INSIGHTSTREAM: NAVIGATE THE NEWS LANDSACPE

NAAN MUDHALVAN PROJECT REPORT

Bachelor of Computer Science

Submitted by

TEAM LEADER

B. SANTHOSH KUMAR (222209332)   **santhosk1019@gmail.com**

TEAM MEMBERS

V. PRAVEEN (222209326)   **praveenvenkat073@gmail.com**

S. SANJAY KUMARAN (222209331)   **sanjaykumaran236@gmail.com**

S. SANTHOSH (222209333)   **santhosh080405@gmail.com**

DEPARTMENT OF COMPUTER SCIENCE

TAGORE COLLEGE OF ARTS AND SCIENCE

(Affiliated to the University of Madras)

MARCH – 2025

TABLE OF THE CONTENT

6.	**SYSTEM TESTING** 6.1 FUNCTIONAL TESTING 6.2 NON FUNCTIONAL TESTING	
7.	**DEVELPMENT PROCESS**	
8.	**FEATURES OVERVIEW**	
9.	**CONCLUSION**	
10.	**APPENDIX** • SCREEN LAYOUTS • SOURCE CODE	

ABSTRACT

InsightStream is an AI-powered, React-based news aggregator designed to provide users with a personalized, efficient, and trustworthy news consumption experience. With the rapid increase in digital information, users often struggle with news overload, misinformation, and biased reporting. InsightStream addresses these challenges by curating a dynamic and user-friendly platform that fetches real-time news from reliable sources and delivers it in a structured, accessible manner. The application integrates AI-powered features such as personalized content recommendations, automated news summarization, sentiment analysis, and category-based filtering. Users can search, bookmark, and receive real-time updates tailored to their interests. Additionally, the platform offers multi-language support, accessibility features, and robust authentication for a secure and inclusive user experience. Insight Stream is built using React.js for the frontend, Node.js with Express.js for the backend, and MongoDB for data management. It leverages third-party newsAPIs such as NewsAPI and Open AI for AI-driven functionalities. The development process includes UI/UX design, API integration, rigorous testing, and cloud-based deployment. Targeting a diverse audience, including professionals, students, and general readers, Insight Stream redefines digital news consumption by combining the power of AI with an intuitive, responsive interface. Future enhancements will include social media integrations, advanced fake news detection, and voice-based news summarization to further improve accessibility and reliability.

SYNOPSIS

CHAPTER 1
SYNOPSIS

1. Introduction

The exponential rise in digital content has made it increasingly difficult for users to stay informed without feeling overwhelmed. Traditional news platforms lack personalization, making it hard for users to find content that aligns with their interests. InsightStream is a React-based news aggregator that leverages AI to provide personalized and real-time news updates, ensuring users access relevant and credible information effortlessly.

2. Objectives

- Develop a dynamic and intuitive news aggregator platform.
- Use AI to personalize content recommendations and summarize news.
- Offer real-time news updates from trusted sources.
- Provide accessibility features such as multi-language support and dark mode.
- Ensure user data security through robust authentication measures.

3. Features

1. **Personalized News Feed** – AI-curated news based on user preferences.
2. **Category-Based Filtering** – Users can filter news by topics like technology, politics, and sports.
3. **Real-Time Updates** – Fetching the latest news from reliable sources.
4. **Search & Bookmarking** – Allowing users to find and save articles.
5. **Multi-Language Support** – Expanding accessibility to a global audience.
6. **AI-Powered Summarization** – Quick and concise news summaries.
7. **User Authentication & Security** – Secure login using Firebase/Auth0.
8. **Trending Topics & Notifications** – Highlighting global and regional news trends.

4. Technology Stack

- **Frontend:** React.js, Redux, Tailwind CSS.
- **Backend:** Node.js with Express.js.
- **Database:** MongoDB.

- **APIs:** NewsAPI, Google News API, OpenAI API.
- **Authentication:** Firebase/Auth0 for secure login.

5. Development Methodology

- **Requirement Analysis:** Understanding user needs.
- **UI/UX Design:** Wireframing and prototyping.
- **Implementation:** Developing frontend and backend components.
- **Testing:** Unit, integration, and usability testing.
- **Deployment:** Hosting on cloud services like AWS, Vercel, or Netlify.
- **Maintenance:** Continuous updates and feature enhancements.

6. AI Integration

- **News Summarization:** AI extracts key points from articles.
- **Personalized Content Recommendations:** AI adapts based on user behavior.
- **Sentiment Analysis:** AI determines article tone (positive, negative, neutral).

7. Security & Privacy

- **Role-Based Access Control (RBAC):** Different access levels for users.
- **Data Encryption:** Secure storage of user preferences and bookmarks.
- **Privacy Compliance:** Adhering to GDPR and industry standards.

8. Target Audience & Use Cases

- **News Enthusiasts:** Individuals seeking a tailored news experience.
- **Professionals:** Users looking for quick, industry-specific updates.
- **Students & Researchers:** Those needing access to reliable sources for studies.

9. Conclusion & Future Enhancements

InsightStream aims to revolutionize how users consume news by integrating AI, real-time updates, and personalization into a seamless user experience. Future developments will focus on expanding social media sharing options, implementing

advanced fake news detection, and introducing voice-based news summarization to enhance accessibility. With these innovations, InsightStream will continue to be a go-to platform for informed and efficient news consumption.

