**COLLEGE ADMISSION MANAGEMENT SYSTEM**

**ABSTRACT**

This project titled as "College Admission Management System" is developed in JAVA as front end and MySQL as back end. The main aim of the project is to control the college admission process through this project. The administrator will login to this project using the username and password. If the username and password are correct, it will open the main form. If the details are not correct, it will say error message. The administrator will enter the department details and the seats allocated for different categories in the specified department. The department detail contains the number of seats, fees details, number of years and the allocation of seats for different categories.

The entered details can be modified or deleted by the administrator. The details will be stored in the database. When a student is placed in a department, then a seat from the category is decreased for the particular category. The administrator cannot place more students in the particular category for the particular department. The college admission is based on the seat available in the seat allocation. The student details are entered and the department in which the student is placed are entered. The student cannot be allocated, if the seat is filled for a particular category. The admission can be modified or deleted by the administrator if the details are wrong. Then the admission fees are entered by the administrator. The fees details can be modified if the entry is wrong. The received fees details can be printed for receipt. The administrator can view all reports in the project. The student reports, admission reports, fees reports are displayed as report to the administrator.

**1. INTRODUCTION**

The main objective of this system is to computerize the entire operation such as information maintenance and the voter voting process. The system is developed as a website for the user compatibility. The user can use this application and complete their need through online. This is an web application voter can voting their vote in this application. The project is aimed to develop by **JAVA** as Front end and **MS SQL SERVER** as Back end. The back end is used to store the information in this system.

**1.1 SYSTEM SPECIFICATION**

**1.1.1 HARDWARE SPECFICATION:**

* Processor : P 4 700 GHz.
* RAM : 4GB RAM
* Hard Disk Drive : 40 GB HDD

**1.1.2 SOFTWARE SPECIFICATION:**

* + Operating System : Windows XP/7/8/10
  + Front End : JAVA
  + Back End : MY SQL

1. **SYSTEM STUDY**

**2.1 EXISTING SYSTEM:**

The academic Management has to handle records for many number of students and maintenance was difficult. Though it has used an information system, it was totally manual. Hence there is a need to upgrade the system with a computer based information system.

**2.1.1 DRAWBACKS:**

* Waste of time
* Allocate student as class wise
* Very hard to manage admission system manually

**2.2 PROPOSED SYSTEM:**

In this system overcome these all kind of issues. Systematically students are assigned the class. With the help of the system it could not be conflict. So it takes very less time to allocate the student process.

**2.2.1 FEATURES:**

By developing the system we can attain the following features:

* It’s automated computerized web based software system
* It uses technologies like html
* It’s easy to operate
* Attractive user interface

1. **SYSTEM DESIGN AND DEVELOPMENT**

**3.1 FILE DESIGN**

The selection of the file system design approach is done according to the needs of the developers what are the needed requirements and specifications for the new design. It allowed us to identify where our proposal fitted in with relation to current and past file system development. Our experience with file system development is limited so the research served to identify the different techniques that can be used. The variety of file systems encountered show what an active area of research file system development is. The file systems may be from one of the two fundamental categories. In one category, the file system is developed in user space and runs as a user process. Another file system may be developed in the kernel space and runs as a privileged process. Another one is the mixed approach in which we can take the advantages of both aforesaid approaches. Each development option has its own pros and cons. In this article, these design approaches are discussed.

**3.2 INPUT DESIGN**

The input design is the link between the information system and the user. It comprises the developing specification and procedures for data preparation and those steps are necessary to put transaction data in to a usable form for processing can be achieved by inspecting the computer to read data from a written or printed document or it can occur by having people keying the data directly into the system. The design of input focuses on controlling the amount of input required, controlling the errors, avoiding delay, avoiding extra steps and keeping the process simple. The input is designed in such a way so that it provides security and ease of use with retaining the privacy. Input Design considered the following things:’

* What data should be given as input?
* How the data should be arranged or coded?
* The dialog to guide the operating personnel in providing input.
* Methods for preparing input validations and steps to follow when error occur.

