

ONLINE E-VOTING SYSTEM

Submitted in partial fulfillment of the requirements for the award of
Bachelor of science Degree in Computer Science

By

KISHOORE.S

(Reg. No.38290040)

VARUN KUMAR.S

(Reg. No. 38290098)



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

**SATHYABAMA
INSTITUTE OF SCIENCE AND TECHNOLOGY
JEPPIAAR NAGAR, RAJIV GANDHI SALAI,
CHENNAI – 600119, TAMILNADU**

APRIL 2020



SATHYABAMA

INSTITUTE OF SCIENCE AND TECHNOLOGY
(DEEMED TO BE UNIVERSITY)

Accredited with Grade "A" by NAAC

(Established under Section 3 of UGC Act, 1956)

JEPPIAAR NAGAR, RAJIV GANDHI SALAI, CHENNAI- 600119

www.sathyabama.ac.in



DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

BONAFIDE CERTIFICATE

This is to certify that this Project Report is the bonafide work of **KISHOORE S (Reg. No. 38290040)** and **VARUN KUMAR S (Reg. No. 38290098)** who carried out the project entitled **"ONLINE E-VOTING SYSTEM"** under my supervision from November 2020 to March 2021.

Internal Guide

Dr. J . ALBERT MAYAN M.E., Ph.D.,

Head of the Department

Submitted for Viva voce Examination held on _

Internal Examiner

External Examiner

DECLARATION

I, **KISHOORE S (Reg. No. 38290040)** hereby declare that the Project Report entitled "**ONLINE E-VOTING SYSTEM**" done by me under the guidance of **Dr. S. DHAMODARAN , M.E.,Ph.D.,** is submitted in partial fulfillment of the requirements for the award of Bachelor of Science degree in Computer Science.

DATE:

PLACE: CHENNAI

SIGNATURE OF THE CANDIDATE

ACKNOWLEDGENT

I am pleased to acknowledge my sincere thanks to Board of Management of **SATHYABAMA** for their kind encouragement in doing this project and for completing it successfully. I am grateful to them.

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

I would like to express my sincere and deep sense of gratitude to my Project Guide **Dr. S. DHAMODARAN, M.E., Ph.D.**, for his valuable guidance, suggestions and constant encouragement paved way for the successful completion of my project work.

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

ABSTRACT

A manual voting process is excessively time consuming and takes a lot of steps to finish the vote. With the rapid development information technology, an electronic voting system (E-voting system) is more convenient to vote for someone from a list of candidates shown on the computer screen. However, security is a big concern for electronic voting. In this project, we build a secure voting system using PHP and SQL. PHP is a kind of public key cryptography that has the SQL database property that can be exploited to calculate the sum of votes without revealing to the system which vote is voted for which vote is voted for which candidate. The voting system design has a simple interface that clients can use easily. The system allows the voter to use an ID and PIN to vote for his/her favorable candidate. This online voting system is able to save time for users and it is more reliable than the traditional system.

TABLE OF CONTENTS

Chapter No..	TITLE	Page No.
	ABSTRACT	2
	LIST OF ABBREVIATION	4
1	INTRODUCTION	6
	1.1 About The Project	7
	1.2 System Specification	8
	1.2.1 Hardware Configuration	8
	1.2.2 Software Specification	8
	1.2.3 Software Description	9
2	SYSTEM STUDY	14
	2.1 Existing System	14
	2.1.1 Drawbacks	14
	2.2 Proposed System	15
	2.2.1 Features	15
3	SYSTEM DESIGN AND DEVELOPMENT	16
	3.1 Input Design	16
	3.2 Output Design	17
	3.3 System Development	18
4	TESTING AND IMPLEMENTATION	20
5	REFERENCES	23
	APPENDICES	24
	A. Tables	25
	B. Data Flow Diagram	28

C. Coding	29
-----------	----

Sample source code

D.Sample Screens	39
-------------------------	-----------

D1.Login form	39
---------------	----

D2.Registered form	40
--------------------	----

D3.Category	41
-------------	----

D4.Candidate form	41-42
-------------------	-------

D5.Election List	44
------------------	----

LIST OF ABBREVIATIONS

PHP	–	Hyper Text Preprocessor
SQL	–	Structured Query Language
HTML	–	Hypertext Markup Language
XML	–	Extensible Markup Language
RAM	–	Random Access Memory

1.

INTRODUCTION

The election process currently in College is a very tough process. The current existing election voting System is conducted manually where the Voter has to Visit to Booths or designed restricted area to Vote, so there is wastage of time and leads to delay of counting of votes and because of the limited voting hours given, many people are left out without voting their preferred leaders. It therefore leaves a very big margin of people who have not voted and because of that the leaders deserving some sits tend to lose them.

In many a times in College election, majority of students do not vote because of the long queues while voting because they perceived it as a wastage of time. The Election Voting System also entails that, the voters have to be manually registered. Also Vote counting has to be done manually by counting the votes cast. All the Information of the Voter or Candidate is to be filling in manually and a voter must be present in campus to give his/her Vote during the day of voting. Because of all these drawbacks, I have come with an idea of developing an Online Module for Students Association of Leaders elections. This module will be simple, attractive and easy to use. It reduces manual efforts and bulk of information can be handled easily.

1.1 ABOUT THE PROJECT

Online Voting is a web-based voting system that will help you manage your elections easily and securely. This voting system can be used for casting votes during the elections held in colleges, etc. In this system the voter does not have to go to the polling booth to cast their vote. They can use their personal computer to cast their votes. There is a database which is maintained in which all the name of the voters with their complete information is stored. The System Administrator registers the voters by simply filling a registration form to register the voters.

Online E-voting System would have Candidate registration, document verification, auto-generated User ID and pass for candidate and Voters. Admin Login which will be handled by Election Commission .Candidate Login which will be handled By Candidate, Voters will get Unique ID and Password, Using which they can vote for a Candidate only once per Election. The project is beneficial for Election Commission, Voters as the can get to know the candidate background and choose wisely, and even for Candidate. The software system allows the Candidate to login in to their profiles and upload all their details including their previous milestone onto the system. The admin can check each Candidate details and verify the documents, only after verifying Candidate's ID and Password will be generated, and can remove faulty accounts.

The software system allows Voters to view a list of Candidates in their area. The admin has overall rights over the system and can moderate and delete any details not pertaining to Election Rules. After voting procedure is over, the votes counting procedure starts in manual manner. The voter cannot change the candidate at the time of voting. The project "**ONLINE E- VOTING**" is aimed to computerize the task involved in the voting process. This project deals with the automation of various polling activities.

