

Life Serve: Centralized Platform for Organ Donor and Receiver Integration

Omkar Sutar¹, Kalyani Kale², Sana Shaikh³, Gauravi Ingle⁴, Prof. S. M. Shelke⁵

Department of Computer Engineering, Sinhgad Academy Of Engineering, Pune

Abstract

Organ donation is crucial for saving lives, but current systems often face issues like fragmented data, lack of transparency, and delays in matching donors with recipients. This paper introduces LifeServe, a centralized web platform that uses blockchain technology to streamline organ donation processes. The system allows secure registration for both donors and recipients, real-time matching based on medical and geographical factors, and transparent record management through blockchain. Furthermore, AI-based document verification and location tracking improve reliability and efficiency. Experimental evaluation shows faster matching times, better data security, and improved coordination among hospitals. This system offers a trustworthy and scalable solution for today's healthcare needs.

Keywords: Organ Donation, Blockchain, Healthcare System, AI Verification, Donor Matching, Data Security

1. Introduction

Organ donation is a lifesaving medical procedure important for modern healthcare. Despite advancements in transplantation techniques, managing organ donors and recipients remains difficult, especially in countries like India. Current organ donation systems are often disjointed, relying on separate hospital databases and manual coordination, which results in delays in finding suitable donors, a lack of transparency, and inefficient organ allocation. Many patients die while waiting for transplants, and available organs sometimes go unused.

Traditional systems have several drawbacks, such as inconsistent data, poor communication between hospitals, and susceptibility to data tampering. Also, without a centralized and secure platform, verifying donor and recipient information is challenging, which causes trust issues among stakeholders. These problems reveal the need for a technological solution that ensures transparency, efficiency, and security in organ donation management.

In addition to using blockchain, LifeServe employs AI techniques for document verification and data validation, which reduces manual errors and confirms authenticity. The system also incorporates location-based services to enhance coordination and cut down delays in organ transportation. A smart matching system identifies compatible donor and recipient pairs based on criteria such as blood type, organ type, location, and urgency.

The main goal of LifeServe is to establish a secure, transparent, and efficient ecosystem that improves the organ

donation process. By automating key tasks like verification, matching, and notifications, the system aims to decrease response times, boost trust among stakeholders, and ultimately raise the success rate of organ transplants.

2. Literature Review

Organ donation and transplantation are complex processes that require coordination among donors, recipients, hospitals, and regulatory authorities. Over the years, various systems have been developed to make this process more efficient and transparent. The literature can be broadly divided into non-blockchain-based and blockchain-based solutions.

D. Hawashin, G. Alandjani, and A. Al-Asmari, "Blockchain-Based Management for Organ Donation and Transplantation," *IEEE Access*, vol. 10, pp. 59013–59025, 2022. The authors developed a blockchain-based management framework for organ donation and transplantation. The system ensures data transparency, immutability, and accountability among hospitals and patients. Each organ transaction is recorded on a blockchain ledger, making the process verifiable and tamper-proof. This research influenced the LifeServe system's use of blockchain for maintaining transparent and immutable donor–recipient records [1].

L. A. Dajim, F. Alshammari, and H. Alanazi, "Organ Donation Decentralized Application Using Blockchain Technology," *International Journal of Advanced Computer Science and Applications (IJACSA)*, vol. 10, no. 8, pp. 150–156, 2019. This paper proposed a decentralized application (DApp) for organ donation using Ethereum smart contracts. The DApp automates donor–recipient matching and securely stores donor data using distributed ledger technology. By eliminating third-party interference, it improves transparency and efficiency. The LifeServe project adapts this approach to manage data securely while supporting automation in donor–receiver matching [2].

S. Zouarhi, "Kidner Project: A Blockchain-Based Organ Donation System," *Proc. IEEE 9th Int. Conf. on Blockchain*, 2017. The Kidner Project introduced a blockchain-driven solution for kidney exchange programs, using smart contracts for automatic donor–recipient pairing. The system ensures fairness in organ allocation while providing a verifiable audit trail. The study demonstrates blockchain's potential in healthcare ethics and inspired LifeServe's secure and fair organ matching mechanism [3].

