**News Caster**

A Minor Project Report

Submitted in partial fulfillment of requirement of the

Degree of

**BACHELOR OF TECHNOLOGY in COMPUTER SCIENCE & ENGINEERING**

BY

**Nikhil Mukati-EN22CS301649**

**Nihal Dubey-EN22CS301643**

**Nilay Jain - EN22CS301651**

Under the Guidance of

**Prof.AMIT GOUD**

**Prof.ARJUN DIXIT**



**Department of Computer Science & Engineering**

**Faculty of Engineering**

**MEDI-CAPS UNIVERSITY, INDORE- 453331**

**JAN-JUNE-2025**

**Report Approval**

The project work**”News Caster”**is hereby approved as a creditable study of an engineering/computer application subject carried out and presented in a manner satisfactory to warrant its acceptance as prerequisite for the Degree for which it has been submitted.

It is to be understood that by this approval the undersigned do not endorse or approve any statement made, opinion expressed, or conclusion drawn there in; but approve the “Project Report” only for the purpose for which it has been submitted.

Internal Examiner

Name:

Designation

Affiliation

External Examiner

Name:

Designation

Affiliation

**Declaration**

I/We hereby declare that the project entitled **“News Caster**” submittedin partial fulfillment for the award of the degree of Bachelor of Technology/Master of Computer Applications in ‘Computer Science Engineering’ completed under the supervision of **Prof.Amit Goud & prof.Ajrun Dixit** Faculty of Engineering, Medi-Caps University Indore is an authentic work.

Further, I/we declare that the content of this Project work, in full or in parts, have neither been taken from any other source nor have been submitted to any other Institute or University for the award of any degree or diploma.

**Nikhil Mukati**

**Nilay Jain**

**Nihal Dubey**

**Certificate**

I/We, **Pro.Amit Goud & Arjun Dixit** certify that the project entitled **“News Caster”** submittedin partial fulfillment for the award of the degree of Bachelor of Technology/Master of Computer Applications by Nikhil Mukati , Nilay Jain & Nihal Dubeyistherecordcarried out by her/them under my/our guidance and that the work has not formed the basis of award of any other degree elsewhere.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**Prof. Arjun Dixit & Amit Goud Dr. Ratnesh Litoriya**

Head of the Department

Computer Science & Engineering Computer Science & Engineering

Medi-Caps University, Indore Medi-Caps University, Indore

**Acknowledgements**

I would like to express my deepest gratitude to the Honorable Chancellor, **Shri R C Mittal,** who has provided me with every facility to successfully carry out this project, and my profound indebtedness to **Prof. (Dr.) D. K. Patnaik,** Vice Chancellor, Medi-Caps University, whose unfailing support and enthusiasm has always boosted up my morale. I also thank **Prof. (Dr.) Pramod S. Nair,** Dean, Faculty of Engineering, Medi-Caps University, for giving me a chance to work on this project. I would also like to thank my Head of the Department **Dr. Ratnesh Litoriya** for his continuous encouragement for the betterment of the project.

It is their help and support, due to which we became able to complete the design and technical report.

Without their support this report would not have been possible.

**Nikhil Mukati , Nilay Jain & Nihal Dubey**

B.Tech. III Year

Department of Computer Science & Engineering

Faculty of Engineering

Medi-Caps University, Indore

**Abstract**

The API provides a concise summary of news articles, allowing users to quickly grasp the content. It includes essential metadata such as image representation, category, and provider information. The News Search API Integration Mini Project aims to develop a web application that utilizes a news search API to retrieve and display relevant news articles based on user queries. The project involves integrating a chosen News API into a web application using modern web development technologies. The application allows users to input keywords or phrases of interest, fetches data from the API in real-time, and presents the results in a user-friendly format.

Keyword or phrase. Eg: find all articles containing the word 'Microsoft'.

Date published. Eg: find all articles published yesterday.

Source domain name. Eg: find all articles published on thenextweb.com.

Language. Eg: find all articles written in English.

**Keywords:**

While not explicitly returned, the API uses search queries to identify relevant keywords and match them with news articles.

1. News API
2. Web development
3. API integration
4. Frontend development
5. Java Script

**Table of Contents**

|  |  | **Page No.** |
| --- | --- | --- |
|  |  |  |
|  | Report Approval | ii |
|  | Declaration | iii |
|  | Certificate | iv |
|  | Acknowledgement | v |
|  | Abstract | vi |
|  | Table of Contents | vii |
|  | List of figures | viii |
|  | List of tables | ix |
|  | Abbreviations | x |
|  | Notations & Symbols | xi |
| Chapter 1 | Introduction (Whichever is applicable) |  |
|  | 1.1 Introduction |  |
|  | 1.2 Literature Review |  |
|  | 1.3 Objectives |  |
|  | 1.4 Significance |  |
|  | 1.5 Research Design |  |
|  | 1.6 Source of Data |  |
|  | 1.7 Chapter Scheme |  |
| Chapter 2 | REQUIREMENTS SPECIFICATION |  |
|  | 2.1 User Characteristics |  |
|  | 2.2 Functional Requirements |  |
|  | 2.3 Dependencies |  |
|  | 2.4 Performance Requirements |  |
|  | 2.5 Hardware Requirements |  |
|  | 2.6 Constraints & Assumptions |  |
| Chapter 3 | DESIGN ( Whichever is applicable) |  |
|  | 3.1 Algorithm (if Applicable) |  |
|  | 3.2 Function Oriented Design for procedural approach |  |
|  | 3.3 **System Design (**Whichever is applicable) |  |
|  | 3.3.1 Data Flow Diagrams (Level 0,Level1) |  |
|  | 3.3.2 Activity Diagram |  |
|  | 3.3.3 Flow Chart |  |
|  | 3.3.4 Class Diagram |  |
|  | 3.3.5 ER Diagram |  |
|  | 3.3.6 Sequence diagram |  |
|  | 3.4 **Database Design** |  |
|  | 3.4.1 Logical Database Design |  |
|  | 3.4.2 Physical Database Design |  |
| Chapter 4 | Implementation, Testing, and Maintenance |  |
|  | 4.1 Introduction to Languages, IDE’s, Tools and Technologies used for Implementation |  |
|  | 4.2 Testing Techniques and Test Plans (According to project) |  |
|  | 4.3 Installation Instructions |  |
|  | 4.4 End User Instructions |  |
|  |  |  |
| Chapter 5 | Results and Discussions |  |
|  | 5.1 User Interface Representation (of Respective Project) |  |
|  | 5.2 Brief Description of Various Modules of the system |  |
|  | 5.3 Snapshots of system with brief detail of each |  |
|  | 5.4 Back Ends Representation (Database to be used ) |  |
|  | 5.5 Snapshots of Database Tables with brief description |  |
| Chapter 6 | Summary and Conclusions |  |
| Chapter 7 | Future scope |  |
|  | Appendix |  |
|  | Bibliography |  |
|  | List of Publications (If any) |  |
|  | Reprints of publications(If any) |  |
| **Note** | **Results and Discussions may not be a separate chapter it may be included in main chapter** |  |

