

COLLEGE ADMISSION MANAGEMENT SYSTEM

1. INTRODUCTION

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.

1.2 SYSTEM SPECIFICATIONS

1.2.1 Hardware configuration

Processor	:	Pentium -IV
Speed	:	1 GHz
Hard Disk Capacity	:	40GB
RAM Capacity	:	1GB RAM
CD-ROM Drive	:	52x speed
Keyboard	:	104 keys
Mouse	:	Logitech
Printer	:	HP3745 series DeskJet printer

1.2.2. Software specification

Operating System	:	Windows 7/8/10
Front End	:	JAVA
Back End	:	MYSQL

2. SYSTEM STUDY

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.

The feasibility of a proposed solution is evaluated in terms of its components. These components are:

- Economic feasibility
- Technical feasibility

Economic Feasibility

The economic feasibility 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.

Technical Feasibility

The technical feasibility 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 modest requirements, as only minimal or no changes are required for implementing this system.

2.1. EXISTING SYSTEM

The academic Management has to handle records for many numbers 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

- It's automated computerized web based software system
- It uses technologies like html.
- It's easy to operate
- Attractive user interface

3. SYSTEM DESIGN AND DEVELOPMENT

Design is concerned with identifying software components specifying relationship Among components. Specifying software structure and providing blue print for the document phase. Modularity is one of the desirable properties of large systems. It implies that the system is divided into several parts. In such a manner, the interaction between parts is Minimal clearly specified. Design will explain software components in details. This will help the implementation of the system. Moreover, this will guide the further changes in the system to satisfy the further requirements.

The design document describes how to transform, the requirement and the functional design into more technical system design specification. This design involves conceiving and planning out in the mind and making a drawing pattern or sketch of it. It includes type of activities, External Design, Architectural Design and Detailed Design. The architectural design and detailed design collectively referred to as internal design.

The external design involves specifying the externally observable characteristics of a software product and the internal design involves specifying the internal structure and processing details of the system. The fundamental concept of the design includes abstraction structure, information hiding Modularity, concurrency, verification and design aesthetics.

3.1 FILE DESIGN

In computing, a file design (or file system) is used to control how data is stored and retrieved. Without a file system, information placed in a storage area would be one large body of data with no way to tell where one piece of information stops and the next begins. By separating the data into individual pieces, and giving each piece a name, the information is easily separated and identified. Taking its name from the way paper-based information systems are named, each group of data is called a "file". The structure and logic rules used to manage the groups of information and their names are called a "file system".

Some file systems are used on local data storage devices; others provide file access via a network protocol. Some file systems are "virtual", in that the "files" supplied are computed on request or are merely a mapping into a different file system used as a backing store. The file system manages access to both the content of files and the metadata about those files. It is responsible for arranging storage space; reliability, efficiency, and tuning with regard to the physical storage medium are important design considerations.

3.2 INPUT DESIGN

The input design is the process of entering data to the system. The input design goal is to enter to the computer as accurate as possible. Here inputs are designed effectively so that errors made by the operations are minimized.

The inputs to the system have been designed in such a way that manual forms and the inputs are coordinated where the data elements are common to the source document and to the input. The input is acceptable and understandable by the users who are using it.

Input design is the process of converting user-originated inputs to a computer-based format input data are collected and organized into group of similar data. Once identified, appropriate input media are selected for processing.

The input design also determines the user to interact efficiently with the system. Input design is a part of overall system design that requires special attention because it is the common source for data processing error. The goal of designing input data is to make entry easy and free from errors.

Input design is the process of connecting the user-originated inputs into a computer to used format.

The goal of the input design is to make the data entry logical & free from errors.

3.3 OUTPUT DESIGN

Output design is the process of converting computer data into hard copy that is understood by all. The various outputs have been designed in such a way that they represent the same format that the office and management used to.

Computer output is the most important and direct source of information to the user. Efficient, intelligible output design should improve the systems relationships with the user and help in decision making. A major form of output is the hardcopy from the printer.

