A Major Project Synopsis on

**NEWS APPLICATION**

Submitted to Manipal University, Jaipur

Towards the partial fulfillment for the Award of the Degree of

MASTER OF COMPUTER APPLICATIONS

2023-2025

By

**Vikash Kumar Dubey**

**23FS20MCA00042**

Under the guidance of

Divya Sharma

Department of Computer Applications

School of AIML, IoT&IS, CCE, DS and Computer Applications

Faculty of Science, Technology and Architecture

Manipal University Jaipur

Jaipur, Rajasthan

2025

**I. Introduction**

In a time when information is rapidly changing, the need for reliable and timely news has become increasingly critical. News Application is a comprehensive news application developed using the MERN stack (MongoDB, Express.js, React, Node.js), aimed at providing users with a smooth and engaging news experience. Whether you are an administrator overseeing content or a reader seeking the latest news, News Application guarantees credibility, accessibility, and user interaction within a single robust platform.

Why Choose News Application?

Conventional news outlets frequently fall short in delivering real-time updates, interactivity, and tailored engagement. News Application addresses these shortcomings by offering an intuitive and feature-rich environment where:

* Administrators can effectively manage and publish news articles through a well-structured content management system.
* Users can access news tailored to their interests, engage with the content, and receive immediate updates.
* Real-time notifications ensure that users remain informed about significant developments, keeping them connected to the latest news.

**Key Features of News Application**

1. Admin-Controlled News System

Administrators possess full authority over the news content, allowing them to publish, edit, and remove articles, thereby ensuring that only verified and pertinent news is delivered to users.

2. Secure and Customized User Experience

The platform is protected by JWT-based authentication, guaranteeing that only registered users can engage with the content. Users have the ability to customize their news feed according to their preferences.

3.Interactive Feedback Mechanism

User engagement is essential in the current digital environment. Users can express their opinions by liking or disliking articles, providing valuable insights into public sentiment and enhancing interactivity on the platform.

4.Instant Updates with Socket.IO

Given the time-sensitive nature of news consumption, the integration of Socket.IO allows for immediate updates to be sent to users, ensuring they remain informed without any delays.

5.Intelligent Watchlist Feature

Users can save articles or follow specific categories, ensuring they stay updated on topics that matter most to them.

**II. Motivation**

Our platform is crafted to transform the consumption and management of news, providing a seamless, interactive, and trustworthy experience for both administrators and users.

**1. For Administrators**

* Simplified News Management – Effortlessly publish, edit, or remove news articles.
* Secure Access & Role-Based Permissions – Utilize JWT-based authentication to effectively manage user roles.
* Real-Time News Delivery – Keep users informed with immediate updates on breaking news through Socket.IO integration.
* Content Analytics & Engagement Tracking – Analyze article performance by monitoring user interactions.
* Effective Moderation – Combat misinformation with a structured approach to content management.

**2. For Users**

* Customized News Feed – Access news tailored to your interests and preferences.
* Interactive Experience – Engage with news articles by liking, disliking, and participating in real time.
* Intelligent Watchlist & Notifications – Save articles and receive prompt updates when new content is available.
* Effortless Navigation – Navigate through news easily with category-based filtering and search functionalities.
* Secure & Ad-Free Browsing – Experience a distraction-free environment with verified content.

With News Application, our objective is to establish a contemporary, efficient, and user-focused news platform that empowers individuals with real-time, authentic, and tailored news updates.

**III. Problem Statement**

**1. For Administrators**

* Ineffective News Management – Conventional platforms do not offer centralized oversight for content publication and moderation.
* Absence of Real-Time Updates – The lag in news dissemination results in users receiving outdated information.
* Restricted User Engagement – Traditional news websites are deficient in interactive elements such as reactions and watchlists.
* Challenges in Content Authentication – Without a structured administrative framework, ensuring the reliability and verification of news is problematic.

**2. For Users**

* Dispersed and Unreliable News Sources – Users face difficulties in locating authentic and pertinent news in a single location.
* Information Delays – Numerous platforms do not provide immediate updates on breaking news events.
* Lack of Personalization – Users frequently encounter generic content that does not cater to their specific interests.
* Absence of Interactive Features – Limited opportunities for engagement, bookmarking, and feedback render news consumption a passive experience.

With News Application, we intend to address these challenges by offering a secure, interactive, and real-time news platform, ensuring that users receive verified, personalized, and promptly updated news content.