**OBJECTIVES**

* Input Design is the process of converting a user-oriented description of the input into a computer-based system. This design is important to avoid errors in the data input process and show the correct direction to the management for getting correct information from the computerized system.
* It is achieved by creating user-friendly screens for the data entry to handle large volume of data. The goal of designing input is to make data entry easier and to be free from errors. The data entry screen is designed in such a way that all the data manipulates can be performed. It also provides record viewing facilities.
* When the data is entered it will check for its validity. Data can be entered with the help of screens. Appropriate messages are provided as when needed so that the user
* will not be in maize of instant. Thus the objective of input design is to create an input layout that is easy to follow

**3.3 OUTPUT DESIGN**

A quality output is one, which meets the requirements of the end user and presents the information clearly. In any system results of processing are communicated to the users and to other system through outputs. In output design it is determined how the information is to be displaced for immediate need and also the hard copy output. It is the most important and direct source information to the user. Efficient and intelligent output design improves the system’s relationship to help user decision-making.

1. Designing computer output should proceed in an organized, well thought out manner; the right output must be developed while ensuring that each output element is designed so that people will find the system can use easily and effectively. When analysis design computer output, they should Identify the specific output that is needed to meet the requirements.

2. Select methods for presenting information.

3. Create document, report, or other formats that contain information produced by the system.

The output form of an information system should accomplish one or more of the following objectives.

* Convey information about past activities, current status or projections of the
* Future.
* Signal important events, opportunities, problems, or warnings.
* Trigger an action.
* Confirm an action.

**3.4 DATABASE DESIGN**

Today's businesses depend on their databases to provide information essential for day-to-day operations, especially in case of electronic commerce businesses who has a definite advantage with up-to-date database access. Good design forms the foundation of any database, and experienced hands are required in the automation process to design for optimum and stable performance.

Software Solutions have been constantly working on these platforms and have attained a level of expertise. We apply proven methodologies to design, develop, integrate and implement database systems to attain its optimum level of performance and maximize security to meet the client's business model.

### Business needs addressed:

* Determine the basic objects about which the information is stored
* Determine the relationships between these groups of information and the objects
* Effectively manage data and create intelligent information
* Remote database administration or on site administrative support
* Database creation, management, and maintenance
* Information retrieval efficiency, remove data redundancy and ensure data security

**3.5 SYSTEM DEVELOPMENT**

**3.5.1 DESCRIPTION OF MODULES**

1. Application Module
2. Admission Module
3. Course Module
4. Hostel Module

**Application Module**

Application Module is used to registration the student via this form. We can collect all the information about the student. The main role of this module is an to collect the student details.

**Admission Module**

An Admission module will be called after the application module will be completed. If we need to allocate the student in class we should complete the application form then only we can join the student in the hostel.

**Course Module**

A course module is used to create a course details for colleges. We can allocate a fees details and all the details we can specify in this module. Which is used to user can easily find the course details.

**Hostel Module**

An hostel module is an to allocating a student in the hostel. We can mention the student details in this module. We can easily find all the details about the student. Warden can easily find the student details.

1. **SYSTEM TESTING AND IMPLEMENTATION**

**SYSTEM TESTING**

System testing is the process of exercising software with the intent of finding and ultimately correcting errors. This fundamental philosophy does not change for web applications, because Web-based systems and application reside on a network and interoperate with many different operating system, browsers, hardware platforms, and communication protocols; the search for errors represents a significant challenge for web application.

The distributed nature of client\server environments, the performance issues associated with transaction processing, the potential presence of a number of different hardware platforms, the complexities of network communication, the need to serve multiple clients from a centralized database and the requirements imposed on the server all combine to make testing of client\server architectures.

