IJIRCCE

e-ISSN: 2320-9801 | p-ISSN: 2320-9798

INTERNATIONAL JOURNAL OF INNOVATIVE RESEARCH

IN COMPUTER & COMMUNICATION ENGINEERING

Volume 9, Issue 6, June 2021

INTERNATIONAL STANDARD SERIAL NUMBER INDIA

Impact Factor: 7.542

International Journal of Innovative Research in Computer and Communication Engineering

| e-ISSN: 2320-9801, p-ISSN: 2320-9798| www.ijircce.com | |Impact Factor: 7.542 |

|| Volume 9, Issue 6, June 2021 ||

| DOI: 10.15680/IJIRCCE.2021.0906206 |

Implementation of "Identity Chain- Identity Verification Framework Using Ethereum"

Shubham Vijay Vhavale[1], Shruti Vasant Daundkar[2], Yash Devising Vishwakarma[3], Prof.Yogendra Patil[4]

Student, Dept. of Computer Engineering, JSPM's Bhivarabai Sawant Institute of Technology & Research, Pune, India[1]

Student, Dept. of Computer Engineering, JSPM's Bhivarabai Sawant Institute of Technology & Research, Pune, India[2]

Student, Dept. of Computer Engineering, JSPM's Bhivarabai Sawant Institute of Technology & Research, Pune, India[3]

Associate Professor, & Guide, Dept. of Computer Engineering, JSPM's Bhivarabai Sawant Institute of Technology & Research, Pune, India[4]

ABSTRACT: In any of the registration processes, it is a hassle to always carry a physical document. Not only that, the process is extended if they are lost, and if unauthorized persons have access to those documents then it will contribute to identity theft, and apart from that India has a huge population. Thus the process of authenticating the identity of every citizen should be reliable, fast, and secure. The current systems in place are functional, but they prove security breaches and data tampering. The current process usually takes several weeks for authenticating a citizen's identity with the government bodies. It is inconvenient, time-consuming, and environmentally expensive. Our proposed system will overcome these shortcomings of existing systems with the help of Blockchain Technology. Our objective is to design a blockchain-enabled Identity Verification System Framework that helps solve identity verification issues and increase ease of access to documents. We generate a decentralized system, using the concept of blockchain, to allow registered persons to access user's documents. In conclusion, This project enhances the area of Identity verification that is vital in helping society to control their details and digitizes the personal identities of the people using blockchain. An identity verification system that increases the trust that data is secured and it's hard to hack or change data.

KEYWORDS: Blockchain, Ethereum, Security.

I. INTRODUCTION

Blockchain is a chain of interconnected blocks that contains information, This technique originally described in 1991 by a group of researchers was originally intended to timestamps the digital documents so it's not possible to back-stamp them or to tamper them like notary but went unused until it was adapted by Satoshi Nakamoto in 2008 to create a digital cryptocurrency bitcoin, Blockchain is distributed ledger that is completely open to anyone have interesting properties one data is inserted into the blockchain it becomes very difficult to change it. It was the founding technology initially used in Bitcoin, a digital currency. Blockchain has since seeped its way into various domains. Blockchain is a decentralized peer-based network, where each peer, also called a node, stores information about all the transactional records of the network. Every transaction is linked to the previous node on the chain, and before being stored needs consensus or also called quorum from more than half the nodes on the chain. Hence, the trust, immutability, and security are high as changing any block of the chain would need previous blocks to be changed too, which is not very feasible to attackers.

India has a large population of nearly 134 crores. With a population that high, the process of issuing and verifying the identity-based documents such as (Passport, Adhaar Card, PAN Card, Voters ID, etc.) of every citizen must be reliable, secure, and quick. The current systems in place are functional, but the efficiency and security need to be improved drastically as the process usually takes several weeks and the citizens applying for the documents may have to visit the issuing authority offices multiple times to get documents done. This is not only inconvenient and time-consuming but also monetarily and environmentally expensive.