Output requirements are designed during system analysis. A good starting point for the output design is the Data Flow Diagram (DFD). Human factors educe issues for design involves addressing internal controls to ensure readability.

The output form in the system is either by screen or by hard copies. Output design aims at communicating the results of the processing of the users. The reports are generated to suit the needs of the users. The reports have to be generated with appropriate levels.

All reports are output formats, maintained details can be reported over crystal reports, this project sustain following reports

3.4 DATABASE DESIGN

The most important consideration in designing the database is how information will be used.

The main objectives of designing a database are:

Data Integration

In a database, information from several files are coordinated, accessed and operated upon as though it is in a single file. Logically, the information are centralized, physically, the data may be located on different devices, connected through data communication facilities.

Data Integrity

Data integrity means storing all data in one place only and how each application to access it. This approach results in more consistent information, one update being sufficient to achieve a new record status for all applications, which use it. This leads to less data redundancy; data items need not be duplicated; a reduction in the direct access storage requirement.

Data Independence

Data independence is the insulation of application programs from changing aspects of physical data organization. This objective seeks to allow changes in the content and organization of physical data without reprogramming of applications and to allow modifications to application programs without reorganizing the physical data.

The tables needed for each module were designed and the specification of each and every column was given based on the records and details collected during record specification of the system study.

3.5 SYSTEM DEVELOPMENT

The key to control maintenance costs is to design systems that are easy to change, so the link between development and maintenance is very strong. Many of the analysis and design methodologies, tools, and techniques employed during system development can be applied to system maintenance, but there are significant differences between development and maintenance. Maintainability is the ease with which software can be understood, corrected, adopted and enhanced.

3.5.1 DESCRIPTION OF MODULES

To develop this project several steps should be followed. There are various modules in this proposed system they are listed below.

Application Module

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

A hostel module is used 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.

4. TESTING AND IMPLEMENTATION

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.

Unit Testing

Unit testing focuses verification effort on the smallest unit of software the module. Using the detailed design and the process specification testing is done to registration by the user with in the boundary of the Login module. The login form receives the username and password details and validates the value with the database. If valid, the home page is displayed.

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.

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:

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.

IMPLEMENTATION

System implementation is the stage of the project that the theoretical design is turned into a working system. If the implementation stage is not properly planned and controlled, it can cause error. Thus it can be considered to be the most crucial stage in achieving a successful new system and in giving the user confidence that the new system will work and be effective.

Normally this stage involves setting up a coordinating committee, which will act as a sounding board for ideas; complaints and problem. The first task is implementation planning; i.e., deciding on the methods and time scale to be adopted. Apart from planning two major task of preparing for implementation are, education takes place much earlier in the project; at the implementation stage the emphasis must be on training in new skills to give staff confidence they can use the system. Once staff has been trained, the system can be tested.

After the implementation phase is completed and the user staff is adjusted to the changes created by the candidate system, evaluation and maintenance is to bring the new system to standards.

5. CONCLUSION

The college admission system is a complex process that involves various factors such as academic performance, extracurricular activities, essays, recommendation letters, and standardized test scores. The goal of the system is to identify the most qualified and deserving candidates for admission to colleges and universities.

While the college admission system is designed to be fair and objective, there are certain criticisms and concerns that have been raised. For example, some argue that standardized tests are biased and do not accurately measure a student's potential or readiness for college. Others argue that the process favors students from privileged backgrounds who have access to better resources and opportunities.

Despite these criticisms, the college admission system remains an important part of the higher education system. Colleges and universities must continue to strive for a fair and equitable process that considers a wide range of factors and provides opportunities for students from all backgrounds to succeed..

BIBLIOGRAPY

Books Referred:

- Bloch, Joshua. Effective Java: Programming Language Guide. Addison-Wesley, 2017.
- Eckel, Bruce. Thinking in Java. Prentice Hall, 2006.
- Freeman, Eric, and Elisabeth Robson. Head First Java. O'Reilly Media, 2005.
- Horstmann, Cay S. Core Java Volume I--Fundamentals. Prentice Hall, 2018.
- Lea, Doug. Concurrent Programming in Java: Design Principles and Patterns. Addison-Wesley, 2000.
- Walrath, Kathy, et al. The Java Tutorial: A Short Course on the Basics. Addison-Wesley Professional, 2018.