System testing is actually a series of different tests whose primary purpose is to fully exercise the computer based system. System testing is the state of implementation that is aimed at assuring that the system works accurately and efficiently. Testing is the vital to the success of the system. System testing makes the logical assumption that if all the parts of the system are correct, the goal will be successfully achieved.

**The objective of testing is as follows:**

* + Testing is the process of executing a program with the intent of finding an error.
  + A successful test is that one of the cover of undiscovered error.

### TESTING ISSUES

* Client GUI considerations
* Target environment and platform diversity considerations
* Distributed database considerations
* Distributed processing considerations

**TESTING METHODOLOGIES**

System testing is state of implementation, which is aimed at ensuring that the system works accurately and efficiently as expect before live operation commences. It certifies that the whole set of programs hang together.

System testing requires a test plan that consists of several key activities and step for run program, string, system and user acceptance testing. The implementation of newly designed package is important in adopting a successful new system

Testing is the important stage in software development. the system test in implementation stage in software development process. The system testing implementation should be confirmation that all is correct and an opportunity to show the users that the system works as expected. It accounts the largest percentage of technical effort in the software development process.

Testing phase in the development cycle validates the code against the functional specification testing is vital to achievement of the system goals. The objective of the testing is to discover errors to fulfills this objective a series of test step unit, integration. validation and system tests were planned and executed the test steps are:

**System Testing**

Testing is an important phase in project development. System testing makes a logical assumption that if all parts of the system are correct, and the goal will be achieved successfully. The software must meet the user specification and it must satisfy according to the needs of the users.

Testing is the process of executing a project within the intend of finding errors. A good test case is one that has a high probability of finding an undiscovered error.

**Unit Testing**

Unit testing focuses verification efforts on the smallest unit of software design of the module. This is also known as “module testing”. This testing is carried out during programming stage itself. In this testing step, each module is found to be working satisfactorily as regards to the expected output of the modules.

**In Project**, Each module such customer registration module, request module, employee details module, stock module, vehicle module and area detail modules are tested individually for example, Customer details module can contain the more forms to maintain the information so all forms could be tested like entered information store appropriately in database access page or not. If correctly accessed means the testing of registration module successfully completed. Likewise all modules are tested successfully.

**Integration Testing**

Data can be lost across an interface, one module can have adverse effect on another sub function when combined it may not produce the desired major functions. Integration testing is a systematic testing for constructing test to uncover errors associated within an interface.

The objectives taken from unit tested modules and a program structure is built for integrated testing. All the modules are combined and the test is made.

A correction made in this testing is difficult because the vast expenses of the entire program complicated the isolation of causes. In this integration testing step, all the errors are corrected for next testing process.

**In Project,** Integration of two modules can be tested together such as customer registration details and customer login module for verification purposes providing proper accessibility to users. The communication of Registration and Login module can test and executed successfully.

**Validation Testing**

After the completion of the integrated testing, software is completely assembled as a package; interfacing error has been uncovered and corrected and a final series of software test validation begins.

Validation testing can be defined in many ways but a simple definition is that validation succeeds when the software function in a manner that can be reasonably expected by the customer. After validation test has been conducted, one of two possible conditions exists:

**In this project,** Admin login details form Enter without username and password in textbox enter the submit button then Login failed message otherwise checks the both textbox value that is true means valid page displayed. Enter Password Displaying password character \*.if it displays the characters security is not availed so testing of software is failed.

**Output Testing**

The next process of validation testing, is output testing of the proposed system, since no system could be successful if it does not produce the required output in the specified format. Asking the user about the format required, list the output to be generated or displayed by the system under considerations.

Output testing is a different test whose primary purpose is to fully exercise the computer based system although each test has a different purpose all the work should verify that all system elements have been properly integrated and perform allocated functions.

The output format on the screen is found to be corrected as the format was designed in the system design phase according to the user needs for the hard copy also; the output testing has not resulted in any correction in the system.

**In project** All the forms are tested as it gives the necessary output to the user’s search such as view response details.

