



JOURNAL OF EMERGING TECHNOLOGIES AND INNOVATIVE RESEARCH (JETIR)

An International Scholarly Open Access, Peer-reviewed, Refereed Journal

FINGER PRINT BASED E-VOTING SYSTEM

Mrs.V.KAVITHA

Mr.M.VEERAMANI

Professor,MCA.,M.Phil.,Ph.D.,

Final MCA Student

PG & Research Department Of Computer Applications(MCA),

Hindusthan College Of Arts And Science (Autonomous), Coimbatore,India.

CHAPTER 1

INTRODUCTION

ABOUT THE PROJECT

Technology has positive impacts on many aspects of our social life. Internet has been a fertile ground for innovation and creativity. The block chain technology is presented as a game changer for many of the existing and emerging technologies/services. With its immutability property and decentralised architecture, it is taking centre stage in many services as an equalisation factor to the current parity between consumers and large corporations/governments. One potential application of the block chain is in e-voting schemes. The objective of such a scheme would be to provide a decentralised architecture to run and support a voting scheme that is open, fair, and independently verifiable. In the project, we propose a potential new e-voting protocol that utilises the block chain as a transparent ballot box. The protocol has been designed to adhere to fundamental e-voting properties as well as offer a degree of decentralisation and allow for the voter to change/update their vote (within the permissible voting period). The project highlights the pros and cons of using block chain for such a proposal from a practical point view in both development/deployment and usage

contexts. Concluding The project is a potential roadmap for block chain technology to be able to support complex applications.

MODULES

SUPER ADMINISTRATOR

- Super Administrator Login
- Programming Details (Vote Description)
- Participants Details
- Decrypt Votes
- Winners Details

BOOTH AGENT ADMINISTRATOR

- New User Registration
- View Users

USER

- User Login
- User Vote
- Encrypt Votes
- View Winners



CHAPTER 2

SYSTEM REQUIREMENTS

HARDWARE SPECIFICATION: (Minimum requirement)

PROCESSOR	:	Intel i3 Processor 2.5GHZ
HARD DISK CAPACITY	:	400 GB
MONITOR	:	15 “SAMTRON MONITOR
INTERNAL MEMORY CARD	:	4 GB
KEYBOARD	:	LOGITECH OF 104 KEYS
CPU CLOCK	:	1.08 GHz
MOUSE	:	LOGITECH MOUSE

SOFTWARE SPECIFICATION:

OPERATING SYSTEM	:	WINDOWS 7
FRONT END	:	PHP
BACK END	:	MYSQL

ABOUT THE SOFTWARE

PHP

PHP stands for Hypertext Preprocessor. PHP scripts run inside Apache server or Microsoft IIS. PHP and Apache server are free. PHP code is very easy. PHP is the most used server side scripting language. PHP files contain PHP scripts and HTML. PHP files have the extension “php”, “php3”, “php4”, or “phtml”.

USING PHP

- Generate dynamic web pages. PHP can display different content to different user or display different content at different times of the day.

- Process the contents of HTML forms. We can use a PHP to retrieve and respond to the data entered into an HTML form.
- Can create database-driven web pages. A PHP can insert new data or retrieve existing data from a database such a MySQL.

WORKING OF PHP

PHP is a standard HTML file that is extended with additional features. Like a standard HTML file, PHP contains HTML tag that can be interpreted and displayed by a web browser. Anything we could normally place in an HTML file Java applets, Blinking text, server side scripts .we can place in PHP. However, PHP has three important features that make it unique.

- PHP contains server side scripts.
- PHP provides several built-in objects.

HYPER TEXT MARKUP LANGUAGE (HTML)

HTML is an application of the Standard Generalized Markup Language (SGML), which was approved as an international standard in the year 1986. SGML provides a way to encode hyper documents so they can be interchanged.

SGML is also a Meta language for formally describing document markup system. Infact HTML uses SGML to define a language that describes a WWW hyper document's structure and inter connectivity.

Following the rigors of SGML, TBL bore HTML to the world in 1990. Since then, many of us have it to be easy to use but sometimes quite limiting. These limiting factors are being addressed but the World Wide Web Consortium (aka W3c) at MIT. But HTML had to start somewhere, and its success argues that it didn't start out too badly.

Hypertext Markup Language (HTML) is the standard markup language for documents designed to be displayed in a web browser. It can be assisted by technologies such as Cascading Style Sheets (CSS) and scripting languages such as JavaScript. HTML is a computer language devised to allow website creation. These websites can then be viewed by anyone else connected to the Internet. It is relatively easy to learn, with the basics being accessible to most people in one sitting; and quite powerful in what it allows you to create. It is constantly undergoing revision and evolution to meet the demands and requirements of the growing Internet audience under the direction of the W3C, the organisation charged with designing and maintaining the language.

HyperText is the method by which you move around on the web — by clicking on special text called hyperlinks which bring you to the next page. The fact that it is hyper just means it is not linear — i.e. you can go to any place on the Internet whenever you want by clicking on links — there is no set order to do things in. Markup is what HTML tags do to the text inside them. They mark it as a certain type of text (italicised text, for example). HTML is a Language, as it has code-words and syntax like any other language. HTML consists of a series of short codes typed into a text-file by the site author — these are the tags. The text is then saved as a html file, and viewed through a browser, like Internet Explorer or Netscape Navigator.

This browser reads the file and translates the text into a visible form, hopefully rendering the page as the author had intended. Writing your own HTML entails using tags correctly to create your vision. You can use anything from a rudimentary text-editor to a powerful graphical editor to create HTML pages.

The tags are what separate normal text from HTML code. You might know them as the words between the <angle-brackets>. They allow all the cool stuff like images and tables and stuff, just by telling your browser what to render on the page. Different tags will perform different functions. The tags themselves don't appear when you view your page through a browser, but their effects do. The simplest tags do nothing more than apply formatting to some text

Web browsers receive HTML documents from a web server or from local storage and render the documents into multimedia web pages. HTML describes the structure of a web page semantically and originally included cues for the appearance of the document.

HTML elements are the building blocks of HTML pages. With HTML constructs, images and other objects such as interactive forms may be embedded into the rendered page. HTML provides a means to create structured documents by denoting structural semantics for text such as headings, paragraphs, lists, links, quotes and other items. HTML elements are delineated by tags, written using angle brackets. Tags such as and <input /> directly introduce content into the page. Other tags such as <p> surround and provide information about document text and may include other tags as sub-elements. Browsers do not display the HTML tags, but use them to interpret the content of the page.

HTML can embed programs written in a scripting language such as JavaScript, which affects the behavior and content of web pages. Inclusion of CSS defines the look and layout of content. The World Wide Web Consortium (W3C), former maintainer of the HTML and current maintainer of the CSS standards, has encouraged the use of CSS over explicit presentational HTML since 1997.

