

# College Documents Handler

**Om Sandeep Thorat<sup>1</sup>, Kalyani Dnyaneshwar Sonje<sup>2</sup>, Surbhi Ramesh Shinde<sup>3</sup>, Mayur Kailas Ugale<sup>4</sup>**

**Mr. R. A. Thomre<sup>5</sup>**

<sup>1,2,3,4</sup>Student, Department of Computer Engineering

<sup>5</sup>Project Guide, Department of Computer Engineering

Matoshri Aasarabai Institute of Technology and Research Centre, Eklahare, Nashik, Maharashtra, India

**Abstract:** *The document presents the design and implementation of a comprehensive Full Stack Document Management System (DMS) utilizing HTML, CSS, JavaScript, PHP, and MySQL technologies. The system aims to provide educational institutions with a robust and user-friendly platform for efficient document creation, storage, retrieval, and collaboration. The front-end is developed using HTML, CSS, and JavaScript, emphasizing simplicity and intuitiveness for seamless user experience. The back-end is powered by PHP, facilitating server-side scripting and secure communication between the user interface and the database. MySQL is employed for structured data storage, with proper indexing and normalization to optimize retrieval performance. The system is deployed on cloud infrastructure to ensure scalability, high availability, data backup, and remote accessibility. Key features include role-based access control, multi-level document approval workflows, automated document generation, digital signatures, and audit logging. The proposed system significantly reduces administrative workload, eliminates manual paperwork, and accelerates document processing for students, faculty, and administrators*

**Keywords:** Cloud Computing, Document Management System, Role-Based Access Control, Automated Document Generation, Web Application, MySQL, PHP, Apache Tomcat, Digital Signature, Academic Records

## I. INTRODUCTION

In modern educational institutions, handling student and college-related documents efficiently is critical. Colleges often deal with large volumes of documents such as admission forms, bonafide certificates, transfer certificates, grade sheets, and ID proofs. Managing these documents manually is time-consuming, error-prone, and inefficient, leading to delays and misplaced records.

The College Documents Handler Using Cloud Computing system is designed to digitize, store, manage, and retrieve college documents efficiently using a web-based platform. It aims to streamline administrative tasks, reduce paperwork, and provide quick access to necessary documents for students, faculty, and administration staff. The electronic document management system is a critical step in realizing the concept of a paperless office — converting paper-based records into secure, retrievable digital assets.

Digital data offers several advantages over physical storage: it eliminates the need for physical filing infrastructure, allows rapid retrieval through search and filtering, supports remote access from any internet-connected device, and dramatically reduces the risk of document loss or unauthorized access. This system intends to integrate PDF document standards and authentication features to achieve secure, efficient college-level document management.

## II. LITERATURE SURVEY

Existing document management solutions have been extensively studied. Microsoft SharePoint is a widely used platform offering version control and workflow automation features suitable for organizational collaboration. Google Workspace provides cloud-based document management through Google Drive, enabling real-time co-editing and



storage. Alfresco offers an open-source alternative with flexible workflow management and records management features.

Chavan (2019) highlighted that DMS reduces paper usage, prevents document loss, and improves operational efficiency in educational institutions. Sharma (2020) demonstrated that web-based document systems provide transparency in request processing and speed up administrative tasks. Singh et al. (2021) emphasized that role-based access improves security and accountability in college administration systems. Patel and Desai (2022) highlighted that cloud-based systems allow students to request documents remotely and ensure data backup. Kaur and Gill (2021) showed that automated alerts significantly reduce delays in document processing and improve transparency.

A common limitation observed across these systems is the lack of automation, role-based security, and integration with mobile or cloud platforms. Many legacy systems store documents in shared drives, increasing the risk of unauthorized access, while traditional college workflows lack automated notification systems. The proposed system addresses these gaps through a cloud-native, fully integrated approach.

### III. SYSTEM DESIGN AND ARCHITECTURE

#### A. Architectural Overview

The system is built on a multi-tier web architecture. The front-end tier consists of HTML, CSS, and JavaScript pages that provide an intuitive interface for students, faculty, and administrators. The application tier is powered by Java Servlets and JSP running on Apache Tomcat, handling authentication, business logic, and document workflows. The data tier comprises a MySQL relational database for structured metadata and a cloud object storage service (e.g., AWS S3 or Google Cloud Storage) for binary document files.

