

ONLINE VOTING SYSTEM

A MINI PROJECT REPORT

Submitted by

Murugappan M (18euit087)

Natraaja kumaran S 18EUIT091)

Nawinkrishna AS (18euit093)

Nekha KS (18EUIT094)

*In partial fulfillment of the requirements
for the award of the degree*

of

BACHELOR OF TECHNOLOGY

in

INFORMATION TECHNOLOGY



SRI KRISHNA COLLEGE OF ENGINEERING AND TECHNOLOGY

(An Autonomous Institution)

**(Approved by AICTE and Affiliated to Anna University,
Chennai)**

ACCREDITED BY NAAC WITH "A" GRADE

MARCH 2021



SRI KRISHNA COLLEGE OF ENGINEERING AND TECHNOLOGY

(An Autonomous Institution)

(Approved by AICTE and Affiliated to Anna University,
Chennai)

ACCREDITED BY NAAC WITH "A" GRADE

BONAFIDE CERTIFICATE

Certified that this project report "**Online Voting system**" is the bonafide work of "Murugappan M (18euit087), Natraaja kumaran S 18EUIT091), Nawinkrishna AS (18euit093), Nekha KS (18EUIT094)" who carried out the project work under my supervision.

SIGNATURE

[Dr. A.SAJEEV RAM]

Assistant Professor

Information Technology,

Sri Krishna College of Engg & Tech
Coimbatore - 641008.

SIGNATURE

[Dr. N. SUSILA]

PROFESSOR & HEAD

Information Technology,

Sri Krishna College of Engg & Tech
Coimbatore-641008.

This project report submitted for the autonomous project viva-voice examination held on

INTERNAL EXAMINER

EXTERNAL EXAMINER

TABLE OF CONTENT:

- **Abstract**
- **Introduction To Online Voting System**
- **Background Of Study**
- **Significance Of Study**
- **Objective of Project**
- **Project Justification**
- **Scope of Study**
- **Old Methods Of Voting**
- **Security Issues Of Online Voting**
- **General Requirements**
- **Problems with Existing Voting Registration System**
- **Software and Hardware Required**
- **Front-End & Back-End**
- **Data Flow Diagram**
- **ER Diagram**
- **Database Architecture**
- **Database Tables**
- **Database Queries**
- **Conclusion**
- **Appendix(Snapshots)**

ABSTRACT

The word “vote” means to choose from a list, to elect or to determine. The main goal of voting (in a scenario involving the citizens of a given country) is to come up with leaders of the people’s choice.

Most countries, Kenya not an exception have problems when it comes to voting. Some of the problems involved include rigging votes during election, insecure or inaccessible polling stations, inadequate polling materials and also inexperienced personnel.

This online voting/polling system seeks to address the above issues. It should be noted that with this system in place, the users, citizens in this case shall be given ample time during the voting period. They shall also be trained on how to vote online before the election time.

INTRODUCTION TO ONLINE VOTING SYSTEM:

“ONLINE VOTING SYSTEM” is an online voting technique. In this system people who have citizenship of Kenya and whose age is above 18 years of age and any sex can give his\her vote online without going to any physical polling station. There is a database which is maintained in which all the names of voters with complete information is stored.

In “ONLINE VOTING SYSTEM” a voter can use his\her voting right online without any difficulty. He\She has to be registered first for him/her to vote. Registration is mainly done by the system administrator for security reasons. The system Administrator registers the voters on a special site of the system visited by him only by simply filling a registration form to register voter. Citizens seeking registration are expected to contact the system administrator to submit their details. After the validity of them

being citizens of India has been confirmed by the system administrator by comparing their details submitted with those in existing databases such as those as the Registrar of Persons, the citizen is then registered as a voter.

After registration, the voter is assigned a secret Voter ID with which he/she can use to log into the system and enjoy services provided by the system such as voting. If invalid/wrong details are submitted, then the citizen is not registered to vote.

BACKGROUND OF STUDY:

The Online voting system (OVS) also known as e-voting is a term encompassing several different types of voting embracing both electronic means of counting votes. Electronic voting technology can include punched cards, optical scan voting systems and specialized voting kiosks (including self-contained direct-recording electronic voting systems or DRE). It can also involve transmission of ballots and votes via telephones, private computer networks, or the internet.

Online voting is an electronic way of choosing leaders via a web driven application. The advantage of online voting over the common “queue method” is that the voters have the choice of voting at their own free time and there is reduced congestion. It also minimizes on errors of vote counting. The individual votes are submitted in a database which can be queried to find out who of the aspirants for a given post has the highest number of votes.

This system is geared towards increasing the voting percentage in Kenya since it has been noted that with the old voting method {the Queue System}, the voter turnout has been a wanting case. With system in place also, if high security is applied, cases of false votes shall be reduced.

With the “ONLINE VOTING SYSTEM”, a voter can use his\her voting right online without any difficulty. He\She has to register as a voter first before being authorized to vote. The registration should be done prior to the voting date to enable data update in the database.

However, not just anybody can vote. For one to participate in the elections, he/she must have the requirements. For instance, he/she must be a registered citizen i.e. must be 18 and above years old. As already stated, the project ‘Online Voting’ provides means for fast and convenient voting and access to this system is limited only to registered voters.

SIGNIFICANCE OF STUDY:

The main purposes of OVS include:

- Provision of improved voting services to the voters through fast, timely and convenient voting.
- Reduction of the costs incurred by the Kenyan Electoral Commission during voting time in paying the very many clerks employed for the sake of the success of the manual system.
- Check to ensure that the members who are registered are the only ones to vote. Cases of “Dead People” voting are also minimized.
- Online voting system (OVS) will require being very precise or cost cutting to produce an effective election management system.
- Therefore crucial points that this (OVS) emphasizes on are listed below.
 - i. Require less number of staff during the election.
 - ii. This system is a lot easier to independently moderate the elections and subsequently reinforce its transparency and fairness.
 - iii. Less capital, less effort, and less labor intensive, as the primary cost and effort will focus primarily on creating, managing, and running a secure online portal.
 - iv. Increased number of voters as individual will find it easier and more convenient to vote, especially those abroad.

Objectives of The PROJECT:

The specific objectives of the project include:

- ❖ Reviewing the existing/current voting process or approach in Kenya;
- ❖ Coming up with an automated voting system in Kenya;
- ❖ Implementing a an automated/online voting system;
- ❖ Validating the system to ensure that only legible voters are allowed to vote.

