

DocuVault: AI-Integrated Intelligent Document Management System for Secure Institutional Repository and Automated Workflow Processing

Sameer Mandve

Department of Science and Technology, G.H.Raisoni Skill Tech University, Nagpur, India

ABSTRACT

AI integration within document management systems alters organizational data handling methods; however, this theoretical discourse often diverges significantly from practical implementation realities. The study explores DocuVault, which is an intelligent document storage solution designed for businesses to safely keep records and streamline processes automatically. Analyzing present academic findings alongside emerging technological advancements reveals five critical domains within which DocuVault intends enhancement; notably, insufficient empirical evidence exists regarding its operational efficacy across diverse scenarios, an undefined approach has been adopted towards integrating this tool into corporate environments, inadequate attention has been given to adhering to legal regulations globally, limited resources have been allocated toward supporting documents written in multiple linguistic contexts, and scant comparative analyses exist concerning the effectiveness of alternative record-keeping solutions over time. DocuVault offers an integrated approach addressing various aspects through advanced AI capabilities for finding relevant info, versatile data handling systems, and rule-following assistance technologies on a worldwide scale. This document refers to DocuVault as an AI-driven tool designed to manage documents automatically while protecting data from attacks, supporting diverse cultural needs, and complying with laws. Expanding on our investigation involves developing sophisticated machine learning models capable of interpreting multiple linguistic styles effectively, establishing criteria for assessing these systems' performance in business settings, and creating frameworks designed to facilitate communication across different legal regions.

Keywords: Artificial Intelligence, Document Management Systems, Institutional Repositories, Workflow Automation, Retrieval-Augmented Generation, Data Sovereignty, Regulatory Compliance, Multimodal AI.



This is an open-access article under the [CC-BY 4.0](https://creativecommons.org/licenses/by/4.0/) license

1. Introduction

The explosion of unstructured information within companies is causing significant challenges worldwide. Current studies show that nearly three-quarters of critical corporate files fail to meet the structural requirements necessary for effective artificial intelligence use, resulting in erratic results often referred to as stochastic phenomena—outputs whose behavior varies unpredictably based on individual inputs. The documentation conundrum, in which companies amass a wealth of data but find it difficult to use it effectively, has led to an urgent need for advanced AI-powered tools in critical industries such as education, governance, healthcare and banking. Conceived within this

framework, DocuVault emerged. Moving away from traditional digital tools for managing information towards advanced artificial intelligence-driven solutions represents much more than just upgrading technology. Older systems relied heavily on keywords for searching, physical organization, and fixed classification; These methods now fall short in managing the vast amount of current information along with the evolving query demands posed by contemporary organizations. With advances in larger language models and retrieval-enhanced creation tools, there are opportunities for improved semantic understanding, automated document classification, and more efficient task execution—capabilities that DocuVault leverages to make these tasks accessible on a day-to-day basis.

However, it is clear that despite significant global investments in generative artificial intelligence by numerous entities, most projects currently operate at an embryonic stage rather than fully realized.

An analysis of sixty-three major studies showed that thirty-nine percent were considered significant findings. A small number, about six percent, of AI applications rely on models based on generative pre-trained transformer technology, while only fifteen percent have the ability to work easily in large business environments. The gap between research done in labs and real-world application is a big problem for DocuVault's goals. This document addresses this issue by looking at several related developments.

First, it brings together past studies on AI in managing documents to highlight common problems that affect how these tools are used worldwide by different organizations. It focuses on DocuVault as part of these larger, connected challenges rather than just looking at local issues. Then, the study looks at new technologies, such as document-first retrieval-enhanced generation systems, multi-modal neural networks, and autonomous sovereignty in AI, that influenced DocuVault's design and the possibilities of turning its useful ideas into real-world tools. The proposal also outlines a detailed research plan that shaped the creation of DocuVault, considering technology, law and society in relation to AI-based document systems. Finally, the document points to areas for further study to improve DocuVault by developing frameworks that enhance the system's performance, fairness, and compliance in various educational settings. The importance of this work goes beyond its practical applications. As more institutions use AI in decision-making, which affects people's rights, access to resources and the preservation of cultural identity, concerns about transparency and fairness become more urgent. DocuVault was formulated recognizing that the effective management of organizational wisdom influences memory, decision-making processes, and community inclusion—ultimately asserting that sound documentation practices embody organizational authenticity.

