

READY NEWS

A Project Report Submitted by:

E. Viswanatha rao-AP23110010110

S. Harshith Reddy -AP23110011193

G. Jyothika-AP23110010121

N. Maansi-AP23110010721

Under the Supervision of:

Mr. Syed Arshad

Assistant Professor

Department of Computer Science and Engineering

SRM University-AP

In partial fulfilment for the requirements of the

FullStack Development-I(SEC 160) Project

BACHELOR OF TECHNOLOGY

IN

COMPUTER SCIENCE AND ENGINEERING



SRM
UNIVERSITY AP

Andhra Pradesh

INTRODUCTION

In the digital era, information travels faster than ever, and users expect instant access to personalized, accurate, and diverse news content. Traditional news portals provide global news but lack personalization, user-driven content creation, and interactivity. To bridge this gap, **Ready News** is developed — a modern, responsive, and highly interactive news management system.

Ready News combines live news consumption with user-generated articles. It offers features like CRUD operations, authentication, and theme switching, making it more versatile than typical news websites. Users can explore global headlines, publish personal articles, edit or delete content, and maintain their own news collection.

The system is built using **React.js**, **Axios**, **JSON Server**, and **REST API integration**, providing a professional-level architecture and advanced UI experience.

This project aims to simulate the workflow of real newsroom environments while showcasing full-stack development principles and modern frontend engineering.

ABSTRACT

Ready News is a full-stack web-based news application that enables users to **read, create, update, and delete news articles** while also consuming **real-time global news** pulled from an external API. The platform supports **user login, custom story creation, and category-based organization** of articles. It integrates **JSON Server** as a mock backend and **React.js** as the frontend framework.

The system provides a modern user interface with theme toggling, API-driven dynamic content, and smooth navigation through React Router. The application demonstrates clear implementation of CRUD operations, RESTful practices, authentication, and responsive UI development.

PROBLEM STATEMENT

Existing news platforms primarily focus on delivering global or curated content, offering little flexibility for users who want to:

- Publish their own news
- Customize feeds
- Edit or maintain articles
- Create personalized news archives

There is a lack of a unified system where users can **consume live news** while also **creating and managing their own stories**.

Ready News solves this problem by providing:

- ✓ A live news engine
- ✓ Custom article creation
- ✓ Local storage using JSON server
- ✓ Editing and deletion options
- ✓ Category-based organization
- ✓ Login and secure access
- ✓ Theme customization

OBJECTIVE

The key objectives of Ready News are:

1. Provide Real-Time News

Integrate a live news API to fetch the latest headlines, ensuring users stay updated with trending stories, especially in technology and global categories.

2. Enable User-Controlled News Publishing

Allow users to act as journalists:

- Create news
- Upload images
- Provide external source links
- Assign categories

3. Simulate Real Newsroom Operations

CRUD operations allow users to function like editors:

- Write
- Modify
- Update
- Delete

◆ 4. Offer a Secure Login System

Authentication ensures personalized access and data security—just like real professional media platforms.

◆ 5. Deliver a Modern, Professional UI

Using React's component-driven architecture and CSS styling, the application ensures:

- Mobile responsiveness
- Professional animations
- Dark/Light themes
- Clean navigation

◆ 6. Develop a Scalable Full-Stack Architecture

React + JSON Server + Axios combination demonstrates true end-to-end development skills.

TECHNOLOGIES USED

Frontend

- React.js
- JSX
- CSS
- Bootstrap
- React Router

Backend

- JSON Server (Local REST API)

API Integration

- NewsAPI.org (Live global news)

Tools & Libraries

- Axios
- Lucide Icons
- Browser Local Storage

MODULES

1. Home Page

Displays:

- Brand heading ("Ready News")
- Live Technology Headline (via API)
- Navigation buttons
- Feature descriptions
- Professional footer

2. Live News Module

- Fetches real-time trending news using **Axios**

Displays:

- Title
- Description
- Image
- Read More (redirects to source article)

3. Add News Module

Users can:

- Add title
- Add description
- Provide external news URL
- Upload or link a news image
- Choose a category
- Save data to JSON Server
- Reflects the **Create** part of CRUD.

