Software Requirements

Specification

For

Bamboo Virtual Library

Version <1.0>

Prepared by

Group Name: AlgoRhythm

|  |  |  |
| --- | --- | --- |
| Johanne Marlowe E. Sanchez | <student #> | jesanchez@student.apc.edu.ph |
| Carl Von A. Nicanor | <student #> | canicanor@student.apc.edu.ph |
| Aldrin Dave B. Alia | <student #> | abalia@student.apc.edu.ph |
| Henzo Ivan G. Valguna | <student #> | hgvalguna@student.apc.edu.ph |
| Gabrielle Ann S. Yogawin | <student #> | gsyogawin@student.apc.edu.ph |

|  |  |
| --- | --- |
| Instructor: | Manuel Sebastian S. Sanchez |
| Course: | SOFTDEV |
|  |  |
|  |  |
| Date: | October 17, 2024 |
|  |  |

Contents

Contents iii

Revisions iii

1 Introduction 1

1.1 Document Purpose 1

1.2 Product Scope 1

1.3 Intended Audience and Document Overview 1

1.4 Definitions, Acronyms and Abbreviations 2

2 Overall Description 3

2.1 Product Overview 3

2.2 Product Functionality 3

2.3 Design and Implementation Constraints 3

2.4 Assumptions and Dependencies 4

3 Specific Requirements 5

3.1 External Interface Requirements 5

3.2 Functional Requirements 6

3.3 Use Case Model 7

4 Other Non-functional Requirements 10

4.1 Performance Requirements 10

4.2 Safety and Security Requirements 10

4.3 Software Quality Attributes 11

Appendix A – Data Dictionary/ERD 12

Revisions

| Version | Primary Author(s) | Description of Version | Date Completed |
| --- | --- | --- | --- |
| Document Making  Section 1: Introduction | Johanne Marlowe E. Sanchez |  | 10/09/24 |
| Document Making  Section 2:  Overall Description | Johanne Marlowe E. Sanchez |  | 11/09/24 |
| Document Making  Section 3:  Specific Requirements | Johanne Marlowe E. Sanchez  Henzon Ivan G. Valguna (Use Case Model) |  | 13/09/24 |
| Document Making  Section 4:  Other Non-Functional Requirements | Johanne Marlowe E. Sanchez |  | 16/09/24 |
| Document Making  Appendix A: Data Dictionary/ERD | Carl Von A. Nicanor |  | 17/09/24 |

# 

# Introduction

## Document Purpose

This document outlines the software requirements for the "Bamboo Warriors Repository of Knowledge" website project. The website is designed to serve as a comprehensive and authoritative resource on bamboo in the Philippines. The purpose of this Software Requirements Specification (SRS) is to provide clear and detailed descriptions of the functionality, features, and constraints of the system. This document will guide the development team through the implementation phases, ensuring the project meets the needs of its stakeholders.

The scope of the product covered by this SRS includes the development of key components, such as a knowledge base, a repository of multimedia resources, an interactive forum, and tools for managing and updating content. This SRS focuses on the website's essential features for its first release, with future iterations anticipated for enhancements.

## Product Scope

The "Bamboo Warriors Repository of Knowledge" website is a platform that provides an educational and collaborative space dedicated to bamboo in the Philippines. It aims to promote awareness about bamboo's cultural significance, ecological importance, and economic potential. This website will offer valuable resources to various stakeholders, such as students, researchers, farmers, policymakers, and environmentalists.

The website’s objectives include increasing public awareness of bamboo’s potential as a sustainable material, fostering knowledge sharing, and supporting the development of bamboo-based industries. Key benefits of the website include the ability to access reliable, well-organized content about bamboo, and opportunities for users to engage in discussions and collaboration through forums and shared resources.

## Intended Audience and Document Overview

This SRS document is intended for various stakeholders involved in the development, use, and maintenance of the "Bamboo Warriors Repository of Knowledge" website. The primary audience includes:

* **Developers**: To understand the system's technical requirements and implement the described functionality.
* **Project Managers**: To track project milestones and ensure the product aligns with the client’s objectives.
* **Clients and Professors**: To review and provide feedback on whether the requirements reflect the project’s vision and goals.
* **Testers**: To develop test cases based on the system requirements.
* **Documentation Writers**: To produce user guides and technical documentation that aligns with the system's features.

