**STUDENT ACADEMIC MANAGEMENT SYSTEM**

**ABSTRACT**

* Student academic management system deals with the maintenance of the student’s attendance details. It is generates the attendance of the student on basis of presence in class. It is maintained on the daily basis of their attendance. the staffs will be  provided with the separate username & password to make the student’s status and also managing student internal and external mark details.
* Teachers can register students mark detail in marks registration screen then student can view their getting marks. Unauthorized can’t access these account. Student can easily find their updated marks.
* The purpose of developing attendance system is to computerized the tradition way of taking attendance. Another purpose for developing this software is to generate the report automatically at the end of the session or in the between of the session
* Over the years the manual attendance management has been carried across most of educational institutions. To overcome the problems of manual attendance, I have developed “web based attendance Management System”. Attendance Management System is based on web server, which can be implemented on any computer. In This application. The system communicates with database residing on a remote server. It calculates automatically, the attendance percentage of students without any manual paper-based work. The system facilitates the end users with interactive design and automated processing of attendance management.

1. **INTRODUCTION**

Main goal of this project is for managing college or schools, we can't managing these two organizations are manually, so we just moving manual to systematically, admin or teachers login and created student list then allocate that student to particular one class, then the teacher register attendance on every end of semester it will automatically calculate attendance percentage, the same way to the teachers register student marks after result, it will be automatically calculate student mark details. The project is aimed to develop by **JAVA and XML** 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:**

* System : Pentium IV 2.4 GHz.
* Hard Disk : 180 GB.
* Floppy Drive : 1.44 Mb.
* Ram : 8 GB.

**1.1.2 SOFTWARE SPECIFICATION:**

* Operating system : Windows 7,8.
* Front End : JAVA and XML
* Back End : SQL Server 2005

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 DISADVANTAGES:**

The existing system has the following drawbacks.

* All the details of the student are maintained in a single record.
* Managing and calculating student attendance percentage is very difficult.
* It’s take too much time to distribute student result.

**2.2 PROPOSED SYSTEM:**

In this proposed system can overcome in these all problems. Which makes very easy to use this application. Everyone can easily understand in this application.

**2.2.1 ADVANTAGES OF THE PROPOSED SYSTEM:**

* By developing the system we can attain the following features:
* Easy to handle and feasible
* Cost Reduction
* Fast and Convenient

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**

* Admin Module
* Staff Module
* Student Module

**MODULES DESCRIPTION:**

Admin:

The Admin is a person who run and manage the system, in this project the owner or director of the school has rights of admin. The Admin creates the structure of a system to creating standard and classroom.

* Admin Functionalities:
* Managing attendance
* Managing student marks
* Attendance report
* Mark report

**Staff :**

The main aim of the attendance system fulfill by staff. Staff can add student detail and fill the attendance daily for his division. The staff can only fill attendance for his division students and he can manage leaves and complains made by only his division student.

* **Staff Functionalities :**
* Add Student
* Fill Attendance
* Fill student marks

**Student :**

All student have unique username and password to access the system. After login into system student can view his attendance reports and manage his account. If any student has any problem regarding education he can directly make complain to his class teacher using this system. Student can apply for leave through the system and he can get reply from his class teacher about his leave status.

**Student Functionality :**

* Manage Account
* Attendance Reports
* Marks Reports

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 student registration module, attendance module, marks module are tested individually for example, student 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 student registration details and admin 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.

**5. CONCLUSION**

**CONCLUSION:**

This project has been designed to complete the requirement of the school system is an completely convert to mobile application. Which is an very smart way to run an organization. A whole process student and faculty can using an mobile application and parents can also noticed their child via application. Which may very help to very user friendly so every people can easily using this application.

