**COCHIN UNIVERSITY COLLEGE OF ENGINEERING KUTTANADU**



**PROJECT REPORT ON**

**“Smart Online Voting System”**

Submitted on partial fulfilment of the requirement for the award of the degree in Master of Computer Applications from COCHIN UNIVERSITY OF SCIENCE AND TECHNOLOGY

Submitted by

Name: Aman kumar

Reg No: 38220203

Sem:4th

**DEPARTMENT OF MASTER OF COMPUTER APPLICATIONS**

2020-2022

**COCHIN UNIVERSITY OF SCIENCE AND TECHNOLOGY**

**COCHIN UNIVERSITY COLLEGE OF ENGINEERING**

**KUTTANADU**



***CERTIFICATE***

This is to certify that this project report entitled “**Smart Online Voting System**” is a bonafide record on partial fulfilment for the Degree of the Master of Computer Applications to the COCHIN UNIVERSITY OF SCIENCE AND TECHNOLOGY through DEPARTMENT OF COMPUTER APPLICATIONS, COCHIN UNIVERSITY COLLEGE OF ENGINEERING KUTTANADU, ALAPPUZHA done by ***Aman Kumar(Reg NO:38220203)***in the year 2022.

**Project Guide Head of the Department**

**Internal Examiner**

**DECLARATION**

I hereby declare that the project entitled “**Smart Online Voting System**” submitted to the DEPARTMENT OF COMPUTER APPLICATIONS, COCHIN UNIVERSITY COLLEGE OF ENGINEERING, KUTTANADU in the partial fulfilment of the requirements for the award of Degree in MASTER OF COMPUTER APPLICATIONS is a record of original work done by me under the guidance of ***Mrs. Radhika B.*** Assistant Professor in MCA Department during my period of study in COCHIN UNIVERSITY COLLEGE OF ENGINEERING, KUTTANADU.

Place : AMAN KUMAR

Date :

**ACKNOWLEDGEMENT**

I am thankful to god almighty for the blessings in the successful completion of my project “**Smart Online Voting System**”. I would like to record my profound gratitude to ***Dr. Jacob Kutty Jacob***, Principal and Head of the Department, MCA, COCHIN UNIVERSITY COLLEGE OF ENGINEERING who has deeply inspired me to do my project.

It’s grateful to express our thanks to ***Mrs. Radhika B*, Mrs.*DEEPA NAIR***, and ***Mrs. SREELEKSHMI R*** Assistant Professor in MCA Department my Project guide, **COCHIN UNIVERSITY COLLEGE OF ENGINEERING, KUTTANADU**, because her effective guidance, constructive criticism and innovative and useful stream of suggestions that helped me to complete our project.

I am thankful to various resources that provide requirements for my projects, because requirements are backbone of every project.

I am also thankful to our teachers, friends, family members, for their support and prayer for me to complete our project.

**SYNOPSIS**

The project “Smart Online Voting System" aims at making the voting process easy in any type of elections.Presently voting is performed using ballot paper and the counting is done manually, hence it consumes a lot of time. There can be possibility of invalid votes. All these make election a tedious task. In recent times in India, due to elections the second wave of COVID transmission also made huge loss of human lives. In our proposed system voting and counting is done with the help of computer in Online. It saves time, avoid error in counting and there will be no invalid votes. It makes the election process easy. It also avoids the process of physical touching or visiting any places and so in the time of pandemic too it will be more helpful to conduct elections. The system deals with the online voting and its details. Allows the user to vote for the candidate online.Can get the details of the candidate and voter as well. Without the wastage of time the citizen can vote the respective candidate. In present existing system we are using ballot paper and counting the number of votes, it takes the lot of time to for the existing process, to overcome the drawbacks in the existing system this particular system was proposed to mark our work much easier and to reduce wastage of time. And more over we doesn’t gets the accurate results in the present existing system. So there is a need for Online Voting Systems.

**CONTENTS**

**No: Title Page**

1. **INTRODUCTION....................................................................................1**
   1. ABOUT THE PROJECT
   2. OBJECTIVE & SCOPE OF THE PROJECT
   3. DEFINITION OF PROBLEM
2. **SYSTEM ANALYSIS..............................................................................4**
   1. EXISTING SYSTEM

2.1.1 DISADVANTAGES

* 1. PROPOSED SYSTEM

2.2.1 ADVANTAGES

2.2.2 MODULES

* 1. FEASIBILITY STUDY

1. **SYSTEM REQUIREMENTS AND SPECIFICATIONS....................9**
   1. HARDWARE CONFIGURATIONS
   2. SOFTWARE CONFIGURATIONS
   3. PLATFORM USED
2. **SYSTEM DESIGN................................................................................12**
   1. DATA FLOW DIAGRAM
   2. CLASS DIAGRAM
   3. USE CASE DIAGRAM
   4. SEQUENCE DIAGRAM
   5. ACTIVITY DIAGRAM
3. **SYSTEM IMPLEMENTATION AND TESTING.............................20**

5.1 SYSTEM IMPLEMENTATION

5.2 SYSTEM TESTING

5.2.1 UNIT TESTING

5.2.2 INTEGRATION TESTING

5.2.3 SYSTEM TESTING

5.2.4 TEST REPORT

5.3 SYSTEM MAINTENANCE

1. **CONCLUSION.......................................................................................26**

6.1 CONCLUSION

6.2 FUTURE SCOPE

1. **SAMPLE CODE.....................................................................................28**
2. **SCREENSHOTS....................................................................................33**

**INTRODUCTION**

* 1. **ABOUT THE PROJECT**
* The project “Smart Online Voting System” aims at making the voting process easy in any type of elections.
* Presently voting is performed using ballot paper and the counting is done manually, hence it consumes a lot of time.

There can be possibility of invalid votes. All these make election a tedious task. In recent times in India, due to elections the second wave of COVID transmission also made huge loss of human lives.

* In our proposed system voting and counting is done with the help of computer in Online. It saves time, avoid error in counting and there will be no invalid votes. It makes the election process easy.
  1. **OBJECTIVE & SCOPE OF THE PROJECT**

The main goal of “**Smart Online Voting System**” is to give the vote virtually.

**The objective and goals of the proposed system are:**

* To Vote from Home.
* No need to arrange anything.
* It will increase the accuracy and save the time and money.

**1.3 DEFINITION OF PROBLEM:**

The main aim of the project entitled “**Smart Online Voting System**” is to give the vote virtually and save the time and cost of the person and increase the accuracy.

**SYSTEM ANALYSIS**

**2.1 EXISTING SYSTEM**

* Existing system is a manual one in which users and the details of the candidates are stored in books.
* The users have to wait a long time in queues for voting.
* Wrong and unwanted votes are given.
* Counting of votes are done manually which takes lots of time and inaccurate counting is done. It is very difficult to maintain historical data.
* In the existing system, there is compulsory need in physical presence in the time of election polling or vote counting.

**2.1.1 DISADVANTAGES**

* If elections are conducted in existing system model in the pandemic time, then there is sure spread of disease like COVID, which happened in the recent elections in India.
* It is difficult to maintain important information in books.
* More manual hours are needed for counting of votes.
* It is tedious to manage historical data which needs much space to keep all the information regarding the voters and the candidates.
* Voters have to wait in long queues for voting they have to travel long distances.