1.1 Scope of the Project

InsightStream is designed as a **web-based application** that will cater to a wide audience, including students, professionals, journalists, and general readers. The project scope includes:

- **Frontend Development:** Building a responsive UI using React.js.
- **Backend Development:** Implementing data handling with Node.js and Express.js.
- **AI Integration:** Incorporating AI models for summarization, recommendations, and sentiment analysis.
- **News API Integration:** Fetching news from sources like Google News API and NewsAPI.
- **Security Measures:** Implementing authentication and data protection protocols.
- **Scalability & Future Enhancements:** Adding features such as social media integration and fake news detection.

1.2 Background & Problem Statement

In today's digital era, the traditional methods of consuming news, such as newspapers and television broadcasts, are being rapidly replaced by online platforms. However, these platforms often present challenges such as:

- **Information Overload:** A vast number of articles from multiple sources can be overwhelming.
- **Misinformation & Fake News:** The spread of unverified or misleading news has increased.
- **Lack of Personalization:** Many platforms do not tailor content to user preferences.
- **Time Constraints:** Users struggle to read lengthy articles in their busy schedules.

InsightStream aims to overcome these challenges by implementing AI-driven content curation, filtering fake news sources, and summarizing articles for quick consumption.

1.3 Objectives of the Project

The primary goals of InsightStream include:

- Developing a **user-friendly** and **intuitive** news aggregation platform.
- Implementing **AI-based personalized recommendations** to filter news relevant to users.
- Providing **real-time updates** to ensure users access the latest news.
- Offering **AI-powered summarization** for quick content consumption.
- Ensuring **multi-language support** for a global audience.
- Enhancing **security and authentication** to protect user data and preferences.

1.4 Significance of the Project

InsightStream brings significant advantages to digital news consumption by:

- **Reducing Information Overload:** AI-curated news ensures relevant content delivery.
- **Enhancing Accessibility:** Multi-language support and accessibility features improve user inclusivity.
- **Providing Trustworthy News:** Filtering out unreliable sources helps combat misinformation.
- **Saving Time:** AI-powered summaries allow users to grasp essential news quickly.

SYSTEM SPECIFICATION

CHAPTER 2

SYSTEM SPECIFICATION

This system specification outlines the hardware, software, security, and network requirements necessary for the smooth development and deployment of the InsightStream application. Future enhancements may require additional resources as the platform scales.

2.1 Hardware Requirements

The following hardware specifications are recommended for optimal performance of the InsightStream application:

For Development Environment:

- **Processor:** Intel Core i5 (8th Gen or above) / AMD Ryzen 5 or higher
- **RAM:** Minimum 8GB (16GB recommended for smooth development)
- **Storage:** SSD with at least 256GB (512GB recommended)
- **Graphics:** Integrated GPU (Dedicated GPU for faster rendering preferred)
- **Network:** Stable internet connection for API calls and cloud services
- **Display:** Minimum 1080p resolution monitor

For Deployment Server:

- **Processor:** Intel Xeon / AMD EPYC (or cloud equivalent)
- **RAM:** Minimum 16GB (32GB recommended for handling high traffic)
- **Storage:** NVMe SSD with at least 500GB
- **Database Storage:** Separate storage for MongoDB, minimum 100GB
- **Network Bandwidth:** High-speed internet connection with at least 100Mbps
- **Cloud Hosting:** AWS, Google Cloud, or Azure-based deployment

2.2 Software Requirements

The following software components are essential for the development and deployment of InsightStream:

Frontend Technologies:

- **Framework:** React.js (latest stable version)
- **State Management:** Redux (or Context API if required)
- **Styling:** Tailwind CSS / Material UI
- **Package Manager:** npm / yarn
- **Browser Compatibility:** Chrome, Firefox, Edge, Safari (latest versions)

Backend Technologies:

- **Server Framework:** Node.js with Express.js
- **Database:** MongoDB (NoSQL database for scalability and flexibility)
- **API Integration:** NewsAPI / Google News API
- **Authentication:** Firebase Auth / Auth0
- **AI/ML Services:** OpenAI API (for AI-based summarization and sentiment analysis)
- **Security:** JWT (JSON Web Token) for user authentication

Development Tools:

- **Code Editor:** Visual Studio Code / JetBrains WebStorm
- **Version Control:** GitHub / GitLab / Bitbucket
- **API Testing:** Postman / Thunder Client
- **Containerization:** Docker (for scalable deployment)
- **CI/CD Tools:** GitHub Actions / Jenkins / Netlify / Vercel for automated deployment

2.3 Network Requirements

- **Minimum Internet Speed:** 10Mbps for development, 100Mbps+ for deployment
- **Cloud Hosting Requirements:** Secure API endpoints and database access control
- **SSL Certificate:** Required for HTTPS-enabled secure communication

SYSTEM ANALYSIS

CHAPTER 3

SYSTEM ANALYSIS

System analysis is a crucial phase in the development of InsightStream, where we examine the functional and non-functional requirements, system feasibility, and architectural design. This ensures that the project meets its objectives effectively and efficiently while providing a seamless user experience.

3.1 Existing System

Traditional news aggregation platforms face the following issues:

- **Lack of Personalization:** Users receive generalized news instead of tailored content.
- **Information Overload:** Large amounts of news articles without proper filtering.
- **Misinformation & Fake News:** Difficulty in distinguishing reliable sources.
- **Inefficient User Experience:** Slow navigation, unstructured content, and lack of AI-driven features.

