PROJECT REPORT ON

"SMART CALCULATOR"

Bachelor of Computer Science Engineering

and Applications

(Batch: 2021-2025)

Submitted To:

Ms. Ankit Arora

(Training And Development)

Submitted By:

Aditya Chauhan(2115000055)

GLA UNIVERSITY, MATHURA (Affiliated to NAAC A+ Grade)

CERTIFICATE

This is to certify that I Aditya Chauhan of BTech (CSE) 5th Semester from GLA University, Mathura has presented this Mini project work entitled "SMART CALCULATOR", a website in partial fulfilment of the requirements for the award of the degree of Bachelor of Technology under our supervision and guidance.

ACKNOWLEDGEMENT

It is our proud privilege to express our profound gratitude to the entire management of GLA University and the teachers of the institute for providing us with the opportunity to avail ourselves of the excellent facilities and infrastructure. The knowledge and values inculcated have proved to be of immense help at the very start of my career. Special thanks to the Hon'ble Founder, GLA University, Mathura for having provided us with an excellent infrastructure.

I am grateful to Ankit Arora for their astute guidance, constant encouragement and sincere support for this project work.

Sincere thanks to all my family members, seniors and friends for their support and assistance throughout the project.

INDEX

INTRODUCTION

Existing Calculator Website:

- Large space and manpower is required to setup voting booths in multiple areas around city or village.
- High security has to be maintained on the date of election.
- Voter must visit the place where voting booth is arranged.
- Sometimes, voter needs to stand in a queue for a long time.
- Again, manpower is required for volunteering and assistant of voters at the place of voting.
- Voting process is done manually on voting machine.
- Vote counting is done with the manual process.
- Then there is a gap of few days for results to be displayed.
- So if we see, here in traditional voting system, we need lot of manpower, energy, and time to conduct this process.

Proposed System

- Now as we all know, almost everything can be done online.
- Like Money transfer, Shopping, Booking, Teaching, Data sharing, Admissions, Job search, etc. And so many other activities are done with the help of internet.
- So with the easy access and use of internet, we are going to take this existing voting system on advance level.
- We are going to develop an online platform with high security so that the same process could be done easily without the waste of time, afford, and energy.
- So firstly, voters and groups/candidates are required to register on online voting system.
- Once registration is done, voter can easily vote to their respected candidate or group by just signing in with the comfort of his/her home.
- And similarly, groups/candidates can do the same as well as also monitor their status with the comfort of home.
- So this system will save a lot of time, energy, and afford for both voters and groups.

SYSTEM DESIGN

1. Voters

Voters are the people who will first sign up on online voting panel. And then at the time of voting, they will login and do vote to their respective group or candidate via system.

Following data from voter side will be provided to the system at the time of registration:

- Name
- Mobile
- Address
- Status
- Votes
- Role (voter/group)
- Photo
- Password

Voter responsibilities:

- Registration on system
- Login to system

- Voting for the candidate

2. System

System is an online platform where election process is held. So the voters and groups are registered here. And with the help of system, voters can do voting and groups can monitor their status.

System responsibilities:

- Registration of both voter and candidate
- Display of registered candidates with respective votes on homepage
- Display of registered candidates on voter dashboard if any
- Display of profile info and voting status on voter dashboard
- Display of profile info, voting status, and votes on candidate dashboard
- Maintaining record for each candidate and voter without making any duplicate record.

3. Groups

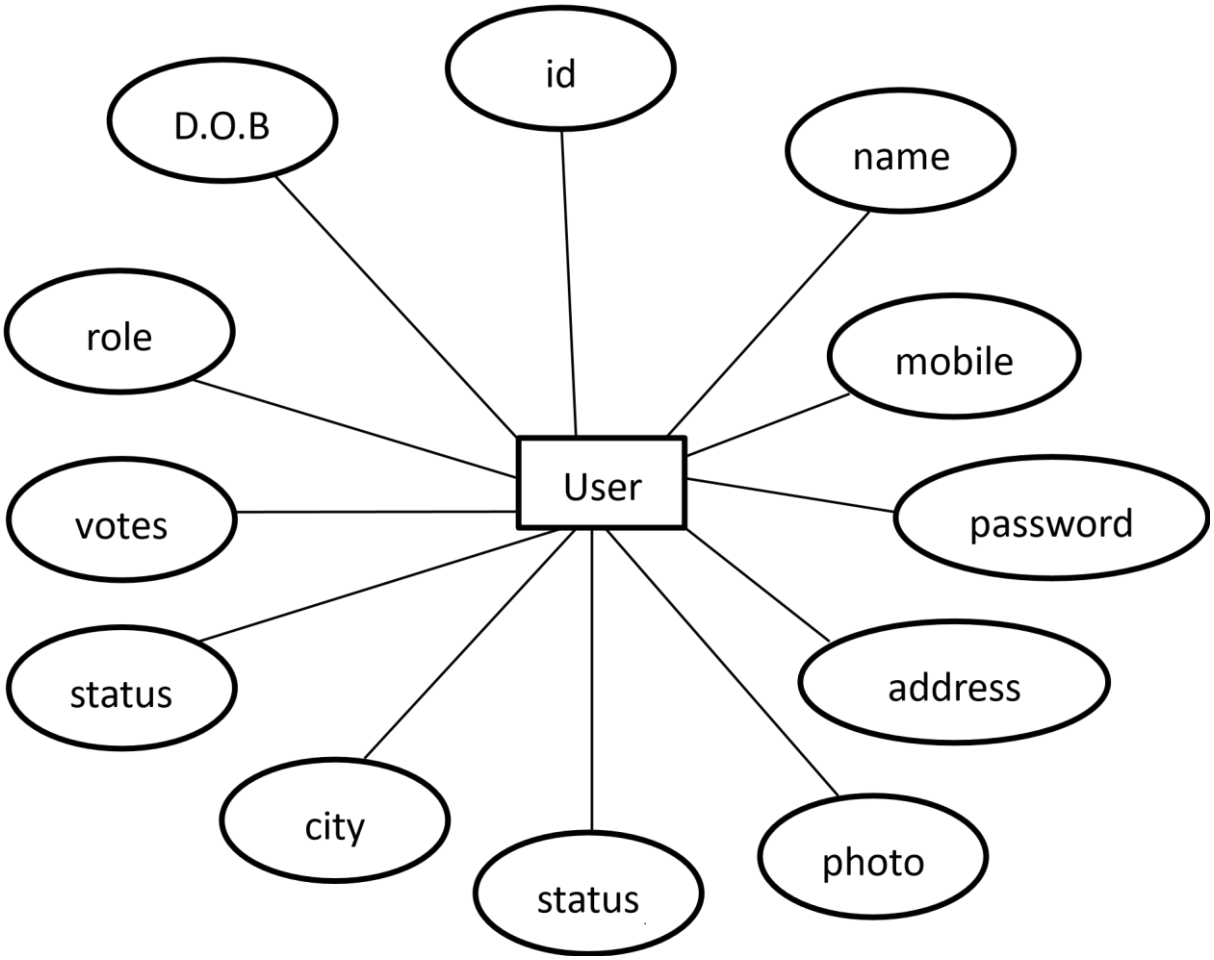
Groups/Candidates/Parties are those who will be given votes at the time of voting. And they can monitor their status by just doing login into system.

