**“Archivez: A Multiplatform Research Paper Repository Management System with Real-Time Review and Evaluation for Baco Catholic, Inc.”**

A non-thesis Project

Presented to the Faculty of the

College of Computer Studies

**MINDORO STATE UNIVERSITY**

Calapan City Campus

Masipit, Calapan City, Oriental Mindoro

In Partial Fulfillment

of the Requirements for the Course of

**APPLICATION DEVELOPMENT AND EMERGING TECHNOLOGY**

by

**Bacay, Sherwin**

**Gungon, Allysa Yvonne T.**

**Montellano, Charmel L.**

**Tolentino, Mary Mae S.**

**BSIT III-F1**

October 2023

**TABLE OF CONTENTS**

**Chapter I. Introduction**

**Project Context ……………………………………………………………………………………………… 1**

**Objectives …………………………………………………………………………………………………………… 4**

**Scope and Limitations ……………………………………………………………………………… 6**

**Definition of Terms …………………………………………………………………………………… 6**

**Chapter II. Requirements Specification**

**Hardware and Software Requirements …………………………………………… 9**

**Functional Requirements ……………………………………………………………………… 12**

**Non-Functional Requirements …………………………………………………………… 17**

Operational Requirement ………………………………………………………… **17**

Performance Requirement ………………………………………………………… **19**

Security Requirement ………………………………………………………………… **19**

**CHAPTER I**

**INTRODUCTION**

In this chapter, researchers will thoroughly examine the fundamental aspects of the project, thereby establishing a solid basis for the system that lies ahead.

**Project Context**

In the ever-evolving landscape of digital content, the need for a comprehensive and accessible repository system is increasingly vital. A digital repository is a mechanism for managing and storing digital content. Repositories can be subject or institutional in their focus. Putting content into an institutional repository enables staff and institutions to manage and preserve it, and therefore derive maximum value from it. A repository can support research, learning, and administrative processes. Repositories use open standards to ensure that the content they contain is accessible in that it can be searched and retrieved for later use. The use of these agreed international standards allows mechanisms to be set up which import, export, identify, store and retrieve the digital content within the repository. A repository system can be a shop window for stakeholders of the university. Its function is to store or catalog scholarly content such as research papers and journal articles. It can also be intended to collect and preserve the research output of universities in digital form such as theses and special problems. Furthermore, Schopfel et al. (2019) observed that in a research university, there is a need for access to scholarly works since scientific communities require access to scientific information for research and teaching activities. A thesis repository system provides several opportunities for advanced learning because it can aid students in widening their interests in research. When research outputs materials availability is limited due to inaccessibility, this can lead to wastage of knowledge in the academic research community. For instance, the quality research project output of students suddenly ends when not digitally archived and shared to future researchers in a certain institution. Additionally, the efforts and ideas poured into the research work stops when it is not adequately shared to the online community of researchers as an additional knowledge realm for future researchers to collaborate and innovate.

As the number of distinct repositories that are managed by individual HEIs increases, the complexity of finding relevant research materials/ideas from widely dispersed online repositories also increases. The chances to collaborate with a certain researcher become slimmer to nothing at all. Using the vastness of the web to obtain potential research areas from varied online digital repositories, a novice researcher can hardly locate relevant research ideas and to directly connect to other researchers with the same interests. Conversely, a readily available, publicly accessible online community of digital repositories offers immense advantages to novice researchers. It provides them with a fertile ground to browse, explore, and extract resource materials and ideas to formulate and advance their research projects with ease. Such an ecosystem not only promotes the dissemination of knowledge but also fosters collaboration, innovation, and a sense of connectedness among researchers, including high school students and alumni. By implementing a digital repository within a high school context, the aim is to empower the school and its stakeholders to efficiently manage and utilize digital resources. Such a repository serves as an invaluable resource, supporting diverse aspects of the educational ecosystem, including research, learning, and administrative processes. The repository's adoption of open standards ensures that the content it houses remains accessible, searchable, and retrievable, thus enabling its sustained use. Its primary function is to efficiently store and catalog scholarly content, such as research papers and journal articles produced by students and faculty. Moreover, it has the capacity to collect, curate, and preserve the school's research output in digital form, which includes theses, special projects, and other scholarly endeavors. In light of these considerations, the development of a multiplatform research paper repository management system for high schools is a project of immense importance, as it bridges the gap in the educational landscape by providing an inclusive, accessible, and collaborative platform for research and scholarly endeavors within high school communities.