1.2 SYSTEM SPECIFICATION

1.2.1 Hardware Configuration

- Processor : I3
- Hard Disk Capacity : 400 GB
- RAM : 4.00 GB
- Monitor : 15" Color Monitor
- Keyboard : 102 Keys
- Mouse : 2 Buttons

1.2.2 Software Specification

- Operating System : Windows 7 Ultimate
- Front-End : PHP
- Back-End : MYSQL

1.2.3 Software Description

PHP is the web development language written by and for web developers. PHP stands for PHP: Hypertext Pre-processor. The product was originally named Personal Home Page Tools, and many people still think that's what the acronym stands for, but as it expanded in scope, a new and more appropriate name was selected by community vote. PHP is currently in its sixth major rewrite, called PHP6 or just plain PHP.

INTRODUCTION OF PHP

PHP is a server-side scripting language, usually used to create web applications in combination with a web server, such as Apache. PHP can also be used to create command-line scripts akin to Perl or shell scripts, but such use is much less common than PHP's use as a web language. Strictly speaking, PHP has nothing to do with layout, events, on-the-fly Document Object Model (DOM) manipulation, or really anything about the look and feel of a web page. In fact, most of what PHP does is invisible to the end user. Someone looking at a PHP page will not necessarily be able to tell that it was not written purely in Hypertext Markup Language (HTML), because the result of PHP is usually HTML

When compared to many other programming languages, PHP makes it easy to develop powerful web application quickly (this is a blessing and a curse). Many of the most useful specific functions (such as those for opening a connection to an Oracle database or fetching email from an Internet Message Access Protocol [IMAP] server) are predefined. Most advanced PHP users (including most of the development team members) are diehard hand coders.

PHP & MYSQL are also both stable in the sense of feature stability. Their respective development teams have thus far enjoyed a clear vision of their project and refused to be distracted by every new fad and ill-through-out user demand that comes along. Much of the effort goes into incremental performance improvements, communicating with more major database, or adding better OOP support.

PHP STRENGTHS

PHP has much strength, including the following:

- Performance
- Scalability
- Interfaces to many different database system
- Build-in libraries for many common web tasks
- Low cost
- Ease of learning and use
- Strong object oriented support
- Portability
- Flexibility of development approach
- Availability of source code
- Availability of support and documentation

NEW IN PHP

Major new features in PHP are as follows:

- Better object oriented support build around a completely new object model
- Exception for scalable, maintainable error handling
- Simple XML for easy handling of XML data

USES OF PHP

PHP is a general-purpose scripting language that is especially suited to server-side web development, in which case PHP generally runs on a web server. Any PHP code in a requested file is executed by the PHP runtime, usually to create dynamic web page content or dynamic images used on websites or elsewhere.

FEATURES IN PHP CLASSIFIEDS

TOP FEATURES

- Open Source (Only one file encrypted)
- Nice layout from start.
- Unlimited level of categories.
- Unique Extra Fields that is also sort able.
- Large user community.
- Regions Module is included (using Ajax).

ADMIN AREA

- Unlimited extra field's creation.
- Over 300 different settings in settings-area.
- User friendly installer
- Set up searchable fields from admin area
- Backup tool

GENERAL FEATURES

- Print functionality removes navigation.
- Picture gallery with only one image per user.
- Sort on all columns.
- Easy renew function
- Advanced search, different search-page for different categories (template fields)
- Translated to 11 languages

INTRODUCTION TO MYSQL

MYSQL is an open source, SQL relational database management system (RDBMS) that is free for many uses (more detail on that later). Each in its history, MYSQL occasionally faced opposition because of its lack of support for some core SQL constructs such as sub selects and foreign keys. Ultimately, however MYSQL found a broad, enthusiastic user base for its liberal licensing terms, perky performance, and ease of use. Its acceptance was aided in part by the wide variety of other technologies such as PHP, Perl, Python, and the like that have encouraged its use through stable, well-documented modules and extensions. Databases are generally useful, perhaps the most consistently useful family of software products (the “killer product”) in modern computing. Like many competing products, both free and commercial, MYSQL isn’t a database until give it some structure and form.

MYSQL STRENGTHS

MYSQL has much strength, including the following

- High performance
- Low cost
- Ease of configuring and learning
- Portability
- Availability of source code
- Availability of support

NEW IN MYSQL

Major changes introduced for MYSQL includes

- View
- Stored procedures
- Basic trigger support
- Cursor support

USES OF PHP AND MYSQL

When setting out to build a website and has different products

- Hardware for the web server
- An operating system
- A database management system
- A programming or scripting language

Some of these are dependent on the other, for example not all operating system run on all hardware not all web servers support all programming language.

The majority of PHP code can be written to be portable between operating system and web servers there are some PHP functions that specifically relate to the file system there are operating system dependent.

2.

SYSTEM STUDY

2.1 EXISTING SYSTEM

The present voting system in our institution is done using forms. It is difficult for the voters to change the Candidate at the time of voting. It is difficult to track the registered voters and the absentee voting. As all the operations are done manually, it requires huge time for counting and publishing the result.

In the present system it is difficult to ensure privacy, security and accuracy. At the time of voting, if a voter makes an incorrect choice, it is difficult to correct it. There is no guarantee that all votes are counted. The process needs massive paper handling and create security problem.

2.1.1 DRAWBACKS

- Everything is maintained in papers. Requires manual counting.
- Difficulty to enforce security during voting and counting
- Probabilities for errors occur while counting number of votes.
- Less reliability and no guarantee vote was counted.

2.2 PROPOSED SYSTEM

The proposed system is designed considering all the drawbacks listed in the existing system. The system supports two groups of users, administrator and voters. Administrator has the permission to create the contestant database and the voter database. Based on the configuration, the voter can use the system for polling.

The system also maintains the voting status for each and every voter. Once voting is completed by the user, the system will not allow again voting for that polling. If a voter makes an incorrect choice, the system allows correcting it. The system can also be configured to announce the results based on timer setting.

2.1.2FEATURES

- User friendly interface for polling.
- Creates a sense of security among voters
- Accurate results can be produced in quick time
- Avoid duplicate votes
- More reliable than the existing system
- Only registered voters are allowed to vote

3.SYSTEM DESIGN AND DEVELOPMENT

3.1 INPUT DESIGN

Input design is a part of overall system design, which requires careful attention. Input of data is designed as user friendly and easier. Input design is the process of converting user- originated inputs to a computer-based format. The objective of the input design is to create an input layout that is easy to follow and prevent errors.

