# Enhanced Privacy for Health Records Using Blockchain Technologies

**Jampana Pravallika[1], Ayalasomayajula S N V Durga Koundinya[2], Allada Teja Sai Kumar[3] , Patnaikuni Vivek[4] , G.Rama Devi[5]**

*[1-4]B.Tech Student , [5] Assistant Professor, LIET*

[1,2,3,4,5] Computer Science and Systems Engineering , Lendi Institute of Engineering and Technology, Vizianagaram

---------------------------------------------------------------------***---------------------------------------------------------------------

**ABSTRACT**

This project deals with the security of digital medical history by implementing advanced measures such as encryption using blockchain methods, access control. Thanks to the integration of encryption, blockchain, dynamic access control and security even after decryption, a robust privacy framework is built to meet evolving cyber threats and strengthen security and availability of health data. This conventional method of storage deprives doctors and researchers of capitalizing on modern technologies and healthcare engines to perform clinical tests, improve patient care, and contribute to future clinical research [1]. By empowering patients using blockchain technology, this project is transforming healthcare data management to improve patient care and promote the development of a reliable healthcare environment, which means remarkable progress towards improving Privacy for Health Records Using Blockchain Technologies.

***Key Words***: Cloud Storage Security, Blockchain Technology, Health Records, Privacy and Security, Blockchain Algorithms, Keccak, Keras, Tesseract

## 1.INTRODUCTION

In an era of increasing threats to the security of medical history in the digital environment, our project aims to address the critical need to secure sensitive medical information from unauthorized access. Inspired by the effective data protection practices observed in electronic health records maintained by hospitals, our primary goal is to create effective strategies that guarantee the integrity and confidentiality of data in digital settings. Using blockchain tech-nology to integrate personal and electronic health records can ensure patient data ownership and promote transparency in data usage, addressing privacy concerns in healthcare systems [2]. The web application focuses on providing a secure online repository for the user's medical records. The user logs into their account and accesses their medical records dashboard. The user submits a photo of the physical prescription they receive from the doctor. Alternatively, the doctor can also present the user with a photo of the prescription he has prescribed. Once any user uploads a photo of a prescription, the app performs optical character recognition and extracts the text from it. To ensure the incorruptibility of the document, a hash of the document is generated and logged in the blockchain. Blockchain ensures that the hash of the document cannot be tampered with under any circumstances. When the user requests the document again, the application compares the hash of the document present in its database with the hash in the blockchain. If they are found to be the same, it means that the article has not been tampered with.

## 2. Proposed Method

Our application uses advanced algorithms to increase the privacy of health records through blockchain technology. The Keccak algorithm ensures data integrity by incorporating a hash of textual information into the blockchain, ensuring the immutability of stored documents. The Tesseract algorithm makes it easy to convert prescribed images into machine-readable text, improving data accessibility. At the same time, the Keras algorithm helps in health assessment by effectively identifying pneumonia on standard X-rays. Together, these carefully selected algorithms strengthen the confidentiality and availability of health records and create a solid foundation for our enhanced privacy solutions.

## 2.1. ALGORITHMS USED

### Keccak

Keccak is the latest algorithm from the Secure Hash Algorithms (SHA) family and generates a hash function and acts as a one-way secure file sharing channel, making it difficult for intruders to interact (Hack - Add ,Remove or change) the data shared through the channel.SHA-3 (Secure Hash Algorithm 3) is the newest member of the Secure Hash Algorithm family and is a subset of the broader Keccak cryptographic primitive family.Keccak is a cryptographic hashing algorithm designed to produce a fixed-size output from an input of arbitrary length. SHA-3 is a subset of the Keccak family standardized by the NIST. The standard lists four specific instances of SHA-3 and two extendable-output functions (SHAKE128 and SHAKE256) [3]. It operates on a state array and consists of a series of rounds, each containing five main steps: $\theta$, $\rho$, $\pi$, $\chi$, and $\iota$. In step $\theta$, each state word is combined with neighboring words to introduce diffusion. The step $\rho$ rotates each state row by a fixed amount. Step $\pi$ permutes word positions within a state. The $\chi$ step introduces nonlinearity by shuffling the bits in each word. Finally, step ι applies a rounded constant to the state. These steps are repeated for a fixed number of rounds, typically 24, after which the final state is converted to a hash output. Keccak is known for its resistance to various cryptographic attacks and has been selected for several important standards, including NIST's SHA-3.