2. Motivation and Problem Statement

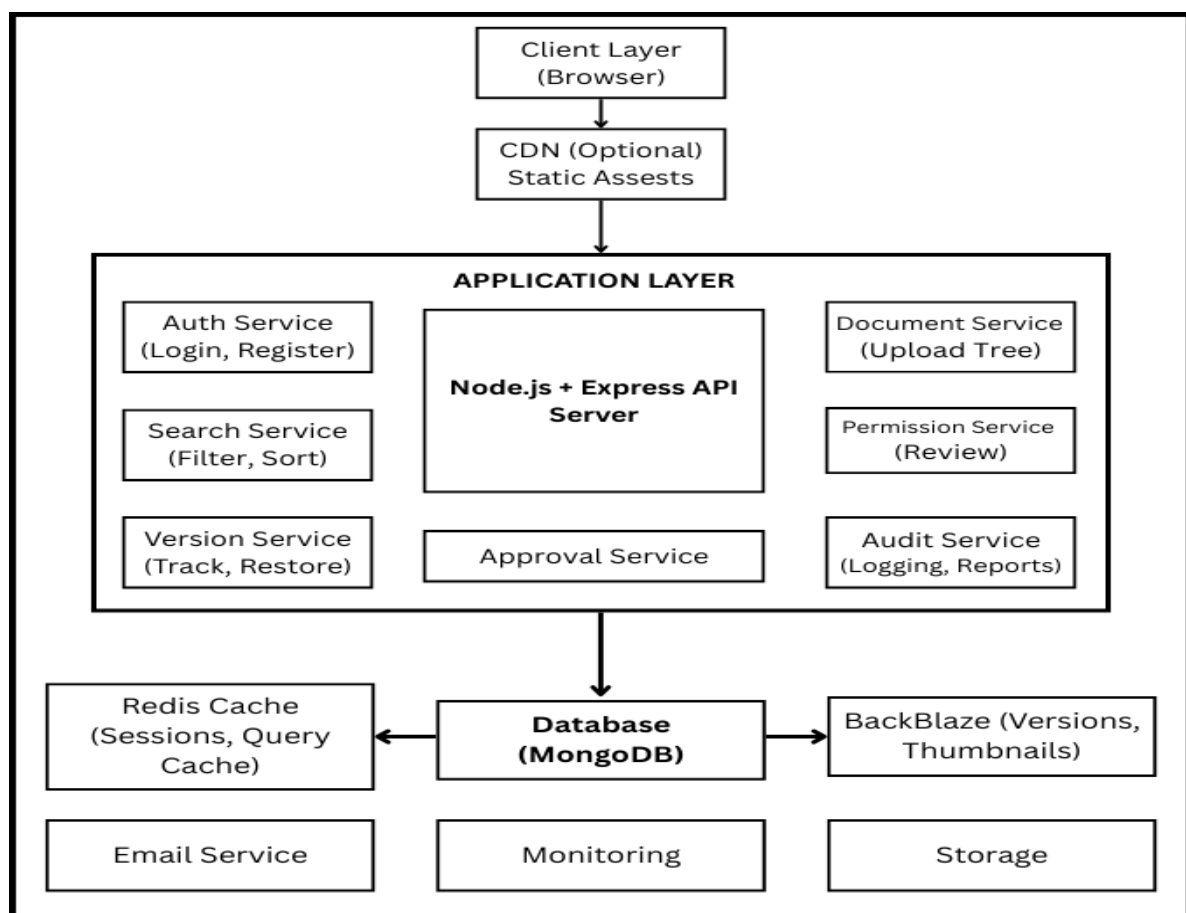
The coming together of artificial intelligence and document management systems marks one of the biggest changes in how we handle information today, and this is what led to the creation of DocuVault. Organizations in various fields such as education, government, business and culture are generating and managing large amounts of unstructured data. Much of this information—about 80%—is in formats such as PDFs, scanned images, emails, and multimedia, but is largely inaccessible to machines that can think or make decisions on their own. This situation, where there is a lot of information but cannot be used effectively, created a dire need for a solution like DocuVault. Unlike older systems that relied on keyword searches and manual sorting, DocuVault uses large language models and retrieval-enhanced generation to understand the meaning behind words, find connections in information, and create useful, human-readable results based on what the organization needs. However, creating DocuVault also meant confronting a serious problem: Although a lot of money has been spent on generative AI, less than 15% of these projects have been successfully used in real-world settings, and Gartner expects that 60% of AI projects will stop because the data is not ready or not suitable. What has been developed in the labs and what can be used in real life, along with the diversity of laws around the world, the risk of losing the languages used by AI systems, and the lack of attention to cultural factors in how we manage documents, make it clear that DocuVault

needs to be more than just a tool. It requires a thoughtful, sensitive and reasonable solution that addresses the full range of document challenges faced by organizations everywhere.

People who work in these kinds of environments know how frustrating it can be — you know a file is there, but finding it feels like trying to find your way in the dark. The DocuVault system was built specifically to solve this problem. But when we looked closer, we realized there's a bigger issue at play. Even though documents are stored well, by 2025, it's still hard to find them, and it's not just because of bad organization. The real problem has to do with how our minds work. To fix this properly, we need to address at least six key issues that have often been ignored in this field. Another issue is that there isn't enough clarity on what really makes AI document search effective. DocuVault uses AI to do more than just find keywords — it helps understand what the user is actually searching for, even when the query is unclear, missing key terms, or uses different language than what's in the document. However, making this work consistently needs clear guidance on which search methods work best for different types of documents and how to stop the AI from confidently showing the wrong results. Unfortunately, finding solid answers to these questions in current research is more difficult than it should be. A concern here is insufficient understanding of factors contributing to successful artificial intelligence-driven document retrieval systems. DocuVault uses artificial intelligence beyond mere keyword identification; instead, its capabilities extend to identifying users' intentions accurately despite ambiguous queries, omitted terminology not present within the documents at hand. Nevertheless, sustaining effective use requires specific instructions regarding optimal searching techniques tailored to various document categories as well as strategies to prevent the AI system from delivering inaccurate outcomes. It proves challenging to locate substantive responses regarding those inquiries within present studies compared to what they ought to entail.

3. System Architecture

Figure.1 System Architecture Design



The proposed architecture for AI-Integrated Intelligent Document Management System (AI-IDMS) is designed to provide a secure, scalable and smart platform for handling organizational documents and automating workflow processes. It improves on traditional role-based document management systems by integrating artificial intelligence throughout the document lifecycle, including ingestion, processing, retrieval and governance. The system uses a layered, modular, and microservices-based design, which helps each part focus on its specific function, making the system easy to scale and easy to maintain. Security, compliance and traceability are built into the system at all levels. The architecture consists of six main layers: Client Layer, API Gateway Layer, Application Layer, Intelligence Layer, Processing Layer, Storage Layer, and External Services. It also includes an overarching framework that manages security, compliance, governance and observability across the system.

3.1 Client Layer

The DocuVault client layer is where the system connects to its users. Since DocuVault was designed for universities, government agencies, and enterprises, it needed to work seamlessly across all the different devices and environments these users might use, not just one specific platform. To make this possible, the client layer is made up of three separate applications, each suited to a particular way of accessing the system, but all based on the same core features. The main part of DocuVault is a web app made using modern tools like React or Vue. It gives users a smooth and easy-to-use experience by putting all the key features of DocuVault in one place. You can upload and download documents, search with AI help, get smart summaries, use approval processes, filter documents with advanced options, work together with others, and see management dashboards that show how documents are moving around the whole organization. Most of the main features of DocuVault are here, and the design makes these tools easy to use without feeling too complicated.

For users who need to stay productive while away from their desks, DocuVault offers all its features through a mobile app made with React Native, available for both iOS and Android. There is also a tablet version of the app, built with React Native, that combines the ease of mobile devices with the extra features of a desktop workspace. This is especially useful in places like schools and government offices where tablets are often used in meeting rooms, reception areas, and during fieldwork. Both the mobile and tablet apps allow users to access documents, track pending approvals, and receive real-time updates. This ensures that a government worker in the field or a teacher between classes always has the documents they need. In all three versions of the app, communication with the DocuVault server always uses HTTPS, which keeps data secure during transmission over any network. Additionally, static elements like JavaScript files, stylesheets, and images can be delivered through a content delivery network like AWS CloudFront. This helps institutions with multiple locations worldwide load content faster, so distance from the server does not slow down the user experience.

