

**DEPARTMENT OF INFORMATION, COMMUNICATION & TECHNOLOGY**

**Project Title: News Portal System**

**Subject : Software Engineering**

**Program & Semester: BCA 4th Semester**

**Group Members: Shehbaz , Ujjwal Juneja, Bhawya Garg, Mahiva Bansal, Akib Javed**

**Session: 2022-23**

**Submitted to: Ms. Geeta Rawat**

**Abstract**

The news portal system plays a vital role in delivering timely and reliable news content to a wide audience. However, many existing news portal systems face challenges such as outdated technology, inefficient content management, poor user experience, limited scalability, inadequate monetization options, and ineffective social media integration.

To address these issues, a new and improved news portal system is proposed. This system aims to enhance efficiency, scalability, and user-friendliness. It incorporates modern technologies and tools to ensure a robust and reliable platform. An intuitive content management system (CMS) is implemented, enabling journalists and editors to create, edit, and publish articles seamlessly.

The user interface of the news portal system is designed to be intuitive and user-friendly, facilitating easy navigation, article search, and access to relevant information. Personalized recommendations and interactive features are incorporated to enhance the user experience and engagement.

Moreover, the system is integrated with social media platforms to enhance the reach and engagement potential of the news portal. Sharing functionalities and interactive social features are implemented to encourage user interaction and attract a wider audience.

In summary, the proposed news portal system aims to overcome the limitations of existing systems by providing a robust, scalable, and user-centric platform. By leveraging modern technologies, an efficient CMS, improved user experience, and effective monetization and social media integration, the system seeks to deliver an enhanced news reading experience for both content creators and readers.

# **System Requirements**

## **Hardware requirements:**

* Processor: Intel dual core
* Ram: 4Gb
* Storage: 16gb

## **Software requirement:**

* Vs code (any IDE)
* Chrome Browser

**Software Technology Used**

* MySQL
* JavaScript
* React
* News API

**Problem Statement**

The problem statement of a news portal system can be defined as follows:

The current news portal system lacks efficiency, scalability, and user-friendliness, leading to a subpar user experience and limited growth potential. Some of the specific issues include:

**Outdated technology:** The existing system relies on outdated technologies, making it slow, prone to crashes, and difficult to maintain. This limits the ability to deliver news content promptly and hampers the overall performance of the platform.

**Inefficient content management:** The system lacks an intuitive and efficient content management system (CMS) that allows journalists and editors to easily create, edit, and publish articles. This results in delays in publishing news and makes it challenging to maintain a consistent publishing schedule.

**Poor user interface and experience:** The user interface of the news portal system is not user-friendly, making it difficult for visitors to navigate, search for specific articles, and access relevant information. The lack of personalized recommendations and interactive features further diminishes the user experience.

**Limited scalability:** The current system struggles to handle increasing traffic and user demands. As the number of visitors and articles grows, the system experiences performance issues, leading to slow loading times, frequent errors, and potential downtime during peak periods.

**Ineffective social media integration:** The system fails to effectively integrate with social media platforms, reducing the reach and engagement potential of the news portal. This hampers the ability to attract and retain a wider audience through social sharing and interactions.

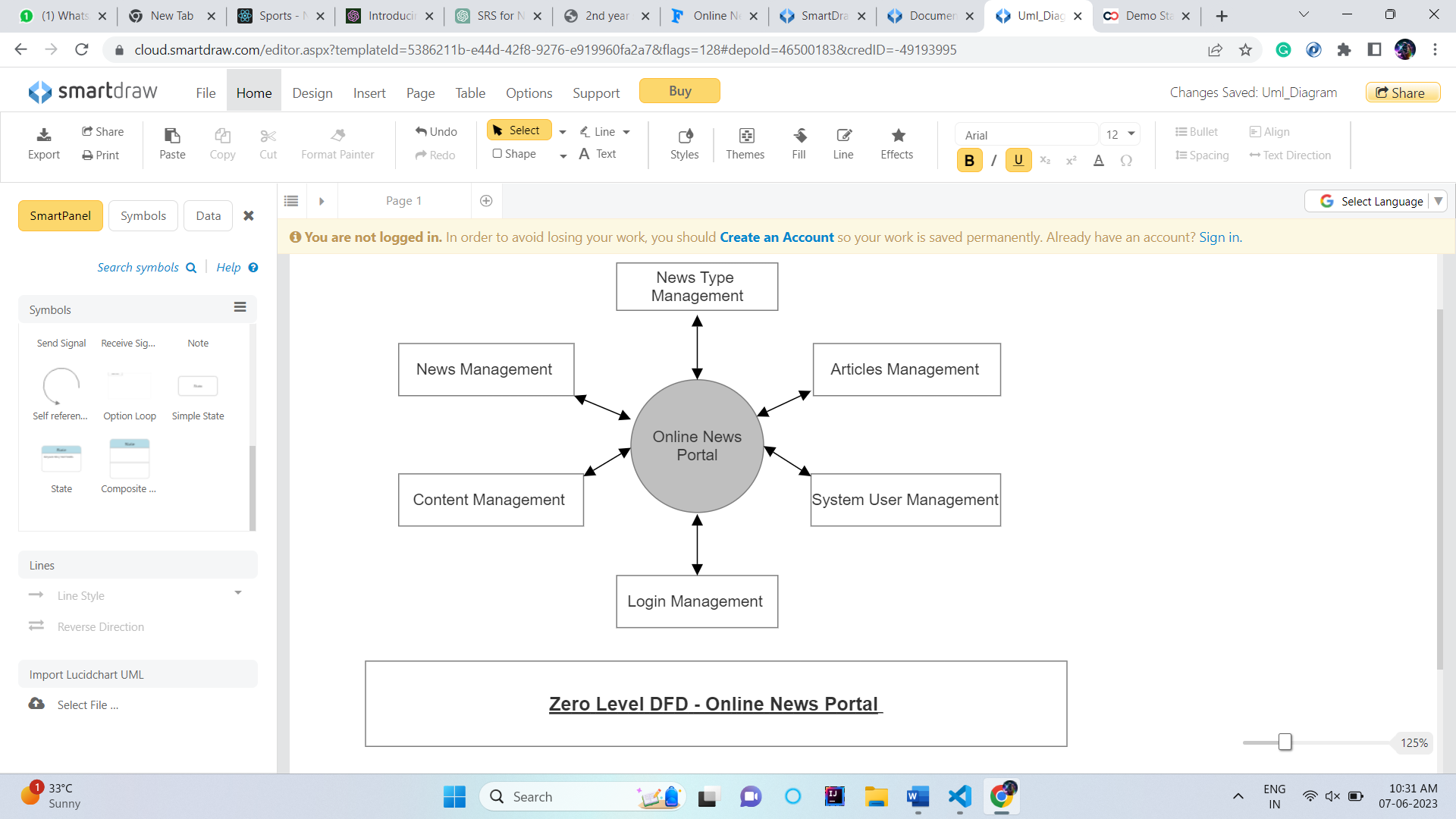
The objective is to develop a robust and user-centric news portal system that addresses these challenges and provides an efficient, scalable, and enjoyable experience for both content creators and readers.