The first publicly available description of HTML was a document called "HTML Tags", first mentioned on the Internet by Tim Berners-Lee in late 1991. It describes 18 elements comprising the initial, relatively simple design of HTML. Except for the hyperlink tag, these were strongly influenced by SGMLguid, an in-house Standard Generalized Markup Language (SGML)-based documentation format at CERN. Eleven of these elements still exist in HTML 4.

After the HTML and HTML+ drafts expired in early 1994, the IETF created an HTML Working Group, which in 1995 completed "HTML 2.0", the first HTML specification intended to be treated as a standard against which future implementations should be based.

Of course, but since making websites became more popular and needs increased many other supporting languages have been created to allow new stuff to happen, plus HTML is modified every few years to make way for improvements. Cascading Stylesheets are used to control how your pages are presented, and make pages more accessible. Basic special effects and interaction is provided by JavaScript, which adds a lot of power to basic HTML. Most of this advanced stuff is for later down the road, but when using all of these technologies together, you have a lot of power at your disposal.

CASCADING STYLE SHEETS (CSS)

Cascading Style Sheets (CSS) is a style sheet language used for describing the presentation of a document written in a markup language like HTML. CSS is a cornerstone technology of the World Wide Web, alongside HTML and JavaScript. CSS is designed to enable the separation of presentation and content, including layout, colors, and fonts. This separation can improve content accessibility, provide more flexibility and control in the specification of presentation characteristics, enable multiple web pages to share formatting by specifying the relevant CSS in a separate .css file, and reduce complexity and repetition in the structural content.

Separation of formatting and content also makes it feasible to present the same markup page in different styles for different rendering methods, such as on-screen, in print, by voice (via speech-based browser or screen reader), and on Braille-based tactile devices. CSS also has rules for alternate formatting if the content is accessed on a mobile device. The name cascading comes from the specified priority scheme to determine which style rule applies if more than one rule matches a particular element. This cascading priority scheme is predictable.

The CSS specifications are maintained by the World Wide Web Consortium (W3C). Internet media type (MIME type) text/css is registered for use with CSS by RFC 2318 (March 1998). The W3C operates a free CSS validation service for CSS documents. In addition to HTML, other markup languages support the use of CSS including XHTML, plain XML, SVG, and XUL.

CSS has a simple syntax and uses a number of English keywords to specify the names of various style properties. A style sheet consists of a list of rules. Each rule or rule-set consists of one or more selectors, and a declaration block.

Before CSS, nearly all presentational attributes of HTML documents were contained within the HTML markup. All font colors, background styles, element alignments, borders and sizes had to be explicitly described, often repeatedly, within the HTML. CSS lets authors move much of that information to another file, the style sheet, resulting in considerably simpler HTML.

Stands for "Cascading Style Sheet." Cascading style sheets are used to format the layout of Web pages. They can be used to define text styles, table sizes, and other aspects of Web pages that previously could only be defined in a page's HTML.

CSS helps Web developers create a uniform look across several pages of a Web site. Instead of defining the style of each table and each block of text within a page's HTML, commonly used styles need to be defined only once in a CSS document. Once the style is defined in cascading style sheet, it can be used by any page that references the CSS file. Plus, CSS makes it easy to change styles across several pages at once. For example, a Web developer may want to increase the default text size from 10pt to 12pt for fifty pages of a Web site. If the pages all reference the same style sheet, the text size only needs to be changed on the style sheet and all the pages will show the larger text.

While CSS is great for creating text styles, it is helpful for formatting other aspects of Web page layout as well. For example, CSS can be used to define the cell padding of table cells, the style, thickness, and color

of a table's border, and the padding around images or other objects. CSS gives Web developers more exact control over how Web pages will look than HTML does. This is why most Web pages today incorporate cascading style sheets.

CSS is created and maintained through a group of people within the W3C called the CSS Working Group. The CSS Working Group creates documents called specifications. When a specification has been discussed and officially ratified by the W3C members, it becomes a recommendation. These ratified specifications are called recommendations because the W3C has no control over the actual implementation of the language.

JAVASCRIPT

JavaScript is a dynamic computer programming language. It is lightweight and most commonly used as a part of web pages, whose implementations allow client-side script to interact with the user and make dynamic pages. It is an interpreted programming language with object-oriented capabilities. JavaScript was first known as LiveScript, but Netscape changed its name to JavaScript, possibly because of the excitement being generated by Java. JavaScript made its first appearance in Netscape 2.0 in 1995 with the name LiveScript. The general-purpose core of the language has been embedded in Netscape, Internet Explorer, and other web browsers

Client-side JavaScript is the most common form of the language. The script should be included in or referenced by an HTML document for the code to be interpreted by the browser. It means that a web page need not be a static HTML, but can include programs that interact with the user, control the browser, and dynamically create HTML content. The JavaScript client-side mechanism provides many advantages over traditional CGI server-side scripts. For example, you might use JavaScript to check if the user has entered a valid e-mail address in a form field. The JavaScript code is executed when the user submits the form, and only if all the entries are valid, they would be submitted to the Web Server. JavaScript can be used to trap user-initiated events such as button clicks, link navigation, and other actions that the user initiates explicitly or implicitly.

JavaScript can be implemented using JavaScript statements that are placed within the `<script>... </script>` HTML tags in a web page.

You can place the `<script>` tags, containing your JavaScript, anywhere within your web page, but it is normally recommended that you should keep it within the `<head>` tags.

The `<script>` tag alerts the browser program to start interpreting all the text between these tags as a script.

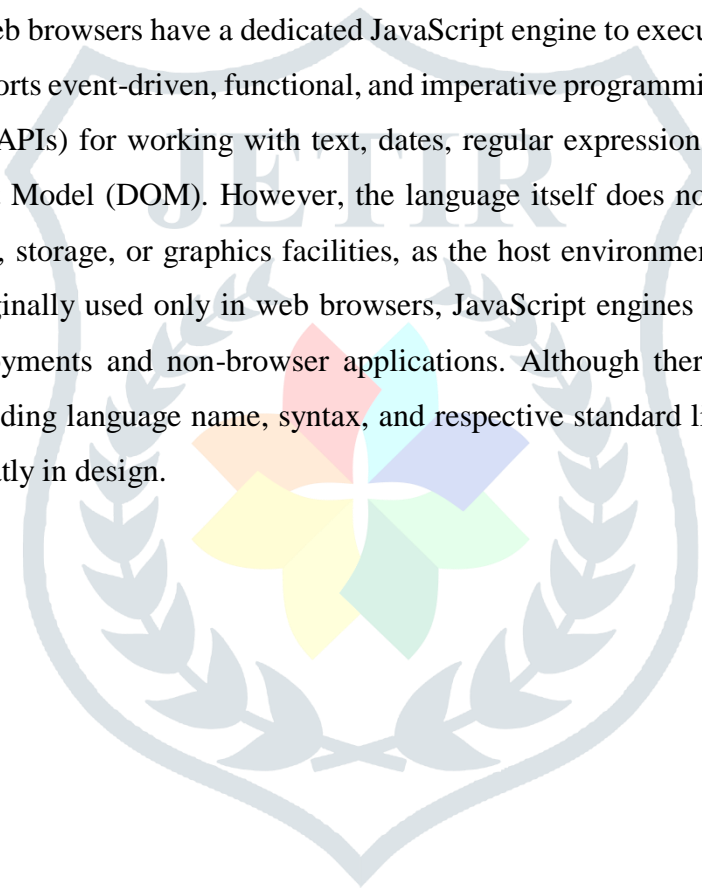
JavaScript ignores spaces, tabs, and newlines that appear in JavaScript programs. You can use spaces, tabs, and newlines freely in your program and you are free to format and indent your programs in a neat and consistent way that makes the code easy to read and understand. Simple statements in JavaScript are generally followed by a semicolon character, just as they are in C, C++, and Java. JavaScript, however, allows you to omit this semicolon if each of your statements are placed on a separate line. For example, the following code could be written without semicolons.