Following data from group/party side will be provided to the system at the time of registration:

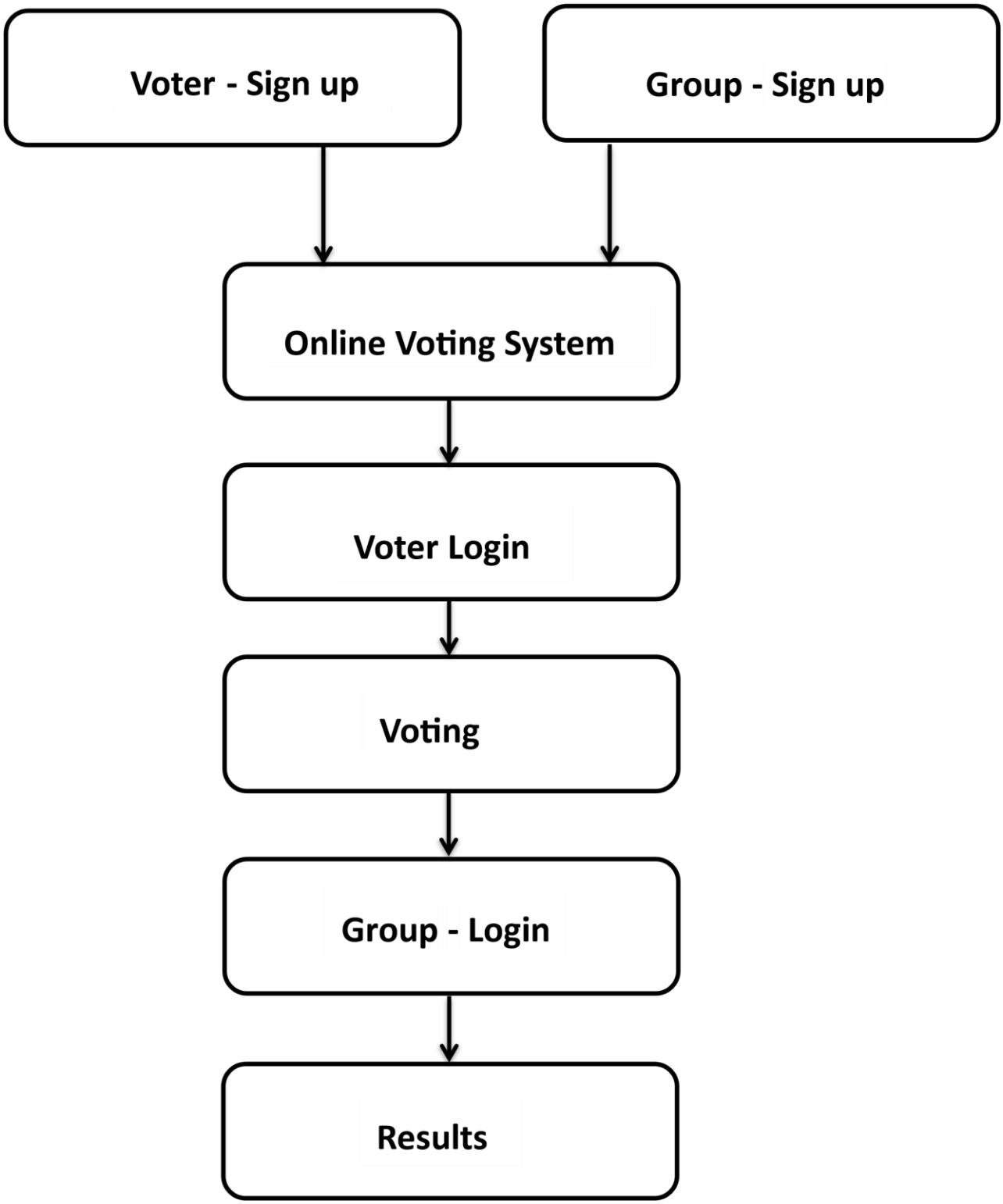
- Name
- Mobile

- Address
- Status
- Votes
- Role (voter/group)
- Photo
- Password

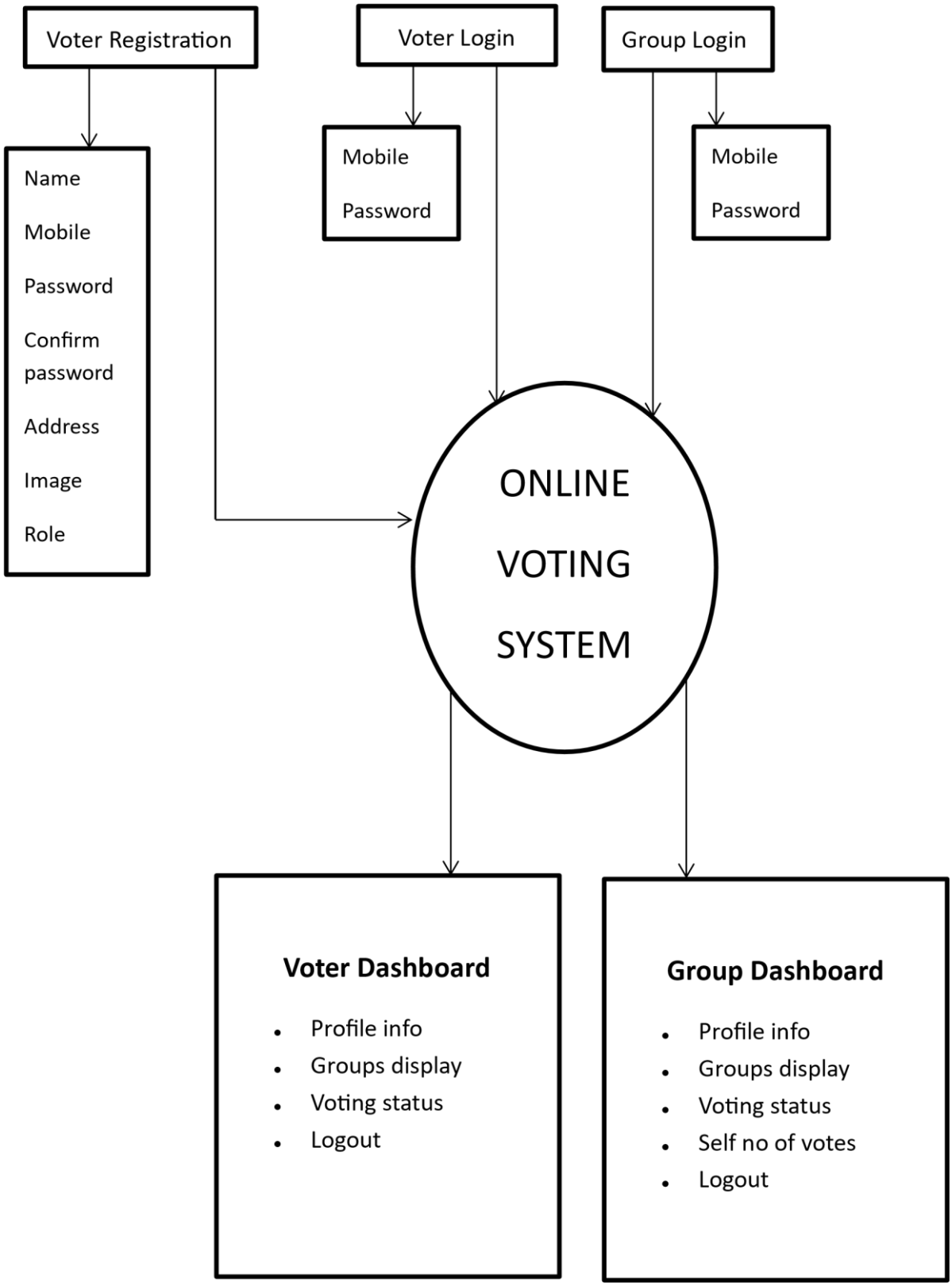
ER DIAGRAM


id
D.O.B
name
role
mobile
User
votes
password
status
address
city
photo
status

FLOW CHART



DFD (DATA FLOW DIAGRAM)



DATABASE DESIGN

Database design

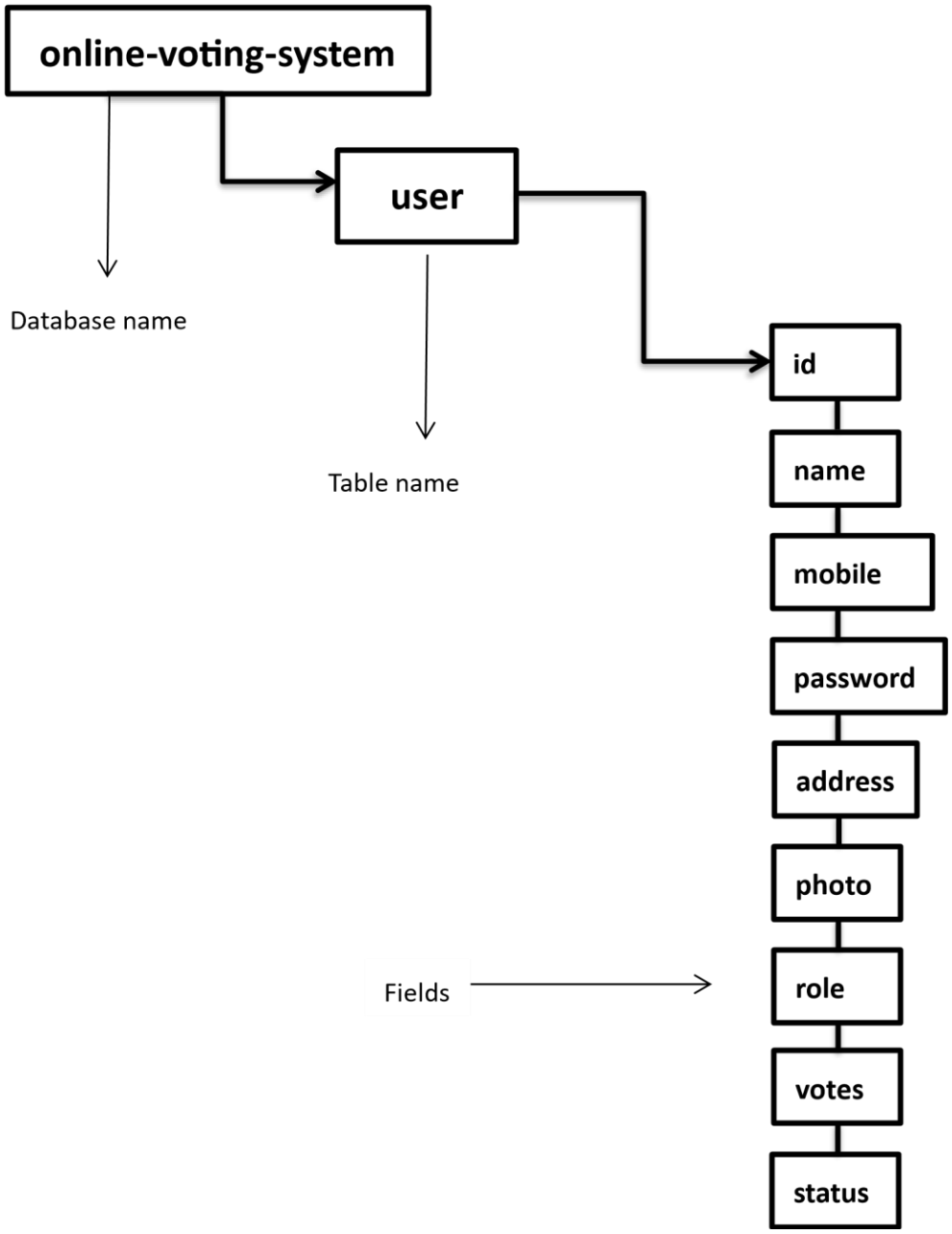
- MySQL is a technology which is used to maintain overall data of voters and candidates in this system.
- We created a database with name ***"online-voting-system"*** in MySQL.
- Then we created a table called ***user*** inside database

Following fields are created in *register* table:

- Name
- Mobile
- Address
- Status
- Votes
- Role (voter/group)
- Photo
- Password

DATABASE STRUCTURE

Database Structure



How to create database, table, and fields

- Open XAMPP application.
- Click on ***Start*** button right next to MySQL module.
- Click on ***Admin*** button next to ***Start*** button on MySQL module.
- You'll see ***phpmyadmin*** panel opened browser. There is a list of default databases on left hand side. So click on ***New***, give the name XYZ, and click on ***Create***.

- New database in created with the name of XYZ. Now inside XYZ database, there is an option ***New*** to create tables.
- Click on ***New*** and you'll see option to add table name on top and below it the names of fields.
- Now add fields inside table like name, email, mobile, etc.
- Once all this done, you have finished the process of creating database, tables, and fields.

How to run project

- Suppose project name is XYZ. So place the XYZ project folder in "*xampp/htdocs/"* location in your respective drive.
- Open XAMPP Control Panel and Start ***Apache*** and ***MySQL.***
- Open browser and type "*localhost/XYZ*". - You will see the output in browser.