In this report, we have discussed a blockchain-based solution for verifying the authenticity of the documents issued by the Indian Government authorities. The advantage of using this system is that it provides a quick, reliable, and secure channel for issuing authorities to access documents of an individual who has other documents directly from the databases of The other issuing authorities. This access is permission and time-bounded so privacy concerns are eliminated.

| e-ISSN: 2320-9801, p-ISSN: 2320-9798| www.ijircce.com | |Impact Factor: 7.542 |

|| Volume 9, Issue 6, June 2021 ||

| DOI: 10.15680/IJIRCCE.2021.0906206 |

II. PROBLEM DEFINITION AND OBJECTIVE

Our objective is to design a blockchain-enabled Identity Verification System Framework which is user-friendly, fast, secure, and helps to solve identity verification issues and increase ease of access to documents.

III. PROJECT SCOPE

In this century, the strategies used in creating verified documents are very slow and time-consuming, thus, building a straightforward web application isn't ideal for this reason because the data can be hacked, the data rules can be changed and the trust of the citizens is lost. Hence we are changing the verifying process. A blockchain-based System Framework that isn't just secure from defilement, yet additionally gives solid protection from hacking and various cybercrimes. Using this technology we can work well under budget and in a limited time, we can achieve more. The most significant factor of this application is that not even the software engineers can modify the data once put together by an authorized body. It easy to sign into, doable, and simple to utilize the straightforward application

IV. USER CLASS AND CHARACTERISTICS

- Authorized User/ Manager: A manager is an end-user who can enter the credentials of documents to check if that person is authorized or not and also he/she can view the documents of any person.
- Nodal Authorizer: Nodal Authorizer is an admin who is a higher authority or a police officer, who can not only add or update the documents of any citizen but also verify that the person is authorized or not and if any person is not authorized then flag that document and inform the manager and police.