3.2 Proposed System

- **AI-Powered Personalization:** Tailored content recommendations based on user preferences.
- **Real-Time Updates:** Fetching the latest news from trusted sources.
- **Fake News Filtering:** AI-driven analysis to detect misinformation.
- **Summarization & Sentiment Analysis:** Quick insights into lengthy articles.
- **Multi-Language Support:** News accessible to a wider audience.

3.3 Feasibility Study

A feasibility study evaluates InsightStream's viability in terms of technical, operational, and financial aspects.

3.3.1 Technical Feasibility

- Utilizes modern technologies like **React.js, Node.js, MongoDB, and AI APIs**.
- Scalable architecture for high-performance news aggregation.
- Secure authentication using Firebase/Auth0.

3.3.2 Operational Feasibility

- User-friendly UI with intuitive navigation and customization.
- AI-enhanced recommendations to optimize news consumption.
- Efficient content filtering and news summarization.

3.3.3 Economic Feasibility

- **Cost-effective development** using open-source technologies.
- **Minimal operational costs** with cloud-based deployment.
- **Potential revenue models:** Advertisements, premium subscriptions, and AI-powered insights.

SYSTEM DESIGN

CHAPTER 4

SYSTEM DESIGN

System design is a critical phase in the development of InsightStream, focusing on defining the system architecture, data flow, component interactions, and security aspects. This ensures that the platform is scalable, efficient, and user-friendly.

4.1 Architectural Design

InsightStream follows a **three-tier architecture**, which includes:

1. **Presentation Layer (Frontend)** – User interface built with React.js.
2. **Application Layer (Backend)** – Handles business logic using Node.js and Express.js.
3. **Data Layer (Database & External APIs)** – Manages data using MongoDB and fetches news from APIs like NewsAPI or Google News API.

4.1.1 System Architecture Diagram
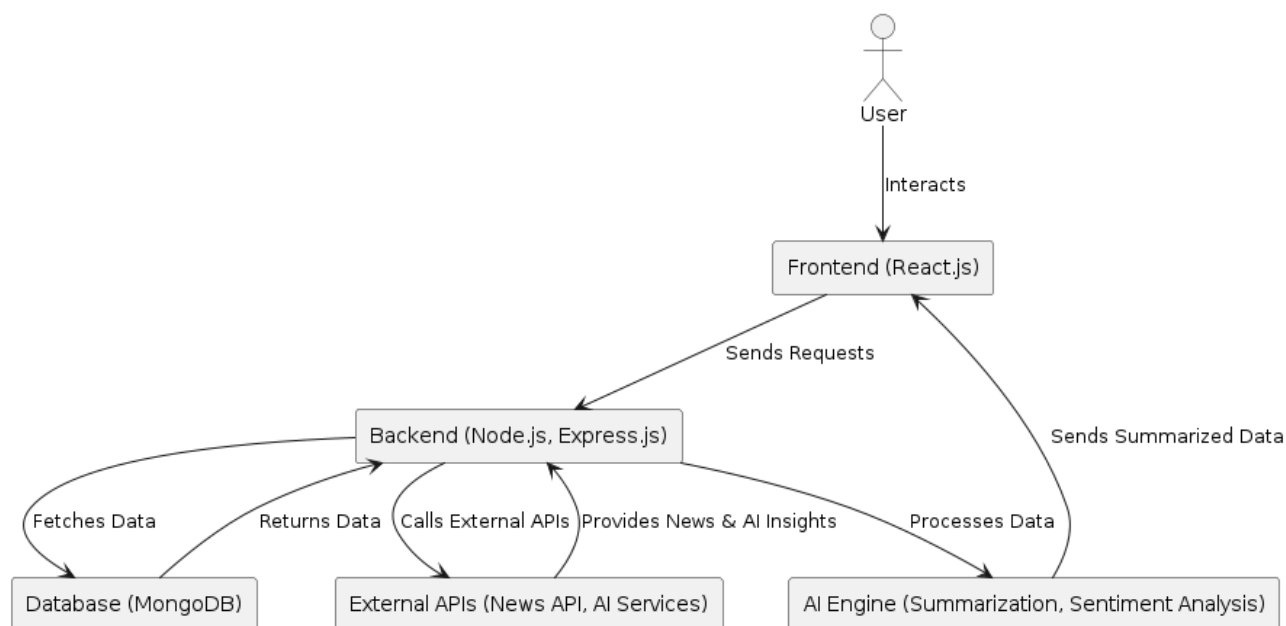
4.2 UML DIAGRAM

A **Use Case Diagram** represents the interaction between users and the system, defining user roles and their accessible functionalities. A **Class Diagram** shows the system structure with its classes, attributes, methods, and relationships, helping in object-oriented design. A **Sequence Diagram** illustrates the step-by-step flow of interactions between system components, detailing message exchanges over time. An **Activity Diagram** visualizes the system workflow, representing activities, decisions, and parallel processes to model business logic effectively.