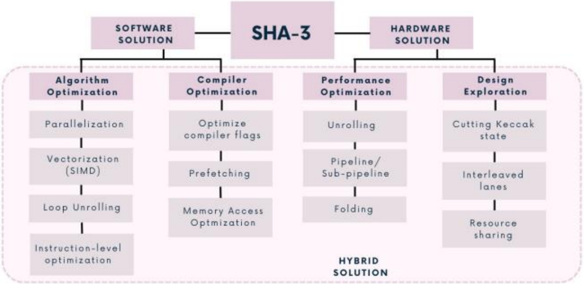

***Fig-1:*** SHA-3 Architecture

**Tesseract**

Tesseract is an optical character recognition (OCR) engine. This is used to extract text that is visually recognizable on images that are given as input. The architecture of the Tesseract algorithm revolves around the use of convolutional neural networks (CNN) for optical character recognition (OCR). It begins with pre-processing steps such as image normalization and binarization. It then feeds the processed images into CNN layers for feature extraction. Further, the recurrent layers interpret the sequence of elements to recognize characters. The line finding algorithm is one of the few parts of Tesseract that has previously been published [4]. Finally, the decoding step breaks down the recognized characters into words or sentences that form the basis of the OCR process. It has different line and word search algorithms like line search, basic settings. It also has word recognition algorithms like chopping connected characters, matching broken characters. This helps us share information seamlessly without having to worry about intruders taking the information (Hash-Code) we share.

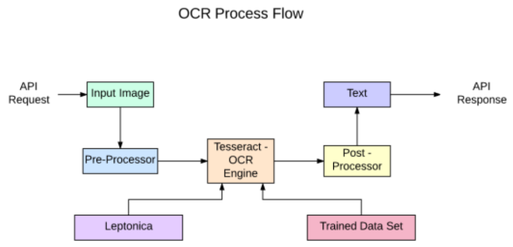

***Fig-2:*** Tesseract Architecture (Optical Character Recognition Process flow)

**Keras**

Keras is a high-level neural network API written in Python that serves as an interface to TensorFlow, Microsoft Cognitive Toolkit (CNTK), or Theano. Its main purpose is to enable rapid experimentation with deep neural networks. An outline of using Keras usually involves several key steps. First, import the necessary modules, including the models and layers needed to construct the neural network. Then define the architecture of the neural network by gradually adding layers, specifying the number of neurons and activation functions. Next, compile the model specifying the optimizer, loss function, and evaluation metrics. After compilation, feed the training data into the model and train it using the fit() function, specifying the number of epochs and the batch size. After training is complete, evaluate the performance of the model on unseen data using the evaluate() function. Optionally fine tune the hyperparameters and architecture based on the evaluation results and retrain the model if necessary. Finally, use the trained model to predict the new data by calling the forecast() function. It is primarily based on the "Show & Not Tell" principle, which makes the outputs generated by it easy to understand for the user.

## 2.2. LIBRARIES USED:

**keras:**

Keras, a high-level neural network API based on Python, makes experimenting with deep neural networks simple. Users import core modules, define a layered network architecture, and build a model by specifying an optimizer, loss function, and metrics. Training includes fitting data using the fit() function, adjusting epochs and batch sizes. After training, the model is evaluated on unseen data using the evaluate() function. Tuning the hyperparameters and refining the architecture based on the results are optional. Predictions of new data are made using the forecast() function in accordance with the "Show & Not Tell" principle for user-friendly outputs.

**tensorflow:**

TensorFlow, an open source machine learning system developed by Google Brain, supports the creation, training and deployment of deep learning models. It deals with tasks such as image classification and natural language processing. Offering both high-level APIs like Keras and low-level operations, TensorFlow boasts a flexible architecture. Extensive documentation, tutorials, and pre-built models increase its popularity for various deep learning projects.

**numpy:**

NumPy, the core Python package for scientific computing, facilitates the efficient manipulation of large, multidimensional arrays. Its ndarray object serves as the basis for various scientific computing libraries that support tasks such as linear algebra and statistical analysis. NumPy allows developers basic math functions for seamless data operations.