Project justification:

The ONLINE VOTING SYSTEM-INDIA shall reduce the time spend making long queues at the polling stations during voting. It shall also enable the voters to vote from any part of the globe as explained since this is an online application available on the internet. Cases of vote miscounts shall also be solved since at the backend of this system resides a well developed database using MYSQL that can provide the correct data once it's correctly queried. Since the voting process shall be open as early as possible, the voters shall have ample time to decide when and whom to vote for.

SCOPE OF STUDY:

It is focused on studying the existing system of voting in Kenya and to make sure that the peoples vote is counts, for fairness in the elective positions. This is also will produce:

- Less effort and less labor intensive, as the primary cost and focus primary on creating, managing, and running a secure web voting portal.
- Increasing number of voters as individuals will find it easier and more convenient to vote, especially those abroad.

Old Methods of Voting:

1. **Paper-based voting:** The voter gets a blank ballot and use a pen or a marker to indicate he want to vote for which candidate. Hand-counted ballots is a time and labor consuming process, but it is easy to manufacture paper ballots and the ballots can be retained for verifying, this type is still the most common way to vote.
2. **Lever voting machine:** Lever machine is peculiar equipment, and each lever is assigned for a corresponding candidate. The voter pulls the lever to poll for his favorite candidate. This kind of voting machine can count up the ballots automatically. Because its interface is not user-friendly enough, giving some training to voters is necessary.
3. **Direct recording electronic voting machine:** This type, which is abbreviated to DRE, integrates with keyboard; touch screen, or buttons for the voter press to poll. Some of them lay in voting records and counting the votes is very quickly. But the other DRE without keep voting records are doubted about its accuracy.
4. **Punch card:** The voter uses metallic hole-punch to punch a hole on the blank ballot. It can count votes automatically, but if the voter's perforation is incomplete, the result is probably determined wrongfully.
5. **Optical voting machine:** After each voter fills a circle correspond to their favorite candidate on the blank ballot, this machine selects the darkest mark on each ballot for the vote then computes the total result.

This kind of machine counts up ballots rapidly. However, if the voter fills over the circle, it will lead to the error result of optical-scan.

Recent years, a considerable number of countries has adopted E-voting for their official elections. These countries include; America, Belgium, Japan and Brazil.

SECURITY ISSUES OF ONLINE VOTING:

Foreign experience revealed that they are often confronted by security issues while the online voting system is running. The origin of the security issues was due to not only outsider (such as voters and attackers) but also insider (such as system developers and administrators), even just because the inheritance of some objects in the source code are unsuitable. These errors caused the voting system to crash.

The proposed solutions were correspondingly outlined to hold back these attacks. For example, to avoid hacker making incursion into the voting system via network, we can design our system to transmit data without network. Another example is to limit voter to input particular data, so that we can prevent the command injection from running

Requirements:

- 1) Registration of the voter is done by ELECTION COMMISSION OF INDIA.
- 2) ELECTION COMMISSION OF INDIA can change the information any time if required.
- 3) Registration of the Voter depends upon the information filled by the user.
- 4) Voter is given a unique ID and PASSWORD.
- 5) In the DATABASE information of every voter is stored.
- 6) Database shows the information of every user.

Problems with the Existing Voter Registration System:

The problems of the existing manual system of voting include among others the following:

- 1. Expensive and Time consuming:** The process of collecting data and entering this data into the database takes too much time and is expensive to conduct, for example, time and money is spent in printing data capture forms, in preparing registration stations together with human resources, and there after advertising the days set for registration process including sensitizing voters on the need for registration, as well as time spent on entering this data to the database.
- 2. Too much paper work:** The process involves too much paper work and paper storage which is difficult as papers become bulky with the population size.
- 3. Errors during data entry:** Errors are part of all human beings; it is very unlikely for humans to be 100 percent efficient in data entry.
- 4. Loss of registration forms:** Some times, registration forms get lost after being filled in with voters' details, in most cases these are difficult to follow-up and therefore many remain unregistered even though they are voting age nationals and interested in exercising their right to vote.
- 5. Short time provided to view the voter register:** This is a very big problem since not all people have free time during the given short period of time to check and update the voter register.
- 6.** Above all, a number of voters end up being locked out from voting.

SOFTWARE REQUIREMENTS:

- i. **MYSQL DBMS-** It allows combination, extraction, manipulation and organization of data in the voters' database. It is platform independent and therefore can be implemented and used across several such as Windows, Linux server and is compatible with various hardware mainframes. It is fast in performance, stable and provides business value at a low cost.
- ii. **Net Beans IDE 7.1.2-** The Net Beans IDE is an award-winning integrated development environment available for Windows, Mac, Linux, and Solaris. The Net Beans project consists of an open-source IDE and an application platform that enable developers to rapidly create web, enterprise, desktop, and mobile applications using the Java platform, as well as PHP, JavaScript and Ajax, Groovy and Grails, and C/C++.

The Net Beans project is supported by a vibrant developer community and offers extensive documentation and training resources as well as a diverse selection of third-party plugins.

- iii. **JAVA coding-**This is for advanced user who find PHP codes easy to work with.
- iv. **Testing-** is done via WAMPSEVER.
- v. **Web browsers:** Mozilla Firefox, Google chrome, Opera and Internet Explorer
- vi. **Reporting Tool** i.e. through Data Report.