4.2.1. Use Case Diagram

A **Use Case Diagram** visually represents the interactions between users (actors) and the system. It highlights various functionalities offered by the system and how users interact with them. This diagram helps in understanding system requirements and user roles.
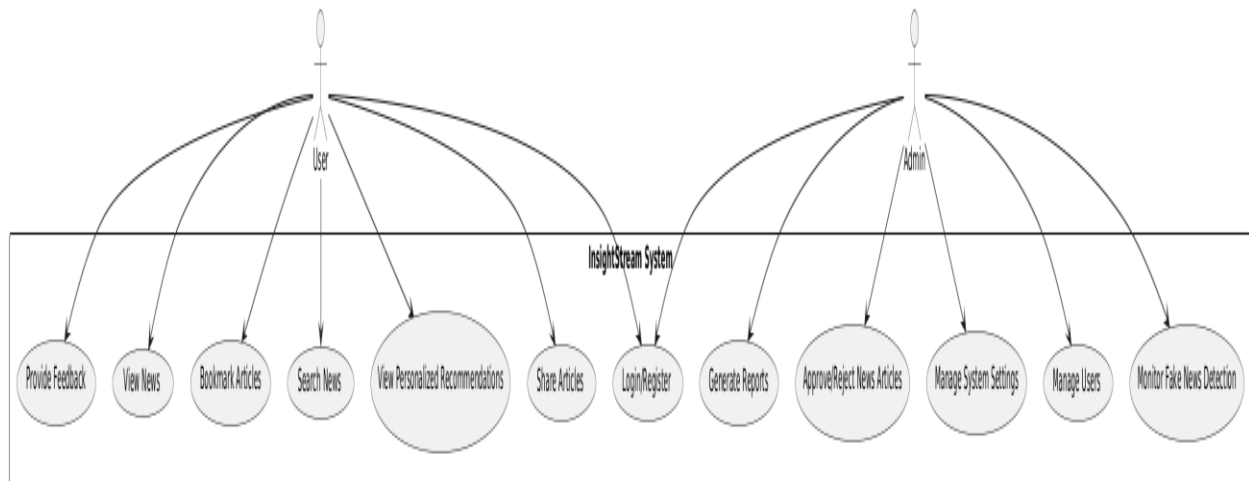
4.2.2. Class Diagram

A **Class Diagram** illustrates the structure of the system by showing its classes, attributes, methods, and relationships. It provides a blueprint for object-oriented design and helps developers understand data organization.
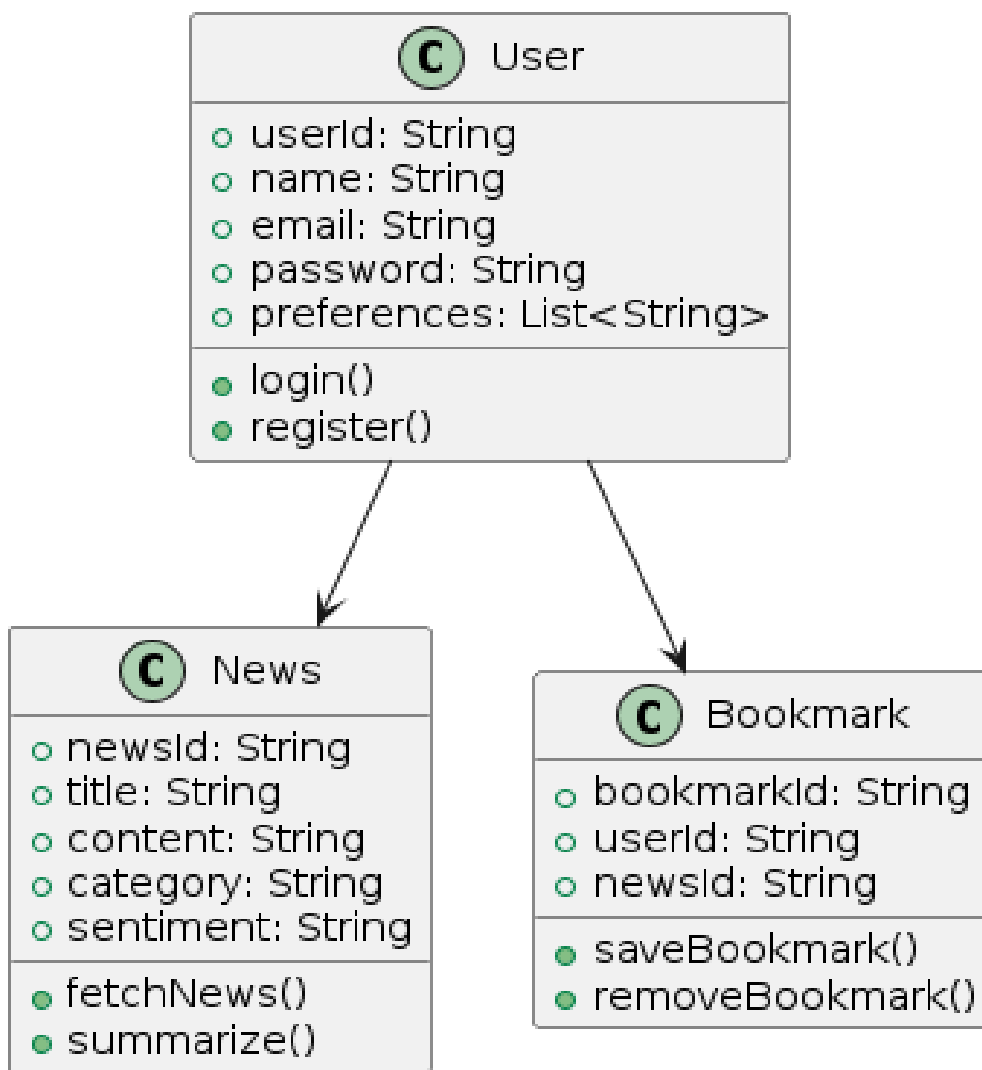
4.3.3. Sequence Diagram