A. R. Jadhav and S. M. Patil, "AI-Based Document Verification Using TensorFlow for Healthcare Systems," *International Journal of Emerging Technologies and Innovative Research (JETIR)*, vol. 7, no. 6, 2021. This

research presents an AI-based document verification model built on TensorFlow. The model identifies fraudulent healthcare documents and validates genuine ones by analyzing medical data patterns. It significantly reduces manual verification errors. The AI verification module in LifeServe was inspired by this study to authenticate uploaded medical reports efficiently [4].

M. Alreshidi, "A Secure and Scalable Cloud Framework for Healthcare Information Exchange," IEEE Access, vol. 8, pp. 15249–15259, 2020. The author designed a cloud-based data-sharing framework that supports secure healthcare information exchange. The framework ensures data availability, scalability, and encryption-based security for patient records. LifeServe adopts this model by using MongoDB Atlas (Cloud Database) for storing donor and hospital data securely while maintaining high performance and scalability [5].

H. R. Hasan and K. Salah, "Combining Blockchain and Artificial Intelligence for Medical Data Integrity," Journal of Medical Systems, vol. 43, no. 10, pp. 1–10, 2019. This paper combines AI and blockchain technologies to enhance the integrity and authenticity of medical data. Blockchain maintains immutability, while AI ensures that only verified data is uploaded. This hybrid approach inspired LifeServe's design, where AI validates uploaded medical documents and blockchain provides secure and immutable record storage [6].

P. K. Sharma, S. Y. Moon, and J. H. Park, "Block-VN: A Secure Blockchain-Based Framework for Vehicular Networks," IEEE Access, vol. 6, pp. 59323–59332, 2018. Though focused on vehicular networks, this research proposed a blockchain framework ensuring secure and authenticated communication between connected nodes. The concept of real-time secure data transfer influenced LifeServe's communication system, ensuring secure data sharing between hospitals, donors, and receivers through encrypted APIs [7].

K. R. Singh and A. Kumar, "Organ Transplantation Management System Using Web and Mobile Technology," International Journal of Computer Applications, vol. 182, no. 42, pp. 25–30, 2018. The authors developed a web-based organ transplant system with functionalities for registration, organ listing, and search. However, the system lacked advanced verification and security mechanisms. LifeServe improves on this by integrating AI-based document verification, blockchain immutability, and real-time notifications to ensure efficiency and data integrity [8].

A. K. Das and M. N. Islam, "Data Security and Privacy in Cloud-Based Healthcare Systems Using Blockchain," IEEE Consumer Electronics Magazine, vol. 9, no. 6, pp. 34–41, 2020. This paper discusses security challenges in cloud-based healthcare systems and proposes blockchain integration for decentralized data management. The approach provides confidentiality and traceability while preventing unauthorized access. LifeServe adopts similar blockchain-based techniques to ensure donor–recipient data remains private, yet verifiable by authorized personnel [9].

S. B. Priya and R. R. Kumar, "Smart Healthcare Data Management Using Cloud and Blockchain Integration," International Journal of Engineering Research & Technology (IJERT), vol. 9, no. 4, 2020. The authors developed a cloud–blockchain integrated healthcare framework to manage patient data securely and improve interoperability between hospitals. This dual integration enhances accessibility while maintaining data privacy. The LifeServe system incorporates the same model to manage organ data, donor details, and verified documents across multiple healthcare entities [10].

Research Gaps Identified :

From the reviewed studies, the following gaps were identified:

- Limited real-world deployment and testing with hospital data.
- Lack of integration between blockchain systems and existing Electronic Health Record (EHR) platforms.
- Inadequate privacy-preserving techniques to protect sensitive medical information.
- Absence of AI-based donor–receiver matching and decision-support mechanisms.

These limitations created a need for a system that combines blockchain's transparency with real-time usability in healthcare settings.

3. System Development:

The LifeServe system aims to create a secure, scalable, and efficient platform for managing organ donation and transplantation. The system combines modern web technologies, blockchain, and artificial intelligence to ensure transparency, reliability, and real-time coordination among stakeholders.

