**MULTIPAGE WEBSIDE FOR SCHOOL MANAGEMENT SYSTEM**

**A PROJECT REPORT SUBMITTED TO**

**SRM INSTITUTE OF SCIENCE & TECHNOLOGY**

**IN PARTIAL FULFILMENT OF THE REQUIREMENTS FOR THE AWARD OF THE DEGREE OF**

**MASTER OF COMPUTER APPLICATIONS**

**BY**

**[SACHIN.C]**

**Reg. No.RA1832241010003**

**UNDER THE GUIDANCE OF**

**Dr. /Ms. S. Chandrakala**

****

**DEPARTMENT OF COMPUTER APPLICATIONS**

**COLLEGE OF SCIENCE AND HUMANITIES**

**SRM INSTITUTE OF SCIENCE & TECHNOLOGY**

Kattankulathur – 603 203

Chennai, Tamilnadu

**DEC – 2020 TO APR-2021**

**BONAFIDE CERTIFICATE**

This is to certify that the project report titled ***“SACHIN.C”*** is a bonafide work carried out by RA1832241010003 (REG.NO.) under my supervision for the award of the Degree of Master of Computer Applications. To my knowledge the work reported herein is the original work done by these students.

**Ms. S. Chandrakala**

Assistant Professor,

Department of Computer Applications

(GUIDE)

**Dr. S. Albert Antony Raj**

Associate Professor and Head,

Department of Computer Applications

Internal Examiner

External Examiner

**ACKNOWLEDGEMENT**

With the profound gratitude to the **ALMIGHTY** I take this chance to thank people who helped me to complete this project.

We take this as a right opportunity to say THANKS to my **Parents** who are there to stand with me always with the words “YOU CAN”.

We are thankful to **Dr.T.R.Paarivendhar**, Chancellor, SRM Institute of Science & Technology who gave me the platform to establish myself to reach greater heights.

We earnestly thank **Dr.J.Jothi Kumar**, Dean, [**Dr. Sridhar Krishnaswami**](http://www.srmuniv.ac.in/content/dr-sridhar-krishnaswami), Deputy Dean and **Dr.A.Duraisamy**, Associate Dean, College of Science and Humanities, SRM Institute of Science & Technology who always encourage us to do novel things.

We express our sincere thanks to **Dr. S. Albert Antony Raj** **M.Sc., M.Phil., Ph.D.,** Associate professor andHead, Department of Computer Applications, for his valuable guidance and support to execute all incline in learning.

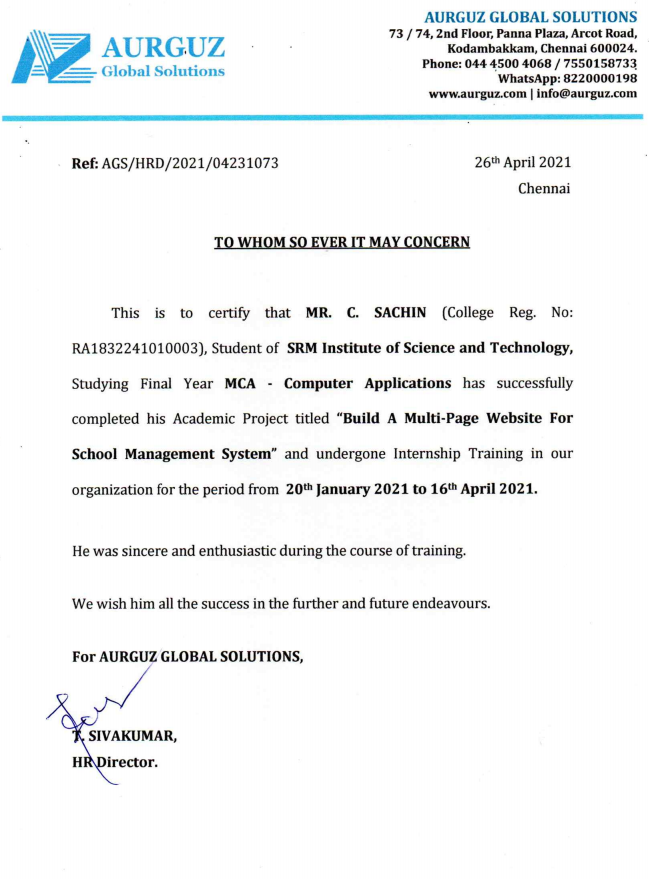
It is my delight to thank our project guide **Your Guide Name,** Assistant Professor, Department of Computer Applications for **his/her** help, support, encouragement, suggestions and guidance throughout development phases of the project.

