Proceedings of the Sixth International Conference on Trends in Electronics and Informatics (ICOEI 2022)
IEEE Xplore Part Number: CFP22J32-ART; ISBN: 978-1-6654-8328-5

A Criminal Record Keeper System using Blockchain

Aditya Vijaykumar Singh[1], Ashwin Omprakash Tiwari[2], Shreyash Sanjay Singh[3], and Vivian Brian Lobo[4]
[1,2,3]U.G. Student, [4]Assistant Professor
Department of Computer Engineering
St. John College of Engineering and Management, Palghar
[1]leoadi246@gmail.com, [2]tiwariashwin55@gmail.com, [3]sshreyash34@gmail.com, [4]lobo.vivian27@gmail.com

***Abstract*—*In India, as per records, it has been observed that with each succeeding year, criminal activities are surging. Crime is an act executed against the law of constitution and is a menace to our society. Thus, crime and criminals need to be monitored conveniently. Unlike USA or other countries, in India, criminal records are not public, which means neither private organizations nor common people can access criminal records for employee verification. A centralized system, which gives access to criminal data, would be a feasible solution to this problem. This study aims to develop a Flutter-based blockchain-enabled permissionless system that would help in the remote access of criminal records. The system would be developed using Ethereum, which offers vivid features such as truffle for enabling Ethereum virtual environment, Ganache for personal testing of smart contracts, InterPlanetary file system for encrypted storage of data and easy tracking of files, and MetaMask, which acts as a digital wallet.*

Keywords*—*Blockchain, Criminal Record Keeper, Ethereum, permissionless

I. Introduction

Crimes are the harsh truth of our society, and with this escalating modern society, the rates of crime are increasing. In 2020, as per National Crime Records Bureau, a total of 66,01,285 crimes were reported, which was noticed to be 28 % more than the previous year [1]. Criminals are adapting new techniques to commit crime. Conversely, agencies are pursuing novel techniques to hunt them. In this hunt of crime and criminals, common people are affected indirectly. One of the principal objectives of governing agencies is to maintain a database of criminal records, but administrating and making such databases could be at times unmanageable.

Vivid Indian law agencies possess different criminal databases, and accessing such databases could be tedious, which may also act as a barrier in communication amongst agencies during emergency. Moreover, separate databases could be subjected to unethical changes. With a steep rise in criminal data, a good criminal record keeper system is needed that can securely store information in a centralized manner and enable data sharing in a convenient manner. This is where blockchain paves its way by ensuring that no single authority can modify data, which means the system will be predominantly secure and transparent.

Using blockchain-based systems offer characteristics like convenience, cost-effectiveness, and efficiency. In India, it has been observed that criminal records are not public due to which Indians cannot verify whether the person next to him is a honest citizen or a criminal. An employer does not know the criminal status of an employee, a house owner has no idea whether his/her maid is a criminal or not. Therefore, to overcome such problems, this study aims to develop a Flutter-based blockchain-enabled criminal record keeping system that would eventually help not only the law system but also private organizations to maintain and access criminal records.

II. Related Work

A great deal of related literature is available wherein blockchain is used to maintain criminal records, and a few of them have been specified.

Omar and others [2] explicated on the increase of crime-related activities, which is difficult to handle as it is tampered quite easily. They were of the opinion that a criminal record storage system could be implemented using blockchain technology for data storage. Their developed system was capable to maintain all criminal records in an efficient manner. The authors preferred to store data in a decentralized manner. For increasing data security, they encrypted the data where a randomly generated key was used. The authors also specified that no two files can have the same key for encryption, and data cannot be accessed directly by an user. This made their blockchain system attain highest level of security.

