

AI BASED SECURED VOTING SYSTEM WITH BLOCKCHAIN MINED DATA STORAGE

Submitted in partial fulfillment of the requirements for the award of
Bachelor of Engineering degree in Computer Science and
Engineering

By

BAIREDDY VENKATA DEVENDHAR REDDY(Reg.No-40110144)

BATTULA SAHITH REDDY (Reg. No - 40110164)



**DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING
SCHOOL OF COMPUTING**

SATHYABAMA

**INSTITUTE OF SCIENCE AND TECHNOLOGY
(DEEMED TO BE UNIVERSITY)**

CATEGORY-1 UNIVERSITY BY UGC

Accredited with Grade "A++" by NACC | 12B Status by UGC | Approved by AICTE

**JEPPIAARNAGAR, RAJIV GANDHI SALAI,
CHENNAI - 600119**

APRIL-2024



SATHYABAMA

INSTITUTE OF SCIENCE AND TECHNOLOGY

(DEEMED TO BE UNIVERSITY)

Category - I University by UGC

Accredited "A++" by NAAC | Approved by AICTE

www.sathyabama.ac.in

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

BONAFIDE CERTIFICATE

This is to certify that this Project Report is the bonafide work of Baireddy Venkata Devendhar Reddy (40110144) and Battula Sahith Reddy (40110164) who carried out the Project Report entitled "AI Based Secured Voting System With Blockchain Mined Data Storage" under my supervision from November 2023 to April 2024.

D. Nancy

Internal Guide

Ms. NANCY KIRUPANITHI, MTech.,

[Signature]

Head of the Department

Dr. L. LAKSHMANAN, M.E., Ph.D.

SATHYABAMA
INSTITUTE OF SCIENCE AND TECHNOLOGY
(DEEMED TO BE UNIVERSITY)
Medaplaa Nagar, Rajiv Gandhi Salai,
Chennai - 600 019.

Submitted for Viva Voce Examination held on 12-4-24

[Signature]
12/4/24
Internal Examiner

[Signature]
12/4/2024
External Examiner

DECLARATION

I, **Battula Sahith Reddy (Reg.No-40110164)**, hereby declare that the Project Report entitled "**AI Based Secured Voting System With Blockchain Mined Data Stroage**" done by me under the guidance of **Ms.Nancy Kirupanithi, MTech.**, is submitted in partial fulfillment of the requirements for the award of Bachelor of Engineering degree in **Computer Science and Engineering**.

DATE: 12-4-2024

PLACE: Chennai



SIGNATURE OF THE CANDIDATE

ACKNOWLEDGEMENT

I am pleased to acknowledge my sincere thanks to **Board Of Management of Sathyabama Institute Of Science And Technology** for their kind encouragement in doing this project and for completing it successfully. I am grateful to them.

I convey my thanks to **Dr.T.Sasikala M.E., Ph.D, Dean, School of Computing, Dr. L. Lakshmanan M.E., Ph.D.,** Head of the Department of Computer Science and Engineering for providing me necessary support and details at the right time during the progressive reviews.

I would like to express my sincere and deep sense of gratitude to my Project Guide **Ms.Nancy Kirupanithi, M.Tech.** for his valuable guidance, suggestions and constant encouragement paved way for the successful completion of my phase-1 project work.

I wish to express my thanks to all Teaching and Non-teaching staff members of the **Department of Computer Science and Engineering** who were helpful in many ways for the completion of the project.

ABSTRACT

Electronic Voting is the standard means of conducting elections using Electronic Voting Machines, sometimes called "EVMs" in India. The use of EVMs and electronic voting was developed and tested by the state-owned Electronics Corporation of India and Bharat Electronics in the 1990s. They were introduced in Indian elections between 1998 and 2001, in a phased manner. The electronic voting machines have been used in all general and state assembly elections of India since 2014. In India, the voting system generally uses the manual approach where voters queue up in a physical space to cast their votes for their choices. Manual voting system without any doubt does not lead to 100% voting rate. This project primarily focuses to provide the people an authentic scientific community with parties' policy positions, with respect to their constituencies. A web application is developed using reactJS that aims to analyze the role of the existing political parties/independents who are contesting in the forthcoming state election. The project also helps the common man to take a survey with the questionnaire to express his/her willingness and views on the current/forthcoming state election. This web application also helps the state government in achieving 100% voting rate in the state elections by providing an e-voting system enabling Face Detection authentication.

TABLE OF CONTENTS

CHAPTER NO.	TITLE	PAGE NO.
	ABSTRACT	V
	LIST OF FIGURES	VIII
1	INTRODUCTION	1
	1.1 Background and Motivation	2
	1.2 Problem Statement	3
	1.3 Objectives and Scope	3
2	LITERATURE SURVEY	5
	2.1 Overview of Existing Research and Literature	5
	2.2 Inferences from Literature Survey	9
	2.3 Open Problems in Existing System	10
3	REQUIREMENTS ANALYSIS	12
	3.1 Feasibility Studies of the Project	12
	3.1.1 Economic Feasibility	12
	3.1.2 Technical Feasibility	12
	3.1.3 Operational Feasibility	13
	3.2 Requirements Specification	13
	3.2.1 Hardware Requirements	13
	3.2.2 Software Requirements	13
4	DESCRIPTION OF PROPOSED SYSTEM	14
	4.1 Description of research approach and methods	14
Face Detection	15	
	Face Recognition	16
	Node API Generation	17
	Mongodb Generation	18
	Web Application Development	21
	Blockchain Integration	22
	Architecture or proposed system	24
	Description of software for Implementation and Testing Plan of the Proposed Model/System	26

5	IMPLEMENTATIONDETAILS	29
	Implementation	29
	Algorithms	29
	HaarCascade	29
	SupportVectorMachine	30
6	RESULTSANDDISCUSSIONS	31
	Results	31
7	CONCLUSION	36
	Conclusion	36
	Futurework	36
	REFERENCES	37
	APPENDIX	39
	A. SOURCECODE	39
	B. SCREENSHOTS	55
	C. RESEARCHPAPER	56

LISTOFFIGURES

FIGURENO	FIGURENAME	PAGENO
4.1	HaarCascade	16
4.2	SVMAlgorithm	17
4.3	NodeAPIIntegration	18
4.4	MongoDBIntegration	21
4.5	WebApplicationDevelopment	22
4.6	BlockchainIntegration	23
4.7	ProposedSystemArchitecture	24
4.8	WebArchitecture	25
6.1	Loginpage	31
6.2	Candidateregistrationform	32
6.3	Userregistrationform	32
6.4	AdminHomepage	33
6.5	Userhomescreenwithcandidatelist	33
6.6	Candidatehomescreen	34
6.7	Voteplaced	34
6.8	ElectionResults	35
6.9	BlockchainStatusisgood	35
B.1	Enrollingfaces	55
B.2	Encodingfaceembedding	55
B.3	TrainingfacesembeddingusingSVM	56
B.4	FaceRecognition	56
B.5	VotePlaced	57
B.6	ElectionResults	57

CHAPTER 1

INTRODUCTION

India is a country, divided into states and union territories, with a parliamentary system governed under the Constitution of India, which defines the power distribution among the federal government and the states. The Election Commission of India is an autonomous constitutional authority responsible for administering Union and State election processes in India. The body administers elections to the Lok Sabha, Rajya Sabha, State Legislative Assemblies in India, and the offices of the President and Vice President in the country.

India is a sovereign, socialist, secular, democratic republic. Democracy runs like a golden thread in the social, economic and political fabric woven by the Constitution given by 'We, the People of India' unto ourselves. The concept of democracy as visualised by the Constitution pre-supposes the representation of the people in Parliament and State legislatures by the method of election. The Supreme Court has held that democracy is one of the inalienable basic features of the Constitution of India and forms part of its basic structure. The Constitution of India adopted a Parliamentary form of government. Parliament consists of the President of India and the two Houses, Rajya Sabha and Lok Sabha. India, being a Union of states, has separate state legislatures for each state. State legislatures consist of the Governor and two Houses, Legislative Council and Legislative Assembly in seven states, namely, Andhra Pradesh, Telangana, Bihar, Jammu & Kashmir, Karnataka, Maharashtra and Uttar Pradesh, and of the Governor and the state Legislative Assembly in the remaining 22 states. Apart from the above, two out of the seven Union Territories, namely, National Capital Territory of Delhi and Puducherry, also have their Legislative Assemblies.

Election Commission is the federal body of India which is enacted under the provisions of the Constitution, responsible for monitoring and administering all the electoral processes of India. This body is responsible for ensuring elections

are free and fair, without any bias. The Election Commission ensures the conduct of members pre-elections, during elections, and post-elections are as per the statutory legislation. The "AI-Based Secured Voting System with Blockchain Mined Data Storage" project represents a groundbreaking initiative at the intersection of advanced technologies, security, and the democratic process. In an era marked by the critical importance of secure and transparent elections, this project seeks to revolutionize the way we conduct voting, ensuring the utmost integrity and accessibility.

Traditional voting systems have faced challenges related to security vulnerabilities and the need for manual verification, often leading to concerns over election integrity. In response to these challenges, this project envisions a novel voting system that leverages cutting-edge technologies: artificial intelligence (AI) and blockchain.

At its core, the project aims to create a secure, tamper-proof, and transparent platform for casting and recording votes, all while maintaining the anonymity of the voter. The integration of blockchain technology, known for its immutable ledger capabilities, guarantees that voting records remain impervious to alteration and provide an auditable history of the electoral process. Meanwhile, AI will play a pivotal role in ensuring voter anonymity, accessibility, and identity verification, contributing to a more inclusive and secure voting experience.

BACKGROUND AND MOTIVATION

The genesis of the project lies in addressing the persistent issue of incomplete voter turnout despite the widespread use of EVMs in Indian elections. Recognizing the potential of technology to bridge this gap, the initiative aims to leverage ReactJS to develop a user-friendly web application. By providing voters with easy access to parties' policy positions specific to their constituencies, the project seeks to empower them with informed choices. Through a survey mechanism, citizens can actively participate in shaping the

political landscape by expressing their opinions and preferences. The integration of an e-voting system, fortified with Face Detection authentication, underscores the commitment to modernize electoral practices and ensure their integrity. This initiative is rooted in the vision of fostering a more robust and inclusive democracy in India, where every citizen's voice is heard and their vote counts toward shaping the future of their state.

PROBLEM STATEMENT

Despite the widespread adoption of Electronic Voting Machines (EVMs) in Indian elections, there persists a significant gap in voter turnout, primarily due to the reliance on manual voting systems. This discrepancy undermines the democratic process and hinders the full representation of citizens' voices in state elections. The lack of easily accessible information about parties' policy positions tailored to specific constituencies further exacerbates this issue, leading to uninformed voting choices. Additionally, the absence of modernized voting mechanisms, such as e-voting with secure authentication features, limits the potential to achieve a higher voter turnout rate. Consequently, there is a pressing need to develop a comprehensive solution that leverages technology to address these challenges effectively. The solution should encompass the development of a user-friendly web application using ReactJS, offering voters access to parties' policy positions, facilitating informed decision-making. Furthermore, the solution should incorporate a survey mechanism to enable citizens to express their views and preferences on the upcoming state election. Moreover, the integration of an e-voting system with advanced authentication features, such as Face Detection, is essential to modernize the voting process and enhance its integrity. By addressing these critical issues, the proposed solution aims to promote greater electoral participation and ensure a more inclusive and transparent democratic process in India's state elections.

OBJECTIVE AND SCOPE

Objective

The objective of the project is to develop an effective technological solution

to empower citizens across the country with awareness about their political

candidates. This includes providing comprehensive information about political parties and independents contesting in forthcoming state elections, encompassing their policy positions, past performance, and relevant details. Through this initiative, voters will be equipped to make informed decisions, thereby fostering a more engaged and informed electorate. Additionally, the project aims to address the challenge of low voter turnout by implementing a secure solution that ensures a 100% voting rate in the country. This will be achieved by facilitating easy access to the electoral process through an e-voting system integrated with Face Detection authentication. By combining these efforts, the project aims to not only enhance the democratic process but also strengthen the integrity and inclusivity of state elections in the country.

Scope

The scope of the project encompasses the development of a user-friendly web application using ReactJS, providing voters access to parties' policy positions tailored to their constituencies. This involves analyzing and presenting parties' policy positions based on manifestos and past actions, alongside integrating a survey mechanism for citizens to express their views on the upcoming state election. Additionally, the project extends to integrating an e-voting system with secure authentication mechanisms like Face Detection for electronic voting. Emphasis is placed on implementing robust user authentication and data security measures to safeguard voter information. The project includes comprehensive testing to validate functionality, usability, and security, ensuring compliance with quality standards. Scalability and accessibility are prioritized to accommodate a wide user base across various devices and platforms. Documentation and training materials will be provided to stakeholders for adoption and utilization. Ongoing support and maintenance services will be offered post-implementation to address issues and ensure long-term sustainability.

CHAPTER 2

LITERATURE SURVEY

Literature survey is the most important step in any kind of research. Before we start developing, we need to study the previous papers of our domain which we are working on and on the basis of study we can predict or generate the drawback and start working with the reference of previous papers.

OVERVIEW OF EXISTING RESEARCH AND LITERATURE.

TITLE: A Framework to make Voting system Transparent using Blockchain Technology

Author: Muhammad Shoaib Farooq, Usman Iftikhar, Andadel Khelifi.

Year: 2022

Description:

A widespread mistrust towards the traditional voting system has made democratic voting in any country very critical. People have seen their fundamental rights being violated. Other digital voting systems have been challenged due to a lack of transparency. Most voting systems are not transparent enough; this makes it very difficult for the government to gain voters' trust. The reason behind the failure of the traditional and current digital voting system is that it can be easily exploited. The primary objective is to resolve problems of the traditional and digital voting system, which include any kind of mishap or injustice during the process of voting. Blockchain technology can be used in the voting system to have a fair election and reduce injustice. The physical voting systems have many flaws in it as well as the digital voting systems are not perfect enough to be implemented on large scale. This appraises the need for a solution to secure the democratic rights of the people. This article presents a platform based on modern technology blockchain that provides maximum transparency and reliability of the system to build a trustful relationship between voters and election authorities. The proposed platform

provides a framework that can be implemented to conduct voting activity digitally through the blockchain without involving any physical polling stations. Our proposed framework supports a scalable blockchain, by using flexible consensus algorithms. The Chain Security Algorithm applied in the voting system makes the voting transaction more secure. Smart contracts provide a secure connection between the user and the network while executing a transaction in the chain. The security of the blockchain based voting system has also been discussed. Additionally, encryption of transactions using cryptographic hash and prevention of attack 51% on the blockchain has also been elaborated. Furthermore, the methodology for carrying out blockchain transactions during the process of voting has been elaborated using Blockchain

Finally, the performance evaluation of the proposed system shows that the system can be implemented in a large-scale population.

TITLE: A Smart contract system for Decentralized Borda Count Voting Author :

Somnath Panja, Samiran Bag, Feng Hao and Bimal Roy.

Year: 2020

Description:

In this article, we propose the first self-tallying decentralized e-voting protocol for a ranked-choice voting system based on Borda count. Our protocol does not need any trusted setup or tallying authority to compute the tally. The voters interact through a publicly accessible bulletin board for executing the protocol in a way that is publicly verifiable. Our main protocol consists of two rounds. In the first round, the voters publish their public keys, and in the second round they publish their randomized ballots. All voters provide Non-interactive Zero-Knowledge (NIZK) proofs to show that they have been following the protocol specification honestly without revealing their secret votes. At the end of the election, anyone, including a third-party observer will be able to compute the tally without needing any tallying authority. We provide security proofs to show

that our protocol guarantees the maximum privacy for each voter. We have implemented our protocol using Ethereum's blockchain as a public bulletin board to record voting operations as publicly verifiable transactions. The experimental data obtained from our tests show the protocol's potential for the real-world deployment.

**TITLE : E-Voting system evaluation based on the Council of Europe
recommendations : Helios Voting.**

Author: Luis Panizo, Mila Gascó, David Y. Marcos del Blanco.

Year: 2018

Description:

Despite the claimed benefits of e-voting initiatives, wider adoption of e-voting mechanisms and implementation processes is slower than expected. Several technical, social, and cultural challenges hinder generability and applicability of e-voting. Amongst them, the evaluation and harmonization of e-voting systems, given different legal and statutory frameworks, is still an important challenge to overcome. Yet, only a few works have addressed this topic in the field.

This article aims to contribute to further understanding this unexplored topic by applying a practical evaluation framework to Helios Voting, one of the most widely used e-voting tools to date. Our framework, strongly based on the technical and security requirements issued by the Council of Europe in 2017, is a valuable source of information for election officials, researchers and voters to understand the strengths and weaknesses of Helios Voting and, as a result, to improve decision-making processes regarding the type and size of elections that can be securely handled by Helios Voting. The ultimate goal of our paper is to conceptually and practically support the gradual, secure and protocolized expansion of e-voting.

TITLE: Blockchain Electronic Votes system.

Author: Mario Navarrete, Rudy Huancas, Paúl Díaz, and Mauro Rivadeneira

Year: 2019

Description:

Blockchain technology has been presented as a support for trust needs between transactions in electronic information systems. Its successful use in cryptocurrencies has allowed it to explore its capabilities in commercial, industrial, and service systems, backed by the operational alternatives offered by Ethereum Smart Contracts and the cryptographic security of public and private key. These keys are used as a way to make online transactions anonymously, with the guarantee offered by the Blockchain network that they are executed safely. With the above in mind, this concept can be extended to the electoral processes, thus allowing its application in electronic voting systems, especially when the protocols currently used lack the trust factor between the different social actors. This document presents a proof of concept in which Blockchain and other technologies are applied, to allow interaction as an electronic voting system for the election of unique candidates.

