# **Business Requirements Specification(BRS)**

| **Document Title** | **News Aggregation System (NAS)** |
| --- | --- |
| **Version** | **1.0** |
| **Date** | **30/02/2025** |
| **Prepared by** | **Team NAS** |
| **Approved by** |  |

### **Table of Contents**

1. **Introduction** 1.1 Background  
    1.2 Objectives  
    1.3 Scope
2. **Key Stakeholders**
3. **Component Diagram**
4. **Components** 4.1 News Collection  
    4.2 Headline Categorization  
    4.3 Source Verification  
    4.4 Filtering and Sorting  
    4.5 Real-Time Updates  
    4.6 Sentiment Analysis  
    4.7 Summary Generation  
    4.8 User Engagement
5. **Non-Functional Requirements**
6. **System Design**
7. **Data Flow Diagrams**
8. **Project Constraints**
9. **Cost-Benefit Analysis**

## **1. Introduction**

### **1.1 Background**

In today's fast-paced digital world, staying informed about current events requires accessing multiple news sources. However, this process can be time-consuming and fragmented, as users must manually search for articles across various platforms. The News Aggregation System (NAS) addresses this challenge by consolidating news articles from diverse sources under unified incident-based headlines. By leveraging advanced technologies such as Natural Language Processing (NLP), web scraping, and real-time updates, NAS provides users with a comprehensive and multi-perspective view of global events.

### **1.2 Objectives**

The primary objective of NAS is to streamline the process of collecting, categorizing, and presenting news articles from multiple sources. Specifically, NAS aims to:

* Enhance accessibility and organization of news.
* Provide users with real-time updates on breaking news.
* Enable filtering and sorting of articles based on date, source, and relevance.
* Offer sentiment analysis and summary generation for quick insights.
* Foster an informed and unbiased understanding of global events.

### **1.3 Scope**

The scope of NAS includes:

* Collecting news articles from APIs, RSS feeds, and web scraping.
* Grouping articles under incident-based headlines using NLP.
* Verifying the credibility of news sources.
* Providing filtering, sorting, and real-time update functionalities.
* Offering sentiment analysis, summary generation, and user engagement features.

