# Software Requirements Specification (SRS)

**For**

## Breaking News Application

### Version 1.0

Prepared by: Mostafa Ehab Elabsawy &

Yahya Ahmed Ramdan

Organization: Faculty of Electronic Engineering software engineering course

Date Created: 1/12/2024

***Copyright © 1999 by Karl E. Wiegers. Permission is granted to use, modify, and distribute this document.***

1. **Introduction**……………………………………………………………………………………………………………………………………….………**1**

* 1. Purpose………………………………………………………………………………………….…………………..……………………………………….1

1.2 Document Conventions…………………………………………………………………………………..……………………………………………1

1.3 Intended Audience and Reading Suggestions………………………………………………….……………………………………………1

1.4 Product Scope………………………………………………………………………………………………………………………………………………1

1.5 References……………………………………………………………………………………………………………………………………………………1

2. **Overall Description……………………………………………………………………………………………………….…………………….2**

2.1 Product Perspective……………………………………………………………………………………………………………………………….….…2

2.2 Product Functions…………………………………………………………………………………………………………………………………………2

2.3 User Classes and Characteristics…..………………………………………………………………………………………………………………2

2.4 Operating Environment……………………………………………………………………………………………………………………….……….3

2.5 Design and Implementation Constraints…………………………………………………………………………………………….….…….3

2.6 User Documentation…………………………………………………………………………………………………………………………………….3

2.7 Assumptions and Dependencies…………………………………………………………………………………………………………..……...3

3. **Detailed Requirements ……………………………………………………………………………………….……4**

3.1 External Interfaces Requirements…………………………………………………………………………….….3

3.2 Hardware Interfaces…………………………………………………………………………………………………………………………………….3

3.3 Software Interfaces………………………………………………………………………………………………………………………………………3

3.4 Communications Interfaces………………………………………………………………………………………………………………………….3

3.3 **Nonfunctional Requirements…………………………………………………………………………5**

5.1 Performance Requirements…………………………………………………………………………………………………………………………5

5.2 Security Requirements…………………………………………………………………………………………………………………………………5

5.3 Software Quality Attributes………………………………………………………………………………………………………………………….5

**Other Requirements……………………………………………………………………..5**

Revision History

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Date** | **Reason For Changes** | **Version** |
| **Yahya Ahmed Ramadan** | **1/12/2024** | **Initiate the document** | **1.0** |
| **Mostafa Ehab Elabsawy** | **1/12/2024** | **Added derailed requirements** | **1.0** |

**1.Introduction**

**1.1 Purpose**

This document specifies the requirements for the "**Breaking News Web Application.**" The application provides users with **categorized news**, **trending headlines**, and **search functionality**. It also includes a **Source Redirection feature**, enabling users to access full articles hosted on external websites.

**1.2 Document Conventions**

1. Requirements are uniquely identified with codes (e.g., REQ-1, REQ-2).
2. Key terms and abbreviations are defined in the glossary.

**1.3 Intended Audience and Reading Suggestions**

* **Developers**: Understand the system’s functional and non-functional requirements.
* **Stakeholders**: Review and validate that the project meets business objectives.
* **Testers**: Create test plans and cases based on detailed requirements.

**1.4 Product Scope**

The "Breaking News Web Application" allows users to view trending headlines, browse categorized news, search for specific articles, and access the full articles on external websites via the Source Redirection feature.

**1.5 References**

1. News API Documentation - [News API Docs] (<https://newsapi.org/docs>)
2. Web development standards: HTML5, CSS3, JavaScript ES6.
3. Modern browser compatibility guidelines

**2. Overall Description**

**2.1 Product Perspective**

The product is a standalone web application that delivers real-time news from multiple sources using the News API, ensuring always-current content. It acts as a centralized platform where users can easily access trending headlines, categorized news, and search for specific topics. To ensure seamless access to real-time data, the system relies on a stable internet connection, allowing users to also view full articles via a "Read More" button that redirects them to the original source in a new browser tab.

**2.2 Product Functions**

* Display Headlines Trends on the main page.
* Browse Categorized News (e.g., Business, Sports).
* Search functionality for finding articles by keywords.
* Source Redirection Users can click the "Read More" button to view the full

article on the original website.

A diagram of a data flow

Description automatically generated

**2.3 User Classes and Characteristics**

* **General Users**: Individuals looking for quick access to news.
* **Power Users:** Users who frequently search for specific news topics.

**2.4 Operating Environment**

* + **Browser** : Google Chrome, Mozilla Firefox, Microsoft Edge.
  + **Devices** : Desktop, tablet, and mobile devices.
  + **Network** : Requires an active internet connection.

**2.5 Design and Implementation Constraints**

- Limited to the API’s request rate and available endpoints.  
- Requires an active internet connection.

**2.6 User Documentation**

* User manual with a guide for navigation and functionality.
* Online help documentation.
* Error messages when there is no result or connectivity issue.

**2.7 Assumptions and Dependencies**

* Assumes availability of the News API.
* Assumes users have internet access.

1. **Detailed requirements**
   1. **External Interface Requirements**

**3.1.1 User Interfaces**

1. **Navbar:** A navigation bar for easy selection of categories such as Business, Sports, Technology, and Entertainment. It allows users to quickly access different news sections.
2. **Search Bar**: An input field that allows users to perform keyword-based searches to find specific news articles. The search results will update dynamically based on the entered keyword.
3. **News Container**: Displays the latest news articles as the default page when the user first interacts with the application. This section is dynamically populated with the most recent articles from various sources.
4. **Source Redirection Button**: A clickable "Read More" button integrated within each news item. When clicked, this button redirects the user to the original article on the news source's website, allowing the user to view the full content.