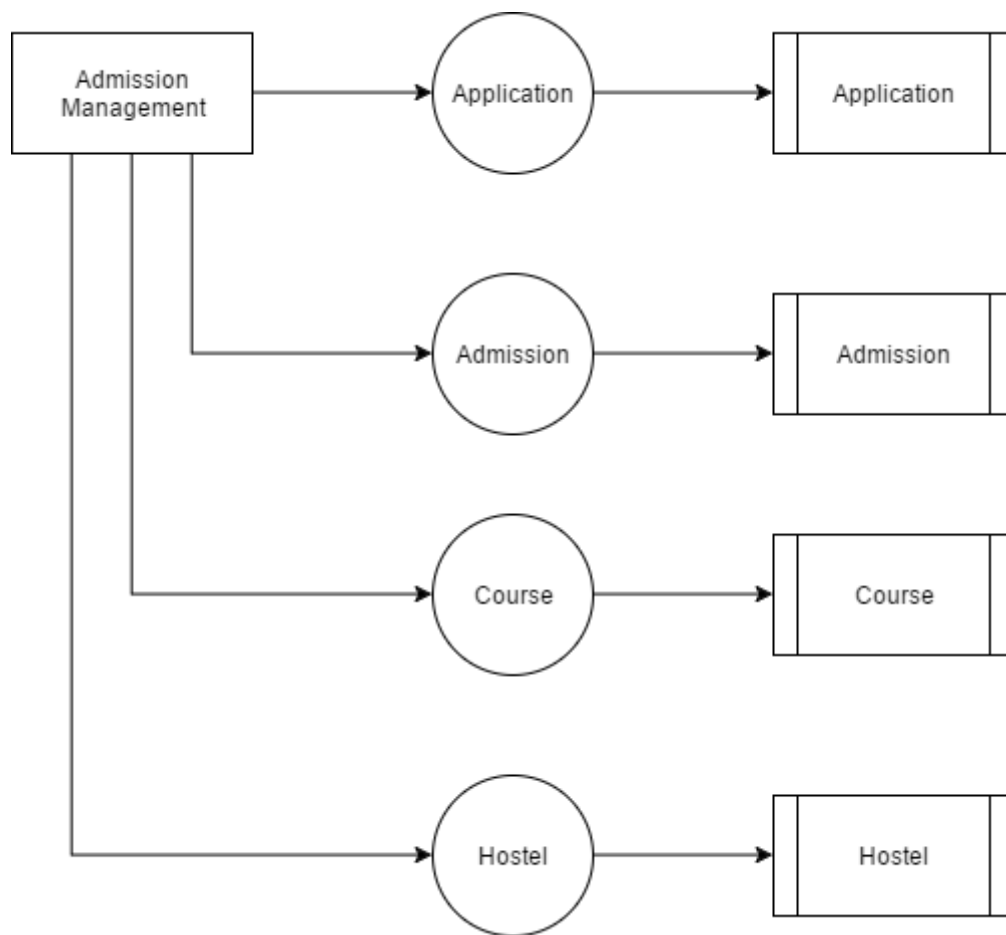
Websites:

- Baeldung. "Java Tutorials and Articles." Baeldung, 2023, <https://www.baeldung.com/java-tutorials>.
- GeeksforGeeks. "Java Programming Language." GeeksforGeeks, 2023, <https://www.geeksforgeeks.org/java/>.
- Tutorials Point. "Java Tutorial." Tutorials Point, 2023, <https://www.tutorialspoint.com/java/index.htm>.
- Vogella. "Java Tutorials." Vogella, 2023,

APPENDICES

A. Data Flow Diagram

Level 0



B. TABLE STRUCTURE**Table Name** : Admin**Primary Key** : Id**Table Description** : This table is used to maintain the details about admin

