

Vote Chain For Secure And Transparent E-Voting Mechanism Using AI And Blockchain Technology

Arshath¹, Mahalingam², Manibharathi³, Sandhoshsivan⁴, Mr. Udhayakumar⁵

^{1,2,3,4}Dept of Computer Science and Engineering

⁵Assistant Professor, Dept of Computer Science and Engineering

^{1,2,3,4,5}Rathinam Technical Campus Coimbatore, Tamil Nadu, India

Abstract- Artificial intelligence (AI) has demonstrated huge potential in a variety of real-world applications. However, some significant considerations like fairness, transparency and trustworthiness are still challenging when applying AI to trust-oriented applications such as E-voting. The technology can ensure the safety of every vote, better and faster and much more accurate counting and automatic tallying. In this project, we aim to facilitate the consolidation of AI ecosystems by developing a blockchain-based traceable self-tallying e-voting system. The proposed system presents a novel voting system by using QR and Fingerprint of Aadhaar card. When the e-voting system is integrated with the Internet of Things, any eligible voter can vote from anywhere as there will be two or more levels of authenticity checks. The system will act as registering module on activating switch by the super admin. For registering module, the QR code of Aadhar is scanned (which is brought by the use) followed by the fingerprint verification. The system permits the elector to cast their vote, block chain technology comes into existence that is integrated within the machine. Each vote is added into each block encrypted by 256-bit SHA hash codes, the hashed block cannot be tampered by any individual as more security is added to the system. By adopting Blockchain within the distribution of information will scale back one in every of the cheating sources of database manipulation. The proposed mechanism of voting using Blockchain not only serves the election conducting bodies but also the voters who get notified in case of any meddling with their votes before the counting announcement.

I. INTRODUCTION

India is the largest democratic and Republic country in the world. In any democratic and republican country elections are necessary and also a heart to the democracy. In a democracy people have the privilege of being ruled by a government of their own choice. People choose their representatives through elections which are the normal features of democracies all over the world. But these elections should be held freely, fairly, transparently and impartially. For this purpose, the constitution of India provides an Election Commission with autonomous (Art 324-329), consisting a

Chief Election Commissioner and other Election Commissioners (at present two other election commissioners). The proposed mechanism of voting using Blockchain not only serves the election conducting bodies but also the voters who get notified in case of any meddling with their votes before the counting announcement.

The different processes involved are:

- Kiosk voting

Voting machines would be located away from traditional polling places, in such convenient locations as malls, libraries, or schools. The voting platforms would still be under the control of election officials, and the physical environment could be modified as needed and monitored (e.g., by election officials, volunteers, or even cameras) to address security and privacy concerns, and prevent coercion or other forms of intervention.

- Remote Internet voting

It seeks to maximize the convenience and access of the voters by enabling them to cast ballots from virtually any location that is Internet accessible. While this concept is attractive and offers significant benefits, it also poses substantial security risks and other concerns relative to civic culture. Current and near-term technologies are inadequate to address these risks.

- Design of secured E-voting systems

It is able to desire with the widespread use of computers and embedded systems. Security is the essential problem should be considered in such systems. This method proposes a new e-voting system that fulfils the security requirements of e-voting. It is based on holomorphic property and blind signature plan. The suggest system is executed on an embedded system which serves as a voting machine. The system employees RFID to store all conditions that comply with the rule of the government to check voter eligibility.

The existing system has lot of problems such as

- Individuals may cast their single vote twice,
- Voting of non-candidates,
- Tampered data and excessive authorities
- Observing remote voting solutions may be more complex/difficult to organise than in person voting.
- There may be information asymmetry between voters who vote in advance and those who vote on Election Day.

II. TECHNOLOGY INVOLVED

A. Front end used- PHP

Using PHP

With PHP, it's a simple matter to embed dynamic activity in web pages. When you give pages the .php extension, they have instant access to the scripting language. From a developer's point of view, all you have to do is write code such as the following: How are you? The opening command. Outside of this construct, everything is sent to the client as direct HTML. So, the text "How are you?" is simply output to the browser; within the PHP tags, the built-in date function displays the current day of the week according to the server's system time. The final output of the two parts looks like this: Hello World. Today is Wednesday. How are you? PHP is a flexible language, and some people prefer to place the PHP construct directly next to PHP code, like this: Hello World.