1. **CONCLUSION:**

This Project has been designed to the admin login and create students add admission details, then the student will be allocated to that particular class. It will be designed as administrate.

**FUTURE ENHANCEMENT**

This application is developed by using JAVA and MS SQL SERVER as back end. In future this system may be developed by android or any other technology which is peak in that time. According to the user requirement every new technology or any techniques are developed. In future the security will be enhanced with high level. A New technology is implemented with this system to get better performance. The future system will be developed with different features according to the users need. The future enhancement includes a more advance features and secures the data which will be incorporate all the methods, which are convenient for the organization to give the better performance.

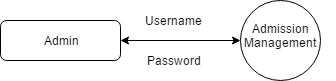
**BIBLIOGRAPHY**

* **AG98**  
  Ken Arnold and James Gosling, The Java Programming Language, second ed., Addison-Wesley, 1998.
* **Chan98**  
  Patrick Chan, The Java Developers Almanac, Addison-Wesley, 1998.
* **CM96**  
  Peter Coad and Mark Mayfield, Java Design: Building Better Apps and Applets, Yourdon Press, 1996.
* **CH97**  
  Gary Cornell and Cay S. Horstmann, Core Java, second ed., SunSoft Press, 1997
* **ELW98**  
  Robert Eckstein and Marc Loy and Dave Wood, Java Swing, O'Reilly, 1998.
* **Englander97**  
  Robert Englander, Developing Java Beans, O'Reilly, 1997.
* **Flanagan96**  
  David Flanagan, Java in a Nutshell, second ed., O'Reilly, 1996.
* **Flanagan99**  
  David Flanagan, Java Foundation Classes in a Nutshell, O'Reilly, 1999.
* **Gea99a**  
  David M. Geary, Graphic Java 2: Mastering the JFC, vol. I, AWT, third ed., Sun Microsystems Press, 1999.
* **Gea99b**  
  David M. Geary Graphic Java 2: Mastering the JFC, vol. II, Swing, third ed., Sun Microsystems Press, 1999.
  + **Gea99c**  
    David M. Geary Graphic Java 2: Mastering the JFC, vol. III, Advanced Swing, third ed., Sun Microsystems Press, 1999(?).

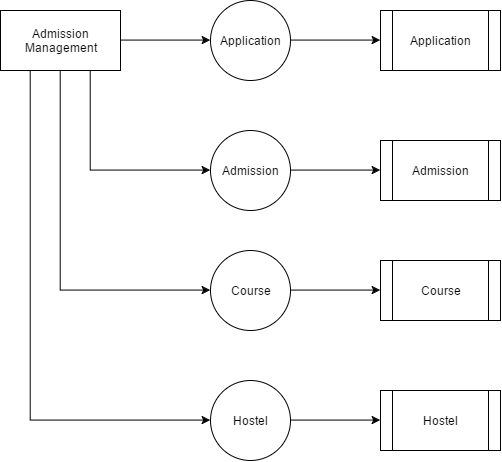
**APPENDICES**

1. **DATA FLOW DIAGRAM**

LEVEL 0:

****

LEVEL 1:

****

1. **TABLE STRUCTURE**

**TABLE NAME : ADMIN**

**PRIMARY\_KEY : ID**

|  |  |  |  |
| --- | --- | --- | --- |
| **FIELD** | **DATA TYPE** | **SIZE** | **DESCRIPTION** |
| Id | Int |  | Admin id |
| Username | Varchar | 30 | Admin username |
| password | Varchar | 30 | Admin password |

**TABLE NAME : APPLICATION**

**PRIMARY KEY : APPLICATIONNO**

|  |  |  |  |
| --- | --- | --- | --- |
| **FIELD** | **DATA TYPE** | **SIZE** | **DESCRIPTION** |
| applicationno | Int | 10 | Application No |
| Name | Varchar | 30 | Application name |
| Address | Varchar | 30 | Address |
| Contactno | Varchar | 10 | Contact number |
| Cutoff | Int | 10 | Cutoff mark, |