The system workflow is as follows: (1) A student or staff member logs in via secure authentication; (2) The student submits an online document request; (3) The system validates user details; (4) The request enters a multi-level verification queue involving office staff, HOD, and Principal; (5) Upon approval, the document is auto-generated from a predefined template; (6) The document is stored in cloud object storage and indexed in the database; (7) The user receives a notification and can download the approved document in PDF format.

#### B. Key System Modules

- **User Interface Module:** Provides a web-based portal for login, document request submission, status tracking, and PDF download.
- **Authentication and Authorization Module:** Implements role-based access control (RBAC) for students, staff, HOD, and administrators.
- **Document Processing Module:** Extracts content, converts documents to standardized formats (PDF/DOCX), and validates document metadata.
- **Verification and Approval Module:** Supports multi-level approval workflows with digital signature integration.
- **Cloud Storage Module:** Stores all documents on cloud infrastructure with redundancy, backup, and fast retrieval.
- **Notification Module:** Sends email and dashboard alerts for request submission, approval status, and document availability.

### IV. REQUIREMENTS ANALYSIS

#### A. Functional Requirements

The system shall provide: (1) Secure user registration and role-based login; (2) Online document request submission for certificates, transcripts, and leaving certificates; (3) Real-time request tracking showing pending, approved, or rejected status; (4) Multi-level document verification by office staff, HOD, and Principal; (5) Automated document generation using predefined templates populated with student database records; (6) Digital signature and authorization stamp



support; (7) Cloud-based document storage with PDF download and secure sharing; (8) Email/dashboard notification system; (9) Admin dashboard for monitoring user activity and system performance; (10) Search and filter functionality by student ID, document type, or date; and (11) Audit trail logging of all user actions.

### B. Non-Functional Requirements

The system shall be available 24×7 with minimal downtime. It must scale dynamically using cloud infrastructure to handle peak academic periods. Security requirements include HTTPS data transmission, AES encryption for stored documents, multi-factor authentication (MFA), and role-based access enforcement. The interface shall be usable across desktops, laptops, tablets, and smartphones via standard web browsers. The system shall maintain response times within acceptable limits under normal and peak loads and comply with institutional data protection policies.

## V. TECHNOLOGY STACK AND IMPLEMENTATION

The technology stack is carefully chosen to ensure reliability, scalability, and security:

Layer	Technologies
Front-End	HTML5, CSS3, JavaScript, Bootstrap (Responsive Design)
Back-End	Java, JSP (Java Server Pages), Java Servlets
Database	MySQL – stores user profiles, document metadata, approval history
Web Server	Apache Tomcat – JSP/Servlet execution and session management
Cloud Platform	AWS / Google Cloud / Microsoft Azure – hosting, scalable infrastructure
Cloud Storage	AWS S3 / Google Cloud Storage – binary document storage
Security	HTTPS, AES/SSL-TLS encryption, Role-Based Access Control (RBAC)
Document Gen.	PDF generation libraries, template-based document creation
Dev Tools	XAMPP, Sublime Text / VS Code, MySQL Workbench

Table 1: Technology Stack

## VI. SECURITY AND SCALABILITY

### A. Security Measures

Security is a primary concern given the sensitive nature of academic records. The system implements multi-layered security including: (1) Role-Based Access Control (RBAC) ensuring users access only role-relevant features; (2) AES and SSL/TLS encryption for documents in transit and at rest; (3) HTTPS protocol for all client-server communication; (4) Multi-factor authentication (MFA) for administrator accounts; (5) Audit trails logging all actions with timestamps; (6) Secure document sharing via time-limited tokens; (7) Firewall and intrusion detection systems monitoring the cloud infrastructure; and (8) Regular security patches and updates.

### B. Scalability Considerations

The cloud-based deployment enables both horizontal scaling (adding server instances) and vertical scaling (upgrading resources). Auto-scaling is configured to provision additional resources during peak periods such as semester examinations. Load balancers distribute incoming requests across multiple server instances. The database employs partitioning, indexing, and replication strategies for consistent query performance. Elastic cloud storage automatically expands as document volume grows. CDN integration and data compression optimize network bandwidth for users in low-connectivity regions.