We convey our gratitude to all the faculty members of the department who extended their support through valuable comments and suggestions during the reviews.

A great note of gratitude to friends and people who are known and unknown to me who helped in carrying out this project work a successful one.

**Your name**

Internship certificate

****

**Plagiarism report**

****

**Table of Contents**

|  |  |  |
| --- | --- | --- |
| **Chapter Number** | **Chapter Name** | **Page.no** |
| **1** | **INTRODUCTION** | 7-10 |
| **2** | **DEVELOPMENT PLATFORM INTRODUCTION** | 11-14 |
| **3** | **SYSTEM ANALYSIS** | 15-22 |
| **4** | **SYSTEM DESIGN CODING STRUCTURE** | 23-35 |
| **5** | **SYSTEM IMPLEMENTATION** | 36-42 |
| **6** | **SYSTEM TEST** | 43-56 |
| **7** | **USER MANUAL** | 27-28 |
| **8** | **CONCLUSION AND REFERENCE** | 64-67 |

**Abstract**

* Many of schools and university across the world have transmitted into fully online experiences with the recent pandemic with each school’s backend stack witnessing new highs in usage in educational system.
* As we all know the days we are facing today with virus so in this situation we can’t leave our education system for that we are working here to maintain knowledge with power.
* Structure database solutions are more important than ever. Here’s a walkthrough on how to create a **‘**Multi-page Website School Management System’ using **“**HTML, CSS, JS,BOOTSTRAP” for identify library system management, with “MySQL DB” as a Server side database.
* MySQL DB allows us to create SQL databases with provides an altogether different mechanism for storage and retrieval. You can perform as many reads/write as you desire.
* Here we can manage our student database for later retrieve and manipulation
* Many of schools and university across the world have transmitted into fully online experiences with the recent pandemic with each school’s backend stack witnessing new highs in usage in educational system.
* As we all know the days we are facing today with virus so in this situation we can’t leave our education system for that we are working here to maintain knowledge with power.
* Here we can manage our student database for later retrieve and manipulation use.

**List of Figure**

|  |  |  |
| --- | --- | --- |
| **Figure Number** | **Figure Name** | **Page Number** |
| 1 | Flow chart | 13 |
| 2 | DFD – data flow | 14 |
| 3 | ER - diagram | 15 |

|  |  |
| --- | --- |
|  |  |

**List of Screenshots**

|  |  |  |
| --- | --- | --- |
| **No.** | **Screenshots Names** | **Page no** |
| 1 | Admin login | 48 |
| 2 | Student login | 49 |
| 3 | Staff login | 52 |

**Chapter 1**

# Introduction

# Introduction

Now a day’s education plays a great role in development of any country. Many of education organizations try to increase education quality. One of the aspects of this improvement is managing of school resources.

Education Management System carried on by any individual or institution engaged in providing a services to students, teachers, guardians and other persons are intermediary that performs one or more of the following functionalities – Student Admission, Employee Registration, Student List, Employee List, Student Attendance, Employee Attendance, Student Routine, Result Management, Payroll & Accounts.

Education Management System (EMS) is such a service which provides all services for an educational institute to make your life easier and faster by assuring its performance. Easy User Management System, Easy Admission Process, Easy Attendance System.

EMS is a system that will provide you a bird’s eye view of the functioning of the entire educational institution. It is a management information system helps to manage the different processes in an educational institution like General Administration, Staff Management, Academics, Student Management, and Accounts etc. The information is made using the latest technologies and help’s to make decision making a lot faster, effective and easier than ever before. Also helps to improve the overall quality of education of the institution.

