



# TrustVault: A Secure Contract and Document Management System

Rugved Sachin Patil, Muhammad Ruzain Rar, Zoya Jameel Momin

*Computer Engineering Department*

*M.H. Saboo Siddik College of Engineering*

Mumbai, India

Email: rugved.221235.co@mhssce.ac.in, ruzain.221236.co@mhssce.ac.in, zoya.221257.co@mhssce.ac.in

Er. Nafisa Mapari

*Assistant Professor, Computer Engineering Department*

*M.H. Saboo Siddik College of Engineering*

Mumbai, India

Email: maparinafisa@mhssce.ac.in

**Abstract**—The digital era necessitates secure and efficient document management solutions. Our proposed system, TrustVault, is a blockchain-based document management system that employs smart contracts and AI-powered Optical Character Recognition (OCR) to facilitate secure storage, automated contract execution, and data integrity. In contrast to existing solutions, TrustVault integrates blockchain immutability, advanced OCR capabilities, and end-to-end automation into a single framework. This comprehensive approach not only ensures enhanced data security and resistance to unauthorized alterations but also streamlines contract management processes and reduces manual intervention. As a result, TrustVault provides improved transparency, scalability, and operational efficiency, setting a new benchmark in the realm of digital document management.

**Index Terms**—Blockchain, Smart Contracts, Document Management, OCR, Security, Automation

## I. INTRODUCTION

The exponential growth in digital information has amplified the need for secure, transparent, and efficient document management. Organizations—including enterprises, governmental agencies, healthcare institutions, and legal firms—routinely handle vast volumes of sensitive documents. Conventional systems based on centralized storage and manual verification are inherently vulnerable to unauthorized modifications, data breaches, and operational inefficiencies [1]. These vulnerabilities can precipitate fraud, legal disputes, and significant operational delays.

Blockchain technology offers a promising alternative by providing a decentralized, immutable ledger that preserves document authenticity and mitigates tampering risks [2]. Moreover, smart contracts facilitate automated contractual agreements, reducing manual intervention and error propensity while obviating the need for intermediaries. TrustVault integrates these technologies with AI-driven OCR to digitize physical documents and streamline processes such as contract execution and workflow management [3].

**Problem Statement:** Traditional document management systems are susceptible to security breaches and inefficiencies due to their centralized architecture and reliance on manual processes. The proposed system addresses these limitations by leveraging blockchain for immutable storage, smart contracts for automated execution, and advanced OCR for accurate digitization.

**Objective:** The primary objective of TrustVault is to develop a secure, scalable, and automated document and contract management system that ensures data integrity, minimizes manual intervention, and enhances transparency across diverse industries.

## II. RELATED WORK

Recent research has explored various aspects of blockchain integration and smart contract applications that support the development of systems like TrustVault.

### A. Blockchain Applications in Decentralized Systems

Vieira and Zhang [10] demonstrated the application of blockchain-based smart contracts in peer-to-peer energy trading within microgrids. Their work highlights the benefits of blockchain in ensuring security, transparency, and efficiency—principles that underpin TrustVault's document immutability and automated contract processing.

### B. Blockchain Adoption Frameworks

Taherdoost [11] provided a critical review of blockchain acceptance models, detailing both the drivers and barriers associated with blockchain technology adoption. This study offers valuable insights into the organizational and technological challenges that need to be addressed when deploying blockchain solutions in real-world applications, further motivating the design choices made in TrustVault.

### C. Foundational Blockchain Concepts

The seminal work of Nakamoto [12] introduced Bitcoin and laid the groundwork for blockchain technology. The principles of decentralization, cryptographic security, and proof-of-work described in this paper are fundamental to the secure transaction framework used in TrustVault's smart contracts.