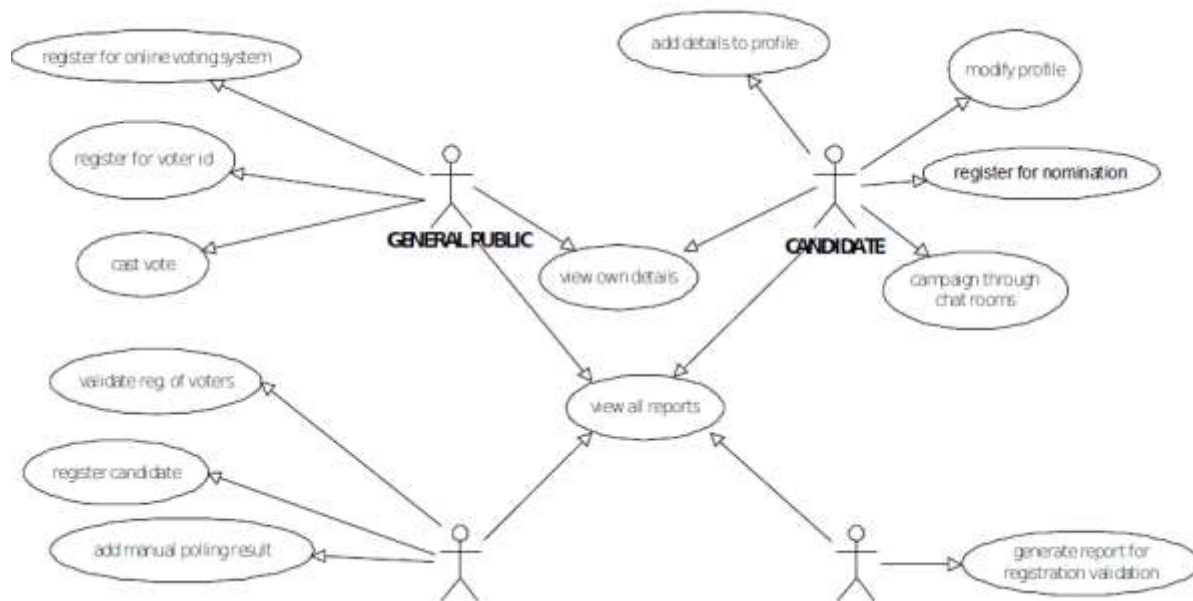
HARDWARE REQUIREMENTS:

- **Microsoft Windows 10 Professional:**

- ❑ **Processor:** Intel(R) Core(TM) i5-7200U CPU @ 2.50GHz 2.71 GHz
- ❑ **Memory:** 8 GB
- ❑ **Disk space:** 120 GB of free disk space

- **Ubuntu 20.04.2.0:**

- ❑ **Processor:** 2.17GHz Intel i5 Core
- ❑ **Memory:** 8 GB
- ❑ **Disk space:** 1 GB of free disk space



(BLOCK DIAGRAM)

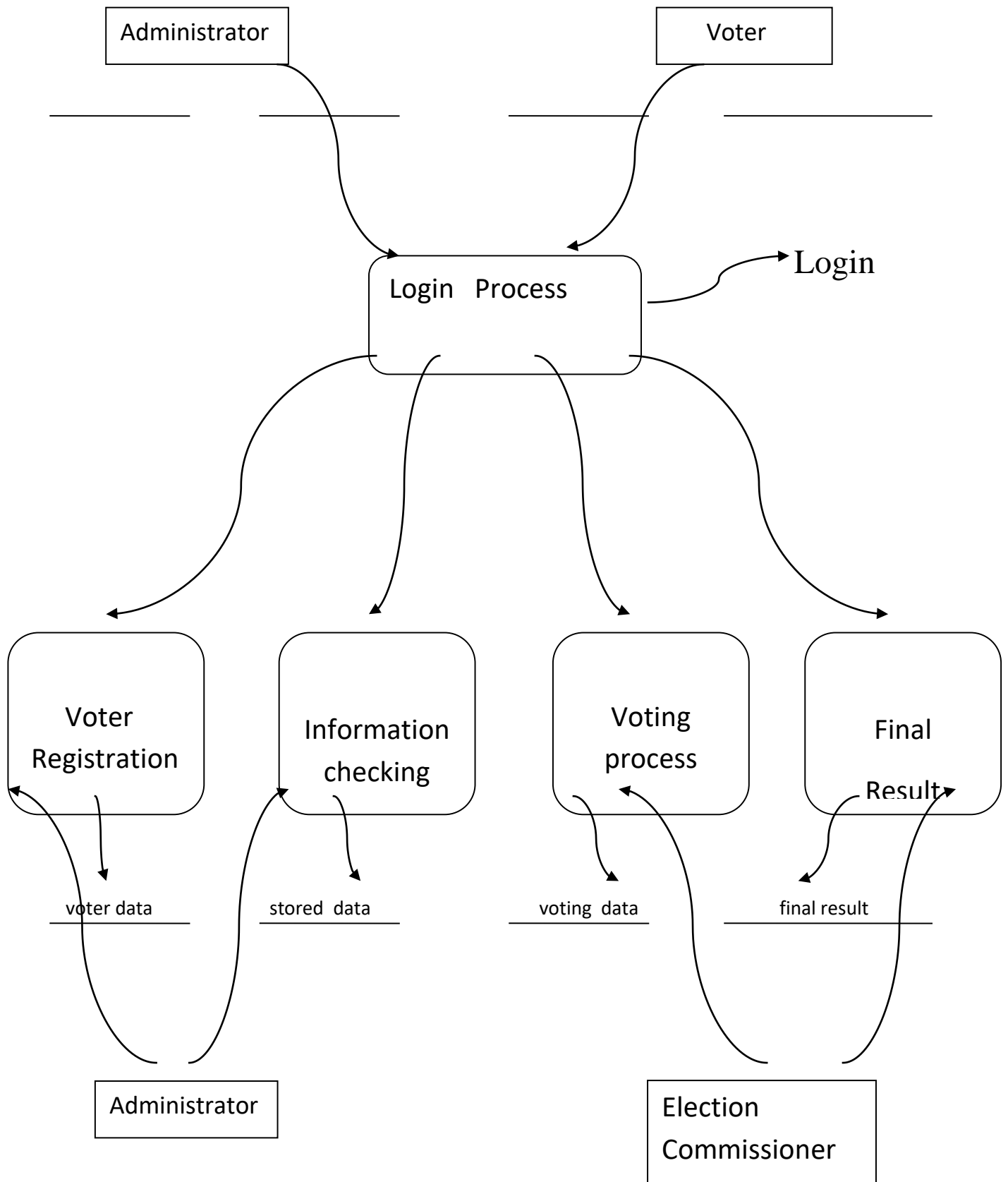
FRONT – END AND BACK-END:

In their most general meanings, the terms front end and back end refer to the initial and the end stages of a process flow. In [software design](#), the **front-end** is the part of a software system that deals with the user, and the **back-end** is the part that processes the input from the front-end. The separation of software systems into "front ends" and "back ends" is a kind of [abstraction](#) that helps to keep different parts of the system separated. The general idea is that the front-end is responsible for collecting input from the user, which can be in a variety of forms, and processing it in such a way that it conforms to a specification that the back-end can use. The connection of the front-end to the back-end is a kind of [interface](#).