Today is. How are you? There are also other ways of formatting and outputting information, which I'll explain in the chapters on PHP. The point is that with PHP, web developers have a scripting language that, although not as fast as compiling your code in C or a similar language, is incredibly speedy and that also integrates seamlessly with HTML code. Using PHP, you have unlimited control over your web server. Whether you need to modify HTML on the fly, process a credit card, add user details to a database, or fetch information from a third-party website, you can do it all from within the same PHP files in which the HTML itself resides.

BACK END

Using MySQL

Of course, there's not a lot of point to being able to change HTML output dynamically unless you also have a means to track the changes that users make as they use your website. In the early days of the Web, many sites used "flat"

text files to store data such as usernames and passwords. But this approach could cause problems if the file wasn't correctly locked against corruption from multiple simultaneous accesses.

Also, a flat file can get only so big before it becomes unwieldy to manage not to mention the difficulty of trying to merge files and perform complex searches in any kind of reasonable time. That's where relational databases with structured querying become essential. And MySQL, being free to use and installed on vast numbers of Internet web servers, rises superbly to the occasion. It is a robust and exceptionally fast database management system that uses English-like commands. The highest level of MySQL structure is a database, within which you can have one or more tables that contain your data. For example, let's suppose you are working on a table called users, within which you have created columns for surname, first name, and email, and you now wish to add another user. One command that you might use to do this is: `INSERT INTO users VALUES ('Smith', 'John', 'jsmith@mysite.com');` Of course, as mentioned earlier, you will have issued other commands to create the database and table and to set up all the correct fields, but the INSERT command here shows how simple it can be to add new data to a database. The INSERT command is an example of SQL (which stands for Structured Query Language), a language designed in the early 1970s and reminiscent of one of the oldest programming languages, COBOL. It is well suited, however, to database queries, which is why it is still in use after all this time. It's equally easy to look up data. Let's assume that you have an email address for a user and you need to look up that person's name. To do this, you could issue a MySQL query such as: `SELECT surname, first name FROM users WHERE email='jsmith@mysite.com';` MySQL will then return Smith, John and any other pairs of names that may be associated with that email address in the database. As you'd expect, there's quite a bit more that you can do with MySQL than just simple INSERT and SELECT commands. For example, you can join multiple tables according to various criteria, ask for results in a variety of different orders, make partial matches when you know only part of the string that you are searching for, return only the nth result, and a lot more. Using PHP, you can make all these calls directly to MySQL without having to run the MySQL program yourself or use its command-line interface. This means you can save the results in arrays for processing and perform multiple lookups, each dependent on the results returned from earlier ones, to drill right down to the item of data you need.

For even more power, as you'll see later, there are additional functions built right into MySQL that you can call up for common operations and extra speed.

The Apache Web Server

In addition to PHP, MySQL, JavaScript, and CSS, there's actually a fifth hero in the dynamic Web: the web server. In the case of this book, that means the Apache web server. We've discussed a little of what a web server does during the HTTP server/client exchange, but it actually does much more behind the scenes. For example, Apache doesn't serve up just HTML files—it handles a wide range of files, from images and Flash files to MP3 audio files, RSS (Really Simple Syndication) feeds, and more. Each element a web client encounters in an HTML page is also requested from the server, which then serves it up. But these objects don't have to be static files, such as GIF images. They can all be generated by programs such as PHP scripts. That's right: PHP can even create images and other files for you, either on the fly or in advance to serve up later. To do this, you normally have modules either precompiled into Apache or PHP or called up at runtime. One such module is the GD library (short for Graphics Draw), which PHP uses to create and handle graphics. Apache also supports a huge range of modules of its own. In addition to the PHP module, the most important for your purposes as a web programmer are the modules that handle security. Other examples are the Rewrite module, which enables the web server to handle a varying range of URL types and rewrite them to its own internal requirements, and the Proxy module, which you can use to serve up often-requested pages from a cache to ease the load on the server. Later in the book, you'll see how to actually use some of these modules to enhance the features provided by the core technologies we cover. About Open Source Whether or not being open source is the reason these technologies are so popular has often been debated, but PHP, MySQL, and Apache are the three most commonly used tools in their categories. What can be said, though, is that being open source means that they have been developed in the community by teams of programmers writing the features they themselves want and need, with the original code available for all to see and change. Bugs can be found and security breaches can be prevented before they happen. There's no worrying about having to purchase additional licenses if you have to scale up your website and add more servers. And you don't need to check the budget before deciding whether to upgrade to the latest versions of these products.