**Chapter-1**

**Introduction**

**1.1 Introduction**

The API streamlines news article retrieval, bridging the gap between users and up-to-date news content. Whether you’re building a news aggregator or enhancing your application, the Bing News Search API simplifies the process.

In today's fast-paced world, staying updated with the latest news and information is crucial. With the abundance of news sources available online, accessing relevant and timely news articles can sometimes be overwhelming. To address this challenge, the News Search API Mini Project offers a solution by harnessing the power of technology to streamline the news discovery process.

This project aims to develop a web application that leverages a News Search API to provide users with a convenient way to search for and access news articles on topics of interest. By integrating a News API into the application, users can easily retrieve real-time news content tailored to their specific preferences and queries.

The News Search API Mini Project not only serves as a practical tool for accessing news but also offers a valuable opportunity for developers to gain hands-on experience in web development, API integration, and user interface design. Through the implementation of modern web technologies and best practices, the project endeavors to deliver a seamless and intuitive user experience, ensuring that users can effortlessly discover and explore the latest news articles across various categories and sources.

**1.2 Key Features**

1. Sorting and Filtering: The API v7 introduces new sorting and filtering options, enabling precise results within trending news topics.
2. Category-Based Retrieval: Request top news articles by category (e.g., sports, entertainment) using the appropriate URL1.
3. Search-Based Retrieval: Retrieve news articles based on user-specific search queries. If the query is empty, the API returns the top news articles1.
4. Trending Topics: Explore trending news topics currently popular on social networks.
5. User Input Interface: The application provides a user-friendly interface where users can input their search queries.
6. API Integration: Integration with a News API allows the application to fetch up-to-date news articles relevant to the user's query.
7. Search Results Display: The retrieved news articles are displayed in a visually appealing format, including titles, summaries, publication dates, and sources.
8. Pagination and Filtering: Implementation of pagination and filtering options to manage large sets of search results effectively.
9. Responsive Design: The application is designed to be responsive, ensuring optimal user experience across various devices and screen sizes.
10. Error Handling: Proper error handling mechanisms are implemented to handle cases such as API request failures or invalid user inputs.
11. Security: Implementation of security measures to protect user data and prevent vulnerabilities, such as secure API key handling and data encryption.
12. Overall, the News Search API Integration Mini Project provides an opportunity to develop practical skills in web development, API integration, user interface design, and data presentation while creating a useful tool for accessing real-time news information.

**1.3 Literature Review**

1. API Integration in Web Development:

(Fielding, 2000; Davidson et al., 2017).

API integration has become a fundamental aspect of modern web development, enabling developers to incorporate external services and data sources seamlessly into their applications.Various techniques and best practices have been established for effectively integrating APIs into web applications, including RESTful API design principles and asynchronous programming methodologies .

1. User Interface Design and User Experience (UX):

(Tullis & Albert, 2013).

User interface design plays a critical role in the success of web applications, particularly in ensuring ease of use and intuitive navigation for end-users .(Morville & Rosenfeld, 2006)Attention to UX principles, such as responsive design and accessibility, can significantly enhance the usability and satisfaction of users interacting with web applications

1. Real-Time Data Retrieval and Visualization:

(Ghazali et al., 2019).

Real-time data retrieval capabilities have become increasingly important in web applications, allowing users to access the most current information available .(Few, 2009)Effective data visualization techniques, including interactive charts and graphs, can aid in presenting complex information in a digestible and engaging manner

1. Security Considerations in Web Development:

(Gupta & Sridhar, 2018).

Security is a paramount concern in web development, particularly when integrating external APIs and handling user data.

(Bishop, 2003).

Best practices for ensuring the security of web applications include implementing secure authentication mechanisms, data encryption protocols, and regular security audits

1. Error Handling and Fault Tolerance:

(Havelund & Rosu, 2001).

Robust error handling mechanisms are essential for handling unexpected issues, such as API failures or network errors, to maintain the reliability and usability of web applications.

(Randall et al., 2002).

Strategies for implementing fault tolerance, such as retry mechanisms and graceful degradation, can help mitigate the impact of errors on the user experience

1. Case Studies and Practical Examples:

(Karyotis et al., 2019; Gogoi et al., 2020).

Previous studies and practical examples of web applications integrating news APIs can provide valuable insights and inspiration for the design and implementation of the News Search API Mini Project

**1.4 Problem Statement**

This Project basically solves the problem regarding to observe the news by going to different sites at one time,but this project dynamically generate the news of multiple sites at one time only.

**1.5 Objectives**

1. API Integration: Integrate a News Search API into the web application to retrieve real-time news articles based on user queries.
2. User Interface Design: Develop an intuitive and user-friendly interface that allows users to input search queries and navigate search results effectively.
3. Search Functionality: Implement robust search functionality that enables users to discover relevant news articles using keywords, phrases, or categories.
4. Data Presentation: Present search results in a visually appealing and organized format, including article titles, summaries, publication dates, and sources.
5. Pagination and Filtering: Implement pagination and filtering options to manage large sets of search results efficiently and enable users to refine their search criteria.
6. Real-Time Updates: Ensure that the application retrieves and displays the latest news articles from the API in real-time, providing users with up-to-date information.
   1. **Technologies Used**

**Hardware platform**

1. Web Server: A server capable of hosting and serving web applications. Common choices include Apache, Nginx, or Microsoft IIS.
2. Application Server: A server responsible for executing server-side code and handling requests from clients. This may involve technologies such as Node.js, Python
3. Database Server: If the application requires data storage, a separate server running a database management system (DBMS) such as MySQL, PostgreSQL, MongoDB, or SQLite may be utilized.
4. Cloud Platforms:Infrastructure as a Service (IaaS) providers such as Amazon Web Services (AWS), Microsoft Azure, or Google Cloud Platform (GCP) offer virtualized computing resources (e.g., virtual machines, storage) that can be used to host and deploy web applications..
5. Development Hardware:Workstations or laptops equipped with sufficient computing power and memory are used by developers for writing code, testing, and debugging the application.
6. Monitoring and Analytics Tools:Monitoring tools such as New Relic, Datadog, or Prometheus may be utilized to monitor server performance, track resource utilization, and identify potential issues.Analytics tools like Google Analytics or Mixpanel can provide insights into user behavior, traffic patterns, and usage metrics, helping to optimize the application over time.
7. Backup and Disaster Recovery:Backup solutions, such as regular data backups to off-site storage or cloud-based backup services, help ensure data integrity and facilitate disaster recovery in the event of hardware failures or data loss.

