APPLIED INFORMATION TECHNOLOGY AND COMPUTER SCIENCE
e-ISSN: 2773-5141

AITCS

Vol. 6 No. 2 (2025) 72-91
https://publisher.uthm.edu.my/periodicals/index.php/aitcs

DocuEase: Document Management System for MZR Global Sdn. Bhd.

Aleez Natasha Mohd Kameel[1], Shahreen Kasim[1]*

[1] *Fakulti Sains Komputer dan Teknologi Maklumat,*
Universiti Tun Hussein Onn Malaysia, Parit Raja, Batu Pahat, 86400, MALAYSIA

*Corresponding Author: shahreen@uthm.edu.my
DOI: https://doi.org/10.30880/aitcs.2025.06.02.005

Article Info

Received: 12 June 2025
Accepted: 3 November 2025
Available online: 30 November 2025

Keywords

Document Management System, Role-Based Access Control, Web Application, Laravel Framework, Agile Methodology, User Acceptance Testing, Digital Transformation, Administrative Efficiency

Abstract

DocuEase, a web-based document management system, was developed to address MZR Global Sdn Bhd's manual filing inefficiencies and security vulnerabilities. The system resolves challenges including data loss, disorganized documents, and time-consuming approval processes. Developed using Agile methodology with Laravel framework, DocuEase features role-based access control, automated approval workflows, secure document storage, and comprehensive audit logging. User Acceptance Testing with 24 participants showed satisfaction rates of 91.3% to 100% across evaluation criteria. The implementation successfully transformed manual processes into an efficient, secure digital solution, improving productivity and reducing administrative workloads.

1. Introduction

In the modern digital age, effective and safe document management is essential to the expansion and productivity of an organization. To manage a number of important documents, including official letters, meeting minutes, and company records. MZR Global Sdn Bhd still uses manual filing system. Although this conventional approach works, it has significant problems with operational efficiency security, and accessibility. Employees usually must wait a long time to retrieve documents, which makes it difficult to make decisions quickly and lowers productivity. The more documents there are, the more likely it is that important documents will be misfiled or lost, which causes inefficiencies.

Furthermore, the manual system's lack of appropriate access control mechanisms presents serious security issues. Unauthorized people could have access to private data, which could result in data breaches and violations of data protection laws. Accountability is restricted and compliance audits become more difficult when user activity and document modifications cannot be tracked. These flaws jeopardize MZR Global's ability to maintain operational integrity and compete in a business environment that is changing quickly.

MZR Global has given the creation of an automated document management system top priority after realizing the need for improvement the need for improvement. Role – based access control, improved data security, and streamlined documentation processes are the goals of the suggested solution, DocuEase. A centralized, safe, and intuitive platform for storing, retrieving, and organizing documents across departments will be made available by DocuEase.

The aim of this project is to develop DocuEase: Document Management System. The objectives of this project are:

- To design the DocuEase Document Management System that enables structured storage, fast retrieval, and systematic organization of departmental documents using a centralized digital platform.
- To develop a secure role-based access control module within the DocuEase system that limits document access to authorized personnel and enhance data protection and manage activity tracking.
- To test the functionality, performance, and usability of the DocuEase system through user acceptance testing (UAT) and system testing to ensure it meets operational requirements and provides a reliable user experience.

By implementing DocuEase, MZR Global is expected to significantly reduce administrative workload, increase operational efficiency, and create a secure and accountable environment and positioning the company for long-term success and scalability.

2. Related Work

This section examines the current document management systems, including Zoho WorkDrive, Dropbox, and DocuWare to evaluate their benefits and drawbacks. While both platforms facilitate cloud storage and collaboration, they differ in features such as audit logs, workflow automation and security.

2.1 Zoho WorkDrive

Zoho WorkDrive, a cloud-based document management and collaboration tool, facilitates file sharing, storage, and teamwork. Because it provides businesses with centralized storage, robust security features, and an easy-to - use interface, it is an excellent tool for efficiently managing documents [1].

Strengths: Collaborative team folder structure enabling real-time file collaboration, version control documenting all modifications with retrieval capabilities, integrated task management allowing direct work assignment and tracking, and document tagging with activity tracking features.

Limitations: Limited advanced workflow automation capabilities, insufficient compliance-focused features like comprehensive audit trails, and may not meet strict regulatory requirements for certain organizations [1].

2.2 Dropbox

A well-liked cloud-based file sharing and storage tool, Dropbox is renowned for being straightforward and user-friendly. Small and medium-sized business that require an effective method of file sharing and storage choose it. Dropbox is so user-friendly that even non-technical users can use it [2].

Strengths: Intuitive interface accessible to non-technical users, excellent third-party integration with Microsoft Office, Zoom, and Slack, seamless file sharing and document co-editing capabilities, and streamlined workflow optimization.

Limitations: Lacks advanced features like task reminders and workflow automation, limited document approval system capabilities, insufficient audit trail functionality for compliance requirements, and may not meet larger organization regulatory standards [2].

2.3 DocuWare

Enterprise-class document management software DocuWare is excellent at automating processes and adhering to regulations. Medium-sized to large businesses that handle a lot of private documents are the target audience. Businesses that need to streamline their documentation procedures and adhere to stringent regulatory standards will find DocuWare particularly useful [3].

Strengths: Strong security and compliance features including audit trails and encryption, advanced user authentication mechanisms, sophisticated search capabilities supported by metadata indexing, and comprehensive workflow automation for document processes.

Limitations: High implementation cost and complexity, requires extensive user training and specialized IT resources, complex implementation process, and pricing structure may be prohibitive for smaller organizations [3].

2.4 System Comparison

To evaluate DocuEase's capabilities, it is compared with other documents management systems like Dropbox, DocuWare, and Zoho WorkDrive as shown in Table 1. While these systems offer features like cloud storage and communication tools, they often lack key functionalities such as document approval workflows, detailed audit logs, and role-based access tailored to MZR Global Sdn Bhd's needs. The tale highlights the strengths and weaknesses of each system compares to DocuEase.

Table 1 *Comparison Between Existing Systems and Proposed System*

Features	Zoho WorkDrive	Dropbox	DocuWare	Proposed System
Cloud Storage	Yes	Yes	Yes	Yes
Document Approval	No	No	Yes	Yes
Audit Logs	No	No	Yes	Yes
Keyword-Based Search	Yes	Yes	Yes	Yes
User Monitoring	Yes	No	Yes	Yes
Cost-Effectiveness	Yes	Yes	No	Yes

3. Methodology

The Agile methodology was used was used in the development of the DocuEase system because of its adaptability, iterative structure, and emphasis on stakeholder involvement [4][5]. This approach ensures the system closely matches user needs by enabling frequent modifications based on evolving requirements and continuous feedback from MZR Global Sdn Bhd stakeholders.

Agile Selection Rationale:

- Iterative Development: Allows incremental feature delivery and continuous improvement.
- Stakeholder Engagement: Regular feedback sessions ensure alignment with business requirements.
- Risk Mitigation: Early identification and resolution of potential issues through short development cycles.
- Flexibility: Accommodates changing requirements during development.
- Quality Assurance: Continuous testing throughout development phases.

Agile methodologies have proven effective in project management due to their comprehensive approach to handling complex software development projects, particularly in dynamic business environments [5].

3.1 System Development Workflow

The following workflow highlights detailed tasks, outputs, and tools used in each development phase with expanded methodology explanations:

Table 2 *Systems Development Workflow*

Phases	Task	Output	Tool
Planning (Weeks 1-2)	Define comprehensive project scope, gather detailed requirements, and establish communication protocols	Requirements specification document, stakeholder needs documentation, risk assessment matrix	Google Meet, Stakeholders interviews
Analysis and Design (Weeks 3-6)	Develop system architecture, create database schema, implement security and framework design	System architecture documentation, database schema with ERD, UI wireframes, security specs	Laravel MVC framework [6], MySQL design
Development (Weeks 7-14)	Implement backend functionality using MVS, pattern and create responsive UI	Working system with integrated database responsive UI, security modules	Laravel [6][7], PHP, MySQL,
Testing (Weeks 15-17)	Perform comprehensive testing including functional, security and user acceptance	Test reports, bug documentation, user feedback	PHPUnit, UAT tools, manual testing
Deployment (Week 18)	System migration, database setup, server configuration	Live operational system, configured servers	Production server environment

Review and Launch (Week 19-20)	System evaluation, user training, performance assessment	Evaluation reports and training materials	Training materials

3.2 Technology Stack Justification

Backend Framework: Laravel was selected for its robust MVC architecture, built-in security features, and comprehensive ORM capabilities, providing rapid development while ensuring code maintainability and security [6][7].