JavaScript is a case-sensitive language. This means that the language keywords, variables, function names,

and any other identifiers must always be typed with a consistent capitalization of letters. So the identifiers Time and TIME will convey different meanings in JavaScript.

All the modern browsers come with built-in support for JavaScript. Frequently, you may need to enable or disable this support manually. This chapter explains the procedure of enabling and disabling JavaScript support in your browsers: Internet Explorer, Firefox, chrome, and Opera.

JavaScript often abbreviated as JS, is an interpreted programming language that conforms to the ECMAScript specification. JavaScript is high-level, often just-in-time compiled, and multi-paradigm. It has curly-bracket syntax, dynamic typing, prototype-based object-orientation, and first-class functions. Alongside HTML and CSS, JavaScript is one of the core technologies of the World Wide Web. JavaScript enables interactive web pages and is an essential part of web applications. The vast majority of websites use it for client-side page behavior, and all major web browsers have a dedicated JavaScript engine to execute it. As a multi-paradigm language, JavaScript supports event-driven, functional, and imperative programming styles. It has application programming interfaces (APIs) for working with text, dates, regular expressions, standard data structures, and the Document Object Model (DOM). However, the language itself does not include any input/output (I/O), such as networking, storage, or graphics facilities, as the host environment (usually a web browser) provides those APIs. Originally used only in web browsers, JavaScript engines are also now embedded in server-side website deployments and non-browser applications. Although there are similarities between JavaScript and Java, including language name, syntax, and respective standard libraries, the two languages are distinct and differ greatly in design.



CHAPTER 3

SYSTEM ANALYSIS

EXISTING SYSTEM

The system study phase studies the requirements of the existing system "**Cyber Voting System**" is one of the tough processes. In existing system voting is conducted in a particular date. At that date all the users are come to centre then vote. Administrator starts to count the vote. All the peoples can ask their Winners. Final winner can be decided by administrator. If any people who wants to participate not available on that place can't participate. There is need for more advertisement for publicity. More man power is needed. Its participant's area is not combatable for all users.

LIMITATION OF EXISTING SYSTEM

- The current voting process is time consuming and confusing.
- There are thousands of peoples who work over eight hours a day and raise a family and for them voting is not a foreseeable option. (i.e) Wastage of Man Power.
- The process can be confusing and the machines can be hard to read.
- Increasing the cost for printing and many other works.
- Paper Wastage.
- Wastage of time for each people.
- Takes high cost
- Administrator has to keep track of all processes
- As files are maintained manually, chances for errors are large
- Needs more manpower
- Voters should be on place

PROPOSED SYSTEM

The proposed system should overcome all the disadvantages of the existing system.

The existing system is not functioning well due to manual process. Thus the proposed system

should minimize the manual efforts. In such a way that it should support the user to participate in the voting independent of the place. And also the administrator has to keep track of all the participants who are participated in program. Time consumption for arrangement will be minimum. Booth Agent admin register Users profile for voting before they participate and also user should provide some identification information like Aadhaar card and voter id information. It saves manual effort and time and provides remote database storage and retrieval.

ADVANTAGES OF THE PROPOSED SYSTEM

- The Internet is changing the way that people shop, communicate, and carry out business. It also has the potential to change the way that Cyber vote.
- Internet voting has the potential to increase voter turnout and it is cheaper. Having the Internet be the means of voting, elections would be cheaper and the results would be easier to tabulate.
- Internet voting is less wasteful of paper.
- It cuts down on processing costs. It also eliminates the amount of time staff spends tabulating votes, plus there are no longer the possibility of hanging chats.
- Another aspect of Internet voting is that it would be easier than the current process.
- Reducing the time for vote.
- Cyber Voting delivers the results within seconds of the closing date.