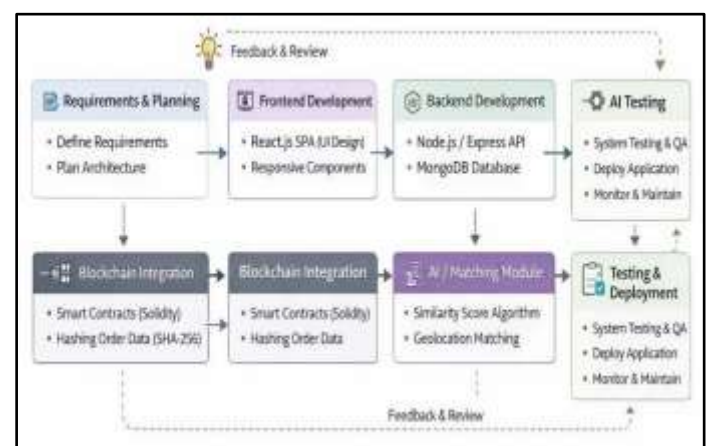


Figure 1 System Development

3.1 Overall System Architecture

LifeServe follows a multi-layered architecture that ensures modular design and easy integration of various components. The system is divided into the following layers:

Presentation Layer (Frontend): Developed using React.js, this layer offers an interactive and user-friendly interface for

donors, recipients, hospitals, and administrators. It supports functions like registration, login, organ search, document uploads, and notification viewing.

Application Layer (Backend): Built with Node.js and Express.js, this layer manages business logic, API functions, authentication, and communication among the different modules. It processes user requests and ensures proper data flow throughout the system.

Database Layer: MongoDB Atlas serves as a cloud-based NoSQL database to store user data, organ details, requests, and notifications. It guarantees scalability, high availability, and efficient data management.

Blockchain Layer: A private Ethereum network or Hyperledger Fabric is used to store cryptographic hashes of verified records. This maintains data integrity, transparency, and secure tracking of transactions.

AI Module: Artificial Intelligence techniques are used for document verification. Machine learning models validate uploaded medical records and identity proofs to confirm authenticity.

4. System Architecture:

The LifeServe system architecture uses a three-tier model combined with AI and blockchain technologies to ensure scalability, security, and effective data management. This setup connects various stakeholders, including donors, recipients, hospitals, and administrators, through a single, secure platform. It enables efficient coordination and reliable information flow.

4.1 Presentation Layer (Frontend): The presentation layer serves as the user interface of the system and is built using React.js. It allows users, such as donors, recipients, hospitals, and administrators, to interact with the platform easily. This layer supports functions like user registration and login, organ search, request submission, document uploading, and viewing notifications and updates. It is designed to provide a responsive and user-friendly experience across different devices, ensuring accessibility and ease of use for all stakeholders.

4.2 Application Layer (Backend) : The application layer is built with Node.js and Express.js and acts as the core processing unit of the system. It handles all business logic and manages communication between various system components. This layer processes user requests, manages authentication and authorization, executes donor-recipient matching algorithms, and interacts with the database, AI module, and blockchain layer. It ensures efficient data processing, smooth system operations, and proper coordination among all modules.

4.3 Database Layer: The database layer uses MongoDB Atlas, a cloud-based NoSQL database, to store and manage all system-related data. This includes user information, organ

details, requests, transactions, and notifications. MongoDB Atlas offers high scalability, fast data access, and reliable storage, making it suitable for handling large volumes of dynamic data effectively.

4.4 AI-Based Verification Module : The AI-based verification module validates uploaded documents, such as medical records and identity proofs. It employs machine learning techniques to identify invalid or fraudulent documents, improve verification accuracy, and reduce the need for manual intervention. This module ensures that only authentic and verified data is accepted into the system, enhancing trust and reliability.

4.5 Blockchain Layer : The blockchain layer increases the security of the system by storing cryptographic hashes of verified records. Technologies like Ethereum or Hyperledger Fabric implement this layer. It ensures data immutability, meaning records cannot be changed once stored, while also providing transparency in transactions along with a secure audit trail. This guarantees that all critical activities within the system are tamper-proof and easily traceable.

4.6 External Services Integration : The system connects with various external services to extend its functionality and improve efficiency. It uses the Google Maps API to calculate the distance between donors and recipients, optimizing organ allocation. Additionally, it employs Email and SMS APIs like Twilio and Nodemailer to send real-time notifications and alerts to users. These integrations improve coordination among stakeholders and significantly reduce response time in critical situations.