Khan *et al.* [3] examined record management in a police station. They identified e-FIR data novelty and found false registrations appended with police stations in a centralized database and addressed a distributed blockchain solution. They used Hyperledger fabric and Ethereum. Moreover, a MATLAB framework was used, which was interfaced with Ethereum using Python. They also used smart contracts to improve data security. Furthermore, they discussed about hashing and its security levels. Finally, the authors demonstrated how a transaction was stored on a ledger. Hingorani *et al.* [4] developed a system on Ethereum that could offer transparency and ensure data privacy in terms of storage. The system was built using Flutter. The developed system proved beneficial to police officers. Encrypted records were stored in InterPlanetary File System (IPFS). The purpose of developing such a system was to protect data and reduce corruption and provide justice to a user. Criminal records that were created were tamperproof and immutable, which offered high-level security. Shukla and others [5] offered a secure method for a first information report (FIR) system using blockchain. The authors were of the opinion that nowadays because of the increasing use of computers and mobiles by many people, data over the internet is unsecure. Keeping this concern in mind, the authors developed an application that presented an advanced and efficient online criminal record checker and created a web-based GUI that significantly reduced the work of entering criminal data manually, thereby offering high-level security. Dube *et al.* [6] used blockchain as it offers services such as agreement, irreversibility, traceability, and online data privacy. Their main aims was to secure data in different departments and offer privacy. Because criminal records are important, a blockchain network was used to maintain records of the system. Dini *et al.* [7] explicated on some of the key advantages of implementing blockchain for a criminal records system. The authors specified on several advantages of blockchain such as decentralization, agreement, agility and response time, data

2022 6th International Conference on Trends in Electronics and Informatics (ICOEI) | 978-1-6654-8328-5/22/$31.00 ©2022 IEEE | DOI: 10.1109/ICOEI53556.2022.9776725

analysis, process automation through smart contracts, and selective access. Ahmed and others [8] developed a system that could retrieve historian records of forensic data for tracking any unauthorized access and changes, thereby validating transactions and securing data via Hyperledger and making it much more secure. The authors do not use Ethereum because it was unsafe for forensic data. Instead, Hyperledger was used wherein historian records of forensic data could not be altered. Moreover, with the help of Hyperledger, the authors-imposed rrestrictions on certain participants based on their role in the system, which was only possible in a permissioned blockchain environment. Kim *et al.* [9] implemented a two-level blockchain system that included both hot and cold blockchains. Herein, it was observed that information that keeps on changing regularly was stored in hot blockchain, and the one that does not change was stored in cold blockchain (e.g., videos). The developed system ensured that only registered users were able to access blockchain that too in a decentralized environment. Moreover, the authors had split data into two blockchains, and the same transaction was stored in ledgers of all participating in the channel. Smart contracts were also used. Criminal information was created in the form of blocks, which could not be deleted by any user.

III. Background Study and Related Terminalogies

A. Blockchain

Blockchain can be counted in the list of technologies that are highly secure, scalable, transparent and flexible. It is a type of shared database that stores data in different computers connected in a network known as nodes. Every node maintains its own authentication protocol for communication, and these nodes are linked with each another in the chain.

Blockchain can primarily be classified into four types:

- Public blockchain
- Private blockchain
- Consortium or Federated blockchain
- Hybrid blockchain

This study is based on a public blockchain that offered the following advantages:

- Data stored is an immutable manner, i.e., no single authority can alter data.
- Rapid access to criminal data.
- Custom security, i.e., only admins can edit records.
- Instant traceability.

The disadvantages are as follows:

- High power consumption
- High technology system requirements
- Blockchain implementation is an expensive process

B. Ethereum

It is an open-source platform that was launched in 2015. It is a public blockchain technology best known for its native cryptocurrency known as ether [10]. Mostly, Ether is considered as a currency like dollar or Bitcoin, but apart from being a cryptocurrency, it can also be considered as a fuel for an Ethereum network. Completing a transaction on a network requires payment, and this payment is termed as gas. Gas can roughly be termed as a fee for miners to power a transaction process in blockchain. As Ethereum is a permissionless blockchain platform, the challenge aroused here were that data is openly available for all. In order to prevent this issue, data encryption was one of the solutions that came up before storing data on a blockchain and keeping the private key with the user.

C. Web3

It can be termed as the future of internet. It serves as a decentralized structure of internet and therefore has a significant use case with blockchain. In Web3, users are considered as owners; therefore, the termed coined for Web3 is read, write, and own [11]. Web3 permits a system to interact with Ethereum nodes. Such nodes could be local or remote that are connected through HTTP, inter-process communication, or web sockets.