We use database and database technology are having a major impact on the growing use of computers. The implementation of the system was done using c# and SQL Server 2012 technologies, allowing system to be run in Windows OS.

In a nutshell, Education Management Software managed your education institution by simplifying and automating processes and addressing the needs of all stakeholders helping them to be more efficient in their respective roles.

## Outline of the Thesis

The overview of related and used technologies in the implementation is given in Chapter 2

The architecture and way of communication between client and service is explained in Chapter 3.

The detailed information about implementation of the system is presented in Chapter 4.

Chapter 5 provides the summary of the implemented system. The Appendices provides some additional information concerning the system.

## Scope of this application

The difference area where we can use this application as:

* Any education institution makes use of it providing class schedule.
* It can be used in offices and modifications can be easily done according to requirements.

## Focusing of the Project

The system is completed under the guidance of the theory and methods of management information systems, database technology support. This paper first discusses the structure of the background, purpose and significance of the graduate design topics. Then describes the development platform and database technology and the advantages of each, followed by more devoted a system requirements analysis, design, implementation, and the implementation of the tasks, techniques and tools. End system to complete the information input, output, data modification, query and statistics, as well as print statements, make operation simple and quick.

In this project, we try to build up sound software which can operate any challenging situation in the modern time. Administrator and users information are making effective decisions. The decisions are more accurate, relevant and timely the information stored or process is more effective.

## Features of the Project

The common features of the projects are:

* This is very easy to use for each user.
* Increase Efficiencies and Reduce Costs
* Transform IT for Higher Education
* Easy Solution
* Easy Admission
* Secure All Data
* Easy Account Maintenance
* Transaction History
* Easy Attendance Process
* The user of the database can see all information and also can edit if necessary.
* Easy implemented routine for student and teacher’s.

## Module of Easy Education Management System

## 

### 1.5.1 Administration

* System User Group Setup
* System User Setup
* System User Authorization Setup
* Class Setup
* Section Setup
* Student Group Setup
* Subject Setup
* Stuff Designation Setup
* Student Routine Setup
* Student Hall Setup
* Student Seat Setup

### 1.5.2 Personal Information Management System (PIMS)

* 1. Student Registration/ Admission
     + Personal Information
     + Guardian Information
  2. Employee/ Stuff Registration
     + Personal Information
     + Academic Information
     + Professional Information
     + Skills/ Training Information

### Attendance

### Student Attendance

### Employee / Stuff Attendance

### Result

1. Student Exam Setup
2. Student Exam Result Entry

### 1.5.5 Accounts

1. Income
2. Expense
3. Bank deposit
4. Withdraw from Bank

### 1.5.6 Reports

1. Student Daily Attendance
2. Student’s Result
3. Employee/ Staff Attendance
4. Student Details
5. Student List Report ( For Each Class)
6. Class Wise Student Routine
7. Employee details Report
8. Employee/ Stuff List
9. All Income By Date
10. Head Wise Income By Month
11. Head Wise Monthly Income by Financial Year
12. All Expense By Date
13. Head Wise Expense By Month
14. Head Wise Monthly Expense By Year

### 1.5.7 Students Information to Parents by Easy SMS Service

1. Admission Information Confirmation
2. Student Attendance Report Each Day ( If Not Come)
3. Student Attendance Report Each Exam ( If Not Come)
4. Exam Date Information
5. Exam Date Cancel Information

# Chapter 2

Platform Introduction

# Development platform Introduction

WHY CHOOSE “PHP":

Internet has joined the people living around the globe. It has, no doubt, gotten quite tough to sustain your identity in the cyber world as the competition has gone beyond the limits. To make it easier for you to compete and excel in the world of internet, PHP is among the best tools that can be used. PHP is abbreviation of “personal homepage” and sometimes is also known as Hypertext Preprocessor. The latter name is particularly used in the cyber circle. It is, in general, a HTML embedded scripting language being used widely for the web application development. The use of the language has increased in recent times due to the ease it offers to the developer. There are various benefits of using the language over the others developed for the same purpose. Some of the major pros pertinent to the language are discussed as under:

Double end web development

Some of the languages used for web development have limitation of purposes. PHP is one of its kinds because it may be used on both front-end and back-end web development. Due to this feature, a programmer may easily alter the present conditions of the website merely by changing a single code. Unlike PHP, other languages need to be encoded to understand the correlation among the back-end and front-end languages making programming time taking and laborious.