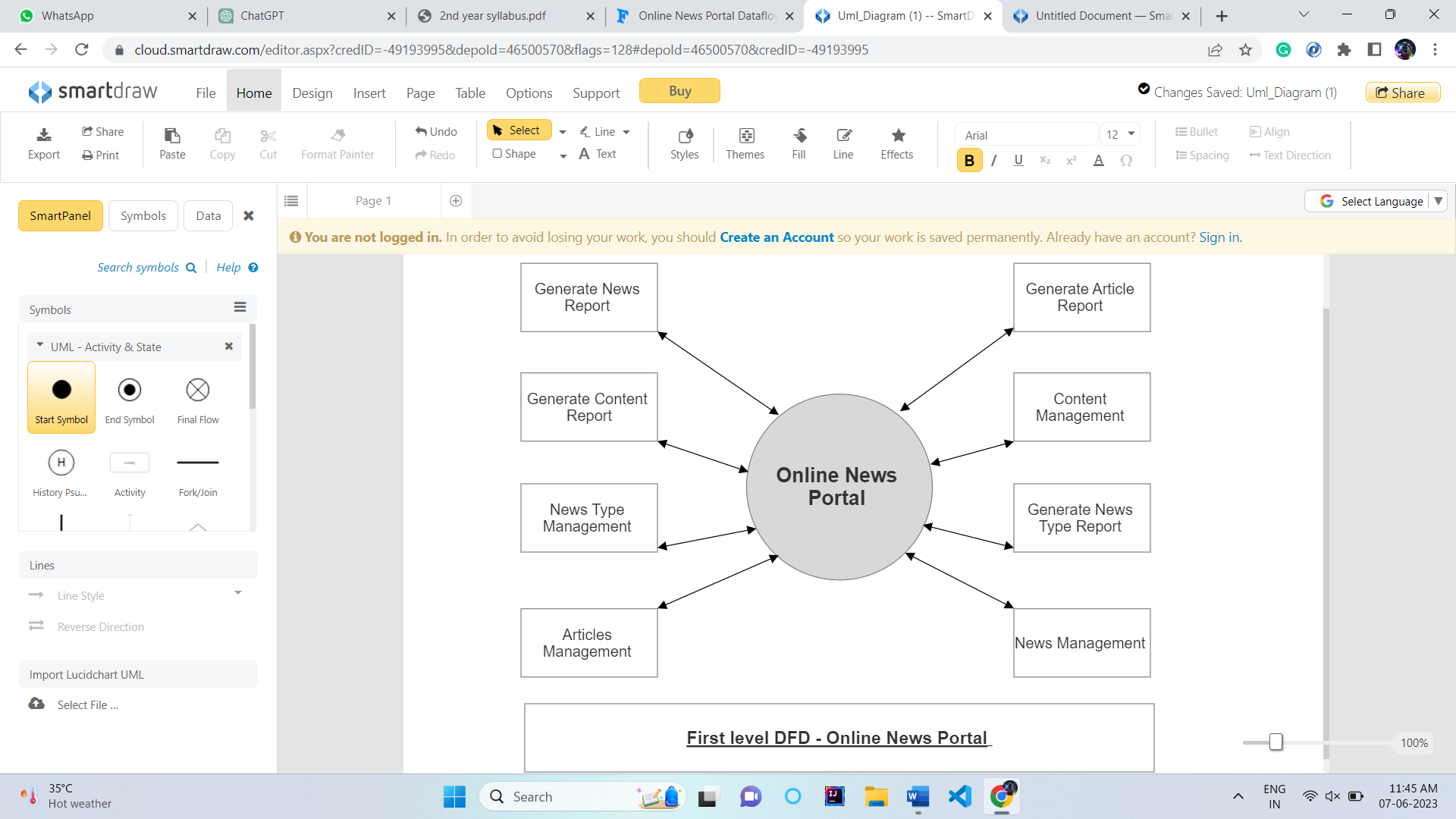
**Feature of NewsWave Portal:**

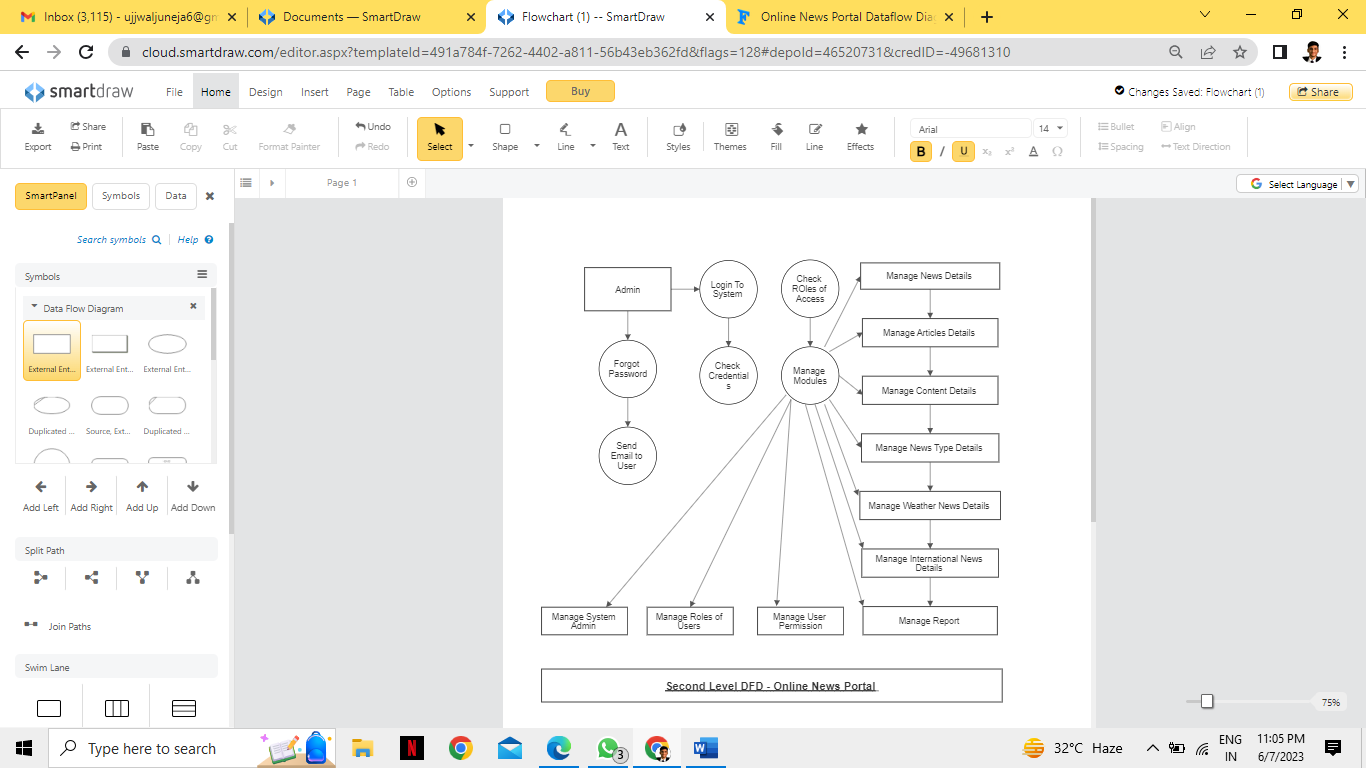
To address the aforementioned problems, the proposed news portal aims to incorporate various features to enhance the user experience and improve overall efficiency. The following key features will be implemented:

* **React Technology:** The news portal will utilize React, a popular JavaScript library, for building user interfaces. React is component-based architecture enables efficient rendering and updating of content, resulting in a seamless and responsive user experience.
* **Infinite Scroll:** Instead of relying on pagination, the news portal will implement infinite scrolling, allowing users to continuously scroll through articles without the need to load new pages. This feature ensures a smoother browsing experience, eliminates the need for multiple page loads, and makes it easier for users to explore a large volume of news content.
* **Single-Page Application (SPA):** The news portal will adopt a single-page application architecture, where all content and functionality are loaded on a single web page. This approach eliminates the need for page refreshes during navigation, providing a faster and more fluid user experience.