The document is organized as follows:

* Introduction: Provides an overview of the document, the product’s scope, intended audience, and key definitions.
* Overall Description: Outlines the product, its major functionality, design constraints, assumptions, and dependencies.
* Specific Requirements: Details the functional requirements, use case models, and external interface requirements.
* Non-Functional Requirements: Covers performance, safety, security, and software quality attributes.
* Other Requirements: Any additional requirements not covered in previous sections.
* Appendices: Includes a data dictionary and group log.
* The document should be read sequentially, starting with the introduction for context, followed by the overall description, and finally the specific and non-functional requirements for detailed implementation and design insights. Each section builds upon the previous one to ensure comprehensive understanding of the project.

Readers should begin with the introduction and scope, and then proceed to the sections most relevant to their role in the project.

## Definitions, Acronyms and Abbreviations

* CMS: Content Management System – A platform used to manage and publish content on the website.
* SRS: Software Requirements Specification – A document that outlines the requirements for the software project.
* SEO: Search Engine Optimization – Techniques to increase website visibility in search engine results.
* UX/UI: User Experience/User Interface – Refers to the design and functionality of the website from the user's perspective.
* API: Application Programming Interface – A set of tools for building software applications.
* Agile: A project management methodology that involves iterative development, allowing for flexibility and continuous improvement.

# Overall Description

## Product Overview

The "Bamboo Warriors Repository of Knowledge" website is a new, self-contained product designed to provide information on bamboo in the Philippines. This platform aims to raise awareness about bamboo as an essential resource for the country’s sustainable development while also offering a space for education, collaboration, and innovation. It is not a replacement for any existing system but rather a novel solution that consolidates various types of content into a single, user-friendly interface.

The website will integrate several subsystems, including a knowledge base, a multimedia repository, and a user forum, with interfaces to external systems for managing and retrieving information such as search engines for Search Engine Optimization (SEO), social media platforms for sharing content, and analytics tools for tracking user interactions.

Key Benefits:

1. **Centralized Knowledge Hub**: The website will serve as a comprehensive, accessible repository for all information related to bamboo in the Philippines, offering a valuable resource for students, researchers, entrepreneurs, and policymakers alike.
2. **Enhanced Public Awareness**: By promoting bamboo's ecological, cultural, and industrial significance, the platform will increase public understanding and appreciation of bamboo as a versatile, sustainable material.
3. **Fostering Collaboration**: The website’s interactive features, such as forums and knowledge sharing tools, will enable collaboration across industries and sectors, driving innovation in bamboo-related products and sustainable practices.
4. **Long-Term** **Economic Impact**: With increased awareness and collaboration, the project is expected to drive innovation in bamboo production, processing, and product development, leading to new market opportunities and bolstering the country's economy.

## Product Functionality

The website will perform the following major functions:

1. Knowledge Base: A searchable database of articles, publications, and resources related to bamboo.
2. Multimedia Repository: A repository storing photos, videos, audio, maps, and infographics.
3. Interactive Forum: A platform allowing users to ask questions, share experiences, and collaborate on bamboo-related topics.
4. Content Management: Admin features for uploading, managing, and updating content.
5. Contact and Outreach: A contact page with social media links for communication and feedback.

## Design and Implementation Constraints

The following constraints may limit development options:

1. Hardware Requirements: The system must support high-traffic volumes, particularly during events or content launches. It should be optimized for performance and scalability.
2. Interfaces to External Applications: Integration with social media APIs for content sharing and SEO tools for improving search visibility.
3. Technological Stack: The website will be built using standard web technologies such as HTML5, CSS, JavaScript (likely with frameworks like React or Angular), and a back-end system using Node.js or Python.
4. CMS Requirements: The content management system should allow secure uploads and updates to maintain content integrity.
5. Security Considerations: The website must include security features such as user authentication, SSL encryption, and secure database access. Data protection policies must comply with local data privacy laws.
6. Design Standards: The COMET method (Concurrent Object Modeling and Architectural Design Method) will be used for software design, and the UML (Unified Modeling Language) will be used to represent system architecture.

* COMET Method Reference: Gomaa, H. (2011). Software Modeling and Design: UML, Use Cases, Patterns, and Software Architectures.
* UML Reference: Booch, G., Jacobson, I., & Rumbaugh, J. (2005). The Unified Modeling Language User Guide.