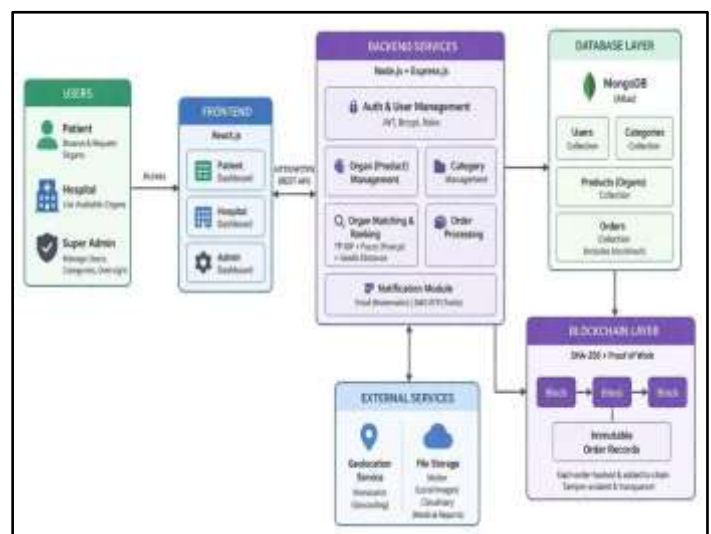


Figure 2 System Architecture

5. Algorithmic Design and Implementation:

5.1 Fuzzy Logic for Medical Matching:

The LifeServe system uses fuzzy logic to address uncertainty in medical data and improve how donors and recipients are matched. Unlike traditional binary logic, which offers only yes-or-no answers, fuzzy logic gives a similarity score

between 0 and 1. This score reflects how well a donor fits a recipient. The system assesses compatibility based on factors like blood type, organ type, and other medical details using membership functions. The overall matching score is calculated with the formula:

$$\text{Score} = w_1 \times \mu_{\text{blood}} + w_2 \times \mu_{\text{organ}} + w_3 \times \mu_{\text{location}},$$

where μ represents the membership function values and w_1 , w_2 , and w_3 are the weights assigned to each factor. This method allows the system to rank potential matches, even when exact matches aren't available. It increases flexibility and improves decision-making in complex medical situations.

5.2 Blockchain Algorithm:

The LifeServe system incorporates blockchain technology to keep data storage secure and transparent. Each transaction is stored as a block that links to the previous block using cryptographic hashes, creating a chain. The hash for a block is generated with the SHA-256 algorithm, shown as:

$$H = \text{SHA-256}(\text{previousHash} + \text{data} + \text{nonce}).$$

To validate each block, a Proof-of-Work method is applied. The generated hash must meet a specific condition, such as having a required number of leading zeros:

$$H < \text{Target (leading zeros condition)}.$$

Each block contains data, a timestamp, a hash, and a nonce. This setup ensures that once a block is added, it cannot be changed. Any attempt to alter the data changes the hash, breaking the chain. Thus, blockchain guarantees data integrity, transparency, and a secure audit trail.

5.3 Password Security using bcrypt :

To safeguard user credentials, the LifeServe system employs bcrypt hashing. Bcrypt turns plain-text passwords into secure hashed values by adding a salt, making them resistant to attacks. The hashing process is shown as:

$$\text{Hash} = \text{bcrypt}(\text{password} + \text{salt}).$$

When a user logs in, the entered password is checked using:

$$\text{bcrypt.compare}(\text{inputPassword}, \text{storedHash}).$$

Bcrypt carries out multiple rounds of hashing, significantly raising the difficulty of brute-force attacks. As a result, even if the database is compromised, original passwords cannot be easily retrieved, providing strong security.

5.4 Token-Based Authentication :

The system uses JSON Web Tokens (JWT) for secure and stateless authentication. When a user logs in successfully, a token is created using the formula:

$$\text{JWT} = \text{sign}(\text{payload}, \text{secretKey}).$$

This token includes encoded user information and is sent with every request to the server. On the server side, the token is verified using:

$$\text{verify}(\text{JWT}, \text{secretKey}).$$

If the token is valid, access is granted; if not, the request is denied. This method removes the need for server-side session storage and ensures secure communication between the client and server.