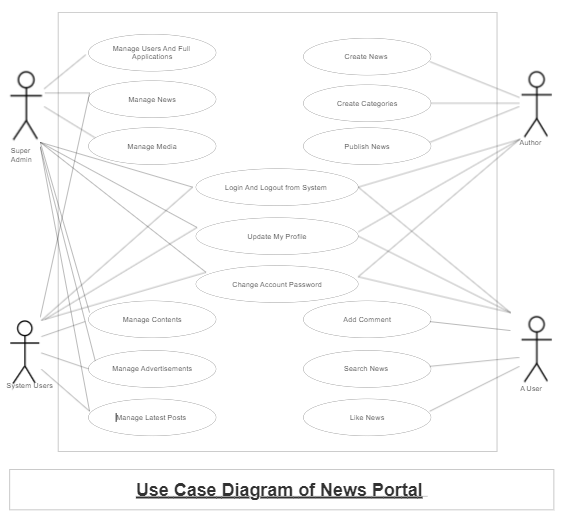
**DFD Diagram of New Website**



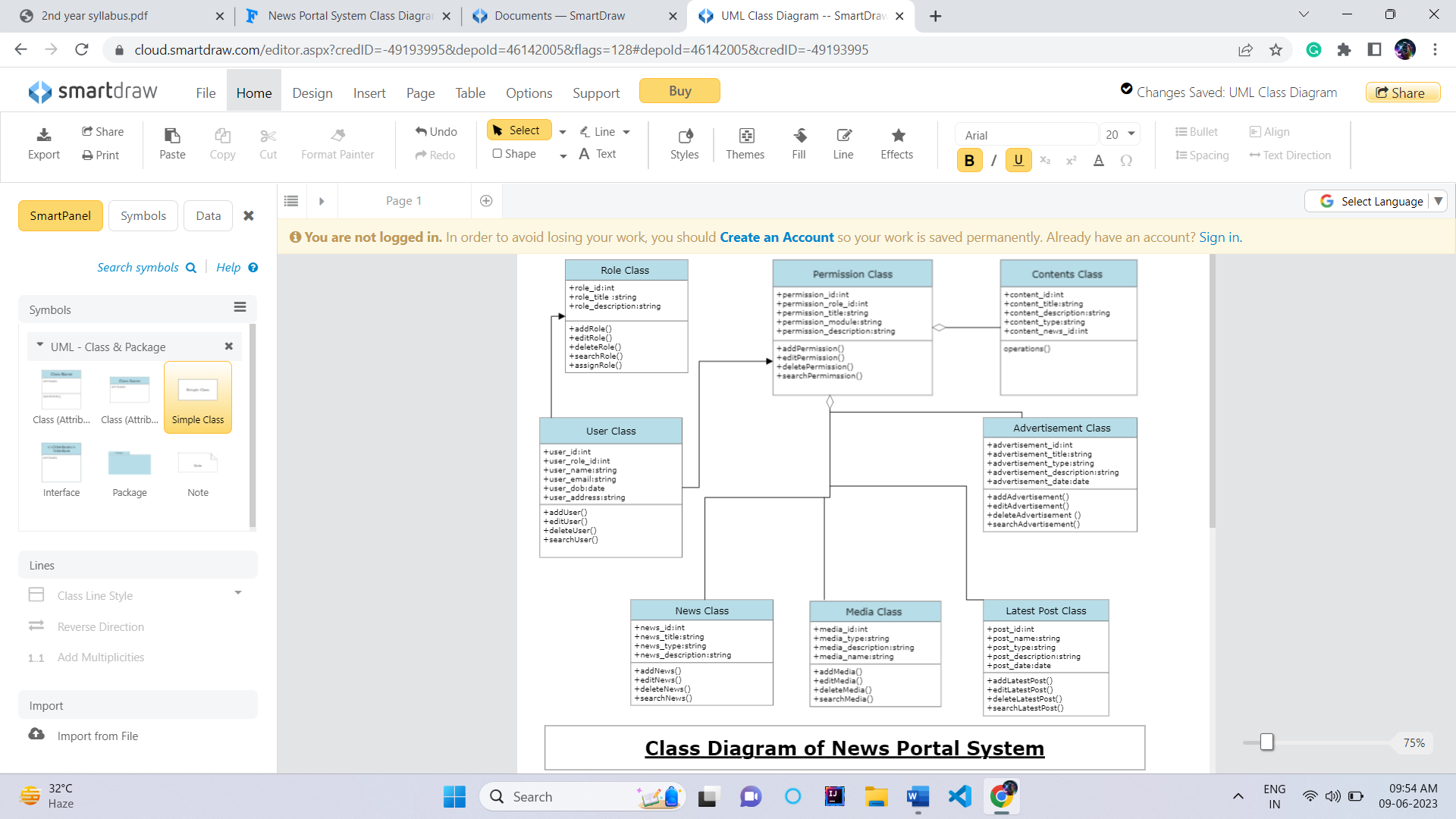




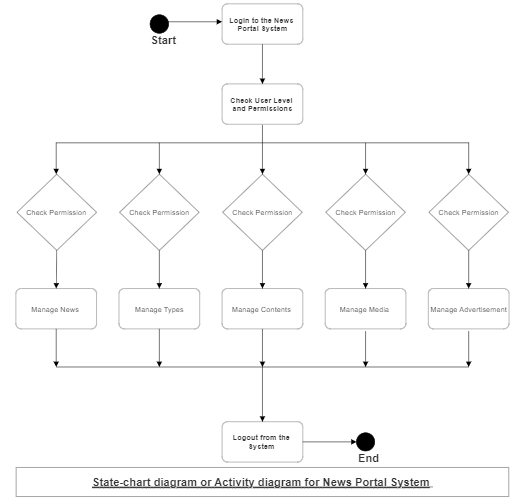
**Use Case Diagram for News Website**