**IV. Methodology / Work Planning**

1. Development of the project's user interface utilizing React.js, which will encompass various components, including:

* Home – Showcases the latest and trending news articles.
* News Preview – Presents detailed news articles along with options for user engagement.
* Categories – Facilitates user navigation through news based on sections such as Politics, Sports, Technology, etc.
* Watchlist – Allows users to bookmark articles for future reading.
* Admin Dashboard – Offers tools for the management of news content and user interactions.
* Profile – Displays user information, preferences, and saved articles.

2. Establishing distinct portals:

* **Admin Portal**
* News Management – Create, modify, and remove news articles.
* User Management – Oversee user roles, authentication, and permissions.
* Analytics Dashboard – Monitor user engagement, article views, and interactions.
* Real-Time Notifications – Dispatch immediate alerts for breaking news.
* Content Moderation – Supervise and verify the authenticity of news to combat misinformation.
* Support & Contact Management – Address user inquiries and feedback.
* **Journalist/Editor Portal**
* Publish Articles – Submit and manage news content.
* Image & Media Upload – Incorporate multimedia elements into news articles.
* Scheduled Publishing – Arrange for news articles to be published at designated times.
* Content Approval – Submit articles for administrative verification prior to publication.
* Update & Archive News – Revise or eliminate outdated articles.
* **User Portal**
* Browse & Read News – Access and explore news across various categories.
* Personalized Feed – Receive recommendations tailored to user preferences.
* Watchlist & Bookmarks – Save articles for later reading.
* Engagement Features – Like, dislike, and comment on news articles.
* Real-Time Updates – Obtain immediate notifications regarding breaking news and followed topics.

3. Collaborating with the backend utilizing:

* Node.js with Express.js – For managing API requests and business logic.
* MongoDB – Serving as the database for the storage of user data and news articles.
* Socket.IO – For real-time news updates and notifications.  
  d. JWT Authentication – To ensure secure access and role-based permissions.

**V. Software Requirements Specifications**

**Functional**

1. Software Requirements:

a. Operating System: Windows, Linux, macOS  
b. User Interface: React.js with Material-UI for styling  
c. Database: MongoDB  
d. Backend: Node.js with Express.js  
e. Real-Time Features: Socket.IO for live updates  
f. Authentication: JWT-based secure login system

2. Hardware Requirements:

a. Processor: Intel Core i3 or higher / AMD equivalent  
b. RAM: Minimum 4GB (Recommended: 8GB for better performance)  
c. Storage: Minimum 20GB of available disk space  
d. Internet Connection: Required for real-time updates and API calls

**Non-Functional**

Performance: Pages should load within 2 seconds; API responses under 500ms.

Scalability: Support up to 10,000 concurrent users and 1M news items.

Availability: Ensure 99.9% uptime with automatic recovery and load balancing.

Security: Use JWT auth, bcrypt for passwords, HTTPS, and role-based access control.

Maintainability: Follow modular code structure and naming conventions with documentation.

Usability: UI must be responsive and mobile-friendly with a fast search experience.

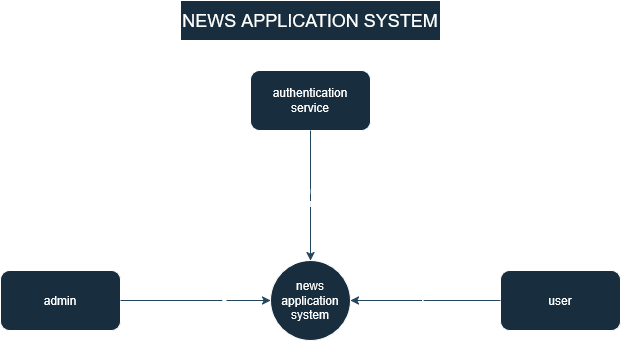
Portability: Compatible with all major browsers and OS platforms.

Logging: Log errors and user activity for monitoring; use tools like Winston or Sentry.

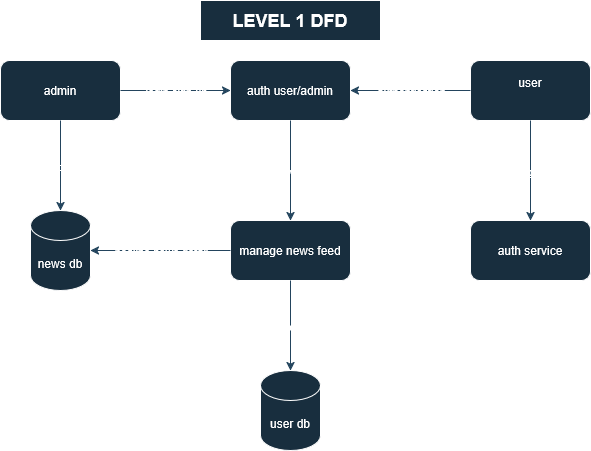
Backup: Daily database backups with 1-hour recovery capability.