Database: MySQL chosen for its reliability, performance, and compatibility with Laravel's Eloquent ORM, supporting complex relational data structures required for document management.

Frontend: Bootstrap framework selected for responsive design capabilities and rapid UI development, ensuring cross-platform compatibility and user-friendly interfaces.

Security Implementation: Laravel's built-in security features including CSRF protection, authentication, and authorization mechanisms provide comprehensive security coverage.

3.3 System Requirement Analysis

The DocuEase system requirements are divided into functional and non-functional categories as shown in Table 3 and Table 4, based on discussion with MZR Global Sdn Bhd stakeholders. The system incorporates role-based access control mechanisms to ensure secure document management [9].

Table 3 *Functional Requirements*

Module	Description
User Registration and Login	Enables users (Admin, Manager, File Admin, Staff) to securely register and log in using role-based access with multi-factor authentication support [9]
User Management	Allows system admins to manage user roles, departments, account details, and access permissions with comprehensive audit trails
Document Upload and Storage	Allows users to upload, organize, and store documents in assigned folders with proper version control and metadata management [8]
Document Approval	Supports role-based document approval flow by Manager and File Admin with configurable approval hierarchies and automated notifications.
Email Notification	Sends automated emails to approvers or uploaders for document status updates.
Document Search	Enables users to search, filter, and retrieve documents efficiently.
Activity Logs	Tracks and displays user activity such as uploads, views, updates, and deletions with secure audit management [10]
File Preview/Download	Provides access to view or download files linked to a log or archive.

Table 4 *Non-Functional Requirements*

Category	Requirements
Performance	Fast system response and document access.
Security	Data protection and enhanced authentication.
Usability	User-friendly interface and navigation.
Reliability	System backup and error management.
Scalability	Expandable storage and consistent performance.

3.4 System Analysis

The DocuEase Document Management System for MZR Global Sdn Bhd is designed as a secure, web-based platform to streamline the management of organizational documents and workflows. The system follows a modular approach, integrating various components such as document storage, approval processes, access control, and audit tracking. This design ensures that each module is independent yet works cohesively to enhance system security, efficiency and user experience.

3.4.1 Flowchart

The System Administration manages the overall system operations through a secure login process. Figure 1 shows how after successful authentication, the administrator can choose to add new user accounts or proceed to manage existing user account information. Their main task includes creating new user accounts, managing user permissions and department, and configuring system settings before logging out. This role ensures proper access control and system security for MZR Global Sdn Bhd.

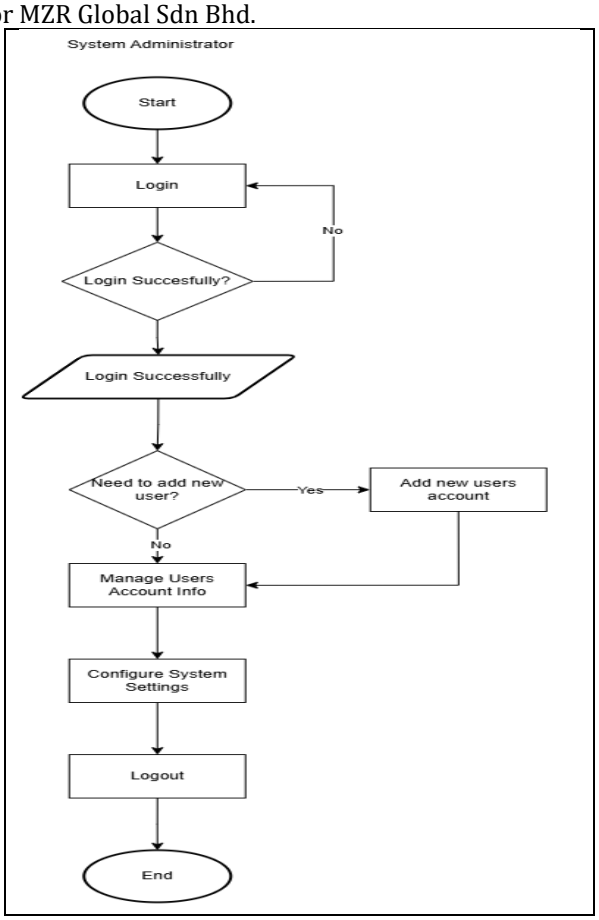


Fig. 1 *System Administrator Workflow Flowchart*

The DocuEase workflow integrates three essential user roles in the document management process. Staff members handle document uploads, tracking, and revisions through their dedicated workflow, while File Administrators review these documents and route them to Managers for final decisions. Figure 2 shows how the Manager workflow depicts the final approval process where managers can either approve or reject documents, completing the workflow chain. This structured process ensures secure and efficient document handling through proper authentication and approval steps, meeting MZR Global Sdn Bhd's requirements for comprehensive document management.

Applied Information Technology and Computer Science Vol. 6 No. 2 (2025) p. 72-91

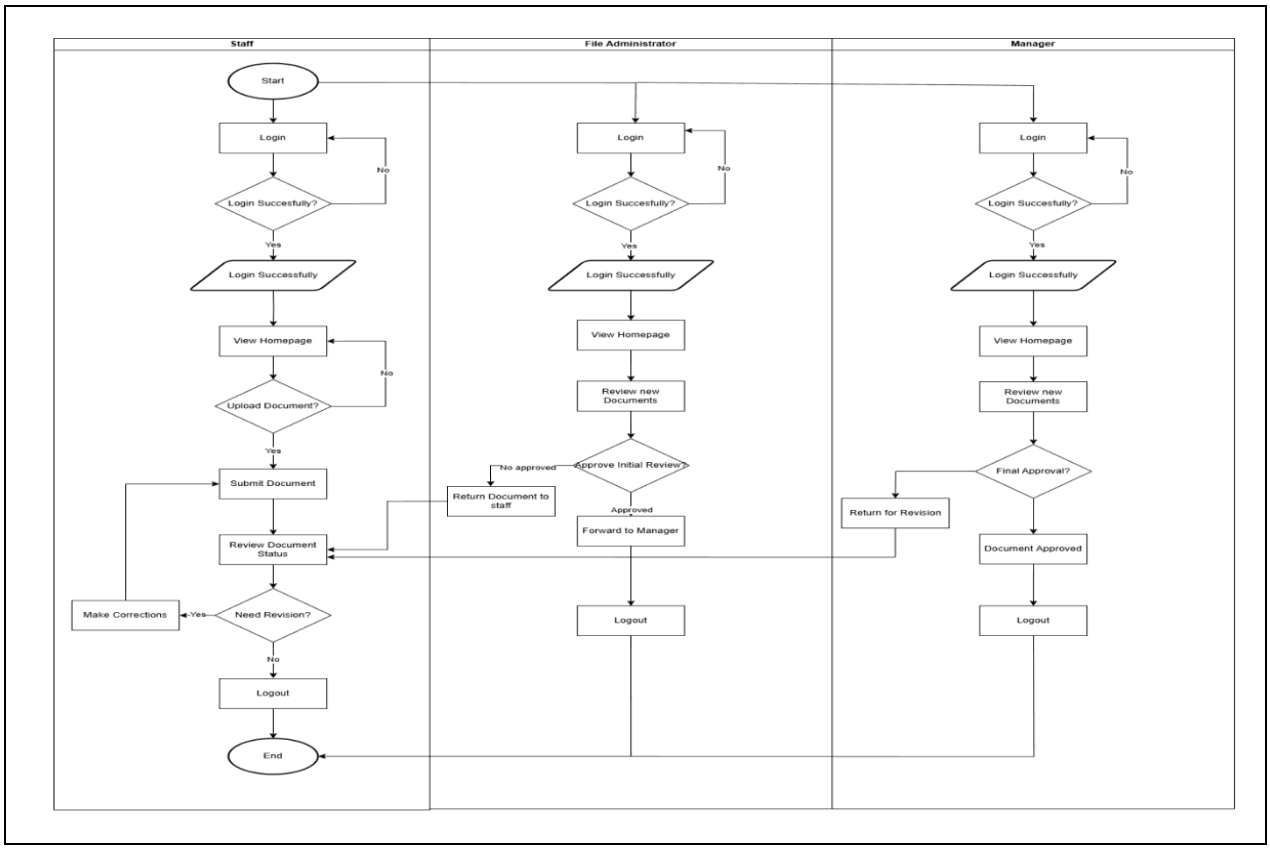


Fig. 2 *Document Management Workflow Flowchart*

3.4.2 Context Diagram

Figure 3 shows DocuEase's Context Diagram, which displays how the system interacts with its four main users: System Admin, File Admin, Manager and Staff. This diagram illustrates how information and document flow between these users and the system.