**opencv-python:**
OpenCV, an open-source computer vision library, excels in image and video processing. It includes tools for object detection, face recognition and feature extraction. Written in C++, OpenCV includes Python bindings for cross-language accessibility. Its comprehensive set of algorithmic tools makes it indispensable in fields such as robotics, augmented reality, and medical imaging.

**pillow:**
Pillow, a fork of the Python Imaging Library (PIL), simplifies image manipulation. Scipy 4.4 PIL (Python Image Library) Currently known as PILLOW, this library provides comprehensive support for all image file formats and basic operations on them, such as resizing, rotating, creating thumbnails, and converting between file formats, but support has been discontinued since 2011 and it still maintains its popularity due to its simplicity and ease of dealing with its dependencies. [5]. Active maintenance and compatibility with Python 2 and 3 contribute to its popularity among developers integrating image processing into Python programs.

**pytesseract:**
Pytesseract, a Python wrapper for Google's Tesseract-OCR engine, facilitates optical character recognition (OCR). Ideal for scanning documents and extracting text from images, it simplifies the integration of OCR into Python applications. With a user-friendly interface, Pytesseract is valuable for digitizing documents, automating data entry and extracting text for further analysis.

## 2.3. TECHNOLOGIES USED:

**Django framework (for backend):**
Django is a high-level Python web framework known for its simplicity, scalability, and versatility in building web applications. It follows the Model-View-Template (MVT) architectural pattern, providing a robust ORM (Object-Relational Mapping) for interacting with databases, built-in authentication and authorization mechanisms, and a powerful admin interface for managing content. Django's extensive ecosystem of reusable components, known as "apps", simplifies development by promoting code reusability and maintainability. Using the tools and commands of these languages we can make an interactive and user-friendly interface that can be easily accessed by people [6].

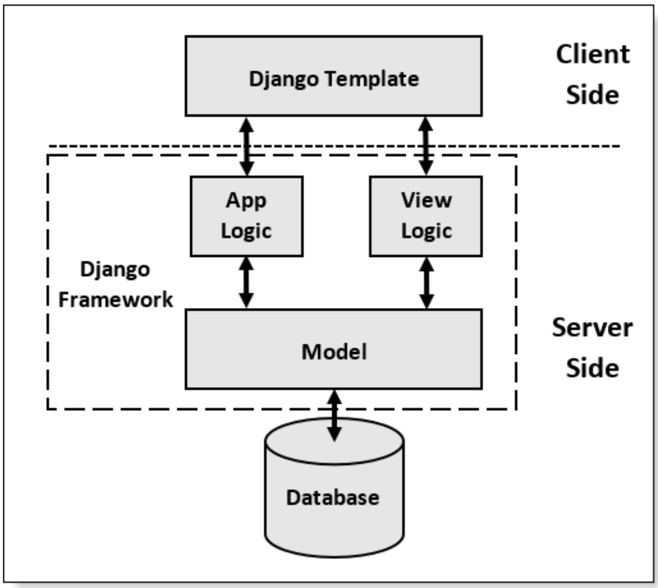

***Fig-3:*** Architecture of Django Framework

**Web3.py:**
Web3.py is a Python library that allows interaction with the Ethereum blockchain. It facilitates tasks such as querying blockchain data, sending transactions, and deploying smart contracts. Web3.py abstracts the complexity of the Ethereum JSON-RPC API and provides developers with a Pythonic interface for building decentralized applications (dApps) and integrating blockchain functionality into their Python projects.

**Ganache:**
Ganache  is a local blockchain environment mainly used for Ethereum development and testing purposes. It allows developers to simulate the Ethereum blockchain network locally on their machines, offering features such as instant mining, adjustable gas settings, and deterministic behavior for easier debugging. Ganache guarantees a detached individual blockchain network you can use for testing brilliant agreements [7]. The user-friendly interface and integration with popular Ethereum development tools make it an indispensable tool for developing smart contracts and testing workflows.