CHAPTER 4

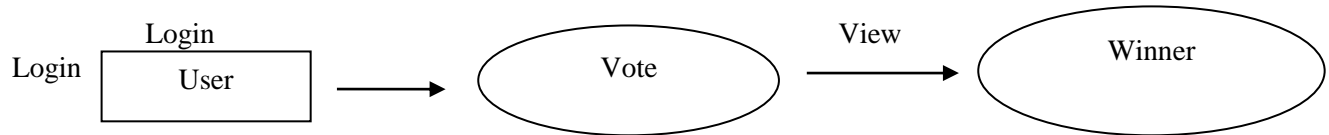
SYSTEMDESIGN

DATA FLOWDIAGRAM

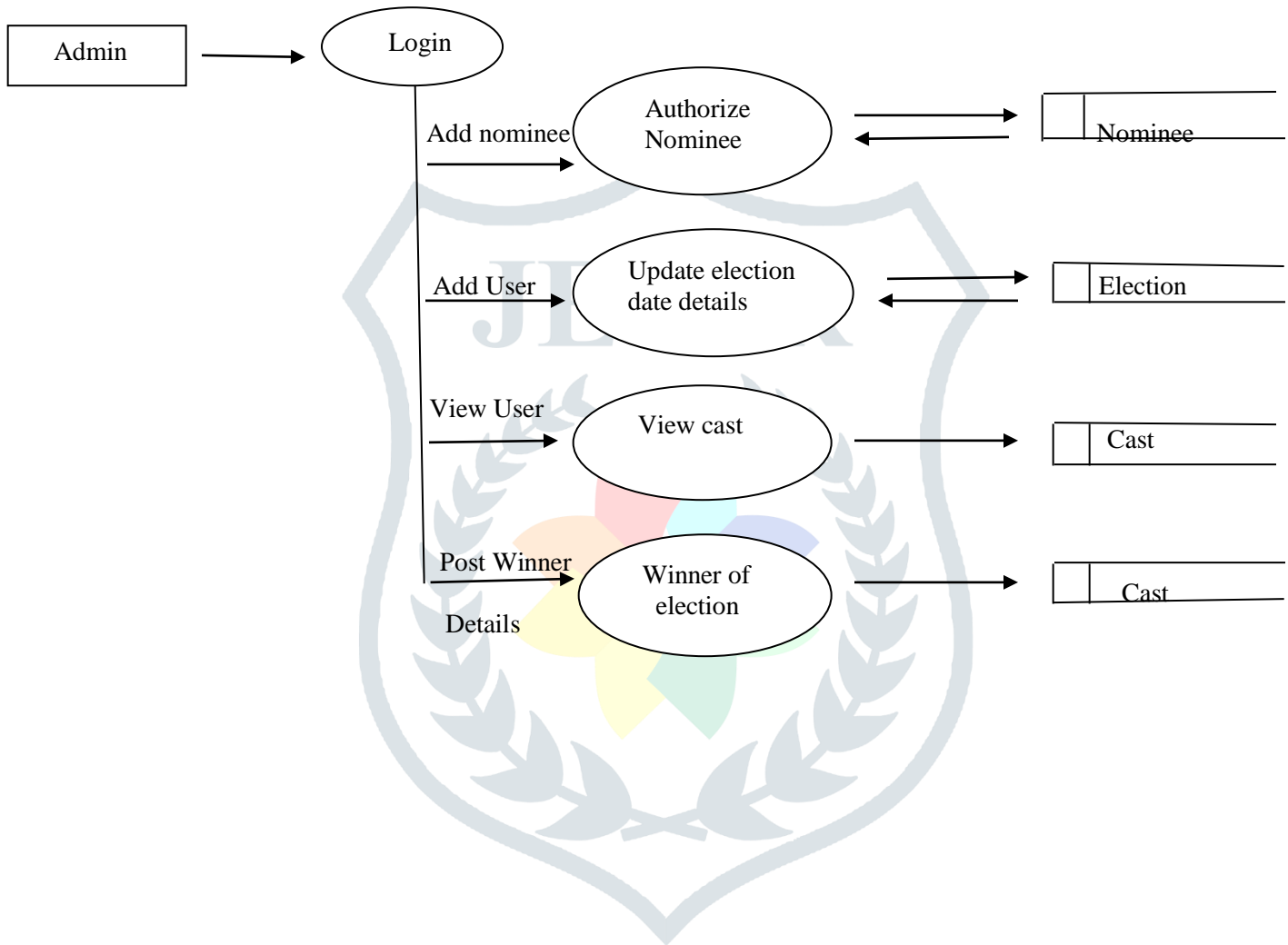
Data Flow Diagram – Finger Print Based E-Voting System

A data flow diagram is a graphical view of how data is processed in a system in terms of input and output. The Data flow diagram (DFD) contains some symbol for drawing the data flow diagram.