Why to pay when it’s free?

Another reason why a programmer must prefer PHP over other languages is its legal free of cost availability. Some of the organizations having its similar programming languages in the market charge programmers against the language they offer. However, PHP may be downloaded and installed using any open source language house easily accessible from one’s computer. Therefore, the basic goal of earning more can be changed into reality by just taking a right step of using PHP as web application development.

Simplicity and user friendliness

It goes without saying that everyone wants ease out of the programming language. This is what PHP offers to the users. Unlike C++ and other similar languages, PHP is quite easy to be understood by the users. There is no need of any formal training prior to use the language for the required purpose. PHP programmed web applications are easy to scale and highly secure as compare to applications built in other language.

Compatible to all the operating software

Versatility in the available operating software, no doubt, has provided variety to the users but on the other hand has also caused some serious complexities for the programmers. However, you may make it easier for yourself by switching over to PHP as it is compatible to all the famous operating systems. Apple’s famous MAC and Microsoft’s Windows is among the top operating systems that are supported by PHP. Linux is also not out of the line of the compatible systems.

ABOUT PHP:

PHP started out as a small open source project that evolved as more and more people found out how useful it was. Rasmus Lerdorf unleashed the first version of PHP way back in 1994.

• PHP is a recursive acronym for "PHP: Hypertext Preprocessor".

• PHP is a server side scripting language that is embedded in HTML. It is used to manage dynamic content, databases, session tracking, even build entire e- commerce sites.

• It is integrated with a number of popular databases, including MySQL, PostgreSQL, Oracle, Sybase, Informix, and Microsoft SQL Server.

• PHP is pleasingly zippy in its execution, especially when compiled as an Apache module on the Unix side. The MySQL server, once started, executes even very complex queries with huge result sets in record-setting time.

## MySQL Features

Why MYSQL?

MySQL is a fast, easy-to-use RDBMS being used for many small and big businesses. MySQL is developed, marketed, and supported by MySQL AB, which is a Swedish company. MySQL is becoming so popular because of many good reasons:

• MySQL is released under an open-source license. So you have nothing to pay to use it.

• MySQL is a very powerful program in its own right. It handles a large subset of the functionality of the most expensive and powerful database packages.

• MySQL uses a standard form of the well-known SQL data language.

• MySQL works on many operating systems and with many languages including PHP, PERL, C, C++, JAVA, etc.

• MySQL works very quickly and works well even with large data sets.

• MySQL is very friendly to PHP, the most appreciated language for web development.

• MySQL supports large databases, up to 50 million rows or more in a table. The default file size limit for a table is 4GB, but you can increase this (if your operating system can handle it) to a theoretical limit of 8 million terabytes (TB).

• MySQL is customizable. The open-source GPL license allows programmers to modify the MySQL software to fit their own specific environments.

Structured Query Language (SQL)

• SQL (pronounced SEQUEL) is the programming language that defines and manipulates the database. SQL databases are relational

## Database Platform

A database is an organized collection of [data](http://en.wikipedia.org/wiki/Data_(computing)). The data is typically organized to model aspects of reality in a way that supports [processes](http://en.wikipedia.org/wiki/Process_(computing)) requiring information, such as modeling the availability of rooms in hotels in a way that supports finding a hotel with vacancies.

Database management systems (DBMS) are [computer software](http://en.wikipedia.org/wiki/Computer_software) applications that interact with the user, other applications, and the database itself to capture and analyze data. A general-purpose DBMS is designed to allow the definition, creation, querying, update, and administration of databases. Well-known DBMSs include [MySQL](http://en.wikipedia.org/wiki/MySQL), [PostgreSQL](http://en.wikipedia.org/wiki/PostgreSQL" \o "PostgreSQL), [Microsoft SQL Server](http://en.wikipedia.org/wiki/Microsoft_SQL_Server), [Oracle](http://en.wikipedia.org/wiki/Oracle_Database), [Sybase](http://en.wikipedia.org/wiki/Sybase) and [IBM DB2](http://en.wikipedia.org/wiki/IBM_DB2).