## Assumptions and Dependencies

The following assumptions and dependencies could impact the project:

1. Third-Party Dependencies: The project assumes the availability of certain third-party APIs (e.g., for SEO, social media, and analytics). If these services become unavailable or change their access policies, it could impact project functionality.
2. Hosting Environment: The website will be hosted on a cloud platform (e.g., AWS or Google Cloud), and its performance will depend on the availability and scalability of the hosting provider.
3. Content Management: The content plan assumes collaboration with bamboo experts and researchers who will provide accurate and up-to-date information.
4. Compliance Requirements: The system assumes compliance with local data privacy laws (e.g., Data Privacy Act of the Philippines). If these laws change, additional development might be needed.

# Specific Requirements

## External Interface Requirements

### User Interfaces

The main user interface for the Bamboo Warriors Repository of Knowledge website will feature an intuitive and interactive design, accessible via standard web browsers. The interface will allow seamless navigation through key sections, such as:

* Articles: Users will be able to read in-depth articles, with options to filter by category or author.
* Multimedia Resources: Videos, images, and other media will be available for users to explore, with tools for viewing, sharing, or downloading.
* Forums: Users can participate in discussions, post questions, and engage with the community.
* Search Bar: A prominently placed search bar will allow users to quickly find specific content, articles, or forum posts.
* User Profiles: Personalized profiles will enable users to track forum participation, save favorite articles, and manage personal settings.
* Catalogue: The content of Bamboo Warriors Repository of Knowledge website are now categorized and for user’s easier access.
* Chatbot: Questions from users that are related to bamboo will be answered. Non-bamboo related questions will not be considered.

The design will include a clean layout with easily accessible buttons, dropdown menus, and a knowledge base, ensuring efficient navigation and user-friendly interaction with the content.

### Hardware Interfaces

The Bamboo Warriors Repository of Knowledge website will be hosted on GitHub Codespaces, which serves as the primary hardware interface for the project. Codespaces provides the necessary cloud-based environment for hosting the website, with all interactions between the software and hardware occurring through this platform. The website will be designed to be accessed and managed exclusively on PCs or laptops, ensuring optimal functionality in these environments. Key characteristics of the hardware interface include:

* Hosting Environment: The website will be deployed and run entirely within GitHub Codespaces, eliminating the need for physical server hardware.
* Storage and File Size: The file size and storage capacity will be dependent on the allocated space within the Codespaces environment, ensuring that all project files and assets fit within these limits.
* Access Devices: The website will be developed specifically for use on PCs or laptops, with an interface optimized for desktop browsing and management.

This setup ensures a flexible, scalable environment, focused on users who access and manage the site via traditional computing devices like PCs and laptops.

### Software Interfaces

The Bamboo Warriors Repository of Knowledge website will interact with several software components to ensure comprehensive functionality and user experience. Key software interfaces include:

* GitHub Codespaces: Provides the cloud-based development environment where the website will be hosted and managed. Codespaces will handle code deployment, version control, and collaborative development.
* Web Browsers: Users will interact with the website through standard web browsers such as Google Chrome, Mozilla Firefox, Safari, and Microsoft Edge. The interface will be optimized for these browsers to ensure compatibility and performance.
* Content Management System (CMS): The website will utilize a CMS for managing and updating content, including articles, multimedia resources, and forum posts. The CMS will be integrated to facilitate easy content creation and management.
* Search Engine: The website will incorporate a search engine to enable users to find specific content quickly. The search engine will be integrated into the website’s interface and configured to index and retrieve content efficiently.
* User Authentication: For user profiles and forum participation, the website will integrate user authentication systems to manage login and registration processes securely.

This combination of software components will ensure that the website functions smoothly, remains up-to-date, and provides a seamless experience for users interacting via PCs or laptops.

## Functional Requirements

#### 3.2.1 F1: Content Management and Display

* **F1.1**: The system shall allow administrators to create new content, including articles, publications, and multimedia resources.
* **F1.2**: The system shall provide content editing functionality, enabling administrators to update existing content.
* **F1.3**: The system shall allow categorization and tagging of content to improve searchability and organization.
* **F1.4**: The system shall allow administrators to delete or archive content that is no longer relevant or outdated.

#### 3.2.2 F2: Search and Filtering

* **F2.1**: The system shall provide a keyword-based search function for users to locate specific content.
* **F2.2**: The system shall enable filtering by content type (articles, multimedia, forum posts) and categories (e.g., conservation techniques, bamboo species).
* **F2.3**: The system shall display search results in an organized list with relevant metadata (title, author, date).
* **F2.4**: The system shall provide sorting options for search results by relevance, date, or popularity.
* **F2.5**: The system shall provide a catalogue view for users that sets the content into specific categories such as articles, books, research papers, at videos.