D. IPFS

It is a file system that enables an individual to store a file and track versions of that file over time. It keeps a track of a distributed network. IPFS facilitates a new permanent web and augments the use of existing protocols such as HTTP [12]. In association with blockchain, IPFS can be used to obtain a storehouse that cannot be altered easily and does not have a door for failure. IPFS assures that entered data remains unique and protected against modifications.

E. Metamask

It is a browser extension designed to access Ethereum's Dapp ecosystem in an easy way. Basically, it is a digital wallet for Ethereum [13]. To establish connection between Ganache and the developed system, the primary step was to connect MetaMask with Ganache. This was done by setting up a local blockchain. Once the connection was established successfully, a user obtained access to the new account.

F. Ganache

It helps in establishing a personal Ethereum blockchain to test smart contracts. Ganache offers more features compared to Remix. It is a part of Truffle Suite Ecosystem [14]. The study used Ganache command line interface (CLI). With Ganache, one can easily deploy a smart contract on an emulator without using actual money for transaction cost. It also provides recyclable accounts.

G. Truffle

It is a development environment that uses Ethereum virtual machine (EVM)—a testing framework and asset pipeline for Ethereum. The slogan associated with Truffle is "*smart contracts made sweeter*" [15]. Truffle provides a developer with built-in smart contract compilation, network management for the deployment of a public and private network, etc. [16]. Truffle manages the system's entire lifecycle for a developer regardless of any platform.

H. Flutter

It is an open-source software development kit that enables a developer to develop cross-platform applications. It was developed and backed by Google. The first release of Flutter was on December 04, 2018 [17]. The system is developed using Flutter software development kit (SDK). The UI of *CRI-SYS* has been shown in the further sections of this paper.