## SQL Server database Introduction

In [computing](http://en.wikipedia.org/wiki/Computing), Microsoft SQL Server is a [relational database management system](http://en.wikipedia.org/wiki/Relational_database_management_system), currently developed by Microsoft. As a database server, it is a software product with the primary function of storing and retrieving data as requested by other [software applications](http://en.wikipedia.org/wiki/Software_applications) which may run either on the same computer or on another computer across a network (including the Internet).

Microsoft markets at least a dozen different editions of Microsoft SQL Server - aimed at different audiences and for workloads ranging from small single-machine applications to large Internet-facing applications with many [concurrent users](http://en.wikipedia.org/wiki/Concurrent_user).

## SQL Server database system features

1. Supports most administrative tasks for SQL Server.
2. A single, integrated environment for SQL Server Database Engine management and authoring.
3. Dialogs for managing objects in the SQL Server Database Engine, Analysis Services, and Reporting Services, that allows you to execute your actions immediately, send them to a Code Editor, or script them for later execution.
4. Non-modal and resizable dialogs allow access to multiple tools while a dialog is open.
5. A common scheduling dialog that allows you to perform action of the management dialogs at a later time.
6. Exporting and importing SQL Server Management Studio server registration from one Management Studio environment to another.
7. Save or print XML Show plan or Deadlock files generated by SQL Server Profiler, review them later, or send them to administrators for analysis.
8. A new error and informational message box that presents much more information, allows you to send Microsoft a comment about the messages, allows you to copy messages to the clipboard, and allows you to easily e-mail the messages to your support team.
9. An integrated Web browser for quick browsing of MSDN or online help.

# Chapter 3

# System Analysis

# Introduction

Systems analysis is a problem solving technique that decomposes a system into its component pieces for the purpose of the studying how well those component parts work and interact to accomplish their purpose. As the software system requirements were predictable, it is decided to follow the classical system development life cycle method. This process demands a systematic, sequential approach to software development that begins at the system level and progress through analysis, design, coding, testing and maintenance. The steps that is applicable to all software engineering paradigms. The program is followed by SDLC (Software Development Life Cycle)

## System Engineering and Analysis

Software is always a part of a large system; work begins by establishing requirement for all system elements and then allocating some subsets of this requirement to software. This system view is essential when software must interface with other elements such as hardware, people and database. System engineering and analysis encompasses requirements gathering at the system level with a small amount of top-level design analysis.

## System Analysis

Analysis involves the requirement determination and specification. Systems analysis is a problem solving technique that decomposes a system into its component pieces for the purpose of the studying how well those component parts work and interact to accomplish their purpose. According to the Merriam-Webster dictionary, systems analysis is the process of studying a procedure or business in order to identify its goals and purposes and create systems and procedures that will achieve them in an efficient way. Analysis and synthesis, as scientific methods, always go hand in hand, they complement one another.

## Requirement Analysis

Requirements analysis in [systems engineering](http://en.wikipedia.org/wiki/Systems_engineering) and [software engineering](http://en.wikipedia.org/wiki/Software_engineering), encompasses those tasks that go into determining the needs or conditions to meet for a new or altered product or project, taking account of the possibly conflicting [requirements](http://en.wikipedia.org/wiki/Requirement) of the various [stakeholders](http://en.wikipedia.org/wiki/Stakeholder_(corporate)), analyzing, documenting, validating and managing software or system requirements.

Requirements analysis is critical to the success of a systems or software project. The requirements should be documented, actionable, measurable, testable, traceable, related to identified business needs or opportunities, and defined to a level of detail sufficient for system design.