APPLICATIONS

This system is applicable in below fields for election purpose:

- **School**
- **College**
- **Industry**
- **Corporate**
- **Government**
- **Hospital**

- **Food and Restaurant**
- **Tourism**
- **Sports**
- **Entertainment**
- **Production**
- **Investment**
- **News and Media**
- **Technology**

FUTURE ENHANCEMENTS

ADMIN ACCESS

- Admin functionality would be used to control overall election process. Admin, means election committee or election authority, could maintain overall election process.
- Like voter and candidate verification. Admin would verify and authenticate whether voter or candidate is qualified to take part or not. Admin could also start and stop election process.
- Admin could monitor voting process, like total number of votes given, total number of remaining votes, voting percentage, etc.
- Admin could get results in excel sheet once voting is completed.

EMAIL VERIFICATION

- Three times voter would have to go through email verification process.
- First, at the time of registration. Second, at the time of voting and third when someone has forgotten his or her password.
- Once verification is completed, he or she would be qualified for voter position.

TECHNOLOGY DETAILS

HTML

HTML stands for HyperText Markup Language. It is used to design web pages using a markup language. HTML is the combination of Hypertext and Markup language. Hypertext defines the link between the web pages. A markup language is used to define the text document within tag which defines the structure of web pages. This language is used to annotate (make notes for the computer) text so that a machine can understand it and manipulate text accordingly.

Most markup languages (e.g. HTML) are human-readable. The language uses tags to define what manipulation has to be done on the text. HTML is a markup language used by the browser to manipulate text, images, and other content, in order to display it in the required format. HTML was created by Tim Berners-Lee in 1991. The first-ever version of HTML was HTML 1.0, but the first standard version was HTML 2.0, published in 1999.

Elements and Tags: HTML uses predefined tags and elements which tell the browser how to properly display the content. Remember to include closing tags. If omitted, the browser applies the effect of the opening tag until the end of the page.

HTML page structure: The basic structure of an HTML page is laid out below. It contains the essential building-block elements (i.e. doctype declaration, HTML, head, title, and body elements) upon which all web pages are created.

<DOCTYPE! html>: This is the document type declaration (not technically a tag). It declares a document as being an HTML document. The doctype declaration is not case-sensitive.

<html>: This is called the HTML root element. All other elements are contained within it.

<head>: The head tag contains the "behind the scenes" elements for a webpage. Elements within the head aren't visible on the front-end of a webpage. HTML elements used inside the <head> element include:

- <style> · <title>
- <base>
- <noscript>
- <script>
- <meta>
- <link>

<body>: The body tag is used to enclose all the visible content of a webpage. In other words, the body content is what the browser will show on the front-end.

An HTML document can be created using any text editor. Save the text file using **.html** or **.htm**. Once saved as an HTML document, the file can be opened as a webpage in the browser.

Note: Basic/built-in text editors are Notepad (Windows) and TextEdit (Macs). Basic text editors are entirely sufficient for when you're just getting started. As you progress, there are many feature-rich text editors available which allow for greater function and flexibility.

Features of HTML:

- It is easy to learn and easy to use.
- It is platform-independent.
- Images, videos, and audio can be added to a web page.
- Hypertext can be added to the text.
- It is a markup language.

Why learn HTML?

- It is a simple markup language. Its implementation is easy.
- It is used to create a website.
- Helps in developing fundamentals about web programming.
- Boost professional career.

Advantages:

- HTML is used to build websites.
- It is supported by all browsers.
- It can be integrated with other languages like CSS, JavaScript, etc.

Disadvantages:

- HTML can only create static web pages. For dynamic web pages, other languages have to be used.
- A large amount of code has to be written to create a simple web page.
- The security feature is not good.

CSS

- Cascading Style Sheets (CSS) is a style sheet language used for describing the presentation of a document written in a markup

language such as HTML.[1] CSS is a cornerstone technology of the World Wide Web, alongside HTML and JavaScript.[2]

- CSS is designed to enable the separation of presentation and content, including layout, colors, and fonts.[3] 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 which reduces complexity and repetition in the structural content as well as enabling the .css file to be cached to improve the page load speed between the pages that share the file and its formatting.
- 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 Braillebased tactile devices. CSS also has rules for alternate formatting if the content is accessed on a mobile device.[4]
- 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) is registered for use with CSS by RFC 2318 (March 1998). The W3C operates a free CSS validation service for CSS documents.[5]

text/css

- In addition to HTML, other markup languages support the use of CSS including XHTML, plain XML, SVG, and XUL.

Advantages of CSS

- **CSS saves time** − You can write CSS once and then reuse same sheet in multiple HTML pages. You can define a style for each HTML element and apply it to as many Web pages as you want.

- **Pages load faster** − If you are using CSS, you do not need to write HTML tag attributes every time. Just write one CSS rule of a tag and apply it to all the occurrences of that tag. So less code means faster download times.
- **Easy maintenance** − To make a global change, simply change the style, and all elements in all the web pages will be updated automatically.
- **Superior styles to HTML** − CSS has a much wider array of attributes than HTML, so you can give a far better look to your HTML page in comparison to HTML attributes.
- **Multiple Device Compatibility** − Style sheets allow content to be optimized for more than one type of device. By using the same HTML document, different versions of a website can be presented for handheld devices such as PDAs and cell phones or for printing.
- **Global web standards** − Now HTML attributes are being deprecated and it is being recommended to use CSS. So its a good idea to start using CSS in all the HTML pages to make them compatible to future browsers.

Who Creates and Maintains CSS?

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. Independent companies and organizations create that software.

NOTE − The World Wide Web Consortium, or W3C is a group that makes recommendations about how the Internet works and how it should evolve.

JavaScript

- JavaScript (/ˈdʒɑːvəˌskrɪpt/),[6] often abbreviated as JS, is a programming language that conforms to the ECMAScript specification.[7] JavaScript is high-level, often just-in-time compiled, and multi-paradigm. It has curlybracket syntax, dynamic typing, prototype-based objectorientation, and firstclass functions.

- Alongside HTML and CSS, JavaScript is one of the core technologies of the World Wide Web.[8] JavaScript enables interactive web pages and is an essential part of web applications. The vast majority of websites use it for clientside page behavior,[9] and all major web browsers have a dedicated JavaScript engine to execute it.

- As a multi-paradigm language, JavaScript supports eventdriven, 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.

- JavaScript engines were originally used only in web browsers, but they are now embedded in some servers, usually via Node.js. They are also embedded in a variety of applications created with frameworks such as Electron and Cordova.

- 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.

Client-Side JavaScript

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.