The process information is stored in for future use. The user should be search in his related information only. The goal of designing input data is to make error free and user-friendly.

The objectives in input design are:

- To produce a cost effective method of input.
- To achieve the highest possible level of accuracy.
- To ensure that the input is acceptable and

understood. The following are some description of the screen which gets input from the user

LOGIN DETAILS

The login form enables both user and admin of the system. In login form, using username and password the admin or the user can enter to the Home page.

REGISTRATION DETAILS

The registration form is used to store the user details like Username, Password, Confirm Password and Email. By using the registered details the user can login with the help of Username and Password.

CANDIDATE DETAILS

The Candidate form is used to store the candidate details like Candidate First name, Last name, Candidate Position and Party name, Gender, Email, Major, Percentage, Candidate Image and Elect.

VOTER DETAILS

The Voter form is used to store details of the voter details like Voter Id, Voter Name

VOTING DETAILS

In this form, the voter has to choose the candidate which is visible under the table. The table has candidate information like Candidate Id, Candidate Position and Party name of the Candidate. This form has Voter Id and Candidate Id. By clicking the vote button after selecting the Voter Id and Candidate Id then it stores the candidate details like Candidate Id, Candidate position and Party name in voter table but the counts of vote in Candidate table.

3.2 OUTPUT DESIGN

Output design generally refers to the results and information that are generated by the system for many end-users; output is the main reason for developing the system and the basis on which they evaluate the usefulness of the application. In any system, the output design determines the input to be given to the application.

The output is designed in such a way that it is attractive, convenient and informative. Forms are designed in PH with various features, which make the console output more pleasing.

As the outputs are the most important sources of information to the users, better design should improve the system's relationships with us and also will help in decision-making. Form design elaborates the way output is presented and the layout available for capturing information.

CANDIDATE DETAILS

In this form, admin can view the candidate details such as Candidate Id, Candidate Name, Position and Party name, Gender, DOB, Email, Major, Percentage and Candidate Image.

VOTER DETAILS

In this form, admin can view the voter details such as Id, Voter Id, Candidate Id, Voter name, Position and Party name they selected.

RESULT ANALYSE

In this form, admin can view the entire candidate vote analyses include Candidate Id, Candidate name, Party name of the Candidate and the counts of vote .

RESULT FORM

In this form, user can view the selected Candidate Id, Candidate Position and Party name and Candidate Image after the pre-defined date and time.

3.3 SYSTEM DEVELOPMENT

DESCRIPTION OF MODULES

The project is designed modularly such a way that any new feature or enhancements can be easily added to the system.

The online E-voting System consist the following modules.

1. Admin Module
2. Configuration Module
3. Voting Module
4. Timer Module
5. Report Module

Admin Module

This module takes care of proper authentication to use the system. There are two groups of users allowed, administrator and voter. The administrator has the privilege to manage the database of candidates and Voters.

Configuration Module

The administrator has the rights to configure the candidates of the election. The administrator also has to create the voter database. The normal users for this system will only have the option to vote.

Voting Module

This module takes care of the whole operation of polling. This module shows the list of candidates contesting the election. User will be given options to change the decision at the time of voting. This also ensures that there will not be any duplicate votes polled into the system.

Timer Module

The result of the polling can be viewed only after the configured timer interval. This module takes care of declaring the result after the user configured time. The system will show denial message if user attempt to see the result before the configured time.

Report Module

This presents the report of the election. It displays candidate names, candidate id, number votes, etc. in well organized format. The reports can also be taken candidate wise.

4 TESTING AND IMPLEMENTATION

SYSTEM TESTING

Testing is an integral part of any system development life cycle. Insufficient and untested applications may trend to crash and result in loss of economic and manpower investment besides user's dissatisfaction and downfall of reputation. Software testing can be looked upon as one among the many process, an organization performs, and that provides the last opportunity to correct any flaws in the developed system. Software testing includes selecting test data that have more probability of giving errors.

The first step in system testing is to develop a plan that tests all aspects of the system. Completeness, correctness, reliability and maintainability of the software are to be tested for the best quality assurance that the system meets the specification and requirements for its intended use and performance. System testing is the most useful practical process of executing a program with the implicit intention of finding errors that make the program fails.

System testing is done in three phases

- Unit testing
- Integration testing
- Validation testing

UNIT TESTING

The procedure level testing is made first. By giving improper inputs, the errors occurred are noted and eliminated. Then the web form level testing is made. For example, storage of data to the table in the correct manner.

In the organization as well as login form, the zero length username and password are given and checked. Also the duplicate username is given and checked. In the login button will send data to the server only if the client side validations are made.

INTEGRATION TESTING

Testing is done for each module. After testing all the modules, the modules are integrated and testing of the final system is done with the test data, specially designed to show that the system will operate successfully in all its aspects conditions. Thus the system testing is a confirmation that all is correct and an opportunity to show the user that the system works.

In this project, the migration between each and every pages are tested successfully. This testing process continued until satisfies the user requirement by integrate and test the whole project. For example, after click the login button the home page must be open and database connectivity also tested.

VALIDATION TESTING

The final step involves Validation testing, which determines whether the software function as the user expected. The end-user rather than the system developer conduct this test most software developers as a process called “Alpha and Beta Testing” to uncover that only the end user seems able to find.

The compilation of the entire project is based on the full satisfaction of the end users. In the project, validation testing is made in various forms. In login entry form, different validations are made to make sure the user enters the valid input.

SYSTEM IMPLEMENTATION

Implementation is process of converting a new or revised system design into an operational one. The first task is implementation planning that is deciding on the methods and time scale to be adopted. The proper implementation involves conversion of existing clerical files to computer media and hence these files as they are get converted. Then the actual changeover from the existing system to the new system takes place.

The changeover plays a vital role, which checks the developed tool for the following requirements, and then only the developed tool will be accepted by the users. The software has been checked with sample data. The changes being made are as per the user requirements and will run in parallel with the manual system to find out any discrepancies. The users also have been apprised of the ways of handling the software, as a part of training the user personnel.

The implementation is the final stage and it is an important phase. It involves the individual programming system testing, user training and the operational running of developed proposed system that constitutes the application subsystems. On major task of preparing for implementation is education of users, which would really have taken place much earlier in the project when were being involved in the investigation and design work. The implementation phase of software development is concerned with translating design specifications into source code. The user tests the developed system and changes are made according to their needs.

