

Project Documentation

A5B news

1. Introduction

- **Project Title:**
- **Team ID** **Team Leader:** Umasankari.G (hyntaff@gmail)
- **Team Members:**
 1. Bharathi.V (bharathiv714@gmail.com)
 2. Sherinprincy.M (msherinprincym@gmail.com)
 3. Ashwini.P.S (pydiaswinips@gmail.com)

2. Project Overview

- **Purpose:** A5B News is a web application built using the MERN stack (MongoDB, Express, React, Node.js). It serves as a platform to display news articles from various categories and allows administrators to manage the content through an admin panel.
- **Features:** User registration and login system
- Admin panel to create, update, and delete news articles
- View, search, and read news articles
- User profile management

3. Architecture

- **Frontend:** React.js with Bootstrap and Material UI
- **Backend:** MongoDB stores user data, project information, applications, and chat messages
- **Database:** Node.js and Express.js managing server logic and API endpoints

4. Setup Instructions

- **Prerequisites:**
 -
 - Node.js
 - MongoDB

- Git
- Express.js - Mongoose - Visual Studio Code
- React.js

- **Installation Steps:**

```
# Clone the repository git clone

# Install client dependencies cd
client npm install

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

5. Folder Structure

```
uma/
|-- client/          # React frontend
|   |-- public/
|   |-- src/
|       |-- components/
|       |-- pages/
|-- controllers/     # Backend logic
|-- models/          # Database schemas
|-- routes/          # API routes
`-- index.js         # Backend server entry point
```

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

- **Projects:**

- /api/projects/:id • **Applications:** /api/apply
- /api/projects/create
- **Chats:**
 - /api/chat/send
 - /api/chat/:userId

8. Authentication

- JWT-based authentication for secure login
- Middleware protects private routes

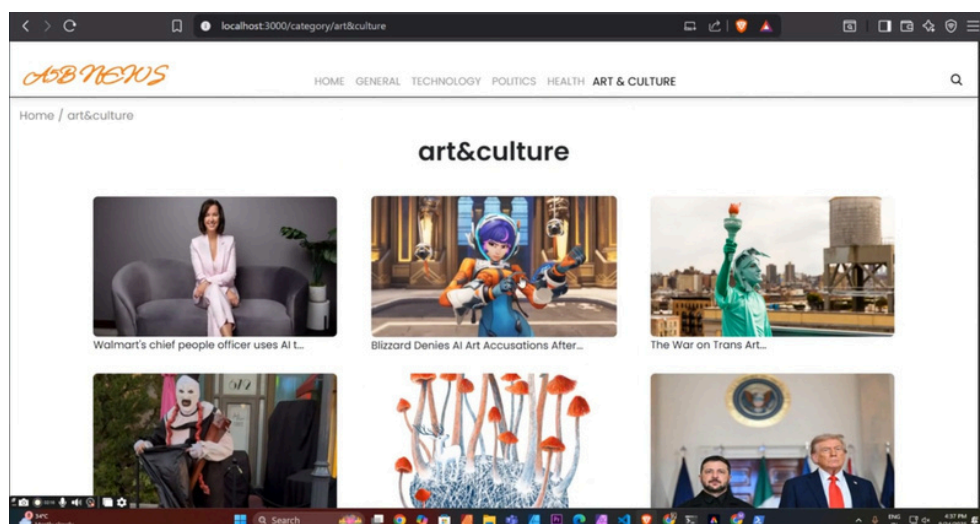
9. User Interface

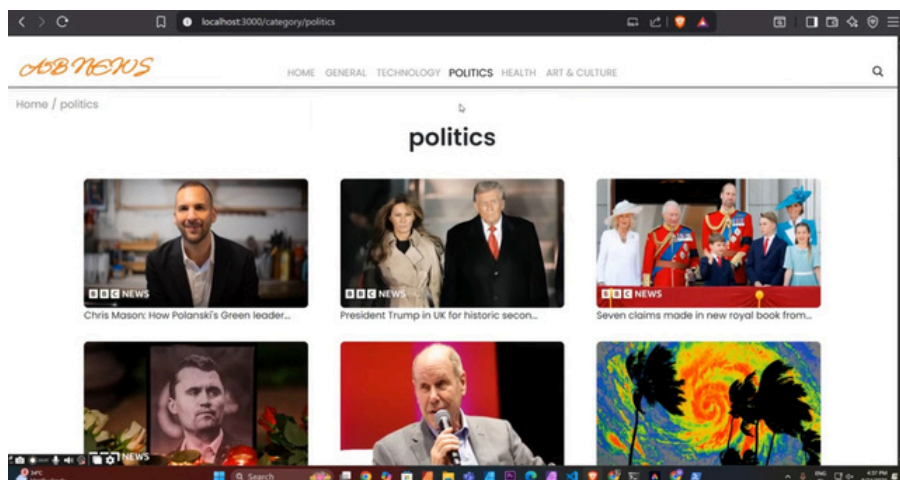
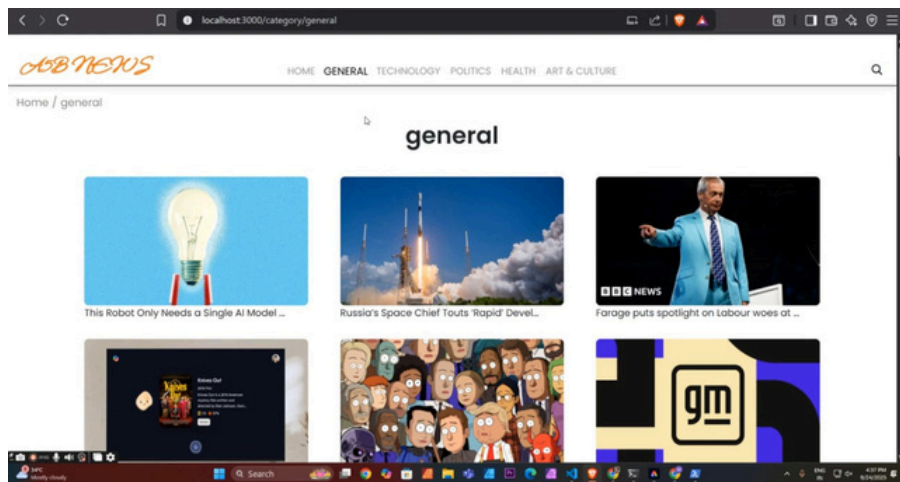
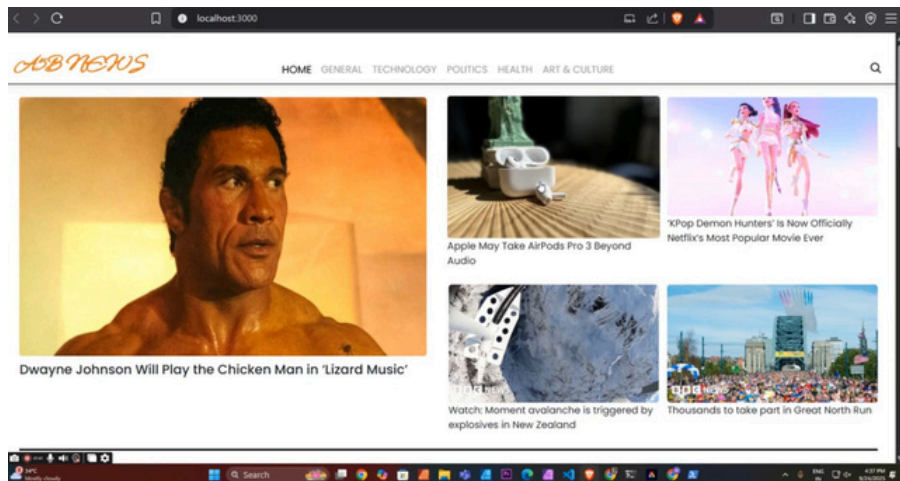
- Landing Page
- Freelancer Dashboard
- Admin Panel
- Project Details Page

10. Testing

- Manual testing during milestones
- Tools: Postman, Chrome Dev Tools

11. Screenshots or Demo :





12. Known Issues:

- User Interface & Experience:
 - The application may not be fully responsive across all device sizes (e.g., mobile phones, tablets, and desktops).
 - User feedback for actions like successful login or failed post creation might be missing or unclear.
 - The user session may not persist after a page refresh, requiring the user to log in again.
- Functionality:
 - The search feature could be case-sensitive or may not handle partial words or special characters correctly.
 - Error handling might be incomplete; for instance, if the database connection fails, the application might crash instead of showing a user-friendly error message.
- Security & Validation:
 - Input fields (like registration forms or comment sections) may lack sufficient validation, potentially allowing for invalid data submission.

13. Future Enhancement :

User-Focused Features:

- Article Bookmarking: Allow logged-in users to save or bookmark articles to read later, which would be accessible from their profile.
- Advanced User Profiles: Expand user profiles to include options for a profile picture, a short biography, and a view of their activity (e.g., articles posted by them if they are an author).
- Social Sharing: Implement buttons on article pages to allow users to easily share news on social media platforms like X (formerly Twitter), Facebook, or LinkedIn.

Content Management & Organization:

- Categorization and Tagging: Add a system for admins to assign categories (e.g., "Technology," "Sports," "Politics") and tags to news articles, allowing users to filter content based on their interests.
- Pagination: Introduce pagination on the homepage and category pages to manage and display a large number of articles without cluttering the user interface.
- Advanced Search: Improve the search functionality with filters, enabling users to search by date range, category, or author.

Technical Improvements:

- API Integration: Connect to a third-party news API to automatically fetch and display news content, supplementing the manually added articles.
- Unit & Integration Testing: Develop a testing suite using frameworks like Jest and React Testing Library to ensure application stability and prevent regressions.
- Deployment: Create deployment scripts and instructions for hosting the application on cloud platforms like Vercel, Netlify, or AWS.