Advantages of JavaScript

The merits of using JavaScript are −

- **Less server interaction** − You can validate user input before sending the page off to the server. This saves server traffic, which means less load on your server.
- **Immediate feedback to the visitors** − They don't have to wait for a page reload to see if they have forgotten to enter something.
- **Increased interactivity** − You can create interfaces that react when the user hovers over them with a mouse or activates them via the keyboard.
- **Richer interfaces** − You can use JavaScript to include such items as drag-and-drop components and sliders to give a Rich Interface to your site visitors.

Limitations of JavaScript

We cannot treat JavaScript as a full-fledged programming language. It lacks the following important features −

- Client-side JavaScript does not allow the reading or writing of files. This has been kept for security reason.
- JavaScript cannot be used for networking applications because there is no such support available.
- JavaScript doesn't have any multi-threading or multiprocessor capabilities.

Once again, JavaScript is a lightweight, interpreted programming language that allows you to build interactivity into otherwise static HTML pages.

JavaScript Development Tools

One of major strengths of JavaScript is that it does not require expensive development tools. You can start with a simple text editor such as Notepad. Since it is an interpreted language inside the context of a web browser, you don't even need to buy a compiler.

To make our life simpler, various vendors have come up with very nice JavaScript editing tools. Some of them are listed here −

- **Microsoft FrontPage** − Microsoft has developed a popular HTML editor called FrontPage. FrontPage also provides web developers with a number of JavaScript tools to assist in the creation of interactive websites.
- **Macromedia Dreamweaver MX** − Macromedia Dreamweaver MX is a very popular HTML and JavaScript editor in the professional web development crowd. It provides several handy prebuilt JavaScript components, integrates well with databases, and conforms to new standards such as XHTML and XML.
- **Macromedia HomeSite 5** − HomeSite 5 is a well-liked HTML and JavaScript editor from Macromedia that can be used to manage personal websites effectively.

Where is JavaScript Today ?

The ECMAScript Edition 5 standard will be the first update to be released in over four although both the languages still support the features that are not a part of the standard.years. JavaScript 2.0 conforms to Edition 5 of the ECMAScript standard, and the difference between the two is extremely minor.

The specification for JavaScript 2.0 can be found on the following site: http://www.ecmascript.org/

Today, Netscape's JavaScript and Microsoft's JScript conform to the ECMAScript standard.

HARDWARE & SOFTWARE REQUIREMENT SPECIFICATION

Hardware specifications

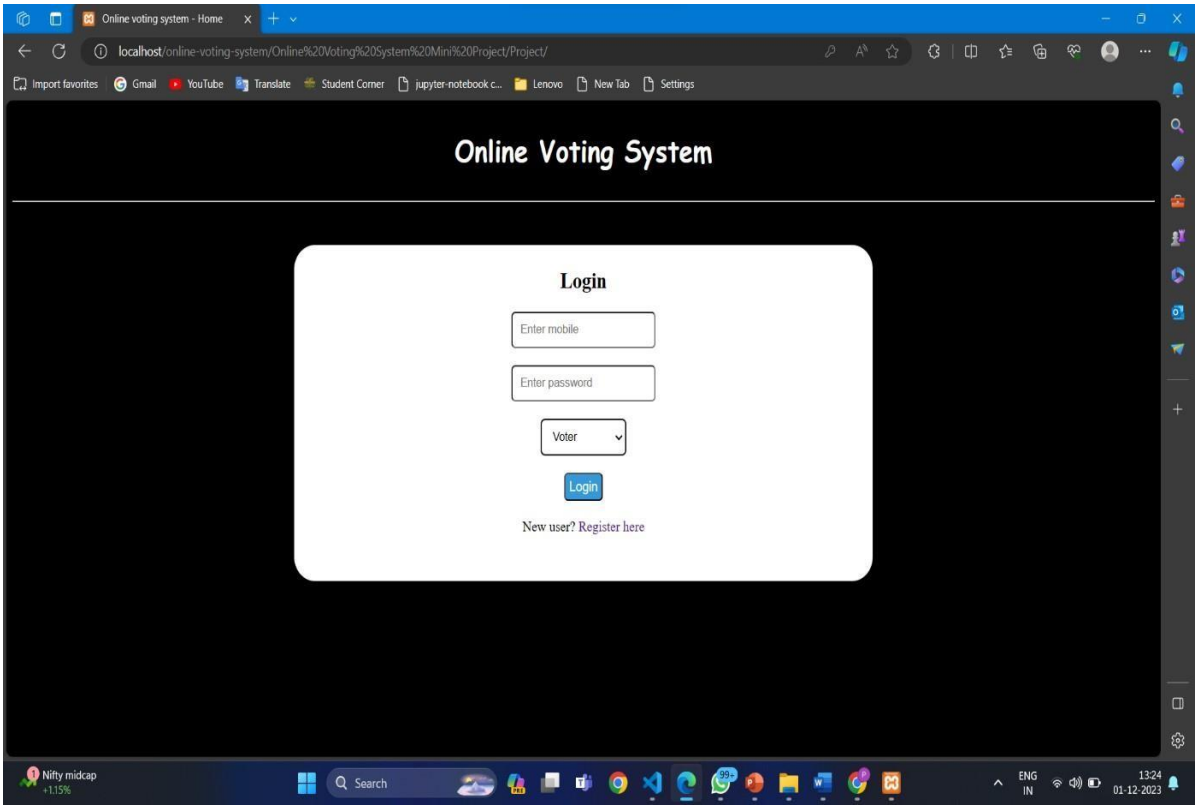
Operating system	**:**	Windows Server 2008 and later Windows Vista and later Mac OS X 10.6 and later CentOS, Ubuntu, Fedora, Gentoo, Arch, SUSE
Platform	:	IA-32 (Windows package only) and x64 (macOS and Linux packages only)
Size	:	Windows: 156 MB Linux: 150 MB macOS: 161 MB