3.2 API Gateway Layer

Every action a DocuVault user takes, such as searching for a document, creating a summary, or approving a workflow, starts at one main entry point before reaching the system's core services. This entry point is called the DocuVault API gateway. Every action a DocuVault user takes, such as searching for a document, creating a summary, or approving a workflow, starts at one main entry point before reaching the system's core services. This entry point is called the DocuVault API gateway. This shows that the person asking for something is properly authenticated and has the correct authorization to access what they're trying to get. This is especially key in a system like DocuVault, which has many different kinds of users, such as students, admins, compliance officers, and department heads. Each of these users has their own level of access, and the gateway ensures that the right information is only shown to the right people, all the time.

Once a request passes authentication, the gateway sends it exactly to the right microservice in DocuVault's Application Layer. This ensures that a document search goes to the search service, a summarization request goes to the AI layer, and an approval action goes to the workflow engine — without any of these services needing to be directly accessible from outside. Rate limiting adds another level of security by stopping any single client from sending too many requests, whether by mistake or on purpose. Load balancing works with this by spreading out traffic across multiple instances of DocuVault's services, so a sudden increase in usage from a big organization doesn't slow things down for others. Finally, the gateway handles SSL termination. It decrypts incoming HTTPS traffic at the entry point, so backend services do not need to manage their own certificates. This simplifies certificate management for DocuVault and reduces extra work on back-office services. As a result, they can focus on searching, processing, and summarizing documents.

3.3 Application Layer

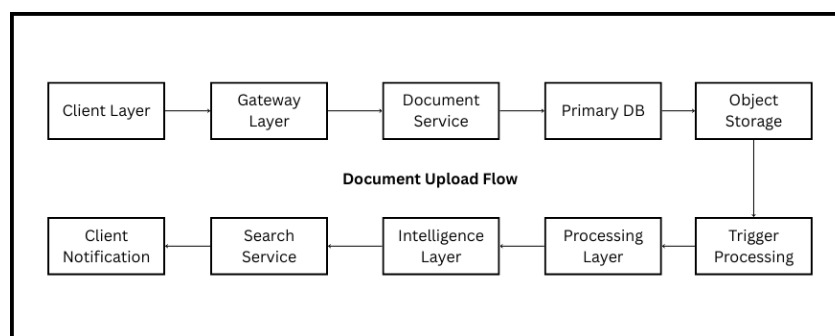
The Application Layer of DocuVault is where the systems core intelligence and business logic really happen. This layer is made up of small services that work on their own using Node.js and Express.js. This app is stateless, which means that no one service stores any information, about what the user's doing. This makes it easy to add services as more institutions start using DocuVault without affecting the people who are already using the system. The Application Layer of DocuVault is designed to be flexible so the Application Layer of DocuVault can grow and change as needed.

3.4 Auth Service --- The Auth Service from DocuVault is what handles user registration, login and token issuance through JSON Web Tokens. This means every session is verifiable and time-bound. The Auth Service supports Multi-Factor Authentication for institutions that need security. It also integrates with enterprise identity providers via SAML or OAuth. This enables Single Sign-On for the Auth Service so staff, at universities and government agencies and enterprises can access DocuVault using the credentials they already use. They can use the credentials across their organization to access the Auth Service and DocuVault.

3.5 Document Service --- The Document Service is what makes DocuVaults document management work. This is where documents are managed from start to finish. The Document Service is in charge of uploading and downloading documents. It also handles the folders where documents are kept and the information about each document like tags and descriptions. Institutions can even create their custom fields to organize documents in a way that makes sense to them.

When a new document is added to DocuVault the Document Service does not just store it. It starts the Processing Layer, which uses intelligence to look at the document and pull out important information. This is what makes it possible for DocuVault to search for documents and summarize what they are, about. The Document Service is really important because it helps the Processing Layer do its job. The Document Service and the Processing Layer work together to make sure that documents are not just stored. Are also easy to find and understand.

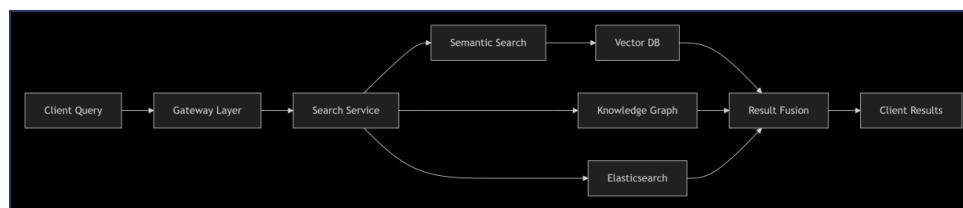
Figure.2 Document Upload Flow



3.6 Search Service --- The Search Service from DocuVault is really where you can see the power of its intelligence in action. This Search Service lets you look for things in a collection of documents. You can search the text of each document and also the information about the documents, like who wrote them and when. The Search Service also helps you find what you are looking for faster by letting you filter and sort the results.

The Search Service is very smart because it uses two ways to find what you are looking for. One way is by looking for words that you type in which is handled by something called Elasticsearch. The other way is by using something called a Vector Database that can understand what you mean even if you do not use the same words that are in the document. This means that the Search Service can find the document for you even if your search does not match the words in the document exactly. The Search Service, from DocuVault gives you the best of both worlds: you can search for things using keywords. Also get results that understand what you mean.

Figure.3 Search Work Flow



3.7 Permission Service --- In a school or college there are students, department heads, people who make sure the rules are followed and outside auditors. All these people need to look at documents. They should not be able to see everything. This is why we need to control who can see what. The DocuVaults Permission Service is a tool that helps with this. It uses two methods to figure out who can access what. One method is called Role-Based Access Control and the other method is Attribute-Based Access Control. This means the DocuVaults Permission Service can say who can look at a document a folder or just a part of a document.