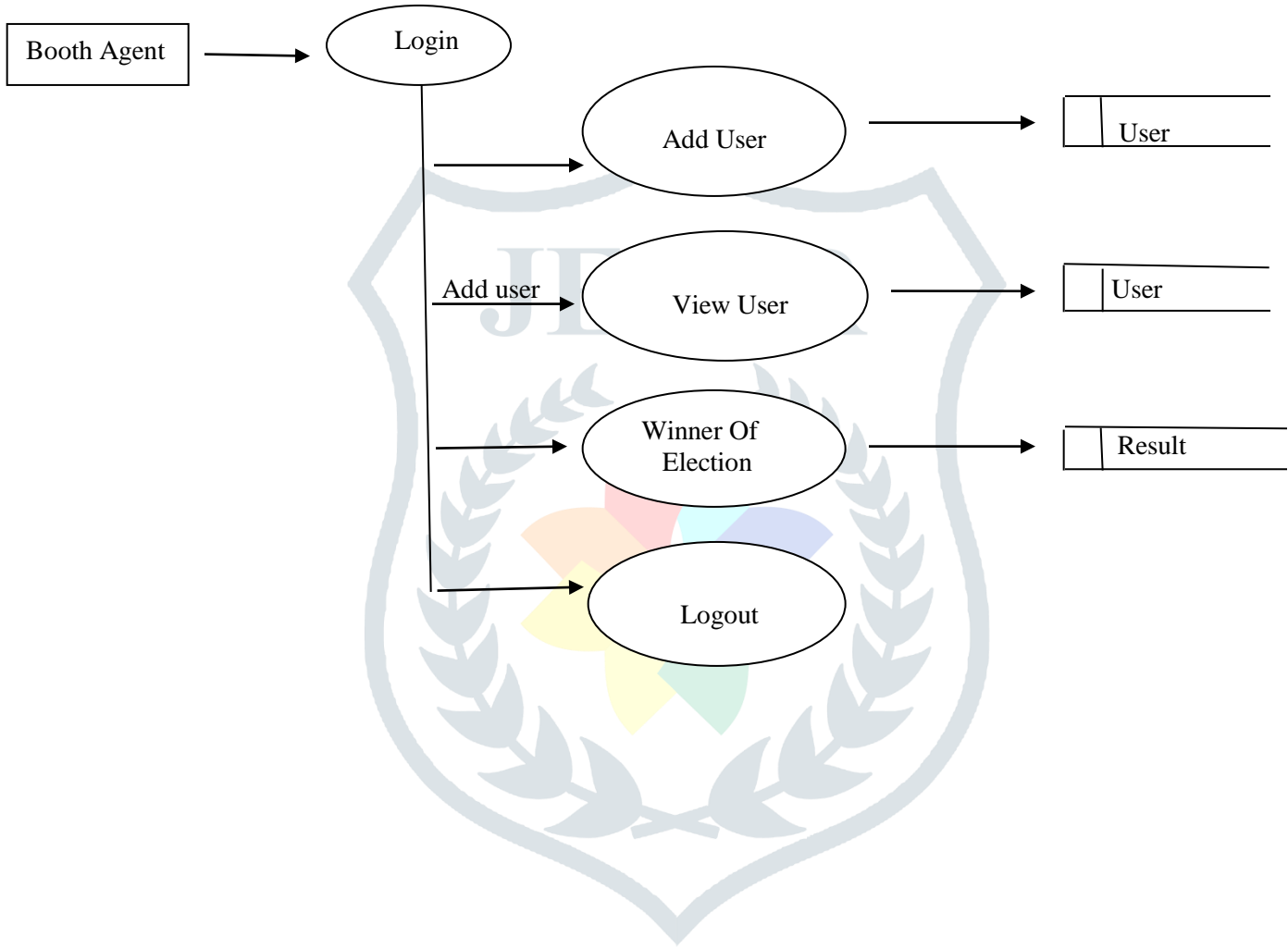
LEVEL 0



LEVEL 1



LEVEL 1



LEVEL 2

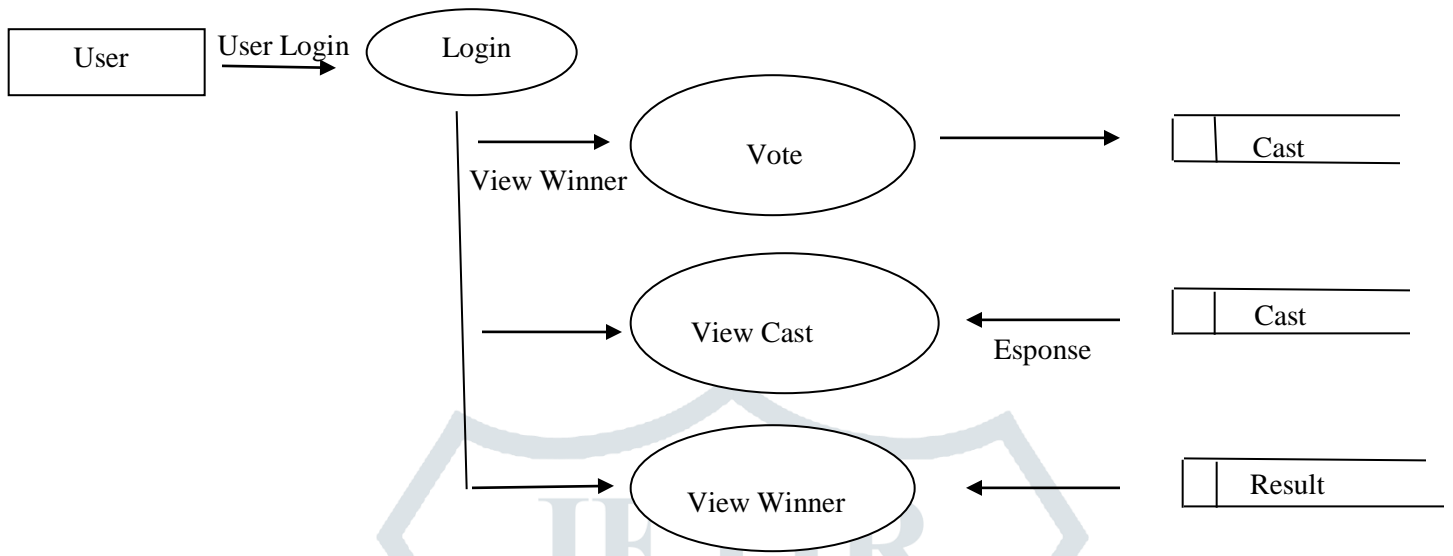


TABLE DESIGN

A table is made up of rows and columns. A row is also called a record (or tuple). Database design is a collection of interactive data store. An effective method of defining, store and retrieving the information in the database, multiple application and user can use the data contained in the database.

Table Name: Individuals

Description: To Add Individuals Data

Primary key : pid

USER TABLE

Primary key=pid

#	Name	Type	Null	Key
1	uid	int(10)	No	Primary key
2	uname	varchar(200)	No	
3	upass	varchar(200)	No	
4	uemail	varchar(200)	No	
5	uphone	varchar(15)	No	
6	fname	text	No	
7	age	text	No	
8	genger	text	No	
9	uaddr	text	No	
10	tfile	text	No	
11	assembly	text	No	
12	adharid	text	No	
13	voterid	text	No	
14	city	text	No	
15	state	text	No	
16	country	text	No	

VOTE TABLE

#	Name	Type	Null	Key
1	cid	int(5)	No	Primary_key
2	fname	text	No	
3	party	text	No	
4	assembly	text	No	
5	pid	text	No	
6	uname	text	No	

RESULT TABLE

#	Name	Type	Null	Key
1	Rsid	int(5)	No	Primary_key
2	Fname	text	No	
3	Party	text	No	
4	Assembly	text	No	

ADMIN TABLE

#	Name	Type	Null	Key
1	Aid	int(5)	No	Primary_key
2	Aname	varchar(20)	No	
3	apass	varchar(20)	No	
4	atype	text	No	

CANDIDATE TABLE

Primary key=pid

#	Name	Type	Null	Key
1	pid	int(5)	No	Primary key
2	fname	text	No	
3	party	text	No	
4	assembly	text	No	

5	age	text	No	
6	gender	text	No	
7	asset	text	No	
8	qualification	text	No	
9	city	text	No	
10	state	text	No	
11	country	text	No	
12	efile	text	No	
13	pfile	text	No	

CHAPTER 5 SYSTEM DESCRIPTION

SYSTEM DESCRIPTION

Maintenance is actually implementation of the review plan as important as it is programmers and analyst is to perform or identify with him or herself with the maintenance. There are psychologically personality, and professional reasons for this. Analyst and programmers spend fair more time maintaining programmer then they do writing them Maintenances account for 50-80% of total system development. Maintenance is expensive .One way to reduce the maintenance costs are through maintenance mgt and software modification audits Types of maintenance are

1. Perfective maintenance
2. Preventive maintenance

Perfective maintenance:

Changes made to the system to add features or to improve the performance.

preventive maintenance:

Changes made to the system to avoid future problems. Any changes can be made in the future and our project can adopt the changes.

MODULE DESCRIPTION

SUPER ADMIN MODULE

Super Admin Login

Super Admin logs into the system by specifying unique username and password.

Programming Details (Vote Description)

In this module administrator can conduct a voting on what date, time details should be entered. administrator can also edit and delete options for voting date, times. after the schedule time of voting the forms are blocked because date has been expired.

Participants details.

In this module administrator can enter a participant details. by checking a assets, age, qualifications. that is approval given by administrator.

Decrypt Vote

In this module admin decrypts the user casted encrypted votes by private encryption key. Now admin can view the vote details in decrypted text.

Winners details

In this module administrator can enter a winners details. Like how many votes should be casted, which participant can get highest vote, lowest vote details.

BOOTH AGENT MODULE

Booth Agent Admin Login

Booth Agent Admin logs into the system by specifying unique username and password.

New user Registration

In this module after login Booth Agent Admin add new user registration details like name, father name, age, gender, address, photo, aadhaar id, voter id, finger print image, city, state, country, assembly etc.

View User details

In this module Booth Agent Admin can view the user details name, father name, age, gender, address, photo, email id, aadhaar id, voter id, city, state, country, assembly etc.

USER LOGIN MODULE

User Login

User can login into the system by inputting finger print image. If user does not having login credentials, booth agent admin register by giving their basic details and get login credentials.

Uservote

In this module a user can view the participant details using the details user can put a vote using own finger image with separate login id. user can put a vote at once duplicate votes cannot be casted.

Encrypt Vote

In this module user casted votes will be encrypted by random private encryption key. If a hacker gets the vote details only encrypted text will be shown. These vote details gets stored in database.

View winners

In this module user can view the winners details. administrator can post a winners list. user can view a list of winners

INPUT DESIGN

Input design is the process of converting the user-oriented. Input to a computer based format. The goal of the input design is to make the data entry easier , logical and free error. Errors in the input data are controlled by the input design. The quality of the input determines the quality of the system output.

All the data entry screen are interactive in nature, so that the user can directly enter into data according to the prompted messages. The user are also can directly enter into data according to the prompted messages. The users are also provided with option of selecting an appropriate input from a list of values. This will reduce the number of error, which are otherwise likely to arise if they were to be entered by the user itself.