Software specifications

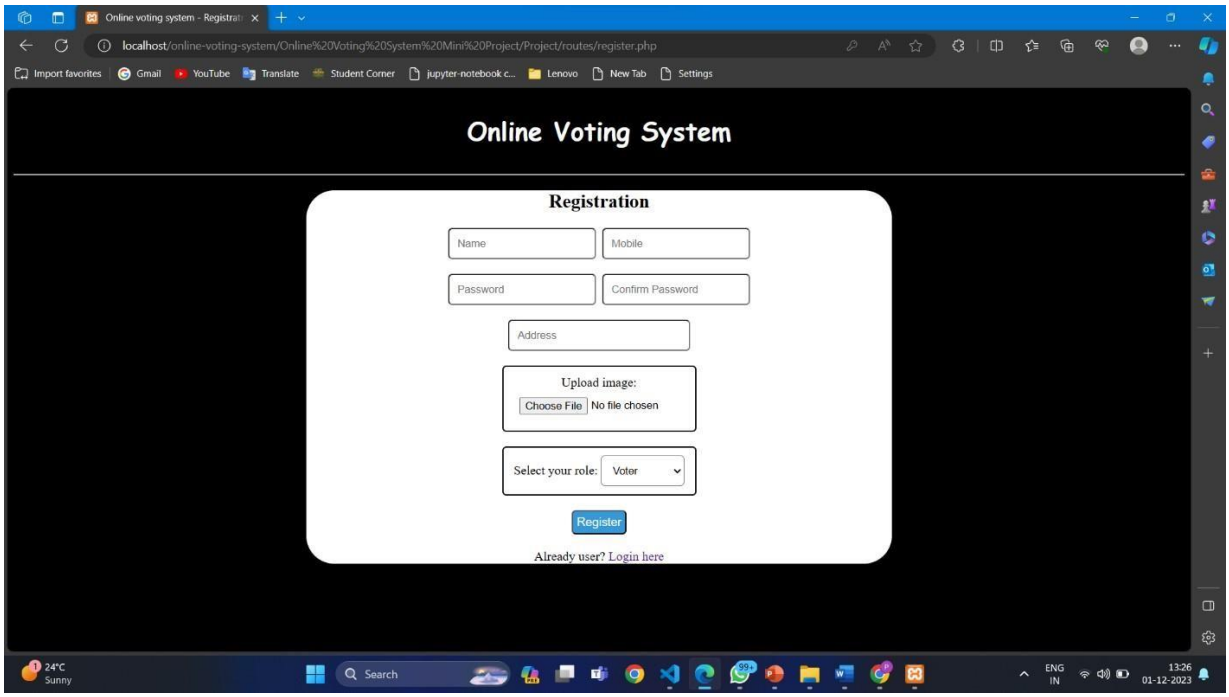
Server	**:**	Replit(Online Compiler)