**2.2 PROPOSED SYSTEM**

* The Online Voting System is a software application which avoids more manual hours that need to spend in record keeping and calculating votes.
* Through this the users and the candidates are registered online. Their information is stored in the database the admin can easily access the details of the voters and the candidates.
* The voters are allowed to vote online they can even vote by sitting at home. Every User allowed to vote only once so there is no chance of duplicated votes.

**2.2.1 ADVANTAGES**

* The objective of the VOTING SOFTWARE is to provide better information for the users of this system easily they can vote from anywhere without facing any difficulty.
* The proposed system does not require any physical presence during vote polling or counting. So it is very easy to conduct elections even during the pandemic situations without any spread of disease or human live losses.
* The proposed system has good authentication so only authorized person can able to vote and also cannot vote multiple types.
* Vote Counting can be made very quickly and results will be displayed in few minutes.

**2.2.2 MODULES**

* Admin Module

Add Elections

Add Candidate.

View Voter and approve voters

View the Vote and results

* Voter Module

Voter Registration

View Candidate details

Vote

**2.3 FEASIBILITY STUDY**

The feasibility of the project is analyzed in this phase and business proposal is put forth with a very general plan for the project and some cost estimates. During system analysis the feasibility study of the proposed system is to be carried out. This is to ensure that the proposed system is not a burden to the company. For feasibility analysis, some understanding of the major requirements for the system is essential.

Three key considerations involved in the feasibility analysis are

1. ECONOMICAL FEASIBILITY
2. TECHNICAL FEASIBILITY

SOCIAL FEASIBILIT The feasibility of the project is analyzed in this phase and business proposal is put forth with a very general plan for the project and some cost estimates. During system analysis the feasibility study of the proposed system is to be carried out. This is to ensure that the proposed system is not a burden to the company. For feasibility analysis, some understanding of the major requirements for the system is essential.

Three key considerations involved in the feasibility analysis are

* ECONOMICAL FEASIBILITY
* TECHNICAL FEASIBILITY
* SOCIAL FEASIBILITY

**1.ECONOMICAL FEASIBILITY**

This study is carried out to check the economic impact that the system will have on the organization. The amount of fund that the company can pour into the research and development of the system is limited. The expenditures must be justified. Thus the developed system as well within the budget and this was achieved because most of the technologies used are freely available. Only the customized products had to be purchased.

### **2.TECHNICAL FEASIBILITY**

This study is carried out to check the technical feasibility, that is, the technical requirements of the system. Any system developed must not have a high demand on the available technical resources. This will lead to high demands on the available technical resources. This will lead to high demands being placed on the client. The developed system must have a modest requirement, as only minimal or null changes are required for implementing this system.

3.**SOCIAL FEASIBILITY**

The aspect of study is to check the level of acceptance of the system by the user. This includes the process of training the user to use the system efficiently. The user must not feel threatened by the system, instead must accept it as a necessity. The level of acceptance by the users solely depends on the methods that are employed to educate the user about the system and to make him familiar with it. His level of confidence must be raised so that he is also able to make some constructive criticism, which is welcomed, as he is the final user of the system.

**SYSTEM REQUIREMENT**

**AND**

**SPECIFICATIONS**

**3.1 HARDWARE CONFIGURATIONS**

* System : Pentium IV 2.4 GHz.
* Hard Disk : 40 GB.
* Floppy Drive : 1.44 Mb.
* Monitor : 15 VGA Colour.
* Mouse : Logitech.
* Ram : 512 Mb.

**3.2 SOFTWARE CONFIGURATIONS**

* Operating system : - Windows XP/7.
* Coding Language : JAVA/J2EE
* Data Base : MYSQL

**3.3 PLATFORM USED**

* Netbeans 8.2
* Windows
* Tomcat server

**SYSTEM DESIGN**

**4.1 Data Flow Diagram**

Vote

View Candidate Details

End process

View Elections

Login Using Voter Id

Get Voter ID

Approve

User Register

Unauthorized user

**Admin**

View Election Results

Add Candidates

End process

Add Elections

Send Voter ID

Approve Voters

View Voter Details

No

Check

Admin Login

Unauthorized user

**4.2 Class Diagram**

Get Voter ID ()

View Election ()

View Candidate Details ()

Vote ()

User

Login

View Voter Details ()

Approve Voter ()

Send Voter ID ()

Add Election ()

Add Candidates ()

View Election Results ()

Admin

Login

**4.3 USE CASE DIAGRAM**

User

Admin

**4.4 SEQUENCE DIAGRAM**

SERVICE

User Admin

Register

view voter details

**Approve user**

Send voter ID

**\**

Get Voter ID

Add Candidate

Add Elections

****

DATA BASE

View Election Results

Vote

**View Candidates**

View Elections

**4.5 ACTIVITY DIAGRAM**

Get Voter ID, View Election, View Candidates, Vote.

LOGIN

User

View Voter Details, Approve Voters, Send Voter ID, Add Elections, Add Candidates, View Election Results.

LOGIN

Admin

Start

**SYSTEM IMPLEMENTATION,**

**TESTING**

**&**

**MAINTENANCE**

**5.1 SYSTEM IMPLEMENTATION**

I am implementing the system in the MATLAB language. [MATLAB®](https://www.mathworks.com/products/matlab.html) is a programming platform designed specifically for engineers and scientists to analyze and design systems and products that transform our world. The heart of MATLAB is the MATLAB language, a matrix-based language allowing the most natural expression of computational mathematics.

**5.2 SYSTEM TESTING**

**System testing** is a level of testing that validates the complete and fully integrated software product. The purpose of a system test is to evaluate the end-to-end system specifications. Usually, the software is only one element of a larger computer-based system. Ultimately, the software is interface with other software/hardware systems. System Testing is actually a series of different tests whose sole purpose is to exercise the full computer-based system.

**Two Category of Software Testing**

* Black Box Testing
* White Box Testing

System test falls under the **black box testing** category of software testing.

**White box testing** is the testing of the internal workings or code of a software application. In contrast, black box or System Testing is the opposite. System test involves the external workings of the software from the user's perspective.

**System Testing involves testing the software code for following**

* Testing the fully integrated applications including external peripherals in order to check how components interact with one another and with the system as a whole. This is also calledEnd-to-End testing scenario.
* Verify thorough testing of every input in the application to check for desired outputs.
* Testing of the user's experience with the application.

That is a very basic description of what is involved in system testing. You need to build detailed test cases and test suites that test each aspect of the application as seen from the outside without looking at the actual source code.

**5.2.1 UNIT TESTING**

Unit testing performed on each module or block of code during development. The programmer who writes the code normally does unit testing. It focuses on smallest unit of software design. In this we test, an individual unit or group of inter related units. Programmer often does it by using sample input and observing its corresponding outputs. The main aim is to isolate each unit of the system to identify, analyse and fix the defects.

## Advantages:

* Reduces Defects in the newly developed features or reduces bugs when changing the existing functionality.
* Reduces Cost of Testing as defects are captured in very early phase.
* Improves design and allows better refactoring of code.
* Unit Tests, when integrated with build gives the quality of the build as well.