**VI. Data Flow Diagram**

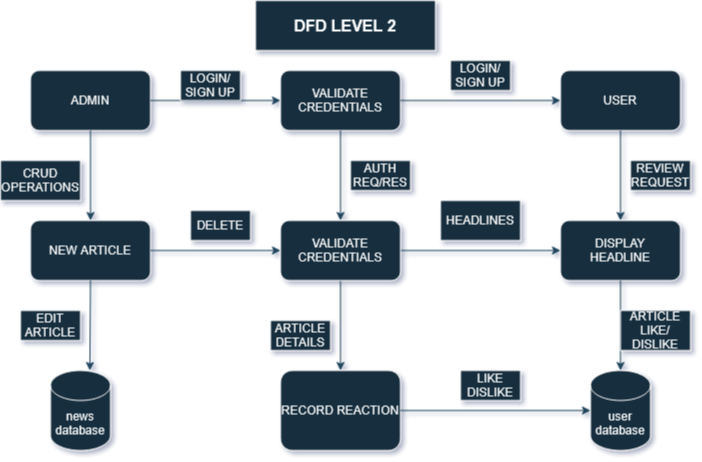
**Level 0 DFD**



**Level 2 DFD**

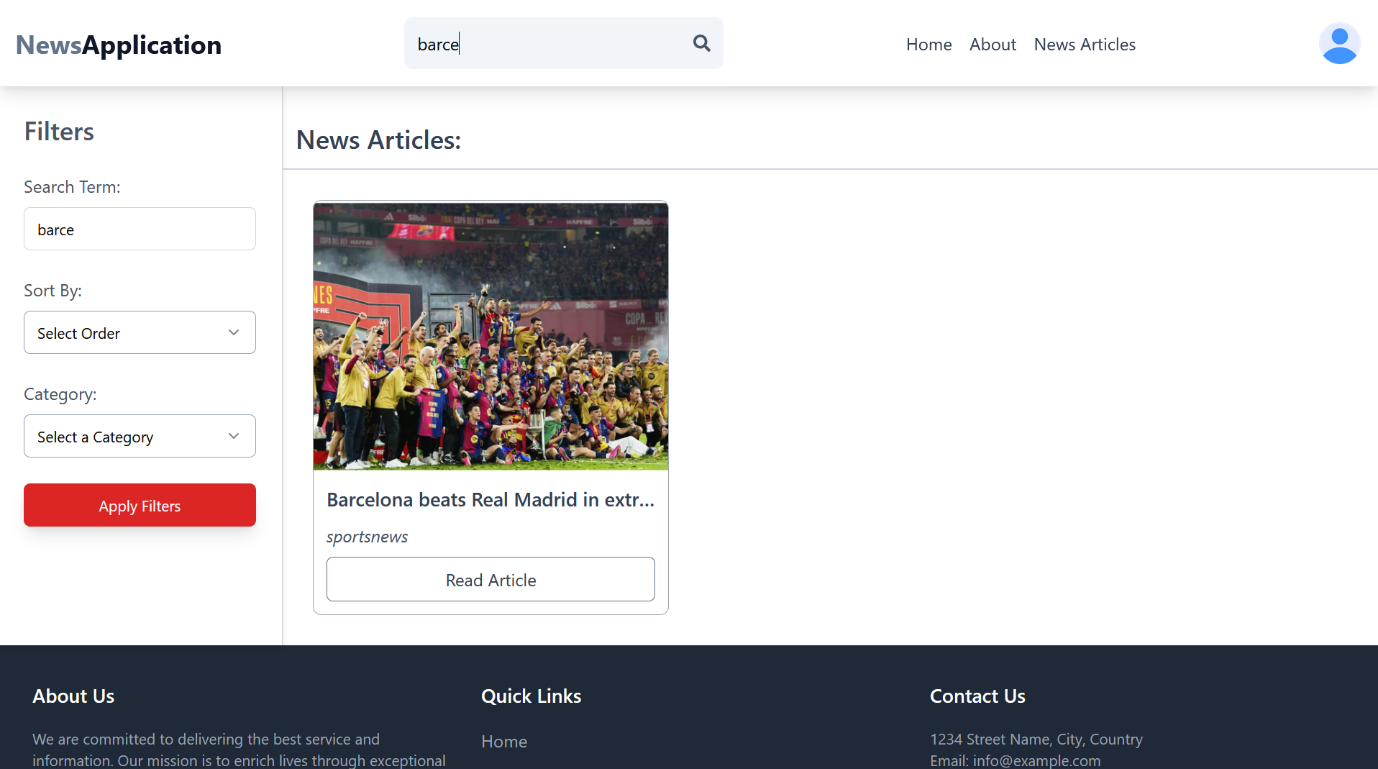


**Level 2 DFD**

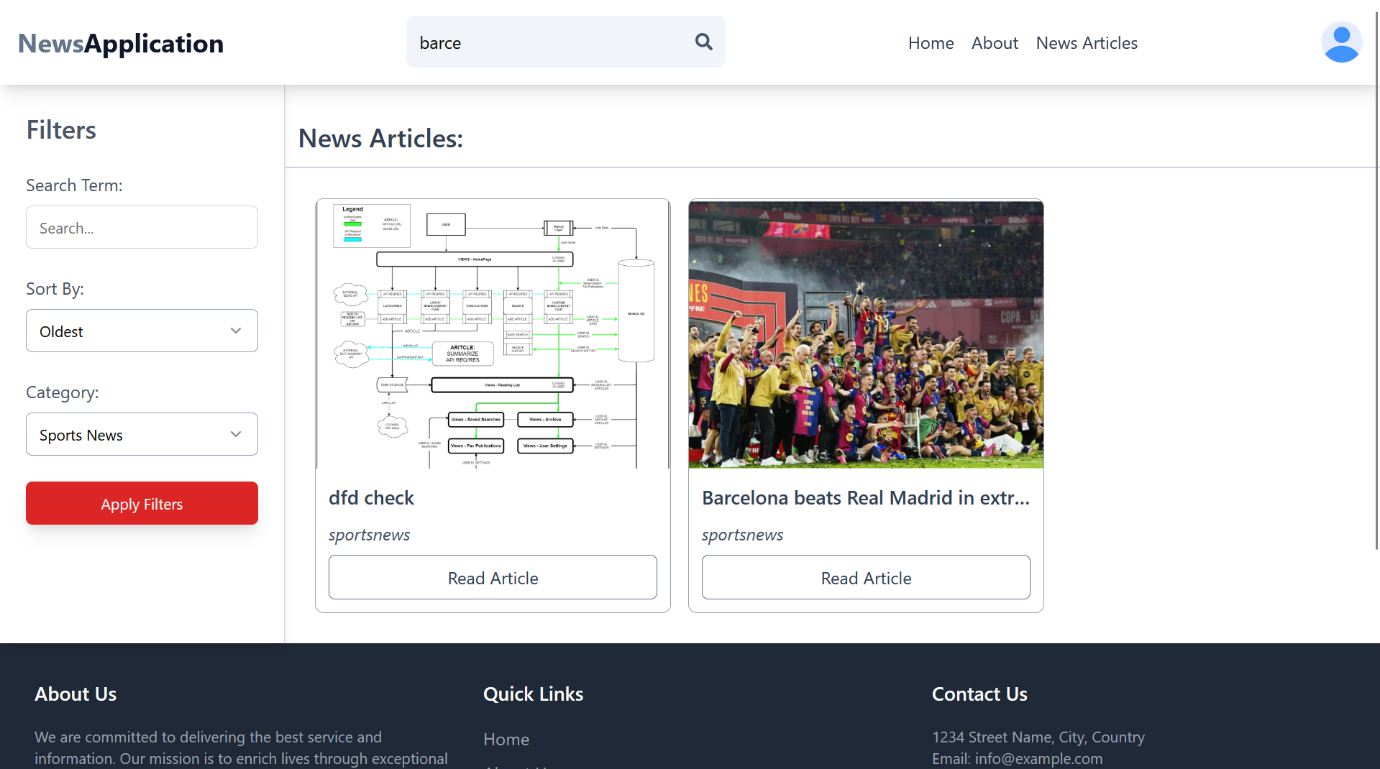