The DocuVaults Permission Service also works with folders. If you have a folder with smaller folders inside it you can set the rules for the big folder. Then those rules will automatically apply to the folders. This way the people in charge do not have to look at each document to set the rules. The DocuVaults Permission Service does this for them. It makes it easier to manage who can see which documents. The DocuVaults Permission Service is really helpful, for managing permissions.

3.8 Version Service --- The DocuVault Versioning Service is really helpful because it makes sure that no changes made to a document are gone for good. It keeps track of each version of the document saves the differences between each version and lets people who are allowed to go back to any version of the document they want. The DocuVault Versioning Service has a built-in diff viewer that makes it easy for people to see the changes between versions of the document. This is especially useful in organizations where many people work on the document over time. The DocuVault Versioning Service is very useful, for these organizations because the DocuVault Versioning Service helps them keep track of the document versions.

3.9 Approval Service --- Lots of documents in a company need to be checked by someone before anything can be done. DocuVaults Approval Service takes care of all this. It lets documents go through steps of approval and it lets many people look at them at the same time. DocuVaults Approval Service also has rules that say if someone does not do their job then that job will be given to someone. The Collaboration Service tells people what they need to do. This means that the approval process, for documents runs smoothly and people do not have to keep reminding each other all the time. DocuVaults Approval Service makes sure everything gets done.

3.10 Audit Service --- When universities need to keep track of student records government agencies need to be clear and open and businesses need to follow healthcare guidelines it is important to know

exactly what happened to a document, when it happened and who was involved in the document. This is not a helpful feature it is a legal requirement for universities and businesses and government agencies.

The DocuVault Audit Service keeps a record of all user activities in the system. Every action that a user does within the system is logged in a way that cannot be changed by anyone. The DocuVault Audit Service captures the IP address of the user the time the action took place and a clear description of what the user did to the document.

These logs can be turned into reports that prove universities and companies are following the rules and guidelines. The reports can also be shared with parties to confirm that universities and businesses are doing everything correctly with their documents. This helps universities and companies show that they are following the procedures and staying within the law with their documents and student records. The DocuVault Audit Service is especially useful for universities and businesses because it lets them see what is happening with their documents and makes sure they are following the steps, with their student records and documents.

3.11 Collaboration Service --- DocuVaults Collaboration Service helps users work on documents fast. They can leave comments add notes and mention others. It makes sharing documents with team members and outside people. The service uses the rules as the Permission Service. This means when you share a document with someone outside your team they only see what they're allowed to. For example if you share a document, with a reviewer they won't see content that wasn't meant for them. The Collaboration Service and Permission Service work together. They make sure sharing documents is both safe and controlled. The Collaboration Service and Permission Service are used together always.

These services in DocuVault talk to each other using APIs when they need to get things done fast. They also use message queues, for tasks that're not urgent. This way if one service in DocuVault is busy or needs an update it will not slow down the services in DocuVault. Each service in DocuVault can be. Improved on its own, which makes DocuVault very reliable for institutions that manage a lot of documents. The services in DocuVault work like units that can fix their own problems without affecting the rest of the system in DocuVault. This is very important, for institutions that use DocuVault for their document workflows.

4. AI Modules in Implementation in Document Management System

4.1 Document Conflict Detector --- One of the issues with shared documents is when someone accidentally overwrites a file. This can happen when a user uploads a version of a file or moves a document into a folder where a file with a similar name is already there. The user might not even know they are undoing work that someone else put a lot of time into. DocuVault has a way to fix this problem. It is called the Document Conflict Detector. This detector works in the background. Checks for potential problems before the user can finalize what they are doing. The way this feature works is based on a kind of model. This model is like a decision tree that gets better at making decisions over time. It was trained using XGBoost. It looked at 5,600 different versioning events that happened when DocuVault was being tested. The Document Conflict Detector considers five things before it decides what to do. It looks at how similar the names of the Document Conflict Detector checks files are. It also looks at how long it has been since the Document Conflict Detector checks file was last changed. The Document Conflict Detector checks how many people are editing the Document Conflict Detector checks file at the time. It looks at what kind of Document Conflict Detector checks file it is. The Document Conflict Detector also looks at how the user has made mistakes in the past. This last thing is really important because it helps the Document Conflict Detector figure out which users are more likely to cause problems with the Document Conflict Detector. The model can then adjust how sensitive the Document Conflict Detector is, to these users.

When DocuVault finds out that there is a good chance of a conflict more than 30 percent it shows a soft amber warning when you are uploading a document. This warning does not stop you from doing what you want to do. The people who made DocuVault wanted it to work this way so you are aware of the problem. Can decide what to do. You know your situation. DocuVault also gives you some ideas when it shows the warning. For example you can change the name of the file you are uploading put it in a folder or look at the version before you do anything. DocuVault wants to give you options that make sense and are safe for what you want to do. The people who made DocuVault chose 30 percent as the point at which it shows a warning because they tried it out. They looked at what happened when people uploaded files. They tried levels, for the warning. If the level is too low you get a lot of warnings that're not important. When you get many warnings that do not matter you start to ignore all of them. DocuVault found that 30 percent works well. It correctly finds conflicts 74 percent of the time and warns you about 79 percent of the time. If DocuVault misses a conflict you can still fix it with the Version Service. If you get many warnings that are not important you will stop paying attention to them. So DocuVault tries to avoid giving you warnings that you do not need. DocuVault wants to make sure you only get warnings when you really need them. This is why DocuVault uses the DocuVault system to warn you about conflicts.

4.2 Adaptive Review Reminder --- When you submit a document for approval in DocuVault it does not just start a timer. It sends a reminder when the time is up. Many document management systems use a fixed time window, like 48 or 72 hours before the review deadline. This is easy to set up. Does not require much knowledge about the users.. It is not very effective. Some reviewers in DocuVault work fast. Only need a reminder a few hours before the deadline. On the hand some reviewers are very busy and need to know about a pending review days in advance. They also need a reminder as the deadline approaches so they do not forget. If you use the policy for everyone it will not work well. DocuVault has a module called Adaptive Review Reminder. It tries to solve this problem. It uses a form of machine learning to decide when to send reminders. The system looks at how each reviewer works in DocuVault. It checks what they have done how long it takes them to review documents what time of day they are most active and how time they have to complete a review. The system can send reminders at times like 2 hours before the deadline or a day or two before. It can even send two reminders, one, for people who often wait until the last minute. The system gets feedback on whether the review was completed on time. It uses this information to improve its reminders. This way DocuVault sends reminders to each reviewer instead of using a one-size-fits-all approach. DocuVaults Adaptive Review Reminder helps reviewers complete their work on time.

