**INSIGHT STREAM NAVIGATE THE NEWS LANDSCAPE**

**1.INTRODUCTION**

* Project Title: Insight Stream Navigate the News Landscape
* Team ID**:** NM2025TMID37897
* Team Leader**:** AARTHISRI B Mail id: [aarthisreeja6@gmail.com](mailto:aarthisreeja6@gmail.com)
* Team Members**:** DIVYA DARSHINI P Mail id: [sasikalaperumal2004@gmail.com](mailto:sasikalaperumal2004@gmail.com)

LAVANYA S Mail id: [lav032006@gmail.com](mailto:lav032006@gmail.com)

DEEPIKA V Mail id: [deepikavasudevann@gmail.com](mailto:deepikavasudevann@gmail.com)

**2. PROJECT OVERVIEW**

Purpose:

* Insight Stream provides students and readers with a reliable way to monitor, filter, and understand the news landscape. It connects them with verified educational updates, exam notices, opportunities, and policy insights in an accessible manner.

Features:

* Real-time news monitoring and alerts

* Categorization (Urgent / Useful / Shareable)

* Quick explainers and flash updates

* Weekly roundup of top highlights

* Multi-language support (English & Tamil)

* Admin dashboard for content management

**3.ARCHITECTURE**

* Frontend: React.js with Bootstrap and Material UI

* Backend: Node.js and Express.js managing server logic and API endpoints

* Database: MongoDB to store user profiles, news articles, tags, alerts, and admin logs

**4. SETUP INSTRUCTIONS**

* Prerequisites:

* Node.js

* + MongoDB

* + Git

* + React.js

* + Express.js

* + Mongoose

* + Visual Studio Code

* Installation Steps:

# Clone the repository git clone

# Install client dependencies cd client npm install

# Install server dependencies cd ../server npm install

**5. FOLDER STRUCTURE**

Insight Stream/

|-- client/ # React frontend

|-- components/

|-- pages/

|-- server/ # Node.js backend

|-- routes/

|-- models/

|-- controllers/

**6. RUNNING THE APPLICATION**

* Frontend:

* cd client npm start

* Backend:

* cd server npm start

* Access: Visit <http://localhost:3000>

**7. API DOCUMENTATION**

* User:

* + - /api/user/register

* + - /api/user/login

* News:

* + - /api/news/create

* + - /api/news/:id

* Alerts:

* + - /api/alerts/create

* + - /api/alerts/:id

* Admin:

* /api/admin/manage

**8. AUTHENTICATION**

* JWT-based authentication for secure login

* Middleware to protect admin routes and user dashboards

**9. USER INTERFACE**

* Landing Page with featured news

* Student Dashboard (customized news feed)

* Admin Panel for news/article approval

* News Detail Page (with related updates & sources)

**10. TESTING**

* Manual testing during development milestones

* Tools: Postman for API testing, Chrome Dev Tools for frontend testing

**11. SCREENSHOTS OR DEMO**

<https://drive.google.com/file/d/1REFmj9D4a0jl-PotAXBh2_kO8PQaMqk2/view?usp=drivesdk>

**12. KNOWN ISSUES**

* Delay in syncing multilingual content

* Occasional duplicate news from overlapping sources

**13. FUTURE ENHANCEMENTS**

* AI-powered summarization of articles

* Personalized feeds based on user interests

* Mobile app with push notifications

* Integration with official exam portals for auto-updates