#### 3.2.3 F3: User Registration

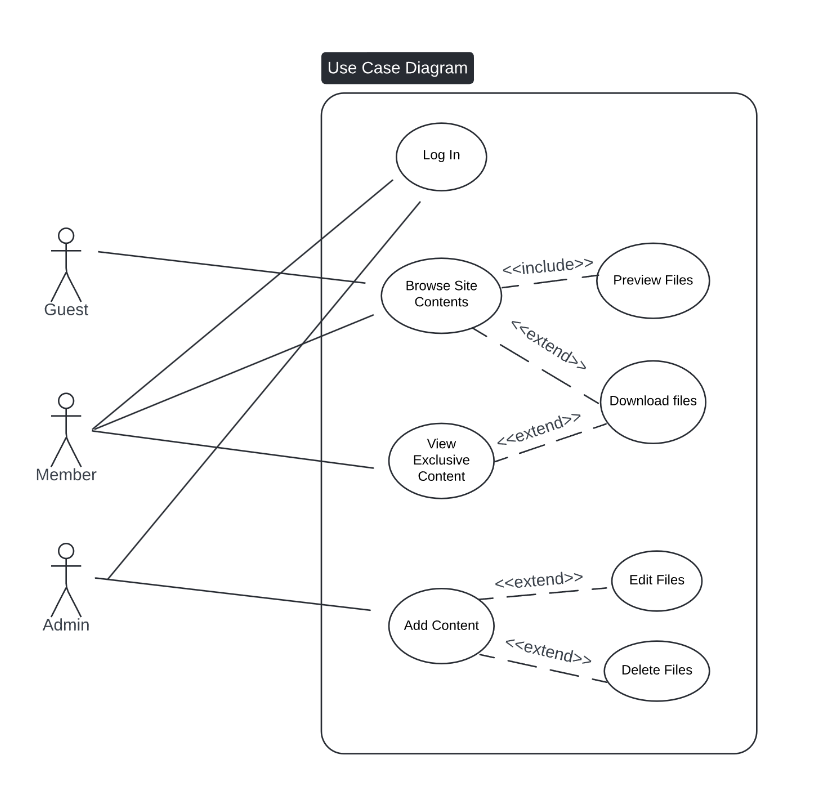
* **F3.1**: The system shall allow users to create personal accounts with basic profile information.
* **F3.2:** The system shall allow registered users to post questions, comments, or replies in discussion forums.
* **F3.3:** The system shall allow users to edit or delete their posts within a specified time frame after posting.

#### 3.2.4 F4: Multimedia Handling

* **F4.1**: The system shall support the uploading of images, videos, and audio files by administrators and authorized users.
* **F4.2**: The system shall provide a multimedia viewer that supports different media formats (JPEG, PNG, MP4, MP3).
* **F4.3**: The system shall allow users to view and interact with media content inline with articles or as standalone resources.
* **F4.4**: The system shall provide basic media management functions (e.g., renaming, deleting, and organizing multimedia).

## Use Case Model

TO DO: Provide a use case diagram that will encapsulate the entire system and all actors.



### Use Case #1 (Log In - U1)

TO DO: Provide a specification for each use case diagram

**Author –** Gabrielle Ann S. Yogawin

**Purpose** - User logs in to the site using their subscription account to access the exclusive contents.

**Requirements Traceability –**

**Priority** - High.

**Preconditions** - The user must have a registered account with valid credentials.

**Post conditions** - If successful, the user is redirected to the dashboard with access to member-exclusive content.

**Actors** – Members and Admin

**Extends –** If this is an extension use case, identify which use case(s) it extends

**Flow of Events**

* 1. Basic Flow -

1.1 - The user navigates to the login page.

1.2 - The user enters their username and password.

1.3 - If valid, the user is redirected to the dashboard.

* 1. Alternative Flow -
  2. Exceptions - Invalid credentials result in an error message and prompt the user to retry.

**Includes** -

**Notes/Issues** -

### Use Case #2 (Browse Site Contents – U2)

**Author –** Gabrielle Ann S. Yogawin

**Purpose** – The user logs in or simply browses the website to preview and download the contents.

**Requirements Traceability –**

**Priority** - High.

**Preconditions** – The website is accessible

**Post conditions** – The users successfully preview and download the contents

**Actors** – Guest, Members, and Admin

**Extends –** None

**Flow of Events**

* 1. Basic Flow -

1.1 - The user navigates to the website.