## Techniques:

* **Black Box Testing -**Using which the user interface, input and output are tested.
* **White Box Testing -**used to test each one of those functions behaviour is tested.
* **Grey Box Testing -**Used to execute tests, risks and assessment methods.

**5.2.2 INTEGRATION TESTING**

The meaning of Integration testing is quite straightforward- Integrate/combine the unit tested module one by one and test the behaviour as a combined unit.The main function or goal of this testing is to test the interfaces between the units/modules.We normally do Integration testing after “Unit testing”. Once all the individual units are created and tested, we start combining those “Unit Tested” modules and start doing the integrated testing.The main function or goal of this testing is to test the interfaces between the units/modules.The individual modules are first tested in isolation. Once the modules are unit tested, they are integrated one by one, until all the modules are integrated, to check the combinational behaviour, and validate whether the requirements are implemented correctly or not.Here we should understand that Integration testing does not happen at the end of the cycle, rather it is conducted simultaneously with the development. So in most of the times, all the modules are not actually available to test and here is what the challenge comes to test something which does not exist!

### **ADVANTAGES**

**There are several advantages of this testing and few of them are listed below:**

* This testing makes sure that the integrated modules/components work properly.
* Integration testing can be started once the modules to be tested are available. It does not require the other module to be completed for testing to be done, as Stubs and Drivers can be used for the same.
* It detects the errors related to the interface.

**5.2.3 SYSTEM TESTING**

System Testing (ST) is a black box testing technique performed to evaluate the complete system the system's compliance against specified requirements. In System testing, the functionalities of the system are tested from an end-to-end perspective.System Testing is usually carried out by a team that is independent of the development team in order to measure the quality of the system unbiased. It includes both functional and Non-Functional testing.Black-box testing treats the software as a "black box", examining functionality without any knowledge of internal implementation, without seeing the source code. The testers are only aware of what the software is supposed to do, not how it does it. Black-box testing methods include: equivalence partitioning, boundary value analysis, all-pairs testing, state transition tables, decision table testing, fuzz testing, model-based testing, use case testing, exploratory testing and specification-based testing.Specification-based testing aims to test the functionality of software according to the applicable requirements. This level of testing usually requires thorough test cases to be provided to the tester, who then can simply verify that for a given input, the output value (or behaviour), either "is" or "is not" the same as the expected value specified in the test case. Test cases are built around specifications and requirements, i.e., what the application is supposed to do. It uses external descriptions of the software, including specifications, requirements, and designs to derive test cases. These tests can be functional or non-functional, though usually functional.Specification-based testing may be necessary to assure correct functionality, but it is insufficient to guard against complex or high-risk situations. One advantage of the black box technique is that no programming knowledge is required. Whatever biases the programmers may have had, the tester likely has a different set and may emphasize different areas of functionality. On the other hand, black-box testing has been said to be "like a walk in a dark labyrinth without a flashlight. “Because they do not examine the source code, there are situations when a tester writes many test cases to check something that could have been tested by only one test case, or leaves some parts of the program untested.

**5.2.4 TEST REPORT**

Test Report is needed to reflect testing results in a formal way, which gives an opportunity to estimate testing results quickly. It is a document that records data obtained from an evaluation experiment in an organized manner, describes the environmental or operating conditions, and shows the comparison of test results with test objectives.

**5.3 SYSTEM MAINTENANCE**

An integral part of software is the maintenance one, which requires an accurate maintenance plan to be prepared during the software development. It should specify how users will request modifications or report problems. The budget should include resource and cost estimates. A new decision should be addressed for the developing of every new system feature and its quality objectives. The software maintenance, which can last for 5–6 years (or even decades) after the development process, calls for an effective plan which can address the scope of software maintenance, the tailoring of the post-delivery/deployment process, the designation of who will provide maintenance, and an estimate of the life-cycle costs. The selection of proper enforcement of standards is the challenging task right from early stage of software engineering which has not got definite importance by the concerned stakeholders.

**CONCLUSION**

**6.1 CONCLUSION**

The project was designed in such a way that future modifications can be done easily. The following conclusions can be deduced from the development of the project.

* Automation of the entire system improves the efficiency
* It provides a friendly graphical user interface which proves to be better when compared to the existing system.
* It gives appropriate access to the authorized users depending on their permissions.
* It effectively overcomes the delay in communications.
* Updating of information becomes so easier.
* System security, data security and reliability are the striking features.
* The System has adequate scope for modification in future if it is necessary.

**6.2 FUTURE SCOPE**

It is an easy way and an effective way for voting which allows the reduction of travelling,waiting in a long queues avoid unwanted votes and duplicated votes.And we can get the accurate number of votes.

Well I and my team member have worked hard in order to present an improved website better than the existing one’s regarding the information about the various activities. Still,we found out that the project can be done in a better way. Primarily, when we register we get an unique id by using that unique id we allowed to login and allowed to vote.Each and every user are allowed to vote only once nobody is allowed to vote more than once and only the registered persons only can vote.The candidates also gets registered by getting a unique id and he can compete from anywhere and can see the results.

The next enhancement that we can add different kinds of buttons for easy access and easy availability to each and every one in remote areas.These are the enhancements that we could think of at present.

**SAMPLE CODE**

**\*JSP CODE**

<%@page import="java.text.SimpleDateFormat"%>

<%@page import="java.text.DateFormat"%>

<%@page import="java.util.Date"%>

<%@page import="java.sql.ResultSet"%>

<%@page import="java.sql.Statement"%>

<%@page import="DBconnection.SQLconnection"%>

<%@page import="java.sql.Connection"%>

<%@page contentType="text/html" pageEncoding="UTF-8"%>

<%

String mail = request.getParameter("email");

String pass = request.getParameter("pass");

String voterid = request.getParameter("voterid");

System.out.println("Check User name And Password : " + mail + pass);

Connection con = SQLconnection.getconnection();

Statement st = con.createStatement();

Statement sto = con.createStatement();

try {

ResultSet rs = st.executeQuery("SELECT \* FROM user\_reg where email='" + mail + "' AND password='" + pass + "' AND id = '"+ voterid +"' AND status = 'Approved'");

if (rs.next()) {

session.setAttribute("uid", rs.getString("id"));

session.setAttribute("uname", rs.getString("name"));

session.setAttribute("umail", rs.getString("email"));

response.sendRedirect("User\_Home.jsp?Success");

} else {

response.sendRedirect("User\_login.jsp?Failed");

}

} catch (Exception ex) {

ex.printStackTrace();

}

%>

<%@page contentType="text/html" pageEncoding="UTF-8"%>

<!DOCTYPE html>

<html lang="en">

<head>

<!-- Required meta tags -->

<meta charset="utf-8">

<meta name="viewport" content="width=device-width, initial-scale=1, shrink-to-fit=no">

<title>Online Election System</title>

<!-- Favicon -->

<link rel="shortcut icon" type="image/icon" href="assets/images/favicon.ico"/>

<!-- Font Awesome -->

<link href="https://maxcdn.bootstrapcdn.com/font-awesome/4.6.3/css/font-awesome.min.css" rel="stylesheet">