FIELD	DATA TYPE	SIZE	DESCRIPTION
Id	Int	10	Admin id
Username	Varchar	30	Admin username
password	Varchar	30	Admin password

TABLE NAME : APPLICATION**PRIMARY KEY : id**

FIELD	DATA TYPE	SIZE	DESCRIPTION
Id	Int	10	Application id
Firstname	Varchar	30	First name
Lastname	Varchar	30	Lastname
Fathername	Varchar	30	Fathername
Mothername	Varchar	30	Mother name
Age	Varchar	10	Age
Gender	Varchar	10	Gender
Mobile	Varchar	10	Mobile number
Email	Varchar	30	Email id
Pincode	Varchar	10	Pincode

Table name : Admission

Primary key : Admission Id

Foreign key : Application, Course

FIELD	DATA TYPE	SIZE	DESCRIPTION
Admission Id	Int	10	Admission id
Application Id	Integer	10	Application id
Course Id	Integer	10	Course id
Amount	Integer	10	Amount
Firstname	Varchar	30	First name
Lastname	Varchar	30	Lastname
Fathername	Varchar	30	Fathername
Mothername	Varchar	30	Mother name
Age	Varchar	10	Age
Gender	Varchar	10	Gender
Mobile	Varchar	10	Mobile number
Email	Varchar	30	Email id
Pincode	Varchar	10	Pincode
Sub1	Varchar	10	Subject 1
Sub2	Varchar	10	Subject 2
Sub3	Varchar	10	Subject 3
Sub4	Varchar	10	Subject 4
Sub5	Varchar	10	Subject 5
Address	Varchar	10	Address
Dob	Varchar	10	Dob
Course name	Varchar	10	Course name
Cut off	Varchar	10	cutoff

Table name : Course**Primary key : Course Id**

FIELD	DATA TYPE	SIZE	DESCRIPTION
Course Id	Int	10	Course Id
Course name	Varchar	30	Course name
Amount	Integer	10	amount

Table name : Hostel**Primary key : Hostel Id****Foreign key : Application, Course**

FIELD	DATA TYPE	SIZE	DESCRIPTION
Hostel Id	Int	10	Admission id
Application	Integer	10	Application id
Room no	Integer	10	Room no
Fees	Integer	10	Fees
Firstname	Varchar	30	First name
Lastname	Varchar	30	Lastname
Fathername	Varchar	30	Fathername
Mothername	Varchar	30	Mother name
Age	Varchar	10	Age
Gender	Varchar	10	Gender
Mobile	Varchar	10	Mobile number
Email	Varchar	30	Email id
Pincode	Varchar	10	Pincode

