**COLLEGE FINDER**

**ABSTRACT**

The project college selector system is an application where students can search and enroll for different colleges that they want. Student can search for the colleges through their system at anytime and anywhere. This system is developed for students looking out for admission in best colleges. Also it will help colleges to get more students being enrolled into the institute. The system will provide a list of colleges to students from which they can select their required college. After selecting the college, student has to give a test in order to enroll themselves for the admission. Students have to give the test and the results are displayed on the spot by the system to the student. The system also sends the result to respective colleges for evaluation purposes along with student’s details. Hence the initial procedure of conducting an admission can be automated through this system.

**1. INTRODUCTION**

A college finder system main objective of that students can finding the their suitable college via this project, the student who are finishing the school they will their college based on marks, its very useful for the students. Admin login and added college details and which could be automatically search and give result to the students. Everyone can using this software, it will be more comfortable to use it.. 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 Existing system of college finder the student has to search the whole college in web site but it will not give proper details about the college also it could not shown the specific college based on the student 12 th mark. This system is very hard to find the college details. Which is not possible to view the college details in same place. So the proposed system will help to avoid the drawbacks.

**2.1.1 DRAWBACKS:**

The existing system has the following drawbacks.

* The system does not check for the student’s identity as there can be fake entries
* There is no human interaction

**2.2 PROPOSED SYSTEM:**

The system automates the manual procedure of applying for admission in colleges . No need for students to go to different college for applying for admissions or enquiry. They can get a brief list of all the colleges at a single place. The system acts as a mediator between the student and college. It notifies the colleges of students enrolling for admission by emailing them so that it becomes easier for them to get students details and evaluate

**2.2.1 FEATURES:**

* Data is not easily lost
* Willing to work long hours
* It is easy to manage the system due to the high number of staff working

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. Admin login
2. Student login
3. College list and Selection
4. College details
5. Admission

**Admin login:**

The system will be under sole control of an admin who provides inputs to the system. Admin will update or add/delete different colleges from the system.

**Student login**:

Any student can login into the system by creating an account. He has to provide all the required information that is needed by a college by filling out a form.

**College list and Selection**:

After creating a login into the system, each student can see a list of colleges along with their details. Student can select different colleges as per their preference.

**College details**:

Student can view the college details, first select the college and view that college staff and student details

**Admission**:

Student choose their college and booking college admission. We can collect the student details from the application form. So which helps to collect 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 will help to find the college with your suitable mark which could be very easy to find your college and websites. Admin will create college details and marks. Any time admin can update college details.

**FUTURE ENHANCEMENT**

In college finder system we have only developed just search a college with based student marks. Which is very hard to use another options, so we have plan give more option for like student can take an admission via this application, which very helpful to everyone. It's take long to upgrade it so we can fully focusing to confirming the admission from this application. every student has an unique account and confirm their admission.

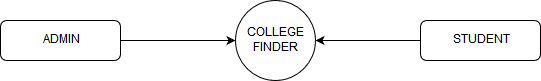
**BIBLIOGRAPHY**

* **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
* **Flanagan99**  
  David Flanagan, Java Foundation Classes in a Nutshell, O'Reilly, 1999.

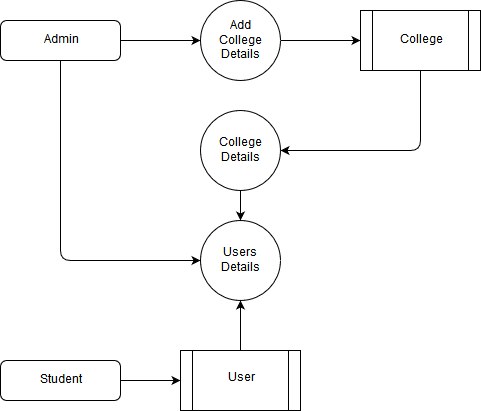
**APPENDICES**

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 : STUDENT**

**PRIMARY KEY : ID**

|  |  |  |  |
| --- | --- | --- | --- |
| **FIELD** | **DATA TYPE** | **SIZE** | **DESCRIPTION** |
| Id | Int | 10 | Student id |
| Name | Varchar | 30 | Application name |
| Address | Varchar | 30 | Address |
| Contactno | Varchar | 10 | Contact number |
| Email id | Varchar | 20 | Email id |
| Password | Varchar | 20 | Password |
| Mark | Int | 10 | Mark |
| School | Varchar | 20 | School name |