<!-- Bootstrap CSS -->

<link rel="stylesheet" href="https://maxcdn.bootstrapcdn.com/bootstrap/4.0.0-beta/css/bootstrap.min.css" integrity="sha384-/Y6pD6FV/Vv2HJnA6t+vslU6fwYXjCFtcEpHbNJ0lyAFsXTsjBbfaDjzALeQsN6M" crossorigin="anonymous">

<!-- Slick slider -->

<link href="assets/css/slick.css" rel="stylesheet">

<!-- Gallery Lightbox -->

<link href="assets/css/magnific-popup.css" rel="stylesheet">

<!-- Skills Circle CSS -->

<link rel="stylesheet" type="text/css" href="https://unpkg.com/circlebars@1.0.3/dist/circle.css">

<!-- Main Style -->

<link href="assets/css/style.css" rel="stylesheet">

<!-- Google Fonts Raleway -->

<link href="https://fonts.googleapis.com/css?family=Raleway:300,400,400i,500,500i,600,700" rel="stylesheet">

<!-- Google Fonts Open sans -->

<link href="https://fonts.googleapis.com/css?family=Open+Sans:400,400i,600,700,800" rel="stylesheet">

</head>

<body>

<!--START SCROLL TOP BUTTON -->

<a class="scrollToTop" href="#">

<i class="fa fa-angle-up"></i>

</a>

<!-- END SCROLL TOP BUTTON -->

<!-- Start Header -->

<header id="mu-hero">

<div class="container">

<nav class="navbar navbar-expand-lg navbar-light mu-navbar">

<!-- Text based logo -->

<a class="navbar-brand mu-logo" href="index.html"><span></span></a>

<!-- image based logo -->

<!-- <a class="navbar-brand mu-logo" href="index.html"><img src="assets/images/logo.png" alt="logo"></a> -->

<button class="navbar-toggler" type="button" data-toggle="collapse" data-target="#navbarSupportedContent" aria-controls="navbarSupportedContent" aria-expanded="false" aria-label="Toggle navigation">

<span class="fa fa-bars"></span>

</button>

<div class="collapse navbar-collapse" id="navbarSupportedContent">

<ul class="navbar-nav mr-auto mu-navbar-nav">

<li class="nav-item"><a href="index.html">Home</a></li>

<li class="nav-item active"><a href="User\_login.jsp">User</a></li>

<li class="nav-item"><a href="Admin\_login.jsp">Admin</a></li>

</ul>

</div>

</nav>

</div>

</header>

<!-- End Header -->

<!-- Start slider area -->

<div id="mu-page-header" style="background-image: url(images/slide2.jpg);">

<div class="container">

<div class="row">

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

<div class="mu-page-header-area">

<h1 class="mu-page-header-title">Online Election System</h1>

</div>

</div>

</div>

</div>

</div>

<main>

<section id="mu-contact">

<div class="container">

<div class="row">

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

<div class="mu-contact-area">

<!-- Title -->

<div class="row">

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

<div class="mu-title">

<h2>User Register</h2>

</div>

</div>

</div>

<!-- Start Contact Content -->

<div class="mu-contact-content">

<div class="row">

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

<div class="mu-contact-form-area">

<div id="form-messages"></div>

<form method="post" action="user\_reg" class="mu-contact-form">

<div class="form-group">

<span class="fa fa-user mu-contact-icon"></span>

<input type="text" class="form-control" placeholder="Name" id="name" name="username" required="required" />

</div>

<div class="form-group">

<span class="fa fa-calendar mu-contact-icon"></span>

<input type="date" class="form-control" placeholder="DOB" id="name" name="dob" required="required" />

</div>

<div class="form-group">

<span class="fa fa-envelope mu-contact-icon"></span>

<input type="email" class="form-control" placeholder="Email" id="name" name="email" required="required" />

</div><div class="form-group">

<span class="fa fa-male mu-contact-icon"></span>

<select class="form-control" name="gender" required="required">

<option value="">&nbsp;&nbsp;&nbsp;Select Your Gender</option>

<option>Male</option>

<option>Female</option>

<option>Others</option>

</select>

</div>

<div class="form-group">

<span class="fa fa-phone mu-contact-icon"></span>

<input type="text" class="form-control" placeholder="Phone No" id="name" name="phone" required="required" pattern="[0-9]{10,10}" required title="MOBILE NUMBER MUST BE EXACT 10 DIGIT ONLY AND IT'S NOT CONTAIN ANY APLHABETS"/>

</div>

<div class="form-group">

<span class="fa fa-home mu-contact-icon"></span>

<input type="text" class="form-control" placeholder="Address" id="name" name="address" required="required" />

</div>

<div class="form-group">

<span class="fa fa-key mu-contact-icon"></span>

<input type="password" class="form-control" placeholder="Password" id="pass" name="pass" required="required" />

</div>

<div class="form-group">

<button type="submit" class="mu-send-msg-btn"><span>Sign Up</span></button>

</div>

</form>

</div>

</div>

</div>

</div>

<!-- End Contact Content -->

</div>

</div>

</div>

</div>

</section>

</main>

<!-- End main content -->

<!-- Start footer -->

<footer id="mu-footer">

<div class="mu-footer-bottom">

<div class="container">

<div class="row">

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

<div class="mu-footer-bottom-area">

<p class="mu-copy-right">&copy;

</div>

</div>

</div>

</div>

</div>

</footer>

<!-- End footer -->

<!-- JavaScript -->

<!-- jQuery first, then Popper.js, then Bootstrap JS -->

<script src="https://ajax.googleapis.com/ajax/libs/jquery/3.2.1/jquery.min.js"></script>

<script src="https://cdnjs.cloudflare.com/ajax/libs/popper.js/1.11.0/umd/popper.min.js" integrity="sha384-b/U6ypiBEHpOf/4+1nzFpr53nxSS+GLCkfwBdFNTxtclqqenISfwAzpKaMNFNmj4" crossorigin="anonymous"></script>

<script src="https://maxcdn.bootstrapcdn.com/bootstrap/4.0.0-beta/js/bootstrap.min.js" integrity="sha384-h0AbiXch4ZDo7tp9hKZ4TsHbi047NrKGLO3SEJAg45jXxnGIfYzk4Si90RDIqNm1" crossorigin="anonymous"></script>

<!-- Slick slider -->

<script type="text/javascript" src="assets/js/slick.min.js"></script>

<!-- Progress Bar -->

<script src="https://unpkg.com/circlebars@1.0.3/dist/circle.js"></script>

<!-- Filterable Gallery js -->

<script type="text/javascript" src="assets/js/jquery.filterizr.min.js"></script>

<!-- Gallery Lightbox -->

<script type="text/javascript" src="assets/js/jquery.magnific-popup.min.js"></script>

<!-- Counter js -->

<script type="text/javascript" src="assets/js/counter.js"></script>

<!-- Ajax contact form -->

<script type="text/javascript" src="assets/js/app.js"></script>

<!-- Custom js -->

<script type="text/javascript" src="assets/js/custom.js"></script>

<!-- About us Skills Circle progress -->

<script>

// First circle

new Circlebar({

element: "#circle-1",

type: "progress",

maxValue: "90"

});

// Second circle

new Circlebar({

element: "#circle-2",

type: "progress",

maxValue: "84"

});