**Software platform**

1. Frontend :HTML CSS and JavaScript can be used to build the frontend of the application, providing efficient rendering, component-based architecture, and state management.
2. Integrated Development Environments (IDEs):

IDEs like Visual Studio Code, Atom, Sublime Text, or JetBrains WebStorm provide robust development environments with features like syntax highlighting, code completion, and debugging tools.

1. Version Control Systems:

Version control systems like Git, along with platforms like GitHub, GitLab, or Bitbucket, are essential for managing and collaborating on the project's source code, tracking changes, and facilitating code review.

1. API Integration and Testing:

Postman, Insomnia, or Paw can be used for testing and debugging API requests, ensuring proper integration with the News API and handling responses.

Swagger or OpenAPI Specification can be employed for documenting and standardizing the API endpoints, parameters, and responses.

**CHAPTER-2**

**Requirements Specification**

**2.1 User Charachteristcs**

1. Developers: These users will be primarily interacting with the API programmatically. They might be building applications, websites, or integrating the news data into other systems.
2. Researchers: Users who are interested in analyzing news trends, conducting sentiment analysis, or performing other types of research using news data.
3. Journalists: Journalists might use the API to gather information for articles, monitor breaking news, or conduct investigative research.
4. Media Companies: Companies in the media industry might use the API to power their news websites, apps, or other digital platforms.
5. Data Scientists: Data scientists may use the API to gather data for machine learning models, natural language processing tasks, or other data-driven analyses.
6. Businesses: Businesses might use the API to stay informed about industry news, monitor competitor mentions, or gather insights for market analysis.
7. Students: Students might use the API for educational purposes, such as learning about APIs, practicing programming skills, or conducting research for school projects.
8. Government Agencies: Government agencies might use the API for monitoring news related to policies, public sentiments, or specific events.
9. Non-profit Organizations: Non-profit organizations might use the API for various purposes, such as raising awareness about specific issues, monitoring news related to their cause, or conducting research for advocacy efforts.
10. General Public: Casual users who are interested in accessing news data for personal consumption, staying informed about current events, or exploring topics of interest.

**2.2 Functional Requirenments**

1. Search: Allow users to search for news articles based on keywords, phrases, categories, or specific criteria.
2. Filtering and Sorting: Enable users to filter and sort search results based on parameters such as date, relevance, source, author, language, location, and sentiment.
3. Article Retrieval: Provide endpoints for retrieving full-text articles, including metadata such as title, publication date, source, author, and article content.
4. Pagination: Support pagination of search results to handle large datasets efficiently, allowing users to navigate through multiple pages of results.
5. Content Summarization: Offer the option to retrieve summarized versions of articles to provide users with quick overviews or previews of content.
6. Multimedia Support: Allow users to access multimedia content associated with news articles, such as images, videos, or audio clips.
7. Real-time Updates: Provide real-time updates or streaming capabilities for breaking news and live events.
8. Personalization: Enable users to customize their news feed based on preferences, interests, or previous interactions.
9. Historical Data Access: Allow users to access historical news data for trend analysis, research, or archival purposes.
10. Authentication and Authorization: Implement authentication and authorization mechanisms to ensure secure access to the API, including user authentication for accessing premium features or restricted content.

**2.4 Hardware Requirenment**

1. CPU:

A multi-core processor (e.g., Intel Core i5 or above) to handle data preprocessing, model training, and inference tasks efficiently.

1. Memory (RAM):

At least 8 GB of RAM to store and manipulate large datasets during data preprocessing and model training.

1. Storage:

Sufficient storage space (e.g., SSD or HDD) to store historical stock price data, machine learning models, and other system components.

1. Graphics Processing Unit (GPU) (Optional):

A dedicated GPU (e.g., NVIDIA GeForce GTX or RTX series) may accelerate model training and inference tasks, especially for deep learning-based approaches.

1. Network Connectivity:

Stable internet connection for accessing external data sources, APIs, and trading platforms.

1. Operating System:

Compatibility with popular operating systems such as Windows, macOS, or Linux distributions like Ubuntu or CentOS.

1. Deployment Environment:

Consideration of cloud-based platforms (e.g., AWS, Google Cloud Platform, Microsoft Azure) for scalability, reliability, and ease of deployment.

1. Monitoring and Maintenance:

Tools for monitoring system performance, resource utilization, and security vulnerabilities to ensure smooth operation and timely maintenance.

**2.4 Constraints & Assumption**

Constraints and assumptions guide the development and use of a news API project. Constraints may include limited access to premium news sources, API rate limits, and data reliability issues. Assumptions encompass factors like consistent data quality, user familiarity with the interface, and reliable internet connectivity. These assumptions may impact system reliability, user experience, and data accuracy. Adapting to constraints requires robust error handling and alternative data sources, while addressing assumptions demands clear communication and user support. Striking a balance between these constraints and assumptions ensures a functional and user-friendly news API project.

**CHAPTER-3**

**DESIGN**

**3.1 System Design**

**3.1.1 Data Flow Diagram**

Data flow diagram is graphical representation of flow of data in an information system. It can depict incoming data flow, outgoing data flow and stored data. The DFD does not mention anything about how data flows through the system.