### III. PROPOSED SYSTEM

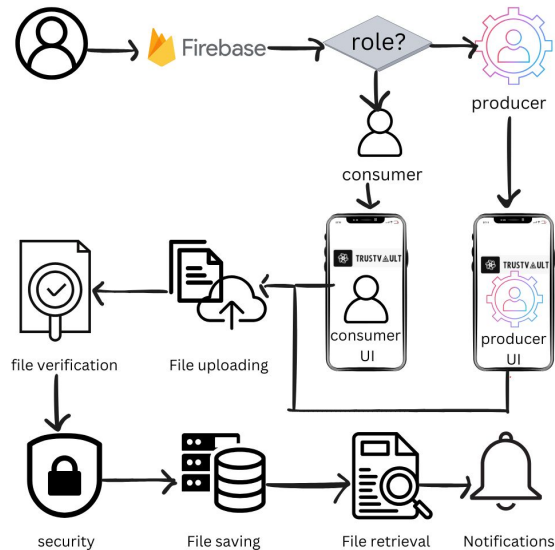


Fig. 1. Overview of the TrustVault System Architecture

The proposed TrustVault system architecture is engineered to deliver a secure, efficient, and scalable solution for document management and contract processing. The modular design incorporates advanced technologies to ensure immutability, transparency, and real-time monitoring. The system is comprised of the following components:

#### A. Frontend

The user interface is implemented using React.js for web applications and React Native for mobile platforms. This component facilitates:

- Document uploading, viewing, and management
- Contract creation, signing, and administration
- Appointment scheduling and reminder setting
- Real-time notifications for critical updates and deadlines

The frontend communicates with the backend via standardized API requests to ensure a seamless and responsive user experience.

#### B. Backend

The backend, developed using Node.js and Express.js, serves as the central processing unit of the system. It is responsible for:

- Handling API requests from the frontend
- Managing user authentication and authorization

- Facilitating data management and intercommunication among microservices

This design promotes high performance and scalability, thereby accommodating a wide range of user requirements.

#### C. Database and Storage

TrustVault employs a hybrid storage strategy to manage and secure data:

- **PostgreSQL:** Utilized as the primary relational database for structured document storage and retrieval.
- **Blockchain Ledger (Smart Contracts):** Implements an immutable and transparent storage mechanism for signed contracts, ensuring data integrity and resistance to tampering.
- **Logs and Analytics Storage:** Archives system performance metrics for debugging, monitoring, and generating actionable insights.

#### D. Key Backend Services

The backend is partitioned into discrete microservices, each addressing specific functionalities:

- **Authentication Service:** Implements secure user authentication and authorization using JWT/Auth0 protocols.
- **Document Management Service:** Oversees document uploads, retrieval operations, and access control.
- **Contract Processing Service:** Orchestrates contract creation, signing, and secure storage via blockchain technology.
- **Notification and Reminder Service:** Automates the dispatch of notifications and reminders through email, SMS, or push notifications.
- **Monitoring and Analytics Service:** Leverages tools such as Prometheus and the ELK stack to monitor system performance and produce actionable reports.

### IV. COMPARATIVE ANALYSIS OF EXISTING SOLUTIONS

Various document management systems are currently available, each offering distinct features and capabilities. However, many of these solutions lack a comprehensive approach that integrates blockchain for data immutability, Optical Character Recognition (OCR) for precise document digitization, and automation for efficient contract management. In the following, we provide a comparative study of key existing solutions:

#### DocuSign

DocuSign is a widely adopted electronic signature platform that enables digital document signing. While it provides essential document management functionalities, it lacks blockchain-based immutability, rendering it susceptible to unauthorized alterations [8].

#### Everledger

Everledger primarily focuses on tracking high-value assets using blockchain technology. Despite its blockchain integration, its application scope is limited to specific industries, and it does not offer a comprehensive document management framework that supports OCR or automated contract execution [9].

### ABBYY FlexiCapture

ABBYY FlexiCapture is an advanced OCR solution that excels in data extraction from documents. However, it does not integrate blockchain for data security or support automated contract execution, limiting its utility in an end-to-end document management system.

### Google Cloud Vision

Google Cloud Vision is a robust AI-powered OCR tool capable of high-precision text extraction from images and scanned documents. Nonetheless, it does not provide document immutability or smart contract functionalities required for automating contract management.