****

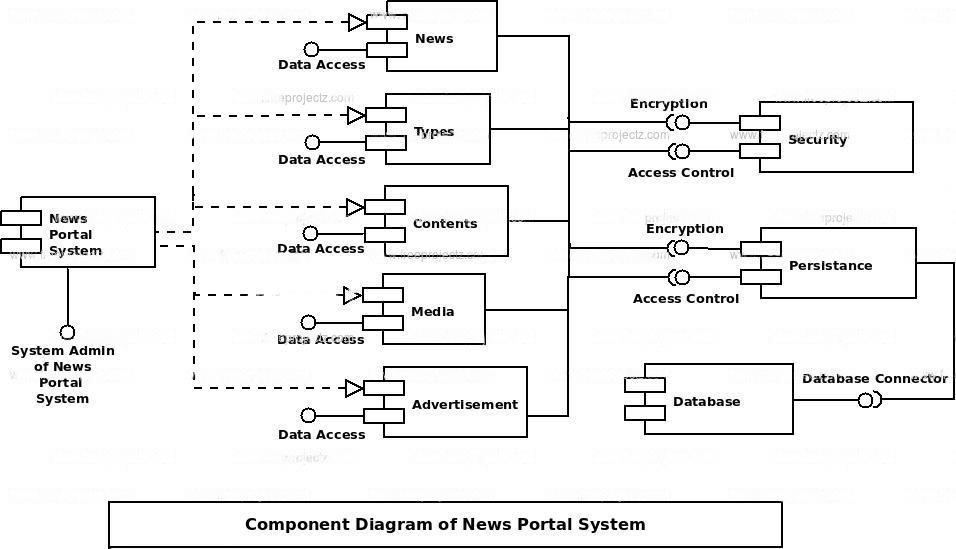
**Class Diagram of News Portal System**



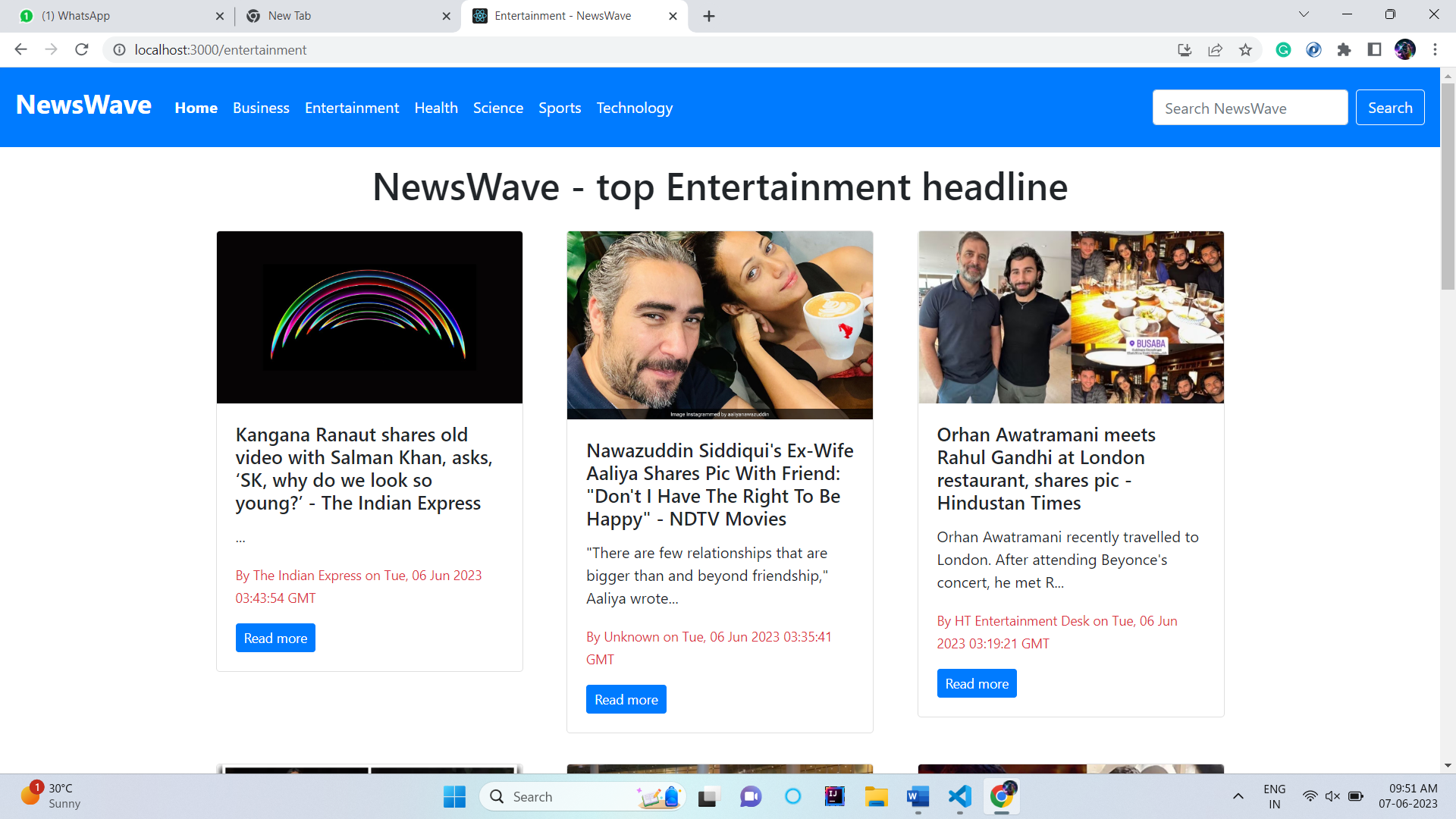
**State Chart Diagram or Activity Diagram of News Portal System**