4. My News Module:

A complete CRUD dashboard:

- ✓ **Read** user-published news
- ✓ **Edit** using a popup window
- ✓ **Delete** articles
- ✓ **Instant update** on save
- ✓ **Search bar** for filtering among created articles

5. Authentication Module

Login system using JSON:

- Email-based login
- Password validation
- Redirect to home after login
- Error handling for invalid users

6. About Page

- Introduces project team members
- Displays roles
- Grid-based professional layout

FEATURES

- **CRUD Operations**
Full control over user-created articles.
- **Live News Fetching**
Real-time global technology news.
- **Dark & Light Theme Toggle**
Provides a stylish, customizable UI experience.
- **Routing and Navigation**
Smooth navigation using React Router.
- **Responsive UI**
Professional dashboard design.
- **Authentication System**
Only registered users can access news creation features.
- **Professional Animations & Effects**
Particles background, glowing UI elements, modern typography.

WORKFLOW

1) User opens **Home Page**

API fetch executes (Displays the latest technology headline)

2) User clicks **Live News**

Axios fetches real-time news from NewsAPI

3) User clicks **Add News**

User enters:

- Title
- Description
- Image
- URL

Saved to JSON Server

4) User clicks **My News**

CRUD interface loads Users can edit or delete articles

5) Login ensures only authenticated sessions can modify news

6) Navigation + Theme toggle ensures smooth user experience

IMPLEMENTATION

Frontend Components:

Component	Function
App.js	Global routing + Theme management
Navbar.js	Navigation bar + Theme switch
Home.js	Live news preview + Feature showcase
LiveNews.js	Fetch and display API-based news
AddNews.js	News creation form
NewsList.js	News dashboard (CRUD)
NewsCard.js	Individual card UI with edit/delete
LoginPage.js	Authentication
About.js	Team info page

Backend (JSON Server)

Stores:

- News articles
- User login data

Endpoints:

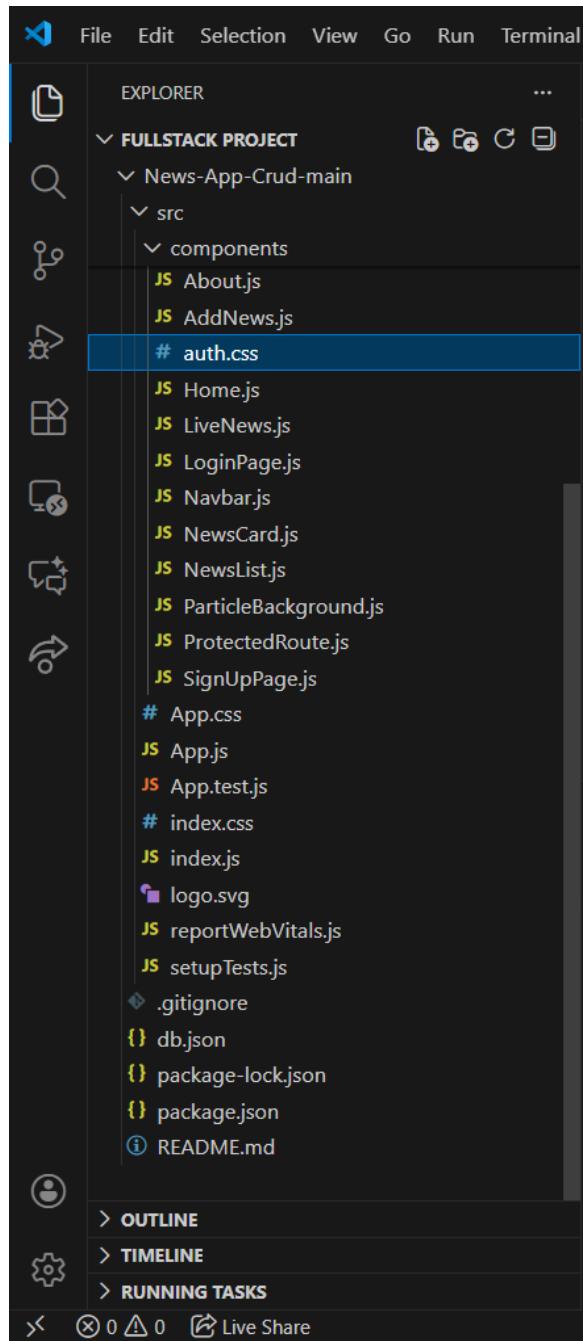
- /news
- /users

Supports:

- GET
- POST
- PUT
- DELETE

acts like a real REST backend.