IV. PROPOSED SYSTEM

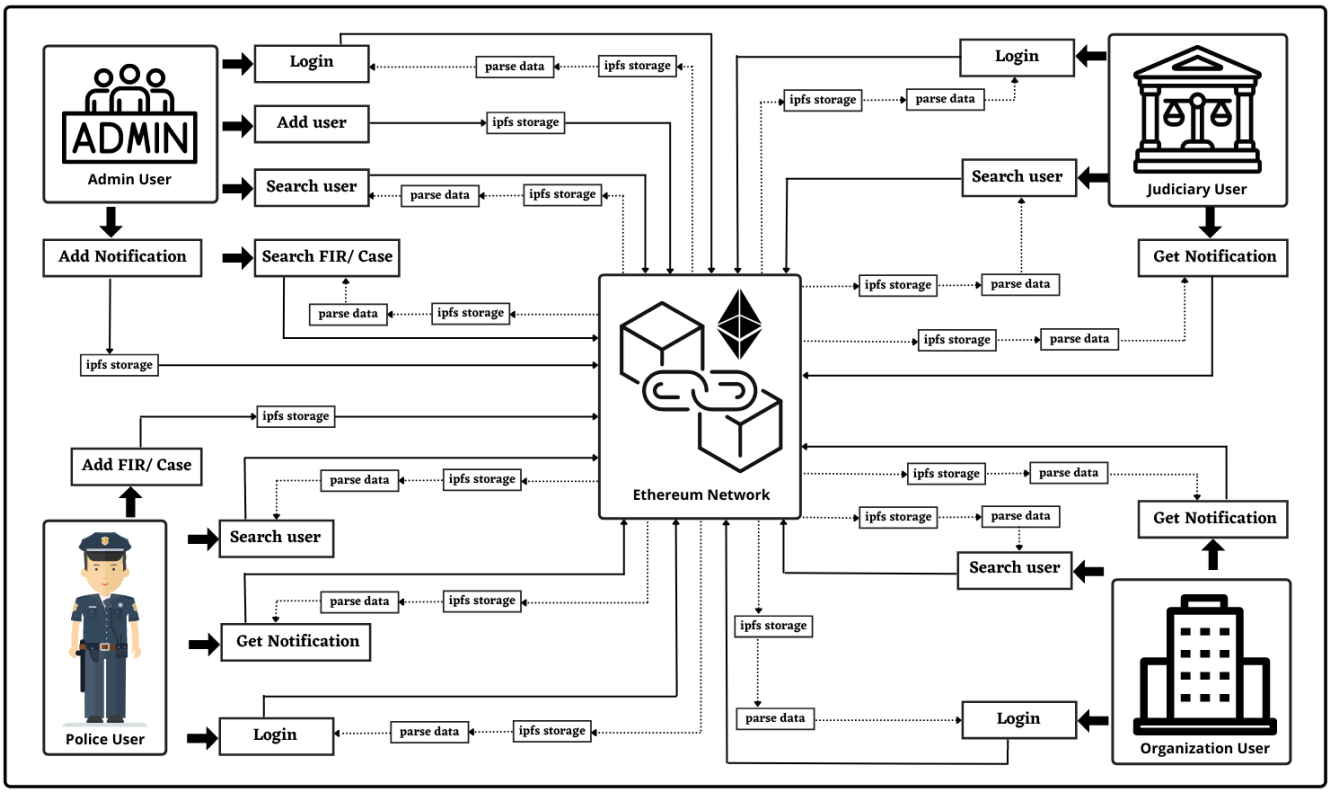


Fig. 1. Block diagram of the proposed system.

The developed system reduces police and judiciary work and provides data security. The primary goal of developing such a system is as follows:

- To provide an instantaneous FIR feature to a policeman who can file an FIR anytime and anywhere using the developed system.
- Fastest way to verify an employee's non-criminal record.
- Easy access to all FIRs and case to judiciary organization.
- Secure way to store FIRs, which cannot be hacked easily, and no one can fabricate the stored data.

The developed system mainly consists of 4 types of users:

- ***Admin user:*** It is the most superior user like state headquarters, administrators, etc. It has the access to all type of data. It can add new users to the system. An admin user can search users and FIRs and can view details. Also, an admin can send messages to all users or users of the same user type. An admin user will have the following options on its dashboard after login.

1. Add a user	2. Search FIRs in multiple ways (i.e., AADHAR, PAN, etc.)
3. Search a user	4. Add notification
5. Check notification	

  New users can request for creating an account. The admin user must fix an appointment date for a user's KYC and verification. After KYC and if all documents are found to be valid, then only user credentials will be mailed to new users.

- ***Policeman user:*** They are police authorities like inspectors and sub-inspectors. A police user can add FIR and cases to the system. It cannot add a new user. Once data is added to the system, it cannot be changed by any user. The user can easily verify a citizen's noncriminal report. A policeman user will have the following options on its dashboard after login.

1. Add an FIR/case	2. Search FIRs in multiple ways (i.e., AADHAR, PAN, etc.)
3. Check notification	

- ***Judiciary user:*** They are higher authority people who work in a judiciary like judges, lawyers, etc. They can easily get any FIR to refer their case in the court. They cannot add any data to the system. They can access all FIRs. A judiciary user will have the following options on its dashboard after login.

1. Search FIRs in multiple ways (i.e., AADHAR, PAN, etc.)	2. Check notification

- ***Organization user:*** They are big companies who deal with huge amount of employee data. They can verify their employees' noncriminal report easily. A major difference between a judiciary user and an organization user is that an organization user can only check whether a person is involved in any criminal activity or not. Unlike a judiciary user, this user can verify a person only via AADHAAR NO. An organization user will have the following options on its dashboard after login.

1. Verify an employee's noncriminal background.	2. Check notification

Following are the terminologies used in the developed system:

- **FIR:** It is a document that is filed after an incident. It is a document that is created before proving the crime of a criminal.
- **Case:** It is a document that is made after crimes are proved and a criminal is found guilty.
- **Notification:** It is a message that an *admin user* can send to all users or certain types of users. It can be some notice, instruction, or alert.

V. IMPLEMENTATION

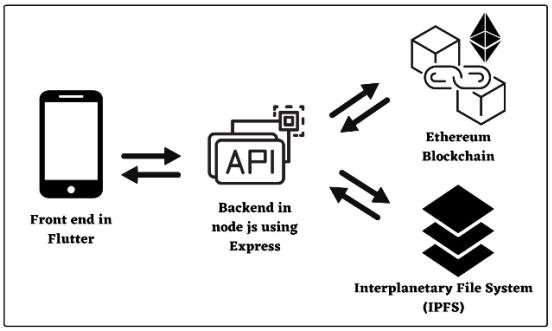


Fig. 2. Application connectivity.