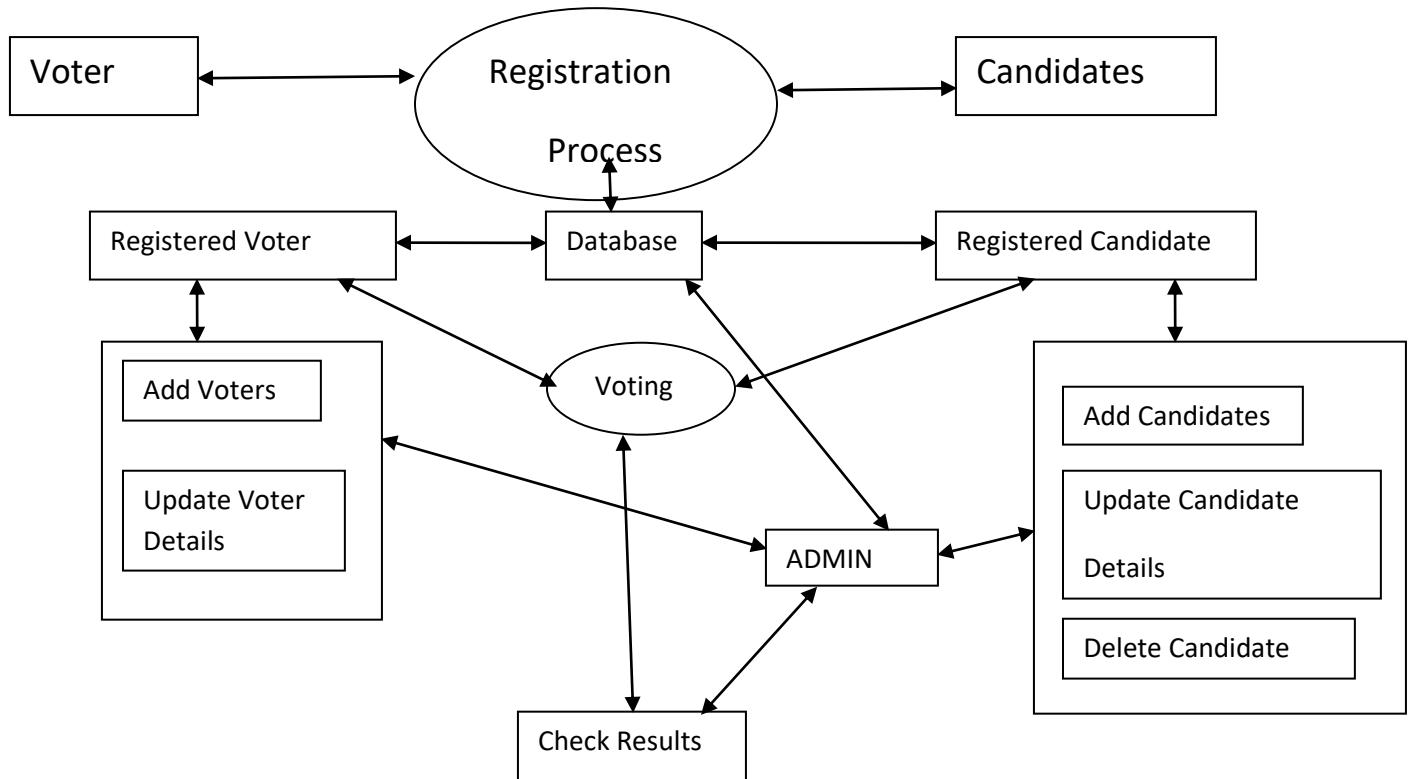
Front-end and back-end are terms used to characterize program interfaces and services relative to the initial user of these interfaces and services. (The "user" may be a human being or a program.) A "front-end" [application](#) is one that application users interact with directly. A "back-end" application or program serves indirectly in support of the front-end services, usually by being closer to the required resource or having the capability to communicate with the required resource. The back-end application may interact directly with the front-end or, perhaps more typically, is a program called from an intermediate program that mediates front-end and back-end activities. These terms acquire more special meanings in particular areas:-

- (1) For [software applications](#), front end is the same as [user interface](#).
- (2) In [client/server](#) applications, the [client part](#) of the [program](#) is often called the front end and the server part is called the back end.
- (3) [Compilers](#), the programs that translate [source code](#) into [object code](#), are often composed of two parts: a front end and a back end. The front end is responsible for checking [syntax](#) and detecting errors, whereas the back end performs the actual translation into object code.

Data Flow Diagram:



ER Diagram:



Architecture:

When automation first hit business, it was in the form of a huge "Mainframe" computer. Here, a central computer served the whole business community and was accessed via dumb terminals. All processing took place on a single computer - and therefore in one place. All resources associated with the

computer (tape and disk drives, printers etc.) were attached to this same computer. This is single tier (or 1-tier) computing. It is simple, efficient, uncomplicated, but terribly expensive to run.



All users run their programs from a single machine. The ease with which deployment and even development occurs makes this model very attractive. The cost of the central machine makes this architecture prohibitive for most companies, especially as system costs and return on investment (ROI) are looked at carefully nowadays.

Data Base Tables:-

This project uses many tables:

- Admin
- Voter
- Candidate

Admin Table:-

Field Name	Data Type	Description
Username	Varchar	Login id for Admin.(Primary key)
Password	Varchar	Password for Login

Voter Table:-

Field Name	Data Type	Description
VoterId	Integer	Login id for Voter(Primary key)
Name	Varchar	Name of the voter
Sex	Varchar	Sex of voter
Age	Integer	Age of voter
City	Varchar	City of voter
Security	Varchar	Security Question
Status	Boolean	Status of voter(he/she can vote or not)

Candidate Table:-

Field Name	Data Type	Description
Symbol	Varchar	Party Symbol (Primary key)
Name	Varchar	Name of the voter

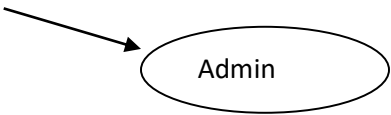
Sex	Varchar	Sex of voter
Age	Integer	Age of voter
City	Varchar	City of voter
Count	Integer	Count the no of votes

User Name

Administrator

Password

1234



Voter Id

123456



CONCLUSION:

This Online Voting system will manage the Voter's information by which voter can login and use his voting rights. The system will incorporate all features of Voting system. It provides the tools for maintaining voter's vote to every party and it count total no. of votes of every party. There is a DATABASE which is maintained by the ELECTION COMMISSION OF INDIA in which all the names of voter with complete information is stored.