SCREENSHOTS



```

    .team-grid {
      display: grid;
      grid-template-columns: repeat(4, 1fr);
      gap: 25px;
    }

    .team-card {
      background: var(--card-bg);
      padding: 20px;
      border-radius: 16px;
      box-shadow: 0 0px 20px rgba(0,0,0,0.12);
      transition: transform 0.2s ease, box-shadow 0.3s ease;
    }

    .team-card:hover {
      transform: scale(1.05);
      box-shadow: 0 0px 20px rgba(0,0,0,0.15);
    }
  
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

```

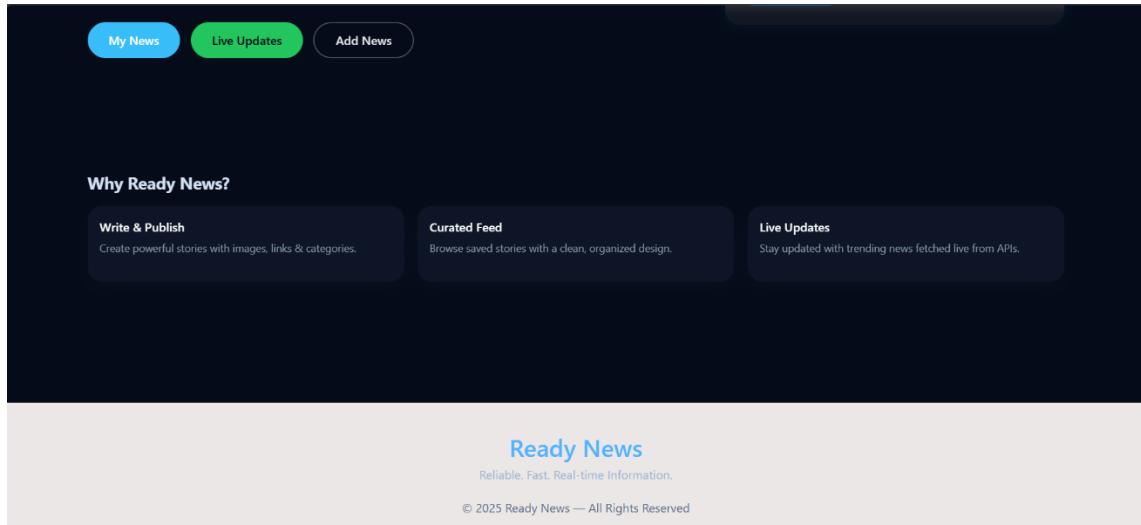
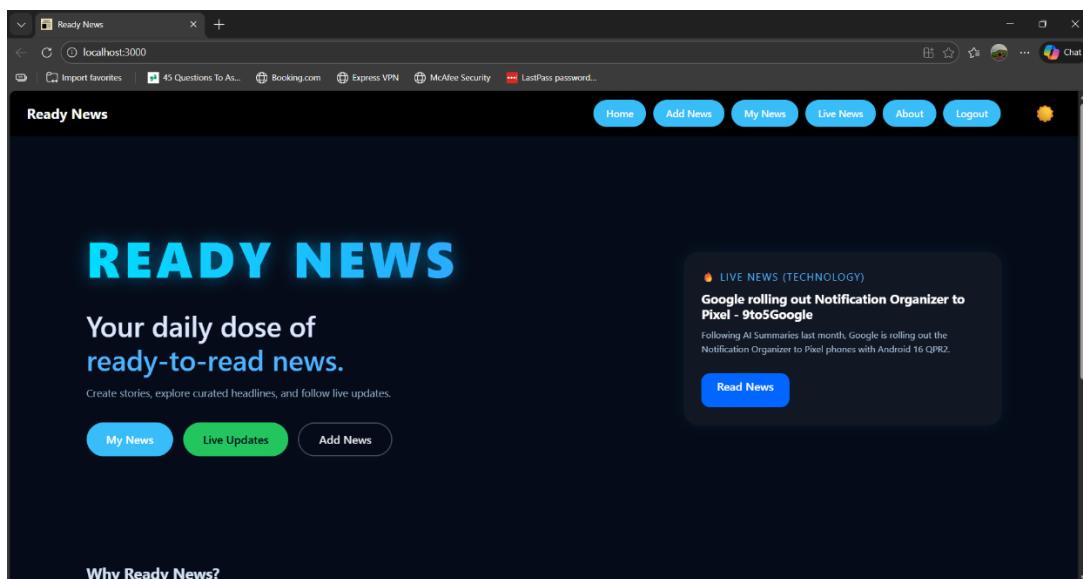
  $ npm start
  You can now view newscrud in the browser.