**TABLE NAME : ADMISSION**

**PRIMARY KEY : ADMISSIONNO**

**FOREIGN KEY : APPLICATION NO**

|  |  |  |  |
| --- | --- | --- | --- |
| **FIELD** | **DATA TYPE** | **SIZE** | **DESCRIPTION** |
| Admissionno | Int | 10 | Admission no |
| Applicationno | Int | 10 | Application no |
| Studentname | Varchar | 30 | Student name |
| Fname | Varchar | 30 | Father name |
| Dob | Date | 30 | Date of birth |
| Age | Int | 10 | Age |
| Gender | Varchar | 10 | Gender |
| 6subject mark | Int | 10 | Subjects mark |
| Cut off mark | Int | 10 | Cut off mark |
| Photo | Varchar | 10 | Student photo |
| Course | Varchar | 20 | course |

**TABLE NAME : COURSE**

**PRIMARY KEY : COURSEID**

|  |  |  |  |
| --- | --- | --- | --- |
| **FIELD** | **DATA TYPE** | **SIZE** | **DESCRIPTION** |
| Courseid | Int | 10 | Course id |
| Course name | Varchar | 30 | Course name |
| Shift | Varchar | 10 | Shift |
| Course fees | Int | 10 | Course fees |
| Stu.strength | Int | 10 | Student strength |

**TABLE NAME : HOSTEL**

**PRIMARY KEY : ADMISSIONNO**

|  |  |  |  |
| --- | --- | --- | --- |
| **FIELD** | **DATA TYPE** | **SIZE** | **DESCRIPTION** |
| Male/female hostel | Varchar | 30 | Male or female hostel |
| Room no | Int | 30 | Room no |
| Room fees | Int | 10 | Room fees |
| Admissionno | Int | 10 | Admission no |

1. **SAMPLE CODING**

<html>

<head>

<link rel="stylesheet" type="text/css" href="/stylesheets/bootstrap.css">

<link rel="stylesheet" type="text/css" href="/stylesheets/login.css">

<link rel="stylesheet" type="text/css" href="/stylesheets/sweetalert.css">

<script src="/javascripts/jquery.js"></script>

<script src="/javascripts/bootstrap.js"></script>

<script src="/javascripts/sweetalert.js"></script>

<script src="/javascripts/src/login.js"></script>

</head>

<body>

<div class="container">

<div class="row">

<div class="col-sm-6 col-md-4 col-md-offset-4">

<h1 class="text-center login-title">Sign in to continue Pooling Vote</h1>

<div class="account-wall">

<img class="profile-img" src="https://lh5.googleusercontent.com/-b0-k99FZlyE/AAAAAAAAAAI/AAAAAAAAAAA/eu7opA4byxI/photo.jpg?sz=120"

alt="">

<input type="text" name="username" class="form-control" placeholder="Username" required autofocus>

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

<button class="btn btn-lg btn-primary btn-block" id="login()" onclick="login()">

Sign in</button>

</div>

</div>

</div>

</div>

</body>

</html>

<html>

<head>

<link rel="stylesheet" type="text/css" href="/stylesheets/bootstrap.css">

<link rel="stylesheet" type="text/css" href="/stylesheets/admin.css">

<link rel="stylesheet" type="text/css" href="/stylesheets/sweetalert.css">

<script src="/javascripts/jquery.js"></script>

<script src="/javascripts/bootstrap.js"></script>

<script src="/javascripts/sweetalert.js"></script>

<script src="/javascripts/chart.js"></script>

<!--<script type="text/javascript" src="https://www.gstatic.com/charts/loader.js"></script>-->

<script src="/javascripts/src/admin.js"></script>

</head>

<body>

<div class="container">

<div class="row">

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

<!-- Nav tabs -->

<div class="card">

<ul class="nav nav-tabs" role="tablist">

<li role="presentation" class="active"><a href="#profile" aria-controls="profile" role="tab" data-toggle="tab">Profile</a></li>