5.CONCLUSION

Online Election systems have many advantages over the traditional voting system. Some of these advantages are less cost, faster generation results, easy accessibility, accuracy, and low risk of human and mechanical errors. It is very difficult to develop e-voting system which can allow security and privacy on the high level. Future development focused to design a system which can be easy to use and will provide security and privacy of votes on acceptable level by proper authentication and processing section. By online voting system percentage of voting is increases and cost and time of voting process is decreases. It is easy to use and it is less time consuming. It is very easy to debug.

BIBLIOGRAPHY

TEXTBOOK REFERENCES

1. Steve Suehring, Tim Converse and Joyce Park, "PHP6 and MySQL bible".
2. Somerville, "Software Engineering", 5th Editions.
3. Luke Welling and Laura Thomson, "PHP and MySQL Web Development-developer Library".
4. Robin Nixon, "Learning PHP, MySQL & JavaScript: with jQuery, CSS & HTML5".

Websites Refer

1. www.php.net
2. www.W3Schools.com
3. www.QuoteProject.com
4. www.go4expert.com
5. www.allprojectreports.com

APPENDICES

A.TABLES

Table Name : Admin

Description : Used to login into the application

Primary Key : Username

Field Name	Data Type	Description
Username	Varchar(30)	Admin username
Password	Varchar(30)	Admin password

Table Name : User

Description : Used to store registered details of user

Primary Key : UserId

Field Name	Data Type	Description
Name	Varchar(50)	Name
Username	Varchar(50)	User Name
Password	Varchar(50)	User password

Table Name : Candidate

Description : Used to store candidate details

Primary Key : CandidateId

Field Name	Data Type	Description
Cfname	Varchar(50)	Candidate first name
Clname	Varchar(50)	Candidate last name
Cposition	Varchar(50)	Candidate position

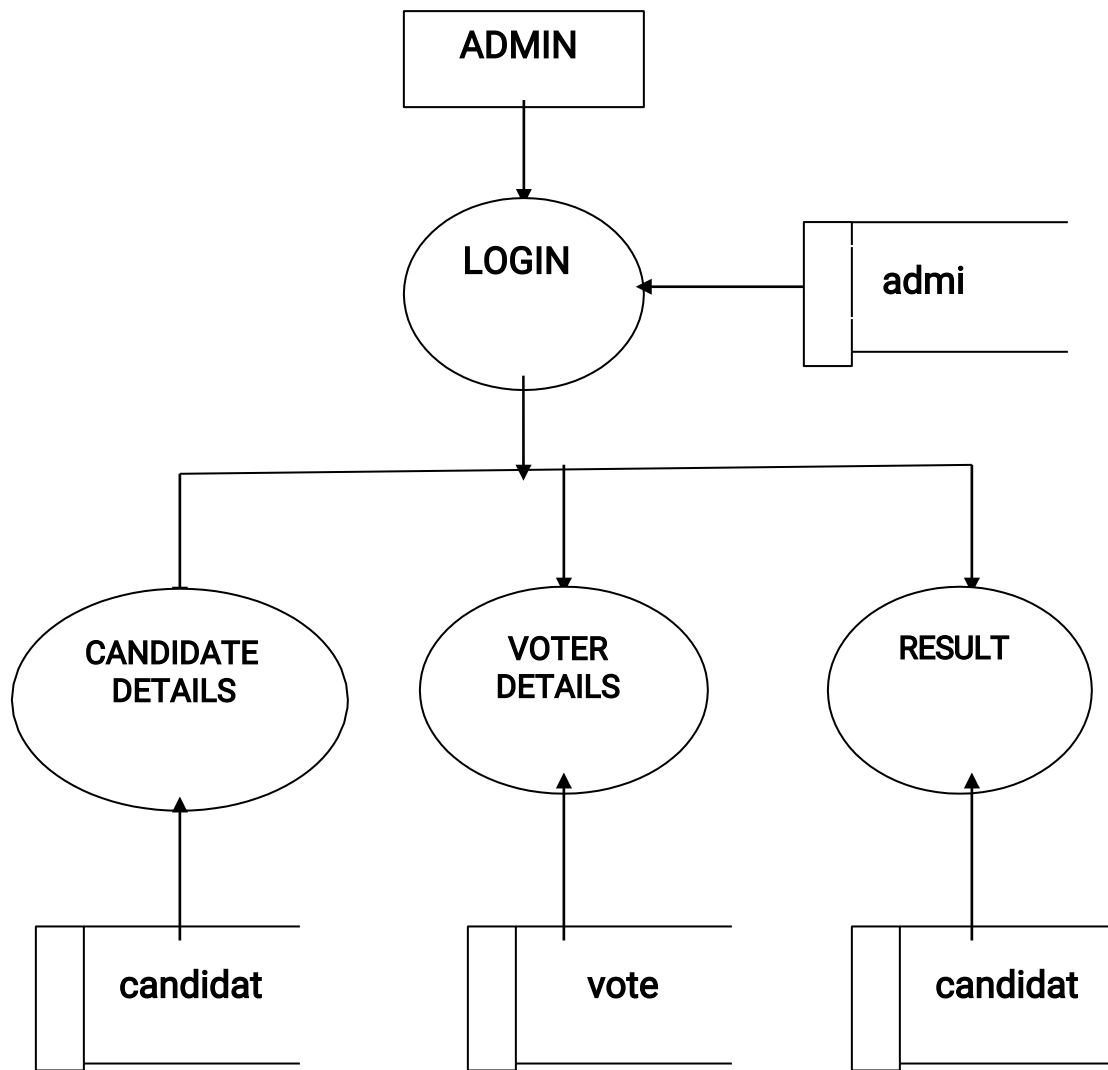
Table Name : Voter

Description : Used to store voter details

Primary Key : Id

Field Name	Data Type	Description
VoterId	Varchar(20)	Voter identification number
Votername	Varchar(50)	Name of Voter

B.DATA FLOW DIAGRAM



C.CODING

SAMPLE SOURCE CODE

```
<?php
```

```
/*
```

EDIT.PHP

Allows user to edit specific entry in database

```
*/
```

```
// creates the edit record form
```

```
// since this form is used multiple times in this file, I have made it a function that  
is easily reusable
```

```
functionrenderForm( $candidateID,$studnum, $fname, $lname, $gender, $course, $yearlvl,  
$position, $partylist, $image, $error)
```

```
{
```

```
?>
```

