

# Project Documentation

## Store Manager: Keep track of inventory

### 1.Introduction.

Project Title: Insight stream: Navigate the news landscape

Team ID : NM2025TMID42353

Team Leader :M.PREETHI & Mail ID: preethiprincess8383@gmail.com

Team member :

A.PUSHPA & Mail ID: pushpasubu51@gmail.com

M.SARASWATHI & Mail ID:saraswathij203@gmail.com

L.SENTHAMILSELVI & Mail ID: senthamilselvilaksmanan@gmail.com

### 2. Project Overview

- To create a platform that curates, analyzes, and delivers reliable insights from diverse information sources, helping users navigate complex data and news landscapes with clarity.
- Features:
  - Real-time news and data feeds
  - Customizable insight streams by topic or seacor
  - Bias and credibility indicators
  - Summarized key takeaways with references
  - Cross-platform accessibility (web and mobile)

### 3. Architecture

- a. Frontend: HTML/CSS/JavaScript (or react) user interface for viewing and managing products.
- b. Backend: Node.js with Express handles API requests and business logic.
- c. Database: MongoDB/MySQL stores product details, stock levels, and transactions.

### 4. Setup Instructions

#### a. Prerequisites:

- Install Node.js LTS from nodejs.org and verify with `node -v / npm -v`.
- Windows only: In PowerShell (Admin) run `Set-ExecutionPolicy Unrestricted` and confirm.
- Install Visual Studio Code or another code editor.
- Download and extract the Store Manager project code.

#### • Installation Steps:

- Install Node.js LTS from nodejs.org and verify with `node -v / npm -v`.
- (Windows only) In PowerShell (Admin) run `Set-ExecutionPolicy`

- Unrestricted→ press Y. Download & Extract the Store Manager project folder.
- Open the folder in VS Code or terminal: `cd insight-stream`. Run `npm install` to install dependencies. Start the app with `npm start` and open `http://localhost:3000` in your browser.

## 5. Folder Structure

Insight-stream/

```

├── package.json    # Project info & dependencies
├── README.md       # Documentation
├── public/ ─── # Static assets (index.html, images, css)
src/ ─── server/ # Frontend code (components, styles)
    └── database/  # Backend (app.js, routes, models, controllers)
                        # Seed or migration scripts

```

## 6. Running the Application

- Install Node.js & MongoDB.
- `npm install`
- Create `.env` with DB URL & POR
- `npm start`
- Open `http://localhost:3000` in browser.

## 7. API Documentation

a. User:

- `/api/user/register`
- `/api/user/login`

b. Projects:

- `/api/projects/create`
- `/api/projects/:id`

c. Applications:

- `/api/apply`

d. Chats:

- `/api/chat/send`
- `/api/chat/:userId`

## 8. Authentication

- Register/Login Users using /auth/register and /auth/login endpoints with hashed passwords.
- Generate JWT Token on login for secure access.
- Protect Inventory Routes using middleware that verifies the JWT before allowing CRUD operations.

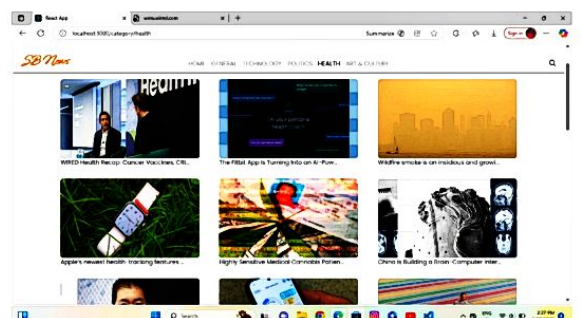
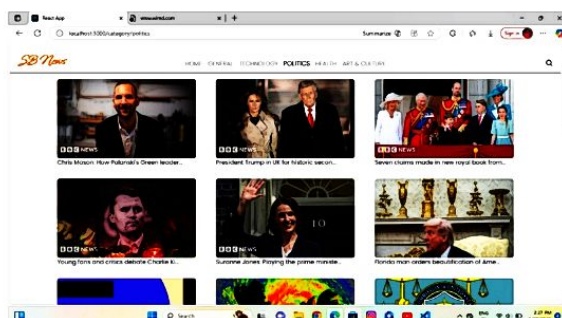
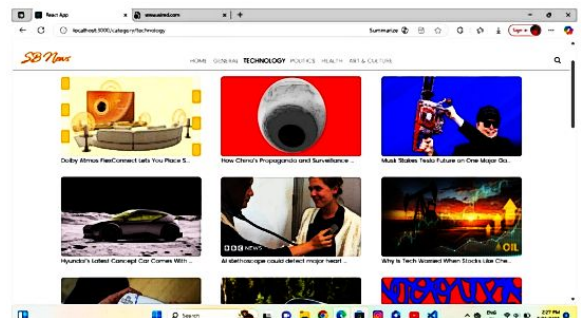
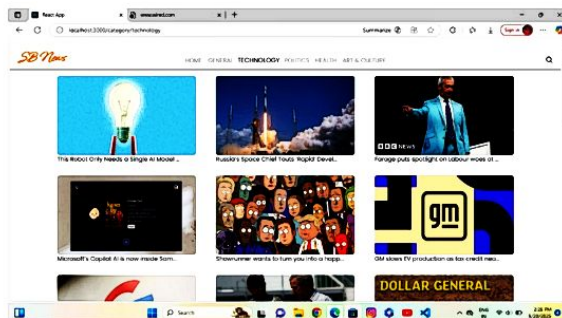
## 9. User Interface

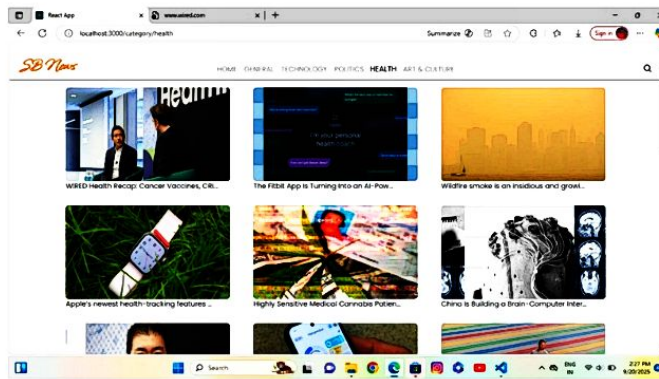
1. Live Data Feed – incoming metrics, logs, or events.
2. Insight Cards/Widgets – AI-driven highlights, anomalies, or patterns extracted.
3. Filtering & Controls – search, filter, and customize what kind of insights you see.
4. Visualization – charts, heatmaps, timelines for context.

## 10. Testing

1. Functional Testing – check login, data flow, and correct insights shown
2. Performance Testing – test speed, load handling, and response time.
3. Security Testing – verify authentication, authorization, and data protection.
4. Integration Testing – ensure data sources, stream, and dashboard work together.
5. User Testing – confirm insights are useful and UI is easy to use.

## 11. Screenshots or Demo





## 12. Known Issues

- No role-based access control; all users have full permissions.
- Limited input validation; negative quantities or prices may be accepted.

## 13. Future Enhancements

- Add role-based access (admin/staff).
- Enable real-time stock alerts.
- Generate inventory reports.
- Make UI mobile-friendly.
- Support barcode/QR code scanning.