1.2 - The user browses different sections of the website.

1.3 - The user can download or preview with content based on permissions

* 1. Alternative Flow – If the guest user tries to access the exclusive content the website, the website will force the user to create an account.
  2. Exceptions -

**Includes** -

**Notes/Issues** -

### Use Case #3 (View Exclusive Contents – U3)

**Author –** Gabrielle Ann S. Yogawin

**Purpose** – Member wants to view content that can only be seen through premium subscription.

**Requirements Traceability –**

**Priority** - High.

**Preconditions** – Has bought the premium subscription on the website.

**Post conditions** – Able to access exclusive content.

**Actors** – Members

**Extends –** None

**Flow of Events**

* 1. Basic Flow -

1.1 - Member navigates through the exclusive content section.

1.2 - The website verifies the member's subscription and shows the exclusive

content.

* 1. Alternative Flow –
  2. Exceptions -

**Includes** -

**Notes/Issues** -

### Use Case #4 (Add Contents – U4)

**Author –** Gabrielle Ann S. Yogawin

**Purpose** – Admin wants to add files into the website.

**Requirements Traceability –**

**Priority** - High.

**Preconditions** – The admin needs to have the file and fill out the necessary details about the file.

**Post conditions** – The added file should be visible now to other users.

**Actors** – Admin

**Extends –** None

**Flow of Events**

* 1. Basic Flow -

1.1 - The admin clicks the add option

1.2 - The website shows the uploaded file

* 1. Alternative Flow –
  2. Exceptions -

**Includes** -

**Notes/Issues** -

# Other Non-functional Requirements

## Performance Requirements

This section outlines the performance criteria that the website must meet to ensure optimal user experience and system efficiency. It specifies how the website should perform under various conditions, including load times, search response times, user concurrency, and data update frequencies. Each performance requirement is designed to address different aspects of system responsiveness and reliability, ensuring that the website remains functional and efficient even as user demands and content volumes fluctuate.

* P1: Page Load Time  
  The website must have a page load time of no more than 3 seconds under normal conditions. This ensures a responsive user experience and maintains user engagement.
* P2: Search Response Time  
  The search engine should return results within 2 seconds for standard queries. This helps users quickly locate the content they are interested in.
* P3: Data Update Frequency  
  Content updates (e.g., news, articles) must be reflected on the website within 10 minutes of the update being made in the CMS. This ensures that users receive the most current information.

## Safety and Security Requirements

This section defines the necessary safeguards and actions to protect the website from potential risks and threats. It covers data encryption, user authentication, data backup and recovery, and regulatory compliance to ensure the security and integrity of user data and the overall safety of the system.

S1: Data Encryption

* Description: All data transmitted between the user's browser and the website must be encrypted using HTTPS with SSL/TLS. This encryption protects sensitive information from being intercepted or tampered with during transmission, ensuring the confidentiality and integrity of user data.

S2: User Authentication

* Description: To secure user accounts, the website must implement strong password protection like two-factor authentication (2FA) for forum participation. This adds an extra layer of security, making it more difficult for unauthorized individuals to gain

S3: Role-Based Access Control (RBAC)

* Description: The website will implement Role-Based Access Control to manage user permissions and ensure that access to features and data is restricted based on user roles. The RBAC system will include the following roles:
* Guests:  
  Permissions: Guests can access public content but cannot interact with forum posts or participate in discussions. They will not have access to personal user profiles or any administrative functions.
* Members:  
  Permissions: Members can access and contribute to forum discussions, post replies, and interact with other members. Members do not have access to administrative features or user management functionalities.
* Admins:  
  Permissions: Admins have full access to all features and functionalities of the website. They can manage user accounts, moderate forum content, configure site settings, and perform content updates. Admins are also responsible for overseeing security and ensuring compliance with safety policies.

## Software Quality Attributes

This section describes the key quality characteristics that the website should possess to meet user and developer expectations. It includes attributes like reliability, usability, maintainability, and adaptability, which contribute to the overall effectiveness and efficiency of the software.

4.3.1 Reliability

* Description: The website should maintain a high level of reliability, achieving 99.9% uptime. This means the site should be available and operational almost all the time, ensuring that users can access the site whenever needed without frequent outages.

