

**DocuDB: Automated tool for COPC of Lapu-Lapu City College**

A Project Study  
Presented to  
the College of Technology  
LAPU-LAPU CITY COLLEGE  
Don B. Benedicto Road, Gun-ob, Lapu-Lapu City

In Partial Fulfillment  
of the Requirements for the Degree of  
**BACHELOR OF SCIENCE IN INDUSTRIAL TECHNOLOGY**  
**MAJOR IN COMPUTER TECHNOLOGY**

**Abondiente, Michael Emmanuel**

**Taghoy, Jhunde Kyle**

**Aparta, Van Clark**

**Monteclar, Allen Jhon**

**Bartiana, Jomilo Andrew**

## TABLE OF CONTENTS

	PAGE
TITLE PAGE	i
TABLE OF CONTENTS	ii
LIST OF TABLES	iv
LIST OF FIGURES	vi
CHAPTER	
1 <b>THE PROBLEM AND ITS SCOPE</b>	1
<b>RATIONALE OF THE STUDY</b>	1
<b>THE PROBLEM</b>	4
Statement of the Problem	4
Objectives of the Study	5
Significance of the Study	6
Scope and Limitations of the Study	8
THEORETICAL BACKGROUND	10
Software Development Project Framework Model	11
REVIEW OF RELATED STUDIES AND SYSTEMS	15
Related Studies	15
Related Systems	17
Comparative Matrix	19
RESEARCH METHODOLOGY	20

Design	20
Environment	21
Respondents	23
Sampling	24
Instruments	24
Data Gathering Procedures	25
Statistical Treatment of Data	28
Scoring Procedures	29
<b>DEFINITION OF TERMS</b>	31
<b>2 PRESENTATION, DATA, ANALYSIS AND INTERPRETATION</b>	36
<b>DOCUDB SYSTEM TECHNICAL REQUIREMENTS</b>	36
Hardware and Software Requirements	36
System Architecture	38
System Workflow	39
Product Backlog	40
System Program Flow	44
<b>DEPLOYMENT PLAN</b>	48
<b>USER GUIDE</b>	53
<b>BIBLIOGRAPHY</b>	64
Release Notes	66
<b>CURRICULUM VITAE</b>	70
Survey Questionnaire	76



## **LIST OF TABLES**

Table	Contents	Page
1	Comparative Matrix	19
2	Likert Scale for System Effectiveness and Overall Level of Acceptability	29
3	Likert Scale for System usability	30
4	Likert Scale for system functionality	30
5	Hardware and Software Requirements	36
6	Product Backlog	40

## **LIST OF FIGURES**

Figures	Contents	Page
1	Iterative Methodology Framework	11
2	Flow of the Study	20
3	Research Environment	22
4	System Architecture	38
5	Admin Workflow	45
6	user Workflow	47

## THE PROBLEM AND IT'S SCOPE

### Rationale of the Study

In today's digital era, the management of institutional files has become a vital concern in higher education. Colleges and universities deal with a wide range of documents daily, such as academic records, administrative files, and official communications. At Lapu-Lapu City College, specifically within the Certificate of Program Compliance (COPC), the manual handling of files often results in inefficiencies, delays, and risks of data misplacement. Traditional methods such as paper-based filing or fragmented digital storage systems fail to meet the demands of accuracy, security, and accessibility required in academic institutions. These issues present a pressing problem: how can colleges adopt a secure and efficient digital document management system that is both user-friendly and tailored to their specific context?

While various document management systems (DMS) are available, many are either too complex, costly, or lack the flexibility to adapt to the unique needs of academic institutions. Previous studies emphasize that while DMS platforms significantly improve productivity and collaboration, there remains a gap when it comes to localized solutions that incorporate institutional requirements such as domain-specific login systems, role-based permissions, and audit logs (Alalwan et al., 2018; Patil & Sangle, 2020). In the Philippine context, few studies have documented customized systems for higher education, leaving a knowledge gap in the literature. Thus, the development of DocuDB, an automated document management system specifically designed for COPC, seeks to fill this gap by providing a secure and efficient platform for transferring, storing, organizing, and retrieving documents.

The features of DocuDB directly address the shortcomings of existing systems. Its file management tools such as uploading, organizing, previewing, downloading, and sharing streamline the everyday workflow of faculty and staff. Content extraction and search functions allow users to retrieve files quickly, improving accessibility and reducing the time spent locating documents. Moreover, security features such as role-based access control, file and folder permissions, and audit logs ensure that confidential data remains protected and that accountability is maintained. Additional tools, including user management, activity monitoring, version history, and automated backups, further enhance system reliability and continuity. By incorporating comments, annotations, and mobile-friendly access, the system also fosters collaboration and flexibility, which are essential in modern academic environments.

The significance of this study is rooted in its practical and theoretical contributions. Practically, the system responds to the urgent need for digital transformation in educational institutions, helping colleges improve governance, streamline administrative processes, and strengthen data security. Theoretically, this study contributes to the literature on educational technology by demonstrating how customized, context-driven systems can address specific institutional challenges, in contrast to generalized DMS solutions. Research on digital learning platforms and cloud-based management tools confirms that properly designed systems enhance organizational performance and learning outcomes (Chong et al., 2020). However, little is known about how these solutions can be localized in academic settings in the Philippines, making this study both timely and necessary.

The novelty of this research lies in its integration of widely used database technology (MongoDB) with institution-specific features designed for Lapu-Lapu City College. Unlike generic systems, DocuDB was developed with consideration of the

college's unique workflows, data privacy requirements, and user environment. The timing of this study is particularly important given the rapid acceleration of digital transformation in education, influenced by global shifts such as the COVID-19 pandemic, which highlighted the urgency of robust digital infrastructures (Bao, 2020). By addressing these current needs, the research not only provides a solution for COPC but also offers a model that can be replicated or adapted by other institutions seeking to transition into efficient and secure digital practices.

In a broader context, this study aligns with national and global efforts to modernize education through digital innovation. The Commission on Higher Education (CHED) in the Philippines has emphasized the importance of integrating technology in institutional management to promote efficiency and resilience (CHED, 2020). Globally, the move toward smart and digital campuses underscores the relevance of building tailored systems that enhance academic operations. Through its unique approach and user-centered design, DocuDB demonstrates how technology can be harnessed not only to solve local problems but also to contribute to the larger goal of digital transformation in education.

## **Statement of the Problem**

This study aims to evaluate the effectiveness, usability, and reliability of **DocuDB**, an automated document management system designed to address the challenges of storing, organizing, and securing institutional files at Lapu-Lapu City College (LLCC), particularly within the Certificate of Program Compliance (COPC). Specifically, this research seeks to answer the following questions:

1. To what extent do DocuDB's core features, such as **file management, content extraction, and search tools**, meet the functional requirements and needs of its users?
2. How **user-friendly and efficient** is the navigation experience provided by DocuDB's interface across both desktop and mobile platforms?
3. To what extent does DocuDB improve **document security and accountability** through its role-based access control, file and folder permissions, and audit log features?
4. How effective is DocuDB in supporting **collaboration and accessibility** through functions such as comments, annotations, file sharing, and version history?
5. Based on the findings of this study, what recommendations can be proposed to further enhance DocuDB in addressing institutional challenges, strengthening digital governance, and meeting the specific needs of Lapu-Lapu City College (LLCC)?

## **Objectives of the Study**

The main objective of this study is to evaluate the effectiveness, usability, and functionality of **DocuDB**, an automated document management system developed for the Certificate of Program Compliance (COPC) at Lapu-Lapu City College. Specifically, this study aims to:

1. **Assess Core Features** – Determine whether DocuDB's core features, such as file uploading, storage, organization, preview, downloading, and sharing, meet the functional requirements and needs of its users.
2. **Evaluate User Experience** – Analyze the user-friendliness, accessibility, and efficiency of the navigation experience provided by DocuDB's interface across both desktop and mobile platforms.
3. **Examine Search and Accessibility** – Assess the extent to which DocuDB enhances file retrieval and accessibility through its content extraction and search tools.
4. **Analyze Security Functions** – Measure the effectiveness of DocuDB's security features, including role-based access control, file and folder permissions, and activity logs, in protecting institutional documents and ensuring accountability.
5. **Support Collaboration and Continuity** – Evaluate how DocuDB promotes collaboration and information continuity through features such as comments, annotations, version history, and automated backups.
6. **Provide Recommendations** – Based on the findings of this study, propose recommendations to further improve DocuDB in addressing institutional challenges, enhancing digital governance, and meeting the specific needs of Lapu-Lapu City College.

## **Significance of the Study**

The development of DocuDB, an automated document management system, is significant in addressing the common problems of disorganized storage, lack of accessibility, and security risks faced in handling institutional files at Lapu-Lapu City College. By introducing a centralized and secure platform, this study contributes to the digital transformation of the Certificate of Program Compliance (COPC). It is expected to benefit the following groups.

### **Institutions and Colleges**

DocuDB provides a structured platform for file management by offering tools for uploading, organizing, previewing, downloading, and sharing documents. With its search and text extraction functions, the system helps reduce delays in retrieving important files. This improves efficiency in administrative and academic operations, allowing colleges to better safeguard institutional records while ensuring accessibility to authorized users.

### **Administrators**

Through features such as role-based access control, file and folder permissions, and audit logs, administrators gain greater control and accountability over the management of sensitive files. Tools for user management, activity monitoring, and version history further strengthen governance, ensuring that document handling is both secure and transparent.

### **Faculty and Staff**

DocuDB assists faculty and staff in performing their tasks more effectively by providing quick access to documents, reducing the risk of misplaced files, and enabling

secure sharing of materials. Collaboration features, such as comments and annotations, allow users to work together seamlessly, while automated backups ensure continuity of information even during technical disruptions.

### **Researchers and Developers**

This study offers practical insights into designing automated document management systems that balance security, usability, and collaboration. By evaluating DocuDB's effectiveness, researchers and developers can use its features as a reference for building or improving similar systems tailored to the needs of academic institutions.

### **Future Researchers**

Future researchers may build on this study by exploring additional features that can enhance document management systems, such as artificial intelligence for predictive search, advanced analytics for usage monitoring, or integration with other academic platforms. In this way, DocuDB serves as both a practical solution for COPC and a foundation for further studies in educational technology and digital governance.

### **Academic Institutions**

This study contributes to academic discussions on strengthening institutional productivity and governance through digital innovation. By presenting DocuDB as a practical example of an automated document management system, the research highlights how file organization, secure storage, and role-based access can directly support the academic and administrative operations of colleges.

Bringing up rear, this study addresses the challenges of file disorganization, accessibility, and data security by introducing DocuDB, a structured and role-based

solution. It benefits multiple stakeholders by streamlining workflows, fostering collaboration, and safeguarding institutional records, while also serving as a foundation for future research and system development in educational technology.

## **Scope of the Study**

This study focuses on the design, development, and evaluation of **DocuDB**, a document management system created for Lapu-Lapu City College. The system is designed to provide a secure and centralized platform for uploading, storing, organizing, and retrieving academic and institutional files.

Key features of the system include:

**User Authentication** – A login system limited to the Lapu-Lapu City College domain to ensure secure access.

**File Management** – Uploading, storing, previewing, downloading, sharing, and organizing documents within folders.

**Content Extraction and Search** – A text extraction feature that allows users to easily search and locate files.

**Security and Permissions** – Role-Based Access Control, file and folder permissions, and audit logs for secure and organized access to data.

**Admin Controls** – Tools for user management, activity monitoring, and version history for efficient system administration.

**Additional Features** – Comments and annotations for collaboration, mobile-friendly accessibility, and automated backups for data protection.

## **Limitations of the Study**

While DocuDB provides a functional and secure document management system for Lapu-Lapu City College, this study is subject to the following limitations:

**Restricted Access** – The login system is limited only to users with the Lapu-Lapu City College domain, which means it cannot be used by external organizations or individuals.

**Limited Integration** – The system does not currently integrate with third-party tools such as Google Drive, Microsoft Office, or other cloud-based platforms.

**Scalability** – The system is designed primarily for small to medium-sized groups and may face performance issues if used for large-scale institutional storage.

**Basic Analytics** – While audit logs and version history are available, the system does not yet provide advanced reporting or analytics for document usage and performance tracking.

**Offline Availability** – The platform requires an internet connection to access and manage files, with no offline mode supported.

**Mobile Limitations** – Although mobile-friendly, certain features such as file preview and annotations may not function as smoothly on smaller devices compared to desktop use.

**Customization** – The system has fixed roles and permissions and does not yet allow extensive customization based on unique departmental requirements.

## Theoretical Background

### Software Development Project Framework Model

The proponents adopted the **Iterative Methodology** as the software development framework for the DocuDB project to ensure a systematic, flexible, and user-centered approach to system development. This model breaks down the project into manageable cycles or iterations, where each cycle involves planning, designing, developing, testing, and refining specific system components. By doing so, it allows early detection of issues and continuous integration of user feedback.

This methodology supports progressive enhancement of the system, ensuring that each version of DocuDB becomes more aligned with the actual needs of its users, particularly the faculty and staff of Lapu-Lapu City College. It allows for regular evaluation and adjustments based on stakeholder input, making the development process more responsive and adaptive to changes.

The Iterative Methodology promotes collaboration between the development team and end-users throughout the project. It also enhances project manageability by reducing risks, improving functionality over time, and ensuring that the final product is both reliable and contextually relevant. Through this approach, the proponents aim to develop a robust, scalable, and efficient document management system tailored to the operational needs of the Certificate of Program Compliance (COPC).