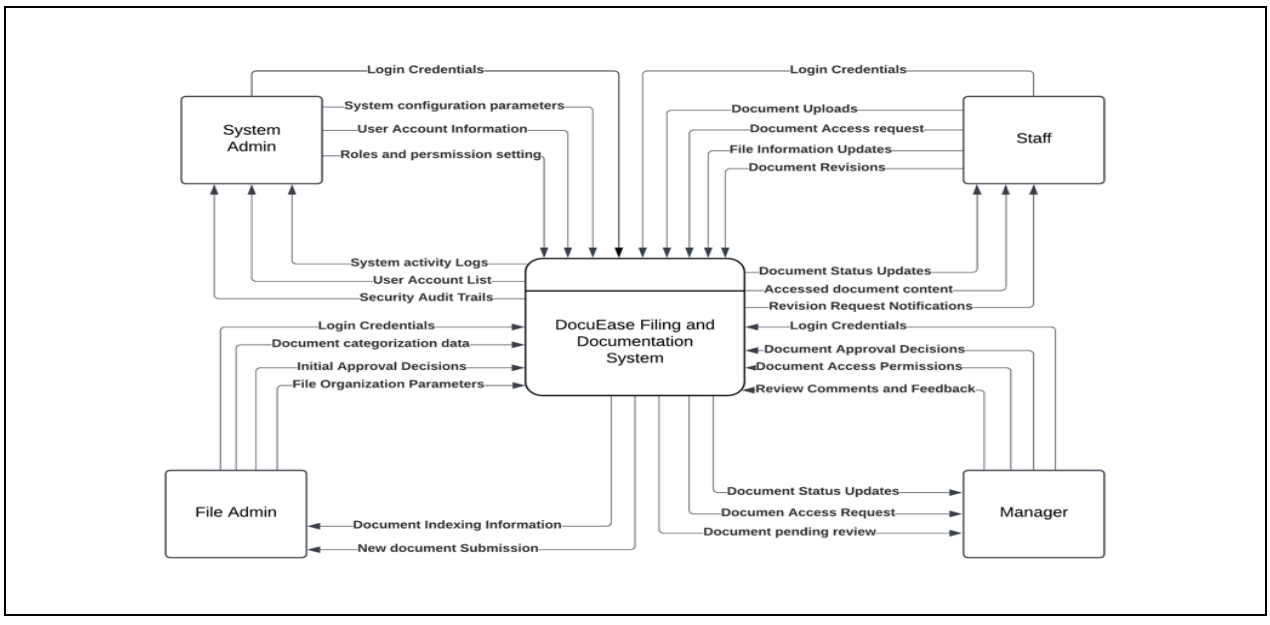


Fig. 3 *Context Diagram of DocuEase: Document Management System*

3.4.3 Data Flow Diagram (DFD)

Figure 4 shows DocuEase's Level 0 Data Flow Diagram (DFD), showing four main processes (User Management, Document Processing, Approval Workflow, and System Monitoring) working with four data stores (User Data Store, Document Store, Document Logs, and File Sharing). This structure enables smooth interaction between system users (System Admin, File Admin, Manager, Staff) and document management functions through secure role-based access control and comprehensive audit tracking.