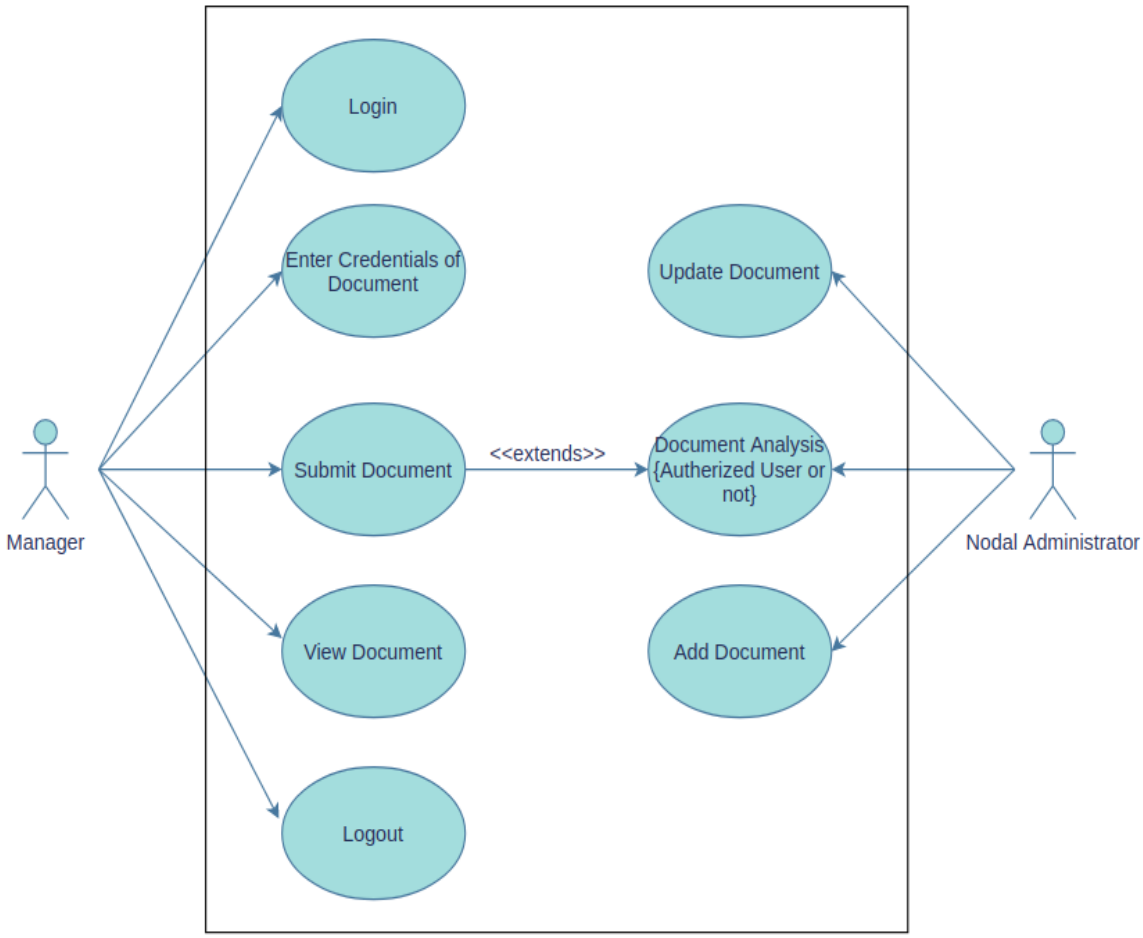

Fig 1: Use Case Diagram

| e-ISSN: 2320-9801, p-ISSN: 2320-9798| www.ijircce.com | |Impact Factor: 7.542 |

|| Volume 9, Issue 6, June 2021 ||

| DOI: 10.15680/IJIRCCE.2021.0906206 |

V. SYSTEM ARCHITECTURE

- **System Flow**
  This system assists in automating the existing manual system. This is paperless work. It can be monitored and controlled remotely. It reduces the manpower required. It always provides accurate information. Malpractice can be reduced.
  - Authentication: This system will authenticate every citizen and verify that the person whom they say they are, is in the real world or not.
  - Decentralized database: In that System, we are using a blockchain which is the most secure decentralized database, and also solve the disadvantages of a centralized system.
  - Fast: The current system has taken 10 to 15 days to verify the user's identity, our proposed system will take only a few minutes to verify the user.
  - Authorized User where it can be used to verify one's identity.
  - The authorized body will add the data into the nodes.

VI.TOOLS AND TECHNOLOGY USED

- **Front End:**
  - HTML, CSS, JavaScript, Bootstrap, Angular.
- **Back-End:**
  - Blockchain (Ethereum)
  - Solidity
  - Meta Mask
  - Node.js
  - Ganache
  - Truffle
  - Remix IDE

VII. TOOLS AND TECHNOLOGY USED

- **What are smart contracts and how they work?**

  Smart contracts are simply programs stored on a blockchain that run when predetermined conditions are met. They typically are used to automate the execution of an agreement so that all participants can be immediately certain of the outcome, without an intermediary's involvement or time loss. They can also automate a workflow, triggering the next action when conditions are met.

  Smart contracts work by following simple "if/when…then…" statements that are written into code on a Blockchain. A network of computers executes the actions when predetermined conditions have been met and verified. These actions could include releasing funds to the appropriate parties, registering a vehicle, sending notifications, or issuing a ticket. The blockchain is then updated when the transaction is completed. That means the transaction cannot be changed, and only parties who have been granted permission can see the results. Within a smart contract, there can be as many stipulations as needed to satisfy the participants that the task will be completed satisfactorily. To establish the terms, participants must determine how transactions and their data are represented on the blockchain, agree on the "if/when...then…" rules that govern those transactions, explore all possible exceptions, and define a framework for resolving disputes. Then the smart contract can be programmed by a developer – although increasingly, organizations that use blockchain for business provide templates, web interfaces, and other online tools to simplify structuring smart contracts.

- **Identity Check Smart Contract :**

```
// SPDX-License-Identifier: GPL-3.0
pragma solidity >=0.4.22 <0.9.0;
pragma experimental ABIEncoderV2;

contract IdentityChain{
   address private owner;
   enum  aadhaar_passport_both{AADHAAR,AADHAAR_PASSPORT}
   enum  flag{GREEN,ORANGE,RED}
   struct aadhaar {...}
```

| e-ISSN: 2320-9801, p-ISSN: 2320-9798| www.ijircce.com | |Impact Factor: 7.542 |