TABLE I  
COMPARATIVE ANALYSIS OF DOCUMENT MANAGEMENT SOLUTIONS

Solution	Blockchain Integration	OCR Capability	Automated Reminders	Customization
TRUSTVAULT	✓	Advanced	✓	✓
DocuSign	✗	Basic	✗	✗
Everledger	✓	Limited	✗	✗
ABBYY FlexiCapture	✗	Advanced	✗	✓
Google Cloud Vision	✗	Advanced	✗	✓

## V. DATA FLOW OVERVIEW

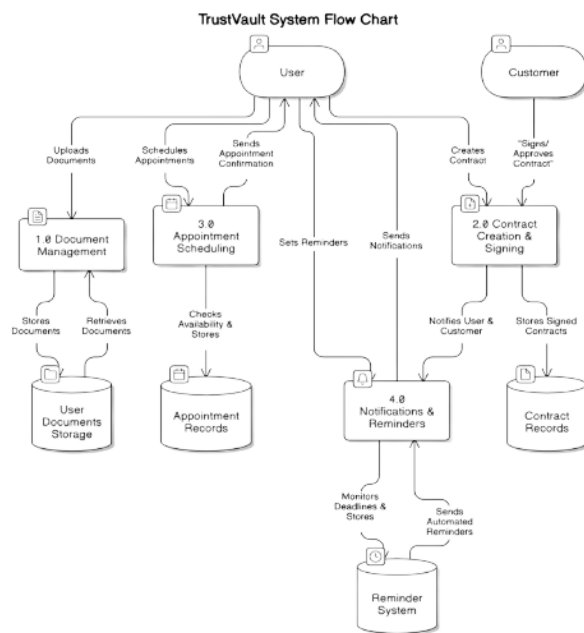


Fig. 2. Data Flow Diagram of the TrustVault System

The data flow within TrustVault is structured to guarantee secure and efficient document management, appointment scheduling, contract processing, and notification delivery. The following outlines the primary interactions:

### A. Document Management

- **User Interaction:** Users upload and access documents via the system.
- **Processes:** Uploaded documents are securely stored in the designated repository, while retrieval requests are processed and returned expeditiously.
- **Storage:** The “User Documents Storage” subsystem manages and organizes the uploaded documents securely.

### B. Contract Creation and Signing

- **User Interaction:** Users generate contracts, which are subsequently reviewed and signed by customers.
- **Processes:** Contracts are created and forwarded to customers for approval; status notifications are dispatched to both parties; signed contracts are archived within the “Contract Records” repository.
- **Storage:** The “Contract Records” ensure both immutability and ready accessibility of signed contracts.

### C. Appointment Scheduling

- **User Interaction:** Users schedule appointments through the system interface.
- **Processes:** The system validates availability and records appointment details in the “Appointment Records” database, subsequently issuing confirmation notifications.
- **Storage:** The “Appointment Records” subsystem efficiently manages scheduled appointments.

### D. Notifications and Reminders

- **Processes:** The system continuously monitors deadlines and key events; reminders are scheduled and stored within the “Reminder System”; notifications are automatically dispatched to users and customers.
- **Storage:** The “Reminder System” underpins timely and automated communication of alerts.

### E. Overall Data Flow

- 1) Users interface with the system via functionalities such as document uploads, appointment scheduling, and contract creation.
- 2) Each module interacts with its respective storage repository for data processing, retrieval, or archival.
- 3) Notifications and reminders are generated based on user activities and preconfigured system deadlines.
- 4) Robust storage mechanisms ensure the integrity of documents, appointments, contracts, and reminders.

## VI. IMPLEMENTATION METHODOLOGY

TrustVault was developed using an Agile, iterative methodology, encompassing the following phases:

- 1) **Requirement Analysis:** A comprehensive study was conducted to identify critical needs, including secure document storage, automated contract execution, and efficient data extraction.
- 2) **Design and Prototyping:** A modular architecture was adopted to decouple the user interface, backend services, and blockchain components.