Implementation of the system is divided into three parts:

A. Frontend (UI)

Since the developed system is a mobile-based system, a cross-platform framework, i.e., Flutter, is used. The developed system initially takes login from a user and directs him/her to the dashboard if the entered credentials are correct. In the dashboard, different users get different options based on their user type. If any error occurs while performing the action, then the developed system will show an error page or else a user will be directed to the result page. The system stores information such as username, JWT token, etc. in an internal storage.

B. Backened (API service)

The developed system's backend is created using NodeJS. For connecting a NodeJS application to an Ethereum node, web3js module is used, and for storing data, IPFS Infura is used. After login authentication is performed using JWT token, JWT token and user information will be stored in the system's local storage. While calling any API, the stored JWT token will be sent to the backend and will be authenticated. Before starting the backend server, the system checks for the ganache server with port 8545, and it gets the deployed contracts.

C. Blockchain (Ethereum Network)

For running an Ethereum network on a local host, ganache CLI is used which provides a few sample accounts and some gas fee for performing transactions. The developed system's smart contract is compiled and deployed on the network using truffle. Finally, in order to connect Ethereum with NodeJS, application web3 module is used. For running the developed system, the Ethereum network needs to be executed using the command *ganache-cli*. Once ganache is listening to the port 8545, the smart contract can be compiled and deployed using the command *truffle migrate*. After deploying the smart contract, the backend of the developed system can be started using the command *npm run start*. Now, the backend server is up, and several actions can be performed on the developed system. Since data can be huge and storing complete data in blockchain can be costly; so, data is stored in IPFS, and IPFS hash is stored in an Ethereum network with an ID.

D. Storing Data in IPFS and a Network

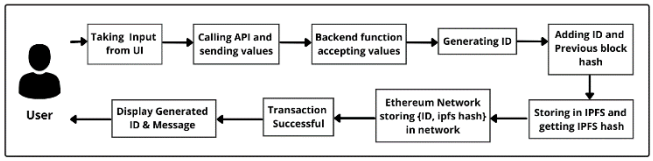


Fig. 3. Storing data in IPFS and a network.

The input is taken from a user and then the entered values are sent to backend using API in JSON format. A new ID is generated, and previous block hash are added to received data. The new data is stored in IPFS, which returns an IPFS hash. Now, the IPFS hash along with ID is stored in an Ethereum network. If the transaction is successful, then the generated ID will be sent to the user with the status as *true* and a message as *data successfully added*. If any error is found in the process, then a response will be sent to the user with a status as *failed* and a message as *failed to add data*.

E. Retrieving Data from a Network and IPFS

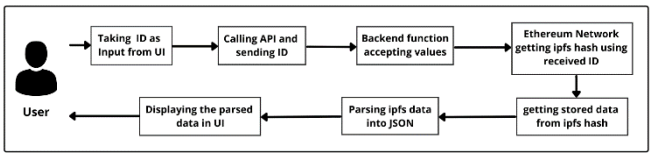


Fig. 4. Retrieving data from a network and IPFS.

For retrieving data from a network, consider a searching operation. In this, the keyword that is to be searched or ID is taken from a user. With the help of the ID, an IPFS hash is obtained by the Ethereum network; from the IPFS hash, data gets stored in IPFS. The data is in byte form, which is not readable; so, there is a need to parse that data into a JSON format. After parsing the data, it will be sent to a user with the status as *true* and a message as *data found*, and if any error is found in the process or no data is found, then the response will be sent to a user with the status as *false* and a message as *No data found*. The received data will be displayed via a GUI.