A **Sequence Diagram** represents the flow of interactions between different components of the system in a step-by-step manner. It describes how processes operate in a sequential order and how messages are exchanged between system elements.
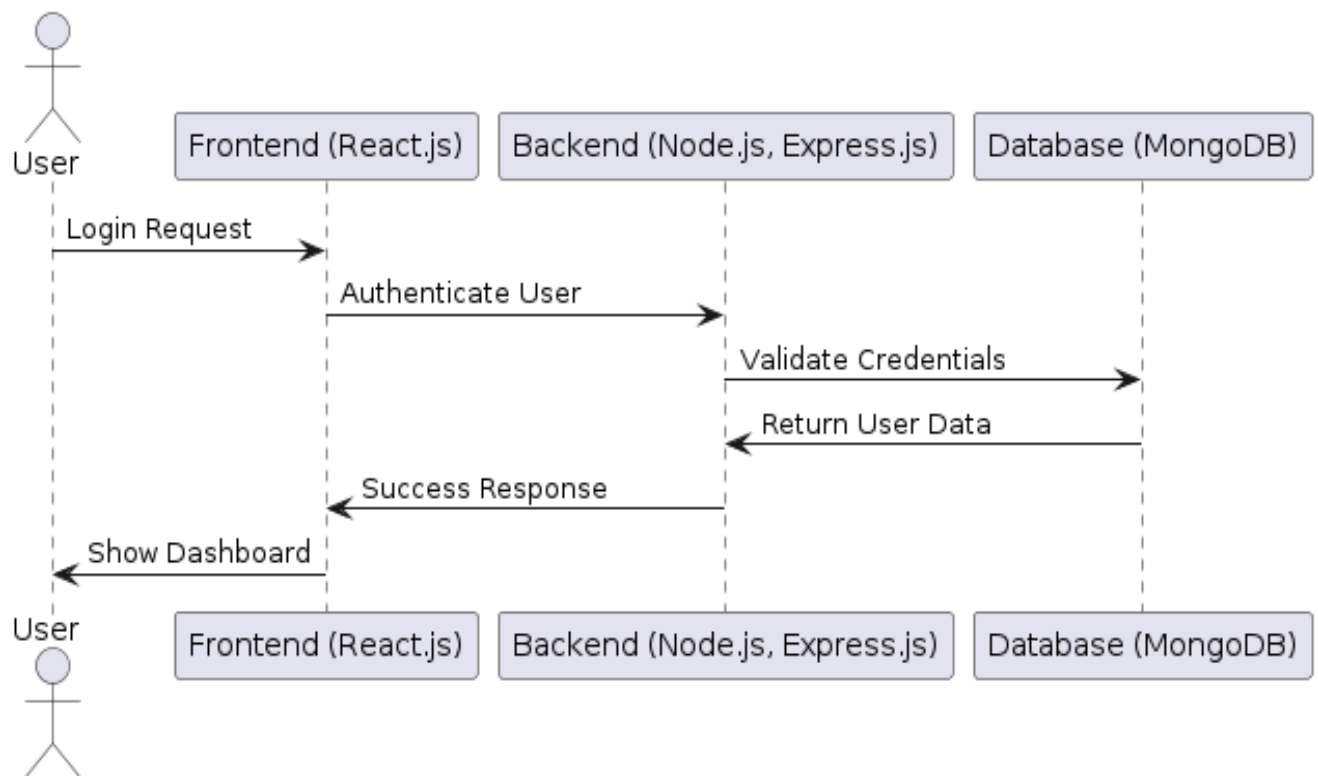
4.3.4. Activity Diagram

An **Activity Diagram** depicts the workflow of a system, showing different activities, decisions, and parallel processes. It is useful for modeling business logic and dynamic behaviors in the system.
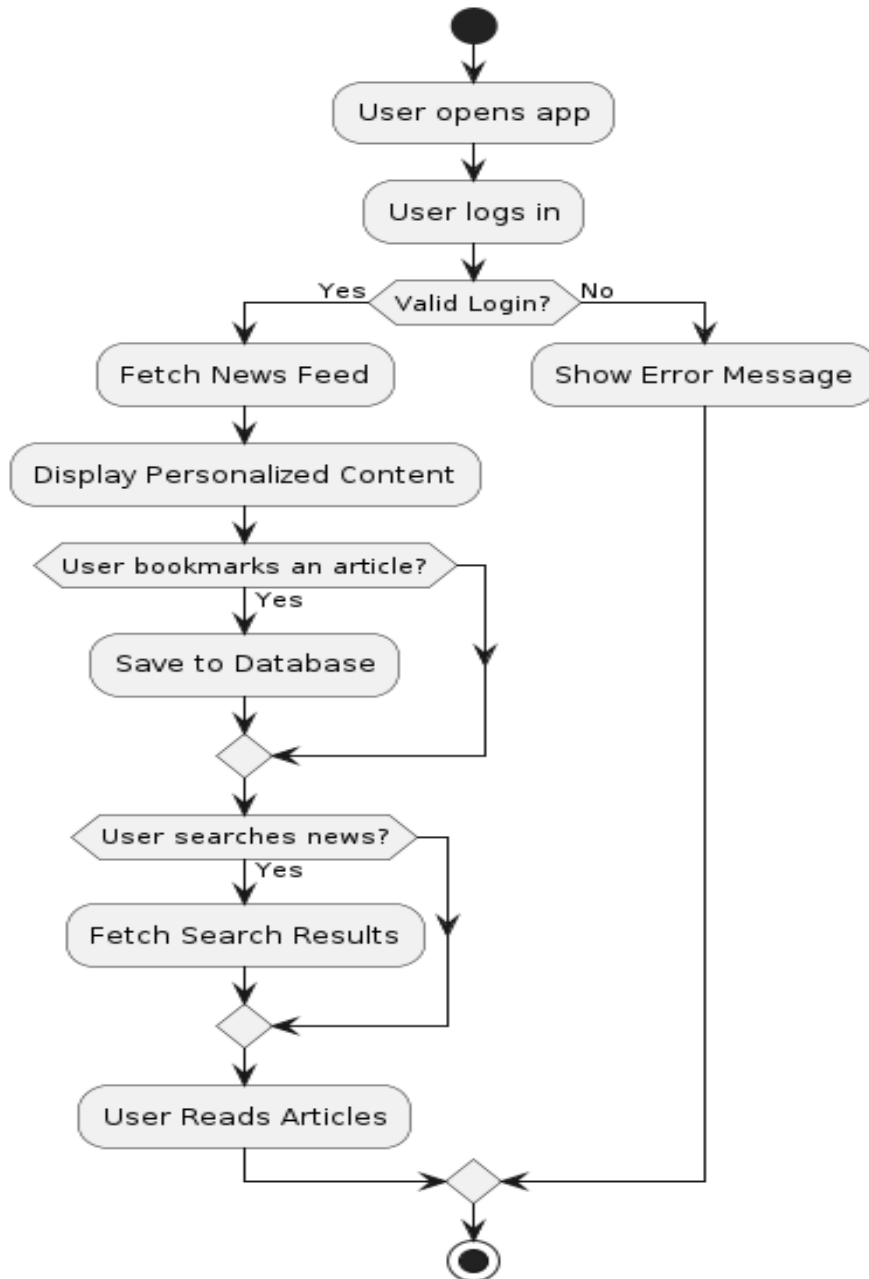
SYSTEM IMPLEMENTATION