// Third circle

new Circlebar({

element: "#circle-3",

type: "progress",

maxValue: "60"

});

// Fourth circle

new Circlebar({

element: "#circle-4",

type: "progress",

maxValue: "74"

});

</script>

</body>

</html>

<%@page contentType="text/html" pageEncoding="UTF-8"%>

<%@page import="java.sql.ResultSet"%>

<%@page import="DBconnection.SQLconnection"%>

<%@page import="java.sql.Statement"%>

<%@page import="java.sql.Connection"%>

<!DOCTYPE html>

<html lang="en">

<head>

<!-- Required meta tags -->

<meta charset="utf-8">

<meta name="viewport" content="width=device-width, initial-scale=1, shrink-to-fit=no">

<title>Online Election System</title>

<!-- Favicon -->

<link rel="shortcut icon" type="image/icon" href="assets/images/favicon.ico"/>

<!-- Font Awesome -->

<link href="https://maxcdn.bootstrapcdn.com/font-awesome/4.6.3/css/font-awesome.min.css" rel="stylesheet">

<!-- Bootstrap CSS -->

<link rel="stylesheet" href="https://maxcdn.bootstrapcdn.com/bootstrap/4.0.0-beta/css/bootstrap.min.css" integrity="sha384-/Y6pD6FV/Vv2HJnA6t+vslU6fwYXjCFtcEpHbNJ0lyAFsXTsjBbfaDjzALeQsN6M" crossorigin="anonymous">

<!-- Slick slider -->

<link href="assets/css/slick.css" rel="stylesheet">

<!-- Gallery Lightbox -->

<link href="assets/css/magnific-popup.css" rel="stylesheet">

<!-- Skills Circle CSS -->

<link rel="stylesheet" type="text/css" href="https://unpkg.com/circlebars@1.0.3/dist/circle.css">

<!-- Main Style -->

<link href="assets/css/style.css" rel="stylesheet">

<!-- Google Fonts Raleway -->

<link href="https://fonts.googleapis.com/css?family=Raleway:300,400,400i,500,500i,600,700" rel="stylesheet">

<!-- Google Fonts Open sans -->

<link href="https://fonts.googleapis.com/css?family=Open+Sans:400,400i,600,700,800" rel="stylesheet">

</head>

<style>

#customers {

font-family: "Trebuchet MS", Arial, Helvetica, sans-serif;

font-size: 20px;

border-collapse: collapse;

width: 100%;

}

#customers td, #customers th {

border: 2px solid black;

align:"center"; cellpadding:"0"; cellspacing:"2";

padding: 15px;

}

#customers th {

padding-top: 12px;

padding-bottom: 12px;

text-align: left;

background-color: #1DA1F2;

color: white;

}

</style>

<script>

var loadFile = function (event) {

var reader = new FileReader();

reader.onload = function () {

var output = document.getElementById('output');

output.src = reader.result;

};

reader.readAsDataURL(event.target.files[0]);

};

</script>

<body>

<!--START SCROLL TOP BUTTON -->

<a class="scrollToTop" href="#">

<i class="fa fa-angle-up"></i>

</a>

<!-- END SCROLL TOP BUTTON -->

<!-- Start Header -->

<header id="mu-hero">

<div class="container">

<nav class="navbar navbar-expand-lg navbar-light mu-navbar">

<!-- Text based logo -->

<a class="navbar-brand mu-logo" href="index.html"><span></span></a>

<!-- image based logo -->

<!-- <a class="navbar-brand mu-logo" href="index.html"><img src="assets/images/logo.png" alt="logo"></a> -->

<button class="navbar-toggler" type="button" data-toggle="collapse" data-target="#navbarSupportedContent" aria-controls="navbarSupportedContent" aria-expanded="false" aria-label="Toggle navigation">

<span class="fa fa-bars"></span>

</button>

<div class="collapse navbar-collapse" id="navbarSupportedContent">

<ul class="navbar-nav mr-auto mu-navbar-nav">

<li class="nav-item"><a href="User\_Home.jsp">Home</a></li>

<li class="nav-item active"><a href="Elections.jsp">Elections</a></li>

<li class="nav-item"><a href="index.html">Logout</a></li>

</ul>

</div>

</nav>

</div>

</header>

<!-- End Header -->

<!-- Start slider area -->

<div id="mu-page-header" style="background-image: url(images/slide2.jpg);">

<div class="container">

<div class="row">

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

<div class="mu-page-header-area">

<h1 class="mu-page-header-title">Online Election System</h1>

</div>

</div>

</div>

</div>

</div>

<main>

<section id="mu-contact">

<div class="container">

<div class="row">

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

<div class="mu-contact-area">

<!-- Title -->

<%

String id = request.getParameter("cid");

Connection con = SQLconnection.getconnection();

Statement st = con.createStatement();

try {

ResultSet rs = st.executeQuery("select \* from add\_candidate where id='" + id + "' ");

if (rs.next()) {

String address = rs.getString("address");

%>

<div class="row">

<div class="mu-title">

<h2>Candidate Profile</h2>

</div>

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

<img src="Getimage?id=<%=id%>" width="300" height="220">

</div>

<div class="col-md-7">

<form method="post" action="" class="mu-contact-form">

<div class="form-group">

<label>Candidate Name :</label>

<input type="text" readonly="" value="<%=rs.getString("candidate\_name")%>" style="width: 320px;height: 40px;color: black;" class="form-group">

</div>

<div class="form-group">

<label>Party Name :</label>&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;

<input type="text" readonly="" value="<%=rs.getString("party\_name")%>" style="width: 320px;height: 40px;color: black;" class="form-group">

</div>

<div class="form-group">

<label>Phone No :</label>&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;

<input type="text" readonly="" value="<%=rs.getString("phone")%>" style="width: 320px;height: 40px;color: black;" class="form-group">

</div>

<div class="form-group">

<label>Address :</label>&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;

<input type="text" readonly="" value="<%=address%>" style="width: 320px;height: 40px;color: black;" class="form-group">

</div>

</form>

</div>

<%

}

} catch (Exception ex) {

ex.printStackTrace();

}

%>

</div>

</div>

</div>

<!-- End Contact Content -->

</div>

</div>

</div>

</div>

</section>

</main>

<!-- End main content -->

<!-- Start footer -->

<footer id="mu-footer">

<div class="mu-footer-bottom">

<div class="container">

<div class="row">

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

<div class="mu-footer-bottom-area">

<p class="mu-copy-right">&copy;

</div>

</div>

</div>

</div>

</div>

</footer>

<!-- End footer -->

<!-- JavaScript -->

<!-- jQuery first, then Popper.js, then Bootstrap JS -->

<script src="https://ajax.googleapis.com/ajax/libs/jquery/3.2.1/jquery.min.js"></script>

<script src="https://cdnjs.cloudflare.com/ajax/libs/popper.js/1.11.0/umd/popper.min.js" integrity="sha384-b/U6ypiBEHpOf/4+1nzFpr53nxSS+GLCkfwBdFNTxtclqqenISfwAzpKaMNFNmj4" crossorigin="anonymous"></script>