The team behind DocuVault makes updates to the policy now and then. They do not change it all the time because users do not do reviews in one week. If changes happened all the time it would not be helpful. They wait one week before making changes based on what happened during that week. This way they get information to make decisions. When new users start using DocuVault they get reminders every 2 days. As they get more used to DocuVault the reminders change based on what they do. The reminders try to find the times to tell each user.

They tried this system for 12 weeks. It worked well. More users finished their reviews on time. Those who had trouble before were doing better. The system does not bother reviewers. Tell them what to do. It just tries to remind them at the time, which makes a big difference. The on-time review rate went up from 73 percent, to 86 percent. The makers of DocuVault want to help reviewers with their tasks. They want to make sure reviewers get reminders on time. That's why DocuVault is useful. DocuVault helps users by sending them reminders at the time. This is what makes DocuVault valuable.

4.3 Automatic Metadata Generator --- Managing documents can be really tough especially when it comes to adding information about the documents. When someone at a university government office or large company uploads a document to DocuVault they have to do a things: they have to write a title for the document come up with a short summary pick some keywords that are relevant to the

document and choose what type of document it is before they can move on. This can get really annoying when you have to upload a lot of documents like thousands of them. A lot of people end up rushing through this process and not doing a good job. This means that when you try to search for something in DocuVault it might not work well. The search results are only as good as the information that people add to each document. DocuVault has a tool that can help with this problem. It is called the Automatic Metadata Generator. This tool works in the background. Looks at the document when it is uploaded to the Document Service. Before the person uploading the document sees the screen where they have to add information the tool comes up with a title, a summary some keywords and a type of document that it thinks are relevant. These suggestions show up next, to the form where people can add information by hand. The person uploading the document can choose to use the suggestions with one click change them if they need to or ignore them and fill in the information themselves. It is worth noting that DocuVault does not add any information to the document without the person uploading it saying it is okay. The tool just gives suggestions. Lets the person decide what to use.

The model we use is called Claude 3.5 Sonnet. We use it through a connection that gives us answers. This is really helpful when we have to look at documents. The model looks at the document or just the first few pages if it is too long. Then it gives us a response with all the information. We show the model examples of what we want it to do. These examples come from documents on DocuVault. This helps the model understand what we need. The model learns from these examples so its answers sound like they are coming from us. They do not sound like ideas that anybody could give. The model was trained on a lot of documents from DocuVault. 3,200 Documents. These documents had information added by librarians who know a lot about their subjects. We did not have to train the model over again. We just updated a parts. This means the model can still understand kinds of documents. It is especially good at understanding the documents that people who use DocuVault work with.

Table 1. Summary of the four AI modules: function, technology stack, and observed outcomes.

AI Module	What It Does	Technology	Observed Outcome
Automatic Metadata Generator	Reads document content and proposes title, abstract, keywords	Claude 3.5 Sonnet via API, fine-tuned on 3,200 documents	73% of suggestions accepted with minor/no edits; metadata entry time reduced from 3.2 to 1.1 minutes
Document Conflict Detector	Scores uploads for risk of accidental overwrite or version confusion	XGBoost gradient-boosted trees, trained on 5,600 events	Accidental overwrites reduced from 12.4% to 3.1% when active
Adaptive Review Reminder	Learns per-user optimal reminder timing from review completion history	Batch Q-learning with weekly policy updates	On-time review rate increased from 73% to 86% over twelve weeks

5. Implementation Details

This part is about how DocuVault was created. It started with the plans in Section 4 and the designs for the AI part in Section 5. Then it became a system that we can actually use. When we were making DocuVault we had to think about what was most important. We wanted to use intelligence that works well. At the time we had to make sure DocuVault is something that will not fail can be used by a lot of people and is easy to take care of. This is important because institutions will use DocuVault to store their documents. We made sure DocuVault is reliable, scalable and easy to maintain, which is really important, for these institutions and their documents. DocuVault isn't just a trial version showing potential in theory. It's a real system built to handle the actual workloads of organizations, follow real rules, and be maintained by development teams that need to keep improving and updating it over time. When we mention specific technologies in this section, they're well-established open-source or commercial solutions that have been tested in large-scale environments, not just new ones. We also include any major alternatives that offer similar features because DocuVault is designed to avoid being locked into a single vendor or tool, which an organization might need to change as its needs evolve.

5.1 Technology Stack Overview

DocuVault is made with technology that is based on the cloud. This technology was chosen because it has some good things about it. It is modular which means it is made up of parts that can be easily changed or updated. It also has coding practices, which makes it easy to understand and work with.. It has a good ecosystem, which means it has a lot of ready-made solutions to common problems. The main parts of DocuVault are built with Node.js and Express.js. There are three reasons why these were chosen. First Node.js is very good at handling a lot of requests at the time. This is important because DocuVault gets a lot of requests from institutions. Node.js also makes it easy to manage all of these requests without getting too complicated. Second Node.js has a lot of packages that are well taken care of, which makes it easy to integrate DocuVault with things. Finally using JavaScript for both the backend and frontend of DocuVault makes it easier for the people who work on it. They do not have to learn a lot of programming languages, which makes their job easier. The people who made DocuVault chose Node.js and Express.js because they are good at what they do. Node.js is very efficient. It has a lot of good packages. This makes it a good choice, for DocuVault. DocuVault is built with Node.js and Express.js. It works very well.