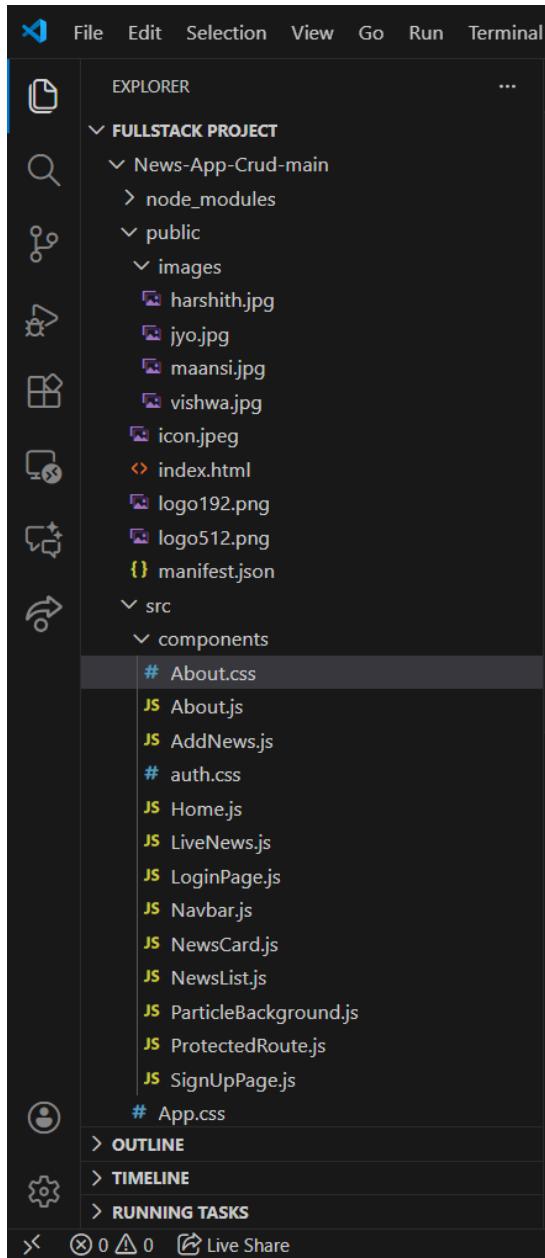
  Local:          http://localhost:3000
  On Your Network: http://172.16.0.2:3000

