Functional Requirement 1 (FR1): Sorting Writings

1. Introduction:

The sorting functionality enables users to organize the list of writings displayed on the website based on various criteria such as author, date written, or title. This enhances user experience by providing flexibility in navigating and accessing the writings according to their preferences.

1. Scope:

The sorting feature will be implemented within the website's interface for users to interact with. It includes providing sorting options, automatic rearrangement of the writings based on selected criteria, and user interface elements to facilitate ease of use.

Functional Requirements:

* Sorting Options (FR1.1):
  + The website interface shall provide options for users to select the sorting criteria from a predefined list, including but not limited to:
    - Author: Sort writings alphabetically based on the author's name.
    - Date Written: Sort writings chronologically based on the date they were written.
    - Title: Sort writings alphabetically based on their titles.
* Automatic Rearrangement (FR1.2):
  + Upon selecting a sorting criterion, the list of writings displayed on the website shall be automatically rearranged according to the chosen criterion.
* User Interface (FR1.3):
  + The sorting functionality should be intuitive and easy to use, with clear indications of the currently selected sorting criterion.
* Ascending and Descending Order (FR1.4):
  + Users should have the option to toggle between ascending and descending order for sorted results.

Use Cases:

* Use Case 1: Sorting Writings
  + Actor: Website User
  + Steps:
    - User accesses the list of writings on the website.
    - User selects the desired sorting criterion from the provided options.
    - The website automatically rearranges the list of writings based on the selected criterion.
    - User views the sorted list of writings according to their preference.

External Interfaces:

* The sorting feature interacts with the website's user interface to provide sorting options and display sorted results to users.

Non-Functional Requirements:

* Performance, usability, and efficiency of the sorting functionality to ensure a seamless user experience.

Assumptions:

* Users are familiar with the concept of sorting and will easily understand how to utilize the sorting options provided.

Dependencies:

* The sorting feature depends on the availability and accuracy of metadata associated with each writing (e.g., author name, date written).

Constraints:

* The sorting functionality should not significantly impact the website's performance or responsiveness, even with a large number of writings to sort.

Acceptance Criteria:

* Users can successfully sort the list of writings by selecting different sorting criteria from the provided options.
* The sorted list of writings accurately reflects the chosen sorting criterion, with clear indications of the sorting order (ascending or descending).
* Users find the sorting functionality intuitive and user-friendly, requiring minimal effort to use effectively.
* The sorting feature performs efficiently even with a large number of writings in the database, ensuring a smooth user experience.

Functional Requirement 2 (FR2): Filtering Writings

Introduction:

The filtering functionality allows users to refine the list of writings displayed on the website by applying specific criteria or categories. This enhances user experience by enabling users to focus on writings that match their interests or preferences.

Scope:

The filtering feature will be integrated into the website's interface, providing options for users to apply filters based on predefined categories or tags associated with each writing.

Functional Requirements:

* Filtering Options (FR2.1):
  + The website interface shall provide options for users to select filtering criteria from a predefined list of categories or tags.
* Dynamic Update (FR2.2):
  + Upon selecting a filtering criterion, the list of writings displayed on the website shall be dynamically updated to show only those that match the selected filters.
* User Interface (FR2.3):
  + The filtering functionality should be intuitive and easy to use, with clear indications of the currently applied filters.
* Multiple Filters (FR2.4):
  + Users should have the option to apply multiple filters simultaneously to narrow down the list of writings based on various criteria.

Use Cases:

* Use Case 2: Filtering Writings
  + Actor: Website User
  + Steps:
    - User accesses the list of writings on the website.
    - User selects one or more filtering criteria from the provided options.
    - The website dynamically updates the list of writings to display only those that match the selected filters.
    - User views the filtered list of writings according to their preferences.

External Interfaces:

* The filtering feature interacts with the website's user interface to provide filtering options and display filtered results to users.

Non-Functional Requirements:

* Performance, usability, and efficiency of the filtering functionality to ensure a seamless user experience.

Assumptions:

* Users understand the purpose and relevance of each filtering criterion and will select filters accordingly.