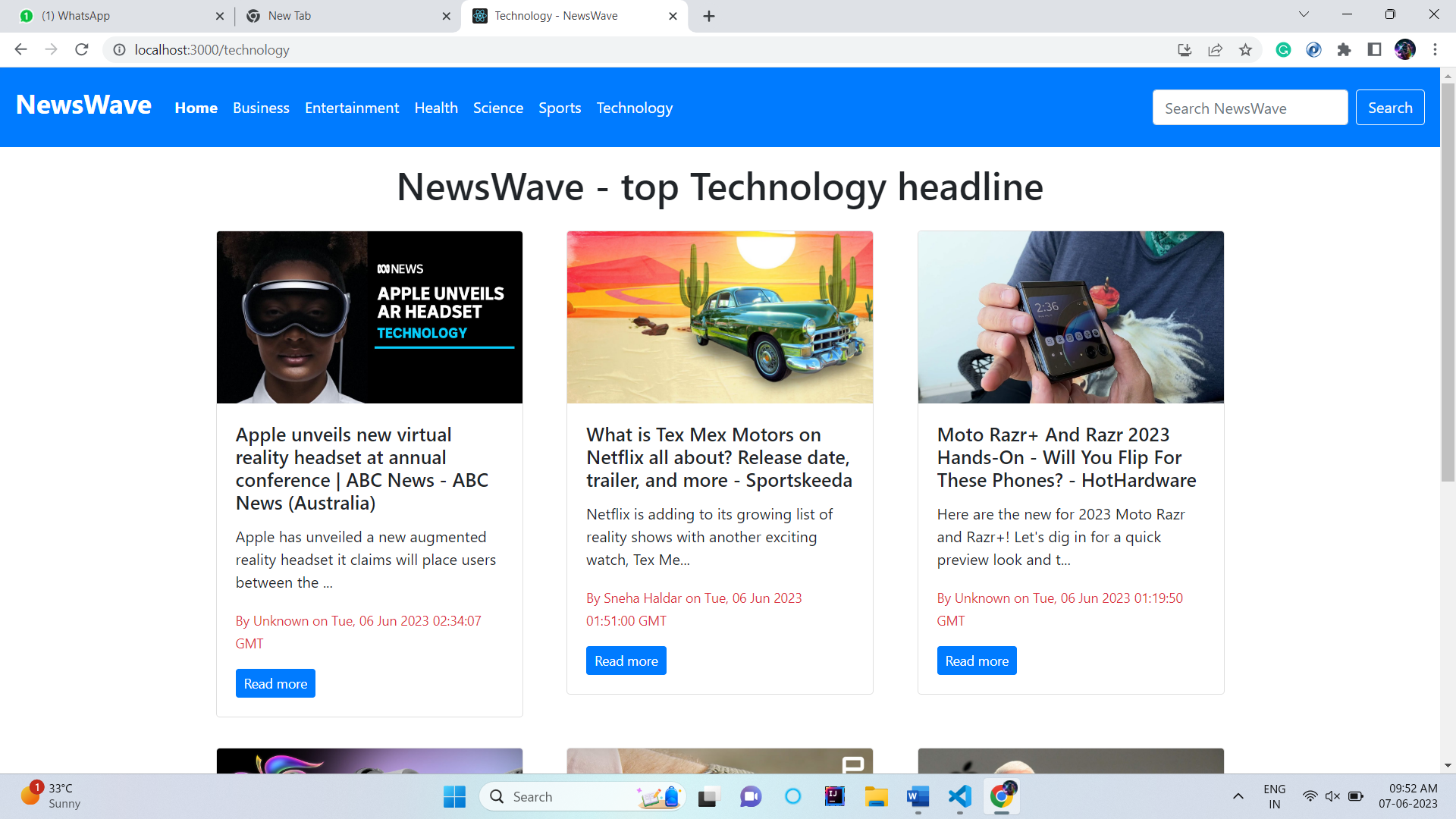
****

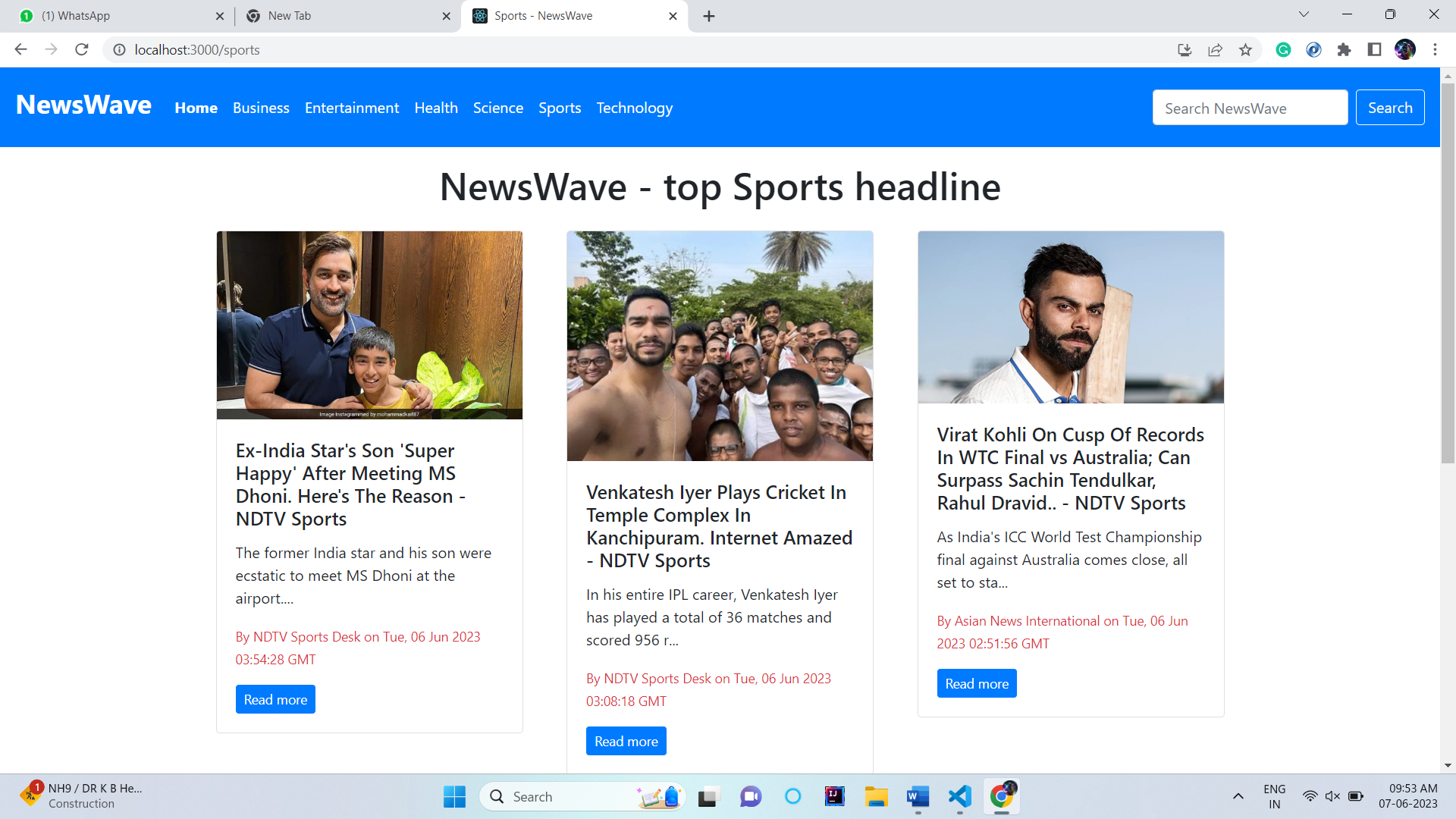
**Component of News Portal System**



**Output**







**Source Code**

**NavBar.js**

import React from "react";

import { Link } from "react-router-dom";

const NavBar=()=>{

    return (

      <div>

        <nav className="navbar fixed-top navbar-expand-lg navbar-dark bg-primary p-3">

          <Link className="navbar-brand" to="/">

            <h3>

              <b>NewsWave</b>

            </h3>

          </Link>

          <button

            className="navbar-toggler"

            type="button"

            data-toggle="collapse"

            data-target="#navbarSupportedContent"

            aria-controls="navbarSupportedContent"

            aria-expanded="false"

            aria-label="Toggle navigation"

          >

            <span className="navbar-toggler-icon"></span>

          </button>

          <div className="collapse navbar-collapse" id="navbarSupportedContent">

            <ul className="navbar-nav mr-auto">

              <li className="nav-item active">

                <Link className="nav-link" to="/">

                  <b>Home</b>

                </Link>

              </li>

              <li className="nav-item active">

                <Link className="nav-link" to="/business">

                  Business

                </Link>

              </li>

              <li className="nav-item active">

                <Link className="nav-link" to="/entertainment">

                  Entertainment

                </Link>

              </li>

              <li className="nav-item active">

                <Link className="nav-link" to="/health">

                  Health

                </Link>

              </li>

              <li className="nav-item active">

                <Link className="nav-link" to="/science">

                  Science

                </Link>

              </li>

              <li className="nav-item active">

                <Link className="nav-link" to="/sports">

                  Sports

                </Link>

              </li>

              <li className="nav-item active">

                <Link className="nav-link" to="/technology">

                  Technology

                </Link>

              </li>

            </ul>

            <form className="form-inline my-2 my-lg-0">

              <input

                className="form-control mr-sm-2"

                type="search"

                placeholder="Search NewsWave"

                aria-label="Search"

              />

              <button

                className="btn btn-outline-light my-2 my-sm-0"

                type="submit"

              >

                Search

              </button>

            </form>

          </div>

        </nav>

      </div>

    );

}

export default NavBar;

**News.js**

import React, { useEffect, useState } from "react";

import NewsItem from "./NewsItem";

import Spinner from "./Spinner";

import PropTypes from "prop-types";

import InfiniteScroll from "react-infinite-scroll-component";

const News = (props) => {

  const [articles, setarticles] = useState([])

  const [loading, setLoading] = useState(true)