F. Data Immutablility and Transaction Security

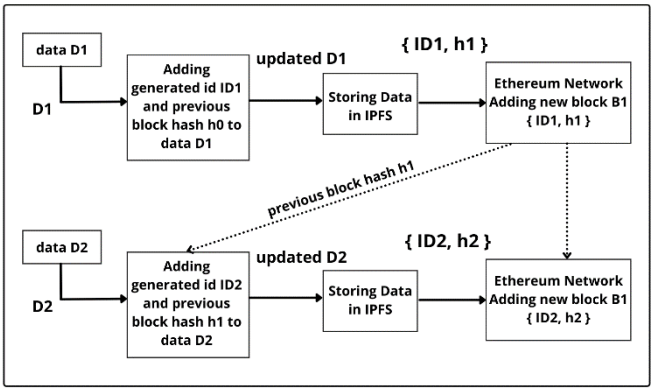


Fig. 5. Data immutability and transaction security.

In the developed system, while storing data, it is necessary to add the previous block hash along with the generated ID and store it in IPFS, which returns a unique hash using SHA-256. Next, the ID and hash are stored in the blockchain. Doing this will make data immutable, and thus, data cannot be fabricated. Assume that someone has changed data in IPFS, then IPFS hash for that data will be changed, and it will mismatch with the hash that is stored in the blockchain and in the next block. This will make the chain invalid, and it will be discarded from the network.

G. *Web3 Interaction with Etherum*

Web3 provides JavaScript API methods. GETH interactive console makes it easy to test and experiment with an Ethereum blockchain. JavaScript API fetches and shows account balance after a user provides an account address. The backend interacts with blockchain using web3.js package offered by Ethereum. One can either use his/her personal Ethereum node (e.g., `http://node.ip.addr:8545`) or INFURA's public node. One can use `http://localhost:8545` URL for a local test RPC node.

```
web3 = new Web3 (new Web3.providers.HttpProvider
        ("https://mainnet.infura.io/"));
```

Once web3 is injected, account balance information can be obtained as it is public; so, it can be accessed without any private key. The address is then queried using web3.js functions using JavaScript.

VI. Results and Discussion

The developed system eliminates the limitations of existing systems by providing a digitalized solution. The system provides a simplest way to add an FIR for police officers, a rapid way to verify employees'/peoples' noncriminal records, an easy access to all FIRs, and cases for all judiciary organization and provides a secure way to store data, which cannot be destroyed or fabricated externally. While developing *CRI-SYS*, a major challenge was to integrate Ethereum blockchain with Flutter SDK as well as node interconnection, but a closely related system helped in eliciting this challenge. However, Ethereum blockchain here is used as a permissionless blockchain where data is encrypted and made available to a user with an appropriate key. The system was tested on numerous parameters such as traceability, data authentication, real-time access, centralized repository, etc.; the developed system showed its efficiency beyond doubt. Furthermore, the system characterizes an impressive UI accomplishing the intention of making a user friendly system with an ease to use. The developed system offers users with an enhanced experience and comprehensive access to trustworthy and precise information. Below are the snapshots of the developed system.

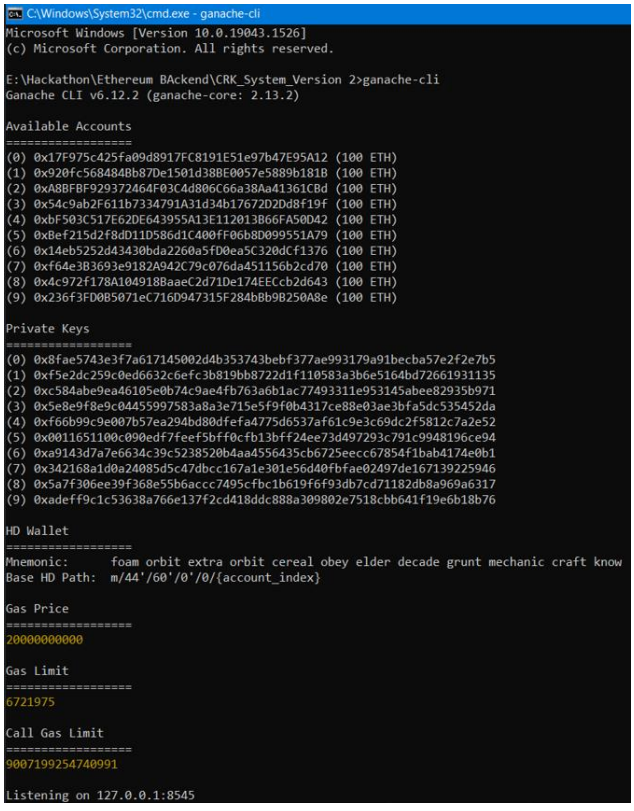


Fig. 6. Running *ganache-cli (Ethereum Network)*.