<li role="presentation"><a href="#candidate" aria-controls="candidate" role="tab" data-toggle="tab">Candidate</a></li>

<li role="presentation"><a href="#voter" aria-controls="voter" role="tab" data-toggle="tab">Voter</a></li>

<li role="presentation"><a href="#report" aria-controls="report" role="tab" data-toggle="tab">Report</a></li>

<li role="presentation"><a onclick="window.location.href='/'" style="cursor:pointer" aria-controls="report" role="tab" data-toggle="tab">Logout</a></li>

</ul>

<div class="tab-content">

<th>Count</th>

</tr>

</thead>

<tbody id="collegesec ">

</tbody>

</table>

</div>

<div class ="col-sm-6 ">

<h1 style="color:BLUE ">Finance Secretary</h1>

<table class="table table-bordered ">

<thead>

<tr>

<th>Rank</th>

<th>Name</th>

<th>Count</th>

</tr>

</thead>

<tbody id="financesec ">

</tbody>

</table>

</div>

<div class ="col-sm-6 ">

<h1 style="color:BLUE ">Treasurer</h1>

<table class="table table-bordered ">

<thead>

<tr>

<th>Rank</th>

<th>Name</th>

<th>Count</th>

</tr>

</thead>

<tbody id="treasurer ">

</tbody>

</table>

</div>

<div class ="col-sm-6 ">

<h1 style="color:BLUE ">Sports Secretary</h1>

<table class="table table-bordered ">

<thead>

<tr>

<th>Rank</th>

<th>Name</th>

<th>Count</th>

</tr>

</thead>

<tbody id="sportssec ">

</tbody>

</table>

</div>

</div>

</div>

</div>

</div>

</div>

</div>

<!-- Modal -->

<div id="myModal " class="modal fade " role="dialog ">

<div class="modal-dialog ">

<!-- Modal content-->

<div class="modal-content ">

<div class="modal-header ">

<button type="button " class="close " data-dismiss="modal ">&times;</button>

<h4 class="modal-title "> Deatail</h4>

</div>

<div class="modal-body ">

<div class="row ">

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

<div class="well well-sm ">

<div class="row ">

<div class="col-sm-6 col-md-4 ">

<img id="modalimg " src="http://placehold.it/380x500 " alt=" " class="img-rounded img-responsive " />

</div>

<div class="col-sm-6 col-md-8 ">

<h4 id="modname "></h4>

<small><cite id="modmobile " title="San Francisco, USA "></cite></small>

<p>

<i class="glyphicon glyphicon-globe "></i><a id="modpost "></a>

<br />

<i class="glyphicon glyphicon-envelope "></i><a id="modemail "></a>

<br />

<i class="glyphicon glyphicon-gift "></i><a id="modaddress "></a>

</p>

<!-- Split button -->

</div>

</div>

</div>

</div>

</div>

</div>

<div class="modal-footer ">

<button type="button " class="btn btn-default " data-dismiss="modal ">Close</button>

</div>

</div>

</div>

</div>

</body>

</html>

var candidateList;

var voterList;

var final\_id;

$(document).ready(function(){

var url\_string = window.location.href;

var url = new URL(url\_string);

final\_id = url.searchParams.get("id ");

$("#profileimg ").attr('src',"/images/user/ "+final\_id+".jpg ");

ViewCandidateList();

setProfile(final\_id);

ViewVoterList();

setReport();

});

function setReport(){

var chair='';

var vicechair='';

var collegesec='';

var financesec='';

var treasurer='';

var sportssec='';

$.ajax({

type:"GET ",

url:"/api/votreport ",

success: function(datas) {

var ch=0,vc=0,cs=0,fs=0,ts=0,ss=0;

for(var i=0;i<datas.length;i++){

var data = datas[i];

if(data.postid==1){

ch++;

chair=chair+`<tr><td>`+ch+`</td>

<td>`+data.firstname+`</td>

<td>`+data.count+`</td></tr>`;