4.3.2 Usability

* Description: The website must offer an intuitive and user-friendly interface. This includes easy navigation and a clear layout that allows users to interact with key features, such as the knowledge base and forums, without difficulty. Usability is crucial for providing a positive user experience and ensuring that users can efficiently find and use the information they need.

4.3.3 Maintainability

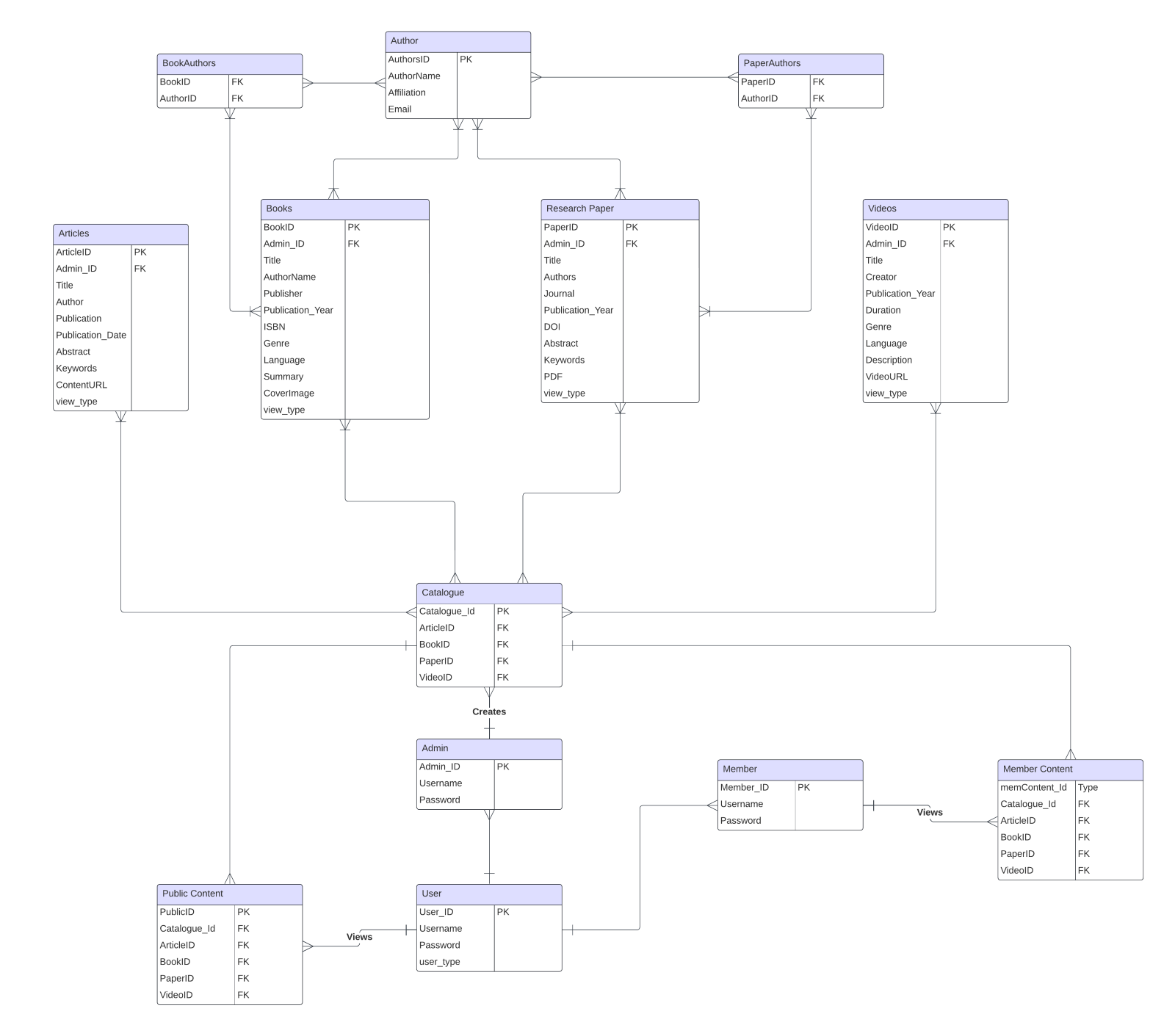
* Description: The website’s codebase should be modular and well-documented, making it easier to update and maintain. This includes clear documentation of APIs, code comments, and adherence to coding standards. Maintainability ensures that the system can be efficiently modified, debugged, and improved over time.