**FUTURE ENHANCEMENT**

In future enhancement of this project is for we have plan to take an attendance daily period wise attendance report, which is more complex to big concept of academic management system, teachers take an attendance by every hours , suppose student take an leave an hour it's should be focused, so parents and teachers easily identify the up-to-date attendance process

**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.

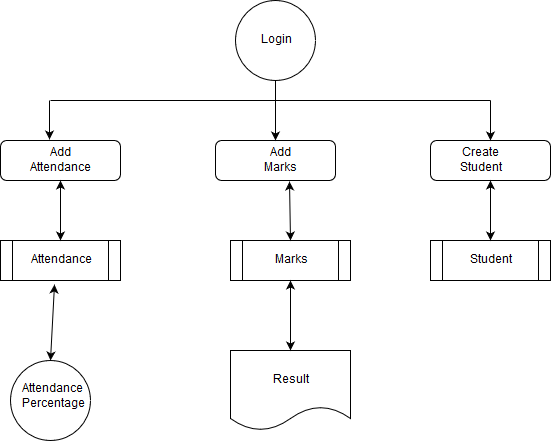
**APPENDIX**

**DATA FLOW DIAGRAM**

**LEVEL 0:**



**LEVEL 1:**



**TABLES:**

**Table Name : ADMIN**

**Primary\_key : ID**

**Description :**Student Table

|  |  |  |  |
| --- | --- | --- | --- |
| **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

**Foreign Key :** cid

**Description :** Student Table

|  |  |  |  |
| --- | --- | --- | --- |
| **FIELD NAME** | **DATA TYPE** | **SIZE** | **DESCRIPTION** |
| ID | INT | 10 | STUDENT ID |
| CID | INT | 10 | CLASS ID |
| ROLLNO | VARCHAR | 30 | ROLL NO |
| REGNO | VARCHAR | 30 | REGISTER NO |
| NAME | VARCHAR | 30 | STUDENT NAME |

**Table Name :** attendance

**Primary Key :** id

**Foreign Key :** cid,sid

**Description :** Attendance Table

|  |  |  |  |
| --- | --- | --- | --- |
| **FIELD NAME** | **DATA TYPE** | **SIZE** | **DESCRIPTION** |
| ID | INT | 10 | USER ID |
| CID | INT | 10 | CLASS ID |
| SID | INT | 10 | STUDENT ID |
| TOTAL | INT | 10 | TOTAL |
| PRESENT | INT | 10 | PRESENT |
| ABSENT | INT | 10 | ABSENT |
| AVG | INT | 10 | AVG |

**TABLES :**

**Table Name : marks**

**Primary Key :** id

**Foreign Key :** cid,sid

**Description :** Home work Table

|  |  |  |  |
| --- | --- | --- | --- |
| **FIELD NAME** | **DATA TYPE** | **SIZE** | **DESCRIPTION** |
| ID | INT | 10 | ID |
| CID | INT | 10 | CLASS ID |
| SID | INT | 10 | STUDENT ID |
| MARK1 | INT | 10 | MARK 1 |
| MARK 2 | INT | 10 | MARK 2 |
| MARK 3 | INT | 10 | MARK 3 |
| MARK 4 | INT | 10 | MARK 4 |
| MARK 5 | INT | 10 | MARK 5 |
| TOTAL | INT | 10 | TOTAL |
| AVG | INT | 10 | AVG |
| RESULT | VARCHAR | 10 | RESULT |

SAMPLE CODING

|  |
| --- |
| <html> |
|  |

|  |
| --- |
| <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="formWrapper"> |
|  |

|  |
| --- |
|  |
|  |

|  |
| --- |
| <div id="form"> |
|  |

|  |
| --- |
| <div class="logo"> |
|  |

|  |
| --- |
| <h1 class="text-center head">Student Academic Management system</h1> |
|  |

|  |
| --- |
| </div> |
|  |

|  |
| --- |
| <div class="form-item"> |
|  |

|  |
| --- |
| <p class="formLabel">Email</p> |
|  |

|  |
| --- |
| <input type="text" name="email" id="email" class="form-style" autocomplete="off"/> |
|  |

|  |
| --- |
| </div> |
|  |

|  |
| --- |
| <div class="form-item"> |
|  |

|  |
| --- |
| <p class="formLabel">Password</p> |
|  |

|  |
| --- |
| <input type="password" name="password" id="password" class="form-style"/> |
|  |