What Is a WAMP, MAMP, or LAMP?

WAMP, MAMP, and LAMP are abbreviations for “Windows, Apache, MySQL, and PHP,” “Mac, Apache, MySQL, and PHP,” and “Linux, Apache, MySQL, and PHP,” respectively. These abbreviations describe a fully functioning setup used for developing

dynamic Internet web pages. WAMPs, MAMPs, and LAMPs come in the form of a package that binds the bundled programs together so that you don't have to install and set them up separately. This means you can simply download and install a single program and follow a few easy prompts to get your web development server up and running in the quickest time with the minimum hassle. The security configurations of such an installation will not be as tight as on a production web server, because it is optimized for local use. For these reasons, you should never install such a setup as a production server. However, for developing and testing websites and applications, one of these installations should be entirely sufficient.

Database

A database is simply a collection of user data just like phone book. MySQL database include such objects as tables, queries, forms, and more.

Tables

In MySQL tables are collection of similar data. With all tables can be organized differently, and contain mostly different information- but they should all be in the same database file. For instance, we may have a database file called video store. Containing tables named members, tapes, reservations and so on.

Relational database

MySQL is a relational database. Relational databases tools like access can help us manage information in three important ways.

- Reduce redundancy
- Facilitate the sharing of information
- Keep data accurate.

Fields are place in a table where we store individual chunks of information.

Primary key and other indexed fields

MySQL use key fields and indexing to help speed many database operations. We can tell MySQL, which should be key fields, or MySQL can assign them automatically.

Controls and objects

Queries are access objects us display, print and use our data. They can be things like field labels that we drag

around when designing reports. Or they can be pictures, or titles for reports, or boxes containing the results of calculations.

Queries and dynasts

Queries are request to information. When access responds with its list of data, that response constitutes a dynaset. A dynamic set of data meeting our query criteria. Because of the way access is designed, dynasts are updated even after we have made our query.

Forms

Forms are on screen arrangement that make it easy to enter and read data. we can also print the forms if we want to. We can design form our self, or let the access auto form feature.

Reports

Reports are paper copies of dynaset. We can also print reports to disk, if we like. Access helps us to create the reports. There are even wizards for complex printouts.

Properties

Properties are the specification we assigned to parts of our database design. We can define properties for fields, forms, controls and most other access objects.

III. SYSTEM ANALYSIS EXISTING SYSTEM

The existing system is not totally automated. Though the system is computerized to a particular extent, it has to do a lot of manual work.

The existing system has lot of problems such as

- Individuals may cast their single vote twice,
- Voting of non-candidates,
- Tampered data and excessive authorities
- Observing remote voting solutions may be more complex/difficult to organise than in person voting.
- There may be information asymmetry between voters who vote in advance and those who vote on Election Day.

