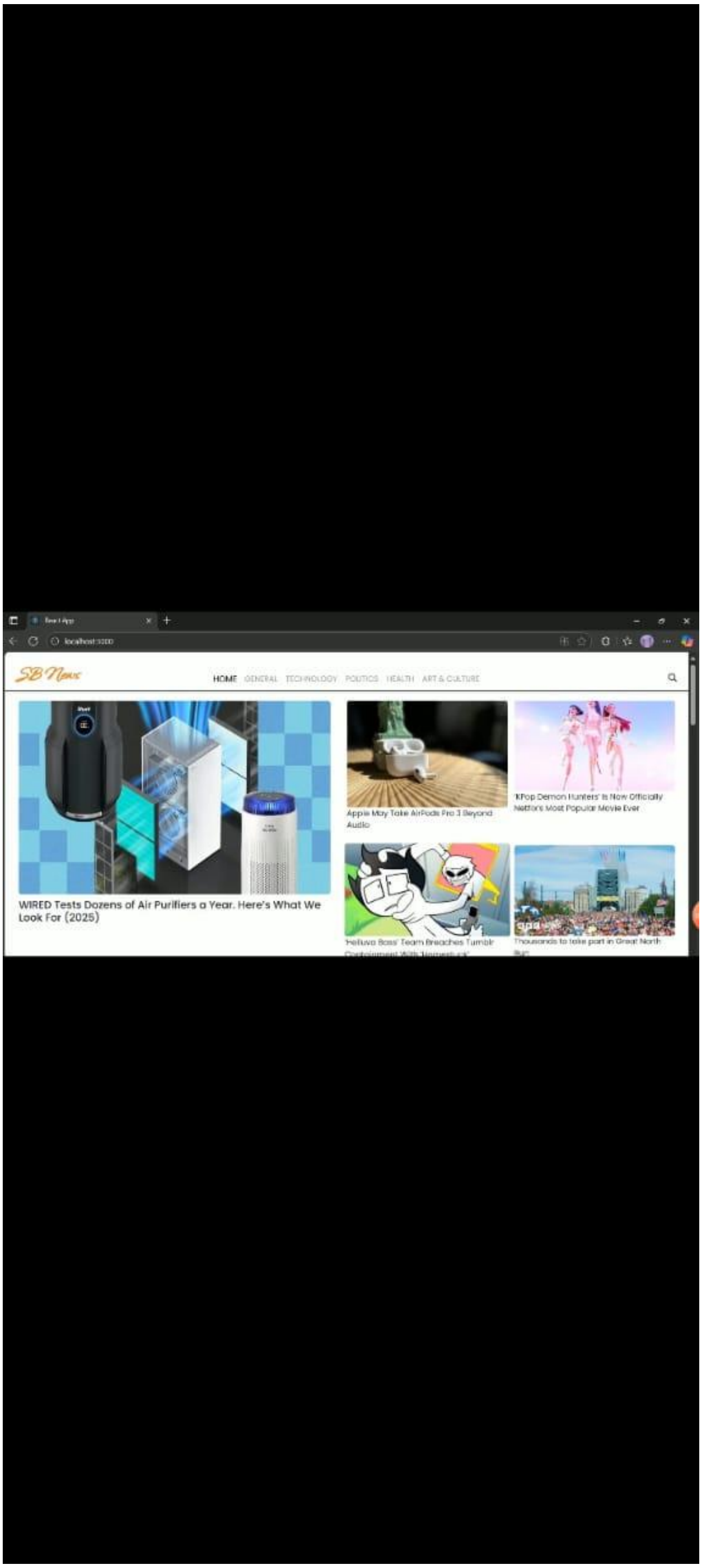
10. Testing

Manual Testing: During development milestones

Tools: Postman, Chrome Dev Tools, Jest (for frontend unit tests)

---

11. Screenshots or Demo



WIRED Tests Dozens of Air Purifiers a Year. Here's What We Look For (2025)

Apple May Take AirPods Pro 3 Beyond Audio

'KPop Demon Hunters' Is Now Officially Netflix's Most Popular Movie Ever

'Helluva Boss' Team Breaches Tumblr Containment With 'Homestuck'

Thousands to take part in Great North Run

---

12. Known Issues

Limited free-tier access to news APIs

Summarization accuracy may vary depending on content length

---

13. Future Enhancements

Multilingual news summarization

Voice-based search and accessibility features

Advanced AI-based bias detection in news sources

Mobile app version (React Native / Flutter)