This has been achieved through the specification of an architecture designed especially for electoral processes, from which it is implemented and a simulation is carried out in order to obtain data that generates value, when evaluating Blockchain technology as an alternative to current voting systems.

TITLE:SecurePhysicalLayerVoting

Author:Nirnimesh Ghose,Bocan Hu,Yan Zhang,and Loukas Lazos

Year:2018

Description:

Distributed wireless networks often employ voting to perform critical network functions such as fault-tolerant data fusion, cooperative sensing, and reaching consensus. Voting is implemented by sending messages to a fusion center or via direct message exchange between participants. However, the delay overhead of message-based voting can be prohibitive when numerous participants have to share the wireless channel in sequence, making it impractical for time-critical applications. In this paper, we propose a fast PHY-

layer voting scheme called PHYVOS, which significantly reduces the delay for collecting and tallying votes. In PHYVOS, wireless devices transmit their votes simultaneously by exploiting the subcarrier orthogonality of OFDM and without explicit messaging. Votes are realized by injecting energy to pre-assigned subcarriers. We show that PHYVOS is secure against adversaries that attempt to manipulate the voting outcome.

TITLE: Distributed Voting/Ranking with Optimal Number of States per Node.

Author/Year: Saber Salehkaleybar/2015.

Description:

Considering a network with n nodes, where each node initially votes for one (or more) choices out of K possible choices, we present a Distributed Multi-choice Voting/Ranking (DMVR) algorithm to determine either the choice with maximum vote (the voting problem) or to rank all the choices in terms of their acquired votes (the ranking problem). The algorithm consolidates node votes across the network by updating the states of interacting nodes using two key operations; the union and the intersection. The proposed algorithm is simple, independent from network size, and easily scalable in terms of the number of choices K , using only $K \times 2^{K-1}$ nodal states for voting, and $K \times K!$ nodal states for ranking. We prove the number of states to be optimal in the ranking case; this optimality is conjectured to also apply to the voting case. The time complexity of the algorithm is analyzed in complete graphs. We show that the time complexity for both ranking and voting is $O(\log(n))$ for given vote percentages, and is inversely proportional to the minimum of the vote percentage differences among various choices.

INFERENCES FROM LITERATURE SURVEY

After going through the previous works we inferred some points. The literature survey reveals a concerted effort towards leveraging advanced technologies to address the shortcomings of traditional voting systems, particularly in terms of transparency, security, and scalability. Blockchain technology emerges as a prominent solution, offering immutable ledgers that

enhance transparency and mitigate risks associated with manipulation. Decentralized systems, coupled with cryptographic techniques such as zero-knowledge proofs and encryption, ensure greater privacy and integrity in the voting process. Smart contracts automate key functions, reducing reliance on intermediaries and enhancing efficiency. Moreover, evaluation frameworks are developed to assess the effectiveness and compliance of e-voting systems, aiding decision-making processes. Efforts are also directed towards designing efficient protocols to optimize voting processes, particularly in time-critical scenarios. Overall, the literature underscores a shift towards technology-driven solutions aimed at fostering trust, security, and reliability in democratic processes.

OPEN PROBLEMS IN EXISTING SYSTEM

AI and blockchain in an e-voting system presents a promising solution to several persistent problems in traditional voting systems. However, despite the advancements, challenges persist within this innovative framework:

Security Concerns: While blockchain offers robust security through its decentralized and immutable ledger, e-voting systems are still vulnerable to cyber threats. AI may assist in detecting and preventing potential attacks, but ensuring the overall cybersecurity of the system remains a primary concern.

Voter Verification: Verifying the identity of voters in an e-voting system poses a challenge. AI-powered identity verification can enhance accuracy, but ensuring that only eligible voters participate without compromising anonymity remains a complex issue.

Accessibility and Inclusivity: E-voting systems must cater to all demographics, including those with disabilities or limited access to technology. Ensuring usability for everyone without marginalizing certain groups is crucial.

Misinformation and Manipulation: Despite blockchain's data integrity, misinformation about candidates or political parties circulating online could still

influence voters. AI may assist in detecting false information, but eradicating misinformation entirely remains challenging.

Public Trust and Acceptance: Convincing the public about the reliability and security of an AI-based e-voting system with blockchain is crucial. Building trust in new technology, especially when it involves sensitive processes like elections, takes time and concerted effort.

Regulatory and Legal Frameworks: Developing appropriate legal frameworks and regulations to govern AI-based e-voting systems, ensuring compliance with privacy laws, and addressing potential biases in AI algorithms are ongoing challenges.

CHAPTER 3

REQUIREMENT ANALYSIS

FEASIBILITY STUDIES OF THE PROJECT

The proposed project, "AI-based Secured Voting System with Blockchain Mined Data Storage," demands a flexible study plan encompassing various phases. Initial research involves comprehending existing voting systems, AI algorithms, and blockchain technology, outlining their strengths, weaknesses, and integration possibilities. Design and development entail iterative stages, allowing for adaptability to address emerging technological challenges and security concerns. Testing and validation phases must encompass diverse scenarios to ensure system robustness and security. Continuous learning and flexibility are key, facilitating adjustments based on evolving legal regulations, technological advancements, and user feedback. A dynamic study plan accommodates these phases, fostering a responsive and adaptable approach crucial for the success of such an innovative and complex project. In this case, three key considerations involved in the feasibility analysis are

- Economical Feasibility
- Technical Feasibility
- Operational Feasibility

Economical Feasibility

Economic feasibility assesses the financial viability of implementing such a system, considering initial setup costs, maintenance expenses, and potential cost savings or revenue generation. It involves analyzing the return on investment (ROI) over the system's lifecycle, factoring in expenses for AI development, blockchain infrastructure, security measures, and ongoing maintenance.

Technical Feasibility

Technical feasibility evaluates the technological aspects, examining the capability of AI algorithms to accurately process votes, the scalability of the blockchain network for storing vast amounts of voting data securely, and the interoperability of these systems with existing election frameworks. Compatibility with diverse devices, network resilience, and the system's ability to withstand cyber threats are also key technical considerations.

Operational Feasibility

Operational feasibility revolves around the practical implementation and utilization of the system within the context of electoral processes. It involves assessing the readiness of election officials, voters, and support staff for embracing and operating the new system. Training needs, potential disruptions during the transition, and the impact on election timelines and procedures are crucial factors to evaluate. Additionally, ensuring the system's accessibility to a diverse electorate is essential for operational success.

REQUIREMENT SPECIFICATION

Hardware Requirements

- System : i5 processor
- Hard Disk : 1TB
- Monitor : 15" LED
- Input Devices : 8GB.

Software Requirements

- Operating System : Windows.
- Coding Language : Python.
- ReactJS, NodeJS IDE : Visual Studio (code editor).

CHAPTER 4

DESCRIPTION OF PROPOSED SYSTEM

After analyzing the drawbacks of previous works, we proposed a new system that develop a web application for candidate registration and public participation in elections marks a significant step towards enhancing accessibility and efficiency in the electoral process. Leveraging advanced face detection and recognition algorithms ensure the authentication of users during voting, enhancing the security and integrity of the system. The integration of Haar cascade face detection and Support Vector Machine (SVM) algorithms enables accurate identification of individuals, ensuring that only authentic users cast their votes. Additionally, the creation of an API facilitates seamless integration between the web application and the backend database, enhancing data management and accessibility. NodeJS is utilized for API development, providing a robust and scalable platform, while MongoDB serves as the database setup, offering flexibility and efficiency in data storage and retrieval. Furthermore, the integration of blockchain technology at the backend enhances data security, providing an immutable and transparent ledger that safeguards against hacking and tampering.

Advantages of the proposed system

- Enhances citizen awareness of the candidates in their locality.
- Ensures 100% voting rate in the country.
- Provides a secure technological solution for citizens to cast their votes.
- Secures the data stored in the database using blockchain.
- Immediate results.

DESCRIPTION OF RESEARCH APPROACH AND METHODS

- Face Detection
- Face Recognition
- Node API Generation
- MongoDB Generation

- WebApplicationDevelopment
- BlockchainIntegration

FaceDetection

ForFaceDetectionmodule,hereweareusingHaarcascadealgorithm.

It is an Object Detection Algorithm used to identify faces in an image or a real time video. The algorithm uses edge or line detection features proposed by Viola and Jones in their research paper “Rapid Object Detection using a Boosted Cascade of Simple Features” published in 2001. The algorithm is given a lot of positive images consisting of faces, and a lot of negative images not consisting of any face to train on them. The repository has the models stored in XML files, and can be read with the OpenCV methods. These include modelsfor face detection, eye detection, upperbody and lower body detection, license plate detection etc.

The first contribution to the research was the introduction of the haar features. These features on the image make it easy to find out the edges or the lines in the image, or to pick areas where there is a sudden change inthe intensitiesofthepixels. Thedarkerareasin thehaarfeature arepixels with values 1,and the lighter areas are pixels with values 0. Each of these is responsible for finding out one particular feature in the image. Such as an edge, a line or any structurein theimage where there is a suddenchange ofintensities. For ex. in theimage above, the haar feature can detecta vertical edge with darker pixels at its right and lighter pixels at its left.

The objective is to find out the sum of all the image pixels lying in the darker area of the haar feature and the sum of all the image pixels lying in the lighter area of the haar feature. And then find out their difference. Now if the image has an edge separating dark pixels on the right and light pixels on the left,then the haar value will be closer to 1. This is just one representation of a particular haar feature separating a vertical edge. Now there are other haar features as well, which will detect edges in other directions and any other image structures. To detect an edge anywhere in the image, the haar feature needs to traverse the whole image.

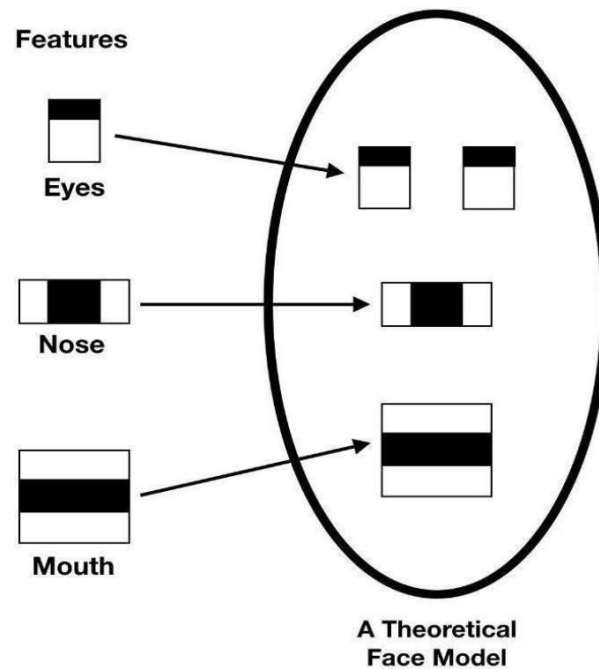


Fig4.1:Haarcascade

FaceRecognition

ForFaceRecognitionmodule,hereweareusingSVMalgorithm.

“Support Vector Machine” (SVM) is a supervised machine learning algorithm that can be used for both classification and regression challenges. However, it is mostly used in classification problems. In the SVM algorithm, we plot each data item as a point in n- dimensional space (where n is a number of features we have) with the value of each feature being the value of a particular coordinate. Then, we perform classification by finding the hyper- plane that differentiates the two classes very well. SVM's are used in applications like handwriting recognition, intrusion detection, face detection, email classification, gene classification, and in web pages. The goal of the SVM algorithm is to create the best line or decision boundary that can segregate n-dimensional space into classes so that we can easily put the new data point in the correct category in the future. This best decision boundary is called a hyperplane.SVM chooses the extreme points/vectors that help in creating the hyperplane. These extreme cases are called as support vectors, and hence algorithm is termed as Support Vector Machine.

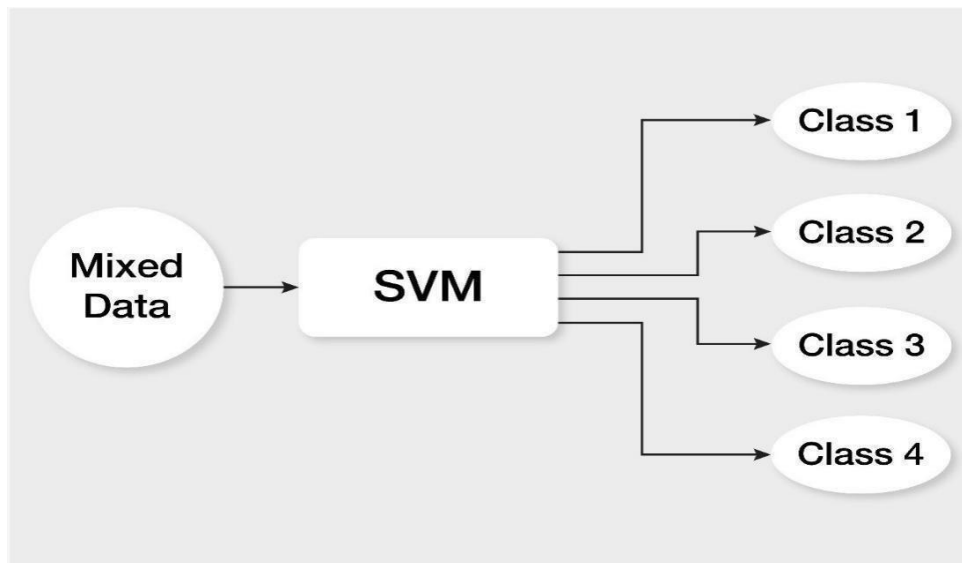


Fig4.2:SVMalgorithm

NodeAPIGeneration

In this project we develop the API (ApplicationProgramming Interface) using the javascript framework nodeJS. Node.js is an open-source, cross- platform, back-end JavaScript runtime environment that runs on the V8 engine and executes JavaScript code outside a web browser. Node.js lets developersuse JavaScript to write command line tools and for server-side scripting running scripts server-side to produce dynamic web page content before the page is sent to the user's web browser. Consequently, Node.js represents a "JavaScript everywhere" paradigm, unifying web-application development around a single programming language, rather than different languages for server-side and client-side scripts.

Though .js is the standard filename extension for JavaScriptcode, the name "Node.js" does not refer to a particular file in this contextand is merely the name of the product. Node.js has an event-driven architecture capable of asynchronous I/O. These design choices aim to optimize throughput and scalability in web applications with many input/output operations, as well as for real-time Web applications (e.g., real-time communication programs and browser games.

The Node.js distributed development project was previously governed by the Node.js Foundation, and has now merged with the JS Foundation to form the OpenJS Foundation, which is facilitated by the Linux Foundation's Collaborative Projects program. Corporate users of Node.js software include GoDaddy, Groupon, IBM, LinkedIn, Microsoft, Netflix, PayPal, Rakuten, SAP, Walmart, Yahoo!, and Amazon Web Services.

Node JS is an open-source JavaScript tool built on Google Chrome's JavaScript Engine. It's used to build scalable network applications using an event-driven, non-blocking I/O model, which makes it fast and light on resources. It allows developers to use a single language (JavaScript) on both the server-side and the Client-Side. Node.js also provides a rich library of various JavaScript modules which simplifies the development of web applications using Node.js to a great extent.

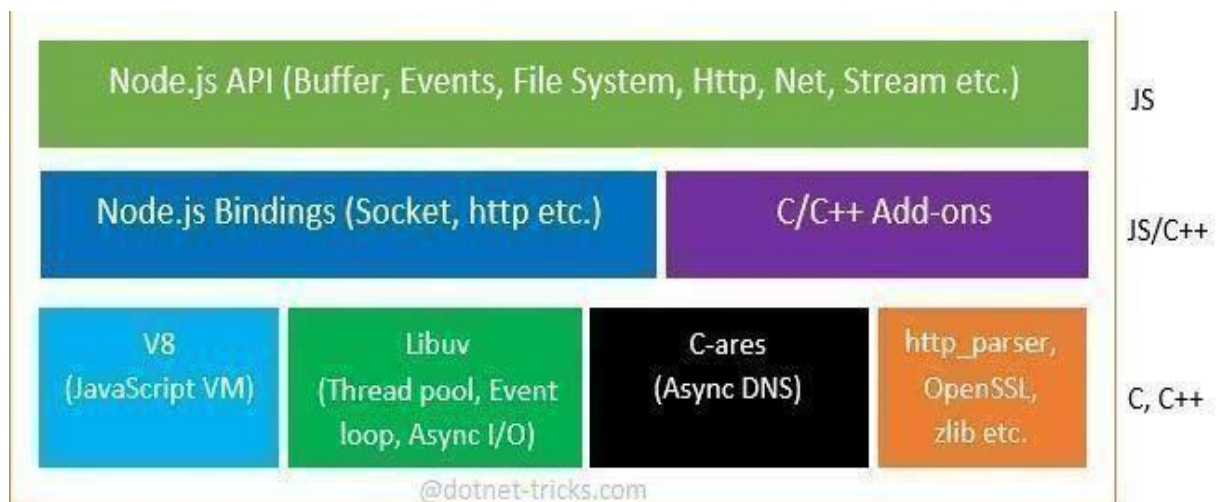


Fig4.3:NodeAPIintegration

MongodbGeneration

MongoDB is a source-available cross-platform document-oriented database

Program Classified as a NoSQL database program, MongoDB uses JSON-like documents with optional schemas. MongoDB is developed by MongoDB Inc. and licensed under the Server Side Public License (SSPL). Database is a physical container for collections. Each database gets its own set of files on the file system. A single MongoDB server typically has multiple databases. Collection is a group of MongoDB documents. It is the equivalent of an RDBMS table. A document is a set of key-value pairs. Documents have dynamic schema. Dynamic schema means that documents in the same collection do not need to have the same set of fields or structure, and common fields in a collection's documents may hold different types of data.