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01//EN"  
"http://www.w3.org/TR/html4/strict.dtd">
```

```
<html>
```

```
<head>
```

```
Edit Record</title>
```

```
</head>
```

```

<body>

<?php

// if there are any errors, display

them if ($error != "")

{

echo '<div style="padding:4px; border:1px solid red; color:red;">'. $error. '</div>';

}

?>

<table bgcolor="#bbbbbb" width="50%" border="0" cellspacing="10" cellpadding="5"
valign="center" align = "center">

<tr>

<td width="25%" valign="top" align="center">

<form action="" method="post" onSubmit="return confirm('Are you sure you want
to save changes?')">

<input type="hidden" name="candidateID" value="<?php echo $candidateID; ?>"/></td>

<div>

<tr>

<td><strong>Candidate ID:</strong><?php echo $candidateID; ?></td>

```

</tr>

<tr>

<td>Student Number: *</td>

</tr>

<tr>

<td><input type="text" name="studnum" value="<?php echo \$studnum; ?>"/></td>

</tr>

<tr>

<td>First Name: *</td>

</tr>

<tr>

<td><input type="text" name="fname" value="<?php echo \$fname; ?>"/></td>

</tr>

<tr>

<td>Last Name: *</td>

<tr>

<td><input type="text" name="lname" value="<?php echo \$lname; ?>"/></td>

</tr>

```
<tr>
```

```
<td><strong>Gender: *</strong></td>
```

```
</tr>
```

```
<tr>
```

```
<td><select name="gender">
```

```
<option value = "Male" <?php if($gender == "Male") { echo "selected"; }?>>Male</option>
```

```
<option value = "Female" <?php if($gender == "Female") { echo "selected"; }?>>Female</option>
```

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

```
<tr>
```

```
<td><strong>Course: *</strong></td>
```

```
</tr>
```

```
<tr>
```

```
<td><select name="course">
```

```
<option value = "Bachelor of Arts in History" <?php if($course == "Bachelor of Arts in History") { echo "selected"; }?>>Bachelor of Arts in History</option>
```

```
<option value = "Bachelor of Arts in Economics" <?php if($course == "Bachelor of arts in Economics") { echo "selected"; }?>>Bachelor of Arts in Economics</option>
```

```
<option value = "Bachelor of Arts in Psychology" <?php if($course == "Bachelor of Arts in Psychology") { echo "selected"; }?>>Bachelor of Arts in Psychology</option>
```

```
<option value = "Bachelor of Arts in Political Science" <?php if($course == "Bachelor of Arts in Political Science") { echo "selected"; }?>>Bachelor of Arts in Political Science</option>
```

```
<option value = "Bachelor of Arts in Communication" <?php if($course == "Bachelor of Arts in Communication") { echo "selected"; }?>>Bachelor of Arts in Communication</option>
```

```
<option value = "Bachelor of Arts in Mathematics" <?php if($course == "Bachelor of Arts in Mathematics") { echo "selected"; }?>>Bachelor of Arts in Mathematics</option>
```

```
<option value = "Bachelor of Arts in Social Work" <?php if($course == "Bachelor of Arts in Social Work") { echo "selected"; }?>>Bachelor of Arts in Social Work</option>
```

```
<option value = "Bachelor of Arts in Business Administration" <?php if($course == "Bachelor of Arts in Business Administration") { echo "selected"; }?>>Bachelor of Science in Business Administration</option>
```

```
<option value = "Bachelor of Science in Accountancy" <?php if($course == "Bachelor of Science in Accountancy") { echo "selected"; }?>>Bachelor of Science in Accountancy</option>
```

```
<option value = "Bachelor of Elementary Education" <?php if($course == "Bachelor of Elementary Education") { echo "selected"; }?>>Bachelor of Elementary Education</option>
```

```
<option value = "Bachelor of Secondary Education" <?php if($course == "Bachelor of Secondary Education") { echo "selected"; }?>>Bachelor of Secondary Education</option>
```

```
<option value = "Bachelor of Science in Library and Information Science" <?php if($course == "Bachelor of Science in Library and Information Science") { echo "selected"; }?>>Bachelor of Science in Library and Information Science</option>
```

```
<option value = "Bachelor of Science in Civil Engineering" <?php if($course == "Bachelor of Science in Civil Engineering") { echo "selected"; }?>>Bachelor of Science in Civil Engineering</option>
```

```
<option value = "Bachelor of Science in Computer Engineering" <?php if($course == "Bachelor of Science in Computer Engineering") { echo "selected"; }?>>Bachelor of Science in Computer Engineering</option>
```

```
<option value = "Bachelor of Science in Electrical Engineering" <?php if($course ==  
"Bachelor of Science in Electrical Engineering") { echo "selected"; }?>>Bachelor of  
Science in Electrical Engineering</option>
```

```
<option value = "Bachelor of Science in Information Technology" <?php if($course ==  
"Bachelor of Science in Information Technology") { echo "selected"; }?>>Bachelor of  
Science in Information Technology</option>
```

```
<option value = "Bachelor of Science in Computer Science" <?php if($course == "Bachelor  
of Science in Computer Science") { echo "selected"; }?>>Bachelor of Science in Computer  
Science</option>
```

```
<option value = "Bachelor of Science in Information Systems" <?php if($course ==  
"Bachelor of Science in Information Systems") { echo "selected"; }?>>Bachelor of Science in  
Information Systems</option>
```

```
<option value = "Two-Year Associate in Computer Technology" <?php if($course ==  
"Two- Year Associate in Computer Technology") { echo "selected"; }?>>Two-Year  
Associate in Computer Technology</option>
```

```
<option value = "Two-Year Refrigeration &Airconditioning Technician Course"  
<?php if($course == "Two-Year Refrigeration &Airconditioning Technician  
Course") { echo "selected"; }?>>Two-Year Refrigeration &Airconditioning  
Technician Course</option>
```

```
<tr>
```

```
<td><strong>Position: *</strong></td>
```

```
</tr>
```

```
<tr>
```

```
<td><select name="position">
```

```
<tr>
```

```
<td><strong>Image: *</strong></td>
```

```
</body>
```

```

</html>

<?php

}

// connect to the

database

include('dbcon.php');

// check if the form has been submitted. If it has, process the form and save it to

the database if (isset($_POST['submit']))