**TABLE NAME : COLLEGE**

**PRIMARY KEY : ID**

**FOREIGN KEY : INST\_ID**

|  |  |  |  |
| --- | --- | --- | --- |
| **FIELD** | **DATA TYPE** | **SIZE** | **DESCRIPTION** |
| Id | Int | 10 | College id |
| Inst | Int | 10 | Institution |
| Name | Varchar | 30 | College name |
| Address | Varchar | 50 | Address |
| Link | Varchar | 30 | Link |
| City | Varchar | 10 | City |
| District | Varchar | 10 | district |

1. **SAMPLE CODING**

<html>

<head>

<link href="https://maxcdn.bootstrapcdn.com/bootstrap/4.1.1/css/bootstrap.min.css" rel="stylesheet"/>

<link href="/bootstrap/datatables.css" rel="stylesheet"/>

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

</head>

<body>

<div class="container login-container">

<div class="row">

<div class="col-md-6 login-form-1">

<h3>College Finder</h3>

<div class="form-group">

<input id="username" type="text" class="form-control" placeholder="Username or Email address \*"

value=""/>

</div>

<div class="form-group">

<input id="password" type="password" class="form-control" placeholder="Your Password \*" value=""/>

</div>

<div class="form-group">

<input onclick="login()" type="button" class="btnSubmit" value="Login"/>

</div>

</div>

<div class="col-md-6 login-form-2">

<h3>Registration Form</h3>

<form>

<div class="form-group">

<input id="remail" type="text" class="form-control" placeholder="Your Email \*" value=""/>

</div>

<div class="form-group">

<input id="rpassword" type="password" class="form-control" placeholder="Your Password \*" value=""/>

</div>

<div class="form-group">

<input id="rname" type="text" class="form-control" placeholder="Your Name \*" value=""/>

</div>

<div class="form-group">

<input id="rschool" type="text" class="form-control" placeholder="Your School Name \*" value=""/>

</div>

<div class="form-group">

<input id="r12" type="number" class="form-control" placeholder="12th Mark \*" value=""/>

</div>

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

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

<!--<select class="form-control" id="inst\_1">-->

<!--</select>-->

<!--</div>-->

<!--</div>-->

<div class="form-group">

<input onclick="register()" type="button" class="btnSubmit" value="Register"/>

</div>

</form>

</div>

</div>

</div>

</body>

<script src="/bootstrap/jquery.min.js"></script>

<!--<script src="/home/home.js"></script>-->

<script src="/bootstrap/datatable.min.js"></script>

<script src="/bootstrap/datatable.bootstrap.min.js"></script>

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

</html>

</body>

<script src="/bootstrap/jquery.min.js"></script>

<!--<script src="/home/home.js"></script>-->

<script src="/bootstrap/datatable.min.js"></script>

<script src="/bootstrap/datatable.bootstrap.min.js"></script>

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

<script>

function register() {

var email = $("#remail").val();

var password = $("#rpassword").val();

var name = $("#rname").val();

var school= $("#rschool").val();

var mark = $("#r12").val();

// var inst= $("#inst\_1").children(':selected').attr('id');

$.ajax({

type: "POST",

data: {

email:email,

password:password,

name:name,

school:school,

mark:mark,

},

url: "/api/add\_customer", success: function (result) {

alert(result)

$("#remail").val("");

$("#rpassword").val("");

$("#rname").val("");

$("#rschool").val("");

$("#r12").val("");

}

});

}

function login() {

var username = $("#username").val();

var password = $("#password").val();