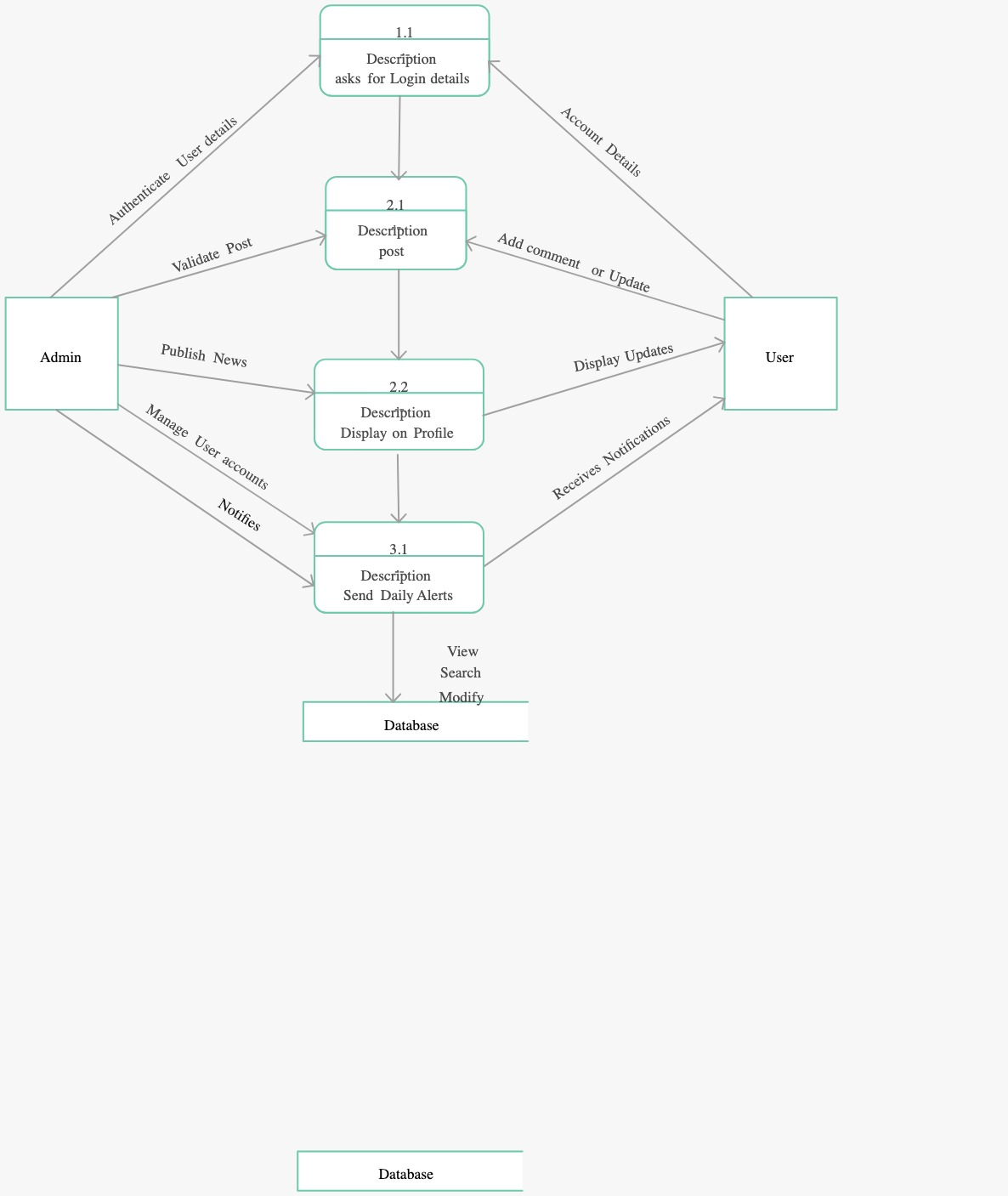
There is a prominent difference between DFD and Flowchart. The flowchart depicts flow of control in program modules. DFDs depict flow of data in the system at various levels. DFD does not contain any control or branch elements.

DFD Level 0

Level 0 - Highest abstraction level DFD is known as Level O DFD, which depicts the entire information system as one diagram concealing all the underlying details. Level 0 DFDs are also known as context level DFDs.