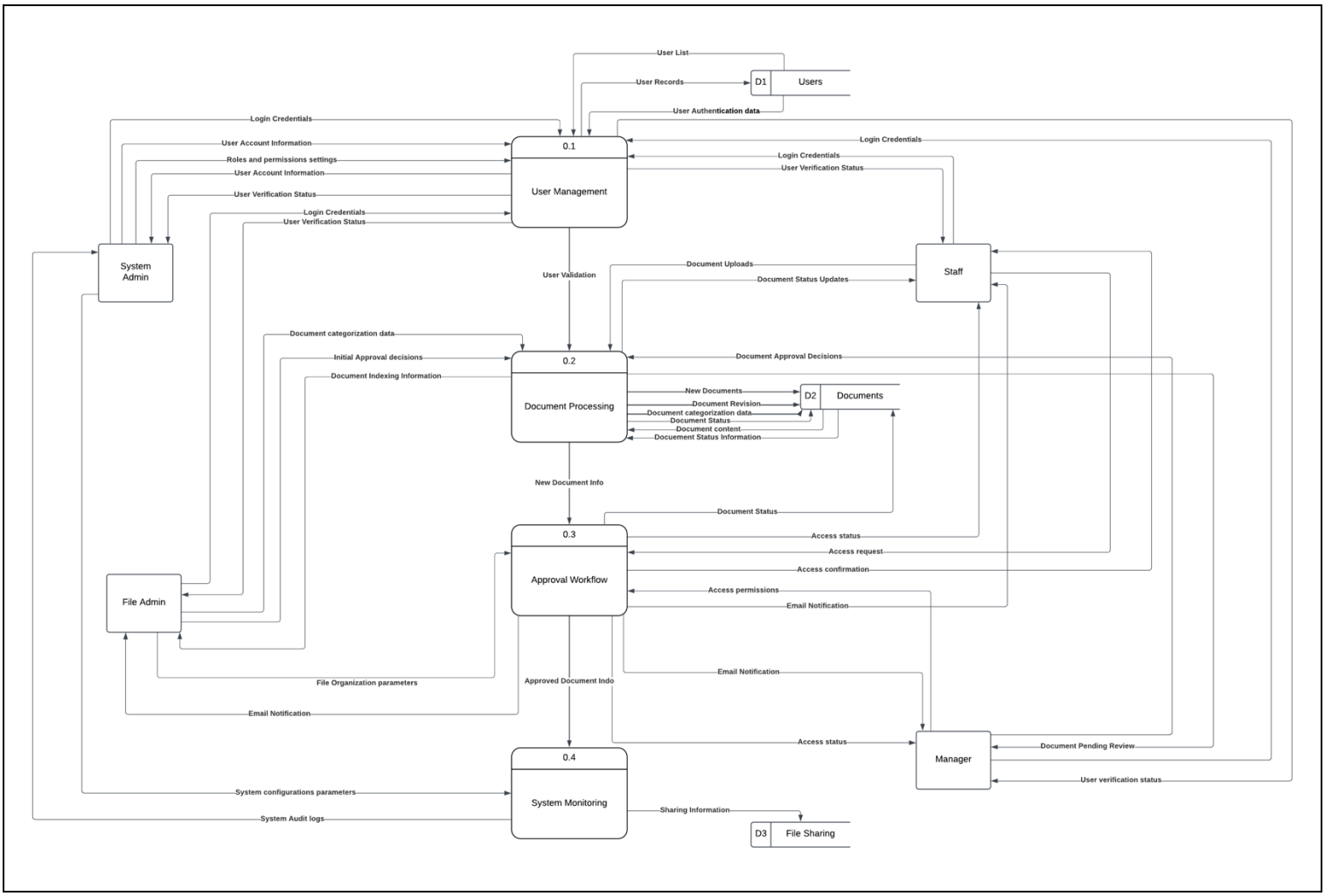


Fig. 4 *DFD Level 0 of DocuEase: Document Management System*

3.4.4 Entity-Relationship Diagram (ERD)

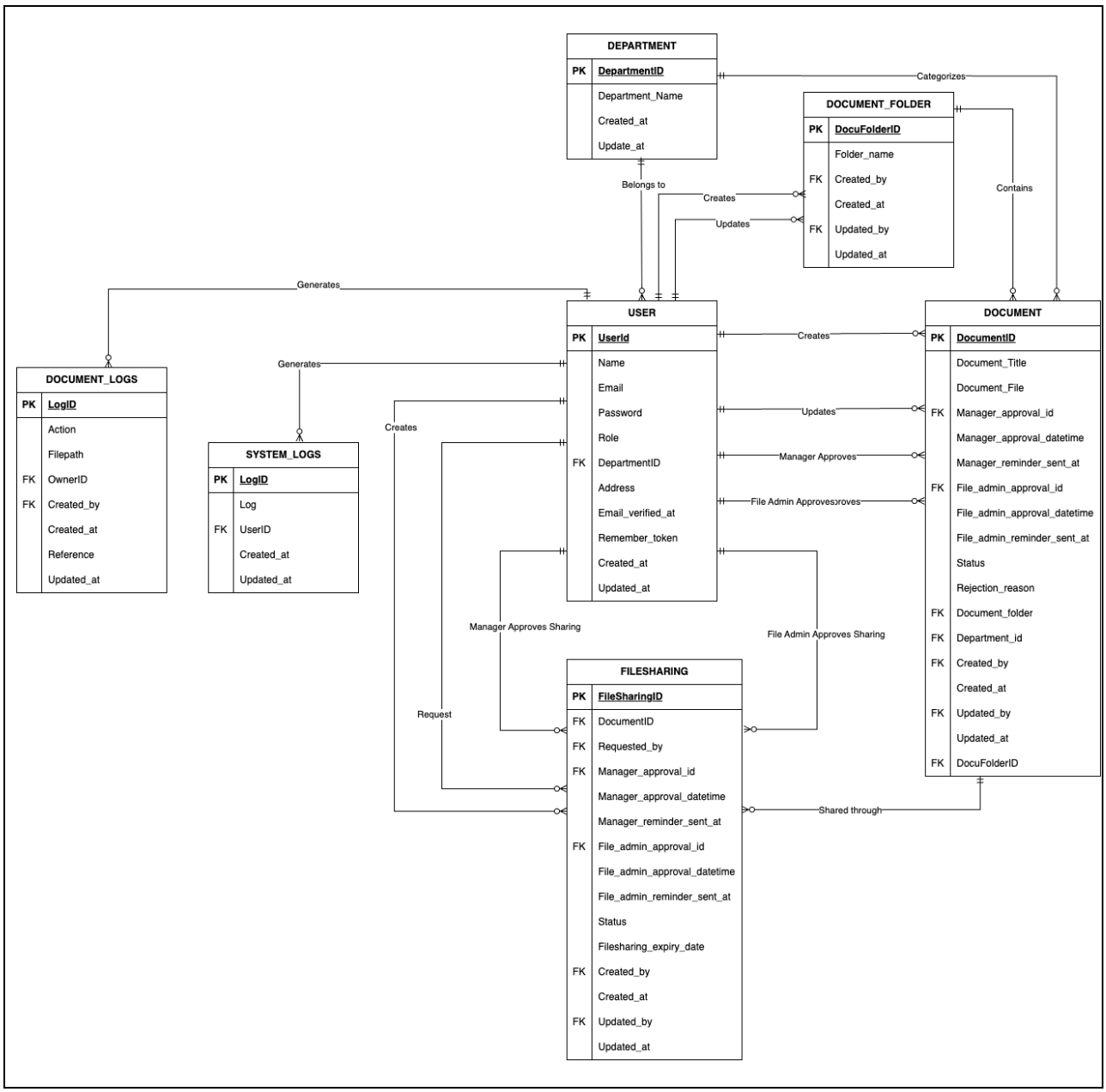


Fig. 5 *Entity Relationship Diagram of DocuEase: Document Management System*

Figure 5 illustrates the DocuEase ERD, showing eight tables: USERS, DEPARTMENTS, DOCUMENTS, DOCUMENT_FOLDER, FILESHARING, DOCUMENT_LOGS, SYSTEM_LOGS, and ACCESS_REQUEST. The diagram highlights their relationships for managing document flow, access control, and organizational structure through primary and foreign key connections that ensure data integrity and support role-based document management operations.

4. Result and Discussion

This section focuses on the implementation and testing of the DocuEase document management system for MZR Global Sdn Bhd. It covers the development and functionality of key modules such as secure login, user registration, role-based access control, document uploading, document approval, and activity tracking. The results are

discussed based on system performance, user interaction and how effectively the system addresses the challenges of the previous manual filing process.

4.1 Implementation and Interface Design

This subsection describes the core functional modules in the DocuEase system and the design of its user interface. The implementation focused on creating a secure, efficient, and role-based document management experience tailored to the operational needs of MZR Global Sdn Bhd. The system was developed using the Laravel framework, which provides robust web application development capabilities [6][7].

DocuEase's interface design was created with usability and clarity in mind, guaranteeing that users can easily navigate the system. To create a simple and responsive layout, it combines concepts from information architecture and current UI styling. The system implements an interactive documentation approach that enhances user experience [8]. To complete tasks like login, registration and role-based document access, users engage with user-friendly elements like dropdown menus, input fields, and modals.

4.1.1 Registration, Login and Role-Based Access Interface

The DocuEase system provides a secure login and registration process with integrated role selection. As shown in Figure 6 and Figure 7, user must login by selecting their designated role (System Admin, File Admin, Manager, or Staff), followed by entering their email and password. This role-based login ensures that each user is granted access only to features relevant to their responsibilities.