{

// confirm that the 'id' value is a valid integer before getting the

form data if (is_numeric($_POST['candidateID']))

{

// get form data, making sure it is valid

$candidateID = $_POST['candidateID'];

$studnum = mysql_real_escape_string(htmlspecialchars($_POST['studnum']));

$name = mysql_real_escape_string(htmlspecialchars($_POST['fname']));

$lname = mysql_real_escape_string(htmlspecialchars($_POST['lname']));

$gender = mysql_real_escape_string(htmlspecialchars($_POST['gender']));

```

```

$course = mysql_real_escape_string(htmlspecialchars($_POST['course']));

$yearlvl = mysql_real_escape_string(htmlspecialchars($_POST['yearlvl']));

$position = mysql_real_escape_string(htmlspecialchars($_POST['position']));

$partylist = mysql_real_escape_string(htmlspecialchars($_POST['partylist']));

$image = mysql_real_escape_string(htmlspecialchars($_POST['image']));

// check that firstname/lastname fields are both filled in

if ($candidateID == " || $studnum == " || $fname == " || $lname == " || $gender == " ||
$course == " || $yearlvl == " || $position == " || $partylist == " || $image == ")

{

// generate error message

$error = 'ERROR: Please fill in all required fields!';

//error, display form

// check that the 'id' matches up with a row in the

database if($row)

{
// get data from db

$candidateID = $row['candidateID'];

$studnum = $row['studnum'];

$fname = $row['fname'];

```

```
$lname = $row['lname'];
```

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

```
$course = $row['course'];
```

```
$yearlvl = $row['yearlvl'];
```

```
$position = $row['position'];
```

```
$partylist = $row['partylist'];
```

```
$image = $row['image'];
```

```
// show form
```

```
renderForm($candidateID,$studnum,$fname,$lname, $gender,$course, $yearlvl, $position,  
$partylist, $image, "");
```

```
}
```

```
else
```

```
// if no match, display result
```

```
{
```

```
echo "No results!";

}

}

else

// if the 'id' in the URL isn't valid, or if there is no 'id' value, display an error

{

echo 'Error!';