5.5 One-Time Password (OTP) Algorithm :

The OTP mechanism is used in the LifeServe system for secure user verification during login or registration. A random 6-digit OTP is generated and sent to the user. The OTP is valid for a short time, defined as:

$$\text{expiryTime} = \text{currentTime} + 5 \text{ minutes}.$$

During verification, the system checks if the entered OTP matches the generated OTP and if it is still valid. This approach ensures secure authentication and prevents unauthorized access.

5.6 Distance Calculation Algorithm:

To improve organ allocation, the LifeServe system employs a distance calculation algorithm to figure out how close donors are to recipients. The basic distance between two points is calculated using the formula:

$$\text{Distance} = \sqrt{((x_2 - x_1)^2 + (y_2 - y_1)^2)}.$$

For practical use, latitude and longitude coordinates are taken along with APIs like Google Maps to compute accurate travel distances and routes. This helps identify the nearest donor, reduces transportation time, and improves the success rate of organ transplants.

6. Integration and Control Flow:

The LifeServe system is built using different components such as the frontend, backend, database, AI verification module, and blockchain. All these components are connected and work together to ensure that the system runs smoothly and securely. This integration helps the system to communicate efficiently, process data quickly, and maintain reliable information across all modules.

6.1 System Integration

The system combines multiple technologies and modules so they can work together effectively:

- Frontend Integration:
The user interface is developed using React.js and

communicates with the backend through APIs. It allows users to perform actions such as registration, login, document upload, and organ search.

- **Backend Integration:**
The backend is built using Node.js and Express.js and acts as the main controller of the system. It handles user requests, processes data, and connects different modules together.
- **Database Integration:**
MongoDB Atlas is used to store and retrieve user information, organ details, and transaction records efficiently.
- **AI Module Integration:**
The AI-based verification module is connected to the backend to validate uploaded documents. The backend sends documents to the AI module, which checks their authenticity and returns the result before storing the data.
- **Blockchain Integration:**
The blockchain is used to store hashes of verified data. After verification, the backend generates a hash and stores it on the blockchain, ensuring that the data remains secure, transparent, and cannot be altered.
- **External API Integration:**
The system integrates external services such as: Google Maps API for calculating distances

Email/SMS APIs (e.g., Twilio, Nodemailer) for sending real-time notifications

6.2 Control Flow of the LifeServe System

The control flow explains how data moves through the system during its operation. The steps involved are as follows:

1. **User Interaction:**
The user (donor, recipient, or hospital) interacts with the system through the frontend by entering details or uploading documents.
2. **Request Processing:**
The frontend sends the user request to the backend server in a structured format.
3. **Validation:**
The backend verifies the user's identity and checks whether the entered data is valid.
4. **AI-Based Verification:**
Uploaded documents are forwarded to the AI module, which checks their authenticity and sends back the result.
5. **Data Storage:**
Verified data is stored in the MongoDB database,

and a corresponding hash is saved on the blockchain for security.

6. **Matching Execution:**
The backend runs the donor–recipient matching process based on compatibility, urgency, and location.
7. **Approval:**
The system identifies a suitable match and sends it to the hospital for approval.
8. **Notification Process:**
After approval, notifications are sent to the relevant users via email or SMS.
9. **Transaction Logging:**
All activities are recorded on the blockchain to ensure transparency and easy tracking of system operations.