- 3) **Integration and Testing:** The system integrated blockchain-based smart contracts, AI-driven OCR, and real-time notification services. Rigorous functional, stress, and security testing were executed [5].
- 4) **Deployment:** TrustVault was deployed on a cloud-based infrastructure to ensure scalability, high availability, and robust performance monitoring.

## VII. RESULTS AND DISCUSSION

Preliminary evaluations indicate that TrustVault satisfies both functional and non-functional requirements. An initial comprehensive review underscored the need for secure user authentication, efficient data management, and the integration of decentralized smart contracts while emphasizing scalability, performance, and an intuitive user interface. This foundational analysis confirmed that the system design was robust and aligned with the overall project objectives.

Subsequently, the adoption of a modular architectural approach effectively segregated the user interface, backend services, and smart contract functionalities. This decoupling simplifies development and maintenance while ensuring seamless integration across system components. Technology selections were guided by requirements for flexibility and efficiency: React Native with Expo was chosen to deliver a cross-platform, intuitive user experience; Express.js provided a lightweight, agile server environment; and the integration of Firebase and PostgreSQL facilitated secure authentication and scalable data management. The system demonstrably achieves:

- Enhanced document security and immutability through blockchain integration.
- Reduction of manual errors via automated contract execution and OCR-based document digitization.
- Improved workflow efficiency with real-time notifications and streamlined access control.

Comparative analyses with existing systems suggest that TrustVault offers substantial improvements in transparency, scalability, and user experience, thereby setting a new benchmark for document management solutions [6], [7].

## VIII. CONCLUSION

Our proposed system, TrustVault, is a robust and user-centric solution for digital contract management and document handling, underpinned by decentralized technologies. By recording transactions on an immutable ledger, TrustVault mitigates traditional inefficiencies and fosters stakeholder trust through enhanced security, transparency, and data integrity. Built on a scalable technology stack comprising React Native, Expo, Express.js, and Firebase, the system delivers real-time processing, efficient data management, and an intuitive user interface, thereby reducing operational overhead. Furthermore, the secure document storage mechanism employs advanced security protocols to protect sensitive information and ensure compliance with data privacy standards. On-the-go access, integrated reminders, and real-time notifications further augment operational efficiency. In summary, our proposed system lays

a strong foundation for the evolution of digital contract management and decentralized applications, offering a scalable and adaptable platform for future innovations.

## REFERENCES

- [1] H. Taherdoost, "Smart contracts in Blockchain Technology: A critical review," Feb. 2023.
- [2] IEEE, "Monitoring file integrity using blockchain and smart contracts," 2020.
- [3] F. Azzam *et al.*, "The use of blockchain technology and OCR in E-Government for document management: inbound invoice management as an example," Jul. 2023.
- [4] K. Bhurani *et al.*, "Smart Contracts for Ensuring Data Integrity in Cloud Storage with Blockchain," Apr. 2024.
- [5] IEEE, "A data integrity verification scheme for centralized database using smart contract and game theory," 2023.
- [6] V. Ajaikumar *et al.*, "Decentralized Data Digitization and Preservation using Blockchain," 2024.
- [7] U. Shahnaz, U. Qamar, and A. Khalid, "Using blockchain for electronic health records," Jan. 2019.
- [8] DocuSign Inc., "DocuSign Electronic Signature Platform," 2020.
- [9] Everledger, "Everledger: Blockchain for Asset Tracking," 2020.
- [10] G. Vieira and J. Zhang, "Peer-to-peer energy trading in a microgrid leveraged by smart contracts," *Renew. Sustain. Energy Rev.*, vol. 143, p. 110900, 2021.
- [11] H. A. Taherdoost, "Critical Review of Blockchain Acceptance Models—Blockchain Technology Adoption Frameworks and Applications," *Computers*, vol. 11, p. 24, 2022.
- [12] S. Nakamoto, "Bitcoin: A peer-to-peer electronic cash system," *Decentralized Bus. Rev.*, p. 21260, 2008. Available online: <https://bitcoin.org/en/bitcoin-paper> (accessed on 1 December 2022).