## Software Requirements

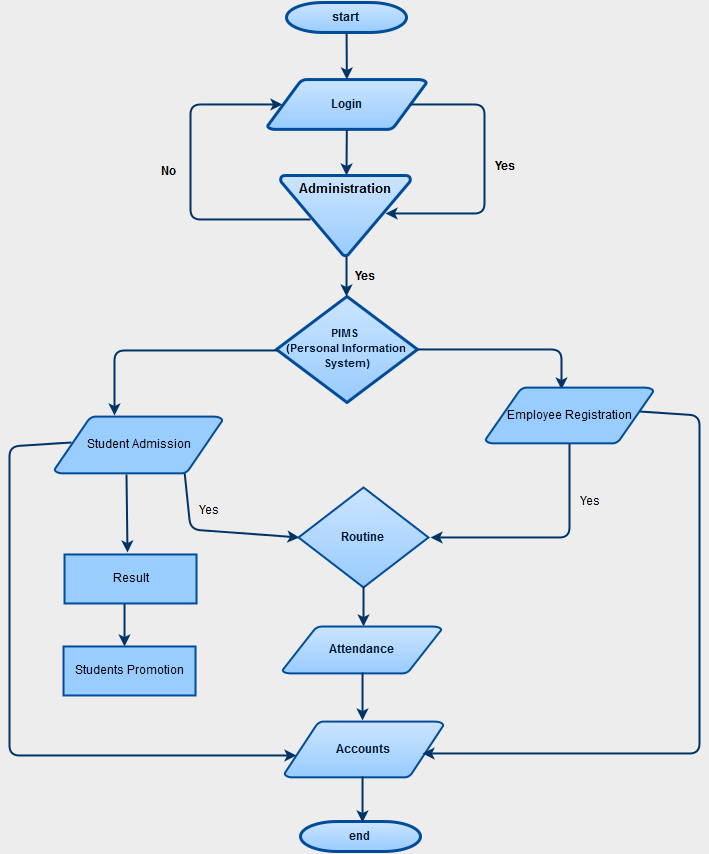
1.31 Windows 10 / XAMP

1. VS Code
2. MS/SQL Server

## Language used

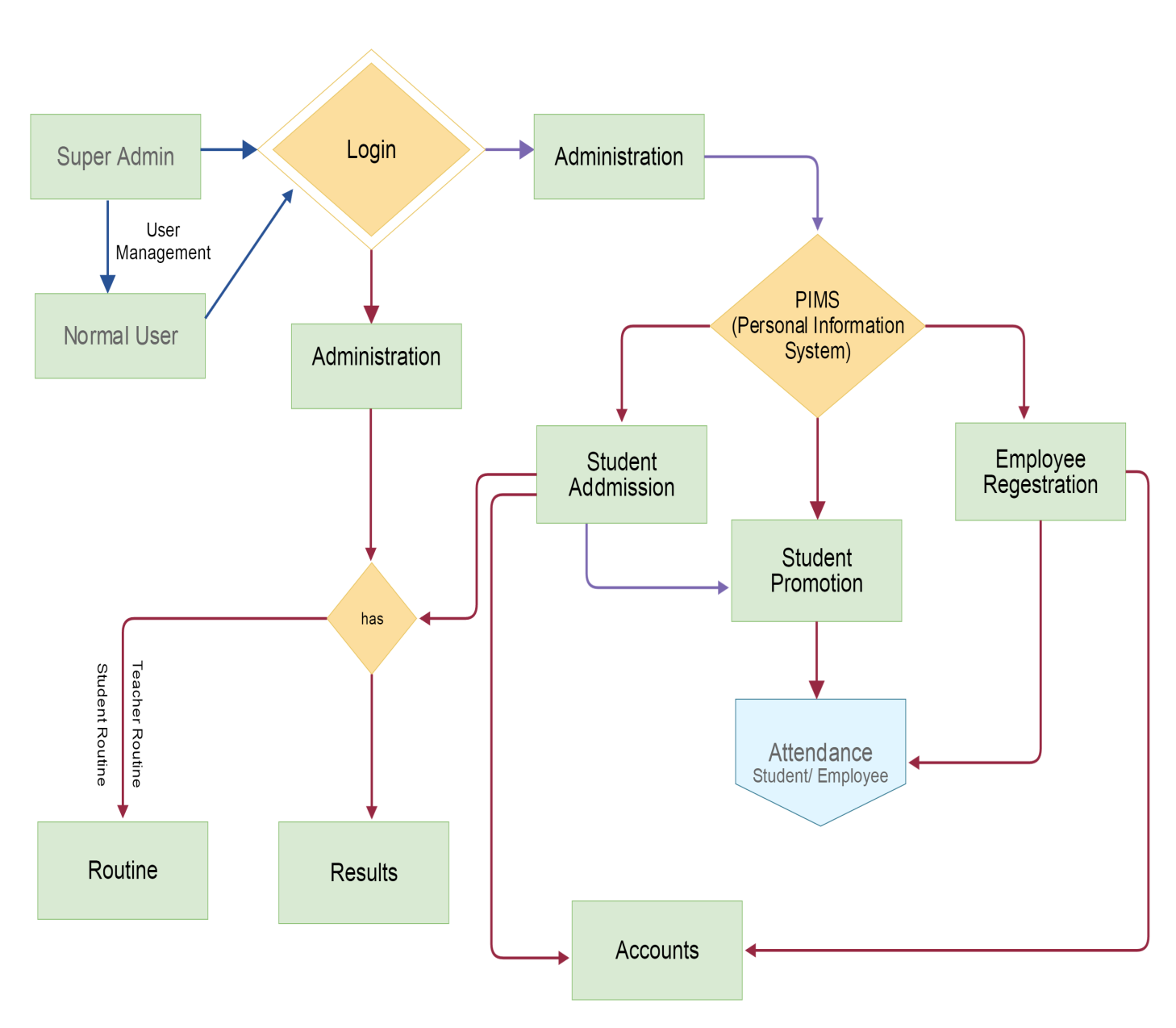
1. HTML /CSS /JS /Bootstrap – Front end
2. PHP
3. MS/ MYSQL – Back end