Input design is one of the most important phase of the system design. Input design is the process where the input received in the system are planned and designed, so as to get necessary information from the user, eliminating the information that is not required. The aim of the input design is to ensure the maximum possible levels of accuracy and also ensures that the input is accessible that understood by the user. The input design is the part of overall system design, which requires very careful attention. If the data going into the system is incorrect then the processing and output will magnify the errors.

The objectives considered during input design are:

- Nature of input processing.
- Flexibility and thoroughness of validation rules.
- Handling of properties within the input documents.
- Screen design to ensure accuracy and efficiency of the input relationship with files.

- Careful design of the input also involves attention to error handling, controls, batching and validation procedures.

Input design features can ensure the reliability of the system and produce result from accurate data or they can result in the production of erroneous information.

OUTPUT DESIGN

The output form of the system is either by screen or by hard copies. Output design aims at communicating the results of the processing of the users. The reports are generated to suit the needs of the users. The reports have to be generated with appropriate levels. In our project outputs are generated by asp as html pages. As its web application output is designed in a very user-friendly this will be through screen most of the time.

The logo is a shield-shaped emblem with a decorative border. Inside the shield, the word "JETIR" is written in a large, bold, serif font. Below the name, there is a circular arrangement of colorful geometric shapes (triangles and squares) in shades of red, blue, green, and yellow, surrounded by a laurel wreath.

JETIR

CHAPTER 6

CONCLUSION

E-voting is a potential solution to the lack of interest in voting amongst the young tech savvy population. For e-voting to become more open, transparent, and independently auditable, a potential solution would be base it on blockchain technology. This paper explores the potential of the blockchain technology and its usefulness in the e-voting scheme. The proposed an e-voting scheme, which is then implemented. The implementation and related performance measurements are given along with the challenges presented by the blockchain platform to develop a complex application like e-voting. The project highlights some shortcomings and presents two potential paths forward to improve the underlying platform (blockchain technology) to support e-voting and other similar applications. Blockchain technology has a lot of promise; however, in its current state it might not reach its full potential. There needs to be concerted effort in the core blockchain technology research to improve is features and support for complex applications that can execute within the blockchain network.

CHAPTER 7

FUTURE ENHANCEMENT

One greater advantage of hosting web is to make the resource to know all and to become familiar which paves a way for development of organization as well as for profit. While hosting the software it should be attractive, user-friendly, and easier to access. This cyber voting software constitutes all this characteristics. Authentication on administrator side can be moved over to biometrics for more secure access. Database used has limited storage, Which can be switched to SQL etc. It can be enhanced according to the client user friendliness. Can be enhanced with blogs in future for tie up with different organization



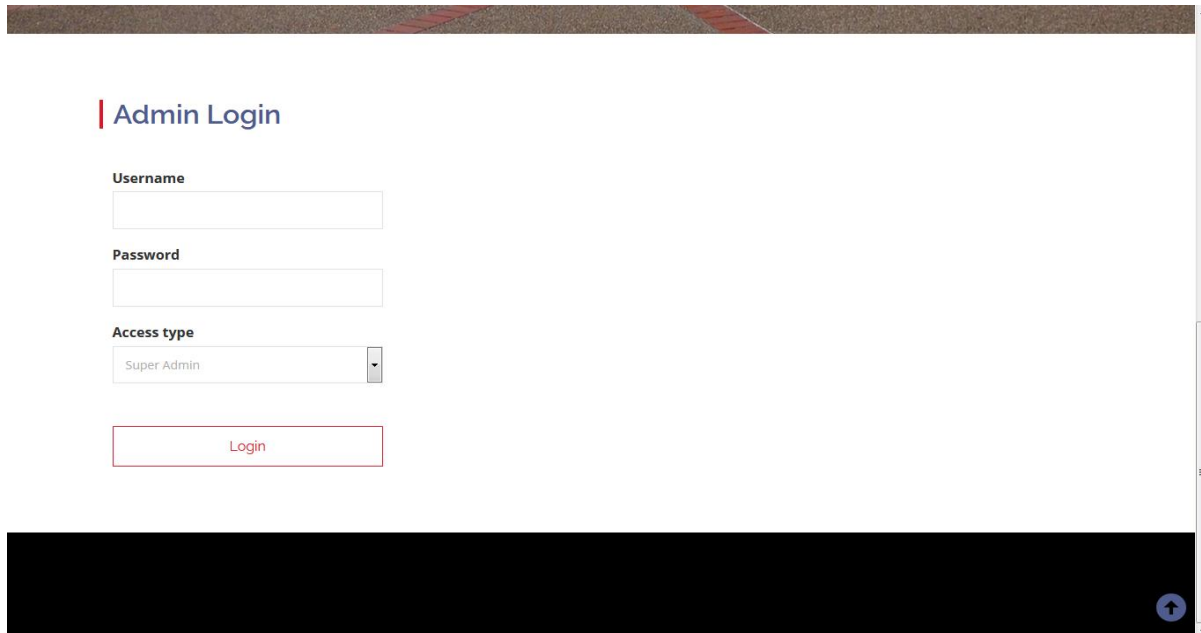
JETIR BIBLIOGRAPHY

1. [PHP Bible, 2nd Edition](#)
2. [PHP and MySQL Web Development \(4th Edition\)](#)
3. [PHP 6 and MySQL 5 for Dynamic Web Sites: Visual QuickPro Guide](#)
4. [Beginning PHP and MySQL: From Novice to Professional, Third Edition](#)
5. [Practical Web 2.0 Applications with PHP](#)
6. <http://php.net/manual/en/tutorial.php>
7. <https://www.guru99.com/php-tutorials.html>
8. <https://www.w3schools.com/php/default.asp>
9. <https://www.homeandlearn.co.uk/php/php.html>
10. <https://teamtreehouse.com/tracks/beginning-php>