}

if(data.postid==2){

vc++;

vicechair=vicechair+`<tr><td>`+vc+`</td>

<td>`+data.firstname+`</td>

<td>`+data.count+`</td></tr>`;

}

if(data.postid==3){

cs++;

collegesec=collegesec+`<tr><td>`+cs+`</td>

<td>`+data.firstname+`</td>

<td>`+data.count+`</td></tr>`;

}

if(data.postid==4){

fs++;

financesec=financesec+`<tr><td>`+fs+`</td>

<td>`+data.firstname+`</td>

<td>`+data.count+`</td></tr>`;

}

if(data.postid==5){

ts++;

treasurer=treasurer+`<tr><td>`+ts+`</td>

<td>`+data.firstname+`</td>

<td>`+data.count+`</td></tr>`;

}

if(data.postid==6){

ss++;

sportssec=sportssec+`<tr><td>`+ss+`</td>

<td>`+data.firstname+`</td>

<td>`+data.count+`</td></tr>`;

}

}

$("#chairman ").html(chair);

$("#vicechairman ").html(vicechair);

$("#collegesec ").html(collegesec);

$("#financesec ").html(financesec);

$("#treasurer ").html(treasurer);

$("#sportssec ").html(sportssec);

},

});

}

function setProfile(id){

$.ajax({

type:"POST ",

url:"/api/getprofile ",

data:{id:id},

success: function(datas) {

var data = datas[0];

$("#proname ").text(data['firstname']+" "+data['lastname']);

$("#promobile ").text(data['mobile']);

$("#proemail ").text(data['email']);

$("#proaddress ").text(data['address']);

},

});

}

function addCandidate(){

swal({

title: "Are you sure? ",

text: "You want to add Candidate! ",

type: "warning ",

showCancelButton: true,

confirmButtonClass: "btn-danger ",

confirmButtonText: "Yes, Sumbit it! ",

cancelButtonText: "No, cancel! ",

closeOnConfirm: false,

closeOnCancel: false

},

function(isConfirm) {

if (isConfirm) {

var firstname=$("input[name=canfirstname] ").val().trim();

var lastname=$("input[name=canlastname] ").val().trim();

var mobile=$("input[name=canmobile] ").val().trim();

var email=$("input[name=canemail] ").val().trim();

var address=$("input[name=canaddress] ").val().trim();

var post=$('#canoption').val().trim();

if(firstname.length==0 || lastname.length==0 || mobile.length==0 || email.length==0 || address.length==0 || post.length==0){

swal( "Required Fields ", 'Should Enter all fields!', 'warning');

}else{

$.ajax({

type:"POST ",

url:"/api/addCandidate ",

data:{firstname:firstname,lastname:lastname,mobile:mobile,email:email,address:address,post:post},

success: function(datas) {

swal("Success! ", "Your Record has been deleted. ", "success ");

$("input[name=canfirstname] ").val(" ");

$("input[name=canlastname] ").val(" ");

$("input[name=canmobile] ").val(" ");

$("input[name=canemail] ").val(" ");

$("input[name=canaddress] ").val(" ");

ViewCandidateList();

},

});

}

} else {

swal("Cancelled ", "Your record is note Added :) ", "error ");

}

});

}

function ViewCandidateList(){

//

$.ajax({

type:"POST ",

url:"/api/viewCandidates ",

success: function(datas) {

var html='';

candidateList=datas;

for(var i=0;i<datas.length;i++){

var data = datas[i];

html =html+ `<li><img onclick="deletecandidate( '`+i+`') " style="float:right " src="/images/delete.png "></img><a onclick="modal( '`+i+`', 'candidate') " style="cursor:pointer " data-toggle="modal ">`+data['firstname']+` `+data['lastname']+`</a></li>`;