  const [page, setpage] = useState(1)

  const [totalResults, settotalResults] = useState(0)

  const capitalizeFirstLetter = (string) => {

    return string.charAt(0).toUpperCase() + string.slice(1);

  };

  const updatedNews = async () => {

    props.setProgress(10);

    const url = `https://newsapi.org/v2/top-headlines?country=${props.country}&category=${props.category}&apiKey=94b536f877744890b4d6f0e03284bbe8&page=${page}&pageSize=${props.pageSize}`;

    setLoading(true)

    let data = await fetch(url);

    props.setProgress(30);

    let parsedData = await data.json();

    props.setProgress(70);

    setarticles(parsedData.articles);

    settotalResults(parsedData.totalResults);

    setLoading(false);

    props.setProgress(100);

  };

  useEffect(() => {

    document.title =`${capitalizeFirstLetter(props.category)} - NewsWave`

    updatedNews();

    // eslint-disable-next-line

  },[])

  const fetchMoreData = async () => {

    const url = `https://newsapi.org/v2/top-headlines?country=${props.country}&category=${props.category}&apiKey=94b536f877744890b4d6f0e03284bbe8&page=${page+1}&pageSize=${props.pageSize}`;

    setpage(page+1)

    let data = await fetch(url);

    let parsedData = await data.json();

    setarticles(articles.concat(parsedData.articles))

    settotalResults(parsedData.totalResults)

  };

  return (

    <div className="container my-3">

      <h1 className="text-center" style={{marginTop:'100px'}}>

        NewsWave - top {capitalizeFirstLetter(props.category)} headline

      </h1>

      {loading && <Spinner />}

      <InfiniteScroll

        dataLength={articles.length}

        next={fetchMoreData}

        hasMore={articles.length !== totalResults}

        loader={!(<Spinner />) ? <Spinner /> : ""}

      >

        <div className="container">

          <div className="row">

            {articles.map((element) => {

              return (

                <div className="col-md-4" key={element.url}>

                  <NewsItem

                    title={element.title ? element.title : ""}

                    description={

                      element.description ? element.description.slice(0, 90): "" }

                    imageUrl={element.urlToImage}

                    newsUrl={element.url}

                    author={element.author}

                    date={element.publishedAt}

                  />

                </div>

              );

            })}

          </div>

        </div>

      </InfiniteScroll>

    </div>

  );

};

News.defaultProps = {

  country: "in",

  pageSize: 6,

  category: "general",

};

News.propsTypes = {

  country: PropTypes.string,

  pageSize: PropTypes.number,

  category: PropTypes.string,

};

export default News;

**Newsitem.js**

import React from "react";

const NewsItem = (props) => {

  let { title, description, imageUrl, newsUrl, author, date } = props;

  return (

    <div className="my-3  m-2">

      <div className="card">

        <img

          src={

            !imageUrl

              ? "https://imgeng.jagran.com/images/2023/may/wwdc%2020231684416004507.jpg"

              : imageUrl}

          className="card-img-top"

          alt="..."