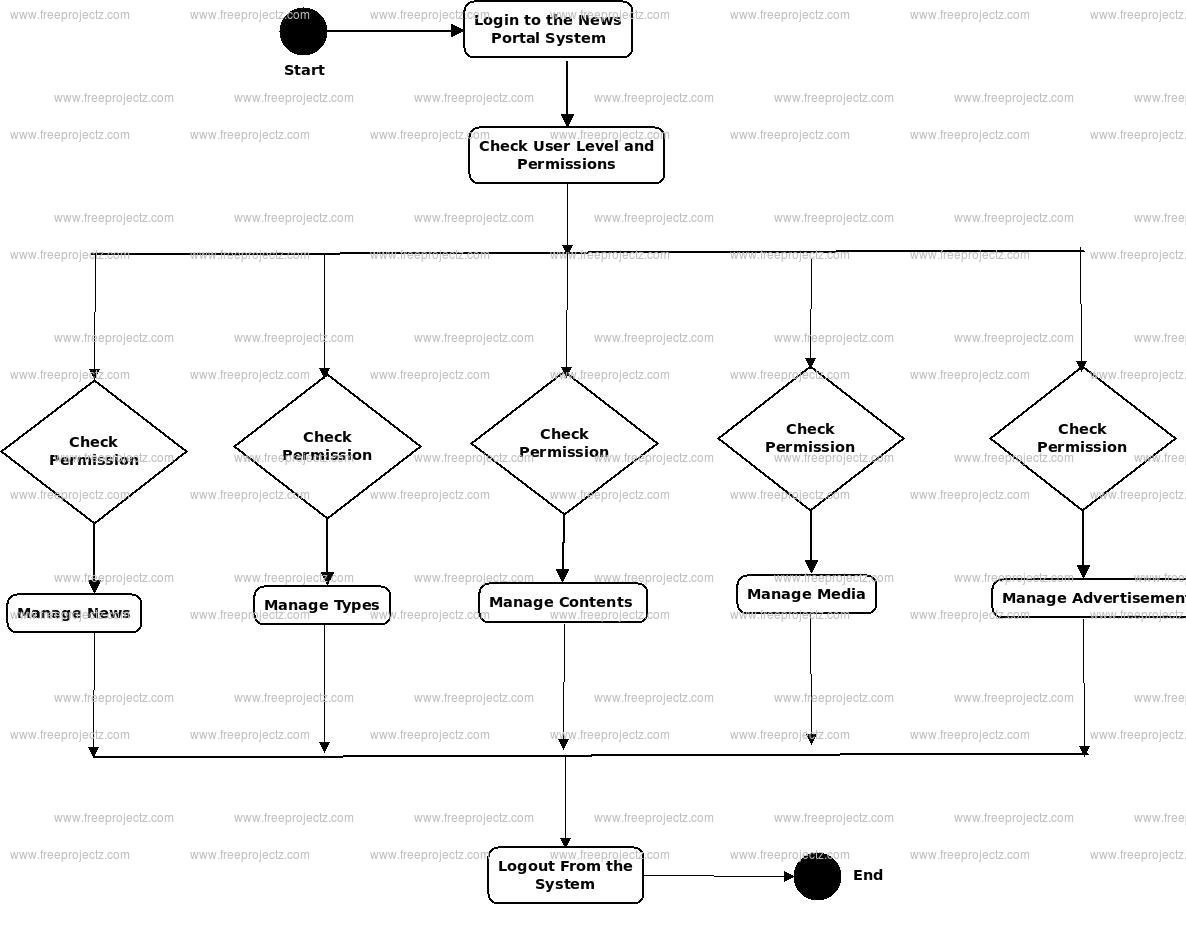
DFD Level 1

The Level 0 DFD is broken down into more specific, Level 1 DFD. Level 1 DFD depicts basic modules in the system and flow of data among various modules. Level 1 DFD also mentions basic processes and sources of information.

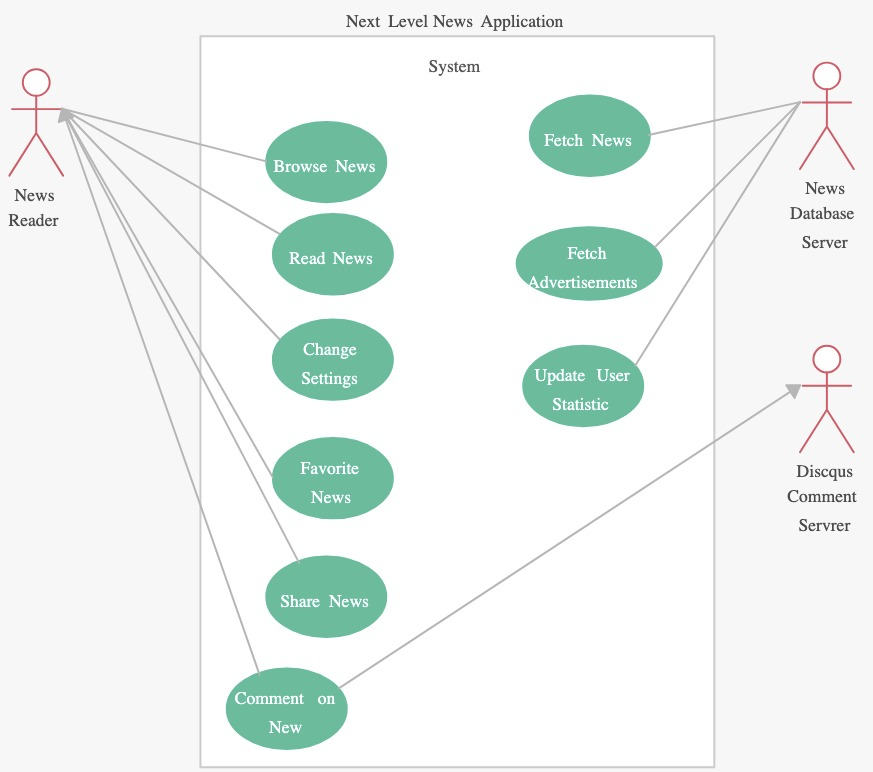


**3.1.2 Activity Diagram**

The basic purposes of activity diagram is similar to other four diagrams. It captures the dynamic behavior of the system. Other four diagrams are used to show the message flow from one object to another, but activity diagram is used to show message flow from one activity to

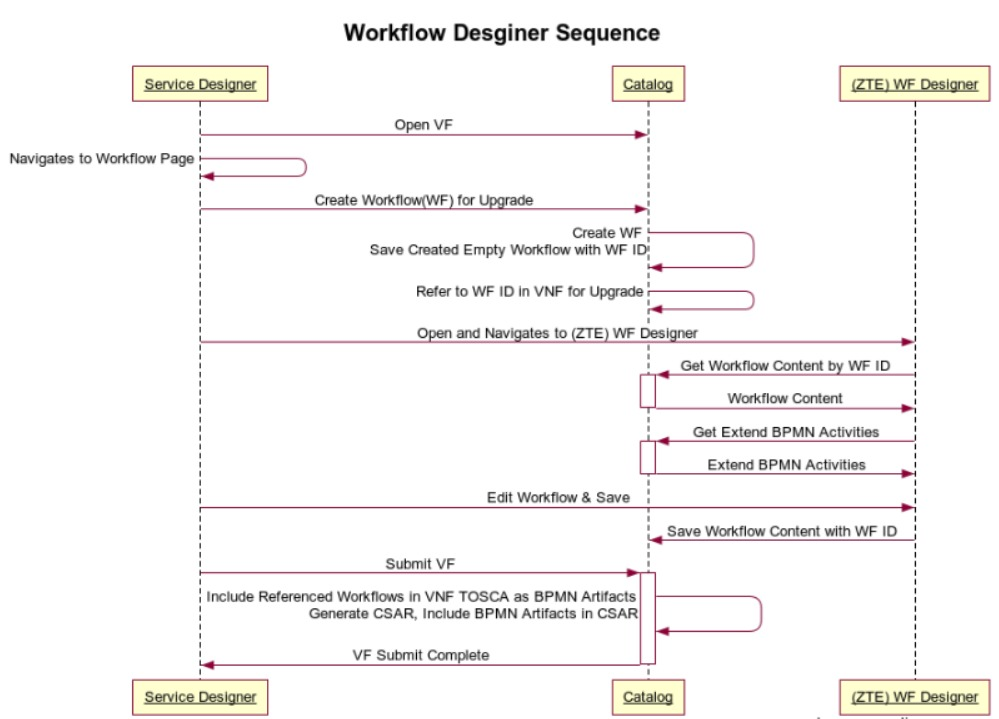


**3.1.3 Use Case Diagram**

The purpose of use case diagram is to capture the dynamic aspect of a system. However, this definition is too generic to describe the purpose, as other four diagrams (activity, sequence, collaboration, and State chart) also have the same purpose. We will investigate some specific purpose, which will distinguish it from other four diagrams.****