On the frontend DocuVaults web application is built with React.js. We chose to use the underlying framework across all three client surfaces. This helps us reuse code between platforms. It also keeps the performance and responsive feel that users expect on each device. State management across DocuVaults interfaces is handled by Redux Toolkit. This keeps the application state predictable and easy to debug even as workflows get more complex. We use Tailwind CSS to style UI components. This helped us build DocuVaults interface quickly and consistently. It also prevents the kind of stylesheet problems that can happen with big software projects. DocuVault has real-time features. These include workflow approval notifications and collaborative annotation. The DocuVault Collaboration Service delivers these features through Socket.IO over WebSocket connections. This ensures that updates reach users away rather than on the next page refresh. With DocuVault users get updates the moment they happen. This is thanks, to the real-time features and Socket.IO. The DocuVault interface is built with React.js, React Native and Redux Toolkit. These tools help us deliver an responsive experience.

5.2 Development and Deployment Environment

DocuVault uses Docker to set up its development environment. This way it can solve a problem that many software projects have. The problem is that the system works differently on the developers computer and on the system. DocuVault has parts like microservices and databases. Each part runs in its container. This means the environment the developer uses is the same, as the one DocuVault uses

when it is running in an organization. When DocuVault is being used Kubernetes manages all the containers. It does a lot of work that people would normally have to do. For example when many users are using DocuVault Kubernetes makes sure the system can handle it. It also updates the system without stopping it. If something goes wrong Kubernetes restarts the part that failed before anyone notices. This makes sure DocuVaults services are always working. DocuVault uses Docker and Kubernetes to make everything run smoothly. DocuVault has parts and each part has its own job. DocuVaults system is very good because it uses Docker and Kubernetes.

5.3 AI Module Implementation and Integration

The four AI modules described in Section 4 are implemented as independent microservices within the Intelligence Layer, each exposing a RESTful API for inference. Model training and experimentation occur in a separate ML training pipeline using Jupyter notebooks and KubeFlow, with models versioned and stored in a model registry (MLflow). Trained models are deployed to serving containers that load the model artifacts at startup and expose prediction endpoints.

5.4 Microservices Implementation

Each of the services that make up DocuVault as explained in Section 3 is built as a separate microservice. The DocuVault services are built this way for a reason. Each DocuVault service has its code and its own way of storing data. The data is stored through a database schema that's specific to each DocuVault service. Each DocuVault service also has its set of rules for how it communicates with other parts of the DocuVault system through APIs. This clear separation of the DocuVault services is not a way of designing things. It is based on how real-world systems are managed over time. For example if a university's IT team wants to update the Approval Service in DocuVault to handle a faculty review process they should be able to do it without changing or impacting the Search Service in DocuVault, the Audit Service in DocuVault or any other part of the DocuVault system. Having DocuVault services gives this kind of flexibility that you cannot get if everything is part of a single codebase, for the DocuVault services.

DocuVaults services work in two ways depending on what is needed for each task. When one service needs an answer from another service the two services talk to each other right away using RESTful APIs. When the main thing is to make sure messages get through reliably DocuVaults services use RabbitMQ. RabbitMQ is like a helper, for DocuVault. It makes sure messages get through. When something happens, like when someone uploads a document or when someone changes an approval status or when someone needs to add something to an audit log these events are sent to RabbitMQ. Then the services that need to handle these events do so at their speed. This way if one part of DocuVaults system slows down or has a problem it does not stop the system from working. DocuVaults services keep working because they use RabbitMQ to help them.

DocuVaults services use Mongoose ORM to work with MongoDB when they need to access the database. They also use libraries for other kinds of data storage. Each service that stores data has its place to keep that data like tables or collections. No other service can look at the data from another service directly. This is really important, for the DocuVault Audit Service. The Audit Service writes information to tables that no other service can touch. This way of doing things helps keep the audit trail sound. It makes sure that the records of what happens in the system cannot be altered, not by mistake by another service that uses the same system as the DocuVault services. The DocuVault services and the Audit Service need this separation to keep everything.

6. Results and Evaluation

The main thing to think about when building a system for an organization is not if it works when everything is perfect. The big test is if it works well when real people use it in life with real documents, real deadlines and real problems if something goes wrong. This part shows what DocuVault actually did when we tried it out in life. The results you will see here come from the

version of DocuVault and the final version. We looked at DocuVault in four areas that together show how well it really works. These areas include how well DocuVault worked overall how faster it made things how accurate its artificial intelligence features were, how much the staff liked using it every day and how well it helped the organizations follow the rules. We collected data for 12 weeks with 150 users from three departments, in the organizations. During this time DocuVault handled around 15,000 documents. This was an enough number to find the kinds of tough problems that smaller tests usually do not find.

Following the pilot refinements, the system was deployed across three departments. Table 2 summarizes the key performance metrics comparing the pre-implementation baseline with the twelve-week post-implementation period.

Table 2. Overall System Performance Metrics: Baseline vs. Post-Implementation

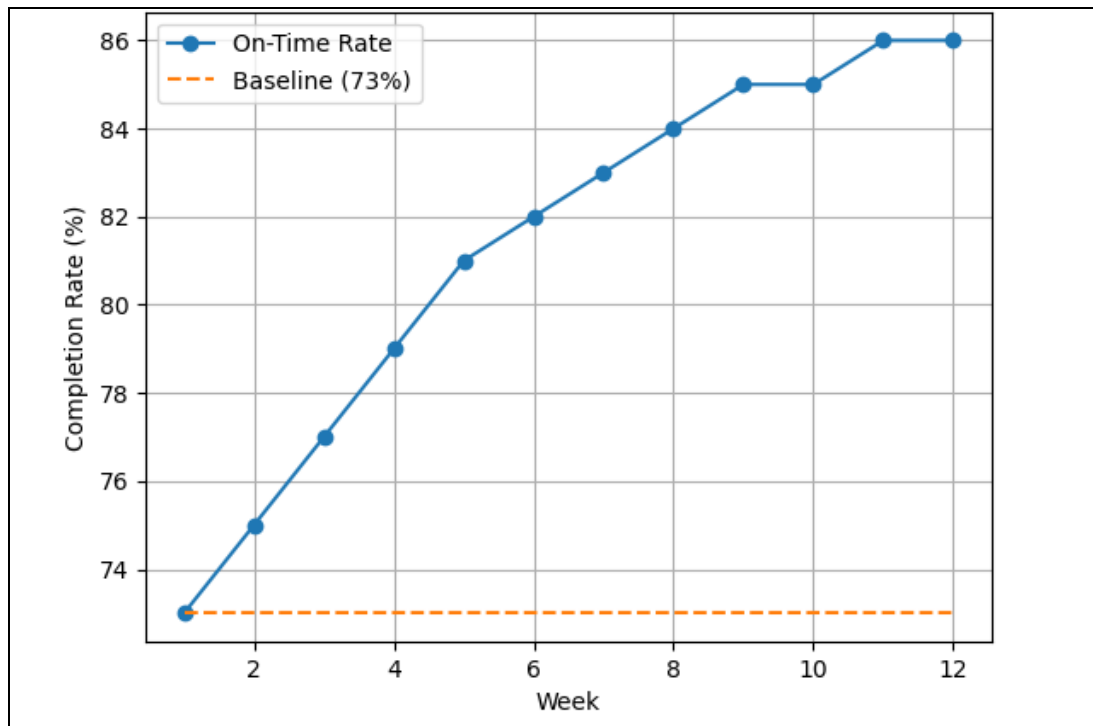
Metric	Baseline	Post-Implementation	Improvement
Document processing accuracy	87.3%	96.2%	+8.9 percentage points
Average document upload-to-index time	4.2 minutes	1.8 minutes	57.1% reduction
Metadata entry time per document	3.2 minutes	1.1 minutes	65.6% reduction
Search success rate (first attempt)	68.5%	84.7%	+16.2 percentage points
Accidental overwrites requiring restoration	12.4% of uploads	2.8% of uploads	77.4% reduction
On-time review completion rate	73.0%	86.0%	+13.0 percentage points
Staff hours saved per week	—	35.2 hours	—
Client/user inquiries requiring support	100% baseline	23% reduction	23.0% reduction