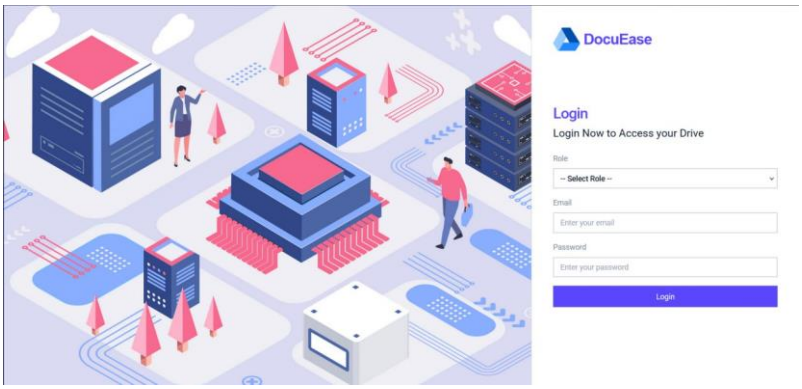


Fig. 6 *Login Interface*

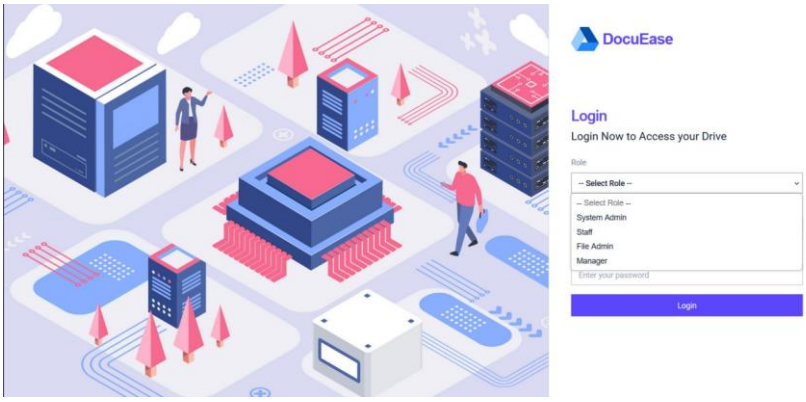


Fig. 7 *Role-Based Login Interface*

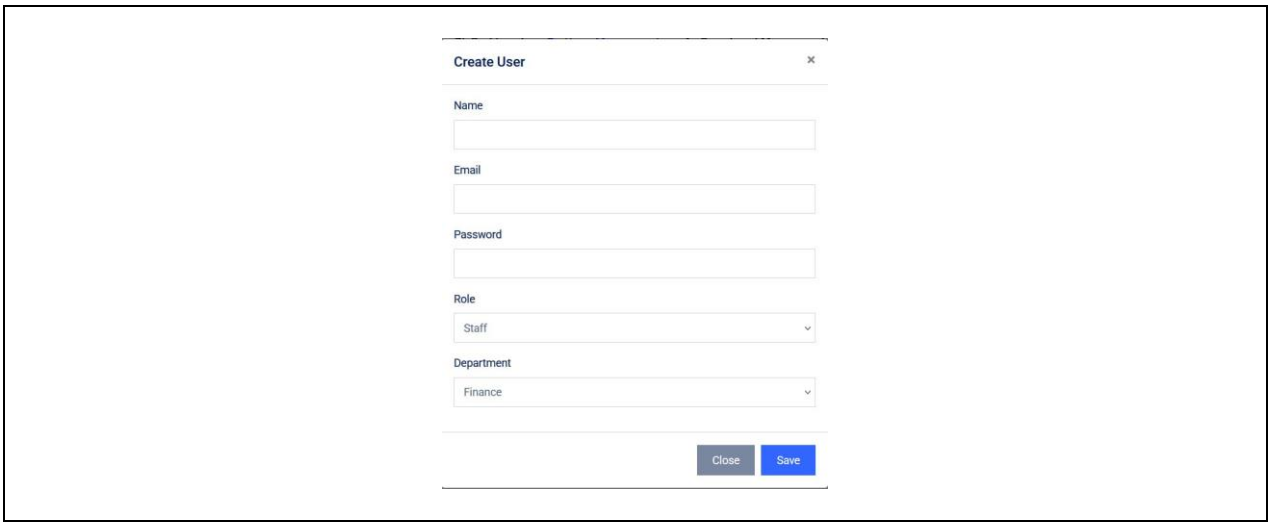


Fig. 8 *User Registration Interface*

After login, user creation or registration is managed through a form interface, as shown in Figure 8. The administrator can register a new user by entering essential details such as name, email, password, role and department. This interface simplifies onboarding while maintaining strict control over user roles and departmental assignment.

These features contribute to DocuEase's goals of offering a secure, efficient, and organized document management experience tailored to each user's access level

4.1.2 Document Upload Function

The Document upload function is a core feature of the DocuEase system, allowing users to upload files securely while capturing relevant metadata such as document title and file type. This functionality ensures that important records are organized and ready for approval workflows by authorized roles such as Managers and File Admins.

```
<div class="modal fade" id="uploadModal">
<div class="modal-dialog">
  <div class="modal-content">
    <div class="modal-header">
      <h5 class="modal-title">Upload File</h5>
      <button class="close" data-dismiss="modal">&times;</button>
    </div>
    <div class="modal-body">
      <form id="uploadForm">
        @csrf
        <input type="text" class="form-control mb-2" name="document_title" placeholder="Document Title" required>
        <input type="file" class="form-control" name="document_file" required>
      </form>
    </div>
    <div class="modal-footer">
      <button class="btn btn-secondary" data-dismiss="modal">Cancel</button>
      <button class="btn btn-primary" id="submitUpload">Upload</button>
    </div>
  </div>
</div>
</div>
```

Fig. 9 *HTML structure for the Document Upload Modal*

The frontend implementation, shown in Figure 9, uses Bootstrap modal components to create a user-friendly interface. Users are prompted to enter the document title and upload file through a form *#uploadForm.* The *@csrf* directive ensures that the submission is protected from cross-site request forgery. Two input fields are present: one for the document title *type="text"* and one for the actual file *type="file".* The "Upload" button triggers the form submission, while the "Cancel" button closes the modal without saving.

```php
protected $fillable = [
    'document_title',
    'document_file',
    'manager_approval_id',
    'manager_approval_datetime',
    'file_admin_approval_id',
    'file_admin_approval_datetime',
    'status',
    'document_folder',
    'dept_id',
    'created_by',
    'updated_by',
];
```

Fig. 10 *Laravel Fillable Fields in Document Model*

The backend logic for handling document data is defined in the Laravel modal class using the *$fillable* property, as shown in Figure 10. This array specifies which fields can be mass-assigned when a document is uploaded. Fields like *document_title, document_file, document_folder, dept_id,* and *timestamps* for manager and file admin approvals are included. This ensures that document-related data, including approval tracking and department categorization, is stored accurately in the database. Together, these implementations support a structured document submission process, ensuring every uploaded file is accompanied by the required metadata and stored securely for review and approval.

4.1.3 Document Approval and Rejection Function

The Document Approval and Rejection functions in the DocuEase system are responsible for handling document verification based on user roles. Only users with the roles Manager and File Admin can approve or reject documents, and both actions are recorded with timestamps for transparency and auditing.

```php
public function approve(Request $request, $id): JsonResponse|mixed
{
    $document = Document::findOrFail(id: $id);
    $user = auth()->user();

    if ($user->dept_id !== $document->dept_id) {
        return response()->json(data: ['message' => 'You must be in the same department to approve.'], status: 403);
    }

    if ($user->role == 'Manager') {
        $document->manager_approval_id = $user->id;
        $document->manager_approval_datetime = now();
    } elseif ($user->role == 'File Admin') {
        if (!$document->manager_approval_id) {
            return response()->json(data: ['message' => 'Manager approval required first.'], status: 403);
        }
        $document->file_admin_approval_id = $user->id;
        $document->file_admin_approval_datetime = now();
        $document->status = 'Approved';
    }

    $document->save();


    if ($document->status === 'Approved') {
        try {
            Mail::to(users: $document->creator->email)
                ->send(mailable: new DocumentApproved(document: $document, approverName: $user->name));
        } catch (\Exception $e) {
            Log::error(message: "Approval email failed: ".$e->getMessage());
        }
    }

    return response()->json(data: ['message' => 'Document approved successfully.']);
}
```

Fig. 11 *Approving Uploaded Documents Function*

In Figure 11, the *approve()* function first checks whether the logged-in user belongs to the same department as the document originator. If not, the approval process is blocked. When a Manager approves a document, their user ID and approval timestamps are stored in the document record. When a File Admin attempts approval, the system verifies that manager approval exists. If approved, the system updates the file admin's ID and timestamp, sets the document status as "Approved", and sends an email notification to the document creator.

```
    public function reject(Request $request, $id): JsonResponse|mixed
    {
        $document = Document::findOrFail(id: $id);
        $user = auth()->user();

        if ($user->dept_id !== $document->dept_id) {
            return response()->json(data: ['message' => 'You must be in the same department to approve.'], status: 403);
        }

        if ($user->role == 'Manager') {
            $document->manager_approval_id = $user->id;
            $document->manager_approval_datetime = now();
            $document->status = 'Rejected';
        } elseif ($user->role == 'File Admin') {
            $document->status = 'Rejected';
            $document->file_admin_approval_id = $user->id;
            $document->file_admin_approval_datetime = now();
        }

        $document->save();

        try {
            Mail::to(users: $document->creator->email)
                ->send(mailable: new DocumentRejected(document: $document, rejecterName: $user->name));
        } catch (\Exception $e) {
            Log::error(message: "Rejection email failed: ".$e->getMessage());
        }

        return response()->json(data: ['message' => 'Document rejected successfully.']);
    }

}
```

Fig. 12 *Rejecting Uploaded Documents Function*

In Figure 12, the *reject()* function performs a similar role but handles rejection instead. Again, role and department check are enforced. It a Manager or File Admin rejects the document, their details and timestamps are logged, and the document status is set to "Rejected". Upon rejection, and email is automatically sent to inform the original uploader, as shown in the *Mail::to()->send()* block. Error handling is included to log any failures in email delivery.