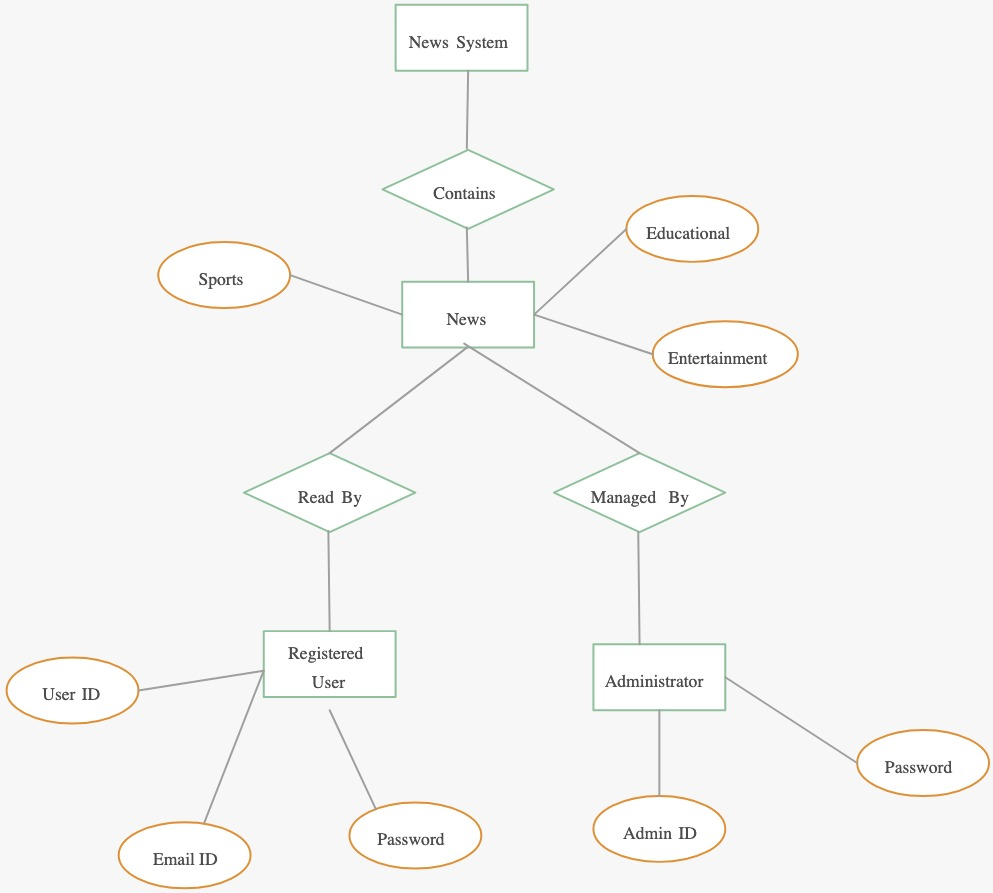
**3.1.4 Sequence Diagram**

The purpose of interaction diagrams is to visualize the interactive behavior of the system. Visualizing the interaction is a difficult task. Hence, the solution is to use different types of models to capture the different aspects of the interaction.

****

**3.1.5 ER Diagram**

An entity relationship (ER) diagram, also known as an ER model, is a flowchart that illustrates how entities relate to each other in a system. ER diagrams are used in a variety of fields, including software engineering, education, research, and business information systems

****

**Chapter-4**

**IMPLEMENTATION,TESTING,MAINTENANCE**

**4.1 Technologies Used**

Text Editor or Integrated Development Environment (IDE):

Visual Studio Code, Sublime Text, Atom, or JetBrains WebStorm: These tools provide features such as syntax highlighting, code completion, and debugging capabilities, making coding more efficient.

Version Control System (VCS):

Git: Essential for managing and tracking changes to your codebase. Platforms like GitHub, GitLab, or Bitbucket can be used for hosting repositories and collaboration.