B. Sample Coding

```
<head>
<link href="/bootstrap/bootstrap.css" rel="stylesheet"/>
<link href="/bootstrap/datatables.css" rel="stylesheet"/>
<link rel="stylesheet" type="text/css" href="/home/login.css">
</head>
<body>
<div id="login">
<h3 class="text-center text-white pt-5">College Admission
</h3>
<div class="container" style="margin-left: 250px">
<div id="login-row" class="row justify-content-center align-items-center">
<div id="login-column" class="col-md-6">
<div id="login-box" class="col-md-12">
<h3 class="text-center text-info">Login</h3>
<div class="form-group">
<label for="username" class="text-info">Username:</label><br>
<input id="username" type="text" name="username" id="username" class="form-control">
</div>
<div class="form-group">
<label for="password" class="text-info">Password:</label><br>
<input id="password" type="text" name="password" id="password" class="form-control">
</div>
<input onclick="test()" type="button" id="login1" name="submit" class="btn btn-info btn-md"
value="submit">
</div>
</div>
</div>
</div>
</div>
```

```
</body>
<script src="/bootstrap/jquery.min.js"></script>
<script src="/home/login.js"></script>
<script src="/bootstrap/datatables.min.js"></script>
<script src="/bootstrap/datatables.bootstrap.min.js"></script>
<script src="/bootstrap/bootstrap.min.js"></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>
<div class="container">
<div class="row">
<div class="col-sm-12">
<nav class="navbar navbar-findcond navbar-fixed-top">l
<div class="container">
<div class="navbar-header">
<button type="button" class="navbar-toggle collapsed" data-toggle="collapse" data-target="#navbar">
<span class="sr-only">Toggle navigation</span>
<span class="icon-bar"></span>
<span class="icon-bar"></span>
<span class="icon-bar"></span>
</button>
<a class="navbar-brand" href="#">Admission Management</a>
</div>
<div class="collapse navbar-collapse" id="navbar">
<ul class="nav navbar-nav navbar-right">
<li class="dropdown">
<a href="#htab1" data-toggle="tab" class="dropdown-toggle" role="button" aria-expanded="false">
```

```

<i class="fa fa-fw fa-bell-o"></i> Application </a>
</li>
<li class="dropdown">
<a href="#htab2" data-toggle="tab" class="dropdown-toggle" role="button" aria-expanded="false">
<i class="fa fa-fw fa-bell-o"></i> Add Course </a>
</li>
<li class="dropdown">
<a href="#htab3" data-toggle="tab" class="dropdown-toggle" role="button" aria-expanded="false">
<i class="fa fa-fw fa-bell-o"></i> Admission </a>
</li>
<li class="dropdown">
<a href="#htab4" data-toggle="tab" class="dropdown-toggle" role="button" aria-expanded="false">
<i class="fa fa-fw fa-bell-o"></i> Hostel </a>
</li>
<li class="dropdown">
<a href="/" class="dropdown-toggle" role="button" aria-expanded="false">
<i class="fa fa-fw fa-bell-o"></i> Logout </a>
</li>
</ul>
</div>
</div>
</nav>
<br>
<div class="tab-content" style="margin: 30px">
<div role="tabpanel" class="tab-pane fade in active" id="htab1">
<h1>Application Form</h1>
<div class="col-md-3">
<div class="form-group">
<label>First Name:</label>
<input id="firstname" type="text" class="form-control">
</div>
<div class="form-group">