In this user who is above 18 year's register his/her information on the database and when he/she want to vote he/she has to login by his id and password and can vote to any party only single time. Voting detail store in database and the result is displayed by calculation. By online voting system percentage of voting is increases. It decreases the cost and time of voting process. It is very easy to use and It is vary less time consuming. It is very easy to debug.

APPENDIX: (SNAPSHOTS)

VOTER LOGIN:

```
<?php
    session_start();
    if(isset($_SESSION['admin'])){
        header('location: admin/home.php');
    }
    if(isset($_SESSION['voter'])){
        header('location: home.php'); }
?>

<?php include 'includes/header.php'; ?>

<body class="hold-transition login-page">

<div class="login-box">
    <div class="login-logo">
        <b>Voting System</b>
    </div>
    <div class="login-box-body">
        <p class="login-box-msg">Sign in to start your session</p>
        <form action="login.php" method="POST">
            <div class="form-group has-feedback">
                <input type="text" class="form-control" name="voter" placeholder="Voter's ID"
required>
                <span class="glyphicon glyphicon-user form-control-feedback"></span>
            </div>
            <div class="form-group has-feedback">
                <input type="password" class="form-control" name="password" placeholder="Password"
required>
```



```

        <span class="glyphicon glyphicon-lock form-control-feedback"></span>

    </div>

    <div class="row">

        <div class="col-xs-4">

            <button type="submit" class="btn btn-primary btn-block btn-flat" name="login"><i
class="fa fa-sign-in"></i> Sign In</button>

        </div>

    </div>

</form>

</div>

<?php
    if(isset($_SESSION['error'])){
        echo "

        <div class='callout callout-danger text-center mt20'>

            <p>".$_SESSION['error']. "</p> </div>";

        unset($_SESSION['error']);
    }?></div>

    <?php include 'includes/scripts.php' ?>

</body>

</html>

```

Admin Dashboard:

```
<?php include
'includes/session.php'; ?>
<?php include
'includes/header.php'; ?>
<body class="hold-transition
skin-blue layout-top-nav">
<div class="wrapper">

<?php include
'includes/navbar.php'; ?>

<div class="content-wrapper">
<div class="container">

<!-- Main content -->
<section class="content">
<?php
$parse = parse_ini_file('admin/config.ini', FALSE, INI_SCANNER_RAW);
$title = $parse['election_title'];
?>
<h1 class="page-header text-center title"><b><?php echo strtoupper($title);
?></b></h1>
<div class="row">
<div class="col-sm-10 col-sm-offset-1">
<?php
if(isset($_SESSION['error'])) {
?>
<div class="alert alert-danger alert-dismissible">
<button type="button" class="close" data-dismiss="alert" aria-
hidden="true">&times;</button>
<ul>
<?php
foreach($_SESSION['error'] as $error){
echo "
<li>".$error."</li>
";
}
?>
</ul>
</div>
<?php
unset($_SESSION['error']);

}
if(isset($_SESSION['success'])) {
echo "
<div class='alert alert-success alert-dismissible'>
<button type='button' class='close' data-dismiss='alert' aria-
hidden='true'>&times;</button>
<h4><i class='icon fa fa-check'></i> Success!</h4>
".$_SESSION['success']."
</div>
";
unset($_SESSION['success']);
}

?>
```



```

<div class="alert alert-danger alert-dismissible" id="alert"
style="display:none;">
<button type="button" class="close" data-dismiss="alert" aria-
hidden="true">&times;</button>
<span class="message"></span>
</div>

<?php
$sql = "SELECT * FROM votes WHERE voters_id = '". $voter['id']. "'";
$vquery = $conn->query($sql);
if($vquery->num_rows > 0){
?>
<div class="text-center">
<h3>You have already voted for this election.</h3>
<a href="#view" data-toggle="modal" class="btn btn-flat btn-primary btn-
lg">View Ballot</a>
</div>
<?php
}
else{
?>
<!-- Voting Ballot -->
<form method="POST" id="ballotForm" action="submit_ballot.php">
<?php
include 'includes/slugify.php';

$candidate = '';
$sql = "SELECT * FROM positions ORDER BY priority ASC";
$query = $conn->query($sql);
while($row = $query->fetch_assoc()){
$sql = "SELECT * FROM candidates WHERE position_id='". $row['id']. "'";
$cquery = $conn->query($sql);
while($crow = $cquery->fetch_assoc()){
$slug = slugify($row['description']);
$checked = '';
if(isset($_SESSION['post'][$slug])){
$value = $_SESSION['post'][$slug];

if(is_array($value)){
foreach($value as $val){
if($val == $crow['id']){
$checked = 'checked';
}
}
}
else{
if($value == $crow['id']){
$checked = 'checked';
}
}
}
}
$input = ($row['max_vote'] > 1) ? '<input type="checkbox" class="flat-red
' . $slug. '" name="'. $slug. '" value="'. $crow['id']. '" ' . $checked. '>' :
'<input type="radio" class="flat-red ' . $slug. '"
name="'. slugify($row['description']). '" value="'. $crow['id']. '"
' . $checked. '>';
$image = (!empty($crow['photo'])) ? 'images/' . $crow['photo'] :
'images/profile.jpg';
$candidate .= '
<li>
' . $input. '<button type="button" class="btn btn-primary btn-sm btn-flat clist
platform" data-platform="'. $crow['platform']. '" data-
fullname="'. $crow['firstname']. ' ' . $crow['lastname']. '"><i class="fa fa-

```

```

search"></i> Platform</button><span class="cname clist">'. $crow['firstname'].'
'. $crow['lastname'].'</span>
</li>
';
}

$instruct = ($row['max_vote'] > 1) ? 'You may select up to
'. $row['max_vote'].' candidates' : 'Select only one candidate';

echo '
<div class="row">
<div class="col-xs-12">
<div class="box box-solid" id="'. $row['id'].'">
<div class="box-header with-border">
<h3 class="box-title"><b>'. $row['description'].'</b></h3>
</div>
<div class="box-body">
<p>'. $instruct.'
<span class="pull-right">
<button type="button" class="btn btn-success btn-sm btn-flat reset" data-
desc="'. slugify($row['description']).'"><i class="fa fa-refresh"></i>
Reset</button>
</span>
</p>
<div id="candidate_list">
<ul>
'. $candidate.'
</ul>
</div>
</div>
</div>
</div>
</div>
';

$candidate = '';

}

?>
<div class="text-center">
<button type="button" class="btn btn-success btn-flat" id="preview"><i
class="fa fa-file-text"></i> Preview</button>
<button type="submit" class="btn btn-primary btn-flat" name="vote"><i
class="fa fa-check-square-o"></i> Submit</button>
</div>
</form>
<!-- End Voting Ballot -->
<?php
}

?>

</div>
</div>
</section>

</div>
</div>

<?php include 'includes/footer.php'; ?>
<?php include 'includes/ballot_modal.php'; ?>