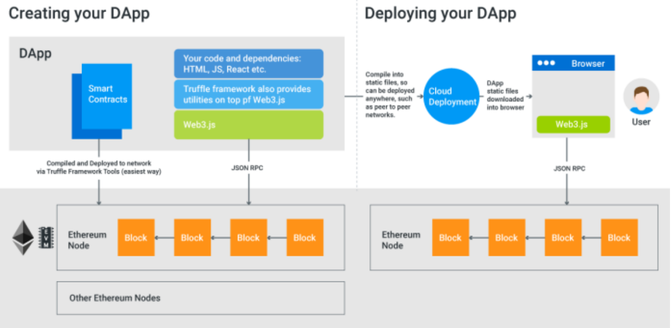
Creating your DApp
Deploying your DApp
DApp
Smart Contracts
Your code and dependencies: HTML, JS, React etc.
Truffle framework also provides utilities on top pf Web3.js
Web3.js
Compile into static files, so can be deployed anywhere, such as peer to peer networks.
Cloud Deployment
DApp static files downloaded into browser
Browser
Web3.js
User
Compiled and Deployed to network via Truffle Framework Tools (easiest way)
JSON RPC
JSON RPC
EVM
Ethereum Node
Block
Block
Block
Block
Ethereum Node
Block
Block
Block
Block
Other Ethereum Nodes

***Fig-4:*** Ganache Process flow

**Solidity:**
Smart contracts are programs stored on the blockchain, often developed in a high-level programming language, the most popular of which is Solidity [8]. Solidity is a high-level programming language specifically designed for writing smart contracts on blockchain platforms, the most notable being Ethereum. It offers JavaScript-like syntax and facilitates the creation of autonomous and self-executing contracts that reside on the Ethereum Virtual Machine (EVM). Solidity enables developers to define rules and logic for blockchain-based applications, including token contracts, decentralized finance (DeFi) protocols, and non-fungible token (NFT) standards such as ERC-721 and ERC-1155.

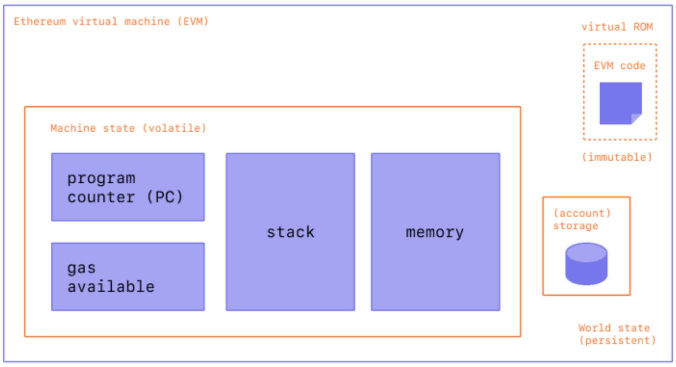

***Fig-5:*** Architecture of Ethereum Virtual Machine

## 2.4.RESULTS:

The implementation of advanced security measures in our project, including encryption using blockchain methods and dynamic access control, represents a significant advance in the protection of digital medical history. The integration of these technologies not only strengthens security in transit, but also provides a robust privacy framework after decryption, thus addressing the evolving cyber threats prevalent in healthcare data management. In our investigation of the effectiveness of these security measures, we delved into the awareness and understanding of implemented blockchain-based security features among healthcare stakeholders. Our investigation was inspired by observed data protection practices in electronic health records and utilized a comprehensive data set from the project's user interactions. Our findings suggest a transformative impact on healthcare data management, particularly in terms of patient empowerment through blockchain technology. However, a major revelation is emerging regarding the broader field of digital health records. Our study highlights different levels of awareness about different aspects of implemented security measures. While a significant number of users were aware of the encryption and access control features, there was a significant gap in understanding the nuances of blockchain technology. This gap was more pronounced when considering the complexity of dynamic access control and post-decryption security measures. The implications of this awareness gap are notable, especially in terms of user engagement and potential resistance to blockchain-based security adoption. Users who are well versed in the security measures implemented are more likely to be actively involved in the management of their medical records. On the contrary, those who are not aware of the complexities, especially when it comes to the role of blockchain in the incorruptibility of documents, may perceive the system as complex and opt for traditional methods. A critical aspect of our study revolves around user interaction with the web application, specifically the process of uploading recipes and subsequent authentication via blockchain. Our survey revealed that the majority of users showed a positive response to the seamless integration of optical character recognition and blockchain technology. However, a smaller segment faced challenges in understanding and implementing this process, indicating a need for better user education and interface clarity. Although our project has succeeded in creating a robust security framework for digital medical records, the awareness gap among users, especially regarding blockchain technology, is a significant challenge. Future iterations should emphasize user education and interface optimization to bridge this gap and ensure widespread understanding and adoption of blockchain-based health data security. This study represents a key step in advancing the privacy paradigm for health records using blockchain technologies and highlights the continued need for user-centric approaches in the digital healthcare transformation.