**VII. Screenshot**

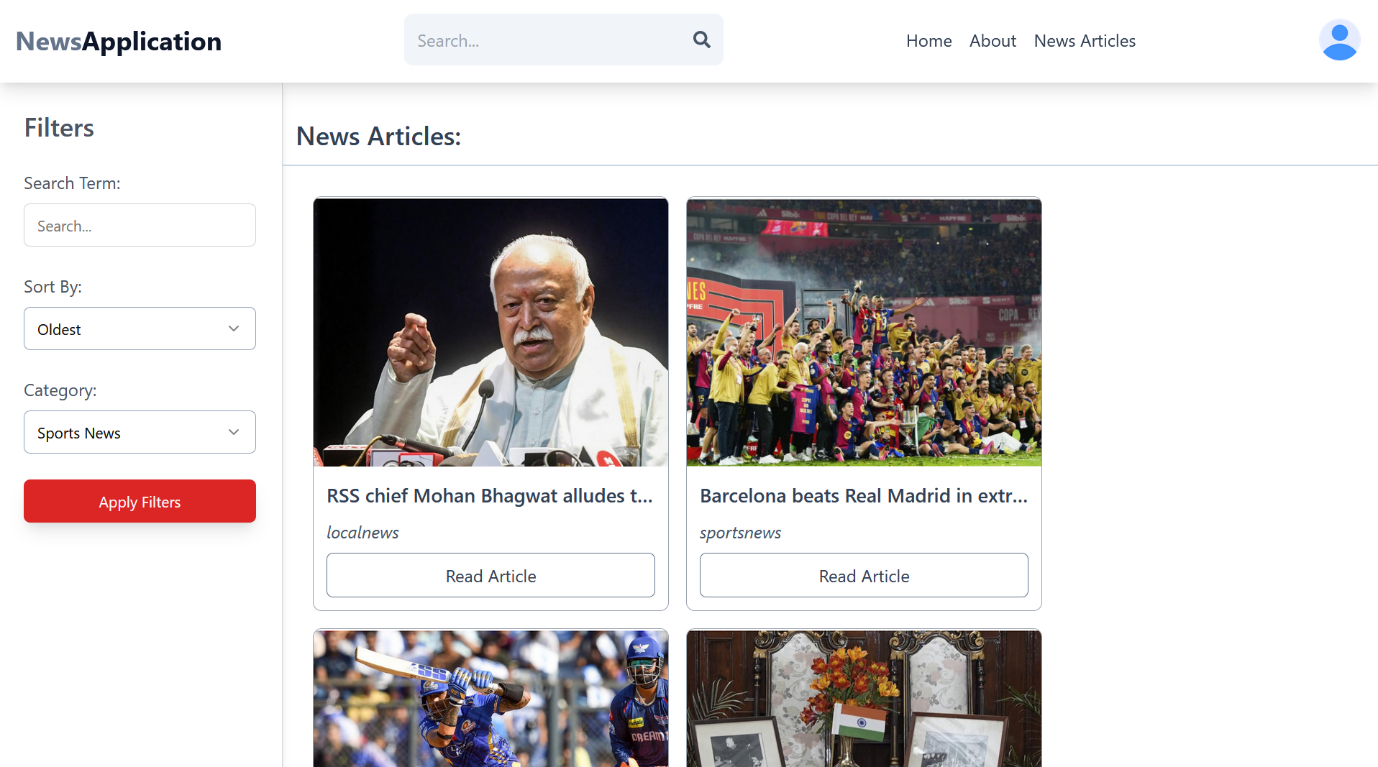
Search post

****

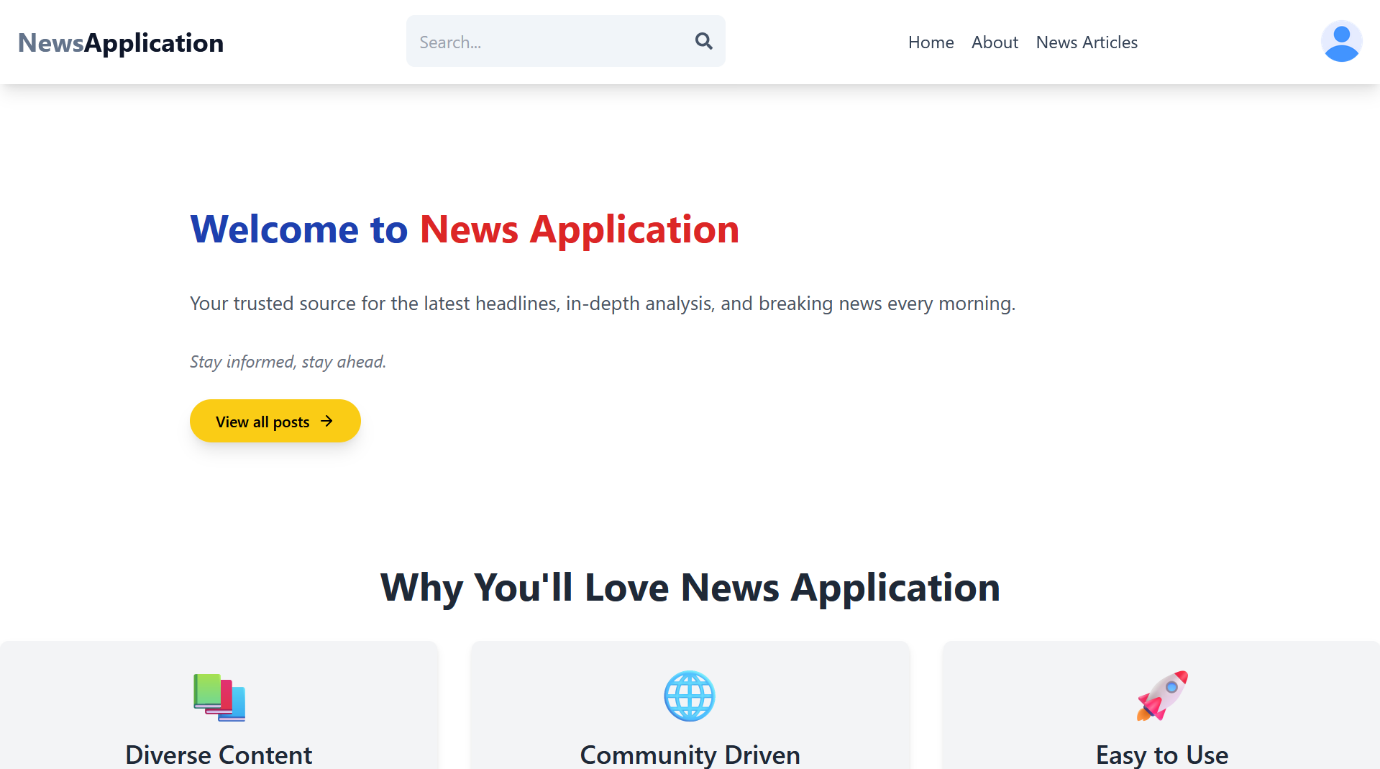
Filter post

****

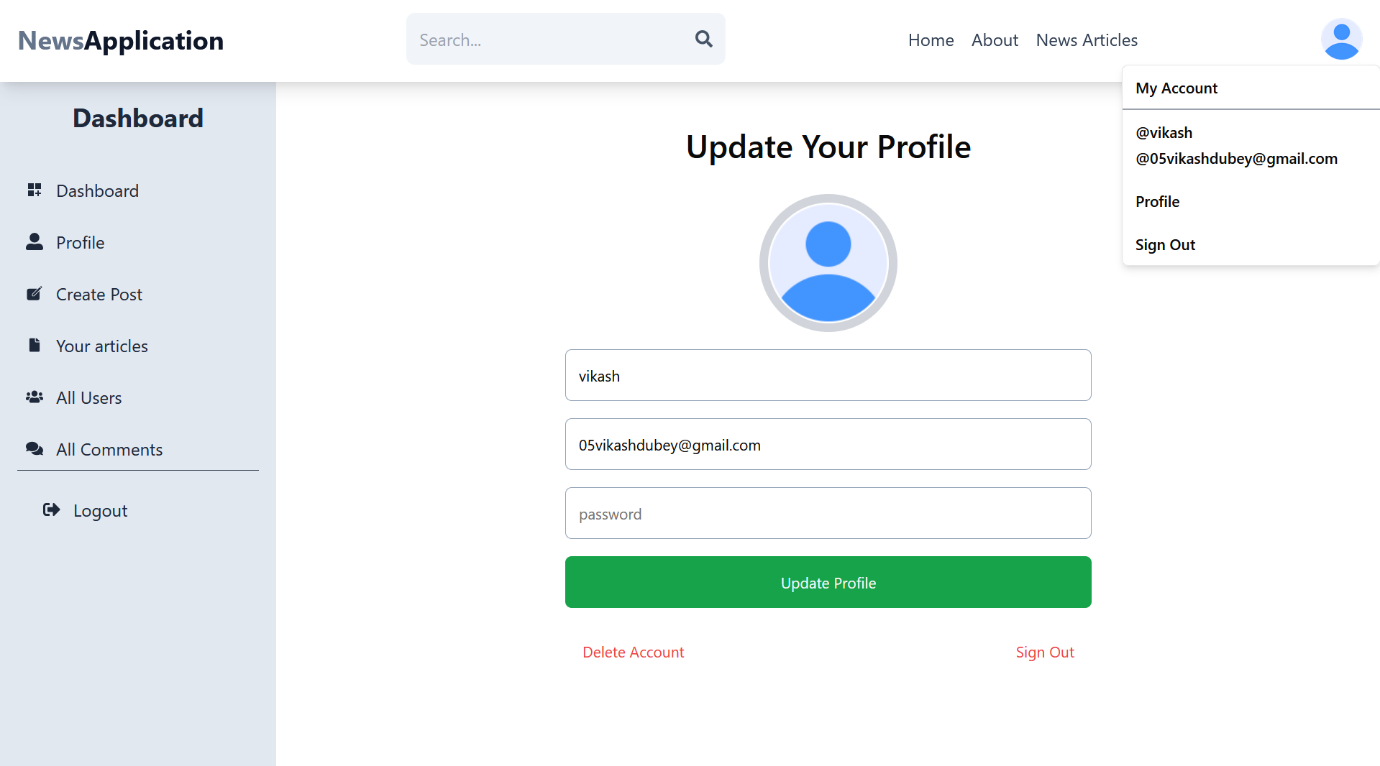
All posts