```



```

</div>

<?php include 'includes/scripts.php'; ?>
<script>
$(function(){
$('.content').iCheck({
checkboxClass: 'icheckbox_flat-green',
radioClass: 'iradio_flat-green'
});

$(document).on('click', '.reset', function(e){
e.preventDefault();
var desc = $(this).data('desc');
$('.'+desc).iCheck('uncheck');
});

$(document).on('click', '.platform', function(e){
e.preventDefault();
$('#platform').modal('show');
var platform = $(this).data('platform');
var fullname = $(this).data('fullname');
$('.candidate').html(fullname);
$('#plat_view').html(platform);
});

$('#preview').click(function(e){
e.preventDefault();
var form = $('#ballotForm').serialize();
if(form == ''){
$('.message').html('You must vote atleast one candidate');
$('#alert').show();
}
else{
$.ajax({
type: 'POST',
url: 'preview.php',
data: form,
dataType: 'json',
success: function(response){
if(response.error){
var errmsg = '';
var messages = response.message;
for (i in messages) {
errmsg += messages[i];
}
$('.message').html(errmsg);
$('#alert').show();
}
else{
$('#preview_modal').modal('show');
$('#preview_body').html(response.list);
}
}
});
});

});
</script>
</body>
</html>

```

Voter's list:

```
<?php include 'includes/session.php'; ?>
```

```
<?php include 'includes/header.php'; ?>
```

```
<body class="hold-transition skin-blue sidebar-mini">
```

```
<div class="wrapper">
```

```
<?php include 'includes/navbar.php'; ?>
```

```
<?php include 'includes/menubar.php'; ?>
```

```
<!-- Content Wrapper. Contains page content -->
```

```
<div class="content-wrapper">
```

```
<!-- Content Header (Page header) -->
```

<section class="content-header">

Votes


```
<ol class="breadcrumb">
```

- [*fa fa-dashboard*](#) Home

```
<li class="active">Votes</li>
```

</section>

```
<!-- Main content -->
```

<section class="content">

```
<?php
```

```
if(isset($_SESSION['error'])){
```

echo "

```
<div class='alert alert-danger alert-dismissible'>
```

The screenshot shows the 'Voters List' page of the Voting System. The table contains the following data:

Lastname	Firstname	Photo	Voters ID	Status
Ad	Adrian		01 1234567890123456	Voted
Li	Li		02 1234567890123456	Voted
Li	Li		03 1234567890123456	Voted
Ad	Adrian		04 1234567890123456	Voted
Adrian	Adrian		05 1234567890123456	Voted
Adrian	Adrian		06 1234567890123456	Voted
Adrian	Adrian		07 1234567890123456	Voted
Adrian	Adrian		08 1234567890123456	Voted
Adrian	Adrian		09 1234567890123456	Voted
Adrian	Adrian		10 1234567890123456	Voted

```
<button type='button' class='close' data-dismiss='alert' aria-hidden='true'>&times;</button>

<h4><i class='icon fa fa-warning'></i> Error!</h4>

".$_SESSION['error']."

</div>

";

unset($_SESSION['error']);

}

if(isset($_SESSION['success'])){\

echo "

<div class='alert alert-success alert-dismissible'>

<button type='button' class='close' data-dismiss='alert' aria-hidden='true'>&times;</button>

<h4><i class='icon fa fa-check'></i> Success!</h4>

".$_SESSION['success']."

</div>

";

unset($_SESSION['success']);

}

?>

<div class="row">

<div class="col-xs-12">

<div class="box">

<div class="box-header with-border">

<a href="#reset" data-toggle="modal" class="btn btn-danger btn-sm btn-flat"><i class="fa fa-
refresh"></i> Reset</a>

</div>

<div class="box-body">

<table id="example1" class="table table-bordered">

<thead>

<th class="hidden"></th>
```

```

        <th>Position</th>

        <th>Candidate</th>

        <th>Voter</th>

    </thead>

    <tbody>

        <?php

            $sql = "SELECT *, candidates.firstname AS canfirst, candidates.lastname AS canlast,
voters.firstname AS votfirst, voters.lastname AS votlast FROM votes LEFT JOIN positions ON
positions.id=votes.position_id LEFT JOIN candidates ON candidates.id=votes.candidate_id LEFT JOIN
voters ON voters.id=votes.voters_id ORDER BY positions.priority ASC";

            $query = $conn->query($sql);

            while($row = $query->fetch_assoc()){

                echo "

                <tr>

                    <td class='hidden'></td>

                    <td>".$row['description']. "</td>

                    <td>".$row['canfirst']. ' ' . $row['canlast']. "</td>

                    <td>".$row['votfirst']. ' ' . $row['votlast']. "</td>

                </tr>

                ";

            }

        ?>

    </tbody> </table> </div> </div> </div> </div> </section> </div>

    <?php include 'includes/footer.php'; ?>

    <?php include 'includes/votes_modal.php'; ?>

</div>

<?php include 'includes/scripts.php'; ?>

</body>

</html>

```

Candidates list:

```
<?php include  
'includes/session.php'; ?>
```

```
<?php include  
'includes/header.php'; ?>
```

```
<body class="hold-transition skin-blue  
sidebar-mini">
```

```
<div class="wrapper">
```

```
    <?php include  
    'includes/navbar.php'; ?>
```

```
    <?php include 'includes/menubar.php'; ?>
```

```
    <!-- Content Wrapper. Contains page content -->
```

```
    <div class="content-wrapper">
```

```
        <!-- Content Header (Page header) -->
```

```
        <section class="content-header">
```

```
            <h1>
```

```
            Candidates List
```

```
        </h1>
```

```
        <ol class="breadcrumb">
```

```
            <li><a href="#"><i class="fa fa-dashboard"></i> Home</a></li>
```

```
            <li class="active">Candidates</li>
```

```
        </ol>
```

```
    </section>
```

```
    <!-- Main content -->
```