}

}

?>
```

D.SAMPLE SCREENS

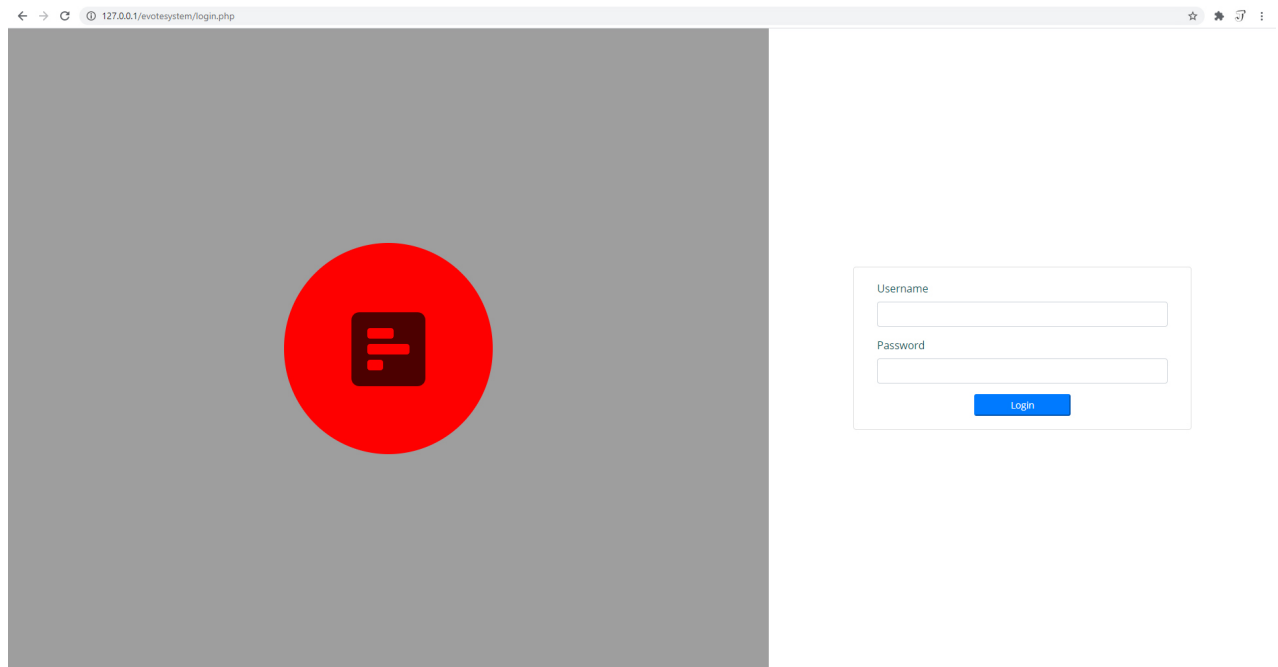


Fig D 1: Login form

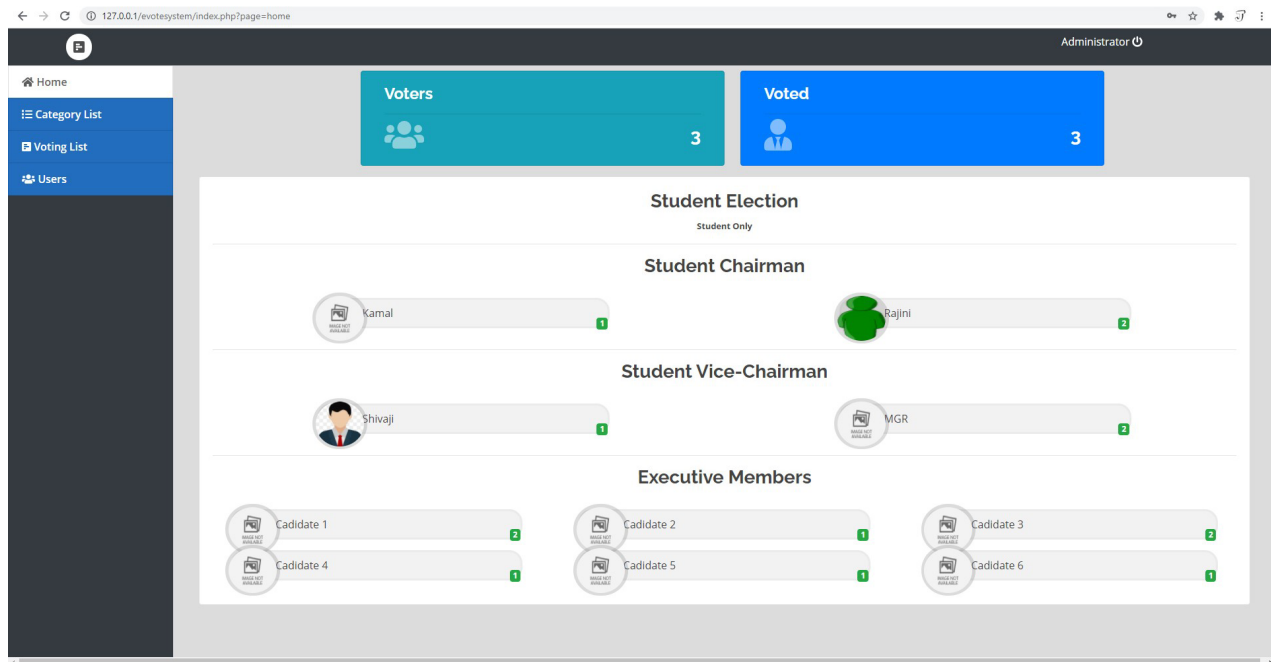


Fig D2 : Registered form

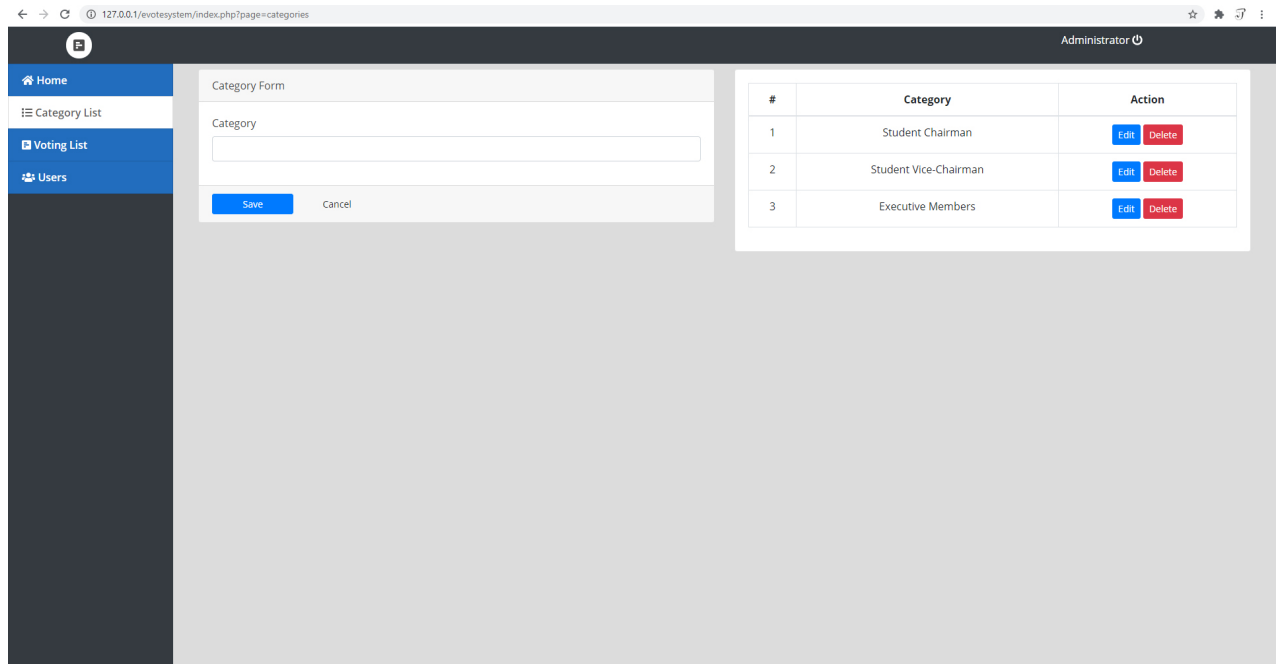


Fig D3 : Catagory

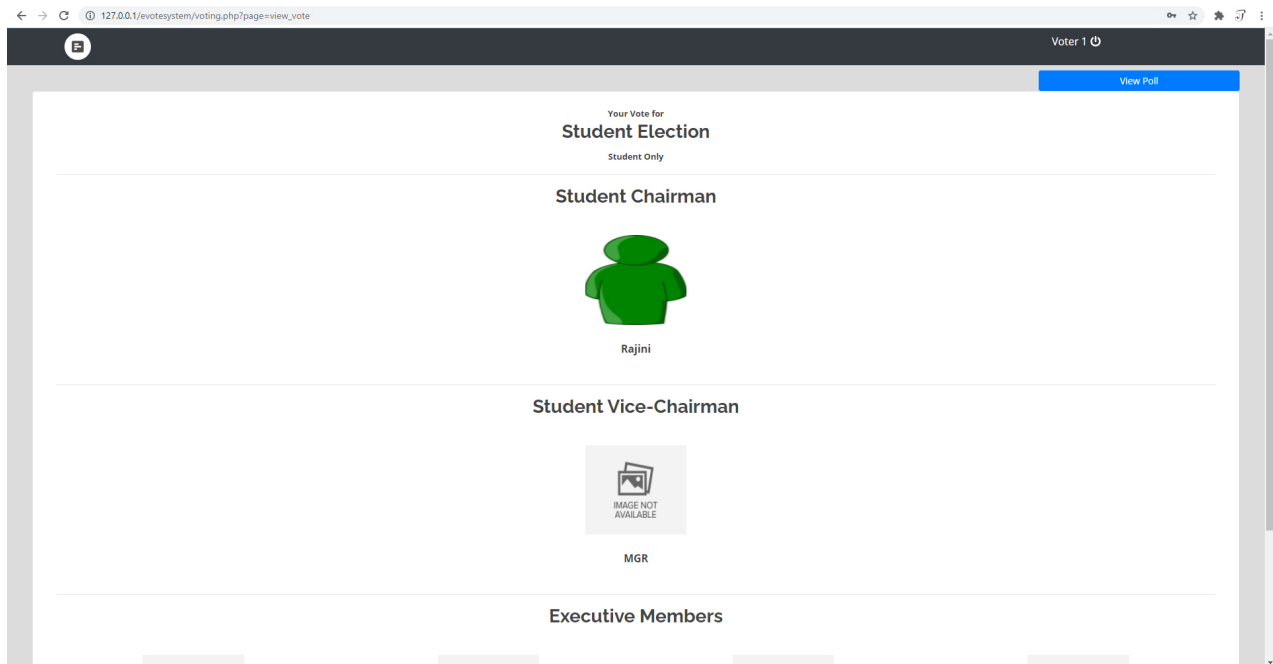


Fig D4 : poling page

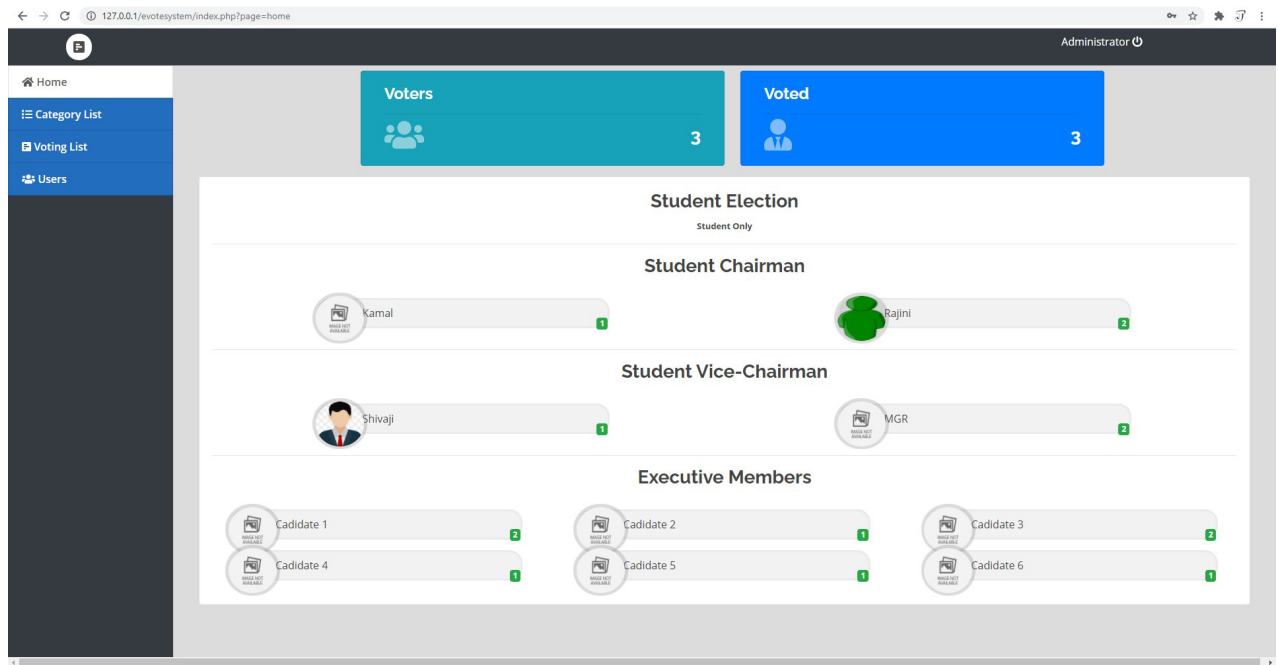


Fig D4 : Candidate form

