

NoteNest: A Web-Based Platform for Sharing and Accessing Handwritten Academic Notes

RUSHIKET JADHAV¹, AMAN CHAUHAN², ATIF HUSSAIN³, PREM SHITOLE⁴

^{1, 2, 3, 4} *Department of Computer Engineering, PK Technical Campus, Kadachiwadi, Chakan-Shikrapur Road, Chakan, Tal: Khed, Dist: Pune, Maharashtra, India*

Abstract- In today's academic world, students often struggle to find good quality notes, especially handwritten ones that cover difficult topics in a simple way. This paper presents NoteNest, a web-based platform where students and teachers can upload and share handwritten notes in PDF format for free. The system uses React.js on the frontend and Node.js with Express.js on the backend. Notes are stored on Google Drive and metadata in MongoDB Atlas. Authentication is handled through Clerk. The platform includes trending notes, search and filter, like/unlike, bookmarks, report system with reasons, PDF preview, admin panel, and a guest access limit of 3 notes before requiring login.

Index Terms- Note Sharing, Web Application, React.js, Node.js, MongoDB, Google Drive API, Clerk Authentication, Academic Platform

I. INTRODUCTION

Every student at some point has missed a lecture or struggled to understand a topic from a textbook. In those moments, a good set of handwritten notes from a senior or classmate can make a huge difference. But finding those notes is never easy — they are scattered across WhatsApp groups, random Telegram channels, or Google Drive folders that nobody organizes properly.

NoteNest is our attempt to fix this problem. It is a web-based platform designed for students and teachers to upload, share, and access handwritten academic notes. Anyone can visit the website, search for notes by subject or branch, and download them for free.

The platform covers multiple branches like Computer Engineering, Mechanical, Civil, and Electrical. It is open to all students from any field.

II. PROBLEM STATEMENT

Currently there is no dedicated platform where students can upload and find handwritten notes in an organized way. Existing solutions have the following problems:

1. Notes shared on WhatsApp or Telegram are hard to search and get lost over time.
2. Most academic websites charge money to access study material.
3. No proper filtering system to search by branch, semester, or subject.
4. No way to know which notes are good and helpful for students.
5. No reporting mechanism for inappropriate or wrong content.

NoteNest addresses all these problems by providing a free, structured, community-driven platform.

III. SYSTEM ARCHITECTURE

The system follows a standard three-tier client-server architecture with the following layers:

Presentation Layer: React.js and Vite with Tailwind CSS. Clerk handles login and signup.

Application Layer: Node.js and Express.js handles API requests, file uploads, and Clerk authentication.

Data Layer: MongoDB Atlas stores note metadata and user data. Google Drive stores actual PDF files.

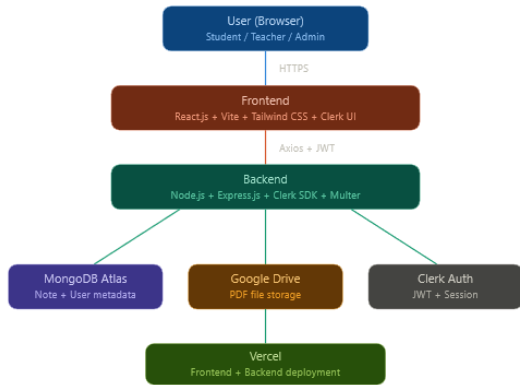


Fig. 1 — NoteNest System Architecture Overview

IV. TECHNOLOGY STACK

- React.js with Vite — user interface
- Tailwind CSS — frontend styling
- Node.js and Express.js — backend server and API
- MongoDB Atlas M0 — note metadata and user data
- Google Drive API — PDF file cloud storage
- Clerk — user authentication and sessions
- Multer — file upload handling on backend
- Vercel — deployment of frontend and backend

V. KEY FEATURES

1. Note Upload: Any logged-in user uploads PDF with title, description, branch, semester, and subject.
2. Search and Filter: Filter by branch, semester, subject. Sort by newest, most liked, most viewed.
3. Trending Notes: Homepage shows trending notes ranked by views + downloads + likes score.
4. Like / Unlike: Heart icon toggles. Red when liked, empty when not. Each user can like once.
5. Bookmark Notes: Save notes to bookmarks for quick access from profile page.
6. Report System: Users select reason — inappropriate content, vulgar images, spam, wrong subject, or copyright violation.
7. PDF Preview: Preview notes inside the platform using Google Drive iframe before downloading.
8. Share Button: Copies direct note link to clipboard for easy sharing on WhatsApp or other platforms.
9. Guest Limit: Visitors can view 3 notes free. After that a popup appears requesting sign in.

10. Admin Panel: Admin views reported notes, deletes harmful content, deactivates users, and sees platform stats.

VI. METHODOLOGY

The project followed the Agile Development Model with the following phases:

6. Requirement Analysis — identifying student problems
7. System Design — database models and API structure
8. Backend Development — API endpoints and Drive integration
9. Frontend Development — React.js pages with Tailwind CSS
10. Testing and Debugging — manual and functional testing
11. Deployment — deploying complete application on Vercel

VII. IMPLEMENTATION

Development was done in two phases — backend first, then frontend.

In the backend phase, Note and User models were designed in MongoDB using Mongoose. The Note model stores title, description, branch, semester, subject, Drive file ID, views, downloads, likes, likedBy array, and reports array with reasons.

The User model stores Clerk ID, name, email, role, and bookmarks array.

In the frontend phase, all pages were built using React.js with Tailwind CSS. React Router handles navigation. Axios handles API calls with a Clerk JWT token interceptor.

IDE: Visual Studio Code, Postman
Language: JavaScript (React.js, Node.js)
Database: MongoDB Atlas M0 (free tier)
Storage: Google Drive API with Service Account
Auth: Clerk (JWT-based authentication)

VIII. RESULTS

- PDFs up to 20MB uploaded successfully to Google Drive
- Note metadata saved correctly to MongoDB Atlas after each upload
- Search and filter returned accurate results by branch, semester, subject
- Like toggle, bookmark, and report all worked with proper authentication
- Trending algorithm ranked notes correctly by score
- Guest limit popup appeared correctly after 3 note views

- [5] Node.js Documentation, "Node.js — Asynchronous Event-Driven JavaScript Runtime," <https://nodejs.org/en/docs>, 2024.
- [6] Vercel Documentation, "Vercel Deployment Platform," <https://vercel.com/docs>, 2024.
- [7] Tailwind CSS, "A Utility-First CSS Framework," <https://tailwindcss.com/docs>, 2024.
- [8] Express.js, "Fast, Minimalist Web Framework for Node.js," <https://expressjs.com>, 2024.

IX. CONCLUSION

NoteNest is a simple but useful platform that solves a real problem faced by students every day. By providing a free and organized space to share handwritten notes, it helps students learn better and support each other. The project demonstrates practical use of React.js, Node.js, MongoDB, Google Drive API, and Clerk authentication.

X. FUTURE SCOPE

1. OCR to extract text from PDFs for full-text search
2. Mobile application for Android and iOS
3. Comment section under each note
4. Rating system from 1 to 5 stars
5. College-specific note sections

REFERENCES

- [1] MongoDB Documentation, "MongoDB Atlas Free Tier," <https://www.mongodb.com/docs/atlas/>, 2024.
- [2] Google Developers, "Google Drive API v3," <https://developers.google.com/drive/api/v3/about-sdk>, 2024.
- [3] Clerk Documentation, "Clerk Authentication," <https://clerk.com/docs>, 2024.
- [4] React Documentation, "React — A JavaScript Library for Building User Interfaces," <https://react.dev>, 2024.