4.1.4 Email Notification Function

The email notification function in the DocuEase system is designed to alert relevant users when a new document is uploaded and pending approval. This feature enhances communication between staff and approvers while reducing delays in the document approval process. The electronic documentation system improves the quality and efficiency of document management processes [11].

```
    public function build(): DocumentUploaded
    {
        return $this
            ->subject(subject: "New Document Pending Approval: {$this->document->document_title}")
            ->view(view: 'emails.document_uploaded')
            ->with(key: [
                'title'      => $this->document->document_title,
                'uploadedBy' => $this->document->creator->name,
                'uploadedAt' => $this->document->created_at->format('M d, Y H:i')
            ]);
    }
}
```

Fig. 13 *Mailable build() method for constructing document approval email*

The feature is implemented using Laravel's Mailable class, where the *build()* method constructs the content of the email. As shown in Figure 13, the subject line is dynamically set based on the uploaded document title (e.g., "New Document Pending Approval: Budget Report"). The *view()* method specifies the Blade template for formatting the email layout, while the *with()* method passes the document title, the name of the user who uploaded it, and the timestamp of upload to the view. These variables are injected into the email to give the recipient clear, context-rich information for review and action.

Fig. 14 *SMTP configurations for email delivery*

In addition to the logic for generating the email, the backend mail delivery is configured using the SMTP mailer setup, as shown in Figure 14. SMTP credentials such as *MAIL_HOST, MAIL_PORT, MAIL_USERNAME, and MAIL_PASSWORD* are stored securely in the *.env* file and accessed via *env()* in the *config/mail.php* file. The encryption method is set to *tls*, ensuring secure transmission of messages. This configuration allows DocuEase to send automated emails using external email services such as Mailgun, ensuring that all notification emails are delivered reliably and securely.

4.1.5 Activity Log Function

The Activity Log function in the DocuEase system provides a detailed record of all user actions performed on documents and folders. This feature is essential for maintaining transparency, accountability, and traceability within the system, especially in environments where document integrity and user access must be strictly monitored. The implementation follows secure audit log management principles to ensure data integrity and compliance [10].

Fig. 15 *User Activity log function*

In Figure 15, the activity log entries are displayed using a *foreach* loop that goes through the *$document-logs* list. The system shows only the logs that belong to the currently logged-in user by checking if the *owner_id* matches the user's ID. Each log action (like "UPLOAD FILE", "DELETE FILE", "RENAME FOLDER", etc.) is shown with a

colored badge using the *$badgeClass* array to make it easier to recognize the type of action. Alongside the action, the log also shows who performed it, a reference if available, and the time it happened.

If the action was done by someone else, the system gets the creator's name from the user's table. If the log was created by the user themselves, it simply shows "You". The date and time are displayed using Laravel's Carbon library in a clear, readable format. If the log includes a file path, a "View File" button is shown. Clicking this button will open the related file in a new tab so the user can view it directly.

4.2 Testing Result

The testing process focused on verifying the core functional modules within the DocuEase document management system. Each test case was designed to validate the successful execution of features such as registration, login, role-based access, document uploading, approval workflows, activity logging, and email notification. The goal was to ensure that every module behaves as expected under normal usage conditions. Table 5 shows the result of functionality testing.

Table 5 *Functionality Testing*

No	Test Case	Expected Result	Actual Result
1	Login with correct role, email, and password	Login successful	Pass
2	Create new user with assigned role and department	Role and department saved	Pass
3	Upload a document with valid title and file	Document uploaded	Pass
4	Manager approves uploaded document	Status updated to "Pending File Admin"	Pass
5	File Admin approves the document	Status updated to "Approved"	Pass
6	Reject document as Manager or File Admin	Status updated to "Rejected"	Pass
7	View activity log after actions	Log displayed with correct action	Pass
8	Trigger email after document upload	Email sent to approver	Pass
9	Access view file link from activity log	File preview opened	Pass

The testing confirms that the main features of DocuEase perform according to system's functional requirements. The result demonstrates the system's reliability in handling document processes, access control, communication, and user interaction. Overall, the successful completion of these test supports the system's readiness for deployment in real-world environment.

Table 3 presents the result of security testing conducted on the DocuEase system. The test case focus on validating input fields, enforcing authentication, protecting against unauthorized access, and verifying role-based restrictions. All test cases passed successfully, confirming that the system enforces proper validation, access control, and secure handling of requests. This ensures that DocuEase provides a safe and reliable environment for document management.

Table 6 *Security Testing*

No	Test Case	Expected Result	Actual Result
1	Login with correct role, email, and password	Error message displayed. Login blocked	Pass
2	Login with incorrect email or password	Invalid credentials message displayed. Login failed	Pass

Table 6 *Cont*

3	Register with missing or incomplete fields	Required field warning displayed. Registration blocked	Pass
4	Register with invalid email format	Warning: invalid email format. Registration blocked	Pass
5	Attempt to access admin routes without authentication	Redirected to login page	Pass
6	Submit upload form without CSRF token	Request blocked	Pass
7	Attempt document approval from a different department	Approval blocked. Error message shown	Pass
8	Send forged document approval request via POST API	Request denied due to user role mismatch	Pass

Table 3 confirms the system's effectiveness in handling common security concerns through proper validation, access control, and session handling. These tests demonstrate that DocuEase has implemented key security measures such CSRF protection, form validation, role-based restrictions, and error handling. As a result, the system is well-prepared to defend against unauthorized access and data manipulation.

4.2.1 User Acceptance Testing Form

User Acceptance Testing (UAT) for the DocuEase system was conducted through an online questionnaire distributed using Google Forms. The objective was to evaluate the system's usability, functionality, and interface form the perspective of actual users. The main form included both quantitative and qualitative sections and was divided into four main parts: system functionality, system usability, interface design, and user suggestions. Respondents were asked to rate each item to rate each item on a scale of 1 (Strongly Disagree) to 5 (Strongly Agree).

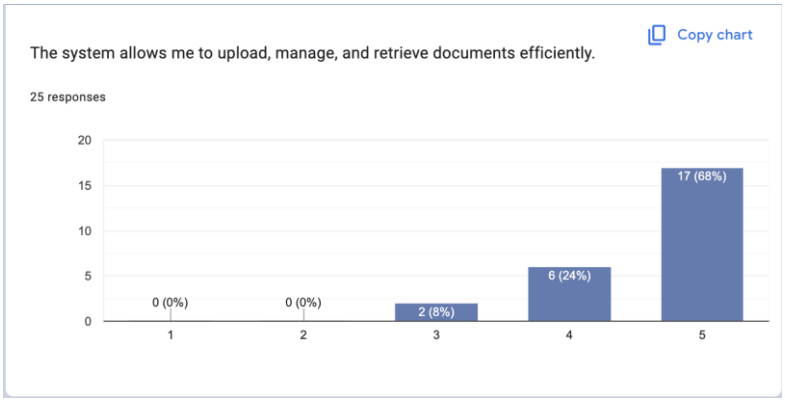


Fig. 16 *Document Management Efficiency Survey*

Figure 16 shows the survey results for the questions "The system allows me to upload, manage, and retrieve documents efficiently". Out of 24 responses, the survey reveals that 0 respondents rated the system with scores of 1 and 2, 2 respondents (8.3%) provided a rating of 3 (neutral), 5 respondents (20.8%) rated the system with a score of 4 (agree), and 17 respondents (70.8%) rated the system with a score of 5 (strongly agree). The results indicate an exceptionally positive reception of the document management functionality, with 91.6% of users expressing satisfaction (ratings 4-5) with the system's efficiency in handling document operations.

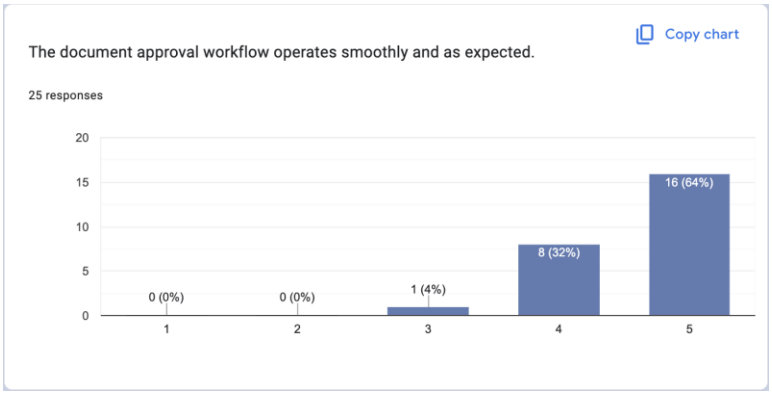


Fig. 17 *Document Approval Workflow Survey*

Figure 17 presents the survey results for "The document approval workflow operates smoothly and as expected." Out of 24 responses, 0 users assigned ratings of 1, 2, and 3, while 8 users (33.3%) provided a rating of 4, and 16 users (66.7%) rated the workflow with a score of 5. The overall evaluation of the approval workflow is exceptionally positive, with 100% of respondents (ratings 4-5) indicating satisfaction with the workflow functionality. This suggests that the role-based approval process effectively meets user expectations and operational requirements without any negative feedback.