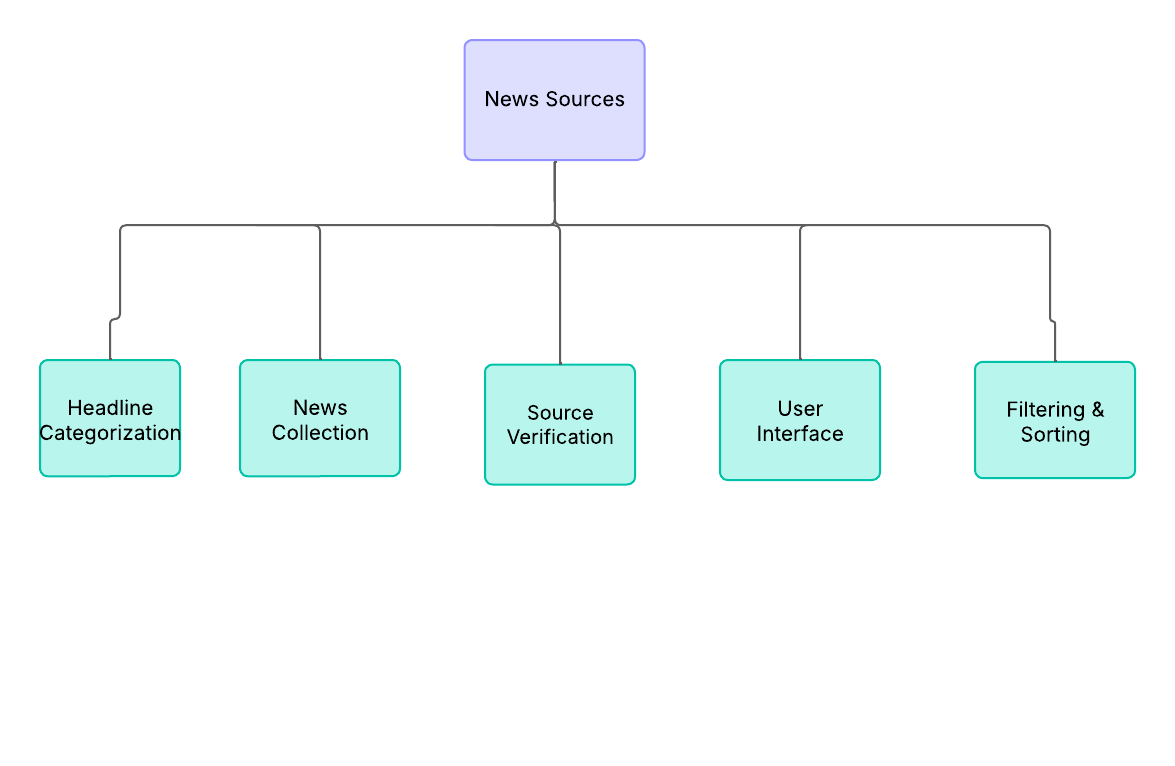
## 

## **2. Key Stakeholders**

| **Name** | **Role** |
| --- | --- |
| General Readers | Individuals seeking consolidated news views. |
| Journalists | Professionals requiring diverse perspectives on events. |
| Researchers | Academics analyzing trends in media coverage. |
| Investigators | Individuals researching news for legal, security, or forensic purposes. |
| National Policymakers | Government officials using aggregated news for policy-making. |
| Politicians | Elected representatives monitoring media coverage and public sentiment. |
| System Administrators | Responsible for maintaining and updating the system. |

## **3. Component Diagram**

The following diagram illustrates the components of NAS:



## 

## 

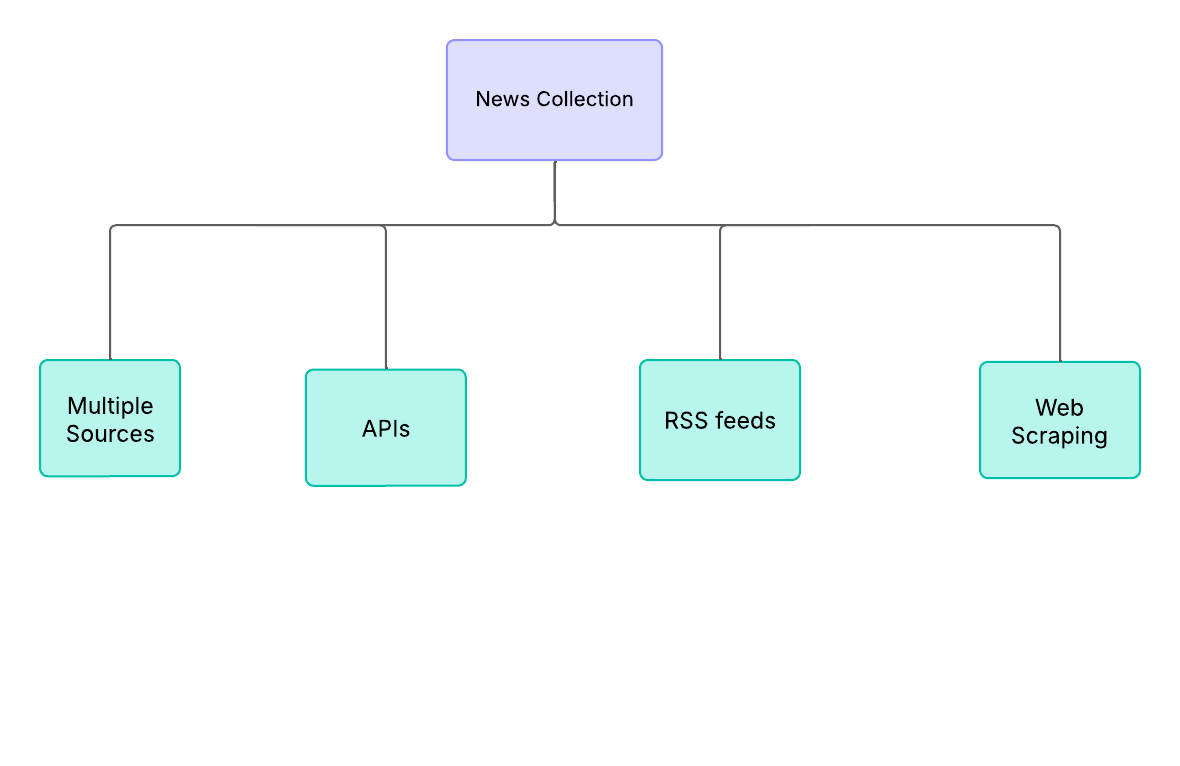
## 

## **4. Components**

### **4.1 News Collection**

#### 

The News Collection module gathers articles from multiple sources, including APIs, RSS feeds, and web scraping.



#### 

#### 

#### **Functional Requirements**

* **Source Integration**

Integrate with APIs, RSS feeds, and public websites.

* **Data Extraction**

Extract article titles, content, metadata, and publication dates.

* **Periodic Updates:**

Periodically check for new articles.

#### **Non-Functional Requirements**

* **Performance**

Handle large volumes of data efficiently.

* **Scalability**

Scale to accommodate increasing numbers of sources.

### **4.2 Headline Categorization**

This module groups articles under incident-based headlines using NLP techniques.

#### **Functional Requirements**

* **Text Analysis**

Analyze article content to identify common incidents.

* **Categorization**

Group articles under unified headlines.

* **Manual Override**

Allow administrators to manually categorize articles.

#### **Non-Functional Requirements**

* **Accuracy**

Ensure high accuracy in categorization.

* **Usability**

Provide an intuitive interface for manual categorization.