CHAPTER 5
SYSTEM IMPLEMENTATION

System implementation refers to the process of integrating and deploying the developed system into a real-world environment. This phase ensures that the system functions as intended and meets user requirements.

1. Implementation Strategy

The system is implemented using a phased approach to minimize risks and ensure smooth deployment. The key strategies include:

- **Pilot Implementation:** Deploying the system to a small group of users for initial testing.
- **Parallel Implementation:** Running the new system alongside the existing system to compare performance.
- **Direct Implementation:** Replacing the old system with the new one immediately.
- **Phased Implementation:** Gradually introducing system components in different stages.

2. Technology Stack

- **Frontend:** React.js for a responsive and dynamic user interface.
- **Backend:** Node.js with Express.js for handling server-side logic.
- **Database:** MongoDB for efficient data storage and retrieval.
- **Authentication:** JWT-based user authentication for secure access.
- **Hosting & Deployment:** Deployed on cloud platforms like AWS or Firebase.

3. Integration and Testing

- **Unit Testing:** Testing individual components to ensure correctness.
- **Integration Testing:** Checking communication between different system modules.

- **User Acceptance Testing (UAT):** Ensuring the system meets user requirements.
- **Performance Testing:** Measuring system speed, scalability, and responsiveness.

4. Security Considerations

- **Data Encryption:** Using SSL/TLS for secure data transmission.
- **User Authentication:** Implementing multi-factor authentication (MFA) for enhanced security.
- **Access Control:** Role-based access control (RBAC) to restrict unauthorized actions.

5. Deployment Process

- **Code Review & Approval:** Ensuring code quality before deployment.
- **CI/CD Pipeline:** Automating deployment using GitHub Actions or Jenkins.
- **Server Setup:** Configuring production servers for hosting.
- **Monitoring & Maintenance:** Using logging tools like LogRocket or Datadog for continuous monitoring.

SYSTEM TESTING

CHAPTER 6

SYSTEM TESTING

System testing is a critical phase in software development that evaluates the entire system's functionality, performance, security, and usability. It ensures that the system meets the specified requirements before deployment.

1. Objectives of System Testing

- To verify that the system functions as expected.
- To identify and fix defects before production.
- To ensure system stability under various conditions.
- To validate system security and data integrity.

Types of System Testing

6.1 Functional Testing

Ensures that all features work as intended by executing test cases based on requirements.

- **Unit Testing:** Tests individual components for correctness.
- **Integration Testing:** Verifies interactions between integrated modules.
- **User Acceptance Testing (UAT):** Ensures the system meets user expectations.

6.2 Non-Functional Testing

Focuses on system performance, security, and usability.

- **Performance Testing:** Measures system speed, responsiveness, and scalability.
- **Load Testing:** Evaluates system behavior under expected and peak loads.
- **Security Testing:** Identifies vulnerabilities to prevent unauthorized access.
- **Usability Testing:** Assesses user-friendliness and overall experience.

1. **Testing Tools and Techniques**

    - **Automation Testing:** Using Selenium, Junit, or Jest for automated test execution.
    - **Manual Testing:** Human testers execute test cases for usability validation.
    - **Penetration Testing:** Ethical hacking methods to find security loopholes.
    - **Regression Testing:** Ensuring new updates do not introduce new bugs.

2. Bug Tracking and Reporting

Bugs and defects are tracked using tools like Jira, Bugzilla, or Trello. Each bug is classified based on severity (Critical, Major, Minor) and resolved accordingly.

DEVELOPMENT PROCESS

CHAPTER 7

DEVELOPMENT PROCESS

The development process outlines the structured approach used to design, build, and implement software solutions. It ensures efficiency, quality, and maintainability throughout the software lifecycle.