|  |
| --- |
|  |
|  |

|  |
| --- |
| </div> |
|  |

|  |
| --- |
| <div class="form-item"> |
|  |

|  |
| --- |
|  |
|  |

|  |
| --- |
| <input id="login" type="submit" class="login pull-right" value="Log In"> |
|  |

|  |
| --- |
|  |
|  |

|  |
| --- |
| <div class="clear-fix"></div> |
|  |

|  |
| --- |
| </div> |
|  |

|  |
| --- |
| </div> |
|  |

|  |
| --- |
| </div> |
|  |

|  |
| --- |
| </body> |
|  |

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

|  |
| --- |
| <script src="/home/login.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="navbar-wrapper"> |
|  |

|  |
| --- |
| <div class="container-fluid"> |
|  |

|  |
| --- |
| <nav class="navbar navbar-fixed-top"> |
|  |

|  |
| --- |
| <div class="container"> |
|  |

|  |
| --- |
| <div class="navbar-header"> |
|  |

|  |
| --- |
| <button type="button" class="navbar-toggle collapsed" data-toggle="collapse" data-target="#navbar" |
|  |

|  |
| --- |
| aria-expanded="false" aria-controls="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="#">Academic Management</a> |
|  |

|  |
| --- |
| </div> |
|  |

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

|  |
| --- |
| <ul class="nav navbar-nav"> |
|  |

|  |
| --- |
| <li><a href="#htab1" data-toggle="tab" class="">Create Class</a></li> |
|  |

|  |
| --- |
| <li><a href="#htab2" data-toggle="tab" class="">Add Student</a></li> |
|  |

|  |
| --- |
| <li><a href="#htab3" data-toggle="tab" class="">Add Attendance</a></li> |
|  |

|  |
| --- |
| <li><a href="#htab6" data-toggle="tab" class="">Attendance View</a></li> |
|  |

|  |
| --- |
| <li><a href="#htab4" data-toggle="tab" class="">Mark Entry</a></li> |
|  |

|  |
| --- |
| <li><a href="#htab5" data-toggle="tab" class="">View Marks</a></li> |
|  |

|  |
| --- |
|  |
|  |

|  |
| --- |
|  |
|  |

|  |
| --- |
| </ul> |
|  |

|  |
| --- |
| <ul class="nav navbar-nav pull-right"> |
|  |

|  |
| --- |
|  |
|  |

|  |
| --- |
| <li class=""><a href="/">Logout</a></li> |
|  |

|  |
| --- |
| </ul> |
|  |

|  |
| --- |
| </div> |
|  |

|  |
| --- |
| </div> |
|  |

|  |
| --- |
| </nav> |
|  |

|  |
| --- |
| <div class="tab-content" style="margin: 100px"> |
|  |

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

|  |
| --- |
| <h1>Create Class</h1> |
|  |

|  |
| --- |
|  |
|  |

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

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

|  |
| --- |
| <label>Class Name:</label> |
|  |

|  |
| --- |
| <input id="class\_name1" type="text" class="form-control"> |
|  |

|  |
| --- |
| </div> |
|  |

|  |
| --- |
| <button onclick="addClass()" id="deptadd1" type="button" class="btn btn-success">Create Class</button> |
|  |

|  |
| --- |
| </div> |
|  |

|  |
| --- |
|  |
|  |

|  |
| --- |
| </div> |
|  |

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

|  |
| --- |
| <h1>Add Student</h1> |
|  |

|  |
| --- |
|  |
|  |

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

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

|  |
| --- |
| <label>Name:</label> |
|  |

|  |
| --- |
| <input id="name2" type="text" class="form-control"> |
|  |

|  |
| --- |
| </div> |
|  |

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

|  |
| --- |
| <label>Roll No:</label> |
|  |

|  |
| --- |
| <input id="rollno2" type="text" class="form-control"> |
|  |

|  |
| --- |
| </div> |
|  |

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

|  |
| --- |
| <label>Reg No:</label> |
|  |

|  |
| --- |
| <input id="regno2" type="text" class="form-control"> |
|  |

|  |
| --- |
| </div> |
|  |

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

|  |
| --- |
| <label>Select Class:</label> |
|  |

|  |
| --- |
| <select id="class2" class="form-control"> |
|  |

|  |
| --- |
| </select> |
|  |

|  |
| --- |
| </div> |
|  |

|  |
| --- |
| <button id="add2" onclick="addStudent();" type="button" class="btn btn-success">Add Student</button> |
|  |

