Frontend Development with React.js

# Project Documentation

## Introduction

Project Title: INSIGHT STREAM- Navigate the News Landscape

Team Leader: D.V.GOPIKA

Team Members:

V.MADHUMITHA

S.MAHALAKSHMI

M.JAYASHREE

T.INNISA

## Project Overview

Purpose: InsightStream is a modern web application designed to help users seamlessly navigate the landscape of news. It fetches real-time articles from APIs and organizes them by categories with advanced search and filtering. The goal is to improve user experience and enable easy access to reliable information.

Features:

* - News search and filtering
* - Category-based browsing
* - User login and authentication
* - Bookmark and save articles
* - Admin control panel

## Architecture

Component Structure:

The frontend is built using React.js. Major components include Header, Home, Search, NewsList, NewsCard, and Dashboard. Pages are organized using React Router.

State Management: Context API is used to handle global states like authentication, news data, and bookmarks.

Routing: React Router v6 is used for navigation between Landing Page, Categories, Dashboard, and Admin Panel.

## Setup Instructions

Prerequisites: Node.js, MongoDB, Git, VS Code

Installation:

1. 1. Clone the repository
2. 2. cd client && npm install
3. 3. cd ../server && npm install
4. 4. Configure environment variables in .env
5. 5. Run npm start for both frontend and backend

## Folder Structure

InsightStream/  
 ├── client/ # React frontend  
 │ ├── src/  
 │ │ ├── components/  
 │ │ ├── pages/  
 │ │ ├── assets/  
 │ │ └── App.js  
 ├── server/ # Node.js backend  
 │ ├── routes/  
 │ ├── controllers/  
 │ ├── models/

## Running the Application

Frontend: cd client && npm start

Backend: cd server && npm start

Access the application at http://localhost:3000

## Component Documentation

Key Components:

* - Header: Navigation bar for site-wide links
* - Home: Displays latest news
* - Search: Handles keyword-based news search
* - NewsCard: Reusable component to display individual articles
* - Dashboard: User page for bookmarks and saved items

Reusable Components: Navbar, Footer, Buttons, Input fields

## State Management

Global State: Managed with Context API for authentication and news data

Local State: Managed using useState and useEffect hooks within components

## User Interface

The application provides:

* - Landing Page with top news
* - Search Results Page
* - User Dashboard
* - Admin Panel

Screenshots can be added in this section.

## Styling

CSS Frameworks/Libraries: Bootstrap and Material UI

Theming: Custom theming for header, dashboard, and admin panel

## Testing

Testing Strategy: Manual and unit testing with Jest and React Testing Library

Code Coverage: Basic coverage ensured for components and API calls

## Screenshots or Demo

Screenshots of Landing Page, Dashboard, Search Results, and Admin Panel should be inserted here.

## Known Issues

- Limited personalization features

- UI requires further responsiveness improvements

## Future Enhancements

- AI-based personalized news recommendations

- Mobile app version with push notifications

- Multi-language news support

- Social media integration