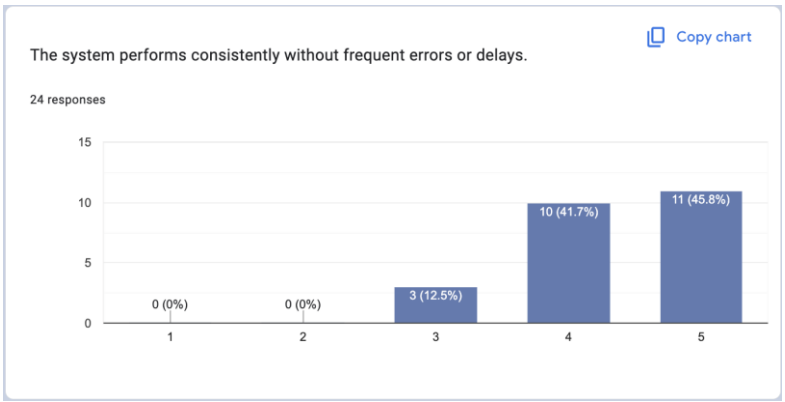


Fig. 19 *Document Approval Workflow Survey*

Figure 19 illustrates the survey results for "The system performs consistently without frequent errors or delays." Out of 23 responses, 0 respondents rated with scores of 1 and 2, 2 respondents (8.7%) provided a rating of 3, 10 respondents (43.5%) rated with a score of 4, and 11 respondents (47.8%) rated with a score of 5. The results demonstrate strong system reliability, with 91.3% of users (ratings 4-5) expressing confidence in the system's consistent performance. This positive feedback confirms the system's stability and readiness for production deployment.

5. Conclusion

In conclusion, MZR Global Sdn Bhd's manual filing system's inefficiencies and security issues were effectively resolved with the creation of the DocuEase Document Management System. User accountability and operational efficiency have been significantly increased by the use of role-based access control [9], secure document handling, approval workflows, audit logs [10], and automated notifications. The implementation leveraged modern web development frameworks [6][7] and followed comprehensive agile project management methodologies [5] to ensure successful delivery.

The electronic documentation system has demonstrated improved quality and efficiency compared to traditional manual systems [8][11]. User acceptance surveys and functional and security testing verify that the system meets organizational requirements effectively while providing consistent performance and a user-friendly interface.

Acknowledgement

The successful completion of this project would not have been possible without the continuous guidance and support of PM. Dr. Shahreen Binti Kasim, whose expertise and encouragement greatly contributed to the quality of this research. I would also like to extend my heartfelt appreciation to MZR Global Sdn Bhd for their cooperation and for providing the opportunity to develop and evaluate the DocuEase system in a real organizational setting. Special thanks are also due to the Faculty of Computer Science and Information Technology, Universiti Tun Hussein Onn Malaysia, for providing the necessary facilities, resources, and academic environment throughout this project.

Author Contribution

The authors confirm contribution to the paper as follows: study conception and design: Aleez Natasha Kameel, Shahreen Kasim; data collection: Aleez Natasha Mohd Kameel, Shahreen Kasim; analysis and interpretation of results: Aleez Natasha Mohd Kameel, Shahreen Kasim; draft manuscript preparation: Aleez Natasha Mohd Kameel, Shahreen Binti Kasim. All authors reviewed the results and approved the final version of the manuscript

References

[1] Zoho Corporation, "Zoho WorkDrive: Online file management for teams," [Online]. Available: https://www.zoho.com/workdrive/.

[2] Dropbox, Inc., "Dropbox training and self-guided courses: Dropbox Learn," [Online]. Available: https://learn.dropbox.com/.

[3] Rasnacis, A., & Berzisa, S. (2017). Method for adaptation and implementation of agile project management methodology. Procedia Computer Science, 104, 43-50.

[4] DocuWare Corporation, "DocuWare Cloud: Intelligent document management," [Online]. Available: https://www.docuware.com/.

[5] Daraojimba, E. C., Nwasike, C. N., Adegbite, A. O., Ezeigweneme, C. A., & Gidiagba, J. O. (2024). Comprehensive review of agile methodologies in project management. Computer Science & IT Research Journal, 5(1), 190-218.

[6] He, R. Y. (2015, January). Design and implementation of web based on Laravel framework. In 2014 International Conference on Computer Science and Electronic Technology (ICCSET 2014) (pp. 301-304). Atlantis Press.

[7] Hossain, M. S. (2019). Web application development with Laravel framework.

[8] Beatty, J. C., Chin, J. S., & Moll, H. F. (1979). An interactive documentation system. ACM SIGGRAPH Computer Graphics, 13(2), 71-82.

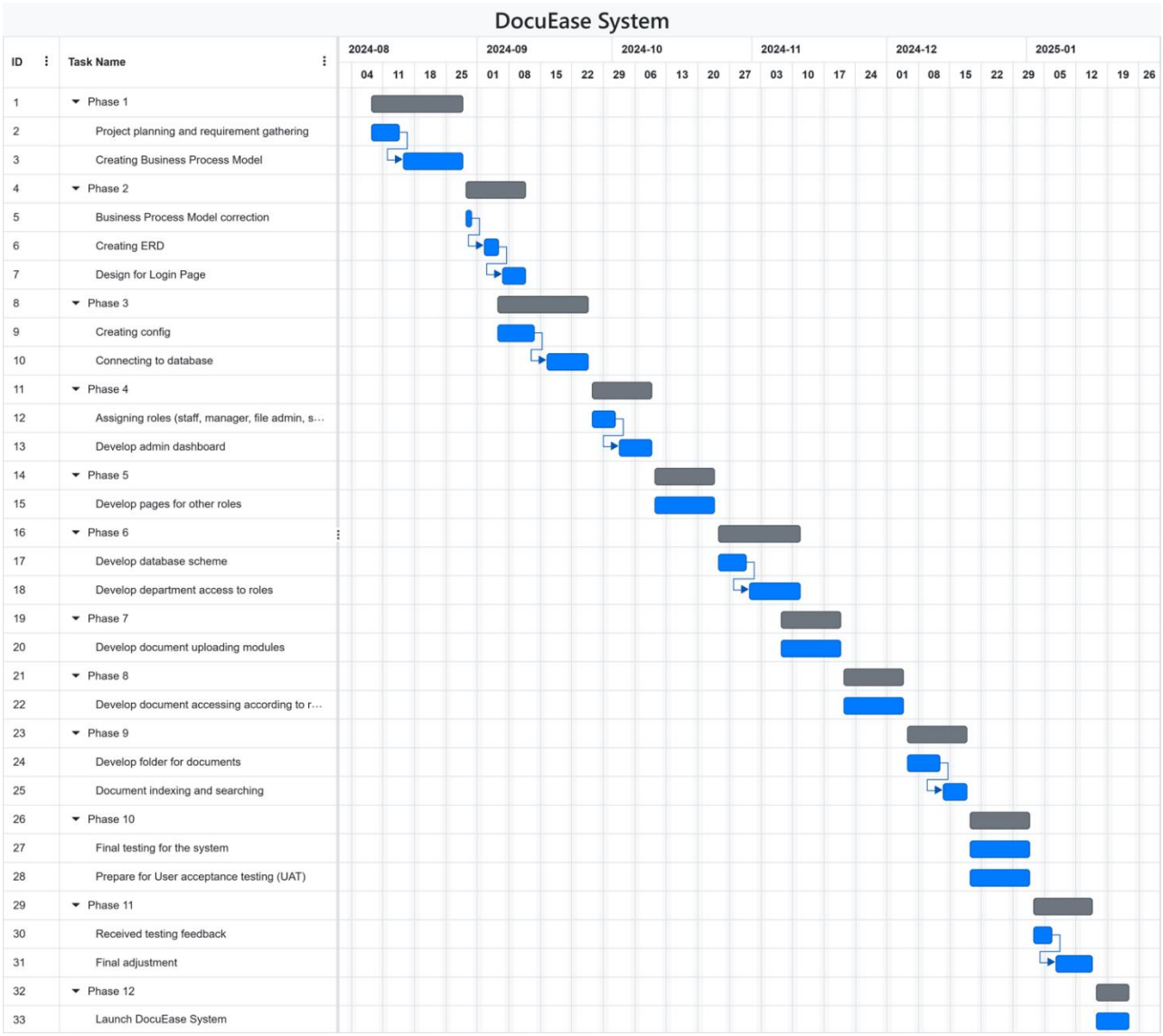
[9] Sandhu, R. S. (1998). Role-based access control. In Advances in computers (Vol. 46, pp. 237-286). Elsevier.