|  |
| --- |
| </div> |
|  |

|  |
| --- |
|  |
|  |

|  |
| --- |
| </div> |
|  |

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

|  |
| --- |
| <h1>Attendance Entry</h1> |
|  |

|  |
| --- |
|  |
|  |

|  |
| --- |
| <div class="col-sm-3"> |
|  |

|  |
| --- |
| <table class="table"> |
|  |

|  |
| --- |
| <tbody id="classlist3body"> |
|  |

|  |
| --- |
|  |
|  |

|  |
| --- |
| </tbody> |
|  |

|  |
| --- |
| </table> |
|  |

|  |
| --- |
| </div> |
|  |

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

|  |
| --- |
| <table class="table table-bordered"> |
|  |

|  |
| --- |
| <thead> |
|  |

|  |
| --- |
| <tr> |
|  |

|  |
| --- |
| <th>No</th> |
|  |

|  |
| --- |
| <th>Roll No</th> |
|  |

|  |
| --- |
| <th>Reg No</th> |
|  |

|  |
| --- |
| <th>Name</th> |
|  |

|  |
| --- |
| <th>No Of working days</th> |
|  |

|  |
| --- |
| <th>Present Days</th> |
|  |

|  |
| --- |
| </tr> |
|  |

|  |
| --- |
| </thead> |
|  |

|  |
| --- |
| <tbody id="studentlist\_body"> |
|  |

|  |
| --- |
|  |
|  |

|  |
| --- |
| </tbody> |
|  |

|  |
| --- |
| </table> |
|  |

|  |
| --- |
| <button id="attendance\_submit" type="button" class="btn btn-success">Add Attendance</button> |
|  |

|  |
| --- |
|  |
|  |

|  |
| --- |
| </div> |
|  |

|  |
| --- |
| <div class="col-sm-3"></div> |
|  |

|  |
| --- |
|  |
|  |

|  |
| --- |
| </div> |
|  |

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

|  |
| --- |
| <h1>Mark Entry</h1> |
|  |

|  |
| --- |
|  |
|  |

|  |
| --- |
| <div class="col-sm-3"> |
|  |

|  |
| --- |
| <table class="table"> |
|  |

|  |
| --- |
| <tbody id="classlist4body"> |
|  |

|  |
| --- |
|  |
|  |

|  |
| --- |
| </tbody> |
|  |

|  |
| --- |
| </table> |
|  |

|  |
| --- |
| </div> |
|  |

|  |
| --- |
| <div class="col-sm-9"> |
|  |

|  |
| --- |
| <table class="table table-bordered"> |
|  |

|  |
| --- |
| <thead> |
|  |

|  |
| --- |
| <tr> |
|  |

|  |
| --- |
| <th>No</th> |
|  |

|  |
| --- |
| <th>Reg No</th> |
|  |

|  |
| --- |
| <th>Roll No</th> |
|  |

|  |
| --- |
| <th>Name</th> |
|  |

|  |
| --- |
| <th>Mark1</th> |
|  |

|  |
| --- |
| <th>Mark2</th> |
|  |

|  |
| --- |
| <th>Mark3</th> |
|  |

|  |
| --- |
| <th>Mark4</th> |
|  |

|  |
| --- |
| <th>Mark5</th> |
|  |

|  |
| --- |
| </tr> |
|  |

|  |
| --- |
| </thead> |
|  |

|  |
| --- |
| <tbody id="studentlist4\_body"> |
|  |

|  |
| --- |
|  |
|  |

|  |
| --- |
| </tbody> |
|  |

|  |
| --- |
| </table> |
|  |

|  |
| --- |
| <button id="mark\_submit" type="button" class="btn btn-success">Add Mark</button> |
|  |