```

```
<label>Last Name:</label>
<input id="lastname" type="text" class="form-control">
</div>
<div class="form-group">
<label>Father Name:</label>
<input id="fathername" type="text" class="form-control">
</div>
<div class="form-group">
<label>Mother Name:</label>
<input id="mothername" type="text" class="form-control">
</div>
<div class="form-group">
<label>Age</label>
<input id="age" type="text" class="form-control">
</div>
<div class="form-group">
<label for="gender">Gender:</label>
<select class="form-control" id="gender">
<option>Male</option>
<option>Female</option>
</select>
</div>
<div class="form-group">
<label>Mobile Number:</label>
<input id="mobile" type="text" class="form-control">
</div>
<div class="form-group">
<label>Email:</label>
<input id="email" type="text" class="form-control">
</div>
<div class="form-group">
<label>Pincode:</label>
```

```
<input type="text" id="pincode" class="form-control">
</div>
<button type="button" id="addstudent" class="btn btn-primary">Add Student</button>
</div>
<div class="col-md-6">
<table class="table">
<thead>
<tr>
<th scope="col">#</th>
<th scope="col">Application No</th>
<th scope="col">First Name</th>
<th scope="col">Last Lastname</th>
</tr>
</thead>
<tbody id="view_application">
</tbody>
</table>
</div>
</div>
<div role="tabpanel" class="tab-pane fade" id="htab2">
<h1>Add Course</h1>
<div class="col-md-3">
<div class="form-group">
<label>Course Name</label>
<input id="coursename" type="text" class="form-control">
</div>
<div class="form-group">
<label>Amount</label>
<input id="amount" type="text" class="form-control">
</div>
<button id="addcourse" type="button" class="btn btn-primary">Add Course</button>
</div>
```

```
<div class="col-md-6">
<table class="table">
<thead>
<tr>
<th scope="col">#</th>
<th scope="col">Course Name</th>
<th scope="col">Amount</th>
</tr>
</thead>
<tbody id="courseBody">
</tbody>
</table>
</div>
</div>
<div role="tabpanel" class="tab-pane fade in" id="htab3">
<h1>Admission</h1>
<div class="col-md-3">
<div class="form-group">
<label>Application Number:</label>
<input id="application_admission" type="text" class="form-control">
</div>
<div class="form-group">
<label>Select Course:</label>
<select class="form-control" id="admission_course">
</select>
</div>
<div class="form-group">
<label>Amount:</label>
<input id="amount3" type="text" class="form-control" disabled>
</div>
<div class="form-group">
<label>First Name:</label>
```



```
<input id="firstname3" type="text" class="form-control" disabled>
</div>
<div class="form-group">
<label>Last Name:</label>
<input type="text" id="lastname3" class="form-control" disabled>
</div>
<div class="form-group">
<label>DOB:</label>
<input type="text" id="dob3" class="form-control" >
</div>
<div class="form-group">
<label>Father Name:</label>
<input type="text" id="father3" class="form-control" disabled>
</div>
<div class="form-group">
<label>Mother Name:</label>
<input type="text" id="mother3" class="form-control" disabled>
</div>
<div class="form-group">
<label>Age</label>
<input type="text" id="age3" class="form-control" disabled>
</div>
<div class="form-group">
<label>Gender:</label>
<input type="text" id="gender3" class="form-control" disabled>
</div>
<div class="form-group">
<label>Mobile Number:</label>
<input type="text" id="mobile3" class="form-control" disabled>
</div>
<div class="form-group">
<label>Email:</label>
```

```
<input type="text" class="form-control" id="email3" disabled>
</div>
<div class="form-group">
<label>Pincode:</label>
<input type="text" class="form-control" id="pincode3" disabled>
</div>
<div class="form-group">
<label>Address:</label>
<input type="text" class="form-control" id="address3" >
</div>
<div class="form-group">
<label>Subject 1:</label>
<input type="text" class="form-control" id="sub1" >
</div>
<div class="form-group">
<label>Subject 2:</label>
<input type="text" class="form-control" id="sub2" >
</div>
<div class="form-group">
<label>Subject 3:</label>
<input type="text" class="form-control" id="sub3" >
</div>
<div class="form-group">
<label>Subject 4:</label>
<input type="text" class="form-control" id="sub4" >
</div>
<div class="form-group">
<label>Subject 5:</label>
<input type="text" class="form-control" id="sub5" >
</div>
<div class="form-group">
<label>Subject 6:</label>
```

```
<input type="text" class="form-control" id="sub6" >
```

```
</div>
```

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

```
<label>Cut Off:</label>
```