4.3.4 Adaptability

* Description: The website should be designed to accommodate future expansions and changes, such as new content categories or integration with additional data sources. This requires a flexible architecture that supports scalability and can easily adapt to evolving requirements.

Appendix A – Data Dictionary/ERD

Figure 1: Entity-Relationship Diagram



**DATA DICTIONARY**

**Admin**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Attribute Name** | **Contents** | **Type** | **Format** | **Range** | **Required** | **PK or FK** | **FK Reference Table** |
| Admin\_ID | Admin unique ID | CHAR(10) | 9999999999 | 10000-99999 | Yes | PK |  |
| Username | Admin username | VARCHAR(20) | XXXXXXXXX |  | Yes |  |  |
| Password | Admin password | VARCHAR(255) | Hashed |  | Yes |  |  |

**Articles**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Attribute Name** | **Contents** | **Type** | **Format** | **Range** | **Required** | **PK or FK** | **FK Reference Table** |
| ArticleID | Article unique ID | CHAR(10) | 9999999999 | 10000-99999 | Yes | PK |  |
| Admin\_ID | Admin ID | CHAR(10) | 9999999999 | 10000-99999 | Yes | FK | Admin |
| Title | Article title | VARCHAR(100) | XXXXXXXXX |  | Yes |  |  |
| Author | Article author(s) | VARCHAR(100) | XXXXXXXXX |  | Yes |  |  |
| Publication | Article publication name | VARCHAR(100) | XXXXXXXXX |  | Yes |  |  |
| Publication\_Date | Article publication date | DATE | YYYY-MM-DD |  | Yes |  |  |
| Abstract | Article abstract | TEXT | XXXXXXXXX |  | No |  |  |
| Keywords | Article keywords | VARCHAR(255) | XXXXXXXXX |  | No |  |  |
| ContentURL | Article content URL | VARCHAR(255) | XXXXXXXXX |  | No |  |  |

**Books**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Attribute Name** | **Contents** | **Type** | **Format** | **Range** | **Required** | **PK or FK** | **FK Reference Table** |
| BookID | Book unique ID | CHAR(10) | 9999999999 | 10000-99999 | Yes | PK |  |
| Admin\_ID | Admin ID | CHAR(10) | 9999999999 | 10000-99999 | Yes | FK | Admin |
| Title | Book title | VARCHAR(100) | XXXXXXXXX |  | Yes |  |  |
| AuthorName | Book author(s) | VARCHAR(100) | XXXXXXXXX |  | Yes |  |  |
| Publisher | Book publisher | VARCHAR(100) | XXXXXXXXX |  | Yes |  |  |
| Publication\_Year | Book publication year | INT | 9999 | 1900-2100 | Yes |  |  |
| ISBN | Book ISBN | CHAR(13) | 9999 | 0-9999999999999 | Yes |  |  |
| Genre | Book genre | VARCHAR(50) | XXXXXXXXX |  | No |  |  |
| Language | Book language | VARCHAR(50) | XXXXXXXXX |  | Yes |  |  |
| Summary | Book summary | TEXT | XXXXXXXXX |  | No |  |  |
| CoverImage | Book cover image | VARCHAR(255) | XXXXXXXXX |  | No |  |  |

**Videos**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Attribute Name** | **Contents** | **Type** | **Format** | **Range** | **Required** | **PK or FK** | **FK Reference Table** |
| VideoID | Video unique ID | CHAR(10) | 9999999999 | 10000-99999 | Yes | PK |  |
| Admin\_ID | Admin ID | CHAR(10) | 9999999999 | 10000-99999 | Yes | FK | Admin |
| Title | Video title | VARCHAR(100) | XXXXXXXXX |  | Yes |  |  |
| Creator | Video creator | VARCHAR(100) | XXXXXXXXX |  | Yes |  |  |
| Publication\_Year | Video publication year | INT | 9999 | 1900-2100 | Yes |  |  |
| Duration | Video duration (mins) | INT | XXXXXX | 0-10000 | Yes |  |  |
| Genre | Video genre | VARCHAR(50) | XXXXXXXXX |  | No |  |  |
| Language | Video language | VARCHAR(50) | XXXXXXXXX |  | Yes |  |  |
| Description | Video description | TEXT | XXXXXXXXX |  | No |  |  |
| VideoURL | Video URL | VARCHAR(255) | XXXXXXXXX |  | Yes |  |  |

**Research Paper**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Attribute Name** | **Contents** | **Type** | **Format** | **Range** | **Required** | **PK or FK** | **FK Reference Table** |
| PaperID | Research paper unique ID | CHAR(10) | 9999999999 | 10000-99999 | Yes | PK |  |
| Admin\_ID | Admin ID | CHAR(10) | 9999999999 | 10000-99999 | Yes | FK | Admin |
| Title | Paper title | VARCHAR(100) | XXXXXXXXX |  | Yes |  |  |
| Authors | Paper authors | VARCHAR(100) | XXXXXXXXX |  | Yes |  |  |
| Journal | Journal name | VARCHAR(100) | XXXXXXXXX |  | Yes |  |  |
| Publication\_Year | Publication year | INT | 9999 | 1900-2100 | Yes |  |  |
| DOI | Digital Object Identifier | VARCHAR(100) | XXXXXXXXX |  | Yes |  |  |
| Abstract | Paper abstract | TEXT | XXXXXXXXX |  | No |  |  |
| Keywords | Paper keywords | VARCHAR(255) | XXXXXXXXX |  | No |  |  |
| PDF | Paper PDF URL | VARCHAR(255) | XXXXXXXXX |  | Yes |  |  |

**Catalogue**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Attribute Name** | **Contents** | **Type** | **Format** | **Range** | **Required** | **PK or FK** | **FK Reference Table** |
| Catalogue\_Id | Catalogue unique ID | INT | XXXXXXXXX |  | Yes | PK |  |
| ArticleID | Foreign key from Articles | CHAR(10) | 9999999999 | 10000-99999 | No | FK | Articles |
| BookID | Foreign key from Books | CHAR(10) | 9999999999 | 10000-99999 | No | FK | Books |
| PaperID | Foreign key from Research Paper | CHAR(10) | 9999999999 | 10000-99999 | No | FK | Research Paper |
| VideoID | Foreign key from Videos | CHAR(10) | 9999999999 | 10000-99999 | No | FK | Videos |

**Public Content**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Attribute Name** | **Contents** | **Type** | **Format** | **Range** | **Required** | **PK or FK** | **FK Reference Table** |
| PublicID | Public content unique ID | INT | XXXXXXXXX |  | Yes | PK |  |
| Catalogue\_Id | Foreign key from Catalogue | INT | XXXXXXXXX |  | Yes | FK | Catalogue |
| ArticleID | Foreign key from Articles | CHAR(10) | 9999999999 | 10000-99999 | No | FK | Articles |
| BookID | Foreign key from Books | CHAR(10) | 9999999999 | 10000-99999 | No | FK | Books |
| PaperID | Foreign key from Research Paper | CHAR(10) | 9999999999 | 10000-99999 | No | FK | Research Paper |
| VideoID | Foreign key from Videos | CHAR(10) | 9999999999 | 10000-99999 | No | FK | Videos |

**Member Content**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Attribute Name** | **Contents** | **Type** | **Format** | **Range** | **Required** | **PK or FK** | **FK Reference Table** |
| memContent\_id | Member content unique ID | INT | XXXXXXXXX |  | Yes | PK |  |
| Catalogue\_Id | Foreign key from Catalogue | INT | XXXXXXXXX |  | Yes | FK | Catalogue |
| ArticleID | Foreign key from Articles | CHAR(10) | 9999999999 | 10000-99999 | No | FK | Articles |
| BookID | Foreign key from Books | CHAR(10) | 9999999999 | 10000-99999 | No | FK | Books |
| PaperID | Foreign key from Research Paper | CHAR(10) | 9999999999 | 10000-99999 | No | FK | Research Paper |
| VideoID | Foreign key from Videos | CHAR(10) | 9999999999 | 10000-99999 | No | FK | Videos |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Operation** | **Description** | **Input** | **Output** | **Requirements** |
| Create Admin | Create admin account in the system | This will include the username and password | Admin account created | Username must be unique  Passwords must be filled out |
| Create Article, Video, Book, Research Paper, and Videos | Adding an entity such as the Article in the system’s database | Relevant information such as the ISBN, Description, Creator or Author and such | Entity (this may be an article, book, etc.) will be recorded in the database | All fields will be required to be filled, this includes the title, description, author, etc. |
| Assign Visibility | Determines who can see the content of the uploaded entity such as books | Choose either public or for members only | Entity will be labeled as public or members only | Required to fill or choose type of visibility (Public or Member) |