|| Volume 9, Issue 6, June 2021 ||

| DOI: 10.15680/IJIRCCE.2021.0906206 |

```
struct passport{...}
struct identity{... }
struct log{...}
struct search_record_struct{... }
mapping (uint64 => identity) private identity_mapped;
mapping (uint64 => uint ) private aadhaar_exists;
mapping (string => uint ) private passport_exists;
mapping(string => uint ) private passport_to_aadhaar_mapping;
mapping(uint64 => search_record_struct) private search_records_aadhaar;
mapping (uint64 => uint ) private aadhaar_records_exists;
string[] private passport_identity_data;
uint64[] private aadhaar_identity_data;
constructor() public {    owner = msg.sender;    }

//--------------------Inserting Data into Blockchain-----------------------------------
 function add_aadhaar_details(
     uint64 aadhaar_number, string memory aadhaar_name, uint aadhaar_dob_year,
     string memory aadhaar_address, string memory  aadhaar_sex
     )   public only_owner check_aadhaar(aadhaar_number,true)
     returns (bool added_aadhaar){... }

function add_passport_using_aadhaar(
    uint64 aadhaar_number,string memory passport_number,string  memory passport_name,
    uint passport_dob_year,  uint passport_expiry_year, string memory old_passport_number)
    public only_owner check_aadhaar(aadhaar_number,false) check_passport(passport_number,true)
    returns (bool aaded_passport_using_aadhaar){...}

function update_criminal_flag(uint64 aadhaar_number,string memory criminal_flag)
  public check_aadhaar(aadhaar_number,false) check_flag(aadhaar_number,criminal_flag)
  returns (bool flag_updated){...}

function add_log_using_aadhaar(
    uint64 aadhaar_number, string memory location,string memory reason)
    public check_aadhaar(aadhaar_number,false)  returns(bool){...}

function add_log_using_passport(
    string memory passport_number, string memory location,string memory reason)
    public check_passport(passport_number,false)  returns(bool){...}

//-------------------------View Only methods-------------------------------
function view_aadhaar_passport_status(uint64 aadhaar_number)
    public view  returns (bool status_aadhar_only){...}

function view_aadhaar_details(uint64 aadhaar_number)
    public view  check_aadhaar(aadhaar_number,false)
    returns( uint64 , string memory , uint , string memory , string memory  ){...}

function view_passport_using_aadhaar(uint64 aadhaar_num)
    public view check_aadhaar(aadhaar_num,false)  check_passport_present_status(aadhaar_num)
    returns(string memory passport_number,string  memory passport_name,uint passport_dob_year,
    uint passport_expiry_year, string memory old_passport_number ){...}

function view_criminal_flag(bool is_aadhaar,uint64 aadhaar_number,string memory passport_number)
    public view check_aadhaar_passport_present_status( is_aadhaar, aadhaar_number, passport_number)
    returns (string memory criminal_flag_result){...}

function view_log_using_aadhaar(uint64 aadhaar_number) public view
```

| e-ISSN: 2320-9801, p-ISSN: 2320-9798| www.ijircce.com | |Impact Factor: 7.542 |

|| Volume 9, Issue 6, June 2021 ||

| DOI: 10.15680/IJIRCCE.2021.0906206 |

```
    check_aadhaar(aadhaar_number,false) check_log_present(aadhaar_number)
    returns(uint count,log[] memory log_array){...}

  function view_log_using_passport(string memory passport_number) public view
    check_passport(passport_number,false) returns(uint count,log[] memory log_array){...}

  //-----------------------------------Modifiers and Pure functions-----------------------------------
  modifier only_owner(){  require(owner==msg.sender,"Access denied!");    _;  }
  modifier check_aadhaar(uint64 aadhaar_number, bool b){...}
  modifier check_passport(string memory passport_number, bool b){...}
  modifier check_passport_present_status(uint64 aadhaar_number){...}
  modifier check_aadhaar_passport_present_status(bool is_aadhaar,uint64 aadhaar_number,string memory
passport_number){...}
  modifier check_flag(uint64 aadhaar_number,string memory criminal_flag){...}
  modifier check_log_present(uint64 aadhaar_number){... }
  function flag_to_string(flag f) private pure  returns (string memory flag_in_string){...}
  function compareStringsbyBytes(string memory s1, string  memory s2) private pure returns(bool{... }
}
```