MongoDB supports field, range query, and regular-expression searches. Queries can return specific fields of documents and also include user-defined JavaScript functions. Queries can also be configured to return a random sample of results of a given size. Fields in a MongoDB document can be indexed with primary and secondary indices or index. MongoDB provides high availability with replica sets. A replica set consists of two or more copies of the data. Each replica-set member may act in the role of primary or secondary replica at any time. All writes and reads are done on the primary replica by default. Secondary replicas maintain a copy of the data of the primary using built-in replication. When a primary replica fails, the replica set automatically conducts an election process to determine which secondary should become the primary. Secondaries can optionally serve read operations, but that data is only eventually consistent by default.

MongoDB scales horizontally using sharding. The user chooses a shard key, which determines how the data in a collection will be distributed. The data is split into ranges (based on the shard key) and distributed across multiple shards. (A shard is a master with one or more replicas.) Alternatively, the shard key can be hashed to map to a shard – enabling an even data distribution. MongoDB can run over multiple servers, balancing the load or duplicating data to keep the system up and running in case of hardware failure.

optional schemas. MongoDB is developed by MongoDB Inc. and licensed under the Server Side Public License (SSPL). Database is a physical container for collections. Each database gets its own set of files on the file system. A single MongoDB server typically has multiple databases. Collection is a group of MongoDB documents. It is the equivalent of an RDBMS table. A document is a set of key-value pairs. Documents have dynamic schema. Dynamic schema means that documents in the same collection do not need to have the same set of fields or structure, and common fields in a collection's documents may hold different types of data.

MongoDB supports field, range query, and regular-expression searches. Queries can return specific fields of documents and also include user-defined JavaScript functions. Queries can also be configured to return a random sample of results of a given size. Fields in a MongoDB document can be indexed with primary and secondary indices or index. MongoDB provides high availability with replica sets. A replica set consists of two or more copies of the data. Each replica-set member may act in the role of primary or secondary replica at any time. All writes and reads are done on the primary replica by default. Secondary replicas maintain a copy of the data of the primary using built-in replication. When a primary replica fails, the replica set automatically conducts an election process to determine which secondary should become the primary. Secondaries can optionally serve read operations, but that data is only eventually consistent by default.

MongoDB scales horizontally using sharding. The user chooses a shard key, which determines how the data in a collection will be distributed. The data is split into ranges (based on the shard key) and distributed across multiple shards. (A shard is a master with one or more replicas.) Alternatively, the shard key can be hashed to map to a shard – enabling an even data distribution. MongoDB can run over multiple servers, balancing the load or duplicating data to keep the system up and running in case of hardware failure.

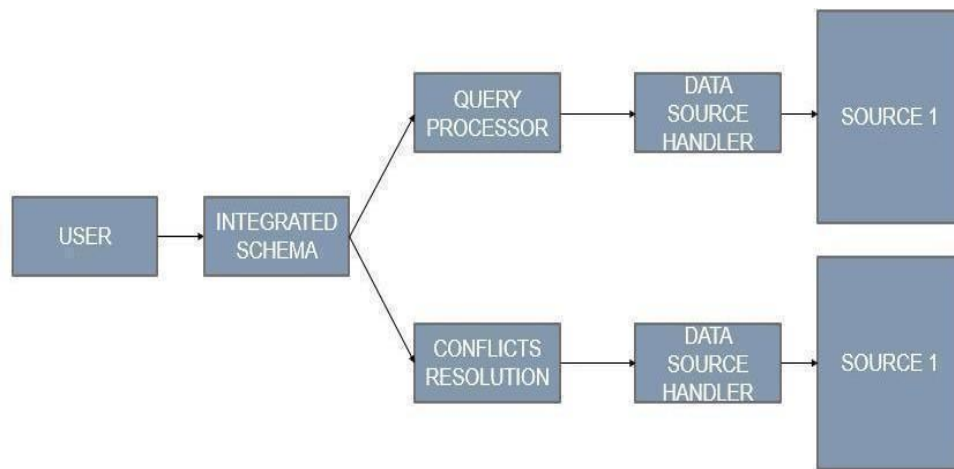


Fig4.4:MongoDBintegration

WebApplicationDevelopment

Web application development is the creation of application programs that reside on remote servers and are delivered to the user's device over the Internet. A web application (web app) does not need to be downloaded and is instead accessed through a network. An end user can access a web application through a web browser such as Google Chrome, Safari, or Mozilla Firefox. A majority of web applications can be written in JavaScript, Cascading Style Sheets (CSS), and HTML5.

Web application development will typically have a short development life-cycle lead by a small development team. Front-end development for web applications is accomplished through client-side programming. Client refers to a computer application such as a web browser. Client-side programming will typically utilize HTML, CSS and JavaScript. HTML programming will instruct a browser how to display the on-screen content of web pages, while CSS keeps displayed information in the correct format. JavaScript will run JavaScript code on a web page, making some of the content interactive.

Server-side programming powers the client-side programming and is used to create the scripts that web applications use. Scripts can be written in multiple scripting languages such as Ruby, Java and Python. Server-side scripting will create a custom interface for the end-user and will hide the source code

that makes up the interface. A database such as MySQL or MongoDB can be used to store data in web application development.

React is a JavaScript library for building user interfaces. React is used to build single- page applications. React allows us to create reusable UI components. Lots of people use React as the V in MVC. React abstracts away the DOM from us, offering a simpler programming model and better performance. React can also render on the server using Node, and it can power native apps using React Native. React implements one-way reactive data flow, which reduces the boilerplate and is easier to reason about than traditional data binding.

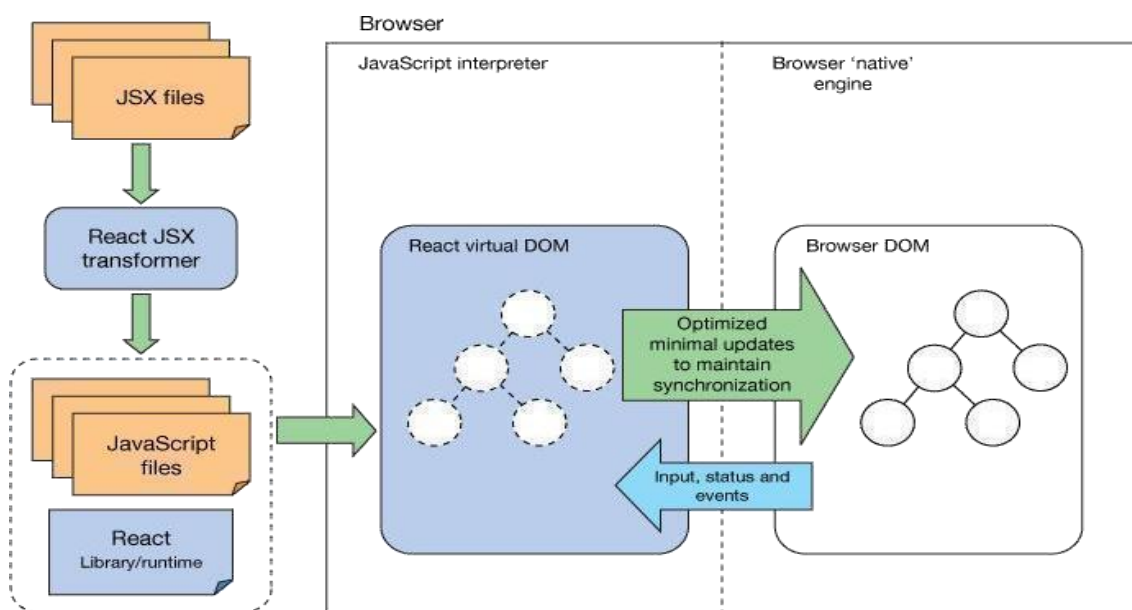


Fig4.5:Webapplicationdevelopment

Block-chainIntegration

Blockchain is integrated at the backend for data security which will secure the data with utmost security eliminating data hacking. Blockchain for integration is a powerful technology which creates an immutable record of transactions that can be used to optimize business processes, enhance security, and maintain trust between stakeholders. It reduces costs by removing middlemen and ensuring that transactions are quickly and

accurately fulfilled.

A blockchain is essentially a digital ledger of transactions that is duplicated and distributed across the entire network of computer systems on the blockchain. Each block in the chain contains a number of transactions, and every time a new transaction occurs on the blockchain, a record of that transaction is added to every participant's ledger. The decentralised database managed by multiple participants is known as Distributed Ledger Technology (DLT). Blockchain is a type of DLT in which transactions are recorded with an immutable cryptographic signature called a hash.

This means if one block in one chain was changed, it would be immediately apparent it had been tampered with. If hackers wanted to corrupt a blockchain system, they would have to change every block in the chain, across all of the distributed versions of the chain. Blockchains such as Bitcoin and Ethereum are constantly and continually growing as blocks are being added to the chain, which significantly adds to the security of the ledger.

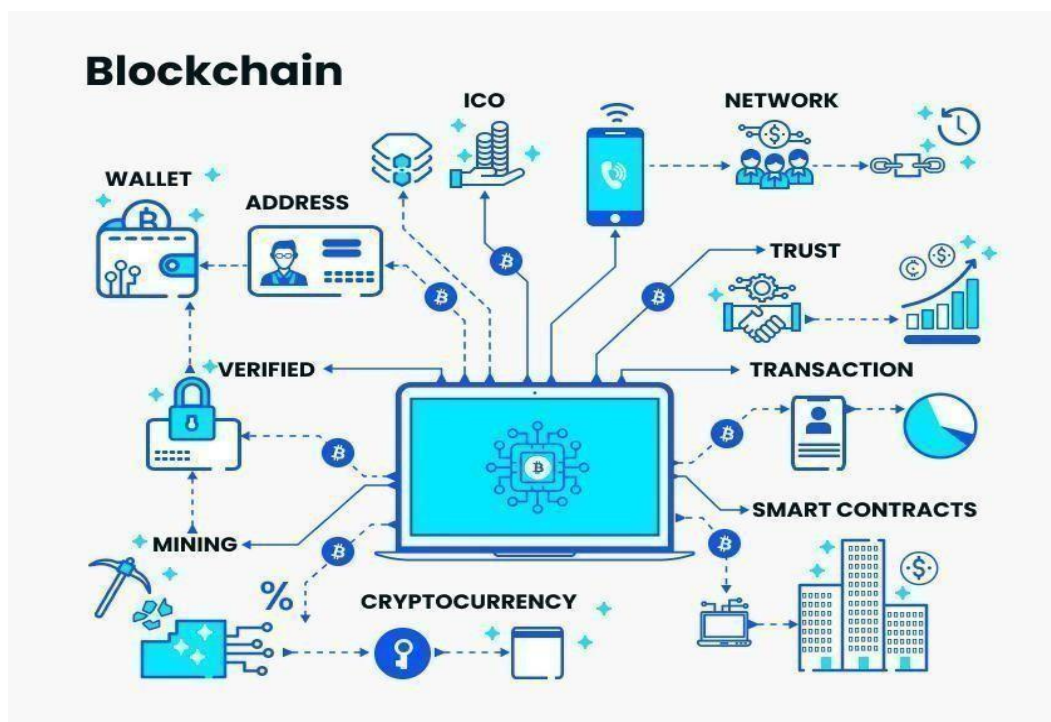


Fig4.6:Blockchainintegration

ARCHITECTURE OR PROPOSED SYSTEM

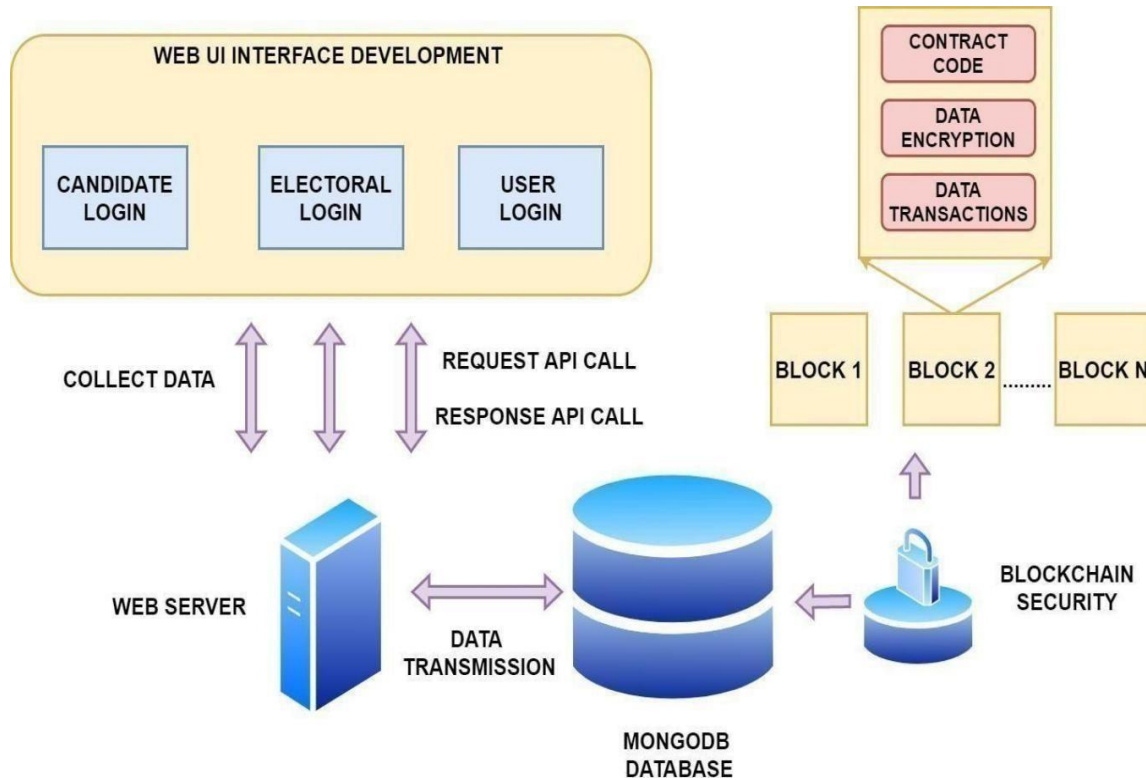


Fig4.7:ProposedSystemArchitecture

In this project, we primarily focus to provide the people an authentic scientific community with parties' policy positions, with respect to their constituencies. A web application is developed using reactJS that aims to analyse the role of the existing political parties/independents who are contesting in the forthcoming state election. Face detection and recognition algorithms are used to effectively identify authentic users when voting during Election Day using the application. For face detection, Haar cascade is used and for face recognition SVM algorithm is used. Then, we developed the API (Application Programming Interface) using the javascript framework nodeJS to effectively integrate the web application with the database. NodeJS is used for API development and mongoDB is used for database setup. Blockchain integration at the backend for data security which will secure the data with utmost security eliminating data hacking.

The project also helps the common man to take a survey with the questionnaire to express his/her willingness and views on the current/forthcoming state election. This web application also helps the state government in achieving 100% voting rate in the state elections by providing an e- voting system enabling Face Detection authentication.

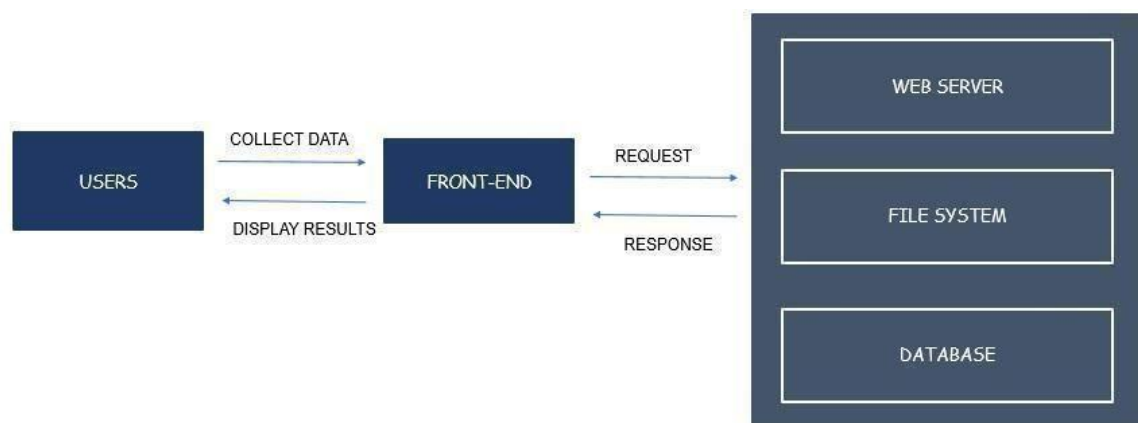


Fig4.8:WebArchitecture

In this project, we primarily focus on providing the people an authentic scientific community with parties' policy positions, with respect to their constituencies. A web application is developed using reactJS that aims to analyse the role of the existing political parties/independents who are contesting in the forthcoming state election. Face detection and recognition algorithms are used to effectively identify authentic users when voting during Election Day using the application. For face detection, Haar cascade is used and for face recognition, SVM algorithm is used. Then, we developed the API (Application Programming Interface) using the javascript framework nodeJS to effectively integrate the web application with the database. NodeJS is used for API development and mongoDB is used for database setup. Blockchain integration at the backend for data security which will secure the data with utmost security eliminating data hacking. The project also helps the ordinary man to take a survey with the

questionnaire to express his/her willingness and views on the current/forthcoming state election. This web application also helps the state government in achieving 100% voting rate in the state elections by providing an e-voting system enabling Face Detection authentication.

4.3 DESCRIPTION OF SOFTWARE FOR IMPLEMENTATION AND TESTING PLAN OF THE PROPOSED MODEL/SYSTEM

The purpose of the Software Requirement Specification is to produce the specification of the analysis task and also to establish complete information about the requirement, behavior and also the other constraint like functional performance and so on. The main aim of the Software Requirement Specification is to completely specify the technical requirements for the software product in a concise and in unambiguous manner.