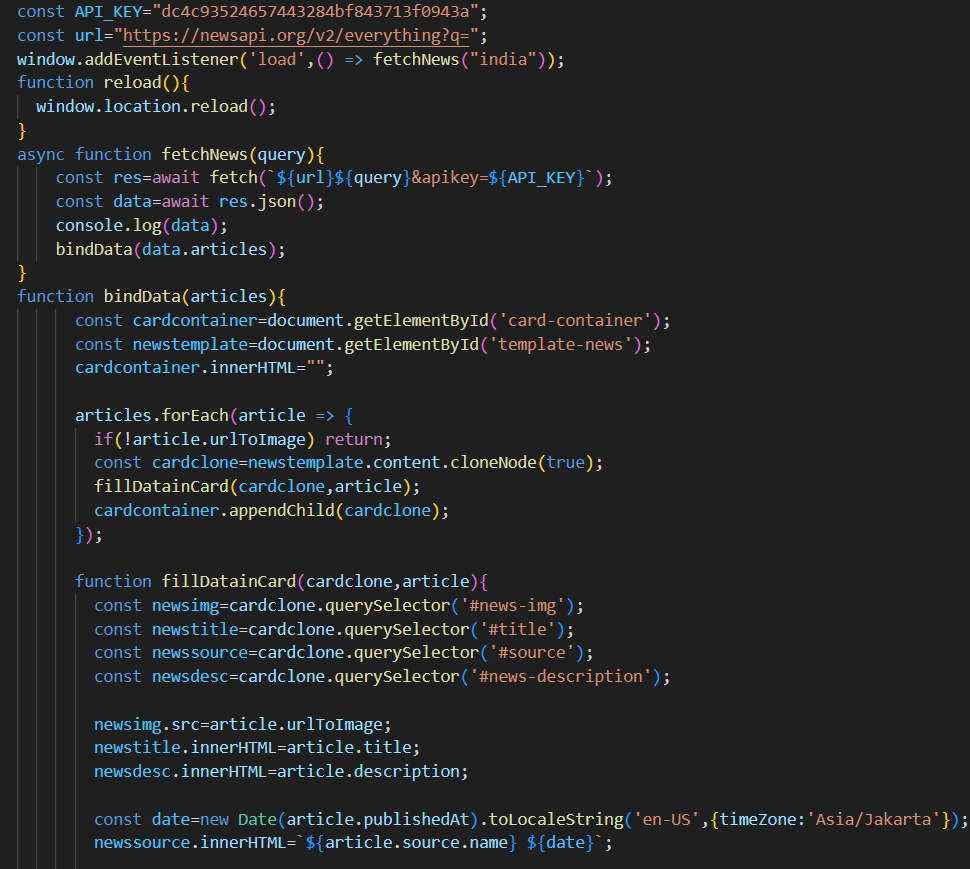
**4.2 Testing Techniques**

1. Unit Testing:
   1. Test individual components of the system, such as data preprocessing functions and machine learning algorithms, in isolation.Use mocking or stubbing to simulate external dependencies and ensure test independence.
2. Integration Testing:
   1. Test the integration between system components, including data pipelines, APIs, and user interfaces.Verify that different modules interact correctly and produce the expected outcomes.
3. Regression Testing:
   1. Re-run previously executed tests to ensure that new changes or updates have not introduced regressions or unintended side effects.Automate regression tests to facilitate continuous integration and deployment pipelines.
4. Performance Testing:
   1. Assess the system's performance under various load conditions, including stress testing, load testing, and scalability testing.Measure response times, throughput, and resource utilization to identify bottlenecks and optimize performance.
5. Data Quality Testing:
   1. Validate the quality and integrity of input data, including historical stock price data and feature engineering outputs.Check for missing values, outliers, and inconsistencies that may affect model training and prediction accuracy.
6. Model Evaluation:
   1. Evaluate the accuracy and generalization performance of machine learning models using appropriate evaluation metrics (e.g., MAE, MSE, RMSE).Perform cross-validation to validate model robustness and assess performance on unseen data.
7. User Acceptance Testing (UAT):
   1. Involve end-users in testing the system's functionality, usability, and user experience.Gather feedback from users to identify usability issues, bugs, or areas for improvement.
8. Security Testing:
   1. Identify and mitigate security vulnerabilities and threats, including unauthorized access, data breaches, and injection attacks.Conduct penetration testing and code review to ensure the system's security posture.

**4.3 End User Instruction**

The news API project offers users a convenient platform to access, search, and engage with a wide range of news articles. Users can effortlessly browse top news, filter and sort articles based on preferences, and save favorites for future reference. With customizable notification settings and easy sharing options, staying informed and connected has never been simpler. The application prioritizes user experience, providing intuitive navigation, robust account management features, and responsive customer support. Built with privacy and security in mind, users can trust their data is handled responsibly. Overall, the news API project empowers users to explore, discover, and stay up-to-date with the latest news effortlessly.