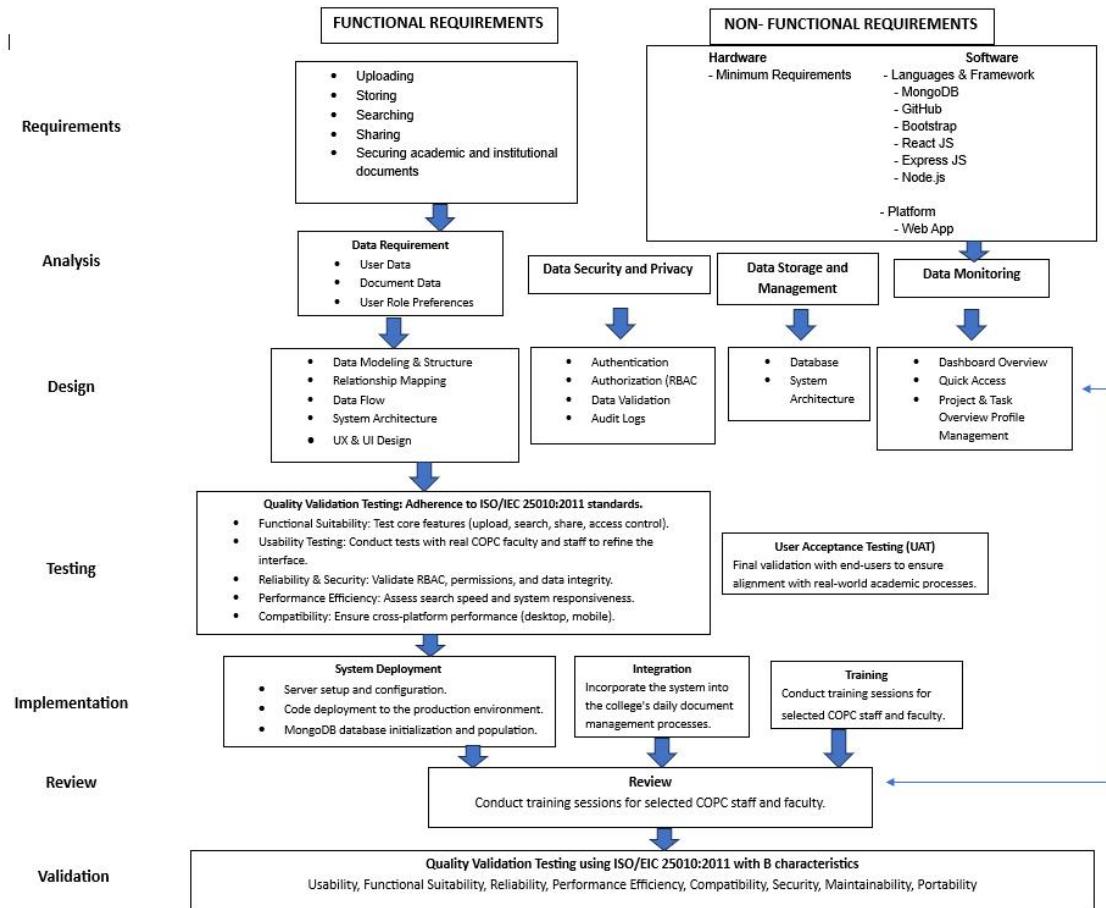


Figure 1: Iterative Methodology Framework

## Iterative Method

## Methodology Framework

Figure 1 illustrates the development methodology adopted for the DocuDB system, characterized by a structured, iterative lifecycle. This cyclical approach facilitates progressive enhancement through each phase, enabling the integration of stakeholder feedback and ensuring the evolving system remains congruent with its specified objectives.

## **Requirement phase**

In this phase, the team identified both functional and non-functional requirements based on consultations with faculty, administrative staff, and IT personnel. Methods such as interviews, workflow observations, and document analysis were used to understand the college's document handling challenges. Functional requirements included features like uploading, storing, searching, sharing, and securing academic and institutional documents. Non-functional requirements covered technical specifications such as system responsiveness, platform compatibility (web-based), and performance expectations. The outcome was a detailed requirements document that served as the foundation for system design and development.

---

## **Analysis Phase**

After gathering requirements, the team analyzed and refined them into detailed system specifications. This included defining data types the system would process, such as user accounts, document metadata, and audit logs. The phase also involved outlining the user experience (UX) and user interface (UI) design principles to ensure usability across devices. Security protocols like domain-based login and Role-Based Access Control (RBAC) were incorporated to meet institutional data privacy standards. MongoDB was selected as the data storage solution due to its scalability and flexibility. Outputs from this phase included initial data flow diagrams, wireframes, and a complete set of technical specifications.

---

## **Design Phase**

Building on the analysis phase, the design phase focused on creating a blueprint for DocuDB's architecture and user experience. Data models were developed to represent the system's document structure, user roles, and access levels, visualized using Entity-Relationship Diagrams (ERDs). UI prototypes were created to demonstrate navigation flow, layout, and functionality. Security mechanisms, including login authentication, file permissions, and activity logs, were finalized to ensure document integrity and user accountability. The system was designed following a client-server architecture to support performance, scalability, and multi-user access. Design deliverables included finalized wireframes, ERDs, dashboard mockups, and system architecture documents.

---

## **Testing Phase**

The testing phase validated that DocuDB met all functional and performance requirements. It was tested against **ISO/IEC 25010:2011** software quality standards, evaluating usability, functional suitability, reliability, and security. Functional testing ensured that core features like document uploading, previewing, sharing, and access control worked correctly. Usability testing involved real COPC faculty and staff, whose feedback helped refine the interface and workflow. After technical validations, user acceptance testing was conducted to verify system

alignment with real-world academic processes. The outcome was a validated, stable, and user-approved system ready for deployment.

---

## **Implementation Phase**

Following successful testing, DocuDB was deployed in the COPC environment. This phase included server setup, application deployment, and MongoDB database initialization. Domain-restricted access was configured to ensure only authorized Lapu-Lapu City College users could access the system. The implementation also involved integrating the system into the college's daily document management processes, training selected staff, and ensuring smooth transition from manual to digital workflows. The result was a fully operational system, optimized for secure and efficient use by faculty and administrators.

---

## **Review Phase**

Post-deployment, a review phase was conducted to gather real-user feedback and evaluate system performance. Faculty and staff reported on ease of use, document retrieval speed, and overall satisfaction. Minor usability issues and enhancement suggestions such as improved mobile responsiveness and search filtering were identified. These insights informed the next set of refinements to ensure that DocuDB continues to evolve in line with the college's digital transformation goals. The output was a review report, including user feedback, system performance evaluation, and recommendations for future updates.

## **Review of Related Studies and System**

In today's digital age, efficient document and folder management plays a vital role in ensuring productivity, collaboration, and accessibility. Traditional methods and many existing systems face limitations in scalability, searchability, and user-based customization. This chapter reviews related studies and systems that provide the foundation for the development of the DocuDB Document Management System.

### **Related Studies**

#### **Challenges in Traditional Document Management**

Traditional methods of document storage, such as physical filing cabinets or even basic digital file systems, often suffer from poor organization, limited accessibility, and difficulties in tracking revisions. According to Raju & Singh (2020), organizations relying on traditional storage encounter problems with duplication, version control, and user access, which hinder efficiency and collaboration. Similarly, Smith (2021) highlights that without a centralized, searchable repository, employees spend significant time searching for documents, reducing overall productivity.

#### **Limitations of Existing Document Management Tools**

While cloud-based platforms like Google Drive, Dropbox, and OneDrive have addressed many issues, research shows they are not without drawbacks. According to Torres et al. (2021), these platforms can be expensive for SMEs, lack customized search features tailored to specific users, and may not provide sufficient metadata filtering options. Moreover, studies by Patel (2022) emphasize that many existing tools

do not adequately separate user-specific content from shared repositories, leading to clutter and reduced efficiency.

### **Role of Search and Metadata in Document Systems**

Efficient search functionalities are key in document management. Studies by Lin & Zhao (2020) demonstrate that metadata-driven search (by filename, type, or date) significantly reduces retrieval time and enhances productivity. However, existing systems often provide limited or generic search capabilities that fail to adapt to user-specific contexts. This highlights the need for systems like DocuDB that integrate user-based search filters and advanced metadata queries.

### **Text Extraction from Documents**

A growing trend in document management is the integration of text extraction technologies, enabling content within documents to be indexed and searched. According to Chen et al. (2022), text extraction not only improves search accuracy but also allows for semantic analysis, tagging, and categorization. This is particularly valuable in academic and organizational contexts where information retrieval goes beyond filenames and metadata. DocuDB integrates this functionality to enhance the precision of search and retrieval.

### **Real-Time Monitoring and User-Based Filtering**

Real-time features ensure users can instantly access their uploaded content without delays. Ekstedt (2019) noted that real-time updates and notifications reduce bottlenecks and increase accountability. In systems where multiple users interact, user-based filtering ensures data privacy and relevance by allowing individuals to see only their own documents and folders.

Thus, literature emphasizes the importance of combining affordability, usability, user-specific search, and advanced text extraction in modern document management systems. DocuDB is designed to address these gaps with an integrated, user-friendly approach.

## **Related Systems**

To better understand the context of the Docudb System, it is essential to evaluate existing task management tools that have gained popularity for their features and functionalities. Each system has strengths tailored to different organizational needs but also presents limitations, especially for Small and Medium-sized Enterprises (SMEs).

### **Traditional Document Storage**

Traditional storage systems include physical filing, local hard drive storage, and spreadsheets for indexing files. While these methods are cost-effective and simple, they lack scalability and real-time retrieval. According to Nabeel (2024), manual systems hinder collaboration, introduce errors in recordkeeping, and fail to integrate with digital workflows.

### **Google Drive**

Google Drive offers cloud-based storage and collaboration, enabling multiple users to share and access files. It supports basic search and integrates with productivity tools like Docs and Sheets. However, its search is often limited to filenames and simple content scanning, and advanced filters require premium subscriptions.

## **Dropbox**

Google Drive offers cloud-based storage and collaboration, enabling multiple users to share and access files. It supports basic search and integrates with productivity tools like Docs and Sheets. However, its search is often limited to filenames and simple content scanning, and advanced filters require premium subscriptions.

## **Microsoft OneDrive**

OneDrive integrates seamlessly with Microsoft 365, providing real-time collaboration and cloud storage. However, its search features are not deeply customizable, and it lacks advanced text extraction unless paired with enterprise-level solutions.

## **M-Files**

M-Files is a professional document management system with strong metadata-driven organization and compliance tools. While powerful, it is complex and expensive, making it unsuitable for small teams or educational institutions.

## **Proposed DocuDB Document Management System**

DocuDB combines affordability, role-based access, and a user-friendly interface while incorporating advanced search, metadata filters, and text extraction from documents. By ensuring that users can only access and search their own uploaded files, DocuDB improves data relevance, privacy, and efficiency. Its real-time updates, folder management, and customizable features make it well-suited for SMEs, educational institutions, and organizations seeking cost-effective yet powerful solutions.

## Comparative Matrix

Table 1: Comparative Matrix

Features	Traditional File System	Google Drive	Dropbox	OneDrive	M-Files	DocuDB (LLCC)
File Storage & Organization	Limited	Yes	Yes	Yes	Yes	Yes
Cloud Accessibility	No	Yes	Yes	Yes	Yes	No
Real-Time Collaboration	No	Yes	Yes	Yes	Yes	Yes
Version Control	Limited	Yes	Yes	Yes	Yes	Yes
Search Functionality	Limited	Yes	Yes	Yes	Yes	Yes
Text Extraction	No	Limited	Limited	Limited	Yes	Yes
Integration with Other Tools	No	Yes	Yes	Yes	Yes	Limited
Security & Access Control	Limited	Yes	Yes	Yes	Yes	Yes
Mobile App	No	Yes	Yes	Yes	Yes	No
Offline Access	Yes	Yes	Yes	Yes	Yes	No
Advanced Metadata Management	No	Limited	Limited	Limited	Yes	Yes
Real-Time Notifications	No	Yes	Yes	Yes	Yes	Yes
User-Friendly Interface	Limited	Yes	Yes	Yes	Moderate	Yes
Admin Controls	None	Limited	Limited	Limited	Yes	Yes

**Legend:**

**Limited** – The feature is available but with restrictions (e.g., partial functionality or limited capacity).

**SME** – Small and Medium -sized Enterprises.

## RESEARCH METHODOLOGY

### Research Design

This study utilized a descriptive survey method to gather data and information, and an experimental design, following an Agile Iterative Methodology model approach. The study used a descriptive survey, as researchers constructed a questionnaire and gave it to the respondents so that researchers were able to gather user experiences in using the Document Management System. The study aimed to evaluate and validate the effectiveness of the proposed Document management system in addressing the inefficiencies in the traditional Document management system, introducing key features and solutions to enhance its workflow. The proposed solution was thoroughly evaluated to ensure that all inefficiencies from the existing system were resolved as per the agreement between both parties, with adjustments made until all requirements were fully satisfied.

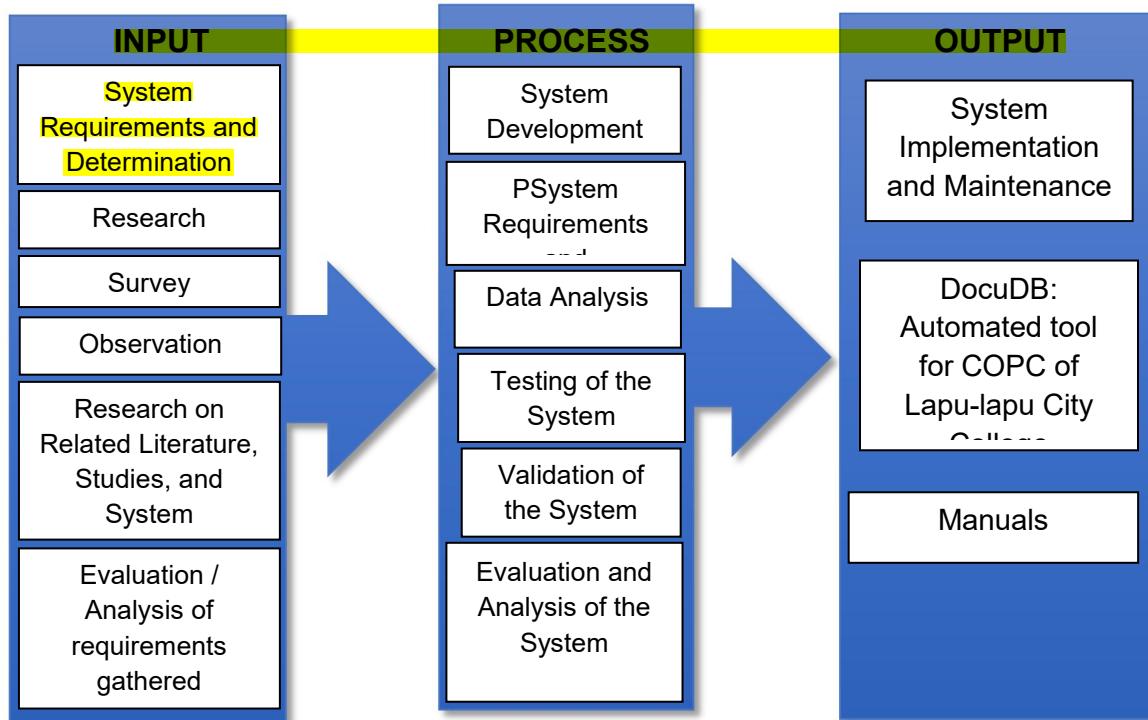
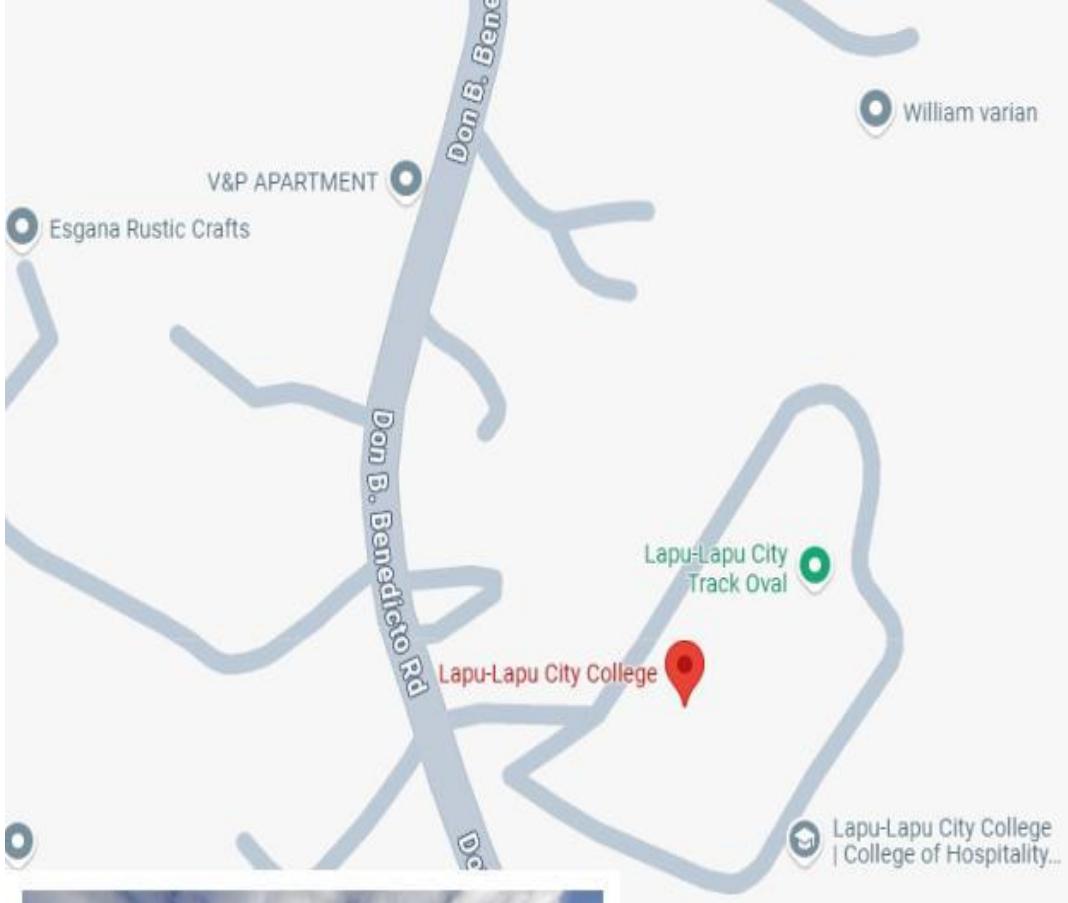


Figure 2: Flow of the Study

## Research Environment

This study was conducted at **Lapu-Lapu City College (LLCC)**, located in Barangay Gun-ob, Lapu-Lapu City, Cebu. The institution was founded in 2010 through City Ordinance No. 354-2010 with the goal of providing quality and affordable education to the residents of Lapu-Lapu City. The college offers various programs such as Bachelor of Elementary Education, Bachelor of Secondary Education, Bachelor of Hospitality Management, and the **Bachelor of Science in Industrial Technology (BSIT)**, which serves as the home department of the researchers. Lapu-Lapu City College continues to grow in both population and infrastructure, with modern facilities that support academic and administrative operations. As a local government college, it faces the challenge of managing numerous institutional documents including academic records, reports, and communications across different departments. These realities make the college an ideal setting for developing and testing **DocuDB**, a digital document management system designed to improve file organization, storage security, and accessibility.

The college's administrative offices and their daily engagement with institutional documents make LLCC a practical and relevant environment for this study. The researchers, being BSIT students of the same institution, were able to observe firsthand the need for a more efficient system that addresses issues of file misplacement, delayed access, and manual record handling. Therefore, conducting the study within LLCC allows for both authentic data gathering and meaningful application of the proposed system.



### Figure 3: Location Map

## Respondents

The respondents of this study were purposively selected individuals from Lapu-Lapu City College who were directly involved in the Certificate of Program Compliance (COPC) process and institutional documentation management. They included a diverse group of administrative staff, faculty members, and personnel responsible for organizing, maintaining, and submitting institutional records.

The purposive sampling technique was applied to ensure that only participants with relevant experience and knowledge in documentation workflows were included. These individuals were considered information-rich cases capable of providing meaningful insights into the effectiveness, usability, and functionality of the DocuDB Document Management System.

All selected respondents had prior exposure to existing document handling processes within the college. Before completing the evaluation survey, participants were given an orientation and guided walkthrough of the DocuDB system to ensure that their feedback was based on actual system usage. Their assessments contributed to the comprehensive evaluation of the system's performance and user experience based on the ISO/IEC 25010:2011 software quality standards, emphasizing its impact on efficiency, accessibility, and data security within the college.

## **Sampling**

This study utilized a purposive sampling technique to identify and select qualified participants. As a form of non-probability sampling, purposive sampling involves the intentional selection of individuals who possess specific characteristics or qualifications necessary for addressing the research problem. The rationale behind this approach lies in its focus on obtaining in-depth and detailed information from participants who are best positioned to provide insights into the system's evaluation.

The chosen respondents were members of Lapu-Lapu City College who have direct involvement in Certificate of Program Compliance (COPC) processes and digital document handling. These participants were considered information rich cases because of their familiarity with documentation workflows, compliance reporting, and institutional record management.

This method was not designed to generalize results to a larger population, but rather to obtain comprehensive and relevant data from individuals qualified to assess the system's performance, usability, and effectiveness within its intended context. Through purposive sampling, the study ensured that all responses reflected the perspectives of key personnel directly engaged with the college's document management process.

## **Research Instruments**

To ensure the accurate and comprehensive collection of data, the researchers utilized a survey questionnaire as the primary research instrument for evaluating the DocuDB Document Management System. The questionnaire was carefully designed in alignment with the ISO/IEC 25010:2011 software quality model, focusing on three major quality attributes: effectiveness, usability, and functionality. It aimed to assess how well

DocuDB performs in addressing the institutional document management needs of Lapu-Lapu City College, particularly for the Certificate of Program Compliance (COPC) documentation process.

The instrument was composed of two main parts. The first part gathered the respondents' demographic and professional information, including their position, familiarity with COPC documentation, and experience in handling institutional files. The second part consisted of a series of evaluative statements related to DocuDB's system performance. Each statement was rated using a four-point Likert scale with the following options: 4 – Strongly Agree, 3 – Agree, 2 – Disagree, and 1 – Strongly Disagree. This format allowed respondents to express their level of agreement toward specific aspects of the system's functionality, usability, and effectiveness.

Before distribution, the survey instrument underwent validation and pilot testing to ensure its clarity, reliability, and appropriateness. A small group of faculty and staff members from Lapu-Lapu City College, who were familiar with document management workflows, participated in the pilot test. Their feedback was used to refine the wording, structure, and content of the questionnaire to ensure that each item accurately measured the intended criteria. This process ensured that the final instrument effectively captured user perceptions and provided meaningful data for evaluating the DocuDB system's overall quality, performance, and user experience.

### **Data Gathering Procedures**

The data gathering process for this study was conducted systematically to ensure the reliability and validity of the collected information. The procedures were designed to align with the descriptive research design and to accurately assess the

DocuDB Document Management System developed for Lapu-Lapu City College. The steps are as follows:

### **1. Preparation Phase**

During this phase, the researchers designed and developed the research instrument the survey questionnaire based on the quality characteristics of the ISO/IEC 25010:2011 standard. The questionnaire was aligned with the study's objectives, focusing on the evaluation of DocuDB's effectiveness, usability, and functionality as a document management system supporting the Certificate of Program Compliance (COPC) process.

### **2. Identification of Respondents**

The respondents were purposively selected from among faculty members, staff, and administrators of Lapu-Lapu City College who are directly involved in COPC documentation and institutional record management. This ensured that all participants had relevant experience with document handling and were capable of providing informed and reliable feedback regarding the DocuDB system.

### **3. Survey Distribution**

The survey questionnaires were distributed personally and through online forms within the college's internal network. Instructions were provided to ensure that respondents fully understood the evaluation process and the criteria used for assessment. The survey captured quantitative responses through the four-point Likert scale to measure the perceived effectiveness, usability, and functionality of DocuDB.

#### **4. System Observation**

Observational data were collected to analyze the actual workflow of existing document management practices within the college environment. This included observing how institutional documents were stored, accessed, and shared before and after DocuDB implementation. The goal was to identify inefficiencies and user pain points that DocuDB sought to address, such as redundancy in manual processes, limited accessibility, and lack of centralized storage.

#### **5. Pilot Testing**

The DocuDB system was initially tested by a small subset of qualified respondents from Lapu-Lapu City College. The pilot group assessed the system's usability, functionality, and security features in a controlled environment. Their feedback was used to identify interface issues, optimize workflow processes, and make refinements before broader deployment.

#### **6. Data Compilation and Validation**

After data collection, all survey responses and observations were compiled and reviewed for completeness and accuracy. Inconsistent or incomplete responses were excluded to maintain data reliability. The validated dataset was then analyzed using descriptive statistics to determine the weighted mean and standard deviation for each criterion under the ISO/IEC 25010 framework.

These data collection methods ensured that the study captured a comprehensive understanding of the DocuDB system's quality, performance, and user acceptance,

leading to a meaningful evaluation of its impact on institutional documentation management at Lapu-Lapu City College.

### **Statistical Treatment of Data**

Descriptive statistical tools were applied to appropriately present, describe, and analyze the collected data from the respondents through survey questionnaires.

1. The Weighted Mean was used to summarize and interpret the level of effectiveness, usability, and functionality of DocuDB in addressing the needs of the institution and users involved in the Certificate of Program Compliance (COPC) process at Lapu-Lapu City College.

Formula:

$$\bar{x} = \frac{\sum fx}{N}$$

Where:  $\bar{x}$  is the weighted mean

$f$  is the frequency

$x$  are the weights assigned

$N$  is the total number of the respondents

2. Standard Deviation was also used to measure the variability or dispersion of the responses.

Formula:

$$s = \sqrt{\frac{\sum (X - \bar{x})^2}{n - 1}}$$

Where:  $s$  is the sample standard deviation

$\bar{x}$  is the sample mean

$X$  is each individual sample value

*n is the total number of samples*

## Scoring Procedures

The four-point Likert scale shown below was employed to categorically rate the extent of the effectiveness and overall performance of DocuDB in addressing the document management needs of Lapu-Lapu City College.

Table 2: Likert Scale for System Effectiveness

Rating	Mean Range Chart	Descriptive Rating
4	3.26 – 4.00	Very Effective
3	2.51 – 3.25	Effective
2	1.76 – 2.50	Less Effective
1	1.00 – 1.75	Not Effective

For the level of system usability of the DocuDB system, the four-point Likert scale shown below was employed to determine how user-friendly, accessible, and easy-to-navigate the system is for both administrators and members.

Table 3: Likert Scale for System Usability

Rating	Mean Range Chart	Descriptive Rating
4	3.26 – 4.00	Very User-Friendly
3	2.51 – 3.25	User-Friendly
2	1.76 – 2.50	Less User-Friendly
1	1.00 – 1.75	Not User-Friendly

For the level of system functionality of the DocuDB system, the four-point Likert scale shown below was employed to determine how well the system performs its core operations such as uploading, organizing, searching, and sharing of institutional files.

Table 4: Likert Scale for System Functionality

Rating	Mean Range Chart	Descriptive Rating
4	3.26 – 4.00	Highly Functional
3	2.51 – 3.25	Functional
2	1.76 – 2.50	Partially Functional
1	1.00 – 1.75	Not Functional

## **Definition of Terms**

**Admin Controls** - A suite of features within the DocuDB system that allows administrators to manage users, monitor system activity, and maintain logs. This includes functions for adding/deactivating users, assigning roles, and viewing audit trails.

**Audit Logs** - A security feature that automatically records user activities within the DocuDB system, such as file uploads, deletions, sharing, and permission changes. Each log entry typically includes the user's identity, the action performed, and a timestamp to ensure accountability.

**Automated Backups** - A feature within DocuDB that automatically creates and stores copies of system data and documents at regular intervals. This ensures data can be recovered in the event of accidental deletion, corruption, or system failure, supporting business continuity.

**Certificate of Program Compliance (COPC)** - A specific department or unit within Lapu-Lapu City College (LLCC) responsible for ensuring academic programs adhere to standards. This study focuses on developing DocuDB to address the document management challenges within this office.

**Client-Server Architecture** - A network structure where client devices (like user computers or phones) request and display data and services from a central server (where DocuDB is hosted). This model allows for centralized management, security, and updates.

**Collaboration Features** - Tools within DocuDB designed to facilitate user interaction and joint work on documents. These include the ability to add comments, make

annotations on files, and share documents with specific permissions, thereby enhancing teamwork and communication.

**Content Extraction** - A functionality in DocuDB that indexes the textual content inside documents (e.g., PDFs, Word files). This allows users to search for files not just by filename, but also by the words and phrases contained within the document itself, significantly improving retrieval efficiency.

**Dashboard** - The main graphical user interface (GUI) presented to a user after logging into DocuDB. It provides an overview of key information and quick access to features, such as recently uploaded files, shared documents, and system statistics, tailored to the user's role (Admin or User).

**Document Security** - The collective measures implemented in DocuDB to protect institutional files from unauthorized access, modification, or destruction. This encompasses role-based access control, file and folder permissions, audit logs, and secure HTTPS communication.

**Digital Document Management System (DMS)** - A software system (like DocuDB) used to store, manage, track, and organize electronic documents and images. It replaces traditional physical filing cabinets and fragmented digital storage with a centralized, secure, and searchable platform.

**DocuDB** - The automated document management system developed in this study. It is a web-based platform designed specifically for Lapu-Lapu City College to provide secure storage, organization, retrieval, and sharing of institutional files for the COPC.

**Express.js** - A back-end web application framework for Node.js used to build the server-side logic and RESTful APIs for the DocuDB system. It handles tasks like routing, processing HTTP requests, and interacting with the database.

**File Management** - The core set of functions in DocuDB for handling documents. This includes uploading files from a user's device, storing them on the server, organizing them into folders, previewing them in the browser, downloading them, and sharing them with other users.

**Functional Requirements** - The specific behaviors and functions that the DocuDB system must perform, as identified during the Requirement Phase. These include file uploading, searching, sharing, and role-based access control.

**Iterative Methodology** - The software development framework adopted for this project. It involves breaking down the development process into repeated cycles (iterations), where each cycle includes planning, design, development, and testing. This allows for continuous refinement and incorporation of user feedback.

**Lapu-Lapu City College (LLCC)** - The academic institution where this study is conducted and for which the DocuDB system is specifically designed. It is located in Gun-ob, Lapu-Lapu City, Philippines.

**Metadata** - Descriptive data about a document stored in the system. In DocuDB, this includes information such as the filename, original name, file type, size, owner, upload date, and any extracted text, which is used to power the search and filtering functions.

**MongoDB** - A NoSQL database program used as the primary data storage solution for DocuDB. It was selected for its flexibility, scalability, and ability to handle the unstructured and semi-structured data typical of document management systems.

**Non-Functional Requirements** - The technical specifications and quality attributes of the DocuDB system, as identified during the Requirement Phase. These include system responsiveness, platform compatibility, security protocols, and performance under a load of 500 concurrent users.

**Product Backlog** - In the context of the Iterative Methodology, a prioritized list of all desired features, enhancements, and tasks for the DocuDB system. It includes user stories and acceptance criteria that guide the development team throughout the project.

**React.js** - A JavaScript library used to build the frontend user interface of DocuDB. It enables the creation of a dynamic and responsive single-page application (SPA) where content updates without requiring a full browser refresh.

**RESTful API** - A set of defined rules and conventions (Representational State Transfer) used by the DocuDB frontend (React.js) to communicate with the backend (Express.js). These APIs allow for operations like creating, reading, updating, and deleting documents and user data.

**Role-Based Access Control (RBAC)** - A security model implemented in DocuDB where system access and permissions (e.g., to view, edit, or share files) are granted to users based on their assigned role within the organization (e.g., Administrator or Regular User). This ensures users can only access data relevant to their duties.

**Scalability** - The capacity of the DocuDB system to handle a growing amount of work—such as more users, files, and transactions—by adding resources to the system. The study notes that the system is designed for small to medium-sized groups, indicating a current limitation in large-scale scalability.

**System Architecture** - The structural design of the DocuDB system, which is organized into three main layers: the Presentation Layer (React.js frontend), the Application Layer (Express.js backend), and the Data Layer (MongoDB database). This architecture separates concerns for better maintainability, scalability, and security.

**Text Extraction** - (See also: *Content Extraction*) The underlying technological process of identifying and indexing machine-readable text from within document files (e.g., using OCR for images or parsing text from PDFs). This data is then used to power the advanced search functionality in DocuDB.

**Three-Tier Architecture** - The specific structural model adopted for DocuDB, which logically separates the system into three independent tiers: the Presentation Layer (UI), the Application Layer (business logic), and the Data Layer (database). This separation enhances modularity, security, and ease of maintenance.

**User Acceptance Testing (UAT)** - The final phase of testing where the completed DocuDB system is evaluated by the actual end-users (COPC faculty and staff) to ensure it meets their needs and works correctly in a real-world environment before final deployment.

**User Authentication** - The security process in DocuDB that verifies a user's identity before granting access to the system. For this study, it is restricted to users with a valid Lapu-Lapu City College domain email address.

**User Story** - A simple, natural language description of a software feature from the end-user's perspective, used in the product backlog to define requirements. For example, "As a user, I want to search documents by name so I can find them quickly."

**Version Control** - (See also: *Version History*) The broader capability of a system to manage changes to documents over time. In DocuDB, this feature allows the system to automatically save versions of a file as it is updated, preventing data loss and allowing users to track the evolution of a document.

**Version History** - A feature that tracks changes made to a document over time. It allows users to view or revert to previous versions of a document, which is crucial for collaboration and maintaining the integrity of important files.

## CHAPTER 2

### PRESENTATION, DATA ANALYSIS AND INTERPRETATION

## TECHNICAL REQUIREMENTS

### Hardware and Software Requirements

Table 5: Hardware and Software Requirements

Category	Requirement	Client (End-User Device)	Server (Hosting Environment)
Hardware	Processor	Intel i3 / AMD equivalent or higher	Quad-core 2.0 GHz or higher
	Memory (RAM)	4 GB minimum (8 GB recommended)	8 GB minimum (16 GB recommended)
	Storage	2 GB free (for browser cache, docs)	200 GB SSD (expandable based on file storage)

	Network	Stable broadband (5 Mbps+)	1 Gbps LAN / Internet backbone
<b>Software</b>	Operating System	Windows 10/11, macOS, Linux, Android/iOS (mobile)	Ubuntu Linux 20.04+ / Windows Server 2019+
	Web Browser	Latest Chrome, Firefox, Edge, or Safari	Nginx / Apache as web server
	Frameworks & Runtime	Not required (runs in browser)	Node.js v18+ / PHP 8+ runtime
	Database	<i>Not required</i>	MongoDB / Firebase / DynamoDB (NoSQL)
	Other Tools	PDF Reader (optional for viewing docs)	Git, Docker (optional), SSL certificate for HTTPS
<b>Performance</b>	Concurrent Users	Handles up to 500 concurrent sessions	Must scale to handle 500+ users
	File Upload	Browser supports uploads up to 50 MB	Server supports uploads up to 50 MB per file
<b>Security</b>	Authentication	Login/Logout with session management	Enforce HTTPS, JWT/Session auth, password hashing (optional)

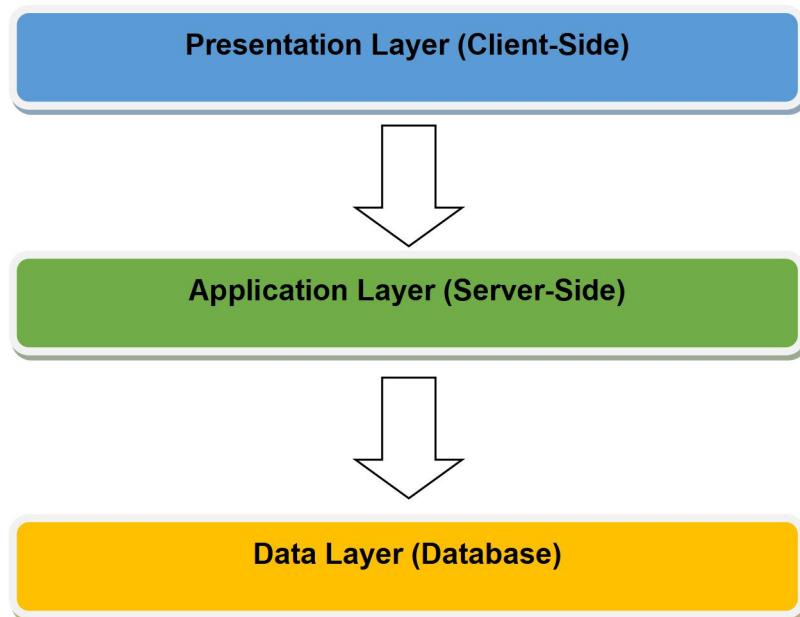
The DocuDB requires precise hardware and software settings to enable dependable and efficient operations. Table 5 shows that the client's hardware needs are focused on providing a smooth user experience, whereas the server's requirements emphasize processing power and storage to support numerous users and database activities. The software requirements comprise the tools and technologies required to construct, host, and operate the system efficiently.

## System Architecture

The DocuDB system is a web-based document and file management platform built using Express.js for the backend, React.js for the frontend, and Bootstrap 5 for

responsive styling. It follows a modern three-tier architecture and can be deployed on cloud hosting or a dedicated server, making it accessible to users over the internet.

## DOCUDB SYSTEM ARCHITECTURE



As shown in Figure 4, the DocuDB system follows a three-tier architecture.

### 1. Presentation Layer (Client-Side)

This layer provides an interactive and responsive user interface built with React.js. Bootstrap is used for styling, ensuring accessibility and responsiveness across devices. Users interact with the system through features such as file/folder navigation, document previews, search functionality, and role-based access controls. All interactions with the server occur through RESTful APIs via HTTPS.

### 2. Application Layer (Backend / API Server)

The server-side logic is implemented using Express.js with Node.js, which manages user authentication, file/folder operations, search functions, and role-

based permissions. Multer is integrated for handling secure file uploads. The backend follows an MVC-like architecture using modular routes, models, and controllers. APIs are exposed to the frontend for tasks such as uploading documents, retrieving file metadata, and managing shared folders.

### **3. Data Layer (Database)**

The MongoDB database, accessed via Mongoose, stores system data including users, files, folders, permissions, and metadata. Each uploaded document's metadata (filename, original name, size, mimetype, owner, and upload date) is stored in MongoDB, while the actual files are saved in a secure /uploads directory. Mongoose provides schema validation, query optimization, and prevents injection vulnerabilities.

## **System Workflow**

The DocuDB workflow consists of four main steps:

### **1. User Request via HTTPS**

Users interact with DocuDB through a browser, submitting HTTPS requests such as uploading a file, searching documents, creating folders, or sharing files. These requests are routed to the Express backend via RESTful API endpoints.

### **2. Request Processing on the Server**

The Express backend receives the request and processes it according to business logic. For example, when a file is uploaded, the backend validates the user, stores the file in /uploads, and saves its metadata in MongoDB. For search

operations, the backend filters based on metadata (original name, type, or upload date).

### 3. Database Interaction

MongoDB queries are executed to fetch, update, or store data. Examples include retrieving all files in a folder, storing details of a new upload, or fetching search results. Mongoose ensures structured queries, schema validation, and safe handling of concurrent operations.

### 4. Frontend Updates via API Response

The Express backend returns a JSON response to the React frontend. React processes this data and dynamically updates the UI without requiring a full page reload. For example, when a user searches for a document, results are instantly displayed in a dropdown list (similar to YouTube's search suggestion system).

## Product Backlog

This backlog provides a structured approach in developing the DocuDB ensuring that key functionalities are prioritized and implemented.

Table 6: Product Backlog

DocuDB	Task	Owner	Start Date	End Date	Duration (days)
Feature 1: User Authentication & Management					
User Story 1.1: As a user, I want the system to automatically detect my role (Admin/User)	Acceptance Criteria: <ul style="list-style-type: none"><li>• No manual login required.</li><li>• Role is identified on entry.</li><li>• Admin → Admin Dashboard, User →</li></ul>				

so I'm redirected to the correct dashboard.	User Dashboard.				
	Implement role-checking logic	Backend	10-Aug-25	13-Aug-25	3
	Setup auto-redirects for Admin/User	Backend	10-Aug-25	13-Aug-25	3
	Design role-specific dashboards	UI/UX	10-Aug-25	13-Aug-25	3
<b>Feature 2: Document &amp; Folder Management</b>					
User Story 2.1: As a user, I want to upload and organize documents in folders so I can store them properly.	Acceptance Criteria: <ul style="list-style-type: none"><li>Upload supports PDF, Word, Excel, images.</li><li>Files can be placed inside folders.</li><li>Upload progress visible.</li></ul>				
	Design file/folder UI	UI/UX	13-Aug-25	15-Aug-25	2
	Implement file upload API	Backend	13-Aug-25	15-Aug-25	2
	Connect frontend upload with backend	Frontend	13-Aug-25	15-Aug-25	2
User Story 2.2: As a user, I want to delete, rename, and move documents so I can manage them easily.	Acceptance Criteria: <ul style="list-style-type: none"><li>Users can delete/rename/move files &amp; folders.</li><li>Changes reflect immediately.</li></ul>				
	Implement rename/delete logic	Backend	15-Aug-25	19-Aug-25	4
	Update UI for folder/file actions	Frontend	15-Aug-25	19-Aug-25	4

	Add confirmation dialogs	Backend	15-Aug-25	19-Aug-25	4
<b>Feature 3: Document Sharing &amp; Permissions</b>					
User Story 3.1: As a user, I want to share a document with another user so they can view or edit it.	Acceptance Criteria: <ul style="list-style-type: none"><li>• Permissions: Read, Write, Full.</li><li>• Shared users can only access assigned documents.</li></ul>				
	Design sharing UI with permissions	UI/UX	22-Aug-25	25-Aug-25	3
	Implement permission logic in backend	Backend	22-Aug-25	25-Aug-25	3
	Connect UI to backend	Frontend	22-Aug-25	25-Aug-25	3
User Story 3.1: As a user, I want to share a document with groups or teams	Acceptance Criteria: <ul style="list-style-type: none"><li>• Share documents/folders with groups or teams</li><li>• Groups have same access levels (Read, Write, Full)</li></ul>				
	Add group sharing option in UI	UI/UX			0
	Implement group-based sharing logic	Backend			0
	Enable group invite and access control	Frontend			0
<b>Feature 4: Search &amp; Filters</b>					
User Story 4.1: As a user, I want to search documents by name/type/date so I can find them quickly.	Acceptance Criteria: <ul style="list-style-type: none"><li>• Search is case-insensitive.</li><li>• Results update in real-time.</li></ul>				
	Design search bar UI	UI/UX	20-Aug-25	21-Aug-25	1
	Implement backend search API	Backend	25-Aug-25	31-Aug-25	6

	Integrate search results in UI	Frontend	25-Aug-25	31-Aug-25	6
User Story 4.2: As a user, I want to search documents the file according to the file name.	Acceptance Criteria: <ul style="list-style-type: none"><li>• Search within document contents (OCR/Text)</li><li>• Show highlighted keyword results</li></ul>				
	Integrate content-based search algorithm	Backend			0
	Add content search toggle in UI	Frontend			0
	Design content-highlighted results	UI/UX			0
<b>Feature 5: Admin User &amp; Role Management</b>					
User Story 5.1: As an Admin, I want to manage user accounts so I can control who has access.	Acceptance Criteria: <ul style="list-style-type: none"><li>• Admin can view, add, edit, deactivate users.</li><li>• User list shows role (Admin/User).</li></ul>				
	Design user management UI	UI/UX	09-Sep-25	12-Sep-25	3
	Implement backend logic for user roles	Backend	09-Sep-25	15-Sep-25	6
	Add audit log for user changes	Backend	09-Sep-25	15-Sep-25	6
<b>Feature 6: System Monitoring &amp; Logs</b>					
User Story 6.1: As an Admin, I want to view system activity logs so I can track usage.	Acceptance Criteria: <ul style="list-style-type: none"><li>• Logs include uploads, deletions, sharing, role changes.</li><li>• Logs show user, action, timestamp.</li></ul>				
	Implement backend logging mechanism	Backend	16-Sep-25	22-Sep-25	6
	Create admin log viewer UI	UI/UX	15-Sep-25	18-Sep-25	3

	Add filter/search for logs	Frontend	15-Sep-25	18-Sep-25	3
--	----------------------------	----------	-----------	-----------	---

## DocuDB Program Flow

The program workflow ensures seamless navigation, enabling users to easily manage, share, and access their documents in a secure environment. The following descriptions illustrate how administrators and members interact with the system, highlighting its intuitive design, role-based features, and efficient document management capabilities.

### Administrator Program Flow

Figure 5 illustrates the administrator's workflow in the DocuDB system. The process begins with the admin login, where credentials are authenticated to grant access to the administrator dashboard. The dashboard provides an overview of uploaded files, folders, users, and shared items.

From the dashboard, administrators can:

- **Manage Users:** Monitor account activity and change status whether active or inactive.

- **Monitor Shared Content:** View all files and folders shared among users or groups.
- **Search and Filter:** Conduct advanced searches by filename, file type, upload date, or owner. The admin can also perform content-based searches that scan extracted text within documents for faster retrieval.
- **System Logs and Reports:** Track user activities such as uploads, deletions, and sharing actions. Logs display timestamps, user IDs, and corresponding actions for system transparency.

Additional features include real-time notifications for uploads, deletions, and shared file updates, as well as text extraction from documents for advanced search and indexing. The administrator ensures system integrity, user accountability, and smooth document management across the platform

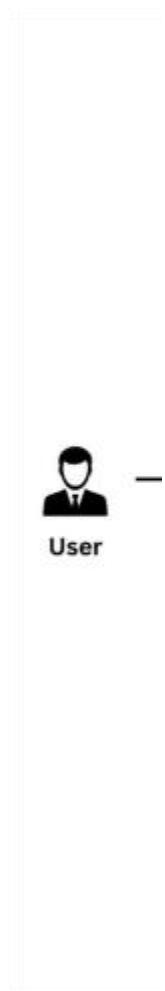


Figure 5: DocuDB Program Workflow

### **Member/User Program Flow**

Figure 6 shows the member's workflow in the DocuDB system. After logging in with their credentials, the user is directed to a personalized dashboard, which summarizes their uploaded files, shared documents, and available folders.

From the dashboard, users can:

- **Manage Files:** Upload new documents (PDF, Word, Excel, and images), download, preview inline, rename, or delete their own files.
- **Organize Folders:** Create, rename, or move folders to properly categorize documents and maintain a structured storage hierarchy.
- **Search and Quick Access:** Use the search bar to find files and folders by filename, file type, or upload date. The system also supports content-based search, allowing users to find files based on extracted text from within documents.
- **Share Documents:** Share files or folders with other users or groups by entering email addresses and assigning permission levels (View, Edit). Shared users can only access documents according to the assigned permissions.
- **Trash Management:** Move deleted files and folders to Trash, restore them if needed, or permanently delete them from the system.

The DocuDB system ensures that members can efficiently manage their own documents, collaborate securely with others, and maintain full control over their personal file repository.

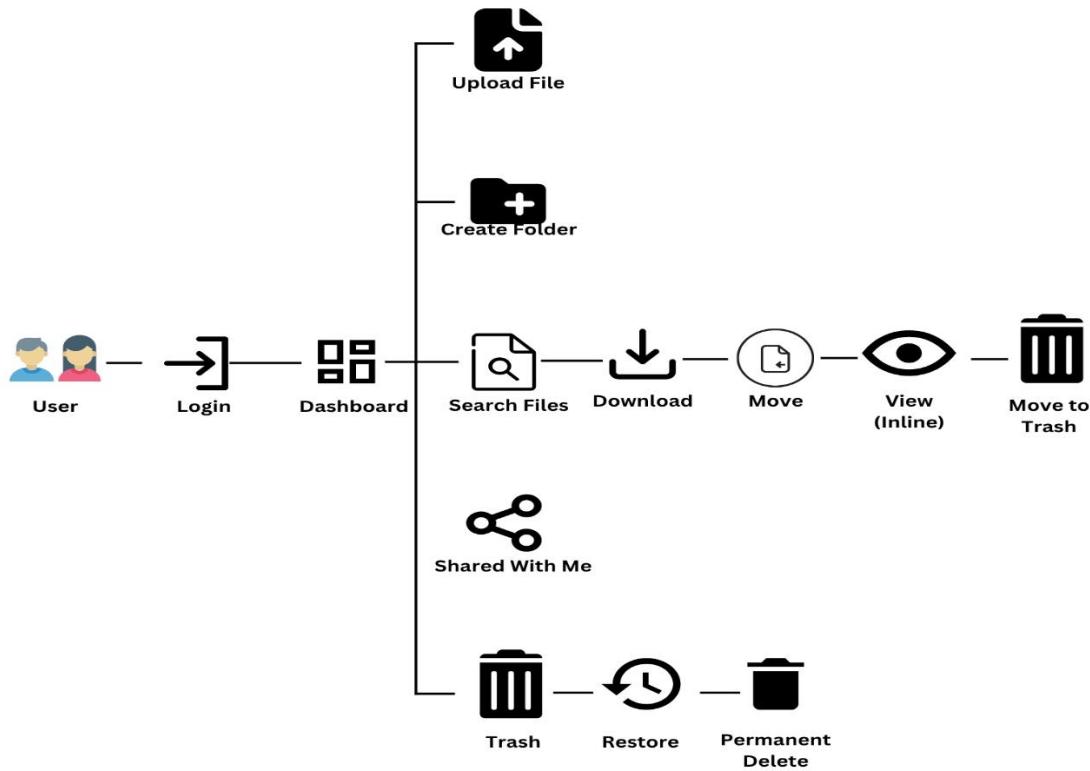


Figure 6: DocuDB Program Workflow

## DEPLOYMENT

## PLAN

This deployment plan outlines the steps and process for deploying the DocuDB Document Management System to the production environment hosted on the Lapu-Lapu City College Server Infrastructure (or any equivalent shared hosting), ensuring a secure,

stable, and optimized deployment that supports the system's intended functionality and availability for all academic and administrative users.

## **Deployment Steps**

### **Step 1: Prepare the Local Server**

1.1 Use a Windows computer or dedicated local server within the Lapu City College IT infrastructure.

1.2 Install the following required software:

- Node.js (version 18 or higher) – for backend and frontend operations.
- MongoDB Community Server – as the system's main database.
- Visual Studio Code (VS Code) – for code editing and testing.
- Git (optional) – for version control and project management.

1.3 Create a main directory, e.g., C:\DocuDB, where the backend and frontend files will be stored.

### **Step 2: Copy and Setup the Project**

2.1 Copy the DocuDB project folder from the development machine to the server directory.

2.2 Ensure the project has two main folders:

- backend/ — contains server and API logic (Node + Express).
- frontend/ — contains the React-based user interface (Vite).

2.3 Open Command Prompt or VS Code Terminal, then install dependencies:

Backend:

```
cd C:\DocuDB\server
```

```
npm install
```

Frontend:

```
cd C:\DocuDB\client
```

```
npm install
```

2.4 Create a .env file inside the backend folder to securely store configuration settings:

```
# Server Configuration
```

```
PORT=3001
```

```
HOST=0.0.0.0
```

```
# Database Configuration
```

```
MONGO_URI=mongodb://127.0.0.1:27017/docudb
```

```
# Security
```

```
JWT_SECRET=docudb_secure_key_2025
```

```
# Client (Frontend)
```

```
CORS_ORIGIN=http://localhost:5173
```

2.5 In your index.js, ensure you have the following line at the top:

```
require("dotenv").config();
```

Then replace all hardcoded values with process.env variables.

### **Step 3: Configure and Start the Database**

3.1 DocuDB will automatically create the required collections (users, files, folders, logs) when it runs

3.3 Verify connection by running:

```
mongo
```

```
show dbs
```

3.4 DocuDB will automatically create the required collections (users, files, folders, logs) when it runs for the first time.

#### **Step 4: Run and Test the Backend**

4.1 Start the backend API:

```
cd C:\DocuDB\backend
```

```
node index.js
```

4.2 If successful, the terminal should display:

```
✓ MongoDB connected
```

```
Server running on http://0.0.0.0:3001
```

4.3 Test the API in your browser or Postman:

```
http://localhost:3001/files
```

```
http://localhost:3001/folders
```

#### **Step 5: Build and Run the Frontend**

5.1 In a new terminal:

```
cd C:\DocuDB\frontend
```

```
npm run dev
```

5.2 Access the system via:

<http://localhost:5173>

5.3 If successful, the DocuDB login page should appear.

5.4 Test core functionalities:

- User authentication and login.
- File upload, preview, and download.
- Folder creation and sharing.
- Search and trash management.

## **Step 6: Deploy DocuDB on the Local Network**

6.1 Get the server's local IP address by typing in Command Prompt:

Ipconfig

6.2 Update your .env file and frontend configuration to use the IP address,  
for example:

VITE\_BACKEND\_URL=http://192.168.1.19:3001

6.3 Restart both the backend and frontend:

npm run dev # frontend

node index.js # backend

6.4 Other computers in the same network can now access:

http://192.168.1.100:5173

## USER GUIDE

### ProTask System User Manual (Organization Admin Side)

#### 8. INTRODUCTION

DocuDB is a web-based document management system designed for Lapu-Lapu City College to securely store, organize, and manage institutional files.

This manual provides step-by-step instructions for administrators to upload, manage, share, and monitor documents efficiently.

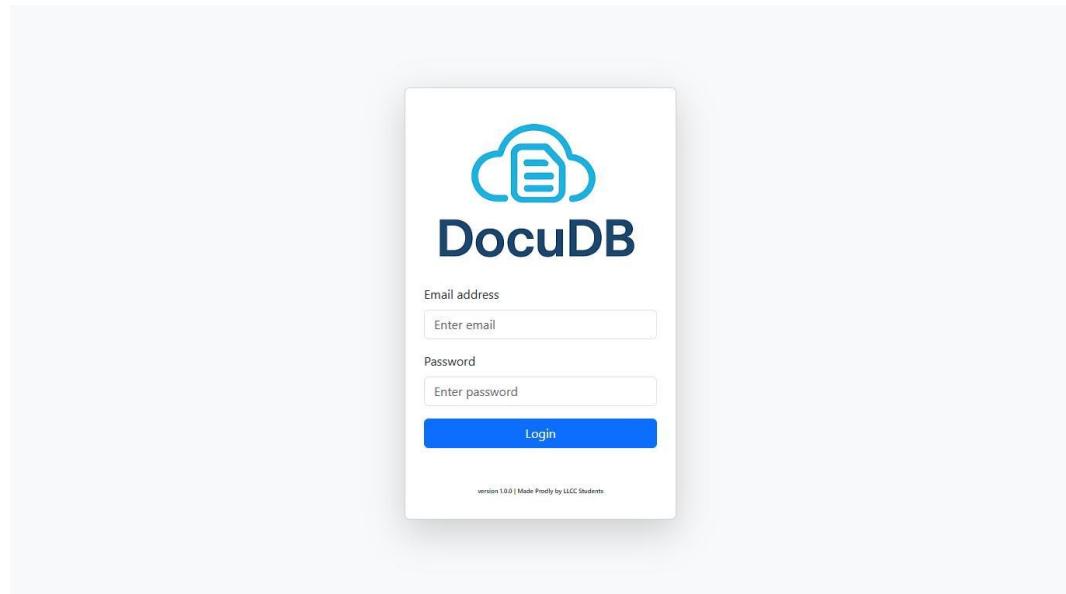
#### 9. ACCESSING THE SYSTEM

##### 2.1 System Requirements

- A device with a web browser (Chrome, Firefox, Edge, etc.)
- Local Network or Internet connection
- URL: <http://localhost:5173> (Differs on Deployment)

##### 2.2 Log in

4. Open your web browser and go to the DocuDB login page.
5. Enter your Lapu-Lapu City College administrator email and password.
6. Click “Login” to access the admin dashboard.

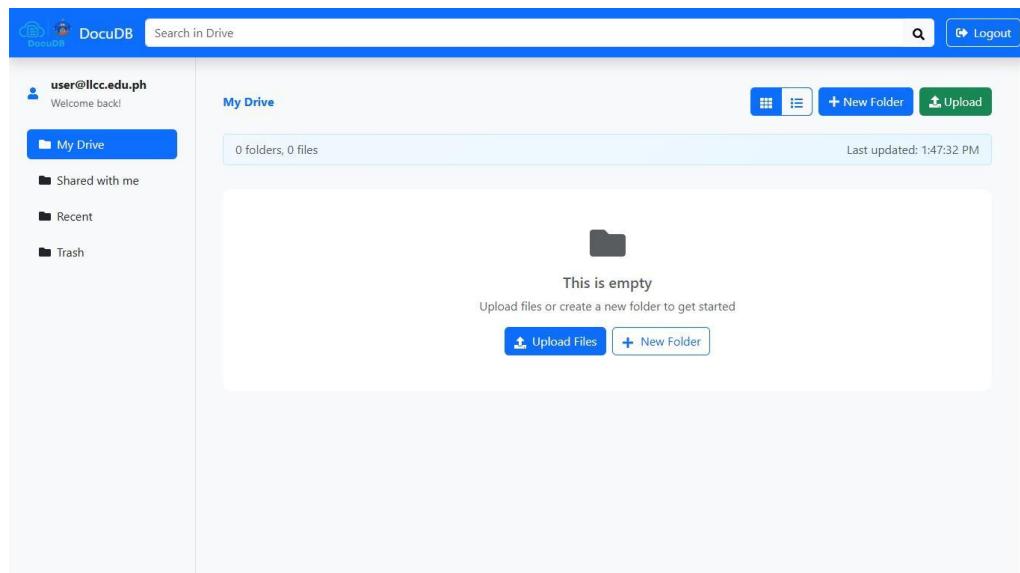


## 10. DASHBOARD OVERVIEW

Once logged in, the Admin Dashboard displays all uploaded files, users, and activities in the system. The dashboard provides direct access to file management, user monitoring, and activity logs.

### 10.1 Navigation Menu

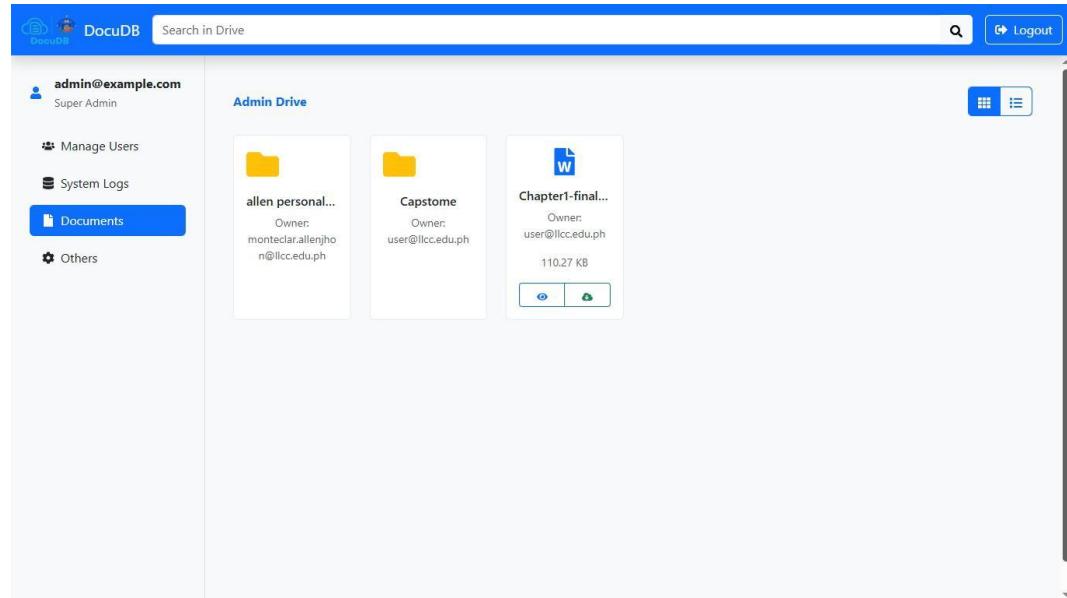
- Nav1: Manage Users – View, add, edit, or remove users.
- Nav2: System Logs – Display a list of all system activities, such as uploads, deletions, and sharing.
- Nav3: Documents – View, preview, or delete uploaded files and folders.
- Nav4: Others – View other features such as settings, filters, and admin configurations.



## 11. MANAGING FILES AND FOLDERS

### 11.1 View and Manage Files

4. Click on the Documents navigation tab.
5. The list of uploaded files and folders will appear.
6. Each document shows its name, owner, size, and control options (View, Download, Delete).



## 12. MANAGING Users

### 12.1 View and Manage Status of Users

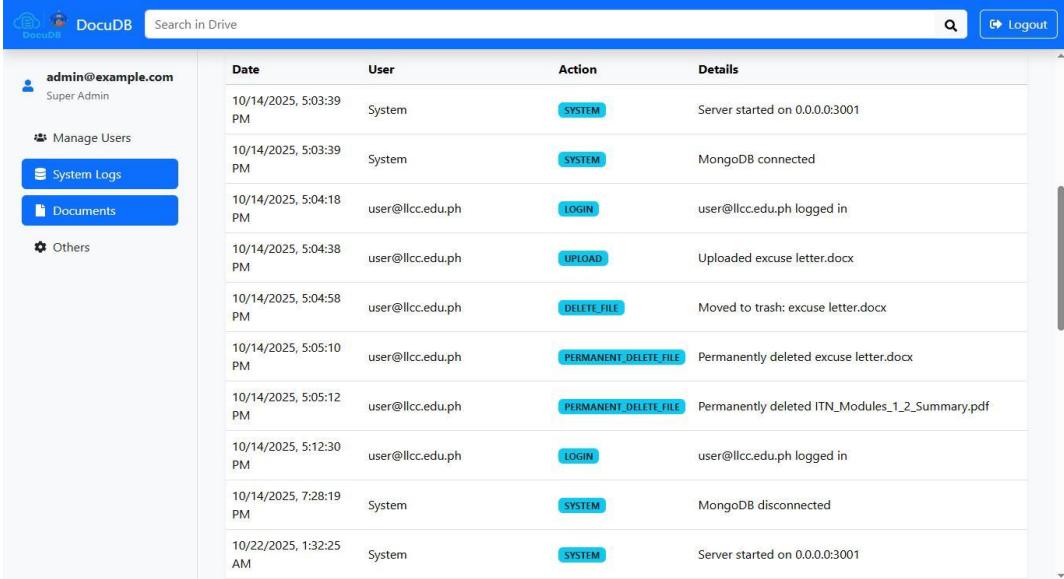
3. From the Manage Users list, click the icon to modify a user's status (Active or Inactive).
4. All actions are recorded in the System Logs for transparency.

ID	Name	Email	Role	Status	Actions
1	N/A	✉️ user@llcc.edu.ph	user	active	<button>Set Active</button>
2	N/A	✉️ admin@llcc.edu.ph	admin	active	<button>Set Active</button>
3	N/A	✉️ abondiente.michael@llcc.edu.ph	user	active	<button>Set Active</button>
4	N/A	✉️ aparta.vanclark@llcc.edu.ph	user	active	<button>Set Active</button>
5	N/A	✉️ bartiana.jomiloandrew@llcc.edu.ph	user	active	<button>Set Active</button>
6	N/A	✉️ monteclar.allenjhon@llcc.edu.ph	user	active	<button>Set Active</button>

## 13. SYSTEM MONITORING

### 13.1 View Logs

3. Click the System Logs tab from the navigation menu.
4. The log table displays:
  - File Uploads
  - File Deletions
  - Shares and Permissions
  - User Additions and Removals
  - Login and Logout Activities



The screenshot shows the DocuDB application interface. On the left, there is a sidebar with user information (admin@example.com, Super Admin), navigation links (Manage Users, System Logs, Documents), and an Others section. The main area is a table titled "System Logs" with columns: Date, User, Action, and Details. The table contains ten log entries. The "System Logs" link in the sidebar is highlighted in blue, indicating it is the active tab.

	Date	User	Action	Details
admin@example.com	10/14/2025, 5:03:39 PM	System	SYSTEM	Server started on 0.0.0:3001
	10/14/2025, 5:03:39 PM	System	SYSTEM	MongoDB connected
Manage Users	10/14/2025, 5:04:18 PM	user@lcc.edu.ph	LOGIN	user@lcc.edu.ph logged in
System Logs	10/14/2025, 5:04:38 PM	user@lcc.edu.ph	UPLOAD	Uploaded excuse letter.docx
Documents	10/14/2025, 5:04:58 PM	user@lcc.edu.ph	DELETE_FILE	Moved to trash: excuse letter.docx
	10/14/2025, 5:05:10 PM	user@lcc.edu.ph	PERMANENT_DELETE_FILE	Permanently deleted excuse letter.docx
Others	10/14/2025, 5:05:12 PM	user@lcc.edu.ph	PERMANENT_DELETE_FILE	Permanently deleted ITN_Modules_1_2_Summary.pdf
	10/14/2025, 5:12:30 PM	user@lcc.edu.ph	LOGIN	user@lcc.edu.ph logged in
	10/14/2025, 7:28:19 PM	System	SYSTEM	MongoDB disconnected
	10/22/2025, 1:32:25 AM	System	SYSTEM	Server started on 0.0.0:3001

### 14. LOGOUT

Click the **Logout** button on the top-right corner to securely exit the system.

## **DocuDB System User Manual (Member/User Side)**

### **1. INTRODUCTION**

DocuDB is a web-based document management system developed for Lapu-Lapu City College students and staff to securely upload, organize, and share files within a centralized digital workspace.

This manual guides users on how to upload, manage, and share documents efficiently.

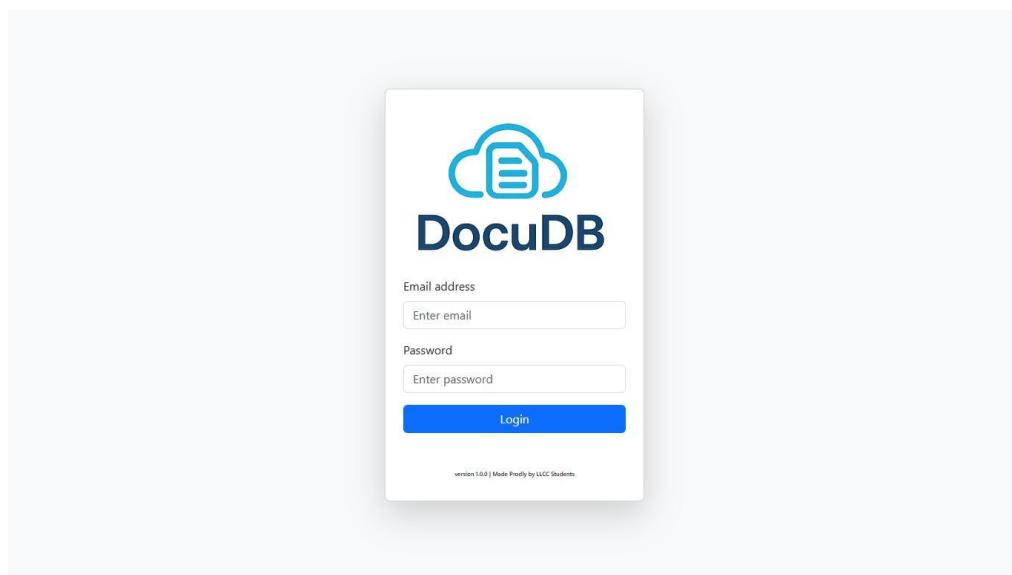
### **2. ACCESSING THE SYSTEM**

#### **2.1 System Requirements**

- A device with a web browser (Chrome, Firefox, Edge, etc.)
- Local Network or Internet connection
- URL: <http://localhost:5173> (Differs on Deployment)

#### **2.1 System Requirements**

4. Go to the DocuDB login page.
5. Enter your Lapu-Lapu City College email and password.
6. Click Login to access your User Dashboard.

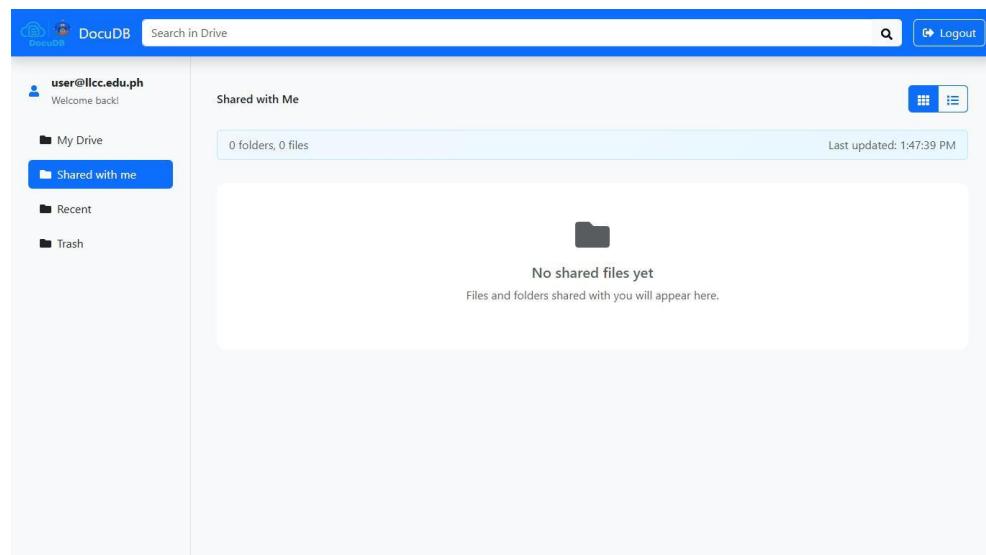


## 2. DASHBOARD OVERVIEW

After logging in, the **User Dashboard** displays your personal file storage area, called **My Drive**, along with other sections for collaboration and management.

### 3.1 Navigation Menu

- Nav1: My Drive – Displays all your uploaded files and folders.
- Nav2: Shared with Me – Lists documents shared with you by other users.
- Nav3: Recent – Shows recently opened or uploaded files.
- Nav4: Trash – Temporarily stores deleted files for recovery.
- Nav5: Logout – Ends the session securely.

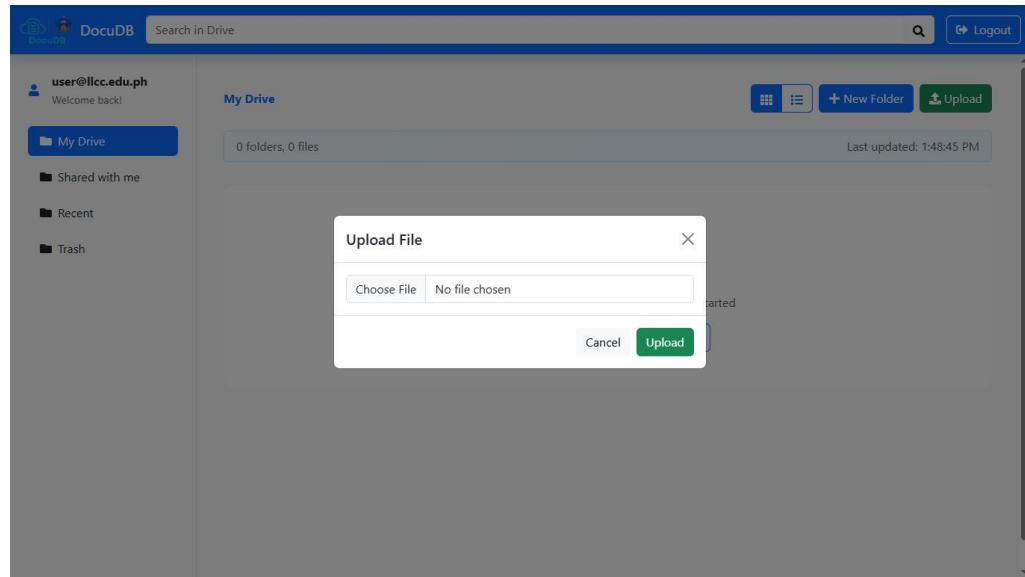


## 4. MANAGING YOUR FILES

### 4.1 Uploading Files

4. Click the **Upload** button.
5. Select files from your computer.

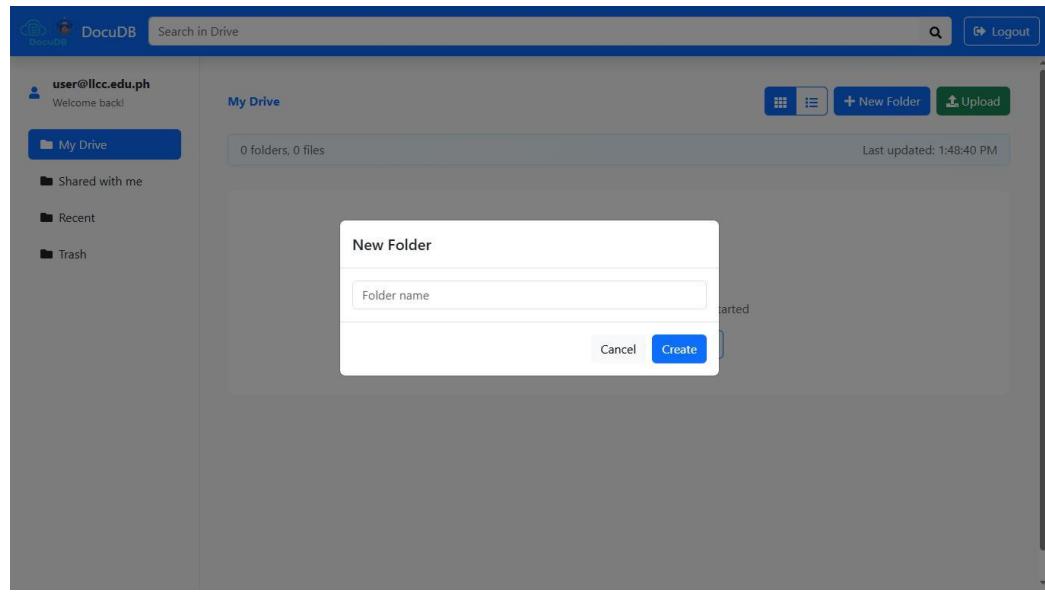
6. Uploaded files will appear in **My Drive**.