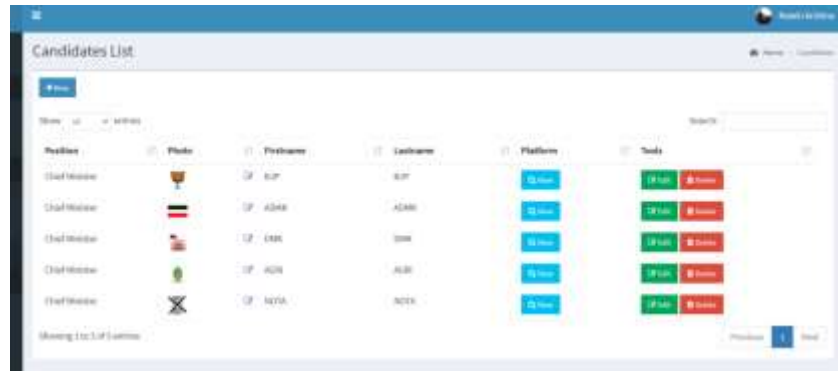
```
    <section class="content">
```

```
        <?php
```

```
        if(isset($_SESSION['error'])){
```

```
            echo "
```

```
            <div class='alert alert-danger alert-dismissible'>
```



```
<button type='button' class='close' data-dismiss='alert' aria-hidden='true'>&times;</button>

<h4><i class='icon fa fa-warning'></i> Error!</h4>

".$_SESSION['error']."

</div>

";

unset($_SESSION['error']);

}

if(isset($_SESSION['success'])){

echo "

<div class='alert alert-success alert-dismissible'>

<button type='button' class='close' data-dismiss='alert' aria-hidden='true'>&times;</button>

<h4><i class='icon fa fa-check'></i> Success!</h4>

".$_SESSION['success']."

</div>

";

unset($_SESSION['success']);

}

?>

<div class="row">

<div class="col-xs-12">

<div class="box">

<div class="box-header with-border">

<a href="#addnew" data-toggle="modal" class="btn btn-primary btn-sm btn-flat"><i class="fa fa-plus"></i> New</a>

</div>

<div class="box-body">

<table id="example1" class="table table-bordered">

<thead>

<th class="hidden"></th>
```

```

<th>Position</th>

<th>Photo</th>

<th>Firstname</th>

<th>Lastname</th>

<th>Platform</th>

<th>Tools</th>

</thead>

<tbody>

<?php

    $sql = "SELECT *, candidates.id AS canid FROM candidates LEFT JOIN positions ON
positions.id=candidates.position_id ORDER BY positions.priority ASC";

    $query = $conn->query($sql);

    while($row = $query->fetch_assoc()){

        $image = (!empty($row['photo'])) ? '../images/'.$row['photo'] : '../images/profile.jpg';

        echo "

        <tr>

            <td class='hidden'></td>

            <td>".$row['description'].</td>

            <td>

                <img src='".$image.'" width='30px' height='30px'>

                <a href='#edit_photo' data-toggle='modal' class='pull-right photo' data-
id='".$row['canid'].'"><span class='fa fa-edit'></span></a>

            </td>

            <td>".$row['firstname'].</td>

            <td>".$row['lastname'].</td>

            <td><a href='#platform' data-toggle='modal' class='btn btn-info btn-sm btn-flat platform'
data-id='".$row['canid'].'"><i class='fa fa-search'></i> View</a></td>

            <td>

                <button class='btn btn-success btn-sm edit btn-flat' data-id='".$row['canid'].'"><i
class='fa fa-edit'></i> Edit</button>

```

```
        <button class='btn btn-danger btn-sm delete btn-flat' data-id='".$row['canid']."'><i
class='fa fa-trash'></i> Delete</button>
```

```
    </td>
```

```
</tr>
```

```
    "; } ?> </tbody> </table> </div> </div> </div> </div> </section> </div>
```

```
<?php include 'includes/footer.php'; ?>
```

```
<?php include 'includes/candidates_modal.php'; ?>
```

```
</div>
```

```
<?php include 'includes/scripts.php'; ?>
```

```
<script>
```

```
$(function(){
```

```
    $(document).on('click', '.edit', function(e){
```

```
        e.preventDefault();
```

```
        $('#edit').modal('show');
```

```
        var id = $(this).data('id');
```

```
        getRow(id);
```

```
    });
```

```
    $(document).on('click', '.delete', function(e){
```

```
        e.preventDefault();
```

```
        $('#delete').modal('show');
```

```
        var id = $(this).data('id');
```

```
        getRow(id);
```

```
    });
```

```
    $(document).on('click', '.photo', function(e){
```

```
        e.preventDefault();
```

```
        var id = $(this).data('id');
```

```
        getRow(id); });
```

```
    $(document).on('click', '.platform', function(e){
```

```
        e.preventDefault();
```



```
var id = $(this).data('id');

getRow(id);

}); });

function getRow(id){

$.ajax({

    type: 'POST',

    url: 'candidates_row.php',

    data: {id:id},

    dataType: 'json',

    success: function(response){

        $('#id').val(response.canid);

        $('#edit_firstname').val(response.firstname);

        $('#edit_lastname').val(response.lastname);

        $('#posselect').val(response.position_id).html(response.description);

        $('#edit_platform').val(response.platform);

        $('#fullname').html(response.firstname+' '+response.lastname);

        $('#desc').html(response.platform);

    }

});

}

</script>

</body>

</html>
```