1. VISUAL STUDIO

In this project the Microsoft Visual Studio is used as an IDE.

Visual Studio Code combines the simplicity of a source code editor with powerful developer tooling, like IntelliSense code completion and debugging. First and foremost, it is an editor that gets out of our way. The delightfully frictionless edit-build-debug cycle means less time fiddling with our environment, and more time executing on our ideas. Visual Studio Code supports macOS, Linux, and Windows - so we can hit the ground running, no matter the platform.

At its heart, Visual Studio Code features a lightning fast source code editor, perfect for day-to-day use. With support for hundreds of languages, VS Code helps us be instantly productive with syntax highlighting, bracket-matching, auto-indentation, box-selection, snippets, and more. Intuitive keyboard shortcuts, easy customization and community-contributed keyboard shortcut mappings let us navigate our code with ease.

For serious coding, we'll often benefit from tools with more code understanding than just blocks of text. Visual Studio Code includes built-in support for IntelliSense code completion, rich semantic code understanding and navigation, and code refactoring. And when the coding gets tough, the tough get debugging. Debugging is often the one feature that developers miss most in a leaner coding experience, so we made it happen. Visual Studio Code includes an interactive debugger, so we can step through source code, inspect variables, view call stacks, and execute commands in the console. VS Code also integrates with build and scripting tools to perform common tasks making everyday workflows faster. VS Code has support for Git so we can work with source control without leaving the editor including viewing pending changes diffs. Visual Studio Code includes a public extensibility model that lets developers build and use extensions, and richly customize their edit-build- debug experience.

2. PYTHON

In this project python is used as a programming language for development. In technical terms, Python is an object-oriented, high-level programming language with integrated dynamic semantics primarily for web and app development. It is extremely attractive in the field of Rapid Application Development because it offers dynamic typing and dynamic binding options. Python is relatively simple, so it's easy to learn since it requires a unique syntax that focuses on readability. Developers can read and translate Python code much easier than other languages. In turn, this reduces the cost of program maintenance and development because it allows teams to work collaboratively without significant language and experience barriers.

Additionally, Python supports the use of modules and packages, which means that programs can be designed in a modular style and code can be reused across a variety of projects. Once a module or package is developed by an user, it can be scaled for use in other projects, and it's easy to import or export these modules. One of the most promising benefits of Python is that both the standard library and the interpreter are available free of charge, in both binary and source form. There is no exclusivity either, as Python and

all the necessary tools are available on all major platforms. Therefore, it is an enticing option for developers who don't want to worry about paying high development costs.

3. REACTJS

ReactJS basically is an open-source JavaScript library which is used for building user interfaces specifically for single page applications. It's used for handling view layer for web and mobile apps. React also allows us to create reusable UI components. React was first created by Jordan Walke, a software engineer working for Facebook. React first deployed on Facebook's newsfeed in 2011 and on Instagram.com in 2012.

React allows developers to create large web applications which can change data, without reloading the page. The main purpose of React is to be fast, scalable, and simple. It works only on user interfaces in application. This corresponds to view in the MVC template. It can be used with a combination of other JavaScript libraries or frameworks, such as AngularJS.

CHAPTER 5

IMPLEMENTATIONDETAILS

IMPLEMENTATION

In this project, it focuses on developing a web application using ReactJS to provide users with insights into political parties' policy positions in their respective constituencies. Leveraging face detection and recognition algorithms such as Haar cascade and SVM, the application ensures secure and authentic user authentication during the e-voting process. The backend API, built with Node.js, integrates seamlessly with MongoDB for efficient data management. Blockchain technology enhances data security, guaranteeing the integrity and immutability of election-related information. Users can express their views on the upcoming elections through a survey questionnaire feature. By enabling e-voting with face detection authentication, the application aims to increase voter participation and achieve a 100% voting rate in state elections. Through rigorous testing, deployment on reliable hosting platforms, and user training, the project ensures a secure, user-friendly, and inclusive platform for political engagement and electoral participation.

ALGORITHMS

Haar Cascade for face detection, a robust object detection algorithm trained on positive and negative image datasets. For face recognition, Support Vector Machine (SVM) is employed, a supervised learning algorithm effective in classifying data points in multidimensional space. Together, these algorithms enable secure and efficient identification of users during the e-voting process.

HaarCascade

The Haar Cascade algorithm is a cornerstone in face detection, renowned for its versatility across various applications. Primarily utilized to identify faces within images or real-time video streams, it operates through an object detection approach. By training on extensive datasets containing both positive images with faces and negative images without faces, the algorithm learns to

differentiate between facial features and background elements. These models, stored as XML files and accessible through OpenCV methods, encompass not only face detection but also the detection of other objects like eyes, upper bodies, lower bodies, and license plates. Its robustness and efficiency make it a valuable tool in computer vision tasks, facilitating accurate and reliable detection within diverse environments and contexts.

Support Vector Machine

The Face Recognition module leverages the Support Vector Machine (SVM) algorithm, a versatile supervised machine learning technique suitable for both classification and regression tasks. SVM operates by mapping each data point into an n -dimensional space, where n represents the number of features, with each feature corresponding to a specific coordinate value. Subsequently, classification is achieved by identifying the hyperplane that effectively separates the two classes. SVM finds extensive applications across diverse domains such as handwriting recognition, intrusion detection, email classification, gene classification, and web page categorization. Its capability to discern patterns and classify data points accurately makes SVM a valuable asset in face recognition systems, contributing to their efficacy and reliability in real-world scenarios.

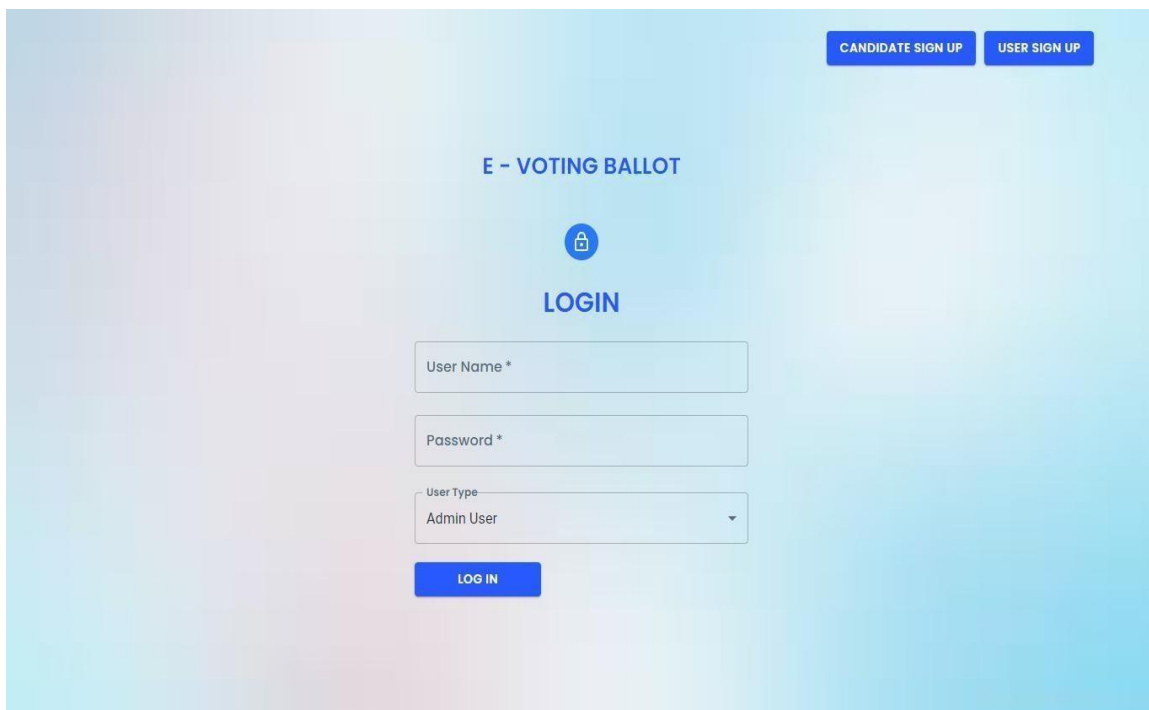
CHAPTER 6

RESULTS AND DISCUSSIONS

RESULTS

To begin with, testing of the trained model, we can split our project into modules of implementation that is done. A web application using a javascript framework reactJS is developed for both the candidate and also the general public to register themselves.

The login page can be shown in the below figure.



The image shows a web application interface for an "E - VOTING BALLOT" system. At the top right, there are two blue buttons: "CANDIDATE SIGN UP" and "USER SIGN UP". In the center, the text "E - VOTING BALLOT" is displayed above a blue padlock icon and the word "LOGIN". Below this, there is a login form with three input fields: "User Name *" (text), "Password *" (password), and "User Type" (a dropdown menu currently showing "Admin User"). At the bottom of the form is a blue "LOG IN" button.

Fig6.1:Loginpage

Candidate Registration Form

Name *	Password *
Age *	Gender Male ▼
FatherName *	MotherName *
Phone Number *	Aadhar Card Number *
Party Name ▼	Constituency ▼
Address *	

SAVE DETAILS
CANCEL

Fig6.2:Candidateregistrationform

Theuserregistrationformcanbeshowninthebelowfigure.

User Registration Form

Name *	Password *
Age *	Gender Male ▼
FatherName *	MotherName *
Phone Number *	Aadhar Card Number *
Address *	

SAVE DETAILS
CANCEL

Fig6.3:Userregistrationform

The admin home screen can be shown in the below figure.

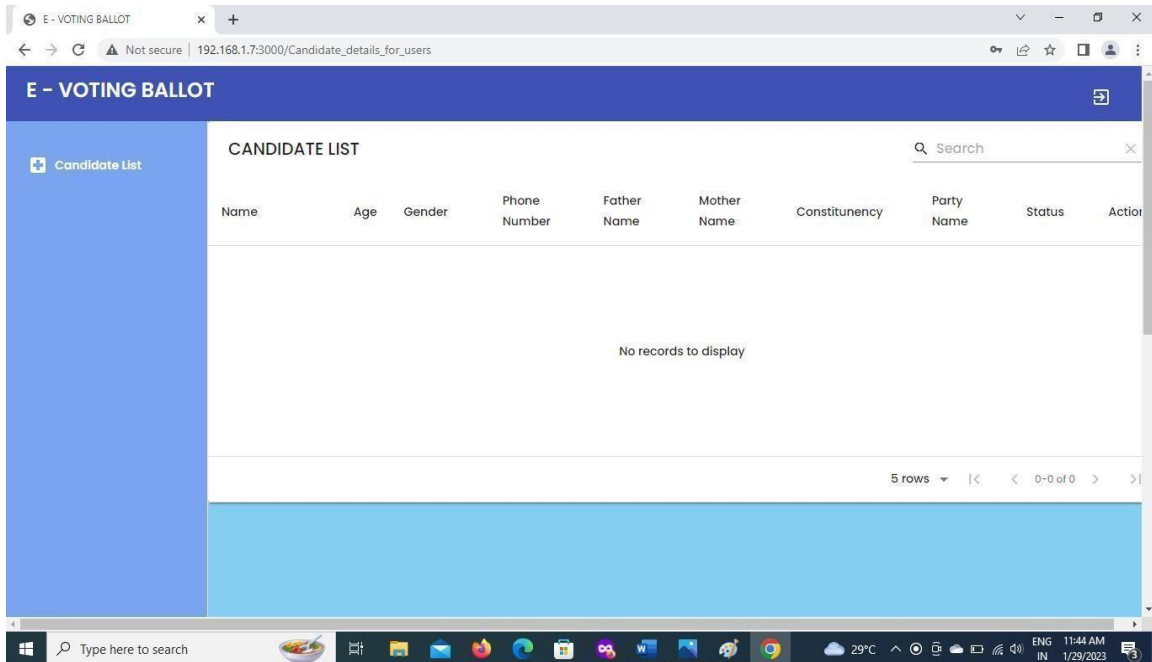


Fig6.4: Admin home screen

The user home screen with candidate list can be shown in the below figure.

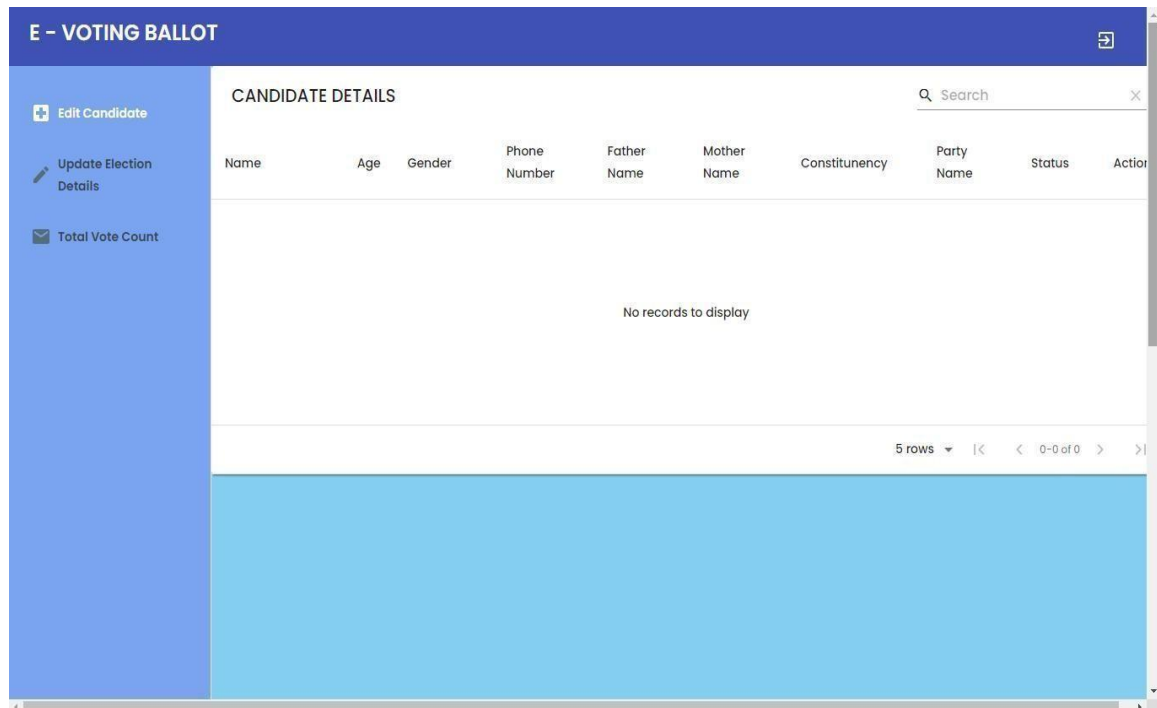


Fig6.5: User home screen with candidate list

The candidate home screen can be shown in the below figure.

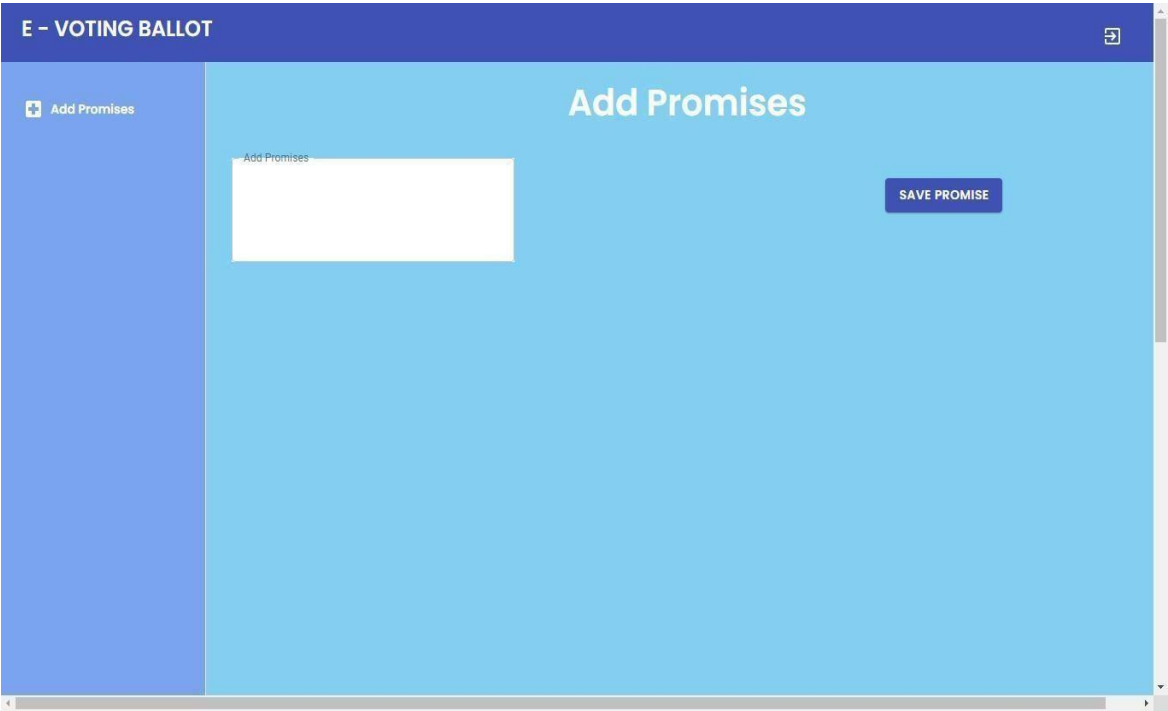


Fig6.6: Candidate home screen

Then, it will check whether the vote has been registered, if registered the vote has been placed and the below figure can be seen as follows:

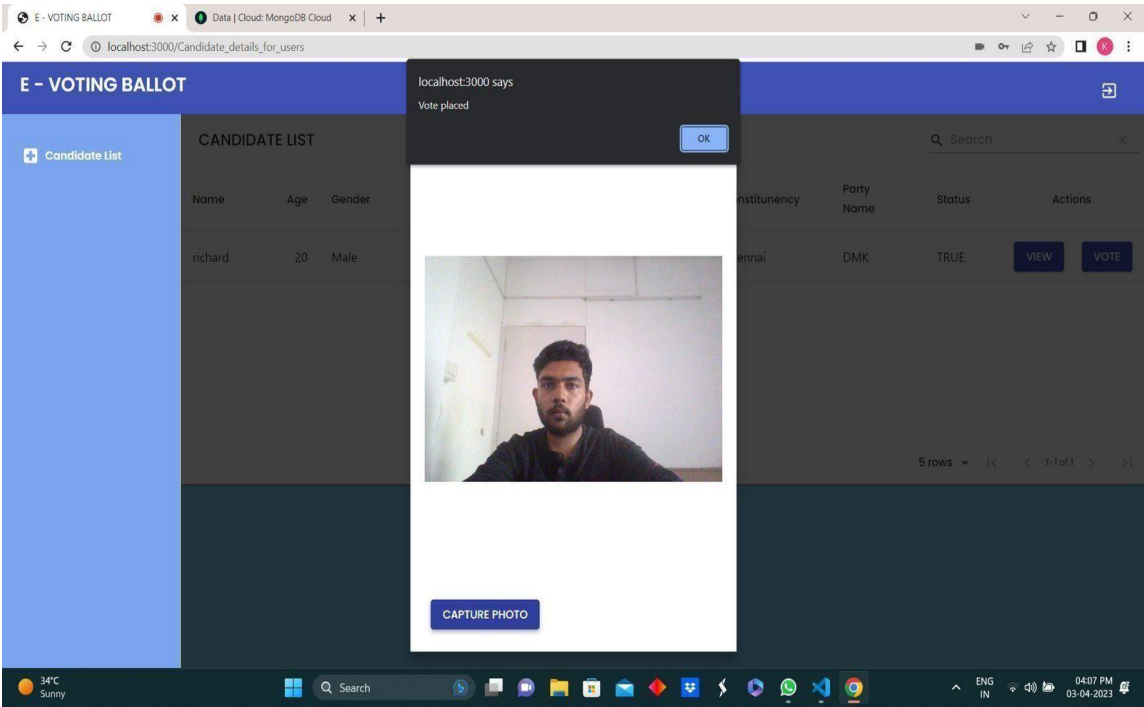


Fig6.7: Vote placed

The end results of the election can be viewed as seen in the image below:

If the backend data is secure, the blockchain status will be good and the below figure can be seen as follows:



Fig6.8 Election results

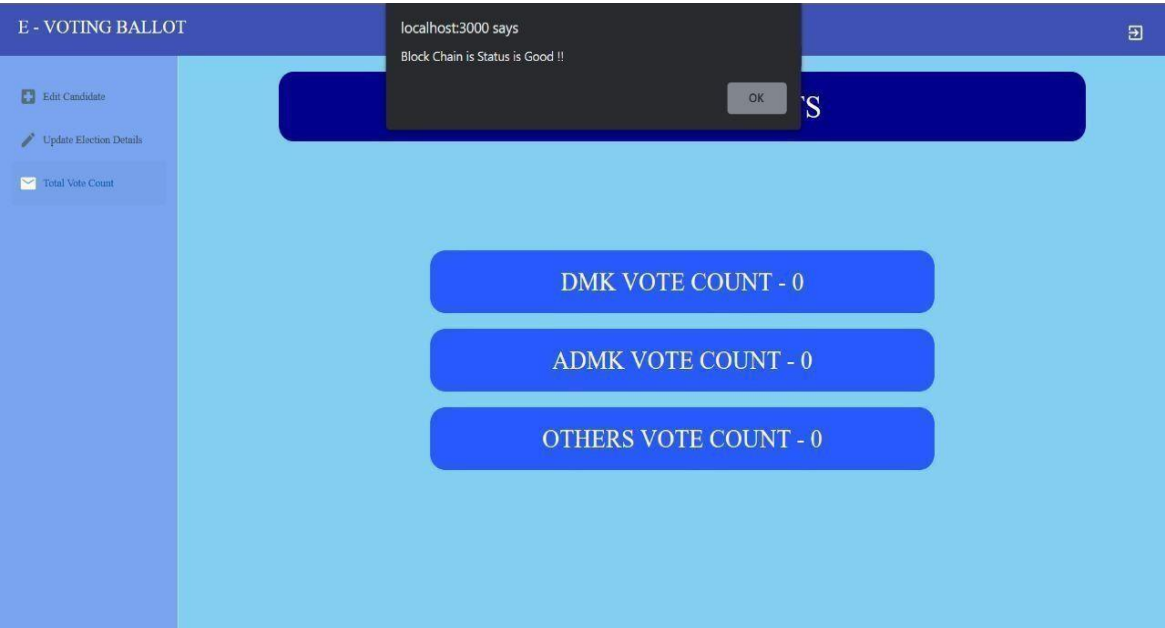


Fig6.9: Blockchain status is good

CHAPTER 7

CONCLUSION

CONCLUSION:

This project has been successfully developed a web application for safe online ballot system which helps the state government in achieving 100% voting rate in the state elections by providing an e-voting system enabling Face Detection authentication. A web application using a javascript framework reactJS is developed for both the candidate and also the general public to register themselves. By this project we effectively enable the general public to know about their candidates. Thus, this project provides an affordable and efficient means to enable the general public to securely cast their vote.

FUTUREWORK

In the coming future, we review the application of the project extensively by the ElectionCommissionofIndia. In the election commission of India, there are more chances to develop or convert this project in many ways. Thus, this project has an efficient scope in the coming future where votes can be cast securely online, increasing voter turnout and enhancing citizen awareness of the candidates in their locality.

REFERENCES

- [1] Christian Galea; Reuben A. Farrugia, "Matching Software-Generated Sketches to Face Photographs With a Very Deep CNN, Morphed Faces, and Transfer Learning", IEEE Transactions on Information Forensics and Security [Vol no: 13, 2018]
- [2] FáberD. Giraldo; BarbosaMilton C.; CarlosE. Gamboa,"ElectronicVoting Using Blockchain And Smart Contracts: Proof Of Concept", IEEE Latin America Transactions [Vol no: 18, 2020]
- [3] Gaurav Goswami; Mayank Vatsa; Richa Singh, "Face Verification via Learned Representation on Feature-Rich Video Frames", IEEE Transactions on Information Forensics and Security [Vol no: 12, 2017]
- [4] Huilin Li; Yannan Li; Yong Yu; Baocang Wang; Kefei Chen, "A Blockchain-BasedTraceableSelf-Tallying E-VotingProtocolinAI Era",IEEETransactions on Network Science and Engineering [Vol no: 8, 2021]
- [5] Jiaojiao Zhao; Jungong Han; Ling Shao, "Unconstrained Face Recognition Using a Set-to-Set Distance Measure on Deep Learned Features", IEEE Transactionson Circuitsand SystemsforVideoTechnology [Volno: 28, 2018]
- [6] Jiwen Lu; Gang Wang; Jie Zhou, "Simultaneous Feature and Dictionary Learning for Image Set Based Face Recognition", IEEE Transactionson Image Processing [Vol no: 26, 2017]
- [7] Jiwen Lu; Junlin Hu; Yap-Peng Tan, "Discriminative Deep Metric Learning for Face and Kinship Verification", IEEE Transactions on Image Processing [Vol no: 26, 2017]
- [8] Lei Zhang; Ji Liu; Bob Zhang; David Zhang; Ce Zhu, "Deep Cascade Model- Based Face Recognition: When Deep-Layered Learning Meets Small Data", IEEE Transactions on Image Processing [Vol no: 29, 2019]
- [9] LUISPANIZOALONSO;MILAGASCÓ;DAVIDY.MARCOSdelBLANCO; José Á. Hermida Alonso, "E-Voting System Evaluation Based on The Council of Europe Recommendations: Helios Voting", IEEE Transactions on Emerging Topics in Computing [Vol no: 9, 2018]
- [10] Nirnimesh Ghose; Bocan Hu; Yan Zhang; Loukas Lazos,"Secure Physical Layer Voting", IEEE Transactions on Mobile Computing [Vol no: 17, 2018]

- [11] Ran He; Xiang Wu; Zhenan Sun; Tieniu Tan, "WassersteinCNN: Learning Invariant Features for NIR-VIS Face Recognition", IEEE Transactions on Pattern Analysis and Machine Intelligence [Vol no: 41, 2019]
- [12] Saber Salehkaleybar; Arsalan Sharif-Nassab; S. Jamaloddin Golestani, "Distributed Voting/Ranking With Optimal Number of States per Node", IEEE Transactions on Signal and Information Processing over Networks [Vol no: 1, 2015]
- [13] Somnath Panja; Samiran Bag; Feng Hao; Bimal Roy, "A Smart Contract System for Decentralized Borda Count Voting", IEEE Transactions on Engineering Management [Vol no: 67, 2020]
- [14] Xiaojuan Cheng; Jiwen Lu; Bo Yuan; Jie Zhou, "Face Segmentor-Enhanced Deep Feature Learning for Face Recognition", IEEE Transactions on Biometrics, Behavior, and Identity Science [Vol no: 1, 2019]
- [15] Yuqi Zhang; Yongzhen Huang; Shiqi Yu; Liang Wang, "Cross-View Gait Recognition by Discriminative Feature Learning", IEEE Transactions on Image Processing [Vol no: 29, 2019].

APPENDIX

A. SOURCECODE

```
from flask import Flask, app,request
from imutils.video import VideoStream
from datetime import datetime
from datetime import date
import numpy as np
import dlib
import argparse
from base64 import b64decode
import imutils
import time
import cv2
import pickle
from imutils import face_utils #
import tensorflow as tf
import urllib.request
import urllib.parse
import string
import random
import time
# from datauri import DataURI
from flask import jsonify
from flask_cors import CORS, cross_origin
import glob
import json
from project.utils import Conf
from imutils.video import VideoStream
from datetime import datetime
from datetime import date
from tinydb import TinyDB
from tinydb import where
import face_recognition
import numpy as np
import argparse
import imutils
import pickle
import time
import cv2
from imutils import face_utils

# construct the argument parser and parse the arguments ap
= argparse.ArgumentParser()
ap.add_argument("-c", "--conf", default="config/config.json",
    help="Path to the input configuration file")
args = vars(ap.parse_args())
```

```

#loadtheconfigurationfile
conf = Conf(args["conf"])
prevPerson = None
curPerson = None
consecCount = 0
db = TinyDB(conf["db_path"])
studentTable = db.table("student")
attendanceTable=db.table("attendance")
recognizer=pickle.loads(open(conf["recognizer_path"],"rb").read())
le=pickle.loads(open(conf["le_path"],"rb").read())

```

```

#print("[INFO]warmingupcamera...") #
vs = VideoStream(src=0).start()
#time.sleep(2.0)

```

```

globalresult_flag

```

```

unknown_flag=0

```

```

process_flag=0
#initializepreviousandcurrentpersontoNone
prevPerson = None
curPerson=None

```

```

#initializeconsecutiverecognitioncountto0
consecCount = 0
imagecount=0

```

```

app=Flask(_name_)
cors = CORS(app)
app.config['CORS_HEADERS']='Content-
Type'face_flag = 0
@app.route('/voting', methods=['GET','POST'])
@cross_origin()
deflogin():
    ifrequest.method=='POST':
        imagecount=0
        result=process(request.json['uri'])
        return result

```

```

defprocess(uri):

```

```

    globalrow_count,names_flag,imagecount
    data_uri = uri

```

```

header,encoded=data_uri.split(", ",1)
data = b64decode(encoded)
f=open("image.png","wb")
f.write(data)
row_count=0
names_flag=0
names=[]
encoding=[]
frame=cv2.imread("image.png")
h, w, _ = frame.shape
img=cv2.cvtColor(frame,cv2.COLOR_BGR2RGB) img
= cv2.resize(img, (640, 480))
img_mean=np.array([127,127,127]) img
= (img - img_mean) / 128
img = np.transpose(img, [2, 0, 1])
img=np.expand_dims(img,axis=0)
rgb=img.astype(np.float32)
rgb=cv2.cvtColor(frame,cv2.COLOR_BGR2RGB)
boxes=face_recognition.face_locations(rgb,
model=conf["detection_method"])

for(top,right,bottom,left)inboxes:
    # draw the face detections on the frame
    cv2.rectangle(frame,(left,top),(right,bottom),(0,255,0),2)
    #calculatethetimerremainingforattendancetobetaken

#checkif atleastonefacehasbeendetected if
len(boxes) > 0:
    encodings=face_recognition.face_encodings(rgb,boxes)
    preds=recognizer.predict_proba(encodings)[0]
    j = np.argmax(preds)
    curPerson=le.classes_[j]
    name =
        studentTable.search(where(curP
erson))[0][curPerson][0]
    return{"success":"true","id":curPerson}
else:
    return{"success":"false"}

if __name__ == '__main__':
    app.run(host='0.0.0.0',port=5000)

#USAGE
#pythonrecognition.py
# import the necessary packages
#frompyimagesearch.utilsimportConf
from project.utils import Conf
fromimutils.videoimportVideoStream
from datetime import datetime

```

```

fromdatetimeimportdate
fromtinydbimportTinyDB
fromtinydbimportwhere
import face_recognition
import numpy as np
importargparse
importutils
#importpyttsx3
import pickle
import time
importcv2
#importonnx
#importonnxruntimeasort
#fromonnx_tf.backendimportprepare #
import dlib
fromimutilsimportface_utils

#constructtheargumentparserandparsethearguments ap
= argparse.ArgumentParser()
ap.add_argument("-c","--conf",default="config/config.json",
    help="Pathtotheinputconfigurationfile")
args=vars(ap.parse_args())

#loadtheconfigurationfile
conf = Conf(args["conf"])

#initializepreviousandcurrentpersontoNone

prevPerson=None
curPerson = None

#initializeconsecutiverecognitioncountto0
consecCount = 0

#initializethedatabase,studenttable,andattendancetable #
objects
db = TinyDB(conf["db_path"])
studentTable = db.table("student")
attendanceTable=db.table("attendance")

#loadtheactualfacerecognitionmodelalongwiththelabelencoder
recognizer = pickle.loads(open(conf["recognizer_path"], "rb").read())
le=pickle.loads(open(conf["le_path"],"rb").read())

#initializethevideostreamandallowthecamerasensortowarmup print("[INFO]
warming up camera...")

```

```

vs=cv2.VideoCapture(0)
#vs=VideoStream(usePiCamera=True).start()
time.sleep(2.0)

#initializepreviousandcurrentpersontoNone

prevPerson = None
curPerson=None

whileTrue:
    #storethecurrenttimeandcalculatethetimedifference
    #betweenthecurrenttimeandthetimefortheclasse
    currentTime = datetime.now()

    #grabthenextframefromthestream,resizeitandflipit #
    horizontally
    ret,frame=vs.read()
    h,w,_=frame.shape
    img=cv2.cvtColor(frame,cv2.COLOR_BGR2RGB)
    img = cv2.resize(img, (640, 480))
    img_mean=np.array([127,127,127])
    img = (img - img_mean) / 128
    img = np.transpose(img, [2, 0, 1])
    img=np.expand_dims(img,axis=0)
    rgb=img.astype(np.float32)

    #converttheframefromRGB(OpenCVordering)todlib #
    ordering (RGB)
    rgb=cv2.cvtColor(frame,cv2.COLOR_BGR2RGB)

    #detectthe(x,y)-coordinatesoftheboundingboxes #
    corresponding to each face in the input image
    boxes=face_recognition.face_locations(rgb,
        model=conf["detection_method"])

    #loopoverthefacedetections
    for(top,right,bottom,left)inboxes:
        # draw the face detections on the frame
        cv2.rectangle(frame,(left,top),(right,bottom),(0,255,0),2)
    #calculatethetimerremainingforattendancetobetaken

    #checkifatleastonefacehasbeendetected if
    len(boxes) > 0:
        #computethefacialembodimentforthe face

```

```

        encodings=face_recognition.face_encodings(rgb,boxes)
        #results=
face_recognition.compare_faces(output/encodings.pickle, encodings,
tolerance=0.5)
        #results=face_recognition.compare_faces(recognizer,rgb,
tolerance=0.5)

        # perform classificationto recognize theface
        preds = recognizer.predict_proba(encodings)[0]
        #preds=recognizer.predict_proba(encodings)[0] j
        = np.argmax(preds)
        curPerson= le.classes_[j]

        #ifthepersonrecognizedisthesameasintheprevious #
        framethenincrementtheconsecutive count
        ifprevPerson==curPerson: consecCount
            += 1

        #otherwise,thesearetwodifferentpeoplesoresetthe #
        consecutive count
        else:
            consecCount=0

        #setcurrentpersontopreviouspersonforthenext #
        iteration
        prevPerson=curPerson

        #ifaparticularpersonisrecognizedforagiven try:

            name=studentTable.search(where(
                curPerson))[0][curPerson][0]
            #ttsEngine.say("{}yourattendancehasbeen
taken.".format(
                #      name))
            #ttsEngine.runAndWait()
        except:
            continue

        #constructalabelsayingthestudenthastheirattendance #
        taken and draw it on to theframe
        label="{},youarenowmarkedaspresentin
        {}".format(name,conf["class"])
        cv2.putText(frame,label, (5, 175),
            cv2.FONT_HERSHEY_SIMPLEX,0.5,(255,0,0),2)