[10] Söderström, O., & Moradian, E. (2013). Secure audit log management. Procedia Computer Science, 22, 1249-1258.

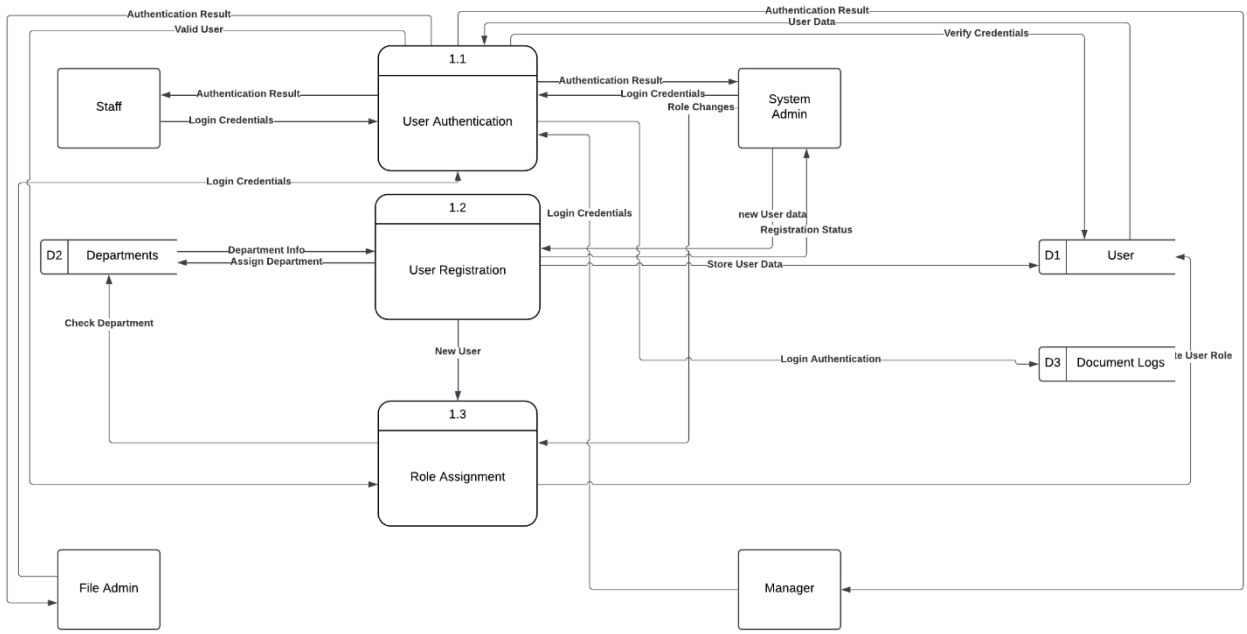
[11] Jamieson, T., Ailon, J., Chien, V., & Mourad, O. (2017). An electronic documentation system improves the quality of admission notes: a randomized trial. Journal of the American Medical Informatics Association, 24(1), 123-129.

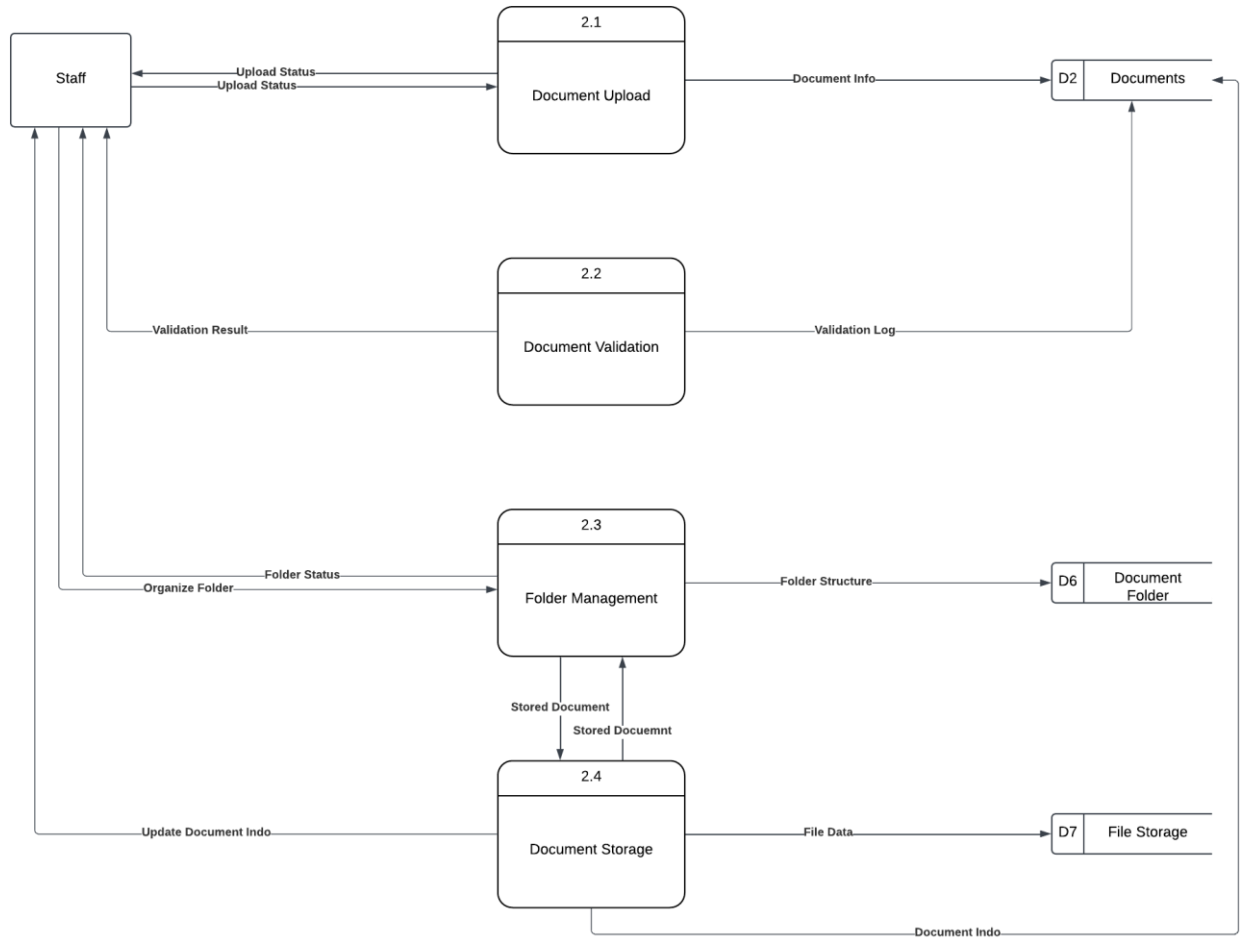
Applied Information Technology and Computer Science Vol. 6 No. 2 (2025) p. 72-91

Appendix A: Gantt Chart



Appendix B: Data Flow Diagram (DFD Level 1)





Applied Information Technology and Computer Science Vol. 6 No. 2 (2025) p. 72-91

Appendix B: Data Flow Diagram (DFD Level 1) (cont)