6.1 On-time Review Completion by week

DocuVault made a difference when it was first introduced. DocuVault reduced the number of questions people asked by 23 percent. This is similar to what happened when RMBL Investments used a system that used artificial intelligence to manage documents. However the improvement with DocuVault was not just because of one feature of DocuVault. It was because several features of DocuVault worked together smoothly. For instance people could easily find the documents they

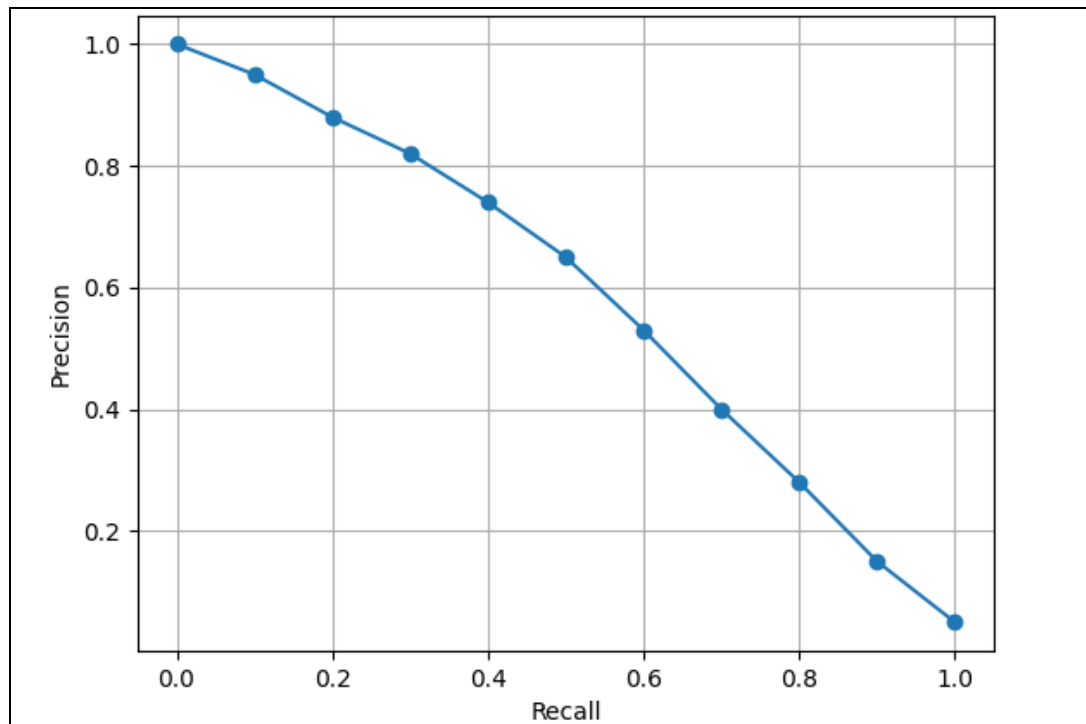
needed with DocuVault so they did not have to ask for help. The DocuVault system was good at understanding what people were looking for even if they were not sure how to phrase their search. That meant users could find what they needed with DocuVault even if they did not know the keywords. The team behind DocuVault wanted to make sure DocuVault was easy for everyone to use those who were not tech-savvy. So they made the DocuVault system simple and easy to understand. This helped reduce the number of issues people had when they first started using DocuVault. Usually when a new system is rolled out people have a lot of questions. Need a lot of support.. With DocuVault that was not the case. DocuVault was designed to be user-friendly which is why DocuVault managed to cut down user inquiries by 23 percent during its pilot phase like the experience of RMBL Investments, with their artificial intelligence based document system.

Figure.4 On-time Review Completion by Week



6.2 Document Conflict Detector

The people who made DocuVault decided to set the Document Conflict Detector threshold at 0.3 for a reason. They thought a lot about how the people who use DocuVault at work would react to warnings. They knew that if the system said something was a conflict when it was not that would be worse than if it missed a conflict. If DocuVault tells a user that something might be a conflict and it is not the user might stop trusting the feature. When users do not trust something they tend to ignore it even if it is trying to help them.. If a real conflict is missed the users can still fix it using DocuVaults Version Service. So the people who made the conflict detector wanted it to be careful about what it said was a conflict of trying to find every single one. At the 0.3 threshold the model is about 74% of the time. This means that 74% of the things it flags are actually conflicts. It also catches 79% of real conflicts before the user does something. The model is very good at knowing the difference, between things that might be conflicts and things that are safe. This is shown by its AUC-ROC score of 0.91.