<script src="https://maxcdn.bootstrapcdn.com/bootstrap/4.0.0-beta/js/bootstrap.min.js" integrity="sha384-h0AbiXch4ZDo7tp9hKZ4TsHbi047NrKGLO3SEJAg45jXxnGIfYzk4Si90RDIqNm1" crossorigin="anonymous"></script>

<!-- Slick slider -->

<script src="https://ajax.googleapis.com/ajax/libs/jquery/2.1.1/jquery.min.js"></script>

<script type="text/javascript" src="assets/js/slick.min.js"></script>

<!-- Progress Bar -->

<script src="https://unpkg.com/circlebars@1.0.3/dist/circle.js"></script>

<!-- Filterable Gallery js -->

<script type="text/javascript" src="assets/js/jquery.filterizr.min.js"></script>

<!-- Gallery Lightbox -->

<script type="text/javascript" src="assets/js/jquery.magnific-popup.min.js"></script>

<!-- Counter js -->

<script type="text/javascript" src="assets/js/counter.js"></script>

<!-- Ajax contact form -->

<script type="text/javascript" src="assets/js/app.js"></script>

<!-- Custom js -->

<script type="text/javascript" src="assets/js/custom.js"></script>

<!-- About us Skills Circle progress -->

<script>

// First circle

new Circlebar({

element: "#circle-1",

type: "progress",

maxValue: "90"

});

// Second circle

new Circlebar({

element: "#circle-2",

type: "progress",

maxValue: "84"

});

// Third circle

new Circlebar({

element: "#circle-3",

type: "progress",

maxValue: "60"

});

// Fourth circle

new Circlebar({

element: "#circle-4",

type: "progress",

maxValue: "74"

});

</script>

</body>

</html>

\*JAVA CODE(SERVLET)

package Online\_Election\_System;

import DBconnection.SQLconnection;

import java.io.IOException;

import java.io.PrintWriter;

import java.sql.Connection;

import java.sql.PreparedStatement;

import java.sql.ResultSet;

import java.sql.SQLException;

import java.sql.Statement;

import javax.servlet.ServletException;

import javax.servlet.http.HttpServlet;

import javax.servlet.http.HttpServletRequest;

import javax.servlet.http.HttpServletResponse;

/\*\*

\*

\* @author JAVA-JP

\*/

public class User\_register extends HttpServlet {

/\*\*

\* Processes requests for both HTTP <code>GET</code> and <code>POST</code>

\* methods.

\*

\* @param request servlet request

\* @param response servlet response

\* @throws ServletException if a servlet-specific error occurs

\* @throws IOException if an I/O error occurs

\*/

protected void processRequest(HttpServletRequest request, HttpServletResponse response)

throws ServletException, IOException {

response.setContentType("text/html;charset=UTF-8");

try (PrintWriter out = response.getWriter()) {

String name = request.getParameter("username");

String mail = request.getParameter("email");

String dob = request.getParameter("dob");

String gender = request.getParameter("gender");

String phone = request.getParameter("phone");

String address = request.getParameter("address");

String pass = request.getParameter("pass");

System.out.println("name" + name + "password" + pass + "address"+ address + "mail" + mail + "dob" + dob + "cell" + phone);

Connection conn = SQLconnection.getconnection();

String message = null;

try {

Statement st = conn.createStatement();

ResultSet rs = st.executeQuery("Select \* from user\_reg where email ='" + mail + "'");

if (rs.next()) {

response.sendRedirect("User\_login.jsp?msg=Mail\_Id\_Exists");

} else {

String sql = "insert into user\_reg(name, email, dob, gender, phone, address, password) values (?, ?, ?, ?, ?, ?, ?)";

PreparedStatement statement = conn.prepareStatement(sql);

statement.setString(1, name);

statement.setString(2, mail);

statement.setString(3, dob);

statement.setString(4, gender);

statement.setString(5, phone);

statement.setString(6, address);

statement.setString(7, pass);

int row = statement.executeUpdate();

if (row > 0) {

response.sendRedirect("User\_login.jsp?Register\_Success");

} else {

response.sendRedirect("User\_login.jsp?Register\_Failed");

}

}

} catch (SQLException ex) {

ex.printStackTrace();

}

}

}

// <editor-fold defaultstate="collapsed" desc="HttpServlet methods. Click on the + sign on the left to edit the code.">

/\*\*

\* Handles the HTTP <code>GET</code> method.

\*

\* @param request servlet request

\* @param response servlet response

\* @throws ServletException if a servlet-specific error occurs

\* @throws IOException if an I/O error occurs

\*/

@Override

protected void doGet(HttpServletRequest request, HttpServletResponse response)

throws ServletException, IOException {

processRequest(request, response);

}

/\*\*

\* Handles the HTTP <code>POST</code> method.

\*

\* @param request servlet request

\* @param response servlet response

\* @throws ServletException if a servlet-specific error occurs

\* @throws IOException if an I/O error occurs

\*/

@Override

protected void doPost(HttpServletRequest request, HttpServletResponse response)

throws ServletException, IOException {

processRequest(request, response);

}

/\*\*

\* Returns a short description of the servlet.

\*

\* @return a String containing servlet description

\*/

@Override

public String getServletInfo() {

return "Short description";

}// </editor-fold>

}

package Online\_Election\_System;

import DBconnection.SQLconnection;

import java.io.IOException;

import java.io.InputStream;

import java.io.PrintWriter;

import java.sql.Connection;

import java.sql.PreparedStatement;

import java.sql.ResultSet;

import java.sql.SQLException;

import java.sql.Statement;

import javax.servlet.ServletException;

import javax.servlet.annotation.MultipartConfig;

import javax.servlet.annotation.WebServlet;

import javax.servlet.http.HttpServlet;

import javax.servlet.http.HttpServletRequest;

import javax.servlet.http.HttpServletResponse;

import javax.servlet.http.Part;

/\*\*

\*

\* @author JAVA-JP

\*/

@WebServlet("/add\_candidate")

@MultipartConfig(maxFileSize = 16177215)

public class Add\_candidate extends HttpServlet {

/\*\*

\* Processes requests for both HTTP <code>GET</code> and <code>POST</code>

\* methods.

\*

\* @param request servlet request

\* @param response servlet response

\* @throws ServletException if a servlet-specific error occurs

\* @throws IOException if an I/O error occurs

\*/

protected void processRequest(HttpServletRequest request, HttpServletResponse response)

throws ServletException, IOException {

response.setContentType("text/html;charset=UTF-8");

try (PrintWriter out = response.getWriter()) {

String election\_name = request.getParameter("election\_name");

String candidate\_name = request.getParameter("candidate\_name");

String party\_name = request.getParameter("party\_name");

String address = request.getParameter("address");

String phone = request.getParameter("phone");

InputStream fis = null;

Part filePart = request.getPart("photo");

if (filePart != null) {

// prints out some information for debugging

System.out.println(filePart.getName());

System.out.println(filePart.getSize());

System.out.println(filePart.getContentType());

// obtains input stream of the upload file

fis = filePart.getInputStream();

}

System.out.println("Election Name :" + election\_name + "Candidate name :" + candidate\_name + "Address :" + address + "Phone No :" + phone);

Connection conn = SQLconnection.getconnection();

String message = null;

try {

Statement st = conn.createStatement();

String sql = "insert into add\_candidate(election\_name, candidate\_name, address, phone, photo, party\_name) values (?, ?, ?, ?, ?, ?)";

PreparedStatement statement = conn.prepareStatement(sql);

statement.setString(1, election\_name);

statement.setString(2, candidate\_name);

statement.setString(3, address);

statement.setString(4, phone);

if (fis != null) {

statement.setBlob(5, fis);

}

statement.setString(6, party\_name);

int row = statement.executeUpdate();

if (row > 0) {

response.sendRedirect("Add\_candidate.jsp?Candidate\_added");

} else {

response.sendRedirect("Add\_candidate.jsp?Failed");

}

} catch (SQLException ex) {

ex.printStackTrace();

}

}

}

// <editor-fold defaultstate="collapsed" desc="HttpServlet methods. Click on the + sign on the left to edit the code.">

/\*\*

\* Handles the HTTP <code>GET</code> method.