**Objectives**

The study aims to design, develop, and implement a Multiplatform Research Repository Management System tailored to the needs of high school and senior high students. This system aims to address the challenges students face in conducting research, creating well-structured theses, and archiving research papers, while fostering a culture of academic integrity.

Specifically, this study aims to:

* Allow users, including researchers and students, to submit research papers and theses to the system.
* Provide tools for document management, including metadata entry (e.g., title, authors, keywords) and version control to track revision and updates.
* Educate users on proper research methodologies, citation practices, and plagiarism awareness through a resource library.
* Offer guidance and resources that help students and researchers navigate the intricacies of scholarly work.
* Create an environment for collaboration and knowledge sharing among users such as students, alumni, and teachers.
* Facilitate interactions between users who share common research interests, thereby enhancing the spirit of community-based learning.
* Create a user-friendly interface to simplify the submission process of research papers, making it accessible and easy to both computer literate and non-literate users.
* Ensure that the Research Repository Management System is accessible across multiple platforms, including web browsers, mobile applications, and other devices.

**Scope and Limitations**

This study is limited to the aspects and factors of creating, designing, and implementing a Multiplatform Research Repository Management System. There are three users in the system: the students, the teachers and Research Instructor/Admin. The student can view and post their research, and the teacher can comment on the posted research. The system's design emphasizes multi-platform accessibility, ensuring that users can seamlessly engage with it through web browsers, mobile applications, and various devices. Furthermore, the study includes features related to efficient document management, encompassing metadata entry, version control, and document tracking. The study does not delve deeply into the management of large volumes of data, which could present a limitation in scenarios with substantial document uploads and interactions. Understanding these scope and limitations is essential for defining the project's boundaries and focus areas.

**Definition of terms**

To enhance clarity and facilitate comprehension, the following terminology is conceptually and operationally elucidated:

**Theses and Research –** the documents that will be uploaded, stored, collected, viewed, searched, and evaluated in the researchers’ system.

**Repository –** the model that is implemented or used into the researchers’ systemthat can capture, store, index, preserve and disseminate the school’s research documents.

**Management System** – serves as the backbone for efficiently handling and facilitating the control and administration of the digital repository.

**Multiplatform –** refers to the capability of the researchers’ system to ensure that it is accessible and usable by the users who may have different types of devices and operating systems.

**Digital Content –** refers to the electronic information, documents and resources that are created, uploaded, collected, and managed within the research paper repository system.

**Alumni –** serve as mentors or collaborators in the system and provide guidance on current high school students research projects.

**Students –** are both contributors and beneficiaries of the research paper repository system. The system supports their learning, research, collaboration, and academic development.

**Scholarly Content –** refers to the academic and research materials that are created and contributed in the research paper repository system.

**Metadata –** is essential for the effective management, organization, retrieval, and understanding of the scholarly content within the research paper repository.

**Novice Researchers –** are individuals which are mainly the students, they are new to the field of research and are just beginning to explore and engage in research activities, in other words they are the primary users of the research repository management system.

**CHAPTER II**

**REQUIREMENTS SPECIFICATION**

This chapter outlines the specific requirements, features, and functionalities significant to the researchers' system development.

**Hardware Requirements**

Hardware Requirements refer to the representation of the hardware used by the system. Table 1 below presents the hardware requirements to be used by the project.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Hardware** | **Functions** | **Specifications** | | **Unit** |
| **Minimum** | **Recommended** |
| Processor (CPU) | It is responsible for executing instructions, performing calculations, managing data, and coordinating the activities of other hardware components to ensure the proper functioning of our computer system. | Intel Core i3 | Intel Core i5 or higher version | 1 |
| Memory (RAM) | A temporary memory storage for the system. It acts as a high-speed, volatile storage medium that supports the immediate needs of the management system, contributing significantly to the systems overall performance and responsiveness. | 4GB | 8GB or higher | 2 |
| Storage | Acts as the persistent memory where the system’s information is saved and contributed to the smooth operation of the system. | 128GB SSD | 256GB or higher | 3 |
| Modem | It enables remote access to the system. It is used by the developers because a stable and low-latency connection is essential to provide a fast and reliable network connection for the system. | 20 Mbps | 35 Mbps or higher | 4 |
| Android Smartphone | It used by the users to easily access the system and to test and try out the responsiveness and compatibility of the system enhancing its efficiency and flexibility. |  |  | 5 |

**Table 1: Hardware Requirements**

**Software Requirements**

Software Specifications refer to the representation of the software used by the system. Table 2 below presents the software specifications to be used by the project.

|  |  |
| --- | --- |
| **Software used** | **Description** |
|
| Windows 10 and Windows 11 | Are versions of the Windows operating system that will be use for the researcher’ system. |
| Laragon | A powerful development environment and server stack for building and managing the researchers’ system. Version 7 to 8.2 is required. |
| phpMyAdmin | The latest version (currently 5.2.1) is required to be used for supporting a wide range of operations and managing the system’s MySQL databases. |
| MySQL | Version 8.0.30 or higher is required for storing and managing the system’s structured data. |
| Visual Studio Code | A free and powerful lightweight code editor providing support to the researchers’ system by conducting development operations like debugging, task running, and version control. The required version is 1.83.1 or higher. |
| CodeIgniter 4 | An open-source PHP framework. Version 4 is the required version to develop the researchers’ system. |
| Bootstrap 5 | Are ready-made web design composed of HTML and CSS along with other optional JavaScript plugins for easy customization of the system’s UI. |
| Web browsers | Samples are Google chrome, Firefox, and Microsoft edge, etc. These are used to enable the system’s access to necessary web-based contents and manage interactions with online services and resources. |

**Table 2: Software Requirements**

**Functional Requirements**

This part enumerates the operations and features that the system must perform. Table 3 represents the descriptions of data requirements, process requirements, and output requirements.