  Note that the development build is not optimized.
  To create a production build, use npm run build.
  webpack compiled successfully
  
```

In 8, Col 20 Spaces: 2 UTF-8 CRLF (CSS 0:0 Go live

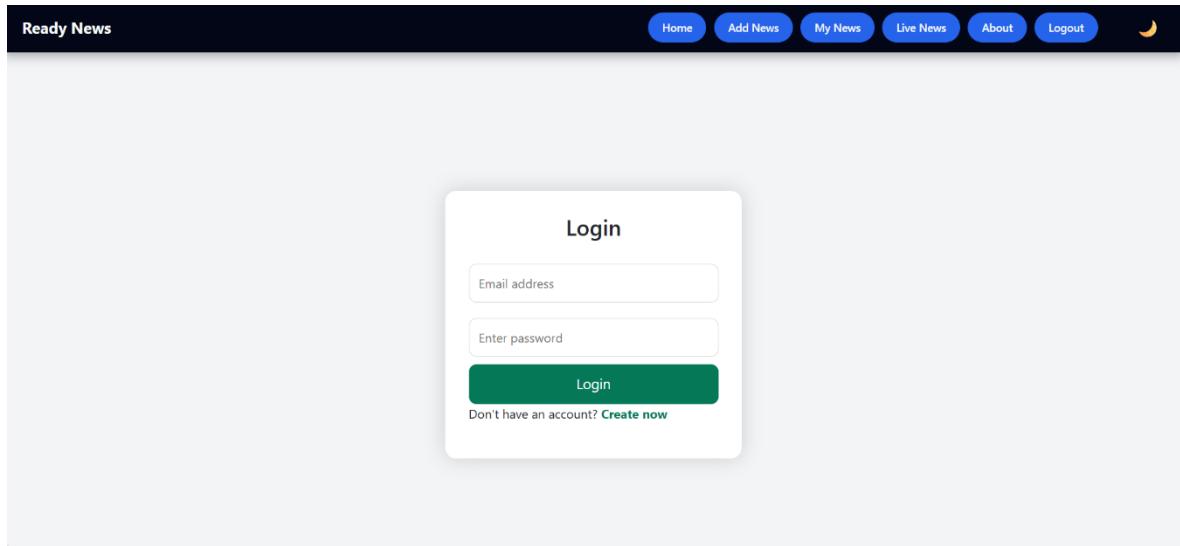
Home Page:





Signup/Login Page:

The screenshot shows a web application interface titled "Ready News". The top navigation bar includes links for Home, Add News, My News, Live News, About, Logout, and a toggle between day and night mode. The main content area features a "Create Account" form with fields for Full Name, Email address, Create password, and Confirm password. A green "Sign Up" button is at the bottom of the form. Below the form, a link says "Already have an account? Login".



About us:

A screenshot of the Ready News website's "About Us" section. The header is identical to the login page, with "Ready News" and various blue navigation buttons. The main content starts with a section titled "About Us" which includes a short description: "We are a passionate team dedicated to building Ready News — a smooth, intelligent news platform that delivers real-time updates, curated stories, and user-powered content." Below this is a section titled "Our Team" featuring four team members in circular profile pictures. Each member has a name, title, and ID listed below their photo: E. Vishwanatha Rao (Team Lead, ID: AP23110010350), Harshith Reddy (Frontend Developer, ID: AP23110011193), Jyothika Gunta (Backend Developer, ID: AP23110010121), and Maansi Nalla (UI/UX Designer, ID: AP23110010721).

Publishing an article:

A screenshot of the Ready News website's "Publish a New Article" page. The header is consistent with the other pages. The main form has fields for "Title" (containing "SRM-AP"), "Description" (containing "example news typed reflection"), "Category" (set to "General" with a "Or type custom" option), and "Source URL" (containing a link to a Hindustan Times article). To the right of the form is a "Live Preview" window showing a thumbnail of a modern building with a green roof, the category "GENERAL", the author "SRM-AP", a snippet of the description, and the full URL of the source article.

Ready News

Home Add News My News Live News About Logout

Description
example news typed reflection

Category General Or type custom

Source URL <https://www.hindustantimes.com/india-news/cyclone-ditwah-live-updates-landfall-imd-a>

Image URL <https://th-i.thgim.com/public/news/national/2mwvdyp/article/70251518.ece/alternates/LAI>

Publish Article

Live Preview

GENERAL
SRM-AP
example news typed reflection
<https://www.hindustantimes.com/india-news/cyclone-ditwah-live-updates-landfall-imd-alert-tamil-nadu-puducherry-chennai-andhra-pradesh-rains-november-30-ne-ws-101764474339986.html>

Live News:

Ready News

Home Add News My News Live News About Logout

Live Trending News

General Business Technology Sports Entertainment Health

Google rolling out Notification Organizer to Pixel - 9to5Google
Following AI Summaries last month, Google is rolling out the Notification Organizer to Pixel phones with Android 16 QPR2.

Read More

Save to My News

Indiana Jones and the Great Circle update adds an outfit suitable to eat chilled monkey brains in - Eurogamer
The team at MachineGames has issued a new update for Indiana Jones and the Great Circle.

Read More

Save to My News

2025 WRAP-UP
PlayStation Wrap-Up 2025 Live Now, Get Your Gaming Stats for the Year - Push Square
Track your playtime, Trophies, and more

Read More

Save to My News

Saved Articles:

Ready News

Home Add News My News Live News About Logout

Your Articles

Search your news...

Former Trump lawyer Alina Habba is disqualified as top New Jersey prosecutor, US appeals court rules - AP News
President Donald Trump's ex-lawyer Alina Habba has been disqualified from serving as New Jersey's top federal prosecutor. A federal appeals court said Monday the Trump administration's maneuvers to keep Habba in the role were contrary to federal laws governing...

Read More

Kris Jenner shows off new Anna Wintour-style bob for the holidays - Page Six
The momager showed off a blunt bob and bangs reminiscent of the Vogue editor's iconic 'do in festive new snaps.

Edit

Delete

Apple just named a new AI chief with Google and Microsoft expertise, as John Giannandrea steps down - TechCrunch
His replacement is Amar Subramanya, a highly regarded Microsoft executive who spent 16 years at Google, most recently leading engineering for the Gemini Assistant. It's a savvy hire, given that Subramanya knows the competition intimately.

Read More