VIII. PROJECT SCREENSHOTS

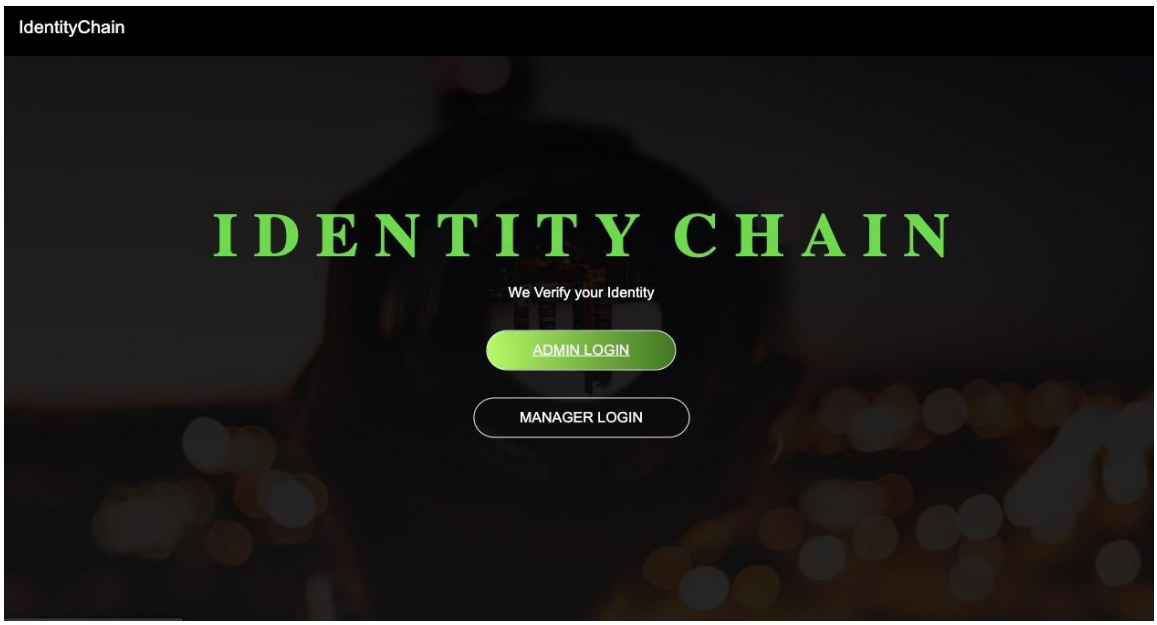

Fig 2: Homepage

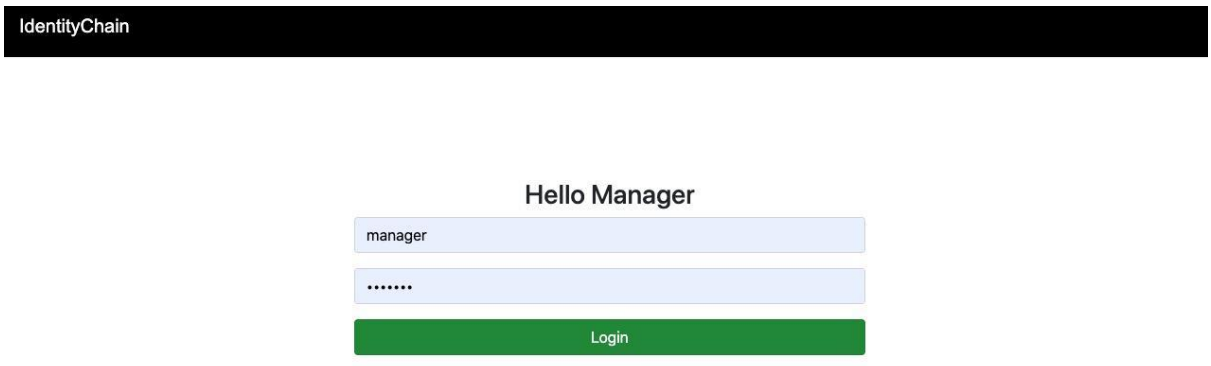

Fig 3: Manager's Login Page

| e-ISSN: 2320-9801, p-ISSN: 2320-9798| www.ijircce.com | |Impact Factor: 7.542 |