PROJECT SNAP SHOTS

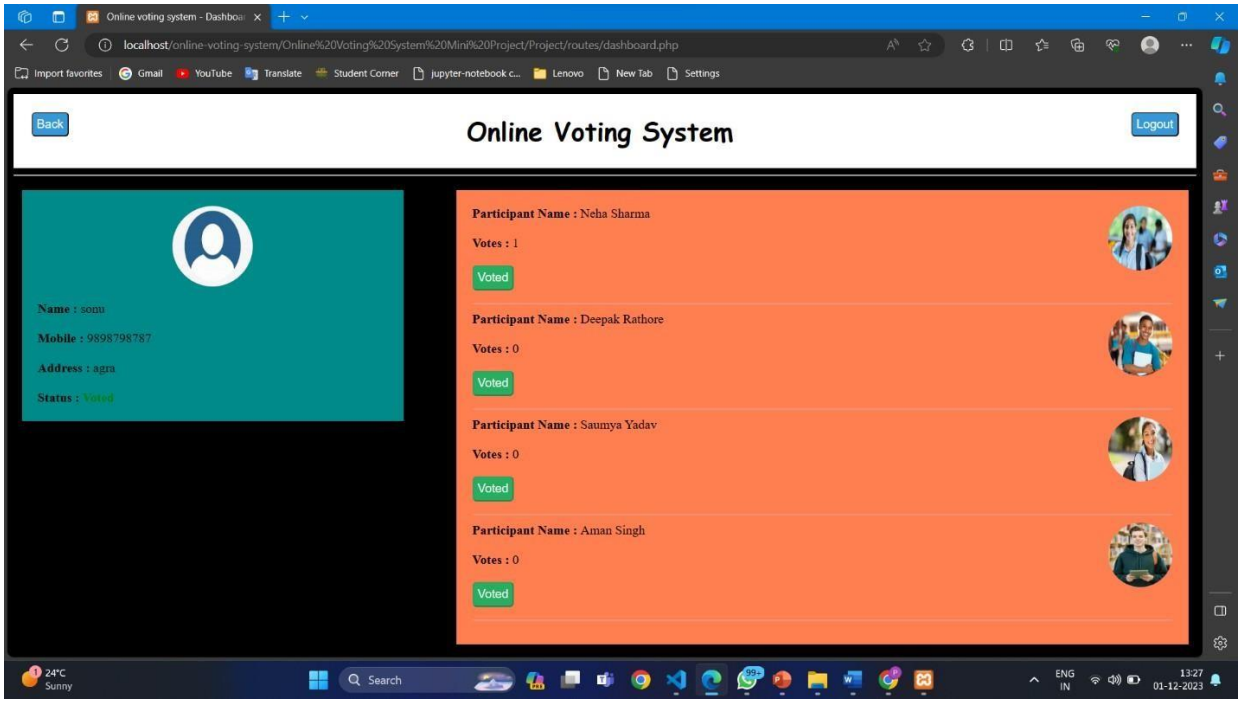
Homepage



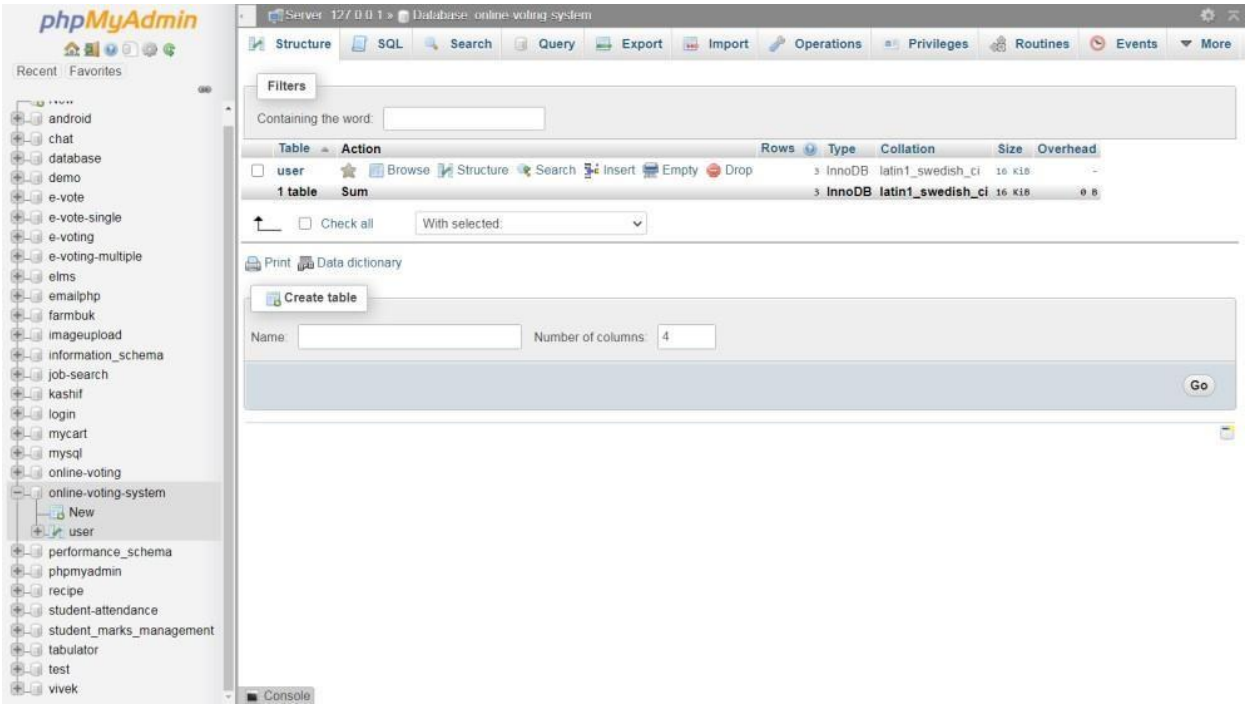
Registration Page



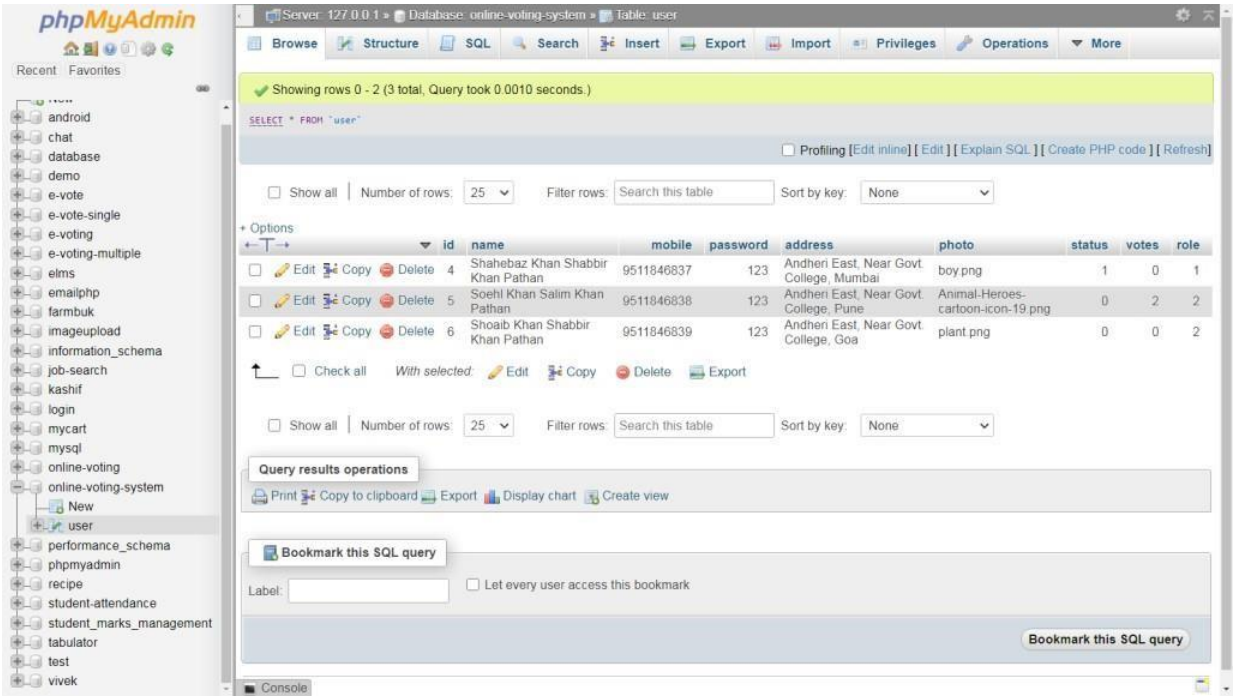
Voter Dashboard (after voting)



MySQL Database



Database table structure



SOURCE CODE

Homepage

```
<html>

    <head>
        <title>Online voting system - Home</title>
        <link rel="stylesheet" href="css/stylesheet.css">
</head>      <body>
            <center>
            <div id="headerSection">
            <h1>Online Voting System</h1>
            </div>
            <hr>


            <div id="loginSection">
                <h2>Login</h2>
                <form action="api/login.php" method="POST">
                    <input type="number" name="mob" placeholder="Enter mobile" required><br
><br>
                    <input type="password" name="pass" placeholder="Enter password" require
d><br><br>
                    <select name="role" style="width: 15%; border: 2px solid black">
```

```
                        <option value="1">Voter</option>
                        <option value="2">Group</option>
                    </select><br><br>
                    <button id="loginbtn" type="submit" name="loginbtn">Login</button><br><
br>
                    New user? <a href="routes/register.php">Register here</a>
                </form>
</div>


            </center>
    </body>
</html>
```

Registration

```
<html>
    <head>
        <title>Online voting system - Registratrion</title>
        <link rel="stylesheet" href="../css/stylesheet.css">
    </head>
    <body>
        <center>
            <div id="headerSection">
            <h1>Online Voting System</h1>
            </div>
<hr>


            <h2>Registration</h2>
                <form action="../api/register.php" method="POST"
enctype="multipart/formdata">
                    <input type="text" name="name" placeholder="Name" required> 
                    <input type="number" name="mob" placeholder="Mobile" required><br><br>
<input type="password" name="pass" placeholder="Password" required>&nbs p
                    <input type="password" name="cpass" placeholder="Confirm Password" requ
ired><br><br>
                    <input style="width: 31%" type="text" name="add" placeholder="Address"
required><br><br>
                    <div id="upload" style="width: 30%">
                        Upload image: <input type="file" id="profile" name="image" required
>
                    </div><br>
                    <div id="upload" style="width: 30%">                            Select
your role:
```

```
                        <select name="role">
                            <option value="1">Voter</option>
                            <option value="2">Group</option>
                        </select><br>
                    </div><br>
                    <button id="loginbtn" type="submit" name="registerbtn">Register</button
><br><br>
                    Already user? <a href="../">Login here</a>
                </form>
            </center>
    </body>
</html>
```