## 3. CONCLUSIONS

Our project addresses the need for increased security in digital medical history management. The combination of encryption, blockchain methods, dynamic access control and post-decryption security has resulted in a stronger privacy protection framework. This framework not only addresses the challenges of evolving cyber threats, but also improves the overall security and availability of health data. Our investigation into the potential of blockchain technology in healthcare data management has shown positive results. By providing patients with the ability to securely store and verify medical records, our project aims to contribute to the creation of a reliable healthcare environment. The integration of blockchain ensures the integrity and inviolability of medical records, which represents an advance in the enforcement of privacy protection for health records. However, like any technological innovation, the success of our project depends on understanding and acceptance by users. Our findings reveal a significant gap in awareness among healthcare beneficiaries, particularly regarding the complexities of blockchain technology. Despite positive reactions to the encryption and access control features, a significant portion of users showed limited understanding of blockchain and its role in ensuring the inviolability of documents. This gap in awareness poses a challenge to the widespread adoption and use of blockchain-based security in healthcare data management. To fully realize the potential benefits of our project, future efforts should focus on user education, interface optimization, and clear communication of the benefits offered by blockchain technology. Bridging this gap is key to encouraging user engagement and creating a positive perception of system complexity. Our study, while highlighting challenges, provides valuable insights that guide future iterations and improvements. The success of our project lies not only in its technological innovation, but also in its ability to be accessible and understandable to a wide user base. By addressing the awareness gap and integrating user-centered approaches, we can ensure that the benefits of increased security and privacy in health data management reach and positively impact a larger population. Our project represents an advance in basic health data security. Through the integration of advanced technologies, we strive to contribute to a healthcare environment where patients feel empowered, make informed decisions, and have confidence in the confidentiality and integrity of their medical records. The journey to a more secure and privacy-focused digital healthcare environment continues, and our project serves as a foundation for continued progress in this critical area.

## ACKNOWLEDGEMENT

We would like to thank the Department of Computer Science and Systems Engineering , Lendi Institute of Engineering and Technology, Vizianagaram for helping us carry out the work and supporting us all the time.

## REFERENCES

1. Stasia,K.,Vikram,S.(2008).Medical Record Privacy and Security in a Digital Environment,Institute of Electrical and Electronics Engineers,10,2.

2. Aleksandr,k.,Vimal,D.,Chibuzor,U.,Alex,N.(2023)Privacy-Conflict Resolution for Integrating Personal and Electronic Health Records in Blockchain-Based Systems,Blockchain in Healthcare Today,6,276,1-27.

3. Alessandra,D.,Maurizio,M.,Guido,M.(2023). Comparative Study of Keccak SHA-3 Implementations,Cryptography,7,60,1-16.

4. Jay,A,P.(2021).Handwritten And Printed Text Recognition Using Tesseract-OCR, International Journal of Creative Research Thoughts (IJCRT),9,9,66-77.

5. J., A.; Eunice, J.; Popescu, D.E.; Chowdary, M.K.; Hemanth, J. Deep Learning-Based Leaf Disease Detection in Crops Using Images for Agricultural Applications. Agronomy 2022, 12, 2395

6. Ashish,Ch.,Pawan,S.(2021).Management of Django Web Development in Python,n. Journal of Management and Service Science,1,2,1-17.

7. Nasurudeen,A,N.(2023).A Build and Deploy Ethereum Smart Contract for Food Supply Chain Management in Truffle - Ganache Framework,International Conference on Advanced Computing and Communication Systems,36-40.

8. Diego,M.,Billy,T.(2023).SSCalc: A Calculus for Solidity Smart Contracts, Software Engineering and Formal Methods,184-204.