APPENDIX

A.FORM DESIGN

ADMIN LOGIN

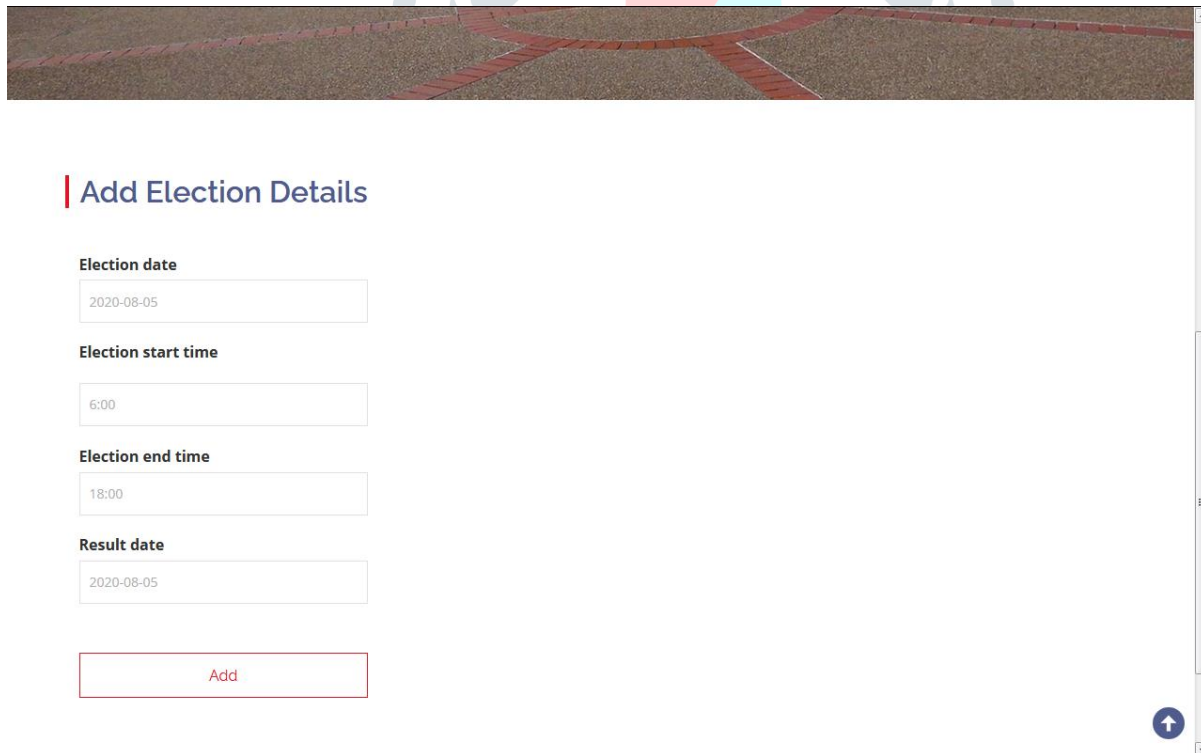


The screenshot shows a web browser window with a header image of a brick path. Below the header, the page title is "Admin Login". The form contains the following fields:

- Username:** A text input field.
- Password:** A text input field.
- Access type:** A dropdown menu with "Super Admin" selected.
- Login:** A red button.

A dark footer bar at the bottom right contains a blue circular icon with an upward-pointing arrow.

ADD ELECTION DETAILS



The screenshot shows a web browser window with a header image of a brick path. Below the header, the page title is "Add Election Details". The form contains the following fields:

- Election date:** A text input field containing "2020-08-05".
- Election start time:** A text input field containing "6:00".
- Election end time:** A text input field containing "18:00".
- Result date:** A text input field containing "2020-08-05".
- Add:** A red button.

A dark footer bar at the bottom right contains a blue circular icon with an upward-pointing arrow.

ADD PARTICIPANT DETAILS

Add Participants Details

Name

Age

Asset

Qualification

Gender

Emblem:

 No file selected.



Profile image:

 No file selected.

Party

VIEW PARTICIPANT

View Participants Details

Name	Age	Asset	Qualification	Emblem	Image	Party	Assembly	City	State	Country
SHANMUGAM	55	8 crore	8			ADMK	Kinathukadavu	Coimbatore	Tamilnadu	India
PRABHAKARAN	50	3 crore	12			DMK	Kinathukadavu	Coimbatore	Tamilnadu	India
Jayaraman	65	20crores	phd			ADMK	pollachi	coimbatore	tamilnadu	India
TAMILMANI	50	9 crore	B.E			DMK	pollachi	coimbatore	tamilnadu	India
Sellur Raju	50	9 crore	B.E			ADMK	madurai	madurai	tamilnadu	India

VIEW CAST



View Cast Details

Name	Party	Assembly	User
PRABHAKARAN	DMK	Kinathukadavu	ram
SHANMUGAM	ADMK	Kinathukadavu	vani
TAMILMANI	DMK	pollachi	vibin
PRABHAKARAN	DMK	Kinathukadavu	kala
Jayaraman	ADMK	pollachi	jaya
Jayaraman	ADMK	pollachi	prasanna
Alagiri	DMK	madurai	kamal
Alagiri	DMK	madurai	giri

VIEW RESULT



View Result

Name	Party	Assembly	Vote Count
Alagiri	DMK	madurai	2
Jayaraman	ADMK	pollachi	2
TAMILMANI	DMK	pollachi	1
SHANMUGAM	ADMK	Kinathukadavu	1
PRABHAKARAN	DMK	Kinathukadavu	3



ELECTION RESULT



Election Result

Name	Party	Assembly
PRABHAKARAN	DMK	Kinathukadavu
Jayaraman	ADMK	pollachi
Alagiri	DMK	madurai



VIEW WINNER




View Election Won Results

Party	Seats won	Total Seats
DMK	2	3
ADMK	1	3



BOOTH AGENT LOGIN




Admin Login

Username

Password

Access type
Booth Admin



BOOTH AGENT USER REGISTER

User Registration

Name

Age


Gender
Select

Email:

Phone:






Address:

Aadhar Id:



VIEW USERS

View Voting User Details

Name	Age	Gender	Email	Phone	Address	Profile Image	Aadhaar number	Voter Id number	Assembly	City	State
ram	18	male	ram@gmail.com	9435345345	coimbatore		98675463345634	543455345345345	kinathukadavu	coimbatore	Tamilnadu
vani	23	female	vani@gmail.com	8797897976	coimbatore		3556457868734	45675678575693456	kinathukadavu	coimbatore	Tamilnadu
vibin	19	male	vibin@gmail.com	9873453224	coimbatore		2311436657	72334266253	pollachi	coimbatore	Tamilnadu
jaya	25	female	jaya@gmail.com	9873453224	coimbatore		72334266253	2311436657	pollachi	coimbatore	Tamilnadu
giri	30	male	giri@gmail.com	9877343434	madurai		43366234224473367	98675463345634	madurai	madurai	Tamilnadu

LOGIN



Voter Login

Finger Print Image:
 No file selected.

CAST VOTE

Cast Vote

Name	Age	Emblem	Image	Party	Assembly	City	Action
SHANMUGAM	55			ADMK	Kinathukadavu	Coimbatore	<input type="radio"/>
PRABHAKARAN	50			DMK	Kinathukadavu	Coimbatore	<input checked="" type="radio"/>

Cast



B.SOURCE CODE**User register**

```

<?php include "header.php"; ?>
<div class="contact">
    <div class="container">
        <h3>User Registration</h3>
        <div class="contact-grid">
            <div class="col-md-7 contact-right">
                <form action="" method="post" enctype="multipart/form-data">
                    <label>Name</label><input type="text" name='fname'
required="">
                    <label>Age</label><input type="text" name='age'
required="">
                    <label>Gender</label>
                    <select name="gender" required="">
                        <option value="">Select</option>
                        <option value="male">Male</option>
                        <option value="female">Female</option>
                    </select>
                    <label>Email:</label>
                    <input type="email" value="" name="uemail" required>
                    <label>Phone:</label>
                    <input type="text" value="" name="uphone" required>
                    <label>Address:</label>
                    <input type="text" value="" name="uaddr" required>
                    <label>Aadhar Id:</label>
                    <input type="text" value="" name="adharid" required>
                    <label>Voter Id:</label>
                    <input type="text" value="" name="voterid" required>
                    <label>Profile:</label>
                    <input type="file" value="" name="tfile" required><br>
                    <label>Assembly</label><input type="text" name='assembly' required="">
                    <label>City</label><input type="text" name='city'

```

```
required="">
```

```
<label>State</label><input type="text" name='state'
```

```
required="">
```

```
<label>Country</label><input type="text" name='country'
```

```
required="">
```

```
<label>Username</label><input type="text" name='uname'
```

```
required="">
```

```
<label>Password</label><input type="password"
name='upass' required="">
```

```
<input type="submit" value="Register" name="submit">
```

```
</form>
```

```
</div>
```

```
<div class="clearfix"> </div>
```

```
</div>
```

```
</div>
```

```
</div>
```

```
<?php include "footer.php"; ?>
```

```
<?php
```

```
if(isset($_POST['submit']))
```

```
{
```

```
$fname=$_POST['fname'];
```

```
$uname=$_POST['uname'];
```

```
$upass=$_POST['upass'];
```

```
$uemail=$_POST['uemail'];
```

```
$uphone=$_POST['uphone'];
```

```
$uaddr=$_POST['uaddr'];
```

```
$age=$_POST['age'];
```

```
$gender=$_POST['gender'];
```

```
$city=$_POST['city'];
```

```
$state=$_POST['state'];
```

```
$adharid=$_POST['adharid'];
```

```
$voterid=$_POST['voterid'];
```

```
$party=$_POST['party'];
```

```
$assembly=$_POST['assembly'];
```

```
$q=mysql_query("select * from user where uname='$uname' and upass='$upass'");
```

```
$c=mysql_num_rows($q);
```

```
if($c>0)
```

```
{
while($r=mysql_fetch_array($q))
{
echo "<script type='text/javascript'>alert('Voter account already exists');</script>";
echo '<meta http-equiv="refresh" content="0;url=index.php"/>';
}
}
else
```

```
{
```

```
$tfile=$_FILES['tfile']['name'];
```

```
move_uploaded_file($_FILES['tfile']['tmp_name'], "upload/$tfile");
```

```
mysql_query("insert into
```

```
`user` (uname,upass,uemail,uphone,uaddr,fname,age,gender,tfile,city,state,country,assembly,adharid,voterid
```

```
)values('$uname','$upass','$uemail','$uphone','$uaddr','$fname','$age','$gender','$tfile','$city','$state','$countr
```

```
y','$assembly','$adharid','$voterid')")or die(mysql_error());
```

```
echo "<script type='text/javascript'>alert('Voter Profile registered successfully');</script>";
```

```
echo '<meta http-equiv="refresh" content="0;url=login.php"/>';
}
}
?>
```

```
Cast vote
```

```
<?php include "header.php"; ?>
```

```
</?php include "header.php"; ?>
```

```
<?php include "header.php"; ?>
```

```
<?php include "header.php"; ?>
```

```
<?php include "header.php"; ?>
```

```
<?php include "header.php"; ?>
```

```

<div class="contact">
    <div class="container">
        <h3>Cast Vote </h3>

        <div class="contact-grid">

            <div class="col-md-12 contact-right">
                <form action="" name="" method="post">
                    <table class="table table-bordered">
                        <thead>
                            <tr>
                                <th>Name</th>
                                <th>Age</th>
                                <th>Emblem</th>
                                <th>Image</th>
                                <th>Party</th>
                                <th>Assembly</th>
                                <th>City</th>
                                <th>Action</th>
                            </tr>
                        </thead>
                        <tbody>
                            <?php

                                $assembly=$_SESSION['assembly'];

                                $city=$_SESSION['city'];
                                $state=$_SESSION['state'];
                                $country=$_SESSION['country'];

                                $q=mysql_query("select * from participants where
                                assembly='$assembly' and city='$city' and state='$state' and country='$country'");
                                while($r=mysql_fetch_array($q))
                                {
                                    $fname=$r['fname'];

```

```
$age=${r['age'];
```

```
$asset=${r['asset'];
```

```
$qualification=${r['qualification'];
```

```
$city=${r['city'];
```

```
$state=${r['state'];
```

```
$country=${r['country'];
```

```
$party=${r['party'];
```

```
$assembly=${r['assembly'];
```

```
$efile=${r['efile'];
```

```
$pfile=${r['pfile'];
```

```
$pid=${r['pid'];
```

```
echo "<tr>
```

```
<td>${fname}</td>
```

```
<td>${age}</td>
```

```
<td><img src='upload/${efile}' height='100px' /></td>
```

```
<td><img src='upload/${pfile}' height='100px' /></td>
```

```
<td>${party}</td>
```

```
<td>${assembly}</td>
```

```
<td>${city}</td>
```

```
<td><input type='radio' name='pid' value='${pid}'
```

```
/></td>
```

```
</tr>";
```

```
}
```

```
?>
```

```
</tbody>
```

```
</table>
```

```
<input type="submit" name="submit" value="Cast" />
```

```
</form>
```

```
</div>
```

```
<div class="clearfix"> </div>
```

```
</div>
```


</div>

</div>

<?php

if(isset(\$_POST['submit']))

{

\$uname=base64_encode(\$_SESSION['uname']);

\$uid=\$_SESSION['uid'];

\$uassembly=base64_encode(\$_SESSION['assembly']);

\$city=base64_encode(\$_SESSION['city']);

\$pid=\$_POST['pid'];

\$sepid=md5(\$_POST['pid']);

\$q=mysql_query("select * from cast where uname='\$uname' and assembly='\$uassembly'");

\$c=mysql_num_rows(\$q);

if(\$c>0)

{

echo "<script type='text/javascript'>alert('Already vote has been casted');</script>";

echo '<meta http-equiv="refresh" content="0;url=logout.php"/>';

}

else

{

\$q=mysql_query("select * from participants where pid='\$pid'");

while(\$r=mysql_fetch_array(\$q))

{

\$fname1=\$r['fname'];

\$party1=\$r['party'];

\$assembly1=\$r['assembly'];

\$uphone=\$_SESSION['uphone'];

\$fname=base64_encode(\$r['fname']);

\$party=base64_encode(\$r['party']);

\$assembly=base64_encode(\$r['assembly']);

```
mysql_query("insert into
`cast`(fname,party,assembly,pid,uname)values('$fname','$party','$assembly','$pid','$uname')");
echo "<script type='text/javascript'>alert('Vote has been casted successfully');</script>";
$to="$suphone";
$msg="You have voted for $fname1-$party1-$assembly1";
echo "<meta http-equiv='refresh'
content='0;url=http://websms.jananatech.com/pushsms.php?username=kovaigrocery&password=DQsJMh
&sender=OMSAAI&message=$msg&numbers=$to'/>";
}
}
}
?>

<?php include "footer.php"; ?>
```