$.ajax({

type: "POST",

data: {

username: username,

password: password,

},

url: "/api/login", success: function (result) {

if (result == "admin") {

window.location = "/home";

} else if (result.length > 10) {

alert(result)

} else {

window.location = "/student?id=" + result;

}

}

});

}

$.ajax({

type: "GET",

url: "/api/get\_institutions",

success: function (result) {

var html = ``;

for (var i in result) {

var data = result[i];

html += ` <option id="` + data.id + `">` + data.name + `</option>

`;

}

// $("#inst\_1").html(html)

}

});

</script>

</html>

<html>

<head>

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

<link href="/bootstrap/datatables.css" rel="stylesheet"/>

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

</head>

<body>

<!-- Remove dark nav -->

<nav class="nav dark-nav">

<div class="container">

<div class="nav-heading">

<button class="toggle-nav" data-toggle="open-navbar1"><i class="fa fa-align-right"></i></button>

<a class="brand" href="#">College Finder</a>

</div>

<div class="menu" id="open-navbar1">

<ul class="list">

<li><a href="#htab1" data-toggle="tab">Add College</a></li>

<li><a href="#htab2" data-toggle="tab">College Details</a></li>

<li><a href="#htab3" data-toggle="tab">Student Details</a></li>

<li><a href="/">Logout</a></li>

</ul>

</div>

</div>

</nav>

<div class="container">

<div class="row">

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

<div class="tab-content" style="margin: 30px">

<div role="tabpanel" class="tab-pane fade in active" id="htab1">

<h1>Add College</h1>

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

<div class="form-group">

<label>College Name:</label>

<input id="name1" type="text" class="form-control">

</div>

<div class="form-group">

<label>Min Total Mark:</label>

<input id="min\_mark1" type="text" class="form-control">

</div>

<div class="form-group">

<label>Link Adress:</label>

<input type="text" id="link\_address1" class="form-control">

</div>

<div class="form-group">

<label >Institution:</label>

<select id="inst\_1" class="form-control">

</select>

</div>

<div class="form-group">

<label>City:</label>

<input id="city1" type="text" class="form-control">

</div>

<div class="form-group">

<label>District:</label>

<input id="district1" type="text" class="form-control">

</div>

<div class="form-group">

<label>Pincode:</label>

<input id="pincode1" type="text" class="form-control">

</div>

<div class="form-group">

<label>Full Address:</label>

<input id="address1" type="text" class="form-control">

</div>

<button onclick="addCollege()" type="button" class="btn btn-danger">Add Colleges</button>

</div>

</div>

<div role="tabpanel" class="tab-pane fade" id="htab2">

<h1>Search Colleges</h1>

<div class="form-group">

<label>Select District:</label>

<select onchange="getCollege()" class="form-control" id="district2">

</select>

</div>

<div class="form-group">

<label>Select College:</label>

<select class="form-control" id="college2">

</select>

</div>

<div class="form-group">

<button onclick="getCollegeDetails();" type="button" class="btn btn-default">Search</button>

</div>

<div class="card" style="border: double;padding: 10px" id="college\_details\_body">

</div>

</div>

<div role="tabpanel" class="tab-pane fade in" id="htab3">

<h1>Student Details</h1>

<table class="table">

<thead>

<tr>

<th scope="col">#</th>

<th scope="col">Name</th>

<th scope="col">Email ID</th>

<th scope="col">School Name</th>

<th scope="col">Mark</th>

</tr>

</thead>

<tbody id="student\_details\_body">

</tbody>

</table>

</div>

<div role="tabpanel" class="tab-pane fade in" id="htab4">

<h1>tab 4</h1>

</div>

</div>

</div>

</div>

</div>

</body>

<script src="/bootstrap/jquery.min.js"></script>

<script src="/home/home.js"></script>

<script src="/bootstrap/datatable.min.js"></script>

<script src="/bootstrap/datatable.bootstrap.min.js"></script>

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

</html>

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