```

```

        # show the frame
        cv2.imshow("AttendanceSystem",frame)
        key = cv2.waitKey(1) & 0xFF

        #checkifthe`q`keywaspressed if
        key == ord("q"):
            break

#cleanup
print("[INFO]cleaningup...")
print('label',label)

vs.stop()
#db.close()
import cv2
importdlib
#Initializethecamera
cap=cv2.VideoCapture(0)

whileTrue:
    #Readaframefromthecamera
    ret, frame = cap.read()

    #Displaytheframe
    cv2.imshow("FaceRecognition",frame)

    #Breaktheloopif'q'ispressed
    ifcv2.waitKey(1)&0xFF==ord('q'):
        break

#Releasethecamera
cap.release()

#CloseallOpenCVwindows cv2.destroyAllWindows()

#USAGE
#pythontrain_model.py

#importthenecessarypackages
from project.utils import Conf
fromsklearn.preprocessingimportLabelEncoder from
sklearn.svm import SVC
importargparse
importpickle

#constructtheargumentparserandparsethearguments ap
= argparse.ArgumentParser()

```

```

ap.add_argument("-c","--conf",default="config/config.json",
    help="Path to the input configuration file")
args=vars(ap.parse_args())

#load the configuration file
conf = Conf(args["conf"])

#load the face encodings
print("[INFO] loading face encodings...")
data=pickle.loads(open(conf["encodings_path"],"rb").read())

# encode the labels
print("[INFO] encoding labels...")
le = LabelEncoder()
labels=le.fit_transform(data["names"])

#train the model used to accept the 128-d encodings of the face and # then
# produce the actual face recognition
print("[INFO] training model...")
recognizer=SVC(C=1.0,kernel="linear",probability=True)
print(recognizer)
recognizer.fit(data["encodings"],labels)

#writetheactualfacerecognitionmodeltodisk
print("[INFO] writing the model to disk...")
f=open(conf["recognizer_path"],"wb")
f.write(pickle.dumps(recognizer))
f.close()

#writethelabelencodertodisk f =
open(conf["le_path"], "wb")
f.write(pickle.dumps(le))
f.close()
const SHA256=require("crypto-js/sha256");
class CryptoBlock {
  constructor(index,timestamp,data,precedingHash="")
  {this.index = index;
    this.timestamp=timestamp;
    this.data = data;
    this.precedingHash=precedingHash;
    this.hash = this.computeHash();
    this.nonce = 0;
  }

  computeHash()
  {
    return
    SHA256(this.index+
    this.timestamp+
    JSON.stringify(this.data)+

```

```

        this.nonce
    ).toString();
}

proofOfWork(difficulty)
{while(
    this.hash.substring(0,difficulty)!=Array(difficulty+1).join("0")
){
    this.nonce++;
    this.hash=this.computeHash();
}
}
}

classCryptoBlockchain
{constructor(){
    this.blockchain=[this.startGenesisBlock()];
    this.difficulty = 4;
}
startGenesisBlock(){
    returnnewCryptoBlock(0,"01/01/2020","InitialBlockintheChain","0");
}

obtainLatestBlock(){
    returnthis.blockchain[this.blockchain.length-1];
}
addNewBlock(newBlock){
    newBlock.precedingHash=this.obtainLatestBlock().hash;
    //newBlock.hash=newBlock.computeHash();
    newBlock.proofOfWork(this.difficulty);
    this.blockchain.push(newBlock);
}

checkChainValidity(){
    for(leti=1;i <this.blockchain.length;i++)
    {constcurrentBlock=this.blockchain[i];
    constprecedingBlock=this.blockchain[i-1];

    if(currentBlock.hash!=currentBlock.computeHash())
    {returnfalse;
    }
    if(currentBlock.precedingHash!=precedingBlock.hash)returnfalse;
    }
    returntrue;
}
}

let smashingCoin = new CryptoBlockchain();

console.log("smashingCoin mining in progress.");

```

```

smashingCoin.addNewBlock(new
  CryptoBlock(1,"01/06/2020",
    {sender: "Iris Ljesnjanin",
      recipient:"CosimaMielke",
      quantity: 50
    })
);

smashingCoin.addNewBlock(new
  CryptoBlock(2,"01/07/2020",
    {sender: "Vitaly Friedman",
      recipient:"RicardoGimenes",
      quantity: 100
    })
);

console.log(JSON.stringify(smashingCoin,null,4));

const cuid = require('cuid')
const{da}=require('date-fns/locale')

const db = require('./db')

constData=db.model('candidate_creation', {
  _id: { type: String, default: cuid },
  name:{type:String,required:true},
  age: { type: String, required: true},
  gender:{type:String,required:true},
  father_name: { type: String, required: true},
  mother_name:{ type:String, required: true},
  aadhar_number:{type:String,required:true,unique:true},
  password:{ type:String,required:true},
  mobile_number:{type:String,required:true},
  Address:{ type: String, required: true},
  party_name:{type:String, required: true},
  constitunency:{type:String,required:true},

  promises:{type:[String],required:true},
  status:{type:String,default:"FALSE"},
})

module.exports=
{list,
create,
login,
get,
edit,
model:Data
}

```

```

asyncfunctionlist(){
  constdata=awaitData.find({})
  return data
}
asyncfunctionget(_id){
  constuser=awaitData.findOne({_id})
  return user
}

asyncfunctionlogin(id,pass){
  constdata=awaitData.find({aadhar_number:id,password:pass,status:"TRUE"}) if
  (data.length == 0)
  {
    return{response:"False"}
  }
  else{
    return{response:"True",data:data}
  }
}
asyncfunctioncreate(fields)
{constdata=newData(fields)
awaitdata.save()
returndata
}

asyncfunctionedit(_id,change)
{const user = await get({_id})
Object.keys(change).forEach(function(key)
  {user[key]=change[key]
  })
awaituser.save()
returnuser
}

constmongoose=require('mongoose')
const{response}=require('express')

mongoose.connect(process.
  env.MONGO_URI||
  'mongodb+srv://test:test@cluster0.1eajoy5.mongodb.net/voting?retryWrites=true&w=majo
  rity',

  {useNewUrlParser:true,useCreateIndex:true,useUnifiedTopology:true}
)

module.exports=mongoose

const cuid = require('cuid')

```

```

const{da}=require('date-fns/locale')

const db = require('./db')

constData=db.model('election_day',{
  _id: { type: String, default: cuid },
  date:{type:String,required:true},
  status:{type:String,default:"INACTIVE"},

})

module.exports=
{list,
create,
get,
edit,
model:Data
}

asyncfunctionlist (mobile){
  constdata=awaitData.find({device:mobile})
  return data
}

asyncfunctioncreate(fields)
{constdata=newData(fields)
awaitdata.save()
returndata
}

asyncfunctionget(_id){
  constuser=awaitData.findOne({_id})
  return user
}

asyncfunctionedit(_id,change)
{const user = await get({_id})
Object.keys(change).forEach(function(key)
  {user[key]=change[key]
  })
awaituser.save()
returnuser
}
constcuid=require('cuid')
const{da}=require('date-fns/locale')

const db = require('./db')

```

```

constData=db.model('user_creation', {
  _id: { type: String, default: cuid },
  name:{type:String,required:true},
  age: { type: String, required: true},
  gender:{type:String,required:true},
  father_name: { type: String, required: true},
  mother_name:{ type:String, required: true},
  aadhar_number:{type:String,required:true,unique:true},
  password:{ type:String,required:true},
  mobile_number:{type:String,required:true},
  Address:{type:String,required:true},
})

module.exports=
{list,
create,
login,
get,
edit,
model:Data
}

asyncfunctionlist (mobile){
  constdata=awaitData.find({device:mobile})
  return data
}

asyncfunctionlogin(id,pass){
  constdata=awaitData.find({aadhar_number:id,password:pass}) if
  (data.length == 0)
  {
    return{response:"False"}
  }
  else{
    return{response:"True"}
  }
}

asyncfunctioncreate(fields)
{constdata=newData(fields)
awaitdata.save()
returndata
}

asyncfunctionget(_id){
  constuser=awaitUser.findOne({_id})
  return user
}

```

```

asyncfunctionedit(_id,change)
{const user = await get({_id})
Object.keys(change).forEach(function(key)
{user[key]=change[key]
})
awaituser.save()
returnuser
}

constcuid=require('cuid')
const{da}= require('date-fns/locale')
constSHA256=require("crypto-js/sha256");

const db = require('./db')

constData=db.model('vote_creation', {
  _id: { type: String, default: cuid },
  date:{type:String,required:true},
  aadhar_id: { type: String, required: true},
  party_name:{ type:String, required: true},
  index :{ type: String, required: true},
  timestamp : { type: String, required: true},
  precedingHash:{type:String,required:true},
  hash: { type:String,required:true},
  nonce:{type:String,default:0},
})

module.exports=
{list,
create,
check_eligible,
total_vote_count,
edit,
create_genesis_block,
checkChainValidity,
model:Data
}

asyncfunctionlist(){
constdata=awaitData.find({})
return data
}

asyncfunctiontotal_vote_count(date1)
{ const data = await
Data.find({date:date1})returndata
}

asyncfunctioncheck_eligible(aadhar_id1,date1){

```

```

console.log(aadhar_id1,date1)
constdata=awaitData.find({date:date1,aadhar_id:aadhar_id1}) console.log(data.length)
if(data.length== 0)
{
return{response:"True"}
}
else{
return{response: "False"}

}
}

asyncfunctioncreate(fields)
{letsize=awaitData.find({})
fields.index = size.length-1

fields.precedingHash=size[fields.index].hash
fields.hash = await
computeHash(fields.index,fields.precedingHash,fields.timestamp,fields.aadhar_id,
fields.party_name,fields.date,fields.nonce)
constdata=newData(fields)
awaitdata.save()
returndata
}
asyncfunctioncreate_genesis_block(fields){

fields.hash=awaitcomputeHash("0","0",fields.timestamp,"0","0","0","0") const
data = new Data(fields)
awaitdata.save()
return data
}

asyncfunctionedit(_id,change)
{const user = await get({_id})
Object.keys(change).forEach(function(key)
{Data[key]=change[key]
})
awaituser.save()
returnuser
}

asyncfunctionget(_id){
constuser=awaitData.findOne({_id})
return user
}

```

```

async function
computeHash(index,precedingHash,timestamp,aadhar,candidate,date,nonce) {
  return
    SHA256(index+
    precedingHash+
    timestamp +
    aadhar +
    candidate+
    date+
    nonce
  ).toString();
}

```

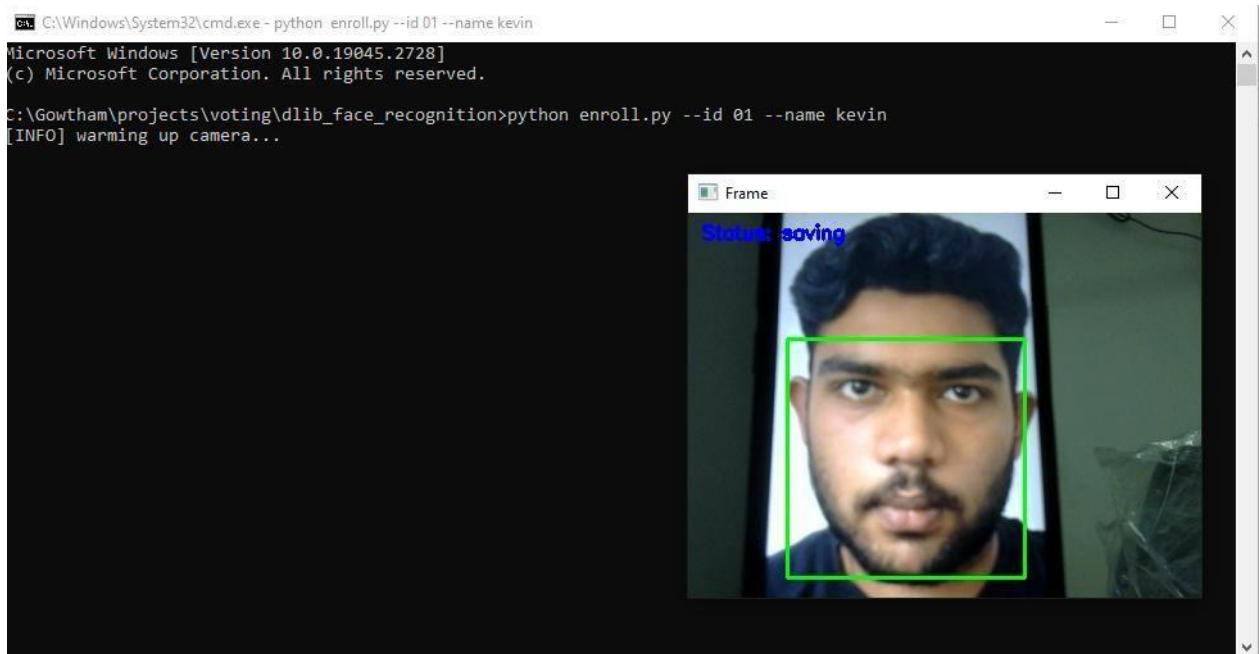
```

asyncfunctioncheckChainValidity()
{letblockchain=awaitData.find({})

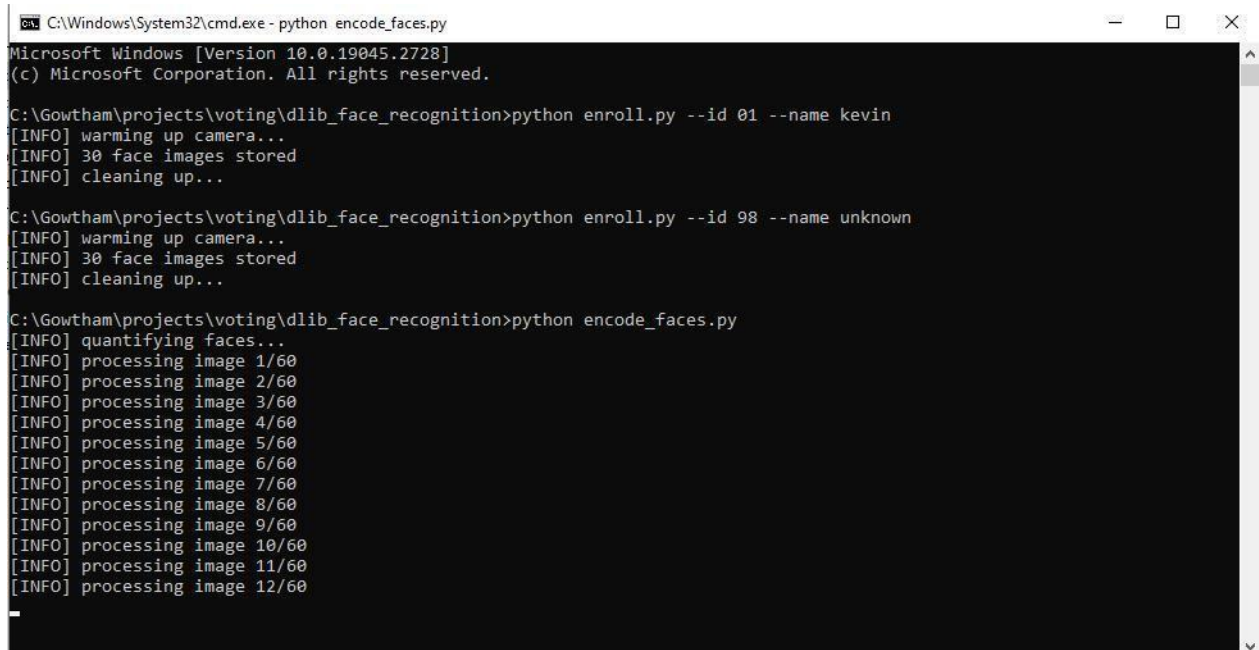
  for(leti=1;i<blockchain.length;i++)
  {const currentBlock= blockchain[i];
  constprecedingBlock=blockchain[i-1]; let
  checkhash =await
computeHash(currentBlock.index,currentBlock.precedingHash,currentBlock.timest
amp,currentBlock.aadhar_id,currentBlock.party_name,currentBlock.date,currentBl
ock.nonce)
  //console.log(currentBlock.hash,checkhash)
  if (currentBlock.hash !==checkhash )
  {returnfalse;
  }
  if(currentBlock.precedingHash!==precedingBlock.hash)returnfalse;
}
returntrue;
}