Supported Formats: PDF, Word, Excel, PowerPoint, Images, and Text files.

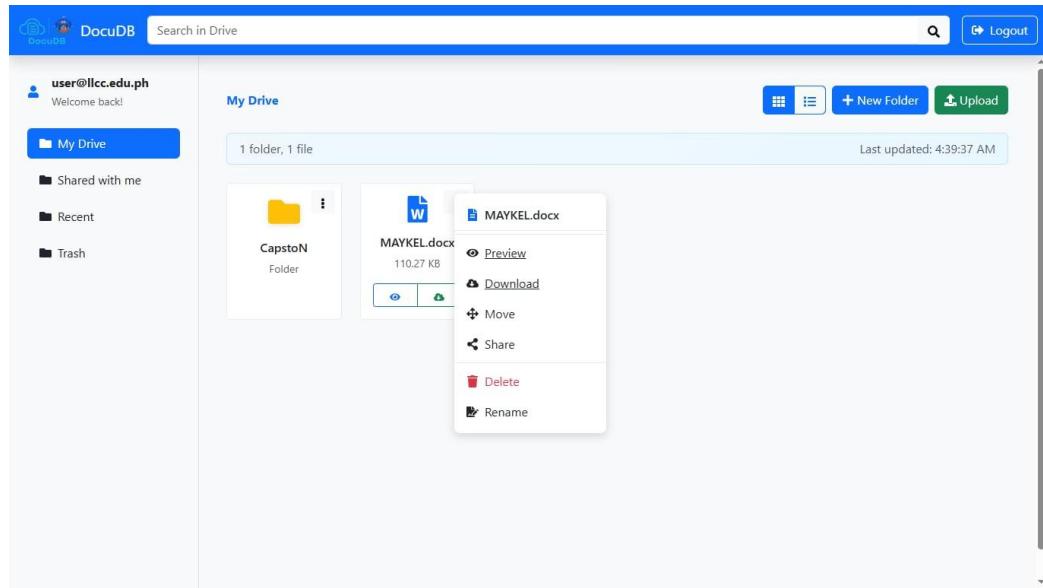
## 4.2 Creating a Folder

4. Click the **New Folder** button.
5. Enter the folder name and confirm.
6. Use folders to organize documents by topic or department.



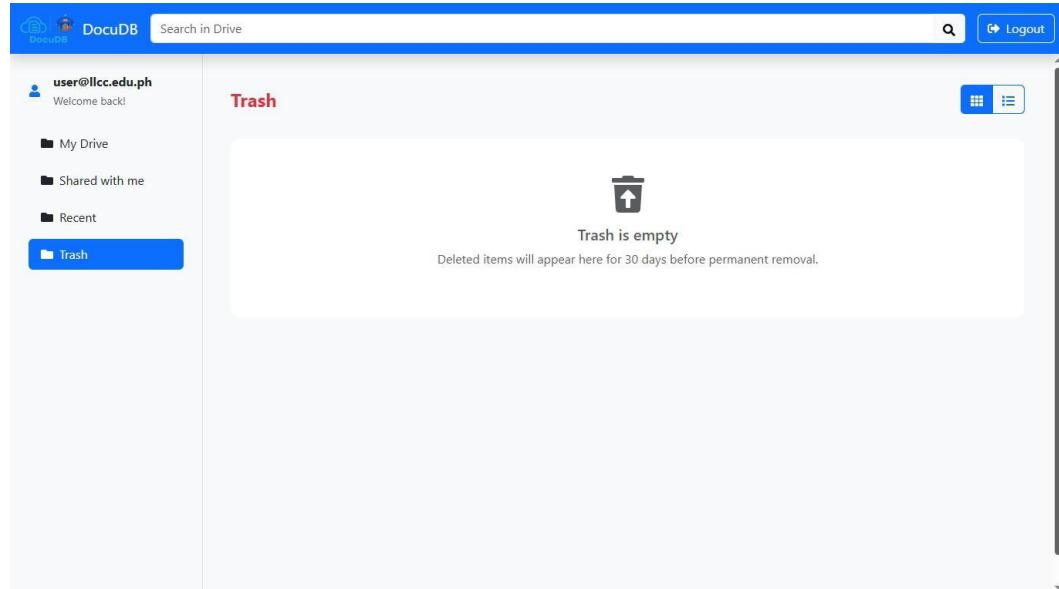
### 4.3 Moving, Renaming, or Deleting Files

- Hover over a file to see action buttons.
- You can:
  - Rename a file/folder.
  - Move it to another folder.
  - Delete it (moves to Trash).



#### 4.4 Restoring or Permanently Deleting Files

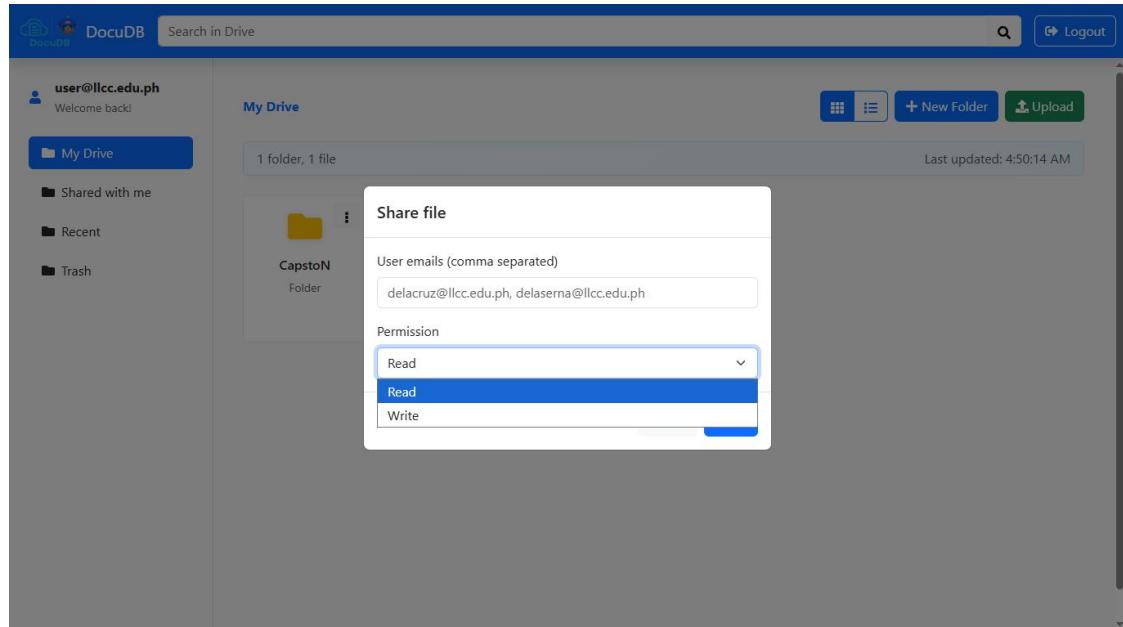
3. Go to **Trash** from the sidebar.
4. You can **Restore** files to their original location or **Permanently Delete** them.



## 5. SHARING AND COLLABORATION

## 5.1 Sharing a File

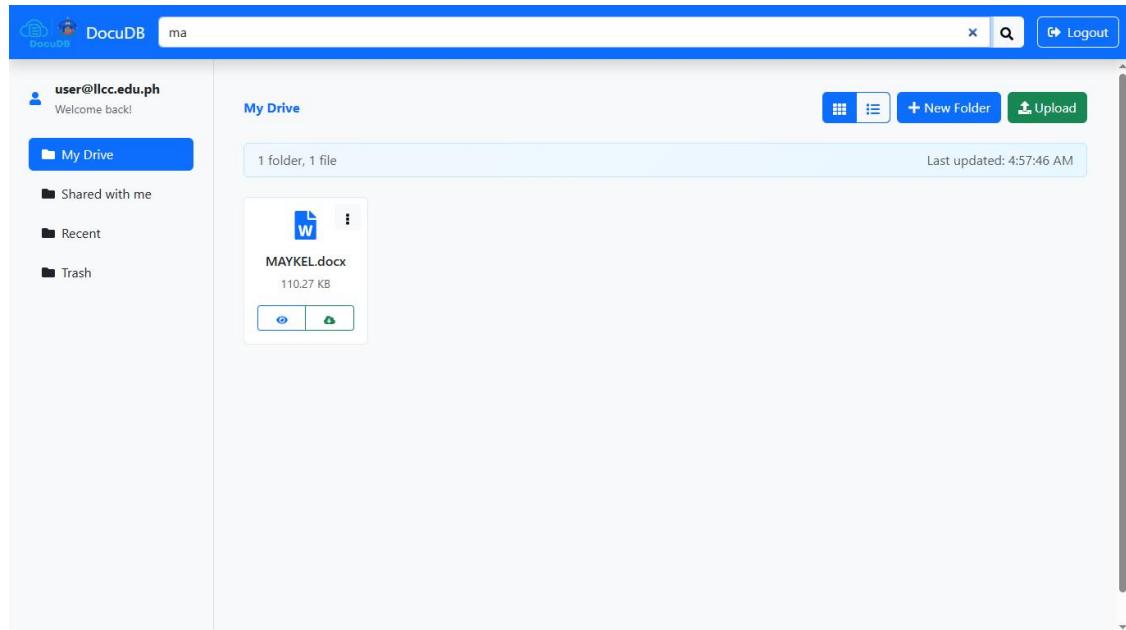
7. Select a file or folder.
8. Click the **Share** icon.
9. Enter the recipient's Lapu-Lapu City College email.
10. Choose permission:
  - **View Only**
  - **Edit Access**
11. Click **Share** to confirm.
12. All shared files will appear in the recipients' **Shared with Me** section.



## 6. SEARCHING DOCUMENTS

### 6.1 File Search

4. Type a keyword in the Search in **Navigation bar**.
5. Search results update in real time as you type.
6. You can search by File Name



## 7. LOGOUT

Click the **Logout** button on the upper-right corner of the page to safely exit the DocuDB system.

## Bibliography

- Alalwan, A. A., Baabdullah, A. M., Rana, N. P., Tamilmani, K., & Dwivedi, Y. K. (2018). Examining adoption of mobile internet in Saudi Arabia: Extending TAM with perceived enjoyment, innovativeness and trust. *Technology in Society*, 55(1), 100–110.  
<https://doi.org/10.1016/j.techsoc.2018.06.007>
- Bao, W. (2020). COVID-19 and online teaching in higher education: A case study of Peking University. *Human Behavior and Emerging Technologies*, 2(2), 113–115.  
<https://doi.org/10.1002/hbe2.191>
- CHED. (2020). CHED Memorandum Order No. 6, Series of 2020: Guidelines for the Implementation of Flexible Learning. Commission on Higher Education.
- Chen, L., Wang, Y., & Zhou, P. (2022). Text extraction and classification in digital document management systems using machine learning approaches. *Journal of Information Systems and Technology Management*, 19(1), 55–68.
- Chong, A. Y. L., Li, B., Ngai, E. W. T., Ch'ng, E., & Lee, F. (2020). Predicting online product sales via online reviews, sentiments, and promotion strategies: A big data architecture and neural network approach. *International Journal of Operations & Production Management*, 40(4), 356–377.
- Ekstedt, M. (2019). Real-time data systems and user accountability in collaborative digital environments. *Information Systems Research Journal*, 27(3), 210–225.
- Lin, Y., & Zhao, X. (2020). Metadata-driven document retrieval: Enhancing search efficiency in enterprise information systems. *Journal of Information Science*, 46(5), 678–692.

- Nabeel, M. (2024). Evaluating the efficiency of manual versus digital document storage systems in educational institutions. *Asian Journal of Educational Technology*, 8(2), 90–102.
- Patel, D. (2022). Assessing cloud-based document management platforms for SMEs: Usability and cost-effectiveness analysis. *International Journal of Business Information Systems*, 41(2), 155–170.
- Patil, A., & Sangle, S. (2020). Digital transformation through document management systems in higher education institutions. *Education and Information Technologies*, 25(6), 5087–5103.
- Raju, P., & Singh, K. (2020). Challenges and innovations in document management for public sector organizations. *International Journal of Digital Management*, 5(1), 33–41.
- Smith, J. (2021). The evolution of digital file systems: From manual archives to intelligent cloud-based management. *Information Technology Today*, 12(4), 45–53.
- Torres, C., Gutierrez, R., & Mendoza, L. (2021). Evaluating usability and limitations of commercial document management tools: A comparative study. *Journal of Computing and Information Science*, 17(2), 95–107.

## **Release Notes**

### **DocuDB: Automated tool for COPC of Lapu-lapu City college - Release Notes**

#### **Version 1.0.0 - Initial Release**

**Release Date: October 10, 2025**

#### **Introduction**

DocuDB is a web-based document management system developed using the MERN Stack (MongoDB, Express, React, Node.js). It is designed to provide Lapu-Lapu City College with a secure, centralized platform for document storage, sharing, and retrieval. This initial release focuses on streamlining document organization, improving accessibility, and ensuring data security through role-based controls and activity tracking.

The system serves both administrators and users, each with dedicated dashboards and permissions to ensure efficient management and collaboration across departments.