        />

        <div className="card-body">

          <h5 className="card-title">{title}</h5>

          <p className="card-text">{description}...</p>

          <p className="card-text">

            <small className="text-danger">

              By {!author ? "Unknown" : author} on{" "}

              {new Date(date).toGMTString()}{" "}

            </small>

          </p>

          <a

            rel="noreferrer"

            href={newsUrl}

            target="\_blank"

            className="btn btn-sm btn-primary"

          >

            Read more

          </a>

        </div>

      </div>

    </div>

  );

};

export default NewsItem;

**Spinner.gif**

import loading from "./Spinner-3.gif"

const Spinner=()=>{

    return (

      <div className='text-center' >

        <img src={loading} alt="loading" />

      </div>

    )

}

export default Spinner

**App.js**

import './App.css';

import React, { useState} from 'react'

import NavBar from './components/NavBar';

import News from './components/News';

import LoadingBar from 'react-top-loading-bar'

import { BrowserRouter as Router, Routes,Route,

} from "react-router-dom";

const App =()=>{

  const pageSize=6;

  const apiKey= '94b536f877744890b4d6f0e03284bbe8'

  const [progress,setProgress] = useState(0)

return (

    <>

    <Router>

    <NavBar/>

    <LoadingBar

      height={3}

      color='#f11946'

      progress={progress}

    />

    <Routes>

    <Route exact path="/" element={<News setProgress={setProgress} apiKey={apiKey} key="general" pageSize={pageSize} country="in" category="general"/>} ></Route>

    <Route exact path="/business" element={<News setProgress={setProgress} apiKey={apiKey} key="business" pageSize={pageSize} country="in" category="business"/>}></Route>

    <Route exact path="/entertainment" element={<News setProgress={setProgress} apiKey={apiKey} key="entertainment" pageSize={pageSize} country="in" category="entertainment"/>} ></Route>

    <Route exact path="/health" element={<News setProgress={setProgress} apiKey={apiKey} key="health" pageSize={pageSize} country="in" category="health"/>}></Route>

    <Route exact path="/science" element={<News setProgress={setProgress} apiKey={apiKey} key="science" pageSize={pageSize} country="in" category="science"/>}></Route>

    <Route exact path="/sports" element={<News setProgress={setProgress} apiKey={apiKey} key="sports" pageSize={pageSize} country="in" category="sports"/>}></Route>

    <Route exact path="/technology" element={<News setProgress={setProgress} apiKey={apiKey} key="technology" pageSize={pageSize} country="in" category="technology"/>}></Route>

    </Routes>

    </Router>

    </>

  )

}

export default App;

**Perform Measurement of complexity with Halstead Metrics for chosen system**

n1 – Number of unique operands - 2

n2- Number of unique operators - 4

n= n1+n2=6

N1 - Frequency of operands – 20

N2 – Frequency of operators – 12

N=N1+N2

N= 32

Volume= Nlog2 n

Volume=32 log2 6= 82.7188000231

D=(n1/2)\* (N2/n2)

D= (2/2)\*(12/4)

D=1\*3

D=3

**Learning Outcomes**

**Requirement gathering and analysis:** Students will gain experience in conducting thorough requirement gathering and analysis to understand the needs and expectations of stakeholders, including news editors, journalists, and readers. They will learn how to translate these requirements into software specifications.

**System design and architecture:** Students will learn how to design the architecture of a news portal system, considering factors such as scalability, performance, security, and maintainability. They will gain experience in selecting appropriate technologies, frameworks, and database models to support the system requirements.

**Software development skills:** Students will develop proficiency in programming languages, frameworks, and tools used in building a news portal system. They will learn how to implement features such as user authentication, content management, search functionality, social media integration, and monetization options.

**Testing and quality assurance:** Students will gain skills in testing methodologies and quality assurance practices to ensure the reliability, functionality, and security of the news portal system. They will learn how to design and execute test cases, perform bug tracking and resolution, and conduct system integration testing.

**Conclusion**

In conclusion, a well-designed and efficiently implemented news portal system is essential for delivering timely and reliable news content to a wide audience. It should address the challenges of outdated technology, inefficient content management, poor user experience, limited scalability, inadequate monetization options, and ineffective social media integration.

By developing a robust news portal system, organizations can benefit from enhanced efficiency, scalability, and user-friendliness. The system should incorporate modern technologies, an intuitive content management system (CMS), and a user-friendly interface to streamline news publishing and improve the overall user experience.

Integrating the system with social media platforms will expand its reach and engagement potential, enabling wider audience acquisition and interaction. Additionally, the system should incorporate analytics and data-driven decision-making to gain insights into user behavior, preferences, and trends, thereby improving content delivery and user engagement.

**Reference**

<https://newsapi.org/>

<https://newsapi.org/v2/top-headlines?apiKey=94b536f877744890b4d6f0e03284bbe8&q=usa>