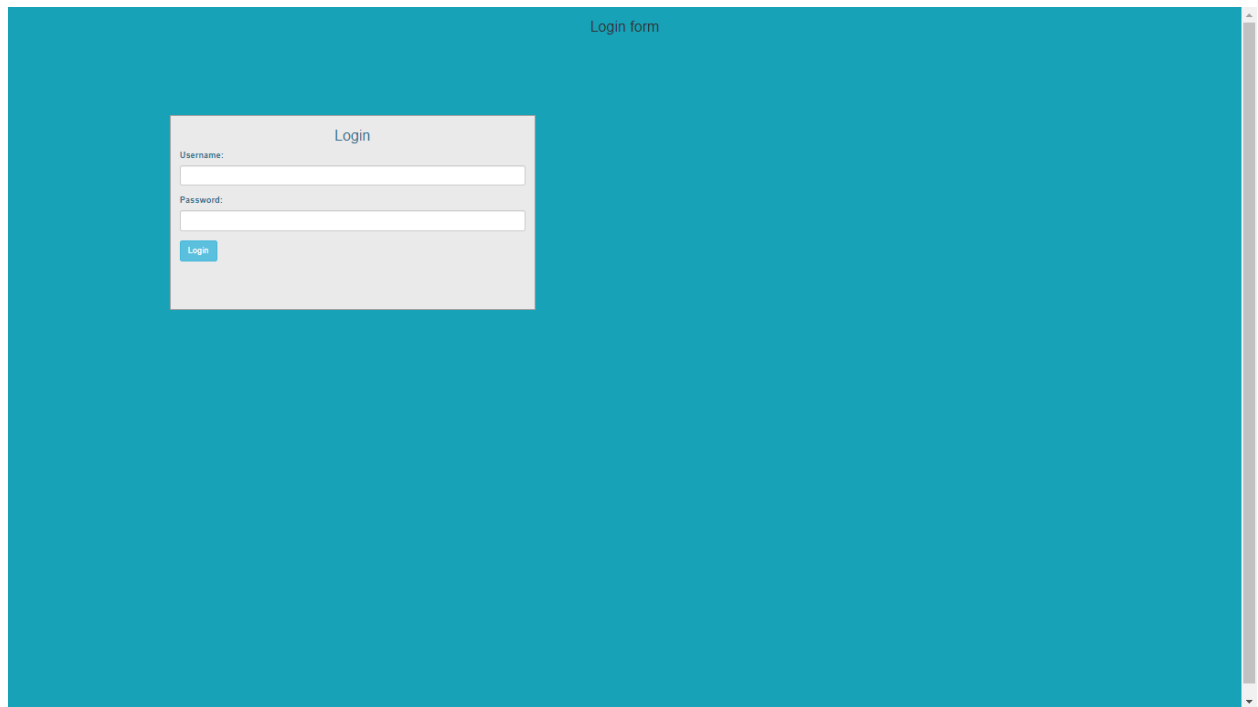
```
<input type="text" class="form-control" id="cutoff" >
```

```
</div>
```

```
<button id="addadmission" type="button" class="btn btn-primary">Add Admission</button>
```

D. SAMPLE INPUT

User Login page



The image shows a web browser window displaying a login page. The background is a solid teal color. In the top right corner, the text "Login form" is visible. Centered on the page is a light gray rectangular box containing the login form. The form has a title "Login" at the top. Below the title, there are two input fields: "Username:" and "Password:". Each field has a small label to its left and a text input area. Below the password field is a blue button with the text "Login".

User login page with user input

Login form

Login

Username:
admin

Password:

Login

Application Form Page

Admission Management

ApplicationAdd CourseAdmissionHostelLogout

Application Form

First Name:

Last Name:

Father Name:

Mother Name:

Age

Gender:

Male

Mobile Number:

Email:

Pincode:

Add Student

#

Application No

First Name

Last Lastname

Course Registration Page

Admission Management

ApplicationAdd CourseAdmissionHostelLogout

Add Course

Course Name

Amount

Add Course

#

Course Name

Amount

Admission Registration Page

Admission Management

ApplicationAdd CourseAdmissionHostelLogout

Admission

Application Number:

Select Course:

Amount:

First Name:

Last Name:

Father Name:

Mother Name:

Age

Gender:

Mobile Number:

Email:

Pincode:

Add Admission

#

Application No

First Name

Last Lastname

Course

Fees

Hostel Registration Page

Admission Management

ApplicationAdd CourseAdmissionHostelLogout

Hostel

Application Number:

Room No:

Hostel Fees:

First Name:

Last Name:

Father Name:

Mother Name:

Age

Gender:

Mobile Number:

Email:

Pincode:

Add Hostel

#

Application No

First Name

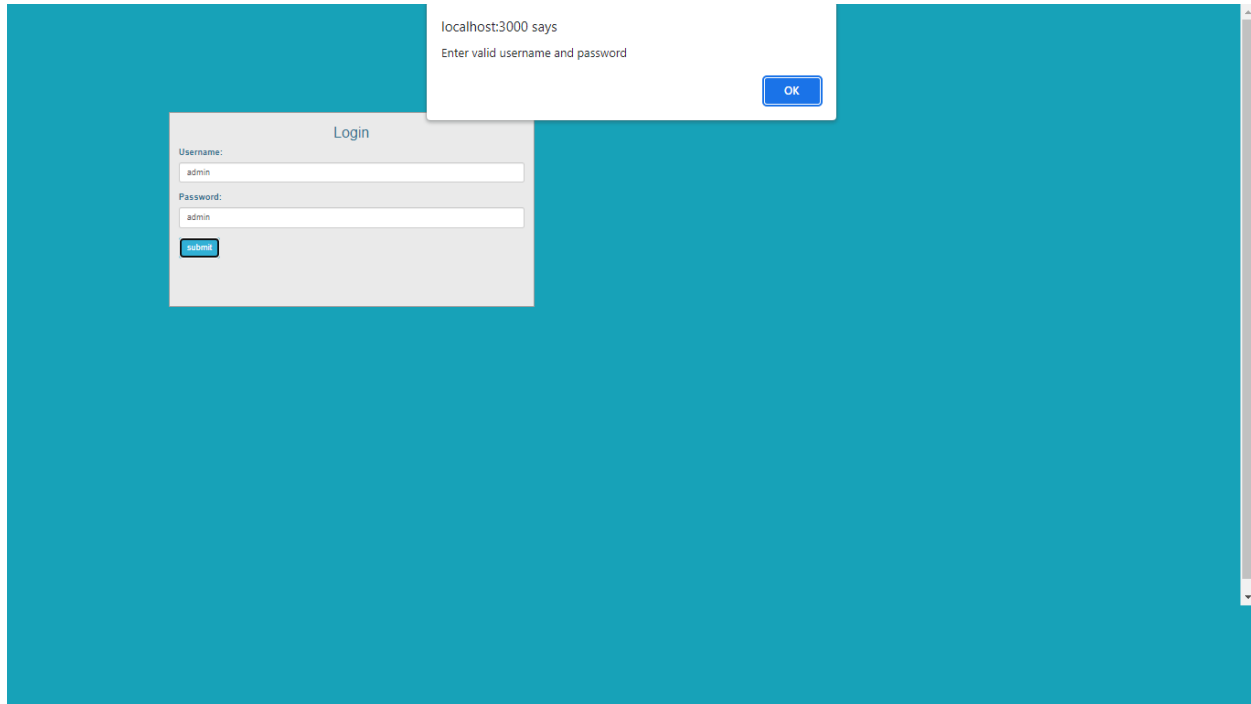
Last Lastname

Room No

Fees

E. SAMPLE OUTPUT

Output of Invalid username or password



Output of Application Form

Admission Management

Application Form

First Name:

Kowsalya

Last Name:

K

Father Name:

Ravi

Mother Name:

Suganthi

Age

25

Gender:

Female

Mobile Number:

8989878789

Email:

kowsalya@gmail.com

Pincode:

641002

Add Student

localhost:3000 says

Application has been created

OK

Application

Add Course

Admission

Hostel

Logout

Output of Application form result

Admission Management

ApplicationAdd CourseAdmissionHostelLogout

Application Form

First Name:

Last Name:

Father Name:

Mother Name:

Age

Gender:

Female

Mobile Number:

Email:

Pincode:

Add Student

#	Application No	First Name	Last Lastname
1	1	Kovsalya	K

Output of Course details

Admission Management

ApplicationAdd CourseAdmissionHostelLogout

Add Course

Course Name

Amount

Add Course

#	Course Name	Amount
1	BCA	\$25000

Output of Admission Result

Admission Management

ApplicationAdd CourseAdmissionHostelLogout

Admission

Application Number:

1

Select Course:

BCA

Amount:

First Name:

Last Name:

Father Name:

Mother Name:

Age

Gender:

Mobile Number:

Email:

Pincode:

Add Admission

#	Application No	First Name	Last Lastname	Course	Fees
1	1	Kowsalya	K	BCA	

Output of Hostel Details

Admission Management

ApplicationAdd CourseAdmissionHostelLogout

Hostel

Application Number:

Room No:

Hostel Fees:

First Name:

Last Name:

Father Name:

Mother Name:

Age

Gender:

Mobile Number:

Email:

Pincode:

Add Hostel

#	Application No	First Name	Last Lastname	Room No	Fees
1	1	Kovsalya	K	101	25000