Dependencies:

* The filtering feature relies on accurate categorization or tagging of writings in the database.

Constraints:

* The filtering functionality should not significantly impact the website's performance or responsiveness, even when applying multiple filters to a large dataset of writings.

Acceptance Criteria:

* Users can successfully filter the list of writings by selecting one or more filtering criteria from the provided options.
* The filtered list of writings accurately reflects the selected filters, displaying only those that match the specified criteria.
* Users find the filtering functionality intuitive and user-friendly, with clear indications of the currently applied filters.
* The filtering feature performs efficiently even with a large number of writings in the database, ensuring a smooth user experience.

Functional Requirement 3 (FR3): Searching Writings

Introduction:

The searching functionality enables users to find specific writings by entering keywords or phrases. This feature enhances user experience by providing a quick and efficient way to locate relevant content within the website.

Scope:

The searching feature will be integrated into the website's interface, allowing users to input search queries and receive relevant results based on matching text within the writings' content, titles, or other relevant fields.

Functional Requirements:

* Search Input (FR3.1):
  + The website interface shall provide a search input field where users can enter keywords or phrases.
* Search Execution (FR3.2):
  + Upon entering a search query, the website shall execute the search and retrieve writings that contain the matching text within their content, titles, or other relevant fields.
* Search Results Display (FR3.3):
  + The search results shall be displayed to the user in a clear and organized manner, showing relevant information such as the title, author, and snippet of content for each matching writing.
* Advanced Search Options (FR3.4):
  + Users should have the option to access advanced search options, allowing them to refine their search by specifying additional criteria such as date range, author, or category.

Use Cases:

* Use Case 3: Searching Writings
  + Actor: Website User
  + Steps:
    - User accesses the search input field on the website.
    - User enters keywords or phrases into the search input.
    - The website executes the search and retrieves relevant writings based on the entered query.
    - User views the search results and selects a writing of interest for further exploration.

External Interfaces:

* The searching feature interacts with the website's user interface to provide search input and display search results to users.

Non-Functional Requirements:

* Performance, accuracy, and efficiency of the searching functionality to ensure quick and relevant results for users' search queries.

Assumptions:

* Users understand the purpose and functionality of the search feature and will enter relevant search queries to find desired content.

Dependencies:

* The searching feature relies on the availability and indexing of text content within the writings stored in the database.

Constraints:

* The searching functionality should perform efficiently even with a large database of writings, ensuring quick response times for users' search queries.

Acceptance Criteria:

* Users can successfully enter search queries into the provided search input field.
* The website executes the search queries accurately and retrieves relevant writings based on matching text within their content, titles, or other relevant fields.
* Search results are displayed to users in a clear and organized manner, providing relevant information for each matching writing.
* Users can access advanced search options to refine their search criteria and obtain more specific search results if needed.

Functional Requirement 4 (FR4): User Interaction with Writings

* Introduction:

User interaction functionality enables users to engage with individual writings by accessing detailed information and navigating through the list of writings seamlessly. This enhances user experience by providing convenient ways to explore and interact with the content.

* Scope:

The user interaction features will be integrated into the website's interface, allowing users to view detailed information about each writing, navigate through the list of writings, and access additional functionalities such as bookmarking or sharing.

Functional Requirements:

* Viewing Detailed Information (FR4.1):
  + Users should be able to access detailed information about each writing, including the author, title, date written, and any associated tags or categories.
* Navigating Through Writings (FR4.2):
  + Users should be able to navigate through the list of writings easily, with options for pagination or infinite scrolling to access more writings.
* Bookmarking or Saving Writings (FR4.3):
  + Users should have the option to bookmark or save writings for future reference, enabling them to easily access their favorite or frequently viewed writings.
* Sharing Writings (FR4.4):
  + Users should be able to share writings with others via email, social media, or other communication channels, promoting collaboration and knowledge sharing among users.