← → 127.0.0.1/evotesystem/voting.php?page=home Voter 1

Voters 3

Voted 3

Student Election

Student Only

Student Chairman

Kamal 1

Rajini 2

Student Vice-Chairman

Shivaji 1

MGR 2

Executive Members

Candidate 1 2

Candidate 2 1

Candidate 3 2

Candidate 4 1

Candidate 5 1

Candidate 6 1

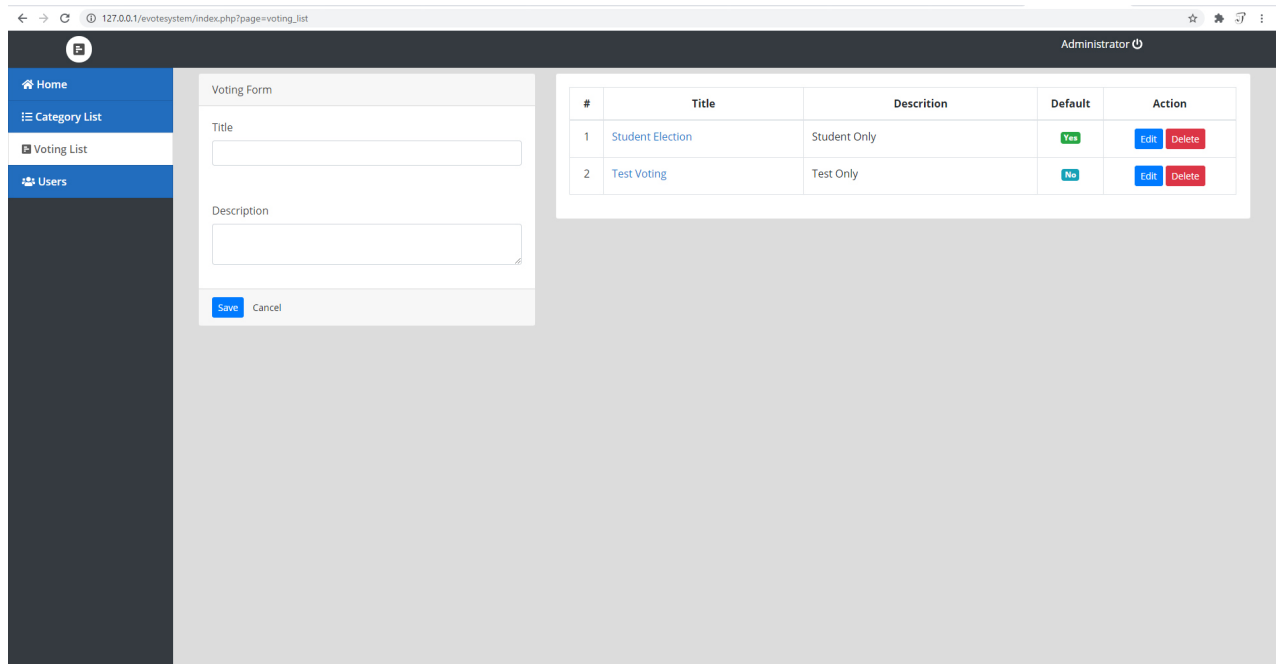


Fig D5 : Election List