|  |  |
| --- | --- |
| **ID NO** | **Requirement Description** |
| **FR1** | **Data Requirements** |
| **Admin/Research Coordinator Account** | |
| 1.1 | The admin/research coordinator should be able to register with their name, email, and password. |
| 1.2 | The admin/research coordinator should be able to authenticate and validate their account using the email address they provided before logging into their account securely. |
| 1.3 | The admin/research coordinator should be able to reset their password if forgotten using the registered email id that they provided during their first registration. |
| 1.4 | The admin must be able to view, add, and edit their user profiles and information. |
| 1.5 | The admin must be able to evaluate the research paper and give comments and suggestions about the research papers of the students. |
| 1.6 | The admin should be able to manage the user accounts, roles, and access control, ensuring a secure and orderly environment. |
| **Research Instructor/Teacher Account** | |
| 1.7 | The research instructor should be able to register with their name, email, and password. |
| 1.8 | The research instructor should be able to authenticate and validate their account using the email address they provided before logging into their account securely. |
| 1.9 | The research instructor should be able to reset their password if forgotten using the registered email id that they provided during their first registration. |
| 2.0 | The research instructor must be able to view, add, and edit their user profiles and information. |
| 2.1 | The research instructor must be able to write comments and give feedback about research materials submitted by the students for approval and evaluation. |
| 2.2 | The research instructor must be able to evaluate the students’ research paper by giving comments and suggestions. |
| 2.3 | The research instructor must be able to update the status of the students’ research paper, if it already had been approved and submitted to the admin for further evaluation before it is published into the system. |
| **Student Account** | |
| 2.4 | The student should be able to register with their name, email, and password. |
| 2.5 | The student should be able to authenticate and validate their account using the email address they provided before logging into their account securely. |
| 2.6 | The student should be able to reset their password if forgotten using the registered email id that they provided during their first registration. |
| 2.7 | The student must be able to view, add, and edit their user profiles and information. |
| 2.8 | The student should be able to upload their research papers and add or write the metadata after completion of upload for research items including the title, author, keywords, and publication date. |
| 2.9 | The student must be able to specify to which research instructor is in charge of the evaluation of their uploaded research paper. |
| 3.0 | The student should be able to assess the quality and relevance of research papers allowing other students to have the option to upvote recommended research papers to provide other users with personalized recommendations and influence recommendations. |
| 3.1 | The student should be able to easily add citations in various citation styles (e.g., APA, MLA, IEEE) for the papers they cite or references. |
| 3.2 | The student should be able to use a bookmark function to save, efficiently manage, and revisit the specific research papers of interest. |
| **FR2** | **Process Requirements** |
| **Admin Account** | |
| 3.3 | The system must be able to receive research papers that were approved by the research instructors. |
| 3.4 | The system must be able to post the students research paper into the system if it already has been approved. |
| 3.5 | The system should have advanced research browsing and search function should enable all the users to efficiently browse, and search documents based on the search algorithm that considers various criteria including keywords, topics, authors, and dates. |
| 3.6 | The system should be able to allow admin/research coordinator to view metadata for research items such as its title, author, keywords, and publication date. |
| **Research Instructor/Teacher Account** | |
| 3.7 | The system must be able to receive and have access to review the research papers submitted by the students. |
| 3.8 | The system can view and track the status of the students’ research paper, if it already had been approved and submitted to the admin for further evaluation before it is published into the system. |
| 3.9 | The system should have advanced research browsing and search function should enable all the users to efficiently browse, and search documents based on the search algorithm that considers various criteria including keywords, topics, authors, and dates. |
| 4.0 | The system should be able to allow research instructors to view metadata for research items such as its title, author, keywords, and publication date. |
| **Student Account** | |
| 4.1 | The system should be able to store the students research papers along with its metadata. |
| 4.2 | The system should be able to view research papers uploaded or published in the system. |
| 4.3 | The system should be able to show the status of their uploaded documents, if they already had been evaluated, approved, and submitted by the research instructor to the admin or research coordinator for further evaluation before it is published into the system. |
| 4.4 | The system should have advanced research browsing and search function should enable all the users to efficiently browse, and search documents based on the search algorithm that considers various criteria including keywords, topics, authors, and dates. |
| 4.5 | The system should be able to allow students to view metadata for research items such as its title, author, keywords, and publication date. |
| 4.6 | The system should have a version control enabled for research documents to allow tracking of document revisions and updates. |
| **FR3** | **Output Requirements** |
| **Admin Account** | |
| 4.7 | The system must provide the system’s analytics and visual representation. |
| 4.8 | The system must track students log document uploads, and engagement in the system. |
| **Research Instructor/Teacher Account** | |
| 4.9 | The system must provide the system’s analytics and visual representation. |
| 5.0 | The system must track students log document uploads, and engagement in the system. |
| **Student Account** | |
| 5.1 | The system must access and view relevant data analytics related to their activities and contributions. |

**Table 3: Functional Requirements**

**Non-Functional Requirements**

These are requirements that pertain to behavior properties that a system must have. It defines how a system is supposed to be or its system properties. It contains the following:

**Operational Requirement**

Table 4 represents the requirement description that will specify the operating environment(s) in which the system must perform and how these might change over time.

|  |  |
| --- | --- |
| **ID No.** | **Requirement Description** |
|
| 1.1 | The system should be able to operate seamlessly on various web browsers and devices, including desktops, tablets, and smartphones. |
| 1.2 | The system should be able to provide clear and user-friendly documentation or guidance to help users navigate and utilize its features effectively. |
| 1.3 | The system must log user activities, including document uploads and checks for auditing and monitoring purposes. |
| 1.4 | The system must adhere to data privacy regulations and maintain user data confidentiality. |
| 1.5 | The system must be able to enhance the user-friendliness of the system's interface, making it more accessible and appealing to a broad range of users. |
| 1.6 | The system's code and interface should comply with the web development standards and best practices to ensure consistent operation across various platforms. |

**Table 4: Operational Requirement**

**Performance Requirement**

Table 5 represents the requirement description that will emphasize the response time, capacity, and reliability of the system.

|  |  |
| --- | --- |
| **ID No.** | **Requirement Description** |
|
| 1.7 | The system’s web pages should be able to load within 2 seconds to provide rapid response times for previews, and when teachers review and comment on student research papers. |
| 1.8 | The system must be able to support at least 1000 concurrent users without significant performance degradation. |
| 1.9 | Search queries should be executed within 3 seconds, even when dealing with a large dataset of documents. |
| 2.0 | The system must maintain a 99.9% uptime percentage. |
| 2.1 | Users should be able to upload/download documents based on the internet speed and size of the file. Example is a 1MB file will be uploaded/downloaded per second if the internet speed is 5Mbps. |

**Table 5: Performance Requirement**

**Security Requirement**

Table 6 represents the requirement description that will address issues with security, such as who has access to the system's data and must have the ability to protect data from disruption or data loss.

|  |  |
| --- | --- |
| **ID No.** | **Requirement Description** |
|
| 2.2 | User accounts must be secured with strong password requirements by using regular expression and data validation. |
| 2.3 | Encryption used is Hash type which involves message authentication. Data in transit and at rest should be encrypted to protect against unauthorized access. |
| 2.4 | Role-based access control must be implemented to ensure that users have appropriate permissions and access to system features. |
| 2.5 | Data must be protected from unauthorized modifications or tampering, only the intended users may access the protected data. |
| 2.6 | The system must comply with relevant data privacy laws and regulations. |

**Table 6: Security Requirement**

**Chapter III**

**DESIGN AND DEVELOPMENT METHODOLOGIES**

**System Design**

The developers of the system crafted a thorough plan in order to develop the Archivez Research Repository System. The students can use this system for easy access to the centralized database of resources and will serve as a guide for other students who are just starting to make their own research papers. On the other hand, Research Instructors benefit from monitoring and feedback processes guiding students while also ensuring consistency in resource sharing and training materials while Research coordinators gain centralized control over data management, quality assurance, and reporting, leading to better compliance and decision-making.

**Database Design**

The design of the database for the Archivez: Research Repository Management System, is a critical aspect of application development. This extensive database covers a wide range of entities in the research environment, giving users crucial knowledge about current research groups and assisting them in making the most out of the application. Tables are defined by the database design, which includes fields that have similar data types and descriptions. This process carefully arranges, verifies, and extracts the most important information from designated data phrases. Additionally, it deftly illustrates the connections between different entities, guaranteeing a unified framework that makes it easier to store, retrieve, and organize research materials and related data.

The developers used MySQL RDBMS for the database design. With MySQL, it organizes information and facilitates the creation and management of relationships between different tables, for handling complex data structures and ensuring data coherence across various parts of the Archivez: Research Repository Management System’s database.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Field Name** | **Data Type** | **Size** | **Default** | **Description** |
| id | int |  | Auto\_Increment | User ID (PK) |
| fullname | varchar | 500 | Not Null | User Fullname |
| idnumber | varchar | 255 | Not Null | User Id Number |
| department | varchar | 255 | Not Null | User Department |
| gradelevel | varchar | 255 | Null | User Grade Level |
| section | varchar | 255 | Null | User Section |
| usertype | varchar | 500 | Not Null | User Usertype |
| email | varchar | 500 | Not Null | User Email |
| password | varchar | 500 | Not Null | User Password |
| created\_at | datetime |  | Current\_Timestamp | Time Created |
| status | varchar | 255 | Not Null | User Status |
| uniid | varchar | 255 | Not Null | User Unique Identifier |
| activation\_date | datetime |  | Not Null | Activation Date |

**Table 7: Fields for users**

Table 7 above contains the field name, data type, size, default, and description of the field in the Users table. Here, the id is the Primary Key (PK).

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Field Name** | **Data Type** | **Size** | **Default** | **Description** |
| id | int |  | Auto\_Increment | Research ID (PK) |
| researchtitle | text |  | Not Null | Research Title |
| submittedto | varchar | 255 | Not Null | Research Instructor |
| subject | varchar | 255 | Not Null | Research Subject |
| author | varchar | 255 | Not Null | Research Author |
| idnumber | varchar | 255 | Not Null | Research Id Number |
| gradelevel | varchar | 255 | Not Null | Student Grade Level |
| section | varchar | 255 | Not Null | Student Section |
| uploaddate | datetime |  | Not Null | Research Upload date |
| abstract | text |  | Current\_Timestamp | Research Abstract |
| keywords | text |  | Not Null | Research Keywords |
| citation | text |  | Not Null | Research Citation |
| status | varchar | 255 | Not Null | Research Status |
| file | varchar | 255 | Not Null | Research File |
| user\_id | int |  | Not Null | Research User Id |

**Table 8: Fields for research**

Table 8 above contains the field name, data type, size, default, and description of the field in the Research table. Here, the id is the Primary Key (PK).

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Field Name** | **Data Type** | **Size** | **Default** | **Description** |
| id | int |  | Auto\_Increment | Comment ID (PK) |
| commentedby | text |  | Not Null | Commented by |
| comment | text | 255 | Not Null | Comment |
| research\_id | int | 255 | Not Null | Research Id |

**Table 9: Fields for comments**

Table 9 above contains the field name, data type, size, default, and description of the field in the Comments table. Here, the id is the Primary Key (PK).

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Field Name** | **Data Type** | **Size** | **Default** | **Description** |
| bookmark\_id | int |  | Auto\_Increment | Bookmark ID (PK) |
| researchtitle | text |  | Not Null | Research Title |
| submittedto | text |  | Not Null | Research Instructor |
| subject | text |  | Not Null | Research Subject |
| author | text |  | Not Null | Research Author |
| idnumber | text |  | Not Null | Research Id Number |
| gradelevel | text |  | Not Null | Student Grade Level |
| section | text |  | Not Null | Student Section |
| uploaddate | datetime |  | Not Null | Research Upload Date |
| abstract | text |  | Current\_Timestamp | Research Abstract |
| keywords | text |  | Not Null | Research Keywords |
| citation | text |  | Not Null | Research Citation |
| status | text |  | Not Null | Research Status |
| file | varchar | 255 | Not Null | Research File |
| user\_id | int |  | Not Null | User Id |

**Table 10: Fields for bookmark**

Table 10 above contains the field name, data type, size, default, and description of the field in the Bookmark table. Here, the bookmark\_id is the Primary Key (PK).

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Field Name** | **Data Type** | **Size** | **Default** | **Description** |
| archive\_id | int |  | Auto\_Increment | Research Id (PK) |
| researchtitle | text |  | Not Null | Research Title |
| submittedto | text |  | Not Null | Research Instructor |
| subject | text |  | Not Null | Research Subject |
| author | text |  | Not Null | Research Author |
| idnumber | text |  | Not Null | Research Id Number |
| gradelevel | text |  | Not Null | Student Grade Level |
| section | text |  | Not Null | Student Section |
| uploaddate | date |  | Not Null | Research Upload Date |
| abstract | text |  | Not Null | Research Abstract |
| keywords | text |  | Not Null | Research Keywords |
| file | varchar | 255 | Not Null | Research File |
| citation | text |  | Not Null | Research Citation |
| status | text |  | Not Null | Research Status |
| user\_id | int |  | Not Null | User Id |

**Table 11: Fields for archives**

Table 11 above contains the field name, data type, size, default, and description of the field in the Archives table. Here, the id is the Primary Key (PK).

**A diagram of a research process

Description automatically generatedArchitectural Diagram/ Block Diagram**

**Figure 1. System Architecture of Archivez Website**

Figure 1 shows the system architecture of the development of the Archivez Application. It illustrates the flow or process of how the system works. The diagram shows that the internet is necessary in order to access the website then the student can register and login into the system, afterwards the student can upload a research paper and submit it to the research instructor for evaluation and then proceed to the admin for the last checking of the research paper before it is published into the system.

**Data Flow Diagram Level 0**

This section shows the Data Flow Diagram Level 0 which is commonly known as an exploded view of the context diagram that shows the detailed process of how A diagram of a company

Description automatically generatedthe system works.

**Figure 2. Data Flow Diagram Level 0**

Figure 2 shows diagram 0 of the system. The student must register first and must verify their email before logging in securely. Then the student can now view and upload their research paper and submit it for evaluation. While the Admin and Instructor can evaluate the students research paper by giving comments and suggestions, but if approved the admin can directly post it in the system for other students to view and use as a guide.

**UML Use-case Diagram**

This section presents a use case diagram that provides a concise overview of the system's fundamental operations. The diagram visually illustrates the relationships among the system, admin/research coordinator, research instructors/teachers, and students. It serves as a graphical representation to elucidate how the system operates in the interactions among these actors. The diagram is a valuable tool for researchers, aiding in the identification and organization of the project's overall functionality.

**A diagram of a diagram

Description automatically generated**

**Figure 3. UML Use-case Diagram**

Figure 3 illustrates the roles and functions of the Administrator, Research Instructor, and Student throughout the entire system process.

**Sample Mock-up**

A screenshot of a web page

Description automatically generatedA sample mock-up serves as a visual representation of a completed website, depicting its appearance and functionality. This tool is employed to further develop the design, pinpoint potential issues, and verify that the system aligns with the user's requirements and expectations. The interfaces for the system's students, research instructors/teachers, and admin/research coordinator are presented below.

**Figure 4. Student Interface**

**A screenshot of a computer

Description automatically generatedA graph on a computer screen

Description automatically generatedFigure 5. Instructor Interface**

**Figure 6. Admin Interface**

**Development Method**

The development of the "Archivez: A Multiplatform Research Paper Repository Management System with Real-Time Review and Evaluation for Baco Catholic, Inc." will adhere to the System Development Life Cycle (SDLC) to ensure the development of an effective and functioning system. Through the implementation of an Iterative and Incremental Model, this project allows continuous improvements throughout the development journey. The SDLC encompasses four phases: requirement collection, design and development, testing, and implementation. These sequential steps will be meticulously **A diagram of a software development process

Description automatically generated**followed to ensure the successful creation of the system.

**Figure 6. SDLC Iterative and Incremental Model**

The research followed the following phase:

1. **Planning.** During this phase, researchers conduct interviews and observations to address the identified problem. They establish the project's objectives and plans, forming the groundwork for generating the desired system output by the end of the study.
2. **Requirements Gathering.** Researchers, in this phase, collect additional information and data requirements essential for system development. They define both the functional and non-functional requirements of the project.
3. **Design.** This phase involves the initiation of hardware and software design based on user requirements. Researchers meticulously plan and implement a trial-and-error approach to ensure the user interface aligns with the desired output.
4. **Development.** Researchers begin the coding process for the system's functionality in this phase, utilizing the Visual Studio Text Editor Application, PhpMyAdmin as the database administrator tool, with CodeIgniter 4 as the framework for the proposed system.
5. **Testing.** During this phase, researchers execute pre-deployment testing to ensure the system functions correctly, employing trial and error to identify and address any issues.
6. **Implementation.** Following the testing phase, researchers create an iteration of the project to analyze and enhance design and functionality, making significant improvements to achieve the project’s objectives.
7. **Maintenance.** During the maintenance phase, researchers uphold the system's functionality during deployment to sustain its performance. They also identify and rectify errors that may arise over time.

**Gantt Chart**

This section features a Gantt Chart illustrating the project's plans and schedules throughout its timeline, encompassing all development stages leading to project completion. The chart serves as a comprehensive documentation of deadlines for task completion, providing researchers with a visual representation of milestones and progress in various project tasks.

**Table 11. Gantt Chart (Group)**

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Task Name** | **Task Date** | | | | | | | | | | | | | | | | | | | |
| Oct | | | | Nov | | | | Dec | | | | Jan | | | | Feb | | | |
| Week 1 | Week 2 | Week 3 | Week 4 | Week 1 | Week 2 | Week 3 | Week 4 | Week 1 | Week 2 | Week 3 | Week 4 | Week 1 | Week 2 | Week 3 | Week 4 | Week 1 | Week 2 | Week 3 | Week 4 |
| **1.Planning** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 1.1 Conduct an interview |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 1.2 Define project objectives |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 1.3 Define project plan |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 1.4 Approval of project plan |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| **2.Requirements Gathering** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 2.1 Data Collection |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 2.2 Functional |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 2.3 Non-Functional |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| **3.Design** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 3.1 Frontend software design |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| **4.Development** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 4.1 Back-end coding |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| **5.Testing** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 5.1 Functionality testing |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 5.2 User interface testing |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| **6.Implementation** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| **7.Maintenance** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 7.1 Project monitoring |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 7.2 Resolve system errors |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |

**Legend:** - Completed/ Done

Table 11 displays the complete process of the development of Archivez, showcasing diverse tasks with corresponding completion dates. This transparent representation ensures that researchers stay updated on the development progress, aiding them in avoiding oversights in steps and discerning the duration required for each task's completion.

**Testing and Evaluation**

This section pertains to the testing and evaluation stage within the software development process. In this phase, developers undertake comprehensive tests on the system to assess its capabilities and limitations. The objective is to proactively identify any issues or potential problems before entering the production and deployment stages. The testing encompasses all the requirements outlined in the Requirements Phase, covering aspects like design, performance, and supportability. The results of these tests undergo evaluation to gauge the system's progress and confirm its alignment with the project requirements. The developers took into consideration the following factors:

1. **Unit Testing-** This is a form of software testing that focuses on evaluating individual units or components within a software system. Performed by the developers due to the necessity for in-depth knowledge of the system's internal structure, this testing method aims to verify the proper functioning of individual functions, modules, and features. The goal is to ensure that each component behaves exactly as intended.
2. **Component Testing-** Entails the examination of individual components within the system. It is essential to meticulously evaluate the individual components of the framework. This involves testing controllers for proper handling of actions, input validation, and routing configuration. Models should undergo thorough testing for accurate database interactions and data validation, while views need to be assessed for correct template rendering and data binding. This approach aims to uncover defects or bugs specific to each component before their integration into the broader system. Additionally, it assists in identifying unforeseen interactions or dependencies between components that may lead to issues or errors in the overall system.
3. **System testing-** Is a type of software testing that evaluates the entire application or system as well as each of its component parts to verify that each module operates properly, and that data is transferred between them accurately. The purpose of this testing phase is to confirm that the system meets its requirements, performs as expected, and works flawlessly in the environment that it has been assigned. System testing, which comes after unit and integration testing, offers a thorough assessment of the complete system, making sure its overall performance and preciseness is working properly.
4. **Performance testing-** is a crucial software testing methodology employed to evaluate and validate the efficiency, responsiveness, and scalability of a system under different conditions. By conducting performance testing, organizations can identify and address performance bottlenecks, ensure that the system meets specified performance criteria, and ultimately deliver a reliable and high-performing software application. This testing approach is essential for preventing performance-related issues, enhancing user experience, and optimizing the overall performance of software systems in real-world scenarios.

**Chapter IV**

**DEVELOPMENT, TESTING, AND EVALUATION**

**Presentation of the System Output**

**Chapter V**

**CONCLUSION AND RECOMMENDATION**

**Conclusion**

**Recommendation**

**REFERENCES**

Satina, Armacheska. (2018). Design and Development of an Online Repository System for Thesis and Special Problem Manuscripts.

Sobejana, Noel & Lalisan, Sarah. (2019). Research and Capstone Project Electronic Repository. Current Journal of Applied Science and Technology. 38. 1-12. 10.9734/cjast/2019/v38i430378.

Bajrami, Enes & Rufati, Eip & Idrizi, Florim & Memeti, Ermira & Rushani, Lëndina. (2021). E-THESIS REPOSITORY SYSTEM.

Rahman, Atta & Al-Haidari, Fahd. (2019). The Digital Library and the Archiving System for Educational Institutes. Pakistan Journal of Library and Information Science. 20. 94-117. 10.47657/2018201453.

Alpasan, Bella. (2022). Theses and Dissertations Abstract E-Archiving System with SMS Support. International Journal of Multidisciplinary Research and Analysis. 05. 10.47191/ijmra/v5-i1-03.