**3.1.2 Hardware Interfaces**

Not applicable (web-based application).

**3.1.3 Software Interfaces**

1. Integrate with the News API to fetch articles.
2. Uses HTML, CSS, and JavaScript for frontend development.

**3.1.4 Communications Interfaces**

* The app communicates with the News API using HTTPS.
* External links are opened in a new browser tab using the Source Redirection feature.

* 1. **Functional Requirements**

We describe the functional requirements by using textual use cases.

**Use Case 1:** View Headlines

**Main Actor**: General User, Power User

**Preconditions**:

* 1. The system is connected to the internet.
  2. The user has accessed the application.

**Main scenario:**

1. The user opens the web application.
2. The system fetches the latest news from the API.
3. The system displays the fetched headlines to the user.

**Exceptions:**

**2. Headlines fetch Failure**:

1. If the API request fails, the system displays an error message:

" No Data Found."

**Use Case 2: View Categorized News**

**Main Actors**: General User, Power User.

**Preconditions**:

1. The user has accessed the application.
2. The system is connected to the internet

**Main scenario:**

1. The user selects a category (e.g., Business, Sports, Technology) from the navigation menu.
2. The system sends a request to the News API for articles in the chosen category.
3. The system displays the articles from the selected category in a grid layout, showing titles, summaries, and images.

**Extensions**:

**2. Category Fetch Failure**:

1. If the system fails to retrieve articles for the selected category, it displays an error message: "No Data Found."

**Use Case 3:** Search for News Articles

**Main Actors**: General User, Power User.

**Preconditions**:

* 1. The system is connected to the internet.
  2. The user has accessed the application.
  3. The search bar is accessed (focused on)

**Main scenario:**

1. The user enters a search keyword into the search bar.
2. The system validates the search query.
3. The system sends the query to the News API and retrieves matching news articles.
4. The system displays the search results to the user.

**Extensions**:

**2. Invalid Input:**

1. If the user enters an empty or invalid query, the system displays an error message: *"*Please enter a valid search term*."*

**3. Query Fetch Failed:**

1. If the system fails to retrieve articles for the selected category, it displays an error message: "No Data Found"

**Use Case 4:** Read More Details

**Main Actors**: General User, Power User.

**Preconditions**:

The user is viewing a list of news.

**Main scenario:**

1. The user selects the "Read More" button on a news card.
2. The system retrieves the article's link from the News API.
3. The system redirects the user to the article's source.

**Extensions**:

**2.** **Missing Article Link:**

1. If the article link is unavailable, the system displays an error message: "This article is currently unavailable."

**3.3 Nonfunctional Requirements**

**3.3.1 Performance Requirements**

* Load time for any page should not exceed 3 seconds on a standard broadband connection.
* Support up to 500 simultaneous users.

**3.3.2 Security Requirements**

* All API calls must use HTTPS to ensure secure communication.

**3.3.3 Software Quality Attributes**

* **Usability**: Intuitive navigation and responsive design.
* **Reliability**: High availability during usage.
* **Maintainability**: Modular JavaScript code for ease of updates.

**3.4 Other Requirements**

Currently no other requirements are known. Should any other requirements occur during the software

implementation process or while running tests, they will be specified in an appendix

* 1. **Decision Tables**

## Decision Table 1: Search Query Handling

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Condition/Actions** | **Rule 1** | **Rule 2** | **Rule 3** | **Rule 4** |
| Valid Query | Y | Y | Y | N |
| API Response | Y | Y | N | - |
| Articles Found | Y | N | - | - |
| Display articles | **x** |  |  |  |
| No Data Found' |  | **x** |  |  |
| 'Unable to fetch data' |  |  | **x** |  |
| ‘Invalid query' |  |  |  | **x** |

## Decision Table 2: Article Display Behavior

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Condition/Action** | **Rule 1** | **Rule 2** | **Rule 3** | **Rule 4** | **Rule 5** |
| Article Link Exist | Y | Y | Y | Y | N |
| Image Available | Y | Y | N | N | - |
| Description Available | Y | N | Y | N | - |
| Display news card | **x** |  |  |  |  |
| Display news card without description |  | **x** |  |  |  |
| Display news with placeholder image |  |  | **x** |  |  |
| Doesn’t display aritcle |  |  |  | **x** |  |
| Display missing link error |  |  |  |  | **x** |

* 1. **Data Flow Diagrams**

The Data Flow Diagram (DFD) depicts the flow of data and the processes involved in the search functionality, It illustrates how the system handles user input, processes search queries, and communicates with external services to retrieve news data

**A diagram of a flowchart

Description automatically generated**

* 1. **Requirements traceability Matrix**

Before presenting the Requirements Traceability Matrix, we have identified the following functional requirements for the Application:

1. **REQ1**: Fetch Data from API
2. **REQ2**: Format Articles into Card Format
3. **REQ3**: View Headlines
4. **REQ4**: View Categorized News
5. **REQ5**: Search for News Articles
6. **REQ6**: Read More Details

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Requirements** | **REQ1** | **REQ2** | **REQ3** | **REQ4** | **REQ5** | **REQ6** |
| **REQ1** | **X** |  | **X** | **X** | **X** |  |
| **REQ2** |  | **X** | **X** | **X** | **X** |  |
| **REQ3** |  |  | **X** |  |  |  |
| **REQ4** |  |  |  | **X** |  |  |
| **REQ5** |  |  |  |  | **X** |  |
| **REQ6** |  |  |  |  |  | **X** |