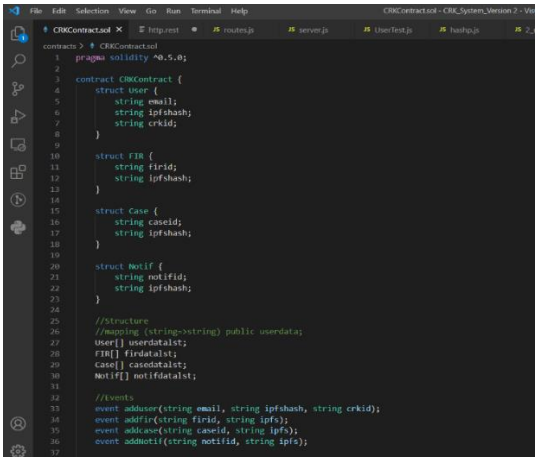


Fig. 7. Smart contract written in solidity.

Fig. 8. Deploying a smart contract to a network using truffle.

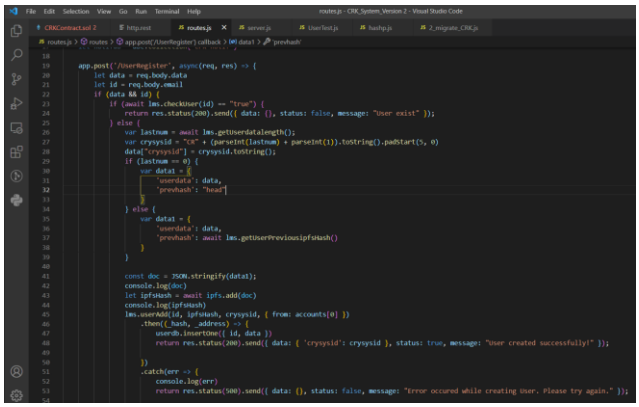


Fig. 9. Backend REST API functions.

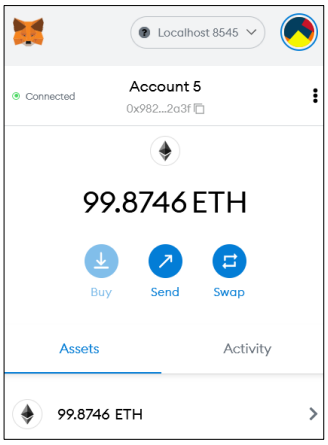
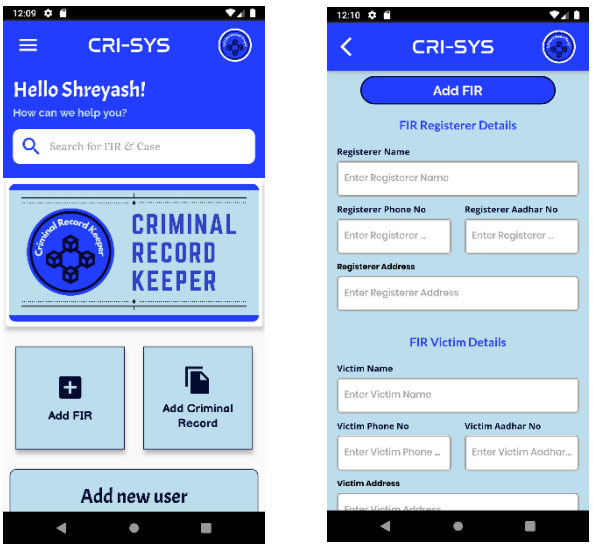


Fig. 10. MetaMask connected to a localhost.



(a) (b)

Fig. 11. (a) Dashboard page and (b) Adding of an FIR.