```

B. SCREENSHOTS



FigureB.1Enrollingfaces



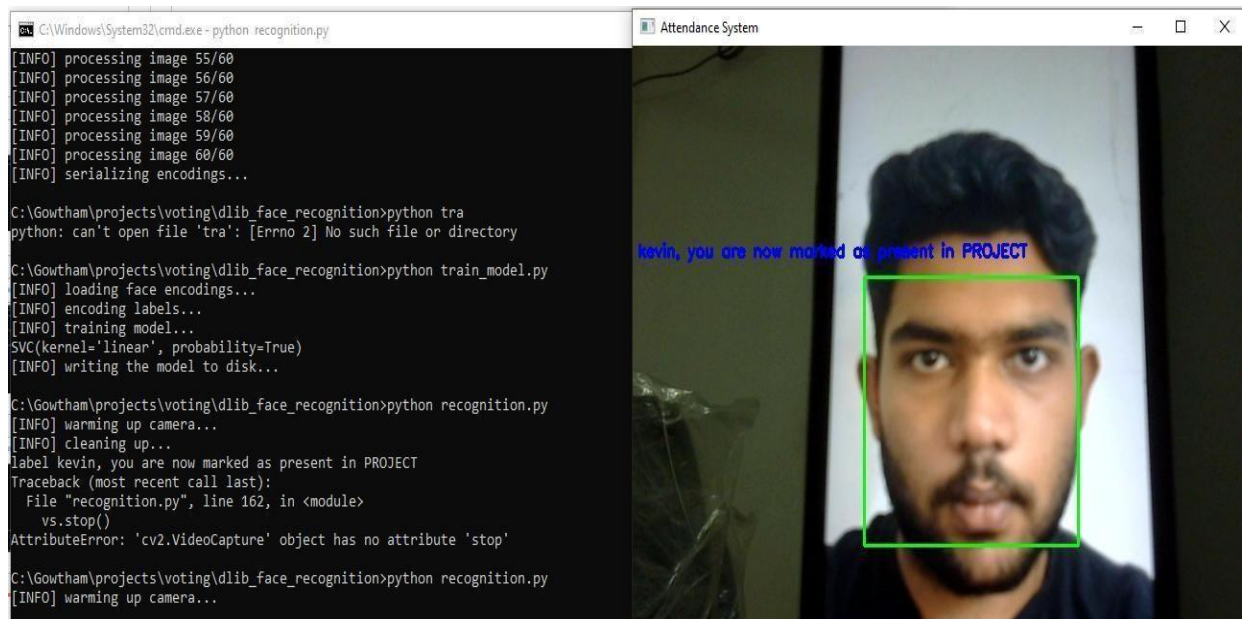
FigB.2:Encodingfaceembedding

```
C:\Windows\System32\cmd.exe
[INFO] processing image 41/60
[INFO] processing image 42/60
[INFO] processing image 43/60
[INFO] processing image 44/60
[INFO] processing image 45/60
[INFO] processing image 46/60
[INFO] processing image 47/60
[INFO] processing image 48/60
[INFO] processing image 49/60
[INFO] processing image 50/60
[INFO] processing image 51/60
[INFO] processing image 52/60
[INFO] processing image 53/60
[INFO] processing image 54/60
[INFO] processing image 55/60
[INFO] processing image 56/60
[INFO] processing image 57/60
[INFO] processing image 58/60
[INFO] processing image 59/60
[INFO] processing image 60/60
[INFO] serializing encodings...

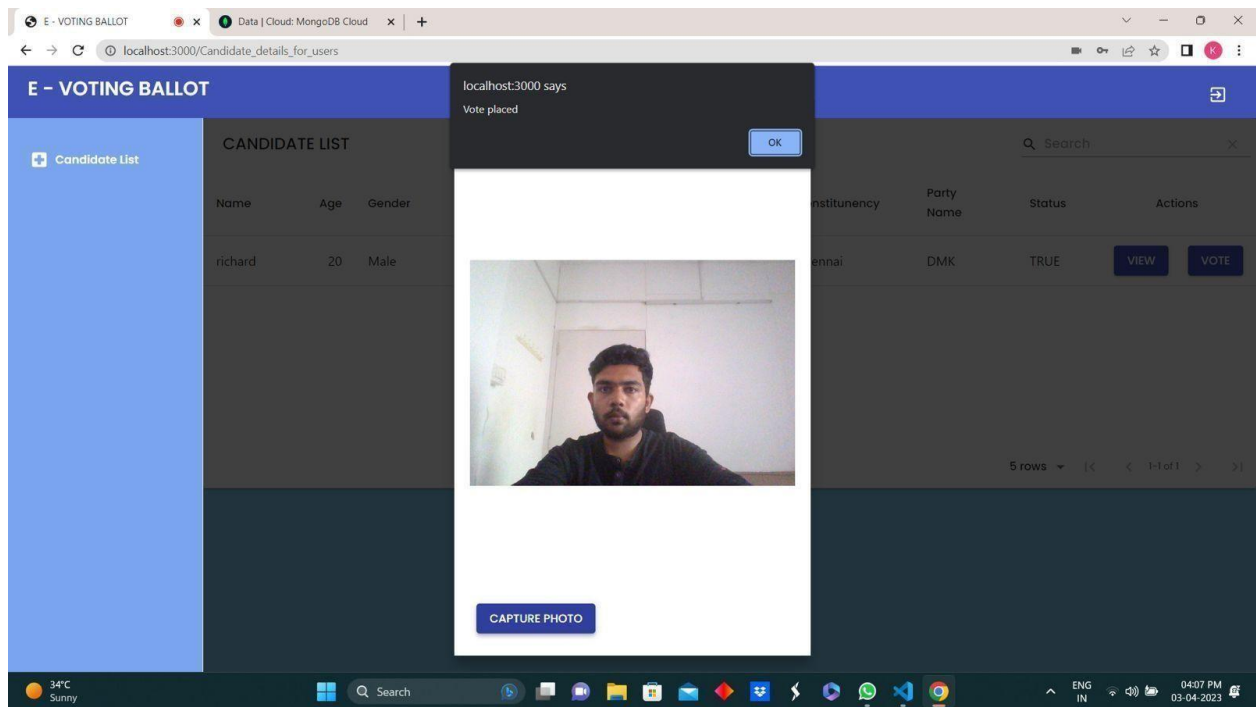
C:\Gowtham\projects\voting\dlib_face_recognition>python train_model.py
[INFO] loading face encodings...
[INFO] encoding labels...
[INFO] training model...
SVC(kernel='linear', probability=True)
[INFO] writing the model to disk...

C:\Gowtham\projects\voting\dlib_face_recognition>
```

FigB.3: Training face embedding using SVM



FigB.4: Face recognition



FigB.5:Vote placed



FigB.6:Electionresults

AI BASED E VOTING SYSTEM WITH BLOCKCHAINMINEDDATA STORAGE

BaireddyVenkataDevevendharReddy¹

¹ResearchScholar,DepartmentofComputerScienceandEngineering,

Sathyabama Institute of Science and Technology.

b.v.devendharreddy2@gmail.com

BattulaSahithReddy²

²ResearchScholar,DepartmentofComputerScienceandEngineering,

Sathyabama Institute of Science and Technology.

sahithreddy708@gmail.com

NancyKirupanithi³

³AssistantProfessor,DepartmentofComputerScienceandEngineering,

Sathyabama Institute of Science and Technology.

nancykirupanithi.cse@sathyabama.ac.in

ABSTRACT: Electronic vote casting is a way of undertaking elections using digital vote casting machines, every so often referred to as "EVMs" in India. EVM and automatic vote counting have been advanced and permitted with the aid of the National Equipment Manufacturing Organization of India and Bharat Electronics in the Nineteen Nineties. They have been used inside the Indian elections of 1998 and in 2001. Electronic balloting machines are used in all famous and countrywide elections. Of India In 2014, the balloting device generally makes use of guide voter entry. A line to the region of the body of the voter. To be truthful, the manual balloting device doesnow notallowdigital ballotvoting. The primary motive of this enterprise is to provide political events in terms of their constituencies with humans relatedto the clinical field. This web application is advanced using React JS and is designed for contemporary political parties / non-birthday celebration applicants contesting the imminent well-known elections. The scheme additionally allows a person to conduct a survey to understand their readiness and opinion

inside the modern/subsequent elections. This internet software lets in the country government to provide 100% vote casting in open functions by imparting an electronic system that allows facial recognition to be authenticated.

KEYWORDS: Voting, Electronic, Elections, Manual Voting, Queue, Web Application.

INTRODUCTION:

Indiaisarusticdividedinto statesand union territories, with a parliamentary equipment consistent with the Constitution of India, which determines the distribution of power among the important government and the states. The Political Commission of India is a self-reliant frame empowered to manage thepoliticaldeviceinIndia,theUnionand theStates. Elections tothe Lok Sabha, Rajya Sabha, legislatures and the President and Vice President of the US are governed through the Constitution of India.

India is a communist U. S. A. Primarily based on balloting, hegemonic ideology and majority rule. "We, the human beings of India" is woven together like a thread inside the social, economic and affairs of state given to us by way of the constitution of the majority rule of the government. The idea of a majority authorities included in the Constitution recollects the representation of the people inside the electricity of the legal guidelines of the nation thru parliamentary elections and their outcomes. The Supreme Court taken into consideration that most of the people of the government is one of the transcendental elements of the Constitution of India and a part of its fundamental form. The Constitution of India adopted a parliamentary gadget of presidency. The Parliament includes the President of India and the homes of Rajya Sabha and Lok Sabha. Since India is a union of states, every country has its personal regulations. Seven states - Andhra Pradesh, Telangana, Bihar, Jammu and Kashmir, Karnataka, Maharashtra and Uttar Pradesh - have the primary representative and homes - the respectable meeting and the legislative meeting - the country governing frame and the legislative assembly. The very last states are 22 states. In addition to the above, of the seven Union Territories, namely Delhi and the State Capital Region of Pondicherry, have installed their own societies.

The Caste Political Association of Government of India is a frame mounted under the provisions of the Constitution and has the duty of overseeing and monitoring all electoral selections in India. The selection framework is loose and a laugh without any attempt. The Election Commission shall make sure that the elections are performed as in step with the regulations before, during and after the election of the filers.

GOALS:

Electronic balloting or electronic voting has been used in numerous bureaucracy for thereason that 1970s, supplying good sized benefits over paper structures, together with growing performance and lowering mistakes. However, there are demanding situations to extensive reception of such frameworks, explicitlywithperceivetoexpandingtheirstrength to strength failure. Block chain is a new-age generation that promises to enhance the overall sustainability of electronic balloting systems. This paper attempts to expand an effective vote casting system based on the blessings of block chain including cryptography and transparency. The proposed concept covers thesimple necessities for electronic voting era and provides for quit-to-give up verification. The paper info thegoals ofe-balloting anditsimplementation

the usage of multi-channel platforms. The paper provides a relatively rated scheme that effectively demonstrates its effectiveness in creating verifiable electronic citizens.

OBJECTIVE:

Electronic voting has been a subject of examination for quite a long time that makes use of automated machines and gadget to cast votes and achieve extremely good and accurate outcomes from voters who participate. In practice, numerous efforts were made to guide the electoral system. Initially, laptop vote counting allowed citizens to vote on paper. These papers have been then scanned and the centrality vote in each cell was calculated (Kadam et al., 2015; 2017 Rockwell; Hao et al., 2010). Later, thegadgetofDirectElectronicVoting(DRE)became delivered, no matter the praise and reputation of voting by means of pc scientists.

LITERATURESURVEY

The following suggests on-line headlines. Existing popular techniques might be discussed underneath.

A framework to make vote casting obvious the usage of block chain generation.

Widespread mistrust of traditional vote casting structures has made popular balloting extraordinarily important in any us of a. People have visible how their simple rights are violated. Other digital voting structures face troubles without transparency. Most voting structures lack sufficient transparency; therefore it is hard for the government to gain the confidence of the citizens. The purpose conventional and modern-day virtual vote casting systems fail is that they may be smooth to apply. The most important purpose is to resolve the issues of conventional and virtual voting, including all of the errors or inequities in the vote casting process. Block chain generation may be utilized in vote casting structures to facilitate fair elections and reduce injustice. Physical vote casting systems have many flaws and digital balloting systems are not advanced enough to be applied on a big scale. He appreciates the want for an answer that ensures human being's democraticrights.This articleoffersaplatform based on modern-day block chain generation, whichenables maximum transparency and reliability of the system to ensure dependable connections between citizens and election officers. The mission provides a platform to offer a framework for balloting digitally through block chain without being blocked by using

any bodily station. The proposed framework helps scalable block chains the usage of flexible consensus mechanisms. The protection link algorithm used in the vote casting gadget makes the voting method very comfy. Brilliant agreements offer a comfortable association between the individual and the organization whilst engaging in non-chain transactions. The protection of the block based voting machine become also mentioned. Additionally, encryption of transactions the use of hash cryptography and fifty one% assault prevention in the block chain has also been achieved. Furthermore, a way has been advanced to address limitations inside the balloting technique the usage of blocks. Finally, evaluation of the proposed system shows that the gadget may be carried out across populations.

Asmart settlement machine for decentralized vote counting at the gate.

In this paper, we recommend the primary decentralized vehicle-counting digital voter protocol for a general vote casting system based on a voter counting portal. Our protocol does not require a credit score account or legal billing account. Voters speak thru a public message board to put into effect the convention in an openly undeniable way. Our prevalent convention includes rounds. In the first round of vote casting they'll post the public key, in the 2nd round they'll publish random votes. All citizens offer non-interactive no-expertise proofs (NIZK) to reveal that they've faithfully followed the protocol description without disclosing their unidentified votes. After the survey is finished, we all, even external the ideal interest group, can arrange inquiries without being legitimate to cast a ballot. We offer assurance confirmation to uncover that our convention guarantees most extreme security for each article. We did our convention the utilization of Ethereum signs on the overall population record to report casting a ballot exchanges as freely obvious exchanges. The results of our experiments demonstrate the protocol's suitability for actual global deployment.

Distributed balloting/rating with maximum quantity of states according to node.

To pick the highest vote in the network of n nodes, we install a unique variety of a couple of votes / order (DMVR), in which each node can first vote for one (or extra) trade. We need care) or order all alternatives according to the votes they get hold of (ranking complexity). The replace algorithm makes use of the country of commonplace nodes with main operations to integrate votes from the community

nodes: merge and break up. The cause of putting in the regulations is very simple, thinking of the dimensions of the region and the quantity of alternatives, without problems adjusting to the usage of the most suitable middle positions $K \times 2$, $K-1$ and $K \times K$! Central orders are required. It is higher for the States to be the following regions: This unit is taken into consideration within the case of a same distribution of votes. The time complexity of the layout rule is characterised by using crucial graphs. We have shown that the complexity of every voter's role and estimation time is $O(\log(n))$ and is inversely proportional to the minimal distinction in the vote proportion between distinct opponents.

The balloting system did no longer boom focus.

Evaluation of electronic voting systems primarily based on the recommendation of the European Council: Helios Voting.

Despite the acclaimed benefits of the e-voting initiative, the substantial adoption and implementation technique of the e-balloting mechanism has been slower than predicted. Many technical, social, and cultural troubles prevent the advent and use of polling. Among them, considering the one of a kind legal and legislative frameworks, the assessment and harmonization of electronic balloting systems stays an important difficulty that wishes attention. However, only a few works on this vicinity were devoted to this topic. This paper objectives to make a contribution to in addition expertise of this research subject matter by means of utilizing a sensible assessment of Helios Casting a ballot, one of the most extreme broadly utilized electronic balloting gear these days. As per the specialized and security necessities gave by involving the European Board in 2017, our framework is loved records for political race officials, analysts and citizens to capture the qualities and shortcomings of the Helios casting a ballot framework. I'm extending it. Dynamic strategies connected with the political race device and environmental factors, which can be done safely the utilization of Helios Casting a ballot. The last objective of our paper is to place in force a drowsy, agreeable, and moral extension of computerized vote projecting.

Electronic Democratic Utilizing Block chain and Brilliant Agreements: Evidence of Concept.

Block chain era changed into brought to defend the requirements of agree with between transactions in digital facts structures. The a success use of crypto currencies has allowed them to discover their capability in business, commercial and service

systems with the assist of operations achieved by way of Ethereum savvy contracts and cryptographic insurance of public and individual keys. These keys are supplied via Network Lock to permit on-line transactions of an unknown writer with a guarantee of safety. Taking into consideration the above, this concept can be extended to electoral methods which could allow its use in digital voting structures, especially whilst presently used protocols lack the ability to differentiate between distinct social actors. This paper offers an idea paper that uses polling and different interoperable technology along with digital vote casting structures to pick character candidates. This is done thru the specification of a framework for the selection method, that's applied and simulated to generate statistics to generate cost, evaluating closed source generation as an opportunity to assist systems.

EXISTING SYSTEM

In a variety of real-world applications, artificial intelligence (AI) has demonstrated significant capability. However, some essential issues consisting of equity, transparency and believe nonetheless undertaking the implementation of AI in consider-based totally applications including digital voting. Electronic voting plays an important position in democratic societies, which requires strong mutual accept as true with between citizens and initiators. The goal of this paper is to make the combination of the artificial intelligence atmosphere by using developing a block chain-based totally track able, self-computing digital voting system. We use event-pushed characteristic institution signatures and time-locked holomorphic puzzles to provide anonymity and responsibility as well as capability for voting extent and e-balloting systems. The proposed e- casting a ballot convention upholds extra features along with multiple selection and automobile- counting. , We display the anonymity, timeliness, confidentiality, connectivity, and complete traceability of the suggested procedure we gauge the gas outflows and time cost of chain tasks, demonstrating that the proposed casting a ballot convention is possible and can be utilized in true applications.

Disadvantages of Existing System

1. Random Internet searches for records about political parties or elections pose a chance of disinformation.
2. The motive of the regulation isn't always to boom voters' awareness of applicants competing of their precincts.

PROPOSED SYSTEM

Electronic balloting is a method of engaging in elections the use of electronic democratic machines, some of the time known as "EVMs" in India. The use of EVMs and electronic democratic was developed and authorize through the State Hardware Company of India and Bharat Gadgets in the Nineties. These were delivered in Indian elections between 1998 and 2001. Electronic balloting machines are being used in all preferred elections and kingdom elections in India given that 2014. In India, the vote projecting gadget typically utilizes manual get admission to, in which citizen's stand in a queue at a bodily area to forge their vote. Manual vote casting system simply does no longer bring about a hundred% vote casting. The essential objective of this framework is to give people from the genuine scientific community the political function of the parties in their constituencies. A net utility turned into evolved using ReactJS which aimed to segment the existing political events/unbiased applicants contesting the approaching public elections. The scheme also allows the common guy to behaviour a survey to express his practise and opinion within the current/upcoming public elections. This net-based totally utility allows the nation authorities to reap 100% vote casting in public elections by using providing an electronic device that provides face-to-face authentication.

Advantages of Proposed System

1. Awareness the various residents about the candidates of their area will increase.
2. Achieved one hundred% early voting inside the location.
3. Provides a relaxed era solution for citizens to vote.
4. Secures the information stored in the database the use of block chain.
5. Instant outcomes.
- 6.