### **4.3 Source Verification**

This module verifies the credibility of news sources.

#### **Functional Requirements**

* **Source Validation**

Check sources against predefined criteria.

* **Flagging**

Flag potentially unreliable sources.

#### **Non-Functional Requirements**

* **Security**

Protect source verification data.

* **Reliability**

Minimize false positives in flagging.

### **4.4 Filtering and Sorting**

This module allows users to filter and sort articles based on parameters like date, source, and relevance.

#### **Functional Requirements**

* **Filter Options**

Provide filters for date, source, and relevance.

* **Sorting Options:**

Allow sorting by "Most Recent" and "Most Relevant."

#### **Non-Functional Requirements**

* **Performance**

Ensure quick response times for filtering and sorting.

## **5. Non-Functional Requirements**

| **Name** | **Explanation** |
| --- | --- |
| Security | Encrypt user data and ensure secure authentication. |
| Usability | Develop an intuitive and accessible user interface. |
| Performance | Optimize system performance for concurrent users. |
| Scalability | Scale to accommodate increasing traffic. |
| Reliability | Ensure high availability and minimal downtime. |

## **6. System Design**

The NAS will be designed as a cloud-based system with the following components:

| **Name** | **Explanation** |
| --- | --- |
| Frontend | Web and mobile applications for user interaction. |
| Backend | APIs for news collection, categorization, and user engagement. |
| Database | Store articles, user data, and system logs. |
| Third-Party Integrations | Leverage APIs for sentiment analysis and NLP. |

## 

## 

## 

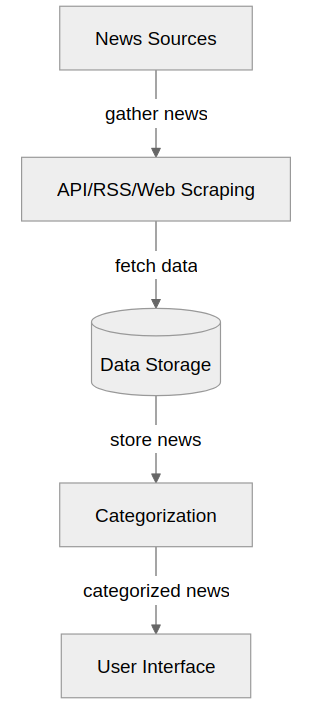
## 

## 

## 

## **7. Data Flow Diagrams**

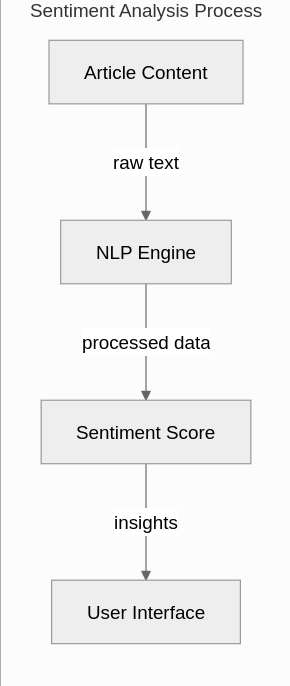
A data flow chart of the **News Aggregation System** illustrates the stages from news collection to presentation, covering data extraction, categorization, and display. It ensures transparency and efficiency in delivering relevant news to users.



#### **Fig: Data Flow of News Collection**

#### **Data Flow of Sentiment Analysis**

This diagram illustrates the flow of sentiment analysis, from **Article Content** processing through an **NLP Engine**, generating a **Sentiment Score**, and displaying results in the **User Interface**.



#### **Fig: Data Flow of Sentiment Analysis**

#### **8. Project Constraints:**

| Constraint | Description |
| --- | --- |
| Legal Compliance | Discuss potential challenges related to copyright laws, fair use policies, and regulatory requirements for aggregating content from multiple sources. |
| Technical Limitations | Consider API restrictions, data processing challenges, and infrastructure scalability. |
| Data Accuracy & Bias | Explore how misinformation, biased reporting, and credibility of sources impact system reliability. |
| Data Accuracy & Bias | Explore how misinformation, biased reporting, and credibility of sources impact system reliability. |
| Security & Privacy | Assess risks related to data protection, user privacy, and cyber threats. |

**9. Cost-Benefit Analysis:**

| **Cost** | **Benefit** |
| --- | --- |
| Development & Maintenance Costs | Estimate costs for software development, server hosting, AI/ML integration, and continuous updates. |
| Operational Efficiency Gains | Discuss how automation reduces manual news tracking efforts and improves accessibility. |
| User Engagement & Revenue Potential | Evaluate the potential benefits of ad revenue, premium subscriptions, or partnerships with media outlets. |
| Scalability & Future Growth | Consider the long-term benefits of system scalability and its adaptability to emerging technologies. |
| Competitive Advantage | Explain how an effective news aggregation platform provides unique value over traditional news sources. |