}

$("#myUL ").html(html);

},

});

}

function modal(pos,path){

var postname='';

if(candidateList[pos]['post']==1){

postname="Chairman ";

}else if(candidateList[pos]['post']==2){

postname="Vice Chairman ";

}

else if(candidateList[pos]['post']==3){

postname="College Secretary ";

}

else if(candidateList[pos]['post']==4){

postname="Finance Secretary ";

}

else if(candidateList[pos]['post']==5){

postname="Treasurer ";

}

else if(candidateList[pos]['post']==6){

postname="Sports Secretary ";

}

$("#modalimg ").attr('src',"/images/ "+path+"/ "+candidateList[pos]['userid']+".jpg ");

$("#modname ").text(candidateList[pos]['firstname']+' '+ candidateList[pos]['lastname']);

$("#modmobile ").text(candidateList[pos]['mobile']);

$("#modpost ").text(postname);

$("#modemail ").text(candidateList[pos]['email']);

$("#modaddress ").text(candidateList[pos]['address']);

$('#myModal').modal('show');

}

function deletecandidate(pos){

swal({

title: "Are you sure? ",

text: "You will not be able to recover Candidate detail! ",

type: "warning ",

showCancelButton: true,

confirmButtonClass: "btn-danger ",

confirmButtonText: "Yes, delete it! ",

cancelButtonText: "No, cancel plx! ",

closeOnConfirm: false,

closeOnCancel: false

},

function(isConfirm) {

if (isConfirm) {

$.ajax({

type:"POST ",

data:{id:candidateList[pos]['userid']},

url:"/api/deleteCandidates ",

success: function(datas) {

swal("Deleted! ", "Your imaginary file has been deleted. ", "success ");

ViewCandidateList();

},

});

} else {

swal("Cancelled ", "Your record is safe :) ", "error ");

}

});

}

function addVoter(){

swal({

title: "Are you sure? ",

text: "You want to add Voter! ",

type: "warning ",

showCancelButton: true,

confirmButtonClass: "btn-danger ",

confirmButtonText: "Yes, Sumbit it! ",

cancelButtonText: "No, cancel! ",

closeOnConfirm: false,

closeOnCancel: false

},

function(isConfirm) {

if (isConfirm) {

var firstname=$("input[name=votfirstname] ").val().trim();

var lastname=$("input[name=votlastname] ").val().trim();

var mobile=$("input[name=votmobile] ").val().trim();

var email=$("input[name=votemail] ").val().trim();

var address=$("input[name=votaddress] ").val().trim();

if(firstname.length==0 || lastname.length==0 || mobile.length==0 || email.length==0 || address.length==0 ){

swal( "Required Fields ", 'Should Enter all fields!', 'warning');

}else{

$.ajax({

type:"POST ",

url:"/api/addVoter ",

data:{firstname:firstname,lastname:lastname,mobile:mobile,email:email,address:address},

success: function(datas) {

swal("Success! ", "Your Record has been Register. ", "success ");

$("input[name=votfirstname] ").val(" ");

$("input[name=votlastname] ").val(" ");

$("input[name=votmobile] ").val(" ");

$("input[name=votemail] ").val(" ");

$("input[name=votaddress] ").val(" ");

ViewVoterList();

},

});

}

} else {

swal("Cancelled ", "Your record is note Added :) ", "error ");

}

});

}

function myFunction() {

var input, filter, ul, li, a, i;

input = document.getElementById("myInput ");

filter = input.value.toUpperCase();

ul = document.getElementById("myUL ");

li = ul.getElementsByTagName("li ");

for (i = 0; i < li.length; i++) {

a = li[i].getElementsByTagName("a ")[0];

if (a.innerHTML.toUpperCase().indexOf(filter) > -1) {

li[i].style.display = " ";

} else {

li[i].style.display = "none ";

}

}

}

1. **SAMPLE INPUT**
2. **SAMPLE OUTPUT**