## Flow Chart



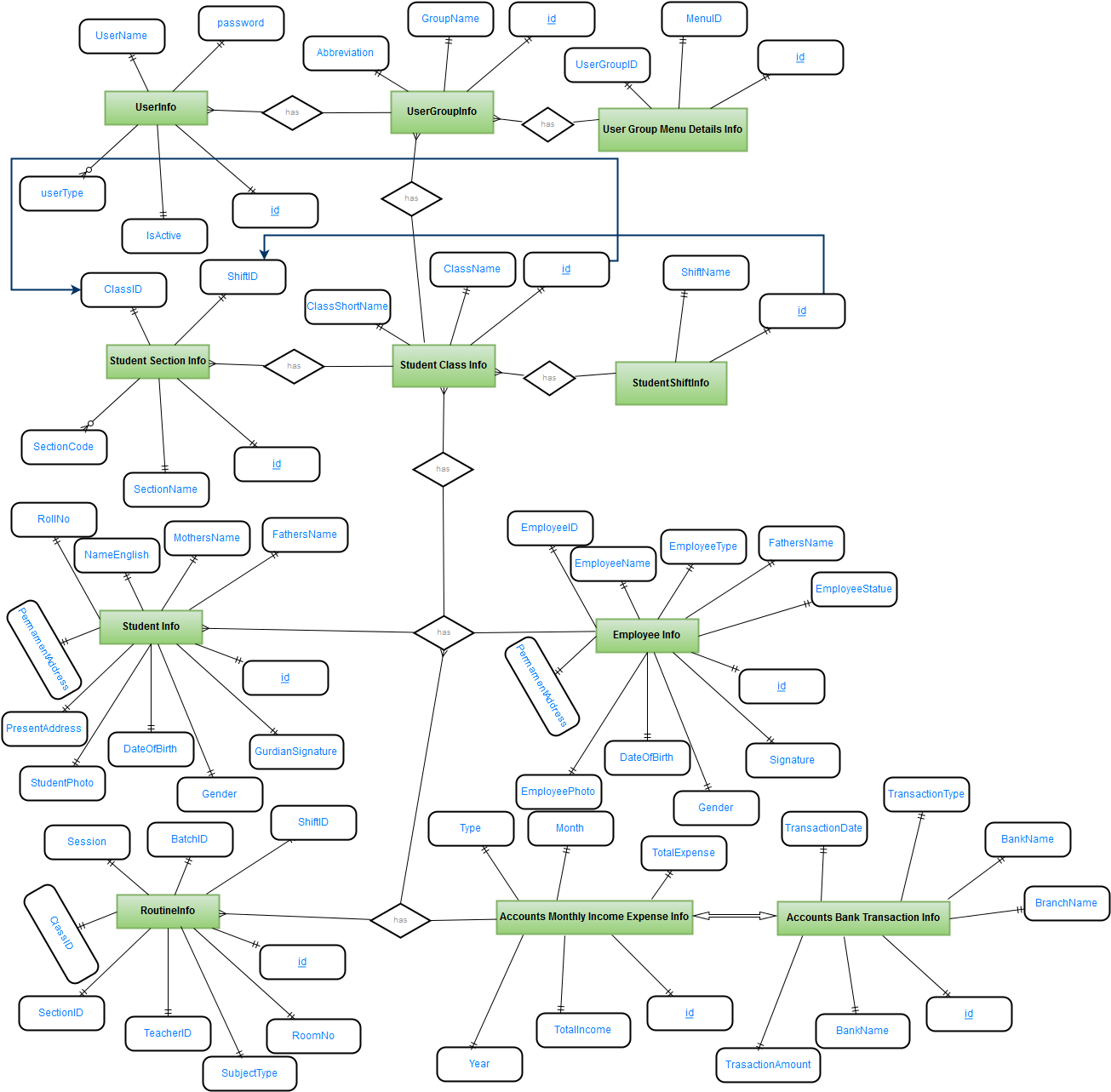
**Figure 0:** DFD Diagram

## Data Flow Diagram (DFD)



**Figure 1:** DFD Diagram