## VII. RESULTS AND DISCUSSION

The implemented system successfully demonstrates all core functionalities. Students can register, log in, and submit document requests within seconds. The multi-level approval workflow routes requests through office staff, HOD, and Principal, with each stakeholder receiving real-time notifications. Automated document generation produces correctly formatted PDF certificates and transcripts with digital authorization stamps, eliminating manual processing errors. Cloud deployment ensures 24×7 availability and enables remote access from any device. The role-based access control correctly restricts unauthorized access, and the audit log provides a complete history of all document lifecycle events. Testing across concurrent users demonstrated consistent response times, validating the scalability of the cloud infrastructure. The system substantially reduces the average document issuance time compared to the manual process, improving operational efficiency across the institution.

## VIII. FUTURE ENHANCEMENTS

Several enhancements are planned for future iterations: (1) Integration of Artificial Intelligence (AI) and Machine Learning (ML) for automated document classification, duplicate detection, and anomaly identification; (2) Blockchain-based document verification for tamper-proof academic records suitable for employer and university verification; (3) Optical Character Recognition (OCR) to extract and index text from scanned documents; (4) Dedicated mobile application with push notifications and offline document caching; (5) Biometric and facial recognition authentication for enhanced security; (6) Integration with national education databases, scholarship portals, and examination boards via standardized APIs; (7) Advanced analytics dashboards for administrators to track approval times, request volumes, and system usage patterns; and (8) Hybrid cloud deployment for optimized cost and data sovereignty compliance.

## IX. CONCLUSION

The College Documents Handler Using Cloud Computing system provides a digital, efficient, and secure solution for managing college documents. It significantly reduces administrative workload, speeds up document issuance, and ensures proper tracking, storage, and auditability, making college administration more organized and reliable. The system leverages a proven full-stack technology suite — HTML, CSS, JavaScript, Java/JSP, MySQL, Apache Tomcat, and cloud infrastructure — to deliver a scalable, role-based document management platform. By transitioning from manual workflows to an automated cloud-native system, educational institutions can reduce costs, minimize errors, enhance document security, and improve stakeholder satisfaction. This work establishes a strong foundation for future enhancements including AI-based automation, blockchain verification, and full digital campus integration.

## REFERENCES

- [1] T. Leikums, "A study on electronic document management system integration needs in the public sector," *International Journal of Advances in Engineering and Technology*, vol. 5, no. 1, pp. 194-205, 2012.
- [2] M. Sebastian, K. Azametii, and E. Adjeii, "Challenges in Academic Records Management in Tertiary Institutions in Ghana," *International Journal of Scientific Research in Education*, vol. 6(3), pp. 287-296, 2013.
- [3] E. Soegoto, "Implementing Laravel framework website as a brand image in the higher-education institution," in *IOP Conference Series: Materials Science and Engineering*, West Java, 2018.
- [4] S. Menon, "Document Management System," Project Report, M.Sc. Application, Cochin University of Science and Technology, India, 2011.
- [5] K. Latham, L. W., and T. Gorichanaz, "A Discussion on Document Conceptualization," *The Document Academy*, vol. 3, no. 2, 2016.
- [6] R. Kumar, "Information Retrieval System: An Overview, Issues, and Challenges," *International Journal of Technology Diffusion*, vol. 9, no. 1, pp. 1-10, 2018.
- [7] K. Ugale, B. Musande, and J. Patil, "Document management system: A notion towards a paperless office," in *1st International Conference on Intelligent Systems and Information Management (ICISIM)*, Aurangabad, Oct 2017.



- [8] D. Barker, Web Content Management: Systems, Features, and Best Practices. O'Reilly Media, 2016.  
[9] J. Valkanen, "Document Management System for Small Business," Thesis, University of Turku, Finland, 2015.  
[10] S. Gupta, A. Kumar, and R. Prasad, Document Retrieval Using Efficient Indexing Techniques. IGI Global, 2018