SYSTEM ARCHITECTURE:

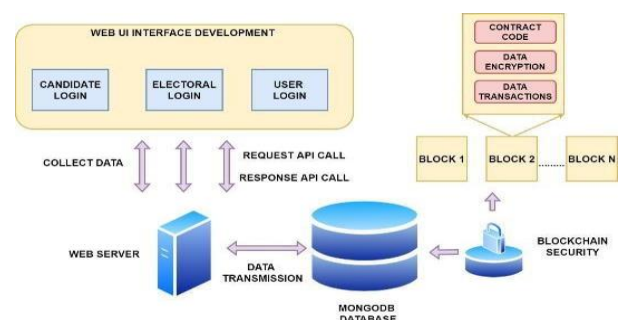


Figure3.1Proposed FrameworkEngineering

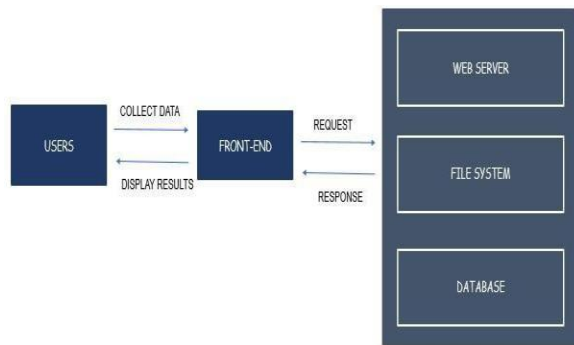


Figure3.2WebArchitecture

WORKING

In this project, we are especially interested by offering political party positions concerning vote casting to actual members of the scientific network. This net application is advanced using ReactJS which pursues to analyse the function of present day political events/independent candidates competing in the approaching government elections. Facial detection and authentication algorithms are used to successfully identify true users to vote on Election Day the usage of the app. Cascade convolution is used for face recognition, and SVM algorithm is used for face popularity. We use the API (Application Programming Interface) NodeJS JavaScript framework to build an internet application database framework. I used NodeJS to create the API and mongo DB to create the database. Integration of Blockchain on the server side for statistics protection, which guarantees maximum facts security by means of getting rid of facts hacking. The scheme also helps the commonplace guy to behaviour a survey to express his preparation and opinion in the cutting-edge/upcoming public elections. This net-based totally utility allows the state authorities to obtain one hundred% balloting in public elections through providing an electronic machine that provides face-to-face authentication.

MODULE DESCRIPTION:

- Face Identification
- Face Acknowledgment
- Hub Programming interface Generation
- MongoDB Generation
- Web Application Advancement
- Blockchain Integration

1 Face Detection

To stumble on face modules, we use the Haar cascade algorithm.

It is an item detection set of rules used to recognize faces in a photo or in real time. The detection algorithm uses part or line capabilities proposed via Viola and Jones of their 2001 paper "Fast item detection the use of prolonged washes of easy features". The algorithm is given a hard and fast of high-quality photos containing faces and negative photos without faces. In storage, patterns are stored in XML documents and may be examined through Open CV methods. This includes models for face recognition, eye recognition, upper and lower body reputation, license plate reputation, etc.

The first contribution is the creation of defeat characterization studies. These picture capabilities make it easy to hit upon edges or traces in a photo or choose regions with sudden modifications in pixel depth.

The darkest places in the Haar item are elements with a value of 1, and the lightest locations are elements with a value of zero. Each of them is liable for detecting a selected characteristic within the photo. For example, an edge, line, or texture in a photo that has a sudden exchange in depth. For instance. In the above photo, the Haar function detects a vertical edge with black pixels at the right and black pixels on the left.

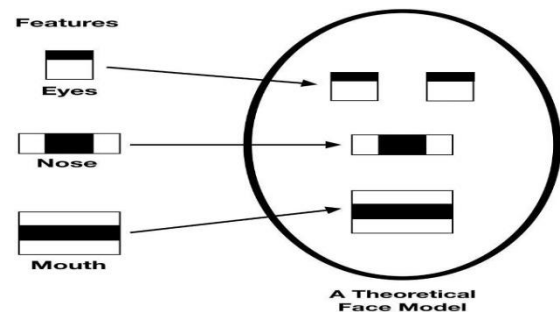


Figure4.1 Haarcascade

2 Face Recognition

We use SVM set of rules for face module detection.

Support vector gadget (SVM) is a supervised gadget gaining knowledge of set of rules that may be used for both category and regression issues. However, it's far specially utilized in class troubles. In the SVM algorithm, we designate every function as a factor in a one-dimensional area (wherein N is the wide variety of traces), and the fee of each feature is a specific coordinate price. We perform

classification via finding the hyper plane that excellent separates the two training. SVMs are utilized in programs which include hand recognition, intrusion detection, face detection, electronic mail type, genetic classification, and net pages.

3 NodeAPI Generation

In this venture we are developing a Programming interface (Application Programming Point of interaction) the utilization of JavaScript JS stage Hub.

Node.js is a server-side, open-source, cross-stage JavaScript runtime that utilizes the V8 motor to run JavaScript code beyond the program. Node.js grants you to record order line gear, make server-viewpoint scripts, and run server-anglescriptsthe use of JavaScript to create dynamic website page content prior to sending it to the customer's program. In this way, Hub.js addresses the "JavaScript all over the place" worldview, binding together webimprovement cycle a solitary programming language as opposed to distinct client-side and server-side dialects feature prearranging.

4 MongoDB Generation

Mongo DB is a pass-platform report-oriented database application with available source code. Described as a NoSQL database program, MongoDB uses JSON files with additional schema. MongoDB Inc. has been developed by way of. And licensed below the Service Public License (SSPL).

A database that consists of physical facts for storage. Each database has its very own rows inside the record gadget. A MongoDB server by and large conveys more than one information bases. An assortment of documents is a MongoDB establishment. The equivalent applies to RDBMS table too. A report is a gathering of key cost matches. Reports have a powerful design. A dynamic schema technique in which files in the same collection do not always have the same fields or shapes, and files within the series may comprise exceptional sorts of records in common fields.

5 Web Application Development

Web development is the development of utility development from a far off server and introduced to the user's gadget through the Internet. Web programs (internet packages) do not need to be released, but are accessed through the network. The give up consumer can help admission to the web programming through an internet browser along with

Google Chrome, Safari or Mozilla Firefox. Most web projects can be written in JavaScript, Flowing Templates (CSS), and HTML5.

Web improvement ordinarily has a short presence cycle, drove through a little team of developers. Buyer-side software is used to carry out front-give up development of internet applications. A computer program that includes an internet browser is the customer. Client-perspective programming by and large utilizes HTML, CSS, and JavaScript. HTML advises the program how to show the pages' substance material at the presentation, even as CSS keeps up with the showed data in the best organization. JavaScript runs on a web page, making a couple of content intuitive.

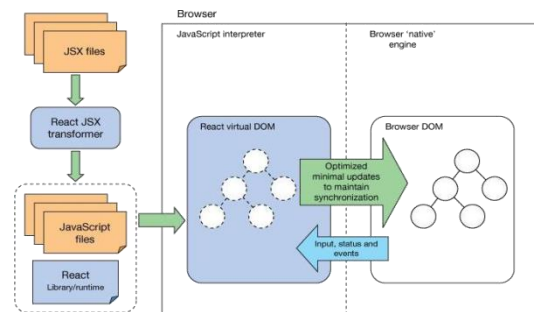


Figure 4.5 Web Application Development

6 Blockchain Integration

As for statistics security, block chain is included into the backup to keep maximum statistics security without statistics traps.

Block chain integration is a powerful technology that creates an immutable file of transactions that may be used to streamline commercial enterprise processes, improve protection, and set up agree with between agencies. By slicing prices it eliminates middlemen and allows them to get things accomplished quickly and as it should be.

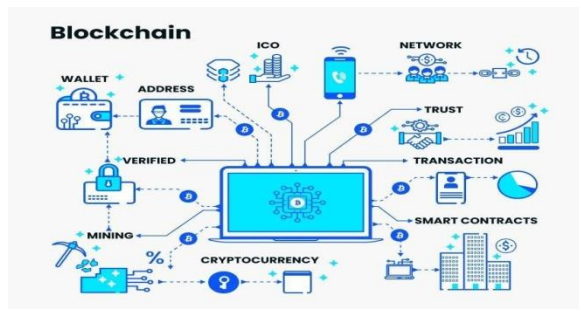


Figure 4.6 Blockchain integration

RESULTSANDDISCUSSIONS

We can run our program into execution blocks to test the instruction model and initialize what is being achieved. The textual content of the application is designed the use of the ReactJS JavaScript framework and the candidate and widespread public registry. The login page can be seen inside the photograph beneath.

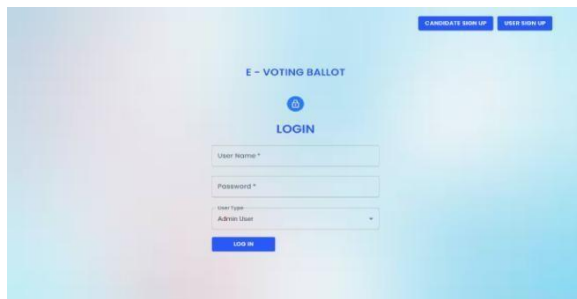


Figure6.1Loginpage

The candidate registration shape is proven within the image beneath. The person registration form may be seen inside the picture below.

Figure6.3Userregistrationform

The major control screen is proven within the parent under.

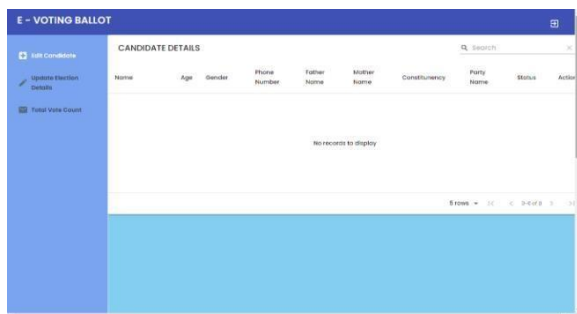


Figure6.4Adminhomescreen

The principal person screen with the list of candidates is shown in the parent below.

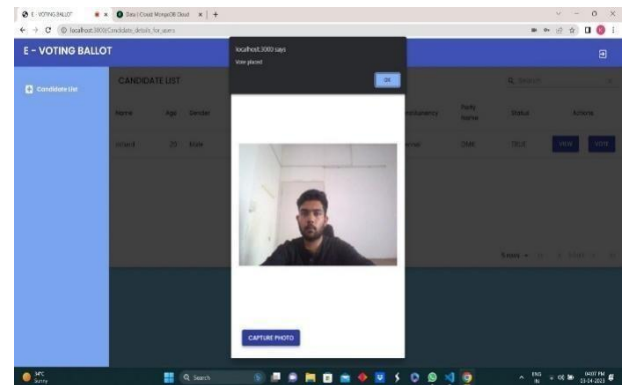


Figure6.7Voteplaced

The final end result of the election can be seen inside the photograph beneath:



Figure6.8Electionresults

If the internal records is covered, the closed nationisright,andthepictureunderneathcanbeseen as follows;

Finds face descriptors for photos inside the database. This can be seen inside the photograph beneath.

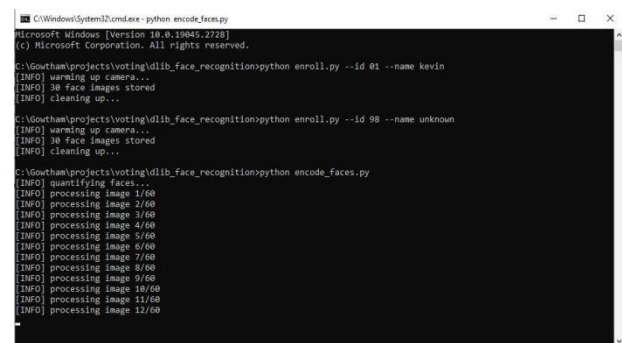


Figure6.11Encodingfaceembedding

Facial popularity the use of the educated SVM model may be visible inside the determine beneath.

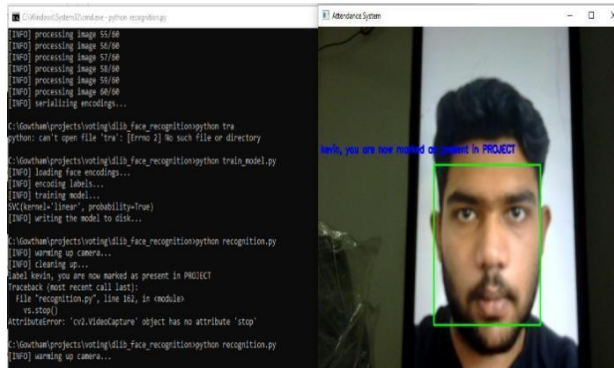


Figure 6.13 Face recognition

CONCLUSION

The undertaking has correctly evolved a web software for a cozier on-line voting device which allows in making sure one hundred% voting percent in state elections by using imparting a digital machine that offers facial recognition authentication. The text of the utility is designed the use of the React JS JavaScript framework and the candidate and trendy public registry. Through this software we effectively cause those too aware about their widespread applicants. In quick, this application presents a low cost and effective manner for the general public to vote safely.

FUTURE WORK

We will see the usage of this scheme with the aid of the Election Commission of India in detail in the future. The Election Commission of India has a whole lot of scope to develop or adjust policy in many ways. Therefore this undertaking has the importance of comfortable electronic balloting within the future, thereby growing the range of electorate and recognition approximately the candidates of their nearby area.

REFERENCE

[1] Christian Galea; Reuben A. Farrugia, "Matching Software-Generated Sketches to Face Photographs With a Very Deep CNN, Morphed Faces, and Transfer Learning", IEEE Transactions on Information Forensics and Security [Vol no: 13, 2018]

[2] Fáber D. Giraldo; Barbosa Milton C.; Carlos E. Gamboa, "Electronic Voting Using Block chain And Smart Contracts: Proof Of Concept", IEEE Latin America Transactions [Vol no: 18, 2020]

[3] Huilin Li; Yannan Li; Yong Yu; Baocang Wang; Kefei Chen, "A Block chain-Based Traceable Self-Tallying E-Voting Protocol in AI Era", IEEE Transactions on Network Science and Engineering [Vol no: 8, 2021]

[4] Jiaojiao Zhao; Jungong Han; Ling Shao, "Unconstrained Face Recognition Using a Set-to-Set Distance Measure on Deep Learned Features", IEEE Transactions on Circuits and Systems for Video Technology [Vol no: 28, 2018]

[5] Jiwen Lu; Junlin Hu; Yap-Peng Tan, "Discriminative Deep Metric Learning for Face and Kinship Verification", IEEE Transactions on Image Processing [Vol no: 26, 2017]

[6] Lei Zhang; Ji Liu; Bob Zhang; David Zhang; Ce Zhu, "Deep Cascade Model-Based Face Recognition: When Deep-Layered Learning Meets Small Data", IEEE Transactions on Image Processing [Vol no: 29, 2019]

[7] LUIS PANIZO ALONSO; MILA GASCÓ; DAVID Y. MARCOS del BLANCO; José A. Hermida Alonso, "E-Voting System Evaluation Based on The Council of Europe Recommendations: Helios Voting", IEEE Transactions on Emerging Topics in Computing [Vol no: 9, 2018]

[8] Nirnimesh Ghose; Bocan Hu; Yan Zhang; Loukas Lazos, "Secure Physical Layer Voting", IEEE Transactions on Mobile Computing [Vol no: 17, 2018]

[9] Ran He; Xiang Wu; Zhenan Sun; Tieniu Tan, "Wasserstein CNN: Learning Invariant Features for NIR-VIS Face Recognition", IEEE Transactions on Pattern Analysis and Machine Intelligence [Vol no: 41, 2019]

[10] Somnath Panja; Samiran Bag; Feng Hao; Bimal Roy, "A Smart Contract System for Decentralized Borda Count Voting", IEEE Transactions on Engineering Management [Vol no: 67, 2020]

[11] Xiaojuan Cheng; Jiwen Lu; Bo Yuan; Jie Zhou, "Face Segmentor-Enhanced Deep Feature Learning for Face Recognition", IEEE Transactions on Biometrics, Behavior, and Identity Science [Vol no: 1, 2019]

[12] Yuqi Zhang; Yongzhen Huang; Shiqi Yu; Liang Wang, "Cross-View Gait Recognition by Discriminative Feature Learning", IEEE Transactions on Image Processing [Vol no: 29, 2019]



12th INTERNATIONAL CONFERENCE ON CONTEMPORARY ENGINEERING AND TECHNOLOGY 2024

ORGANIZED BY
ORGANIZATION OF SCIENCE & INNOVATIVE ENGINEERING AND TECHNOLOGY (OSIET), CHENNAI, INDIA.

IN COLLABORATION WITH
SAMARKAND STATE UNIVERSITY, UZBEKISTAN

IN ASSOCIATION WITH

PRINCE SHRI VENKATESHWARA PADMAVATHY ENGINEERING COLLEGE PRINCE DR. K. VASUDEVAN COLLEGE OF ENGINEERING & TECHNOLOGY

(Approved By All India Council For Technical Education, Affiliated To Anna University)
Medavakkam - Mambakkam Main Road, Ponmar, Chennai - 600 127.



Certificate of Presentation



This is to certify that Mr/Mrs/Dr. **Baireddy Venkata Devevendhar Reddy** from
Sathyabama Institute of Science and Technology has presented a paper titled
AI BASED E VOTING SYSTEM WITH BLOCK CHAIN MINED DATA STORAGE
in the "12th International
Conference on Contemporary Engineering and Technology 2024" held on 23rd & 24th March 2024.


Dr. Akhatov Akmal Rustamovich
Vice Rector of International Affairs,
Samarkand State University, Uzbekistan


Dr. Christo Ananth
Professor,
Samarkand State University, Uzbekistan


Dr. P. Aravinthan, M.E., Ph.D.,
Technical Lead, OSIET
Secretary


Prof. A. Krishnamoorthy, M.E., (Ph.D.)
Technical Lead, OSIET
Administrator


K. Janani, M.Tech.,
CEO, OSIET
Director