****

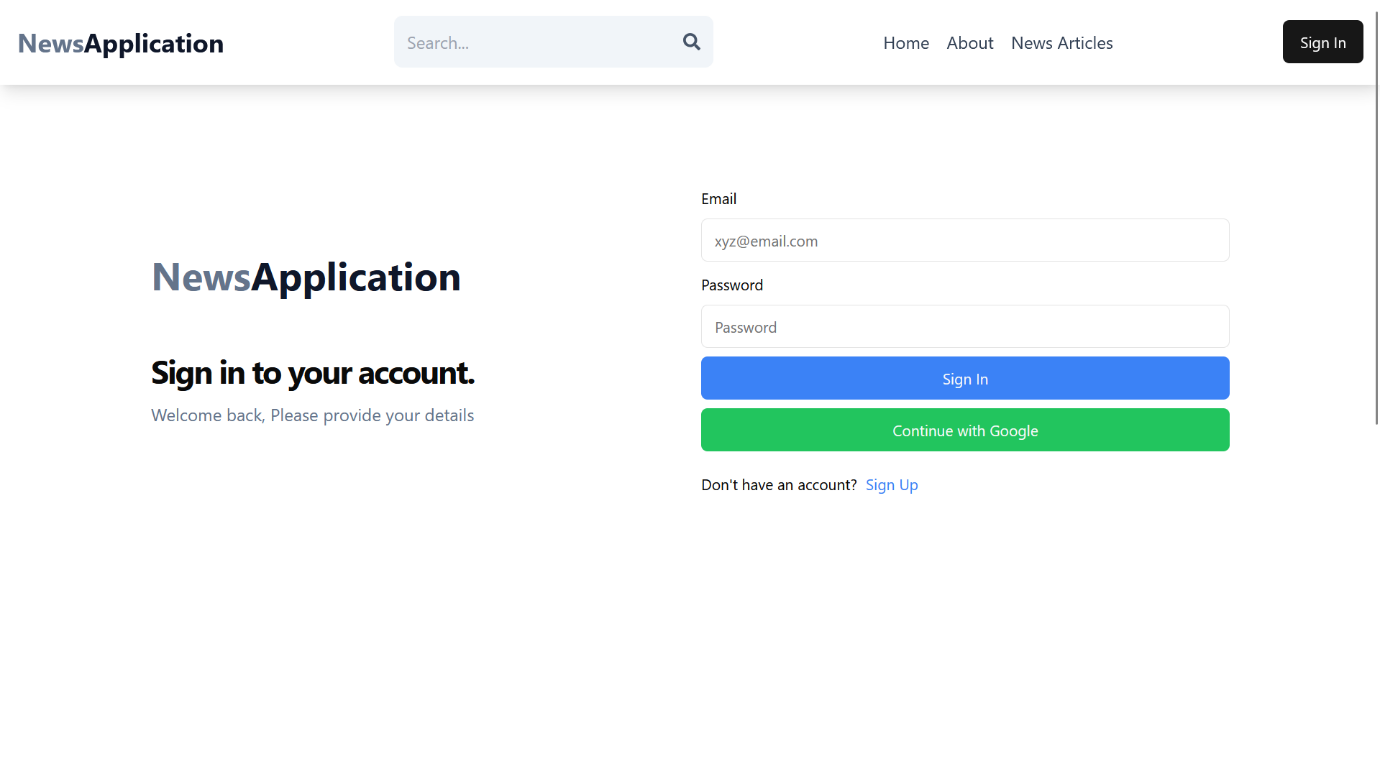
**Home page**

****

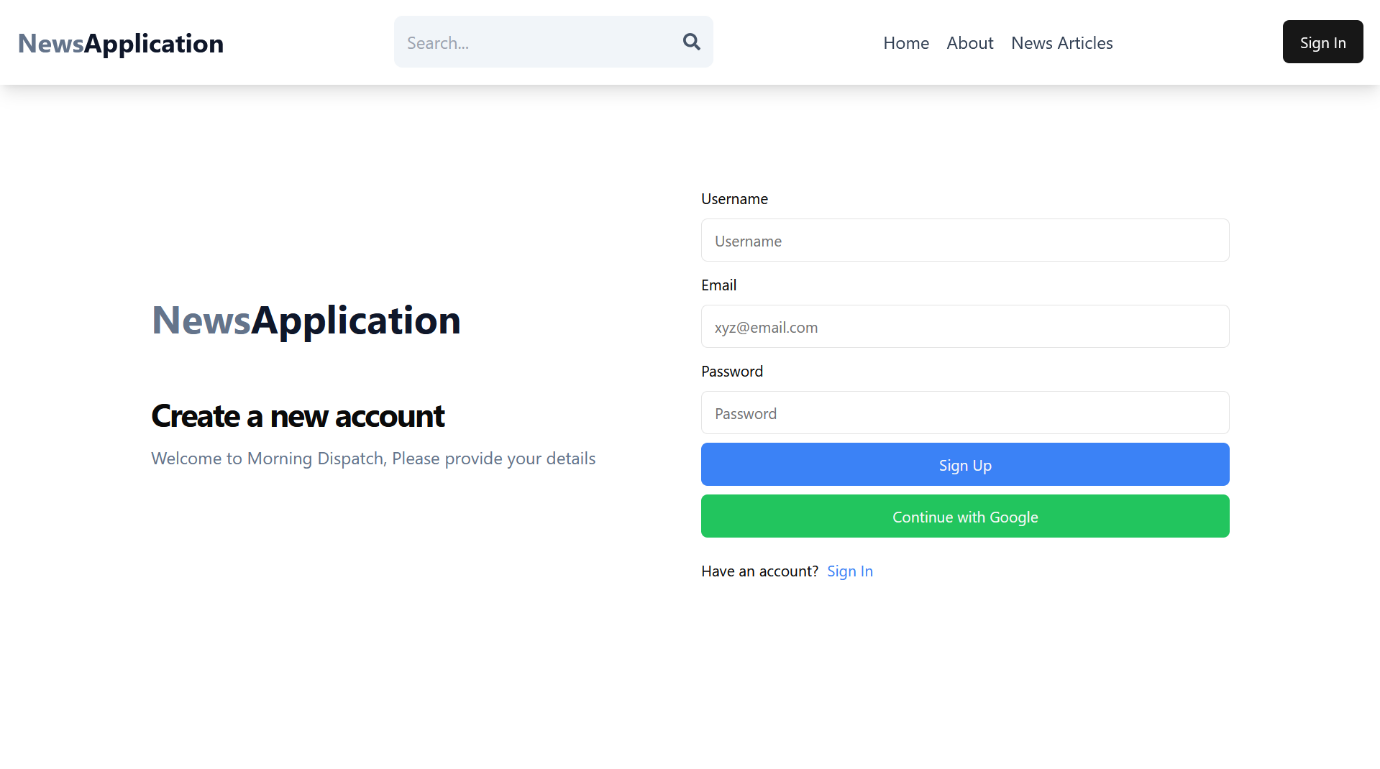
Admin Dashboard

****

Sign in

****

Sign up

****

**VIII. Bibliography/References**

1. **MERN Stack Documentation**
   * MongoDB: <https://www.mongodb.com/docs/>
   * Express.js: <https://expressjs.com/>
   * React.js: <https://react.dev/>
   * Node.js: <https://nodejs.org/>
2. **Real-Time Communication**
   * Socket.IO: https://socket.io/docs/
3. **Authentication & Security**
   * JSON Web Tokens (JWT): https://jwt.io/introduction/
4. **API Development & Best Practices**
   * RESTful API Guide: <https://restfulapi.net/>
5. **Frontend & UI Design**
   * Material-UI: <https://mui.com/>
6. **News Industry & Best Practices**
   * Digital News & Media Trends: <https://www.niemanlab.org/>