VII. Conclusion and Future Scope

In this technological revolution period, novels ideas are changing the way things used to be in former times. Indian law enforcement agencies still have a back foot on the technological front when it comes to lodging an FIR or verifying an individual's background. Indian law agencies state that an FIR should be filed immediately to avoid the consequences of delay and the developed system effectively supports this idea by providing an instant FIR service. The developed system ensures transparency and confidentiality of stored data. The developed system will serve as a helping hand to the Indian E-governance system. The system works on an Ethereum platform, which is a permissionless blockchain, and so, security could be compromised to a certain extent. The future scope would be to use a private blockchain such as Hyperledger Fabric to make the network more reliable and secure.

References

[1] "Crime in India", National Crime Records Bureau, Ministry of Home Affairs, [Online], Available: https://ncrb.gov.in/sites/default/files/CII%202020%20Volume%201.pdf (Accessed on April 18, 2022).

[2] M. A. Tasnim, A. O. Abdullah, M. S. Rahman, Md. Bhuiyan, and Z. Alam, "Crab: Blockchain based criminal record management system," *Int. Conf. Security, Privacy and Anonymity in Computation, Communication and Storage*, pp. 294–303. Springer, Cham, 2018.

[3] M. Ahmed, S. Reno, N. Akter, and F. Haque, "Securing medical forensic system using hyperledger based private blockchain," *In 2020 23rd Int. Conf. Computer and Information Technology (ICCIT)*, pp. 1–6, IEEE, 2020.

[4] I. Hingorani, R. Khara, D. Pomendkar, and N. Raul, "Police complaint management system using blockchain technology," *In 2020 3rd Int. Conf. Intelligent Sustainable Systems (ICISS)*, pp. 1214–1219, IEEE, 2020.

[5] P. Shukla, R. Tyagi, and A. Tyagi, "Implementation of blockchain on criminality record checker," *Int. J. Engineering Research & Technology (IJERT)*, vol. 09, no. 04, pp. 682–686, April 2020.

[6] D. Bhushan, M. Gangarde, A. Singh, J. Pawar, and S. Dhanake, "Blockahin-based crime record management system," *J. Composition Theory (JAC)*, vol. 13, no. 7, July 2020.

[7] A. T. Dini, E. G. Abete, M. Colombo, J. Guevara, B. S. M. Hoffmann, and M. C. Abeledo, "Analysis of implementing blockchain technology to the argentinian criminal records information system," *In 2018 Congreso Argentino de Ciencias de la Informática y Desarrollos de Investigación (CACIDI)*, pp. 1–3, IEEE, 2018.

[8] N. D. Khan, C. Chrysostomou, and B. Nazir, "Smart FIR: Securing e-FIR data through blockchain within smart cities," *In 2020 IEEE 91st vehicular technology conference (VTC2020-Spring)*, pp. 1–5. IEEE, 2020.

[9] D. Kim, S.-Y. Ihm, and Y. Son, "Two-level blockchain system for digital crime evidence management," *Sensors* 21, no. 9 (2021): 3051.

[10] "What is Ethereum?" [Online] Available: https://ethereum.org/en/what-is-ethereum/ (Accessed on April 18, 2022).

[11] Introduction to Web3. [Online] Available: https://www.dappuniversity.com/articles/web3-js-intro (Accessed on April 18, 2022).

[12] A. Rajalakshmi, K. Lakshmy, M. Sindhu, and P. Amritha, "A blockchain and IPFS-based framework for secure research record keeping," *Int. J. Pure and Applied Mathematics*, vol. 119, no. 15 (2018), pp. 1437–1442.

[13] "What is MetaMask?" [Online] Available: https://metamask.io/ (Accessed on April 18, 2022).

[14] "Ganache," [Online] Available: https://trufflesuite.com/docs/ganache/index.html#:~:text=Ganache%20is%20a%20personal%20blockchain,flavors%3A%20a%20UI%20and%20CLI (Accessed on April 18, 2022).

[15] "Truffle," [Online] Available: https://moralis.io/truffle-explained-what-is-the-truffle-suite/ (Accessed on April 18, 2022).

[16] https://github.com/trufflesuite/truffle

[17] S. Boukhary, and E. Colmenares, "A clean approach to flutter development through the flutter clean architecture package," *In 2019 Int. Conf. on Computational Science and Computational Intelligence (CSCI)*, IEEE, pp. 1115–1120, 2019.