```

{
  "news": [
    {
      "url": "https://www.washingtonpost.com/national-security/2025/12/04/signal-pete-hegseth-inspector-general/",
      "imageURL": "https://www.washingtonpost.com/wp-apps/imrs.php?src=https://arc-anglerfish-washpost-prod-washpost.s3.amazonaws.com/public/1JLWVZKQHJF5XGZPQYDZ5R54I.jpg"
    }
  ],
  "users": [
    {
      "id": "6937",
      "fullName": "trail1",
      "email": "trail@gmail.com",
      "password": "1234"
    },
    {
      "id": "f278",
      "fullName": "trail2",
      "email": "trail2@gmail.com",
      "password": "1234"
    }
  ]
}

```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

engantti.vishwanatha@LAPTOP-4HS2M6B8 MINGW64 ~/Downloads/Fullstack project/News-App-Crud-main

\$ npm start
You can now view [newscrud](#) in the browser.
Local: http://localhost:3000
On Your Network: http://172.16.0.2:3000
Note that the development build is not optimized.
To create a production build, use `npm run build`.
webpack compiled successfully

```

const App = () => {
  <Routes>
    <Route path="/news" element={<ProtectedRoute>
      <NewsList />
    </ProtectedRoute>} />
    <Route path="/live" element={<ProtectedRoute>
      <LiveNews />
    </ProtectedRoute>} />
  </Routes>
}

```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

engantti.vishwanatha@LAPTOP-4HS2M6B8 MINGW64 ~/Downloads/Fullstack project/News-App-Crud-main

\$ npm start
You can now view [newscrud](#) in the browser.
Local: http://localhost:3000
On Your Network: http://172.16.0.2:3000
Note that the development build is not optimized.
To create a production build, use `npm run build`.
webpack compiled successfully

```

const Navbar = ({ toggleTheme, theme }) => {
  <Link className="nav-btn" to="/add">
    Add News
  </Link>

  <li className="nav-item">
    <Link className="nav-btn" to="/news">
      My News
    </Link>
  </li>

  <li className="nav-item">
    <Link className="nav-btn" to="/live">
      Live News
    </Link>
  </li>

  <li className="nav-item">
    <Link className="nav-btn" to="/about">
      About
    </Link>
  </li>
}

```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

engantti.vishwanatha@LAPTOP-4HS2M6B8 MINGW64 ~/Downloads/Fullstack project/News-App-Crud-main

\$ npm start
You can now view [newscrud](#) in the browser.
Local: http://localhost:3000
On Your Network: http://172.16.0.2:3000
Note that the development build is not optimized.
To create a production build, use `npm run build`.
webpack compiled successfully

RESULTS

The system successfully provides:

User Perspective

- Read breaking news instantly
- Write personal articles
- Maintain private news database
- Modify and delete news anytime
- Navigate smoothly with a modern UI

Developer Perspective

- Demonstrates React architecture
- Shows mastery of CRUD APIs
- Integrates real REST API
- Implements professional UI/UX
- Adds authentication and theme switching

CONCLUSION

The *Ready News* application demonstrates the successful development of a modern, full-stack web platform that merges real-time news consumption with user-generated content management. Through the integration of React.js, JSON Server, REST API communication, and Axios-based data handling, the system offers users a seamless, interactive, and highly responsive browsing experience. The application effectively performs CRUD operations for personalized news articles while simultaneously fetching live global headlines using external APIs, showcasing the practical implementation of both client-side and server-side functionalities. Features such as authentication, dark/light theme switching, intuitive navigation, and professional UI design contribute to an engaging, user-friendly environment that mirrors real-world digital news platforms. The project demonstrates a strong understanding of component-based architecture, state management, routing flows, and asynchronous API operations. Overall, *Ready News* stands as a robust and scalable model for news-based applications, fulfilling its objectives of delivering dynamic content, empowering user participation, and presenting a polished full-stack system ready for future enhancements.

Ready News stands as a complete modern full-stack web application featuring:

- Live API-based news
- CRUD functionalities
- Login system
- Theme toggling
- Professional UI
- Page routing
- Responsive layout

It demonstrates core full-stack concepts and practical implementation of real-world project development.

FUTURE SCOPE

- User comments
- Firebase authentication
- Push notifications
- Category-wise filtering
- AI news recommendation

PROJECT GITHUB REPO LINK:-

<https://github.com/vishwa-575/Ready-News-APP.git>

PROJECT DEMO VIDEO AND CODE VIDEO FOLDER LINK:-

https://drive.google.com/drive/folders/119_D8Pp6wVTY7bSRzWJv8-3L3wkdd6Gi?usp=sharing