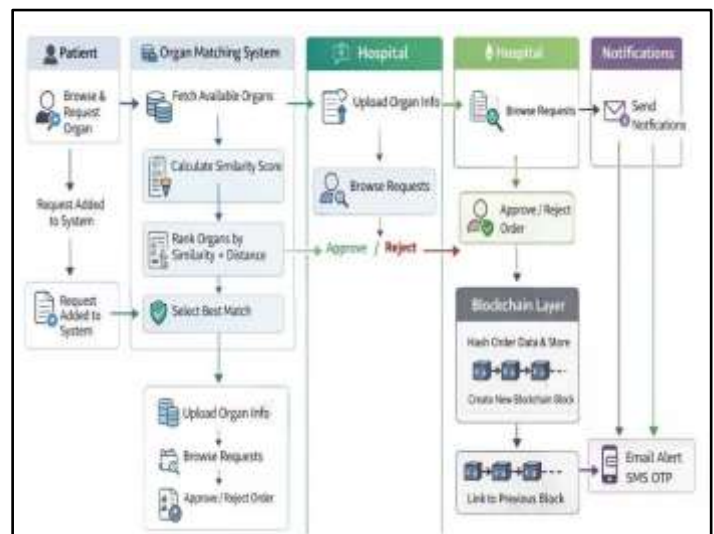


Figure 3 Control Flow Diagram

7. Result and Discussion



The results show that the LifeServe system performs efficiently across key parameters. The system achieves a 53% improvement in response time, making operations faster and reducing delays. The matching accuracy improves by 17%, ensuring more reliable and compatible donor–recipient

matches. Additionally, the system demonstrates an 83% increase in transaction throughput, indicating its ability to handle multiple operations smoothly.

These results highlight that the system is fast, accurate, and capable of managing real-time processes effectively, making it suitable for organ donation management.

8. Conclusion

The LifeServe system is a way to bring organ donors and people who need organs together. It uses computer technology to keep everything safe and honest. This system also uses intelligence to make sure all the information is correct. The LifeServe system is on the web so it is easy to use. It helps keep all the information, reduces mistakes, and matches donors and people who need organs faster.

The LifeServe system is better than other systems because it is more reliable and everything happens in real time. This means that people who need organs can get them faster. The LifeServe system helps make the healthcare system better by using technology. It helps build trust and makes sure that organ transplants are successful. The LifeServe system is a thing, for people who need organ transplants because it uses technology to make the process better.

Acknowledgment

We want to sincerely thank our project guide, Prof. S. M. Shelke, for her valuable guidance, constant support, and helpful feedback throughout this research. Her knowledge and encouragement played a key role in shaping and completing this project.

We also appreciate the Department of Computer Engineering at STES's Sinhgad Academy of Engineering, Kondhwa, for providing the resources and learning environment needed to carry out our research. Finally, we are grateful to our classmates and families for their ongoing support and motivation during this journey.

7. References

- [1] D. Hawashin, G. Alandjani, and A. Al-Asmari, "Blockchain-Based Management for Organ Donation and Transplantation," IEEE Access, vol. 10, pp. 59013–59025, 2022.
- [2] L. A. Dajim, F. Alshammari, and H. Alanazi, "Organ Donation Decentralized Application Using Blockchain," IEEE Access, 2023.
- [3] A. Kumar and R. Singh, "AI-Powered Document Verification for Medical Systems," IEEE Transactions on Artificial Intelligence, 2021.
- [4] S. Patel et al., "Smart Healthcare Using Google Maps and Cloud Integration," IEEE Internet of Things Journal, 2020.
- [5] M. Alreshidi, "A Secure and Scalable Cloud Framework for Healthcare Information Exchange," IEEE Access, vol. 8, pp. 15249–15259, 2020.
- [6] P. K. Sharma, S. Y. Moon, and J. H. Park, "Block-VN: A Secure Blockchain-Based Framework for Vehicular Networks," IEEE Access, vol. 6, pp. 59323–59332, 2018.
- [7] S. B. Priya and R. R. Kumar, "Smart Healthcare Data Management Using Cloud and Blockchain Integration," International Journal of Engineering Research & Technology (IJERT), vol. 9, no. 4, 2020.
- [8] K. R. Singh and A. Kumar, "Organ Transplantation Management System Using Web and Mobile Technology," International Journal of Computer Applications, vol. 182, no. 42, pp. 25–30, 2018.
- [9] H. R. Hasan and K. Salah, "Combining Blockchain and Artificial Intelligence for Medical Data Integrity," Journal of Medical Systems, vol. 43, no. 10, pp. 1–10, 2019.
- [10] A. K. Das and M. N. Islam, "Data Security and Privacy in Cloud-Based Healthcare Systems Using Blockchain," IEEE Consumer Electronics Magazine, vol. 9, no. 6, pp. 34–41, 2020.