|  |
| --- |
| </div> |
|  |

|  |
| --- |
| </div> |
|  |

|  |
| --- |
| <div role="tabpanel" class="tab-pane fade in" id="htab5"> |
|  |

|  |
| --- |
| <h1>Mark View</h1> |
|  |

|  |
| --- |
|  |
|  |

|  |
| --- |
| <div class="col-sm-3"> |
|  |

|  |
| --- |
| <table class="table"> |
|  |

|  |
| --- |
| <tbody id="classlist5body"> |
|  |

|  |
| --- |
|  |
|  |

|  |
| --- |
| </tbody> |
|  |

|  |
| --- |
| </table> |
|  |

|  |
| --- |
| </div> |
|  |

|  |
| --- |
| <div class="col-sm-9"> |
|  |

|  |
| --- |
| <table class="table table-bordered"> |
|  |

|  |
| --- |
| <thead> |
|  |

|  |
| --- |
| <tr> |
|  |

|  |
| --- |
| <th>No</th> |
|  |

|  |
| --- |
|  |
|  |

|  |
| --- |
| <th>Roll No</th> |
|  |

|  |
| --- |
| <th>Reg No</th> |
|  |

|  |
| --- |
| <th>Name</th> |
|  |

|  |
| --- |
| <th>Mark1</th> |
|  |

|  |
| --- |
| <th>Mark2</th> |
|  |

|  |
| --- |
| <th>Mark3</th> |
|  |

|  |
| --- |
| <th>Mark4</th> |
|  |

|  |
| --- |
| <th>Mark5</th> |
|  |

|  |
| --- |
| <th>Total</th> |
|  |

|  |
| --- |
| <th>Average</th> |
|  |

|  |
| --- |
| <th>Result</th> |
|  |

|  |
| --- |
| </tr> |
|  |

|  |
| --- |
| </thead> |
|  |

|  |
| --- |
| <tbody id="student\_mark\_details"> |
|  |

|  |
| --- |
|  |
|  |

|  |
| --- |
|  |
|  |

|  |
| --- |
| </tbody> |
|  |

|  |
| --- |
| </table> |
|  |

|  |
| --- |
|  |
|  |

|  |
| --- |
| </div> |
|  |

|  |
| --- |
| </div> |
|  |

|  |
| --- |
| <div role="tabpanel" class="tab-pane fade in" id="htab6"> |
|  |

|  |
| --- |
| <h1>Attendance View</h1> |
|  |

|  |
| --- |
|  |
|  |

|  |
| --- |
| <div class="col-sm-3"> |
|  |

|  |
| --- |
| <table class="table"> |
|  |

|  |
| --- |
| <tbody id="classlist6body"> |
|  |

|  |
| --- |
|  |
|  |

|  |
| --- |
| </tbody> |
|  |

|  |
| --- |
| </table> |
|  |

|  |
| --- |
| </div> |
|  |

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

|  |
| --- |
| <table class="table table-bordered"> |
|  |

|  |
| --- |
| <thead> |
|  |

|  |
| --- |
| <tr> |
|  |

|  |
| --- |
| <th>No</th> |
|  |

|  |
| --- |
| <th>Roll No</th> |
|  |

|  |
| --- |
| <th>Reg No</th> |
|  |

|  |
| --- |
| <th>Name</th> |
|  |

|  |
| --- |
| <th>Total</th> |
|  |

|  |
| --- |
| <th>Present</th> |
|  |

|  |
| --- |
| <th>Absent</th> |
|  |

|  |
| --- |
| <th>Average</th> |
|  |

|  |
| --- |
| </tr> |
|  |

|  |
| --- |
| </thead> |
|  |

|  |
| --- |
| <tbody id="attendance\_body"> |
|  |

|  |
| --- |
|  |
|  |

|  |
| --- |
| </tbody> |
|  |

|  |
| --- |
| </table> |
|  |

|  |
| --- |
|  |
|  |

|  |
| --- |
|  |
|  |

|  |
| --- |
| </div> |
|  |

|  |
| --- |
| <div class="col-sm-3"></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> |
|  |