|| Volume 9, Issue 6, June 2021 ||

| DOI: 10.15680/IJIRCCE.2021.0906206 |

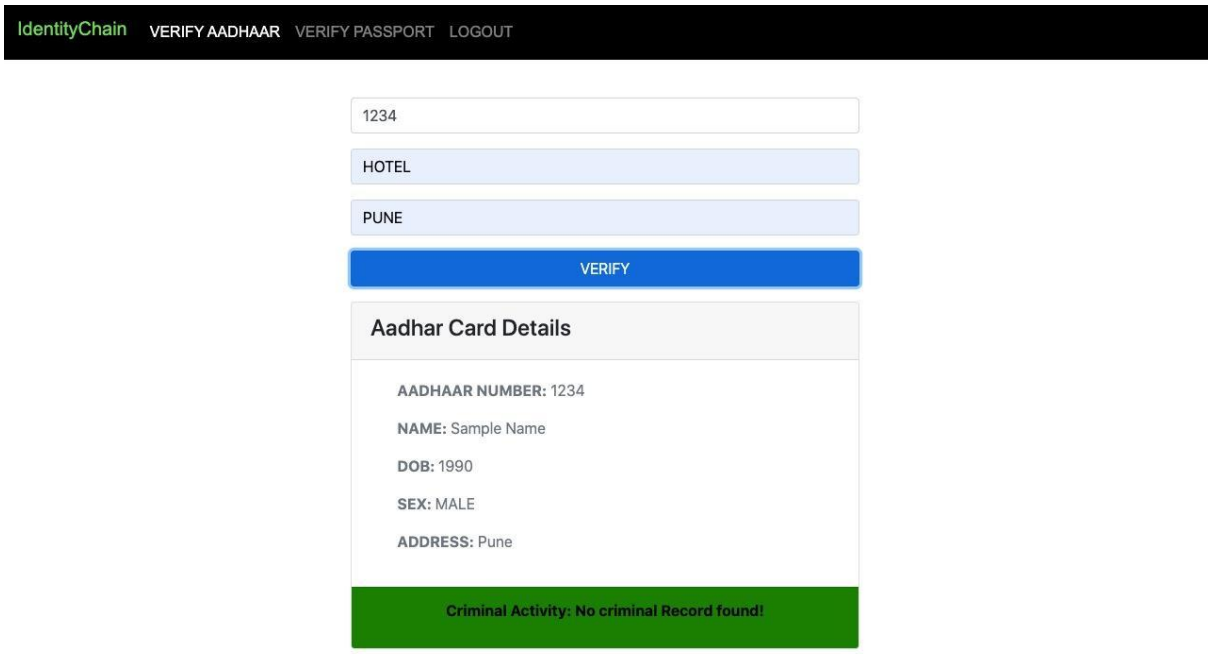

Fig 4: Manager's Option 1

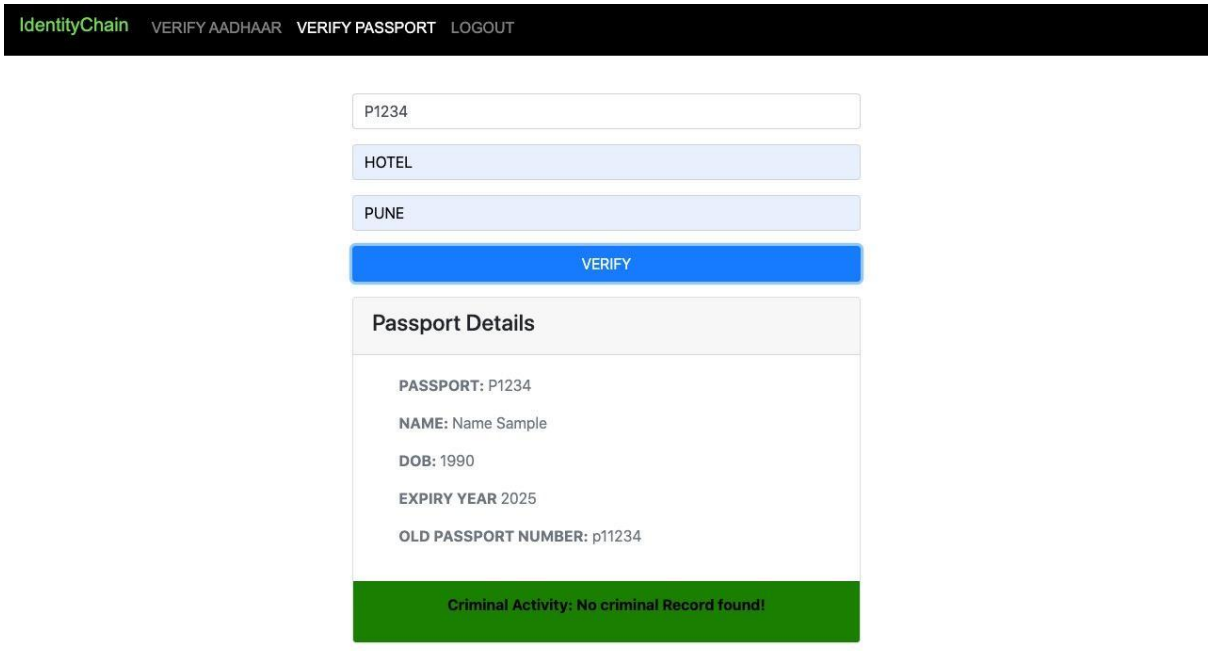

Fig 5: Manager's Option 2

IX. ADVANTAGES

- As the default record keeper for society, Government entities are large targets for hackers. But rather than accepting such attacks as the cost of doing business in the information era, they could be avoided through the responsible deployment of blockchain data structures.
- Because of the blockchain, the system is Decentralized, Transparent, and Immutable.
- This Framework is Efficient, Tamper-proof, and Reliable

| e-ISSN: 2320-9801, p-ISSN: 2320-9798| www.ijircce.com | |Impact Factor: 7.542 |

|| Volume 9, Issue 6, June 2021 ||

| DOI: 10.15680/IJIRCCE.2021.0906206 |

X. LIMITATIONS

- If wrong data is stored in the UIDAI database then the system might not be able to correctly authenticate the user. As this framework would receive user credentials from such databases.

XI. APPLICATION

- The IdentityChain has numerous applications. We can use this framework in all types of elections as it provides a means to authenticate a user's identity. It would be the most feasible & effective option in situations such as electing
  - Organizational leaders,
  - Departmental Heads, etc.
- This framework can be used in any government activity where the user/candidate needs to Verify.