#### **Features**

##### **Working Features**

#### **1. User Authentication & Role Management**

- Secure login for all users (Lapu-Lapu City College domain only).
- Role-based access control (RBAC) to distinguish between Admin and User dashboards.
- Automatic redirection to role-specific dashboards upon login.

#### **2. File and Folder Management**

- Upload and organize files into folders for structured document storage.
- Supports multiple file formats: PDF, Word, Excel, and images.
- Move, rename, and delete files or folders.
- View files inline within the browser or download directly.
- Upload progress and validation messages during file submission.

#### **3. Document Sharing & Permissions**

- Share files or folders with individual users or groups within the organization.

- Assign permissions: View, Edit, or Full Access.
- Manage shared items from both sender and receiver sides.
- Real-time updates for shared file changes and access modifications.

#### **4. Search & Content Retrieval**

- Search by file name, file type, owner, or upload date.

#### **5. System Logs & Monitoring**

- Logs all critical activities such as uploads, deletions, moves, and sharing actions.
- Admins can view timestamped logs for auditing and accountability.
- Role-based viewing permissions to ensure data privacy.

#### **6. Trash Management**

- Soft deletion (move to Trash) before permanent deletion.
- Option to restore or permanently delete files/folders.
- Admin visibility over deleted content for monitoring purposes.

#### **7. Admin Dashboard & Analytics**

- Overview of total users, files, and folders in the system.
- Upload activity visualized through graphs (uploads per day).
- Summary of user actions (uploads, deletions, sharing, etc.).

#### **8. Security & Backup**

- Access restricted to college domain accounts.
- Role-based permission control for each operation.
- Regular database backup support and audit-ready logs.

## In Progress Features

- ❖ **Mobile Optimization** – Further UI responsiveness and file preview improvement on mobile devices.
- ❖ **Advanced Analytics** – Generate statistical summaries of uploads, active users, and file categories.
- ❖ **User Profile Management** – Enable profile updates (name, password, photo).
- ❖ **Automated Backup & Restore** – Scheduled data backup and restoration features.
- ❖ **Multi-Department Integration** – Support for department-level document grouping and access.

## Technical Details

- Frontend: React.js (Vite build)
- Backend: Node.js (Express.js)
- Database: MongoDB (Local Server)
- Styling: Bootstrap 5 & CSS
- File Handling: Multer for file uploads and directory management
- Authentication: Role-based session management
- Deployment Environment: Local Server (College Network)

## Known Issues

- Email notifications and automated backups are not yet implemented.
- The system currently operates only within the college network (no public cloud hosting).
- Text extraction for non-English or scanned PDF files may have limited accuracy.
- File previews for certain non-standard document types (e.g., legacy Office files) may not render inline.

## Limitations

- Restricted to authorized users with college-issued credentials.
- Does not yet integrate with third-party cloud services (e.g., Google Drive or OneDrive).
- Lacks offline functionality — internet/local network connection is required.
- Limited customization of user roles and permissions.

- No native mobile app (web-only access).

# **CURRICULUM VITAE**

# JHUNDE KYLE TAGHOY



## CONTACT

- +63 9913 147 243
- [Jkyletaghoy@gmail.com](mailto:Jkyletaghoy@gmail.com)
- Maribago, Lapu-Lapu City

## PERSONAL INFORMATION

Date of Birth: January 29, 2005

Place of Birth: Cebu City

Sex: Male

Citizenship: Filipino.

Civil Status: Single

Religion: Roman Catholic

## SKILLS

- Problem Solving
- Analyzing Skills
- Quality Skills
- Communication Skills
- Positive Attitude

## REFERENCE

Rita Debalucos (HR Manager)

*General Milling Corporation*

Renz Joseph Erojo (Quality Team Leader)

*Knowles Electronics Philippines*

Phone +63 943 6707 055

Email: [Renz.Erojo@Knowles.com](mailto:Renz.Erojo@Knowles.com)

Florencio Penicitos (Sr. Quality Engineer)

*Knowles Electronics Philippines*

Email: [Florencio.Penicitos@Knowles.com](mailto:Florencio.Penicitos@Knowles.com)

## EDUCATIONAL BACKGROUND

### TERTIARY 2023-PRESENT

Bachelor of Science and Industrial Technology -

Major in Computer Technology

Lapu-Lapu City College - Don B. Benedicto Rd.

Carajay, Gun-ob, Lapu-Lapu City, Cebu

### SECONDARY: SENIOR HIGH SCHOOL 2020-2021

University of Cebu Lapu-Lapu and Mandaue

Information and Communications Technology (CSS  
and Computer Programming)

### JUNIOR HIGH SCHOOL 2020-2021

University of Cebu Lapu-Lapu and Mandaue  
Information and Communications Technology (CSS  
and Computer Programming)

### ELEMENTARY 2014-2015

*Maribago Elementary School*

## AWARDS, HONORS AND HONORARY MENTIONS

- *Introduction to Cybersecurity (Cisco Networking Academy)*
- 27 April 2021
- *Get Connected (Cisco Networking Academy)*
- 29 April 2021
- *IEntrepreneurship (Cisco Networking Academy)*
- 26 April 2018
- *Summer Leadership Training (Babag National High School)*

# VAN CLARK G. APARTA



## CONTACT

+63 9670114353

supgvan@gmail.com

Datag Maribago Lapu-Lapu City

## PERSONAL INFORMATION

Date of Birth: September 19, 2002

Place of Birth: Lapu-Lapu City

Sex: Male

Citizenship: Filipino.

Civil Status: Single

Religion: Roman Catholic

## SKILLS

- Communication Skills
- Positive Attitude

## REFERENCE

Justine Rentyaya  
09277929506

Roz Polaris Abarico  
09325039123

Kitz Castillo  
09587692359

## EDUCATIONAL BACKGROUND

### TERTIARY 2023-PRESENT

Bachelor of Science and Industrial Technology -  
Major in Computer Technology  
Lapu-Lapu City College - Don B. Benedicto Rd.  
Carajay, Gun-ob, Lapu-Lapu City, Cebu

### SECONDARY: SENIOR HIGH SCHOOL 2022-2023

Einstein School Cebu Abuno, Pajac, Lapu-Lapu City,  
Philippines

### JUNIOR HIGH SCHOOL 2020-2021

Maribago High School  
Datag, Maribago Lapu-Lapu City

### ELEMENTARY 2016-2017

Maribago Elementary School  
Datag, Maribago Lapu-Lapu City

## AWARDS, HONORS AND HONORARY MENTIONS

- Elementary - With Honors
- High School - With Honors
- Senior High - With Honors

# MICHAEL EMMANUEL B. ABONDIENTE



## CONTACT

- 📞 +63 9913 147 243
- ✉️ michaelabondiente1@gmail.com
- 📍 Bout, Punta Enagño Lapu-Lapu City

## PERSONAL INFORMATION

Date of Birth: November 15, 2002

Place of Birth: Lapu-Lapu City

Sex: Male

Citizenship: Filipino.

Civil Status: Single

Religion: Roman Catholic

## SKILLS

- Problem Solving
- Analyzing Skills
- Quality Skills
- Communication Skills
- Positive Attitude

## REFERENCE

Julius Genon  
09842729506

Jayvee Reyes  
09325986437

Clyde Baring  
09589876432

## EDUCATIONAL BACKGROUND

### TERTIARY 2023-PRESENT

Bachelor of Science and Industrial Technology -  
Major in Computer Technology  
Lapu-Lapu City College - Don B. Benedicto Rd.  
Carajay, Gun-ob, Lapu-Lapu City, Cebu

### SECONDARY: SENIOR HIGH SCHOOL 2022-2023

Asian learning Center Pajo , Lapu-Lapu City,  
Philippines

### JUNIOR HIGH SCHOOL 2020-2021

Punta Engaño Highschool  
Punta Engaño Lapu-Lapu City

### ELEMENTARY 2015-2016

Tabunan Elementary School  
Borongan Eastern Samar

## AWARDS, HONORS AND HONORARY MENTIONS

- **Elementary** - With Honors
- **High School** - Best in Filipino
- **Senior High** - With Honors

# ALLEN JHON C. MONTECLAR



## CONTACT

+63 9929539141

michaelabondiente1@gmail.com

Park Place 2, Babag, Lapu-Lapu City,  
Cebu

## PERSONAL INFORMATION

Date of Birth: May 13, 2005

Place of Birth: Lapu-Lapu City

Sex: Male

Citizenship: Filipino.

Civil Status: Single

Religion: Roman Catholic

## SKILLS

- Problem Solving
- Analyzing Skills
- Quality Skills
- Communication Skills
- Positive Attitude

## REFERENCE

Geramie C. Monteclar

Email: geramiecaday@gmail.com  
+639324293703

Janrey L. Monteclar

Email: janreymonteclar@gmail.com  
+639475298876

Joan J. Dico

Email: joandico17@gmail.com  
+639940330952

## EDUCATIONAL BACKGROUND

### TERTIARY

#### 2023-PRESENT

Bachelor of Science and Industrial Technology -  
Major in Computer Technology  
Lapu-Lapu City College - Don B. Benedicto Rd.  
Carajay, Gun-ob, Lapu-Lapu City, Cebu

### SECONDARY:

#### SENIOR HIGH SCHOOL

#### 2022-2023

Babag National Senior Highschool  
Babag 1, Lapu-Lapu City, Cebu

### JUNIOR HIGH SCHOOL

#### 2020-2021

Babag National Highschool  
Babag 1, Lapu-Lapu City, Cebu

### ELEMENTARY

#### 2015-2016

saint Alphonsus Catholic School (SACS)  
(Dela Serna St, Poblacion, Lapu-Lapu City, Cebu

## AWARDS, HONORS AND HONORARY MENTIONS

- **High School** - With Honors
- **Senior High** - With Honors
- **College** - Dean's Lister (1<sup>st</sup> year)

# JOMILO ANDREW S. BARTIANA



## CONTACT

09762551440

asayson108@gmail.com

Beverly Place, Mactan Lapu-Lapu

## PERSONAL INFORMATION

Date of Birth: January 29, 2005

Place of Birth: Cebu City

Sex: Male

Citizenship: Filipino

Civil Status: Single

Religion: Roman Catholic

## SKILLS

- Problem-solving skills
- Organization and time management skills
- Communication Skills

## REFERENCE

Heart Jade Rumaguera

09923342541

Ivan Mark Casinillo

09621597048

## EDUCATIONAL BACKGROUND

### TERTIARY 2023-PRESENT

Bachelor of Science and Industrial Technology - Major in Computer Technology

Lapu-Lapu City College - Don B. Benedicto Rd.  
Carajay, Gun-ob, Lapu-Lapu City, Cebu

### SECONDARY: SENIOR HIGH SCHOOL 2022-2023

Benthel Asia School of Technology Inc.  
Sudtungan, Lapu-Lapu City, Philippines

### JUNIOR HIGH SCHOOL 2020-2021

Marigondon High School  
Marigondon Lapu-Lapu City

### ELEMENTARY 2016-2017

Marigondon Elementary School  
Marigondon Lapu-Lapu City

## AWARDS, HONORS AND HONORARY MENTIONS

- **Elementary** - Achiever
- **Senior High** - With Honors

## **SURVEY QUESTIONNAIRE FOR RESPONDENTS**

### **DocuDB: Automated tool for COPC of LApu-lapu City College**

**To the respondents:** Please answer each item honestly. Your responses will be treated with strict confidentiality. Put a check mark (✓) on your chosen answer for each statement.

Category Options / Description Name (Optional):\_\_\_\_\_

**Age:**  21–25     26–30     31–35     36–40     41 and above

**Sex:**  Male     Female

**Civil Status:**  Single     Married     Widowed     Separated

### **PART I - SYSTEM EFFECTIVENESS**

The following items assess how effectively the DocuDB system performs its intended tasks and meets user requirements.

#### **RATINGS:**

**1 - Not effective**

**2 - less effective**

**3 - effective**

**4 - Very effective**

NO	STATEMENT	VERY EFFECTIV E	EFFECTIV E (3)	LESS EFFECTIV E	NOT EFFECTIV E
.					

		(4)		(2)	(1)
<b>1</b>	DocuDB allows users to upload files and folder without errors or interruption				
<b>2</b>	The system accurately stores and retrieves uploaded documents				
<b>3</b>	Files and folders are organized and easy to access within the interface				
<b>4</b>	The system performs operations(upload, delete, move) efficiently and quickly				
<b>5</b>	Sharing permission(view/edit)function as intended without issue				
<b>6</b>	The system ensures that deleted files are moved to the trash and recoverable when needed				
<b>7</b>	The overall performance of DocuDB meets the expectations of its users				

## **PART II - SYSTEM USABILITY**

This section evaluates the system's ease of use, design intuitiveness and accessibility for users and administrator

### **RATINGS:**

**1 - Not user-friendly**

**2 - Less user-friendly**

**3 - User-Friendly**

**4 - Very user-friendly**

NO.	STATEMENT	VERY USER- FRIENDLY (4)	USER- FRIENDLY (3)	LESS USER- FRIENDLY (2)	NOT- USER FRIENDLY (1)

1	<p><b>The Interface of DocuDB is visual clear, organized and easy to navigate</b></p>				
2	<p><b>Icons, buttons, and labels are easy to understand and use correctly</b></p>				
3	<p><b>The dashboard provides a clear overview of users activities and documents</b></p>				

4	<p><b>The system's navigation (folders, breadcrumbs, search) is intuitive and logical</b></p>				
5	<p><b>The search bar and filters help users quickly locate documents.</b></p>				
6	<p><b>The system layout and text are readable and accessible across devices</b></p>				
7	<p><b>The system is easy to learn and requires minimal guidance for first-time user</b></p>				

## **PART III - System Functionality**

**This section assesses the technical and operational capabilities of DocuDB that contribute to secure and consistent functionality**

**RATINGS:**

**1 - Not Functional**

**2 - Partially Functional**

**3 - Functional**

**4 - Highly Functional**

NO.	STATEMENT	HIGHLY FUNCTIONAL (4)	FUNCTIONAL (3)	PARTIALLY FUNCTIONAL (2)	NOTFUNCTIONAL (1)

<b>1</b>	<b>User accounts and access roles are properly restricted (e.g, Admin/User)</b>				
<b>2</b>	<b>Shared documents and folders are only accessible to authorized users.</b>				
<b>3</b>	<b>The activity logs accurately record all user actions (uploads, deletion, edits and shares)</b>				
<b>4</b>	<b>The system consistently functions without crashes or data loss</b>				
<b>5</b>	<b>The system supports group and file-based search without errors</b>				
<b>6</b>	<b>The system maintains integrity and security of documents during upload/download</b>				
<b>7</b>	<b>The system can be updated or maintained without affecting stored data</b>				