\*

\* @param request servlet request

\* @param response servlet response

\* @throws ServletException if a servlet-specific error occurs

\* @throws IOException if an I/O error occurs

\*/

@Override

protected void doGet(HttpServletRequest request, HttpServletResponse response)

throws ServletException, IOException {

processRequest(request, response);

}

/\*\*

\* Handles the HTTP <code>POST</code> method.

\*

\* @param request servlet request

\* @param response servlet response

\* @throws ServletException if a servlet-specific error occurs

\* @throws IOException if an I/O error occurs

\*/

@Override

protected void doPost(HttpServletRequest request, HttpServletResponse response)

throws ServletException, IOException {

processRequest(request, response);

}

/\*\*

\* Returns a short description of the servlet.

\*

\* @return a String containing servlet description

\*/

@Override

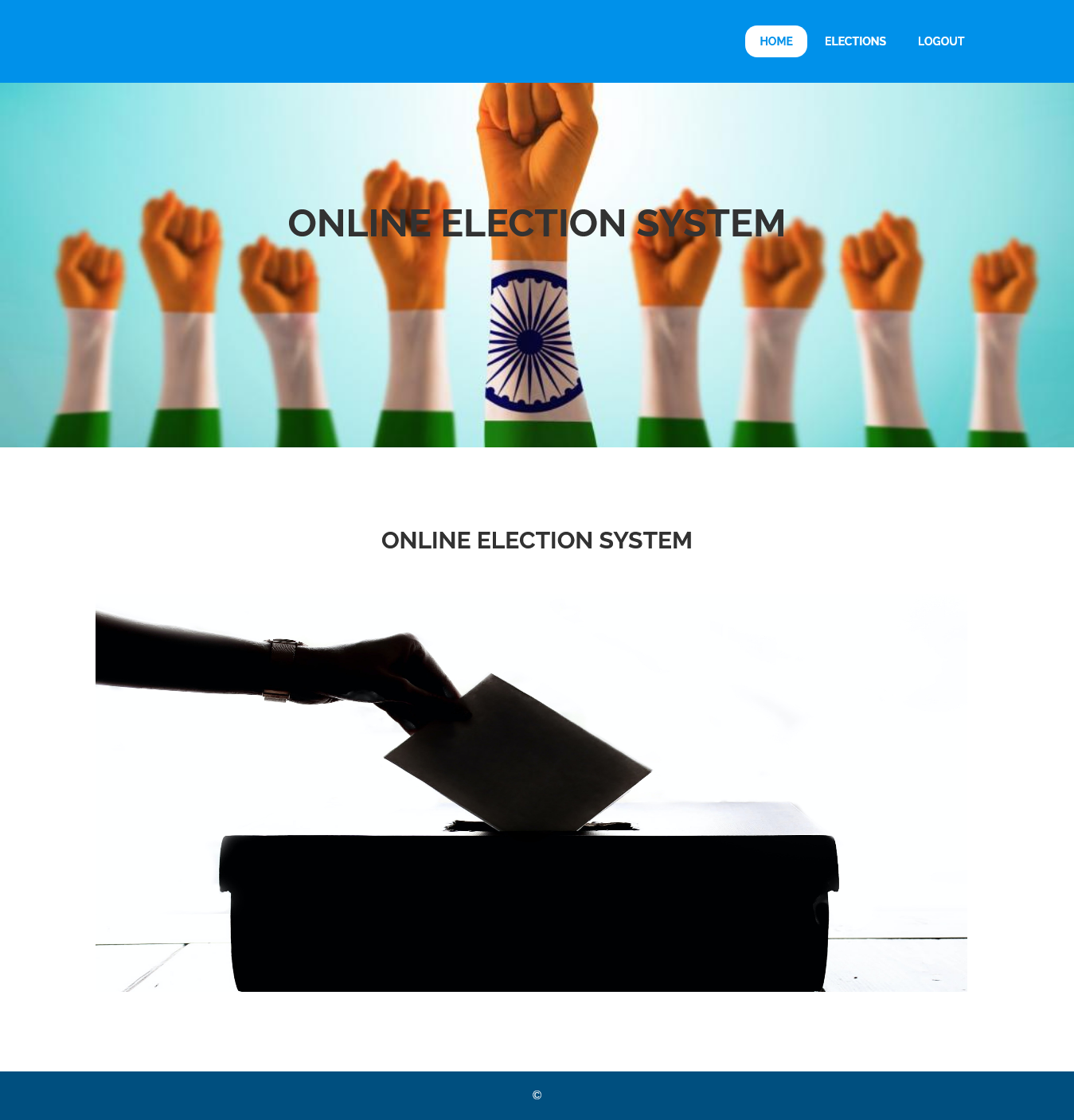
public String getServletInfo() {

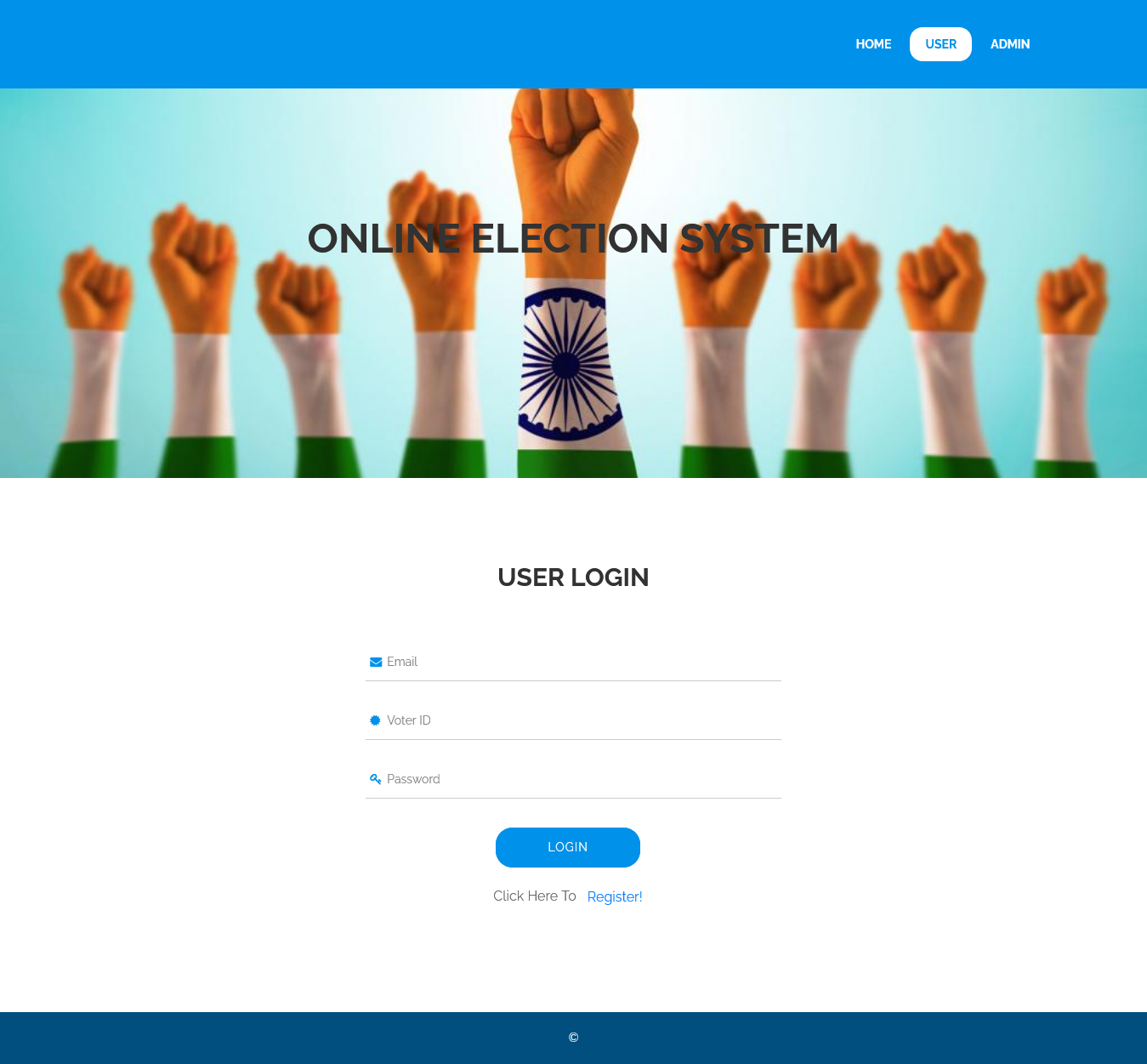
return "Short description";

}// </editor-fold>

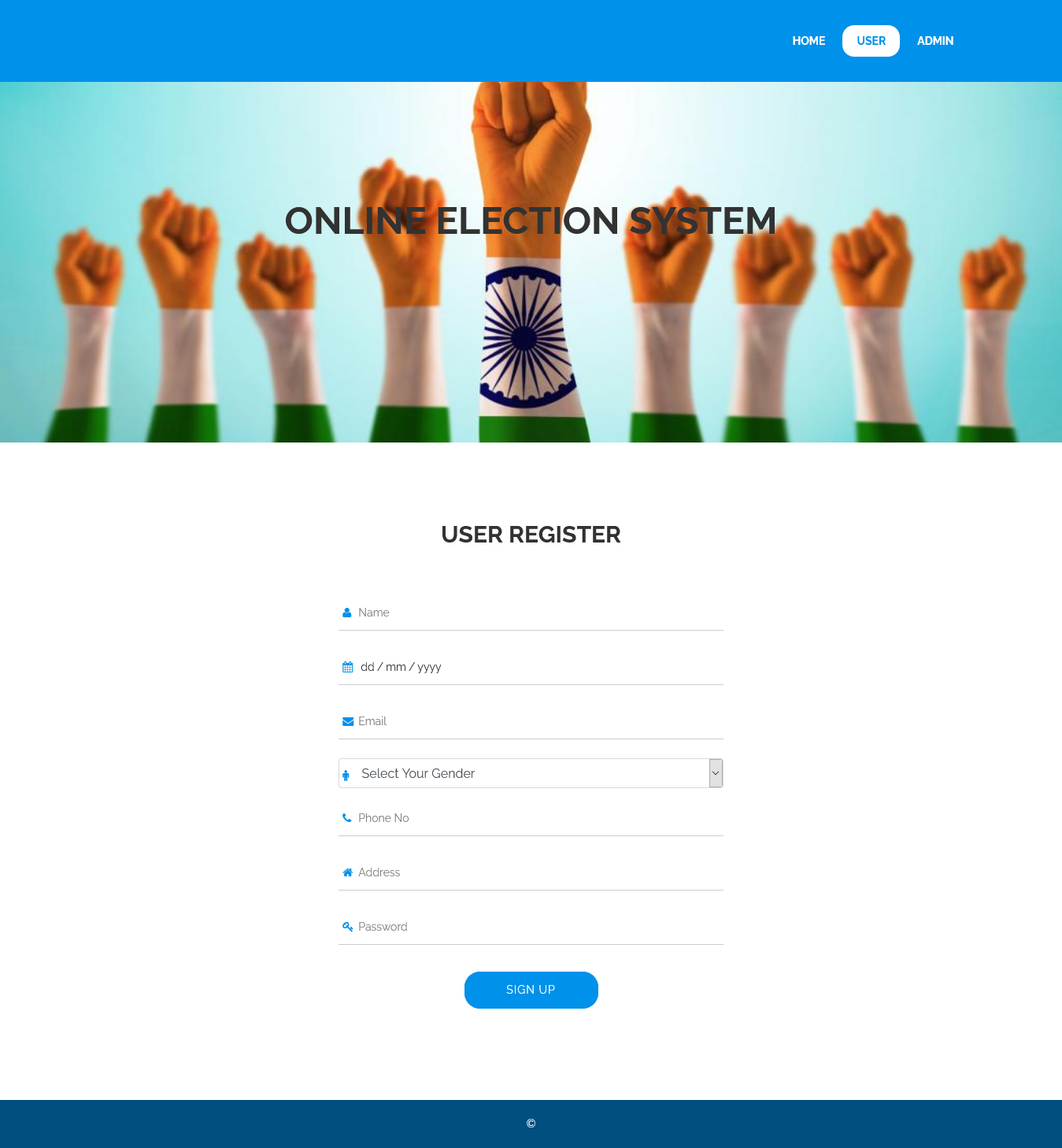
}

**SCREENSHOT**











**REFERENCES**

[1]. Malwade Nikita, PatilChetan, ChavanSuruchi, Prof. Raut S. Y, Secure Online VotingSystem Proposed By Biometrics AndSteganography, Vol. 3, Issue 5, May 2017.

[2]. AnkitAnand, PallaviDivya, AnEfficient Online Voting System, Vol.2,Issue.4, July-Aug. 2019, pp- 2631-2634.

[3]. Alaguvel.R, Gnanavel.G, Jagadhambal.K,Biometrics Using Electronic Voting System withEmbedded Security, Vol. 2, Issue. 3, March2018.

[4]. Firas I. Hazzaa, SeifedineKadry, OussamaKassemZein, Web-Based Voting System UsingFingerprint: Design and Implementation, Vol.2, Issue.4, Dec 2019.

[5] Alexander. Stakeholders: Who is yoursystem for? IEEE: Computing and ControlEngineering, 14(1):22{26,April 2003}.