Ballot Position:

```
<?php include 'includes/session.php'; ?>

<?php include 'includes/header.php'; ?>

<body class="hold-transition skin-blue sidebar-mini">

<div class="wrapper">

<?php include 'includes/nawin.php'; ?>

<?php include 'includes/menubar.php'; ?>

<!-- Content Wrapper. Contains page content -->

<div class="content-wrapper">

  <!-- Content Header (Page header) -->

  <section class="content-header">

    <h1>

      Ballot Position

    </h1>

    <ol class="breadcrumb">

      <li><a href="#"><i class="fa fa-dashboard"></i> Home</a></li>

      <li class="active">Ballot Preview</li>

    </ol>

  </section>

  <!-- Main content -->

  <section class="content">

    <?php

      if(isset($_SESSION['error'])){

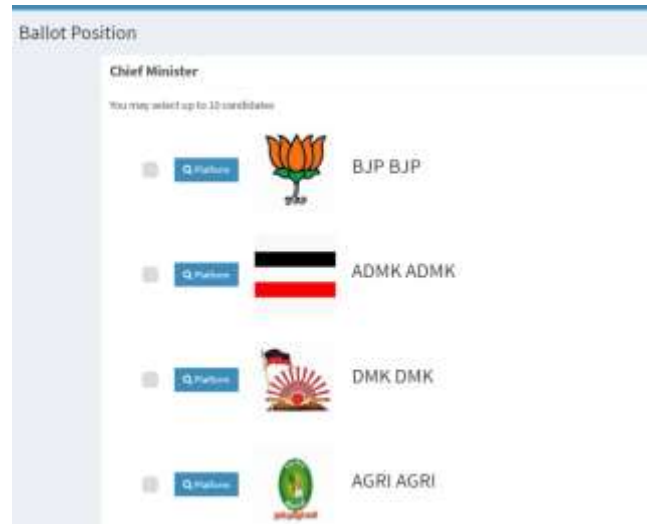
        echo "

          <div class='alert alert-danger alert-dismissible'>

            <button type='button' class='close' data-dismiss='alert' aria-hidden='true'>&times;</button>

            <h4><i class='icon fa fa-warning'></i> Error!</h4>


```



```

        ".$_SESSION['error'].
    </div>

    ";

    unset($_SESSION['error']);
}

if(isset($_SESSION['success'])) {
    echo "

    <div class='alert alert-success alert-dismissible'>

        <button type='button' class='close' data-dismiss='alert' aria-hidden='true'>&times;</button>

        <h4><i class='icon fa fa-check'></i> Success!</h4>

        ".$_SESSION['success'].
    </div>

    ";

    unset($_SESSION['success']);
}

?>

<div class="row">

    <div class="col-xs-10 col-xs-offset-1" id="content">

        </div>

    </div>

</section>

</div>

<?php include 'includes/footer.php'; ?>

</div>

<?php include 'includes/scripts.php'; ?>

<script>

$(function(){

    fetch();

```

```
$(document).on('click', '.reset', function(e){  
    e.preventDefault();  
    var desc = $(this).data('desc');  
    $('.'+desc).iCheck('uncheck');  
});
```

```
$(document).on('click', '.moveup', function(e){  
    e.preventDefault();  
    var id = $(this).data('id');  
    $('#'+id).animate({  
        'marginTop' : "-300px"  
    });
```

```
$.ajax({  
    type: 'POST',  
    url: 'ballot_up.php',  
    data:{id:id},  
    dataType: 'json',  
    success: function(response){  
        if(!response.error){  
            fetch();  
        }    } });
```

```
$(document).on('click', '.movedown', function(e){  
    e.preventDefault();  
    var id = $(this).data('id');  
    $('#'+id).animate({  
        'marginTop' : "+300px"  
    });
```

```
$.ajax({
```

```
type: 'POST',  
url: 'ballot_down.php',  
data:{id:id},  
dataType: 'json',  
success: function(response){  
    if(!response.error){  
        fetch();  
    } } }); }); });
```

```
function fetch(){  
    $.ajax({  
        type: 'POST',  
        url: 'ballot_fetch.php',  
        dataType: 'json',  
        success: function(response){  
            $('#content').html(response).iCheck({checkboxClass: 'icheckbox_flat-green',radioClass: 'iradio_flat-green'});  
        } }); }  
</script>  
</body>  
</html>
```

Votes list:

```
<?php include 'includes/session.php'; ?>
```

```
<?php include 'includes/header.php'; ?>
```

<body class="hold-transition skin-blue sidebar-mini">

```
<div class="wrapper">
```

```
<?php include 'includes/navbar.php'; ?>
```

```
<?php include 'includes/menubar.php'; ?>
```

```
<!-- Content Wrapper. Contains page content -->
```

```
<div class="content-wrapper">
```

```
<!-- Content Header (Page header) -->
```

<section class="content-header">

Votes


```
<ol class="breadcrumb">
```

- [*fa fa-dashboard*](#) Home

```
<li class="active">Votes</li>
```

</section>

```
<!-- Main content -->
```

<section class="content">

```
<?php
```

```
if(isset($_SESSION['error'])){
```

echo "

```
<div class='alert alert-danger alert-dismissible'>
```

[illegible]


```

        <th>Position</th>

        <th>Candidate</th>

        <th>Voter</th>

    </thead>

    <tbody>

        <?php

            $sql = "SELECT *, candidates.firstname AS canfirst, candidates.lastname AS canlast,
voters.firstname AS votfirst, voters.lastname AS votlast FROM votes LEFT JOIN positions ON
positions.id=votes.position_id LEFT JOIN candidates ON candidates.id=votes.candidate_id LEFT JOIN
voters ON voters.id=votes.voters_id ORDER BY positions.priority ASC";

            $query = $conn->query($sql);

            while($row = $query->fetch_assoc()){

                echo "

                <tr>

                    <td class='hidden'></td>

                    <td>".$row['description']. "</td>

                    <td>".$row['canfirst']. ' ' . $row['canlast']. "</td>

                    <td>".$row['votfirst']. ' ' . $row['votlast']. "</td>

                </tr> "; } ?>

            </tbody>

        </table> </div> </div> </div> </div> </section> </div>

        <?php include 'includes/footer.php'; ?>

        <?php include 'includes/votes_modal.php'; ?>

    </div>

    <?php include 'includes/scripts.php'; ?>

</body>

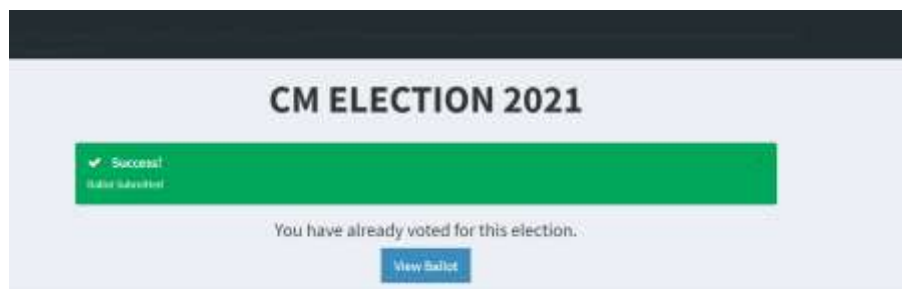
</html>

```


Election page:



Vote Success:



Voter can vote only once:



Results of Election:

