A Project Synopsis

On

# DocNest

Submitted to **Walchand College of Engineering, Sangli** for the partial fulfillment of Bachelor of Engineering (UG) Degree in **Computer Science &Engineering**


Submitted By

**Ms. Srushti Somnath Jamdade | PRN : 22510006**
**Ms. Shreya Rajesh Sharma | PRN : 22510007**
**Ms. Shreya Ramesh Desai | PRN : 22510049**
**Ms. Shruti Ramesh Lakade | PRN : 22510080**

Guided By

**Dr. Anil Shanta Ramchandra Surve**




**Department of Computer Science and Engineering**

**Walchand College of Engineering, Sangli**


Academic Year 2025 – 26 (Sem-VII)

A Project Synopsis
On

# DocNest


## Abstract:

The proposed project focuses on building a web-based accreditation support system for academic institutions. Using a ReactJS frontend, NodeJS/ExpressJS backend, and MongoDB database, the system will allow faculty and administrators to upload, organize, and archive documents in a structured manner. The solution will include features like year-wise categorization, metadata-based search, dashboards, audit logs, and accreditation-ready reports.

By integrating real-time collaboration, automated archival with expiry tracking, and role-based authentication, the platform ensures transparency, document integrity, and efficiency. Designed to be scalable and cost-effective, it can serve both small institutions and large universities, making accreditation processes seamless, faster, and paperless.



***Keywords: Accreditation support system, Academic institutions, Document management, Audit logs, Real-time collaboration, Role-based authentication, Automated archival, Scalable system, Paperless accreditation.***


## 1. Introduction

Accreditation plays a vital role in ensuring that educational institutions maintain quality standards in academics, research, and administration. Bodies like NBA (National Board of Accreditation) and NAAC (National Assessment and Accreditation Council) require institutions to provide well-structured and transparent documentation of their activities and achievements. However, the current manual or semi-digital methods of record-keeping often lead to redundancy, poor accessibility, and disorganized repositories.

To address these challenges, this project proposes a Smart Digital Documentation and Accreditation Support Platform. The system will enable year-wise categorization of documents, interactive dashboards, smart search functionality, and role-based secure access, ensuring institutions remain accreditation-ready at all times.

### 1.1. Motivation:

Institutes often face difficulties in accreditation processes due to:

- Manual handling of documents, leading to loss or duplication.
- Lack of structured categorization (year/criteria/department-wise).
- Poor accessibility during accreditation audits.
- Absence of audit logs and accountability.

## 1.2. Objective:

- To study the existing manual documentation practices and identify challenges in storage, organization, and retrieval of institutional records for accreditation.
- To design a centralized digital documentation system with structured year-wise and section-wise categorization, along with a user-friendly interface.
- To implement interactive search, dashboards, and systematic presentation features to ensure accessibility, reduce redundancy, and support accreditation needs.
- To assess the system's efficiency in enhancing transparency, quick retrieval, and overall preparedness of institutes during accreditation processes.

## 1.3. Problem Statement:

To develop a smart digital platform that streamlines and visualizes institutional documents year-wise, enabling effortless and efficient review by accreditation bodies like NBA.

# 2. Related Work / Literature Survey

ERPNext Education Module, developed by Frappe Technologies Pvt. Ltd. in 2011, is an open-source ERP platform designed for schools and colleges. It provides features like course management, student records, and document storage, built on the Frappe Framework using Python, MariaDB, and Redis. However, while it supports general educational document management, it lacks accreditation-specific functionalities such as auto-archiving, structured categorization, and compliance-focused audit tools.

TCS iON Smart Campus, launched by Tata Consultancy Services (TCS) in 2012 as a cloud-based ERP. Built using Java, Spring Boot, and Oracle Database, it enables institutions to manage academics, finance, HR, and accreditation readiness. Despite its wide adoption, the system is primarily enterprise-oriented, making it costly and less adaptable for smaller institutions requiring NBA/NAAC-specific document workflows.

Fedena ERP, created by Foradian Technologies in 2009, is a widely used open-source education ERP based on Ruby on Rails, PostgreSQL, and jQuery. It helps institutions manage student records, exams, attendance, and document uploads. However, Fedena is more focused

on general educational management and lacks structured accreditation workflows, metadata-driven search, and audit-ready documentation.

OpenEduCat, introduced by Tech Receptives in 2013, extends the Odoo ERP framework for educational management. Using Python and PostgreSQL, it supports modules for students, courses, faculty, and basic document storage. While useful as an ERP, it provides limited support for accreditation-specific needs such as version control, auto-expiry of outdated documents, and structured year-wise categorization required by NBA/NAAC evaluations.

## 3. Proposed Methodology / Prototype / Idea / Algorithm

- Document Upload & Categorization
  A ReactJS-based user interface will be developed to allow faculty/admin to upload documents. Uploaded files will be stored in MongoDB's GridFS to handle large files, while metadata (year, department, NBA criteria, document type: academic, research, event, etc.) will be stored in a structured schema. TextTrack integration will automatically log every upload event (who uploaded, when, and under which category) to ensure document traceability and accountability.

- Smart Search & Filtering System
  The backend, powered by NodeJS and ExpressJS, will provide APIs for advanced search queries. MongoDB indexing will be used for fast retrieval, supporting keyword search, filters (year, department, activity type), and tag-based searching. TextTrack logs will also be searchable, allowing reviewers to trace document history (upload, update, comments) alongside content.

- Interactive Visualization & Navigation
  An admin and evaluator dashboard will be built in ReactJS with dynamic charts and graphs (using libraries like Chart.js/Recharts). Dashboards will display year-wise uploads, department-wise contributions, and pending submissions. TextTrack events will feed into the dashboard, showing activity timelines (uploads, comments, approvals), making navigation and document status tracking more intuitive.

- Secure Access & Document Integrity
  A role-based authentication system (admin, faculty, evaluator) will be implemented using JWT (JSON Web Tokens) in NodeJS/ExpressJS. MongoDB access rules will restrict unauthorized modification/deletion. TextTrack will maintain an immutable log of all access attempts to preserve document integrity and provide accountability.

- Archive & Auto-Expiry System
  Using MongoDB TTL (Time-to-Live) indexes, older/non-relevant documents will be automatically archived to a separate collection. Archived documents remain retrievable but

do not clutter the main system. TextTrack entries will mark when documents were archived/expired for accreditation history records.

- Real-Time Collaboration & Comments
  Faculty and internal reviewers can add comments/feedback directly under each uploaded document. Comments will be stored in MongoDB as sub-documents linked to the original file metadata. TextTrack integration will log all comments, edits, and feedback timestamps, ensuring evaluators see not only the document but also the collaborative review history.

- Deployment & Scalability (Docker + CI/CD)
  The entire application (ReactJS frontend, NodeJS backend, MongoDB database, TextTrack services) will be containerized using Docker for easy deployment and scaling. CI/CD pipelines (GitHub Actions/Jenkins) will automate building, testing, and deployment to ensure smooth updates.

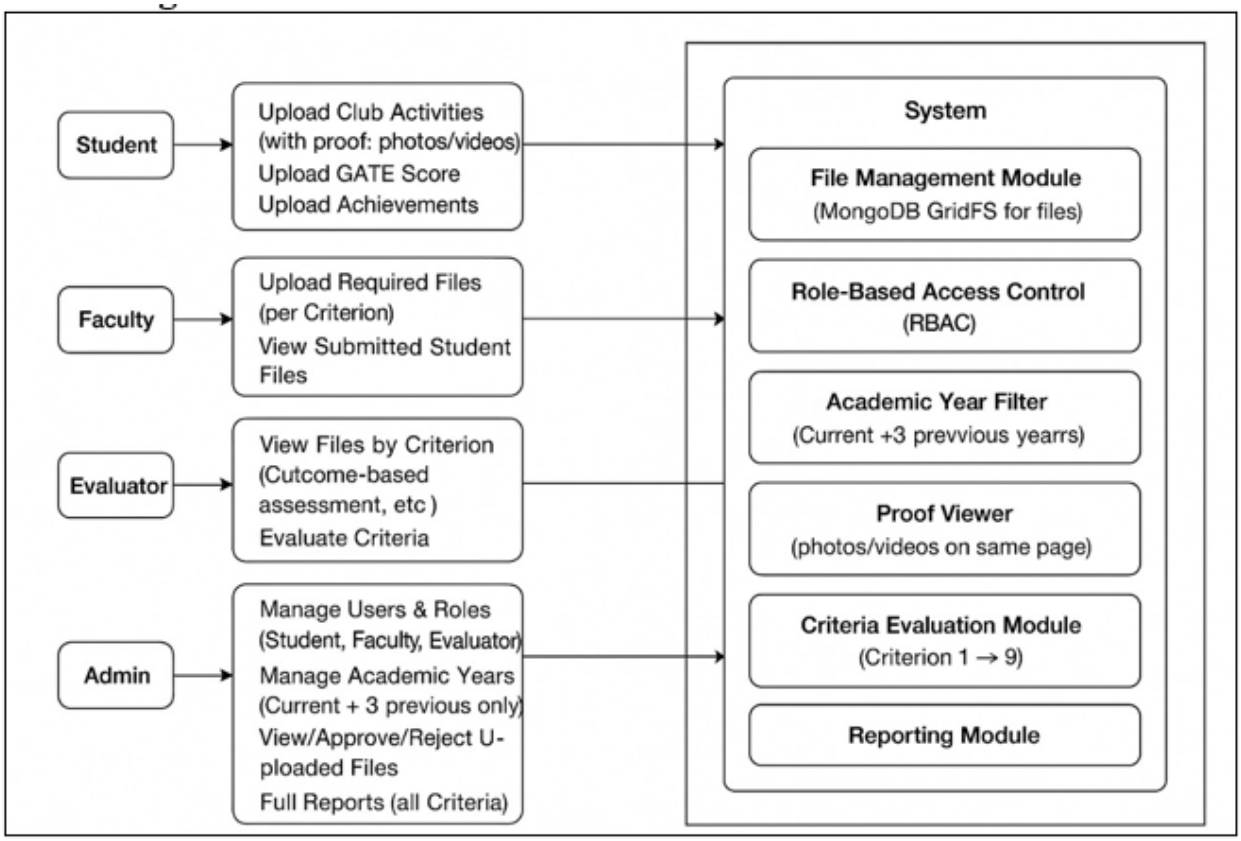


**Figure 3.1: Flowchart**

| Sr. No. | Task | Week 1 | Week 2 | Week 3 | Week 4 | Week 5 | Week 6 | Week 7 | Week 8 | Week 9 | Week 10 |
|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | Project Planning & Research | ✔ | ✔ | | | | | | | | |
| 2 | Requirements Gathering & Analysis | ✔ | ✔ | | ✔ | | ✔ | | | | |
| 3 | System Design | ✔ | ✔ | ✔ | ✔ | ✔ | ✔ | ✔ | | | |
| 4 | Iteration 1 – Development (Documents Upload & Categorization + Testing) | | ✔ | ✔ | ✔ | ✔ | | | | | |
| 5 | Iteration 2 – Development (Dashboard & Filtering + Testing) | | | | ✔ | ✔ | ✔ | ✔ | | | |
| 6 | Iteration 3 – Development (Feedback, Audit Logs & Performance Testing) | | | | | ✔ | ✔ | ✔ | ✔ | | |
| 7 | Deployment & Launch Preparation | | | | | | | ✔ | ✔ | ✔ | ✔ |

**Fig. 3.2: Project Plan**

## 4. Conclusion

The proposed system will serve as a centralized digital repository for institutional documents, streamlining the process of accreditation. With features like search and filter options, dashboards, auto-archiving, role-based access, and accreditation-ready reports, the platform eliminates redundancy and ensures quick document retrieval. In the long run, this solution will support scalability, sustainability, and compliance readiness, making accreditation faster, transparent, and stress-free. Moreover, by reducing dependence on paper-based documentation, it contributes to environmental sustainability as well.

## 5. Hardware / Software Details

Hardware Specification:

| | | |
|---|---|---|
| Processor (Server/Host) | Intel Core i5, 7th Gen / AMD Ryzen 5 | Intel Xeon / AMD Ryzen 7 |
| RAM | 8 GB | 16 GB or higher |
| Storage | 512 GB HDD | 1 TB SSD (NVMe preferred) |

Network 100 Mbps LAN 1 Gbps LAN / Cloud Deployment
Client Systems Dual-core processor, 4 GB RAM, 250 GB HDD Quad-core processor, 8 GB RAM, SSD

Software Specification:
Database MongoDB (with GridFS for file storage)
Containerization Docker, Docker Compose
CI/CD GitHub Actions / Jenkins
Authentication JWT (JSON Web Tokens)
Visualization Chart.js / Recharts
Version Control Git & GitHub
Operating System Ubuntu Server 20.04 LTS (or Windows Server)

Front End: ReactJS, HTML5, CSS3, JavaScript

Back End: NodeJS, ExpressJS, MongoDB, Docker

## 6. Estimate Cost:

| S. N. | Name of the Equipment and Details | Cost (Rs.) |
|---|---|---|
| 01 | Hosting Domain registration + Cloud hosting | ₹2,000 – ₹5,000 per year |
| 02 | Miscellaneous Maintenance, backup drives, UPS | ₹10,000 – ₹15,000 |

## References


[1] NBA Official Website
https://www.nbaind.org

[2] ERPNext Education Module
https://docs.erpnext.com/docs/user/manual/en/education

[3] "Smart Campus Solutions for Accreditation Readiness" - TCS iON
https://www.tcsion.com/


| **Dr. Anil R. Surve** | **Dr. Medha A. Shah** | **Dr. A. R. Surve** |
|---|---|---|
| Guide | Project Coordinator | Head |

Department of Computer Science and Engineering
**Walchand College Of engineering, Sangli**