BOTTLENECKS OF THE EXISTING SYSTEM

The existing system has lot of problems such as

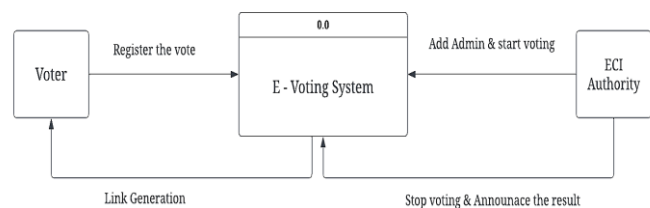
- There may be information asymmetry between voters who vote in advance and those who vote on Election Day.
- Remote voting solutions which take place in an uncontrolled environment may present a higher risk of fraud, coercion, family voting, impersonation, violation of ballot secrecy or other compromises to the integrity of the vote.
- They may have financial and administrative consequences for Member States or for particular hosting institutions (such as hospitals or prisons), depending on whether they
- are introduced in addition to - or instead of - existing methods.
- There may be political disagreement over the method and extent of voting by adiaspora, particularly if this is seen to be politically advantageous to a particular party.

PROPOSED SYSTEM

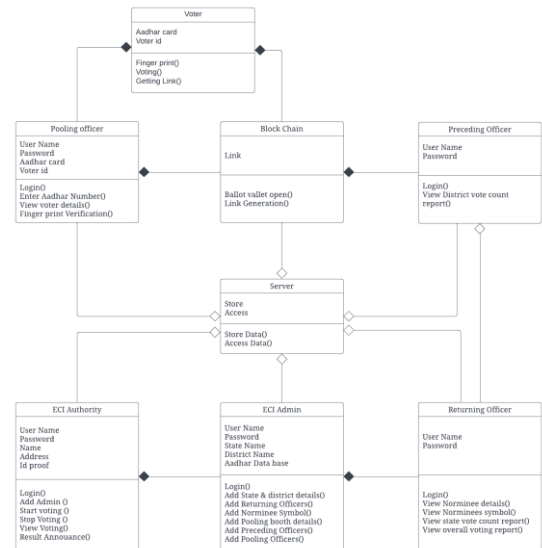
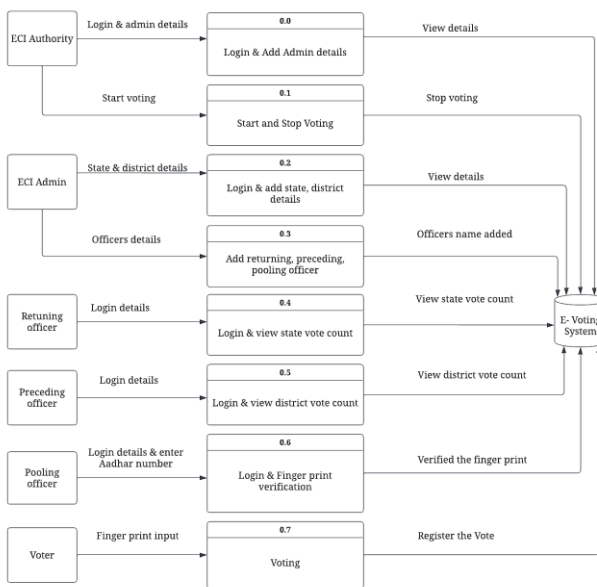
In the proposed solution, all the activities are managed using a Blockchain based mechanism. We have proposed a two-end mechanism in which all the activities are coordinated by the national and state bodies at various levels and voters play an equal part in it. The integration of Blockchain mechanism and voting system may reduce the risks with transparent and decentralized feature of Blockchain technology.