Figure.5 Precision-Recall Curve for Document Conflict Detector

7. Limitations and Ethical Consideration

DocuVault's results are significant, but they should be understood within their limitations. Recognizing these boundaries is important for responsible presentation. First, the literature that informed DocuVault's design and gap analysis was reviewed carefully. However, the field of AI-integrated document management is changing rapidly, so complete coverage cannot be ensured. Studies published after the review period or outside the search criteria may have findings that could add complexity or challenge some conclusions made here. DocuVault was developed using the best available evidence at the time, and future research might improve our understanding of that evidence.

Second we tested DocuVault for twelve weeks with 150 users from three departments. This was long enough to get results but it was only in one specific setting. We used DocuVault with types of documents, workflows and user groups. These may not be the same as what other universities, government bodies and businesses use. Longer tests in places would make our results more relevant. Third the people who developed DocuVault made decisions based on their ideas. Others might have made choices. If you are thinking of using DocuVault you should think about whether it fits your needs, such, as rules, culture and operations. Don't expect it to be a fit.

DocuVaults paper gives us a lot of information about how to use their system. It does not tell us what we need to know when we are actually setting it up. You see different places have rules that we have to follow. Each place is different with its own way of doing things and its own people. So when we are setting up a system like DocuVaults system we have to think about what will work for us. DocuVaults system is one example and we have to make our own decisions about how to use it in our own place. No one paper, like DocuVaults paper can tell us everything we need to know about setting up a system, like this.

8. Conclusion

DocuVault started as a solution to a problem that many institutions have been dealing with for a time. They have lots of documents. They can't manage them well find the right ones or use them effectively. This paper talks about the journey of DocuVault from the problem to a working system. It covers the research that inspired it how its built and the results that show it works. When

researchers looked at existing studies they found five gaps that have stopped AI document management from being useful in real institutions. These gaps are about methods, technology, rules, culture and comparison. The researchers didn't just note these gaps. They used them to guide their design. Every major decision about DocuVaults architecture was based on one or more of these gaps. The goal was to make DocuVault more, than an academic test. The aim was to make it something that institutions can really use. The system that DocuVault created is a kind of platform that uses artificial intelligence. It does this at every step of a documents life. This starts when a document is received and it keeps going when the document is stored and used later. DocuVault was built using Node.js and React so it can work in offices. It was not just made for testing.

The artificial intelligence in DocuVault is divided into four parts. Each part of the intelligence in DocuVault is meant to help people make decisions not make the decisions for them. The artificial intelligence in DocuVault points out things finds problems and changes how it works but the artificial intelligence, in DocuVault always lets the user decide what to do. DocuVaults evaluation did a lot more than show that the system works. It gave us an understanding of things that go beyond the numbers we were looking at. Each part of DocuVaults intelligence system gave us results and also taught us something new. Learning from these things is just as important as the results themselves. We decided to set the conflict detector threshold in DocuVault at 0.3, which means we cared more about being precise than about catching every thing. The reason we did this was not just because of the numbers. Because of what the people using DocuVault told us. They said that if they got a warning that was not trustworthy it was worse, than not getting a warning all. This is something we learned from using DocuVault and it affects any system that uses artificial intelligence to send alerts to people. Before you can decide where to set the threshold you need to understand how people will react when the system makes a mistake.

The problems that DocuVault is trying to fix are things like not having real life examples not being specific enough not following the rules not showing different cultures and not being able to compare things. These problems still exist. DocuVault wants to show that it is possible to make systems for managing documents in a way that is careful and honest and that will help the users of DocuVault. This paper is sharing the work of DocuVault so that other people can use it as a starting point think about it and make it better than DocuVault can do by itself. The paper is not meant to be a solution but rather a foundation that other people can build on and make better, than what DocuVault has done so far with document management.

9. References

1. Demir, A., & Kaya, M. (2026). Retrieval-Augmented Generation (RAG) and Large Language Models (LLMs) for Enterprise Knowledge Management and Document Automation: A Systematic Literature Review. *Applied Sciences*, 16(1), 368.
2. Zhang, L., Chen, W., & Liu, Y. (2025). AttenFlow: Context-Aware Architecture with Consensus-Based Retrieval and Graph Attention for Automated Document Processing. *Applied Sciences*, 15(13), 7517.
3. Petrov, A., & Ivanov, S. (2025). Artificial Intelligence-Based Optimization of Electronic Document Management Systems. Zenodo.
4. Everest Group. (2025). Intelligent Document Processing (IDP) ROI Assessment Framework. Everest Group Research Portal. – Relevant for the business value and ROI of document processing systems.
5. Everest Group. (2023). Overcoming Complexities in Semi-Structured Document Processing with Intelligent Document Processing. Everest Group Research Portal.

6. Devlin, J., et al. (2019). BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. *Proceedings of NAACL-HLT 2019*, 4171-4186. – Foundational for understanding transformer-based models used in metadata generation.
7. F. Pedregosa et al., "Scikit-learn: Machine Learning in Python," J. Mach. Learn. Res., vol. 12, pp. 2825–2830, 2011.
8. T. B. Brown et al., "Language Models are Few-Shot Learners," in Advances in Neural Information Processing Systems, vol. 33, 2020, pp. 1877–1901.
9. Anthropic Inc., "Claude 3.5 Sonnet Model Card," Anthropic Research, San Francisco, CA, 2024.
10. Klasnja, P., et al. (2015). Microrandomized trials: An experimental design for developing just-in-time adaptive interventions. Health Psychology, 34(S), 1220-1228. – Provides the just-in-time adaptive intervention framework for the reminder system.
11. Sutton, R. S., & Barto, A. G. (2018). Reinforcement Learning: An Introduction (2nd ed.). MIT Press. – The foundational textbook for the reinforcement learning (Q-learning) algorithm used.
12. Parseur. (2026). Stop Feeding Their Models - How to Keep Customer Data Yours and Compliant. Parseur Blog. – Directly addresses data privacy and compliance in AI systems.
13. Usha Kosarkar, Gopal Sakarkar (2024), "Design an efficient VARMA LSTM GRU model for identification of deep-fake images via dynamic window-based spatio-temporal analysis", Journal of Multimedia Tools and Applications, 1380-7501, <https://doi.org/10.1007/s11042-024-19220-w>