7.1 Software Development Life Cycle (SDLC) Phases

1. Requirement Analysis

- Understanding user needs and system requirements.
- Documenting functional and non-functional requirements.

2. Planning

- Defining project scope, timeline, and resources.
- Creating a roadmap for development.

3. Design

- Architectural and UI/UX design of the system.
- Creating wireframes, database schema, and system models.

4. Development

- Writing code based on design specifications.
- Implementing features, backend logic, and database integration.

5. Testing

- Conducting unit, integration, and system testing.
- Fixing bugs and ensuring system stability.

6. Deployment

- Releasing the application to a production environment.
- Configuring hosting and database servers.

7. Maintenance and Updates

- Monitoring system performance and security.
- Rolling out new features and fixes as needed.

8. Development Methodologies

- **Waterfall Model:** Sequential and structured development approach.
- **Agile Methodology:** Iterative and flexible approach for continuous improvements.
- **DevOps:** Integration of development and operations for faster releases.

7.2 Tools and Technologies

- **Version Control:** Git, GitHub, GitLab
- **Development Frameworks:** React.js, Node.js, Express.js
- **Database:** MongoDB, MySQL
- **Testing Tools:** Jest, Selenium
- **CI/CD:** Jenkins, GitHub Actions

FEATURES OVERVIEW

CHAPTER 8

FEATURES OVERVIEW

The features of the system define its functionality, usability, and efficiency. These features are designed to enhance user experience and provide seamless navigation within the platform.

Key Features

1. User Authentication and Authorization

- Secure login and registration.
- Role-based access control (Admin/User).
- Multi-factor authentication for added security.

2. Personalized News Feed

- AI-driven news recommendations based on user interests.
- Real-time updates with categorized news.
- Bookmarking and saving articles for later.

3. Advanced Search and Filtering

- Keyword-based search functionality.
- Filters for categories, sources, and date range.
- Sorting options based on relevance and popularity.

4. Content Management

- Admin dashboard for managing news sources.
- Ability to publish, edit, and remove news articles.
- Fake news detection and verification system.

5. Interactive User Experience

- User comments, likes, and sharing options.

- Dark mode and customizable themes.
- Multi-language support for global accessibility.

6. AI-Powered Insights and Analytics

- Sentiment analysis of news articles.
- Trending topics and news highlights.
- Graphical reports for user engagement.

7. Notifications and Alerts

- Personalized push notifications for breaking news.
- Daily and weekly news digest emails.
- Custom alert settings for preferred topics.

CONCLUSION

CONCLUSION

The development of InsightStream: Navigate the News Landscape has successfully created an intelligent, user-friendly, and secure news platform that enhances digital news consumption. By integrating modern web technologies such as React.js, Node.js, MongoDB, and a structured content management system, the platform ensures seamless performance, scalability, and an engaging user experience. Security features like multi-factor authentication, role-based access control, and fake news detection strengthen credibility and user trust. With real-time updates, interactive user engagement through comments and sharing, and an intuitive admin dashboard for content management, InsightStream provides a comprehensive and efficient way for users to stay informed.

The platform's impact includes personalized content organization, advanced search and filtering capabilities, real-time notifications, and multi-device accessibility, making it a modern and innovative solution for news consumption. Future enhancements could focus on improving content categorization, expanding language support, refining the user interface, and strengthening verification mechanisms to further improve the accuracy and reliability of news delivery. Additionally, integrating offline reading capabilities and improved performance optimizations would make the platform more versatile and user-friendly. By continuously evolving and adapting to user needs, InsightStream can remain a pioneering force in digital journalism, providing a seamless, reliable, and engaging news experience for users worldwide.

APPENDIX

SCREEN LAYOUTS:

Your New Favorite Sex Toy Might Be a Drugstore 'Egg'

Deadpool Co-Creator Rob Liefeld Says He's Done With Marvel

Samsung's Most Popular Foldable Phone Is About to Get More Affordable

Magic: The Gathering Is Adding Avatar to Its Crossover Collection

JBL PartyBox 310 Is Back to Its Holiday Price and One of Amazon's Most Popular Portable Speakers

National Science Foundation Fires 168 Workers as Federal Purge Continues

Firings at the NSF included permanent employees who had already completed their one-year probationary period, as well as at-will workers.

KIM ZETTER

DOGE's Website Is Just One Big X Ad

The source code for the new Department of Government Efficiency's "official US government website" points to X as its primary source of authority, while sharing links to the site sends users to x.com.

DAVID GILBERT

DOGE's Misplaced War on Software Licenses

DOGE claims that a government agency has nearly three times as many software licenses as employees. Experts say there are plenty of good reasons for that.

PARESH DAVE

Here's a Map of the For-Sale Government Properties the GSA Pulled From Its Website

The General Services Administration plans to sell hundreds of government buildings, including FBI headquarters. Use our interactive tools to see where they are—and the congressional districts they're in.

DHRUV MEHROTRA

Dozens of CFPB Workers Fired in After-Hours Blitz

Some affected employees of the Consumer Financial Protection Bureau were notified with an email that addressed them as [EmployeeFirstName][EmployeeLastName], [Job Title], [Division].

DOGE Put Him in the Treasury Department. His Company Has Federal Contracts Worth Millions

Experts say the conflicts posed by Tom Krause's dual roles are unprecedented in the modern era.