- This Framework can be used in the recruitment process of various organizations.
- Also, this framework can be used to verify guests checking in to a hotel.

XII. CONCLUSION

The idea of adapting the Ethereum Based Identity Verification System framework is to make the Identity Verification System secure and user friendly because the traditional system has limitations of identity theft, data loss, and also these systems take 10 to 15 days to authenticate the documents. To avoid those shortcomings we have developed the Ethereum Based Identity Verification System which can solve those shortcomings using blockchain technology.

REFERENCES

1. Arshad Jamal, Rabab Alayham Abbas Helmi, Ampuan Siti Nurin Syahirah, Mariam-Aisha Fatima "Blockchain-Based Identity Verification System" 2019 IEEE 9th International Conference on System Engineering and Technology (ICSET)**.**
2. H. Do and W. Ng, "Blockchain-based System for Secure Data Storage with Private Keyword Search", 2017 IEEE 13Th World Congress On Services, 2017.
3. Yeoh, "Regulatory issues in blockchain technology", Journal Of Financial Regulation And Compliance, vol. 25, no. 2, pp. 196-208, 2017.

INNO SPACE
SJIF Scientific Journal Impact Factor
Impact Factor: 7.542

doi cross ref

ISSN INTERNATIONAL STANDARD SERIAL NUMBER INDIA

निस्केयर NISCAIR

INTERNATIONAL JOURNAL OF INNOVATIVE RESEARCH

IN COMPUTER & COMMUNICATION ENGINEERING

9940 572 462 6381 907 438 ijircce@gmail.com

Scan to save the contact details