IV. DATAFLOW DIAGRAM

LEVEL 0



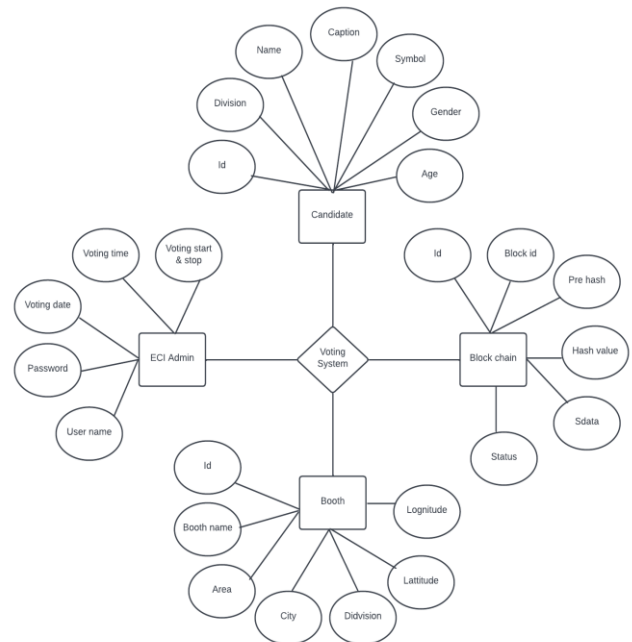
LEVEL 1



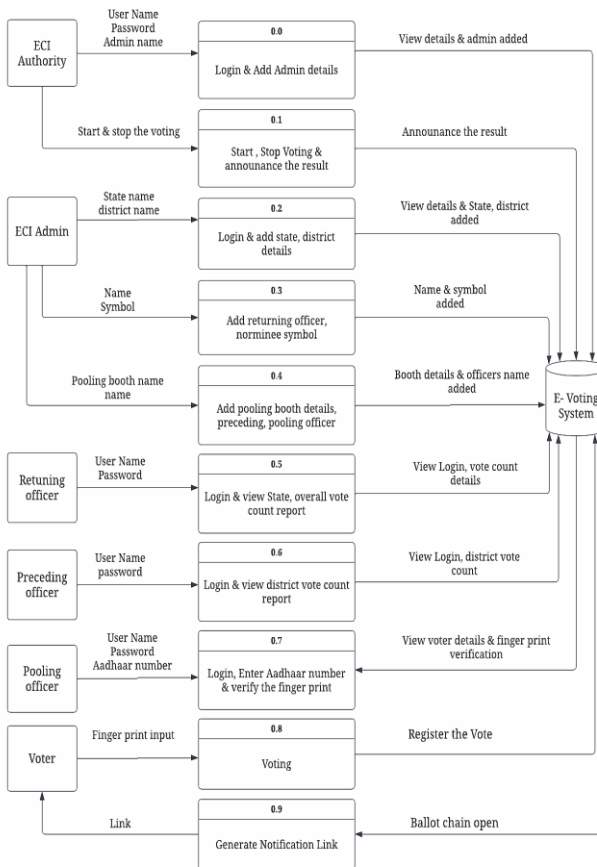
ER DIAGRAM

Scope

The main objective of the project is to bring a full-fledged computerized organization, and to enable the transaction details to maintain records, which makes of the employees easier. Computerization of the entire system will enhance more accuracy and reduces major part of clerical works. Fast, clear and legible reports can be generated without any ambiguity. Integrated database design and ease of maintenance is a major advantage of the system. User friendliness is a unique feather of the system.



LEVEL 2



UML DIAGRAMS

CLASS DIAGRAM

REFERENCES

- [1] K. Curran, "E-voting on the blockchain," =e Journal of British Blockchain Association, vol. 1, no. 22–7, 2018.
- [2] N. Kshetri and J. Voas, "Blockchain-enabled E-voting," IEEE Softw., vol. 35, no. 4, pp. 95-99, Jul. 2018.
- [3] K. Curran, "E-voting on the blockchain," J. Brit. Blockchain Assoc., vol. 1, no. 2, pp. 1-6, Dec. 2018.
- [4] P. Tarasov and H. Tewari, "The future of E-voting," IADIS Int. J. Comput. Sci. Inf. Syst., vol. 12, no. 2, pp. 1-19, 2017.
- [5] F. Al-Turjman, "5G-enabled devices and smart-spaces in social-IoT: An overview," Future Gener. Comput. Syst., vol. 92, pp. 732-744, Mar. 2019.
- [6] www.w3schools.com
- [7] www.php.net/manual/en/tutorial.php
- [8] www.youtube.com
- [9] www.tutorialspoint.com/php/
- [10] www.codecademy.com/en/tracks/php
- [11] <https://code-projects.org>