**4.4 Implementation**

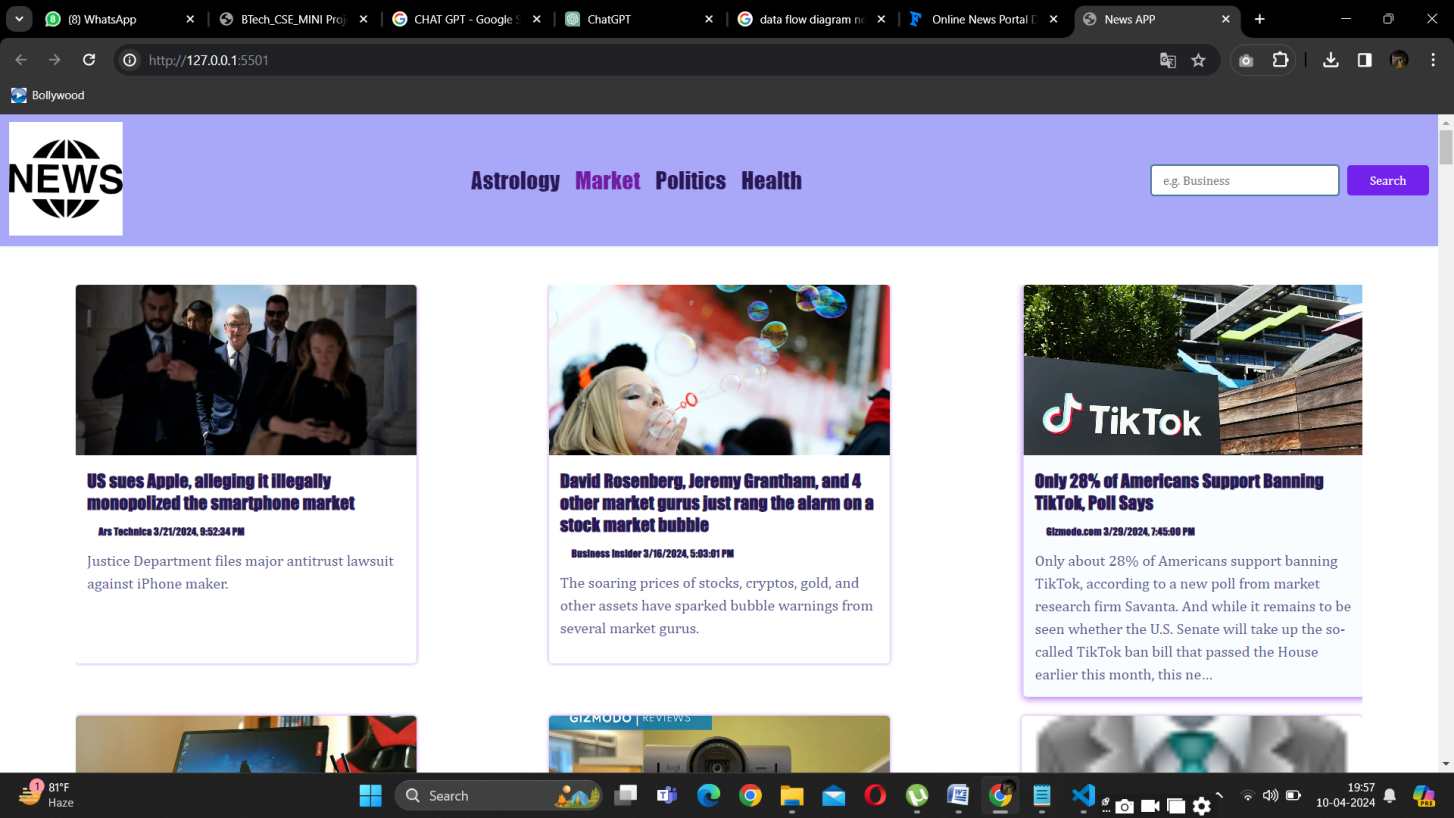
****

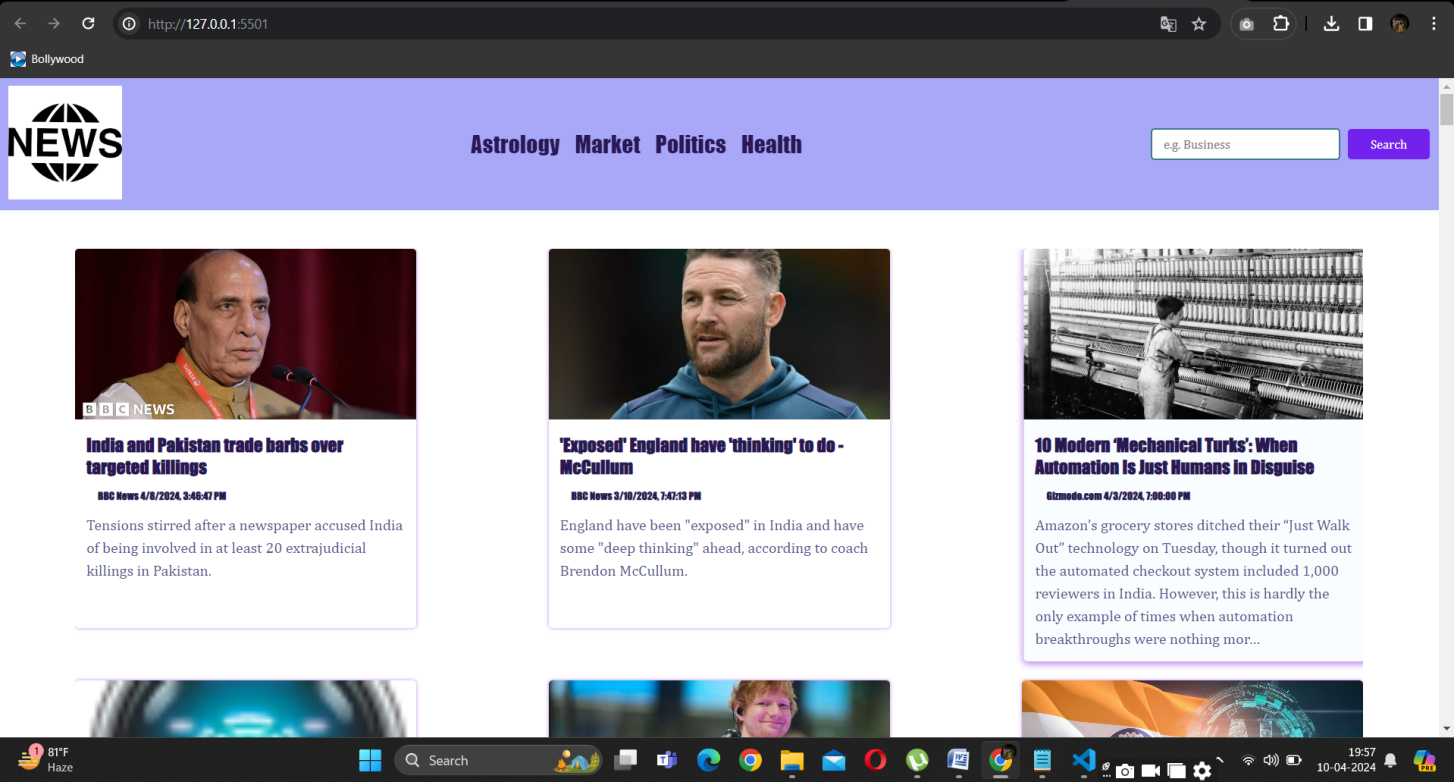
****

**CHAPTER-5**

**RESULT & DISCUSSION**

**5.1 User Interface Representation**

****

****

**CHAPTER-6**

**SUMMARY & CONCLUSION**

In conclusion, the News Search API Mini Project presents a valuable solution for streamlining the process of accessing and discovering news content online. By integrating a News API into a web application, users can efficiently search for relevant news articles tailored to their interests and preferences. Throughout the development process, key aspects such as user interface design, API integration, data presentation, and security measures have been carefully considered to ensure a seamless and user-friendly experience.

The project offers numerous advantages, including efficient news discovery, real-time updates, personalized content, and enhanced user experience. Moreover, it serves as a practical learning opportunity for developers to hone their skills in web development, API integration, and user interface design. By leveraging modern web technologies and best practices, the News Search API Mini Project provides a robust and feature-rich platform for accessing news content, empowering users to stay informed and engaged with the latest developments in their areas of interest.

**CHAPTER-7**

**FUTURE SCOPE**

1. Advanced Search Filters: Implement additional search filters such as date range, language, location, or sentiment analysis to provide users with more precise control over their search queries.
2. User Authentication and Profiles: Introduce user authentication mechanisms to enable personalized features such as saved searches, bookmarking favorite articles, and personalized recommendations based on browsing history.
3. Content Curation and Recommendation Engine: Develop algorithms to analyze user behavior and preferences, providing curated news recommendations tailored to individual interests and browsing patterns.
4. Multimedia Support: Enhance the application to support multimedia content such as videos, podcasts, and interactive graphics, expanding the types of news content available to users.
5. Social Media Integration: Integrate social media sharing functionality to allow users to easily share interesting news articles with their networks on platforms like Twitter, Facebook, or LinkedIn

**Bibliography(Refrence)**

1. C Grozea, [DC Cercel](https://scholar.google.co.in/citations?user=jdsY1HEAAAAJ&hl=en&oi=sra), [C Onose](https://scholar.google.co.in/citations?user=roW72ygAAAAJ&hl=en&oi=sra)… - 2017 16th RoEduNet …, 2017 - ieeexplore.ieee.org

… Interface (API) provided by the platform developers.

1. [I Hopkinson](https://scholar.google.co.in/citations?user=ER9SufIAAAAJ&hl=en&oi=sra), S Maude, [M Rospocher](https://scholar.google.co.in/citations?user=wkAcWjMAAAAJ&hl=en&oi=sra) - ISWC (Developers Workshop), 2014 - ceur-ws.org