PARESH DAVE

Elon Musk's Man in the Treasury Is Still Holding Down His Day Job as Software CEO

An email sent to workers at the US Treasury Department's Bureau of the Fiscal Service said DOGE's Tom Krause is "currently" the CEO of the Cloud Software Group. Critics believe this could be a massive conflict of interest.
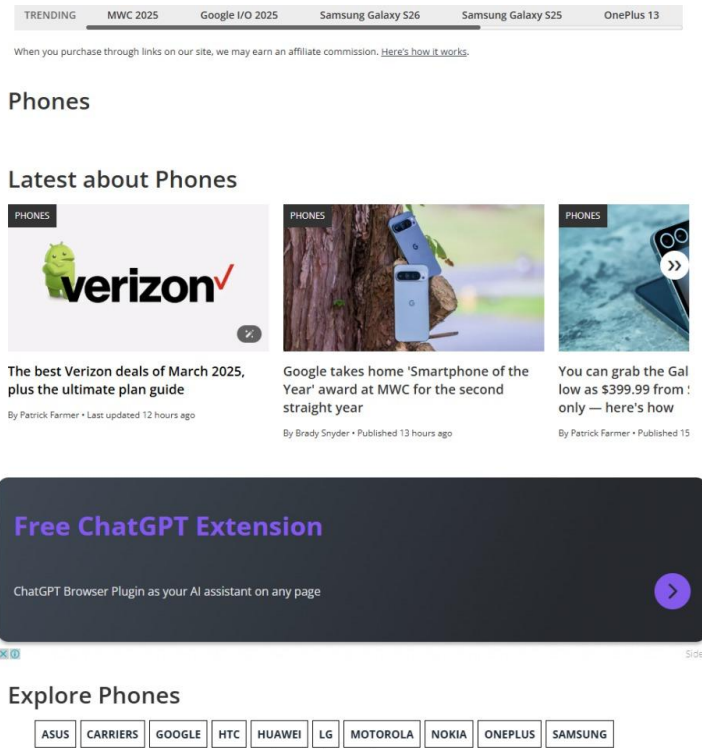
Boston Dynamics Led a Robot Revolution. Now Its Machines Are Teaching Themselves New Tricks

Boston Dynamics founder Marc Raibert says reinforcement learning is helping his creations gain more independence.

WILL KNIGHT

Phones

Latest about Phones

The best Verizon deals of March 2025, plus the ultimate plan guide

By Patrick Farmer • Last updated 12 hours ago

Google takes home 'Smartphone of the Year' award at MWC for the second straight year

By Brady Snyder • Published 13 hours ago

You can grab the Gal low as $399.99 from only — here's how

By Patrick Farmer • Published 15

Explore Phones

ASUS | CARRIERS | GOOGLE | HTC | HUAWEI | LG | MOTOROLA | NOKIA | ONEPLUS | SAMSUNG

SOURCE CODE:

```
import React, { useContext } from 'react'

import '../styles/HomeArticles.css'

import { GeneralContext } from '../context/GeneralContext'

import { useNavigate } from 'react-router-dom';

import { Spinner } from 'react-bootstrap';


const HomeArticles = () => {


  const navigate = useNavigate();


  const {businessNews, technologyNews, politicsNews} =
useContext(GeneralContext)
```

```jsx
    return (

      <div className='home-articles-container'>

        <div className='home-articles-body'>

            <div className="home-articles-head">
              <h2>Business</h2>
              <p onClick={()=> navigate('/category/business')} >View all</p>
            </div>

            {businessNews.length > 0 ?

              <div className="home-articles">

                  {businessNews.map((news, index)=>{
                    return index < 3 && (

                        <div className="home-article" onClick={()=>
window.open(news.url, '_blank')}>
                              <img src={news.urlToImage} alt="placeholder" />
                              <p>{news.title}</p>
                          </div>
                        )
                    })}
```

```jsx
        </div>
      :

    <div className="spinners">
      <Spinner animation="grow" size="sm" />
      <Spinner animation="grow" size="sm" />
      <Spinner animation="grow" size="sm" />
   </div>
   }
</div>


<div className='home-articles-body'>

    <div className="home-articles-head">
     <h2>Technology</h2>
     <p onClick={()=> navigate('/category/technology')}>View all</p>
    </div>
    {technologyNews.length > 0 ?

    <div className="home-articles">

      {technologyNews.map((news, index)=>{
       return index < 3 && (
```

```jsx
          <div className="home-article" onClick={()=>
window.open(news.url, '_blank')}>
              <img src={news.urlToImage} alt="placeholder" />
              <p>{news.title}</p>
            </div>
          )
        })}


      </div>
    :


    <div className="spinners">
      <Spinner animation="grow" size="sm" />
      <Spinner animation="grow" size="sm" />
      <Spinner animation="grow" size="sm" />
    </div>
    }
  </div>



  <div className='home-articles-body'>


      <div className="home-articles-head">
       <h2>Politics</h2>
       <p onClick={()=> navigate('/category/politics')}>View all</p>
```

```jsx
            </div>
        {politicsNews.length > 0 ?


        <div className="home-articles">


            {politicsNews.map((news, index)=>{
              return index < 3 && (


                <div className="home-article" onClick={()=>
window.open(news.url, '_blank')}>
                    <img src={news.urlToImage} alt="placeholder" />
                    <p>{news.title}</p>
                  </div>
                )
              })}


          </div>
        :
        <div className="spinners">
          <Spinner animation="grow" size="sm" />
          <Spinner animation="grow" size="sm" />
          <Spinner animation="grow" size="sm" />
        </div>
        }
    </div>
```

```
    </div>
  )
}
```