Use Cases:

* Use Case 4: Interacting with Writings
  + Actor: Website User
  + Steps:
    - User selects a writing from the list of writings displayed on the website.
    - User accesses detailed information about the selected writing, including the author, title, date written, and associated tags or categories.
    - User navigates through the list of writings using pagination or infinite scrolling to explore more content.
    - User bookmarks or saves a writing for future reference if desired.
    - User shares a writing with others via email, social media, or other communication channels.

External Interfaces:

* The user interaction features interact with the website's user interface to provide options for viewing, navigating, bookmarking, and sharing writings with users.

Non-Functional Requirements:

* Usability, efficiency, and accessibility of the user interaction features to ensure a seamless and enjoyable user experience.

Assumptions:

* Users understand the purpose and functionality of the user interaction features and will utilize them to explore and engage with the content effectively.

Dependencies:

* The user interaction features rely on the availability and accuracy of metadata associated with each writing, as well as integration with external sharing platforms (e.g., social media).

Constraints:

* The user interaction features should not significantly impact the website's performance or responsiveness, even with a large number of users accessing and interacting with the content simultaneously.

Acceptance Criteria:

* Users can access detailed information about each writing, including the author, title, date written, and associated tags or categories.
* Users can navigate through the list of writings easily using pagination or infinite scrolling to access more content.
* Users can bookmark or save writings for future reference, and the bookmarked writings are accessible from their user account.
* Users can share writings with others via email, social media, or other communication channels, and the shared content is displayed accurately and includes relevant information about the writing.

Functional Requirement 5 (FR5): Performance and Scalability

Introduction:

Performance and scalability are crucial aspects of the website to ensure that it can handle a large number of users and writings while maintaining responsiveness and efficiency. These aspects directly impact user experience and satisfaction with the website.

Scope:

The performance and scalability requirements encompass various aspects of the website's design, infrastructure, and implementation to ensure optimal performance and scalability under different usage scenarios.

Functional Requirements:

* Response Time (FR5.1):
  + The website shall respond to user actions (e.g., sorting, filtering, searching) within a specified timeframe (e.g., 2 seconds) under normal load conditions.
* Scalability (FR5.2):
  + The system architecture shall be designed to scale horizontally and vertically to accommodate an increasing number of users and writings without significant degradation in performance.
* Database Optimization (FR5.3):
  + Database queries and operations related to retrieving, updating, or deleting writings shall be optimized to minimize response times and resource utilization.
* Caching Mechanism (FR5.4):
  + A caching mechanism shall be implemented to cache frequently accessed data and reduce the need for repeated database queries, improving overall system performance.
* Load Testing (FR5.5):
  + The website shall undergo regular load testing to assess its performance and scalability under various load levels, ensuring that it meets the specified performance criteria.

Use Cases:

* Use Case 5: Ensuring Performance and Scalability
  + Actor: System Administrator
  + Steps:
    - System administrator monitors website performance metrics such as response time and resource utilization.
    - In case of performance degradation or increased load, the system administrator scales up the infrastructure by adding more resources (vertical scaling) or adding more servers (horizontal scaling).
    - System administrator optimizes database queries and operations to improve overall system performance.
    - System administrator configures and manages caching mechanisms to reduce the load on the database and improve response times.

External Interfaces:

* Performance monitoring tools and load testing frameworks are used to assess and optimize website performance.

Non-Functional Requirements:

* Reliability, efficiency, and scalability of the website infrastructure and implementation to ensure consistent and responsive user experience.

Assumptions:

* The website's performance and scalability requirements are based on expected usage patterns and growth projections.

Dependencies:

* Performance and scalability depend on various factors such as server hardware, network infrastructure, database optimization, and application design.

Constraints:

* Budget constraints may limit the extent to which performance optimization measures can be implemented.

Acceptance Criteria:

* The website responds to user actions within the specified timeframe (e.g., 2 seconds) under normal load conditions.
* The website scales seamlessly to accommodate an increasing number of users and writings without significant degradation in performance.
* Database optimization measures result in improved response times for database queries and operations related to writings.
* The caching mechanism effectively reduces the load on the database and improves overall system performance, as demonstrated by performance metrics and load testing results.