Main Dashboard

```
<?php     session_start();
if(!isset($_SESSION['id'])){         header("location:
../");
    }
    $data = $_SESSION['data'];


if($_SESSION['status']==1){
        $status = '<b style="color: green">Voted</b>';
    }     else{
        $status = '<b style="color: red">Not Voted</b>';
    }
?>



<html>
    <head>
        <title>Online voting system - Dashboard</title>
        <link rel="stylesheet" href="../css/stylesheet.css">
    </head>
<body>
            <center>
            <div id="headerSection">
            <a href="../"><button id="back-button"> Back</button></a>
            <a href="logout.php"><button id="logout-button">Logout</button></a>
            <h1>Online Voting System</h1>                </div>
```

```
            </center>
<hr>


            <div id="mainSection">
                <div id="profileSection">
                    <center><img src="../uploads/<?php echo $data['photo']?>" height="100"
width="100"></center><br>
                    <b>Name : </b><?php echo $data['name'] ?><br><br>
                    <b>Mobile : </b><?php echo $data['mobile'] ?><br><br>
                    <b>Address : </b><?php echo $data['address'] ?><br><br>
<b>Status : </b><?php echo $status ?>
                </div>
                <div id="groupSection">
                    <?php
if(isset($_SESSION['groups'])){
$groups    =    $_SESSION['groups'];
for($i=0; $i<count($groups); $i++){
?>
                                <div style="border-bottom: 1px solid #bdc3c7; marginbottom:
10px">
                                <img style="float: right" src="../uploads/<?php echo $group
s[$i]['photo']?>" height="80" width="80">
                                <b>Group Name : </b><?php echo $groups[$i]['name']?><br><br
>

                                <b>Votes :</b> <?php echo $groups[$i]['votes']?><br><br>
                                <form method="POST" action="../api/vote.php">
                                <input type="hidden" name="gvotes" value="<?php echo $group
s[$i]['votes'] ?>">
                                <input type="hidden" name = "gid" value="<?php echo $groups
[$i]['id'] ?>">
                                <?php
                                 if($_SESSION['status']==1){
                                    ?>
                                    <button disabled style="padding: 5px; fontsize:
15px; background-color: #27ae60; color: white; borderradius: 5px;"
type="button">Voted</button>
                                    <?php
                                }                                    else{
                                    ?>
                                    <button    style="padding:    5px;
fontsize: 15px; background-color: #3498db; color: white; borderradius:
5px;" type="submit">Vote</button>                                    <?php
                                }
                                ?>
                                </form>
                                </div>
                            <?php
                        }
}                   else{
?>
                            <div style="border-bottom: 1px solid #bdc3c7; marginbottom:
10px">
                                <b>No groups available right now.</b>









                            </div>
                        <?php
                    }
                    ?>
```

```
                </div>
            </div>
    </body>  </html>
```

Logout

```
<?php     session_start();     session_destroy();
header('location:../index.php');  ?>
```

CSS

```
input {   padding:
10px;
borderradius: 5px;
}  select {
padding: 10px;
border-radius: 5px;
}


#upload {   padding: 10px;
border-radius: 5px;
border: 2px solid black;  }


#headerSection {   padding:
2px;
  font-family: Cursive;  }


#loginSection {   padding:
5px;
}           body      {
backgroundcolor:
#b8e994;  }
```

```
#loginbtn {   padding: 5px;
font-size: 15px;
background-color: #3498db;
color: white;
borderradius: 5px;  }


#reglink {   padding: 5px;
font-size: 15px;
background-color: #3498db;
color: white;
borderradius: 5px;
textdecoration: none;  }
a {   textdecoration:
none;  }


#mainSection          {
padding: 10px;  }


#profileSection {
width: 30%;   float:
left;   background-color:
white;   padding: 20px;  }


#groupSection {   width:
60%;   float: right;
background-color: white;
padding: 20px;  }


#back-button {   float:
left;   margin-left: 20px;
margin-top: 20px;
padding: 5px;   font-size:
15px;   background-color:
#3498db;   color: white;
border-radius: 5px;  }


#logout-button {   float:
right;   margin-right: 20px;
margin-top: 20px;   padding:
5px;   font-size: 15px;
background-color:
#3498db;   color: white;
```

```
border-radius:
5px;
}
```

Login API

```
<?php     session_start();
    include("connection.php");


    $mobile = $_POST['mob'];
    $pass = $_POST['pass'];
    $role = $_POST['role'];


    $check = mysqli_query($connect, "select * from user where mobile='$mobile' and password
='$pass' and role='$role' ");


if(mysqli_num_rows($check)>0){
        $getGroups = mysqli_query($connect, "select name, photo, votes, id from user where
role=2 ");         if(mysqli_num_rows($getGroups)>0){
            $groups = mysqli_fetch_all($getGroups, MYSQLI_ASSOC);
            $_SESSION['groups'] = $groups;
        }
        $data = mysqli_fetch_array($check);
        $_SESSION['id'] = $data['id'];
        $_SESSION['status'] = $data['status'];
        $_SESSION['data'] = $data;         echo '<script>                 window.location
= "../routes/dashboard.php";
            </script>';
    }     else{         echo '<script>
alert("Invalid credentials!");                 window.location
= "../";
```

```
            </script>';
    }
    ?>
```

Database connectivity

```
<?php
$connect = mysqli_connect("localhost", "root", "", "online-voting-system");   ?>
```

Registration API

```php
<?php
include("connection.php");


    $name = $_POST['name'];
    $mobile = $_POST['mob'];
    $pass = $_POST['pass'];
    $cpass = $_POST['cpass'];
    $add = $_POST['add'];
    $image = $_FILES['image']['name'];
    $tmp_name = $_FILES['image']['tmp_name'];
    $role = $_POST['role'];       if($cpass!=$pass){
echo '<script>                alert("Passwords do not
match!");                 window.location =
"../routes/register.php";
            </script>';
}     else{
move_uploaded_file($tmp_name,"../uploads/$image");
        $insert = mysqli_query($connect, "insert into user (name, mobile, password, address
, photo, status, votes, role) values('$name', '$mobile', '$pass', '$add', '$image', 0, 0, '
$role') ");         if($insert){            echo '<script>
alert("Registration successfull!");
window.location =
"../";
                </script>';
        }
    }
?>
```

Voting API

```
<?php     session_start();
    include("connection.php");
$votes = $_POST['gvotes'];
    $total_votes= $votes+1;

$gid = $_POST['gid'];
    $uid = $_SESSION['id'];


    $update_votes = mysqli_query($connect, "update user set votes='$total_votes' where id='
$gid'");
    $update_status = mysqli_query($connect, "update user set status=1 where id='$uid'");
if($update_status and $update_votes){
        $getGroups = mysqli_query($connect, "select name, photo, votes, id from user where
role=2 ");
        $groups = mysqli_fetch_all($getGroups, MYSQLI_ASSOC);
        $_SESSION['groups'] = $groups;
$_SESSION['status'] = 1;         echo '<script>
alert("Voting                   successfull!");
window.location = "../routes/dashboard.php";
                </script>';
    }     else{         echo '<script>                    alert("Voting
failed!.. Try again.");                     window.location =
"../routes/dashboard.php";
                </script>';
    }
```

```
?>
```

CONCLUSION

So the final conclusion we make here is that our new online voting system is much better and easy to use than traditional voting system. Almost all problems that we have discussed in existing voting system are resolved by the help of this application. So the launch of this application would create many opportunities for those who are frequently involved in conducting elections for different purposes.

REFERENCES

Programming languages

- https://www.w3schools.com
- https://www.javapoint.com
- https://www.youtube.com
- https://www.wikipedia.com

Software

- www.apachefriends.org
- https://code.visualstudio.com/