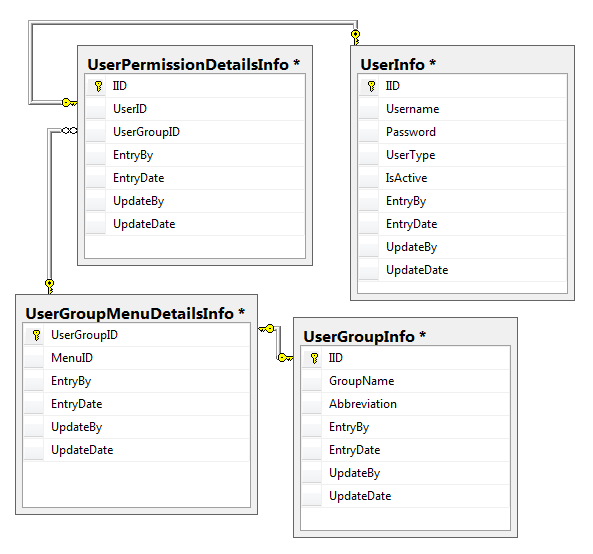
## ER Diagram



**Figure 2:** ER Diagram

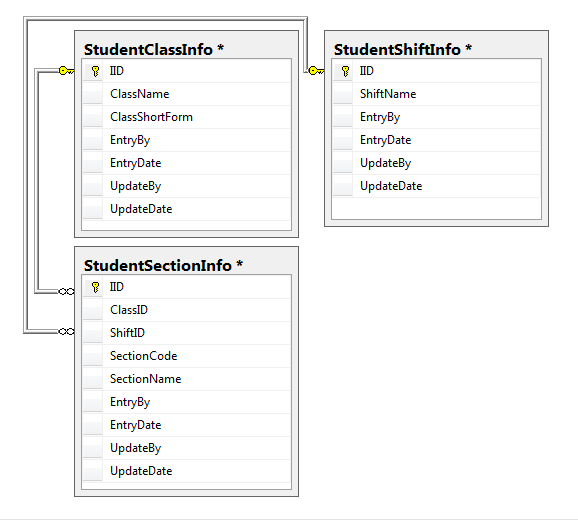
## Relational Database

### User



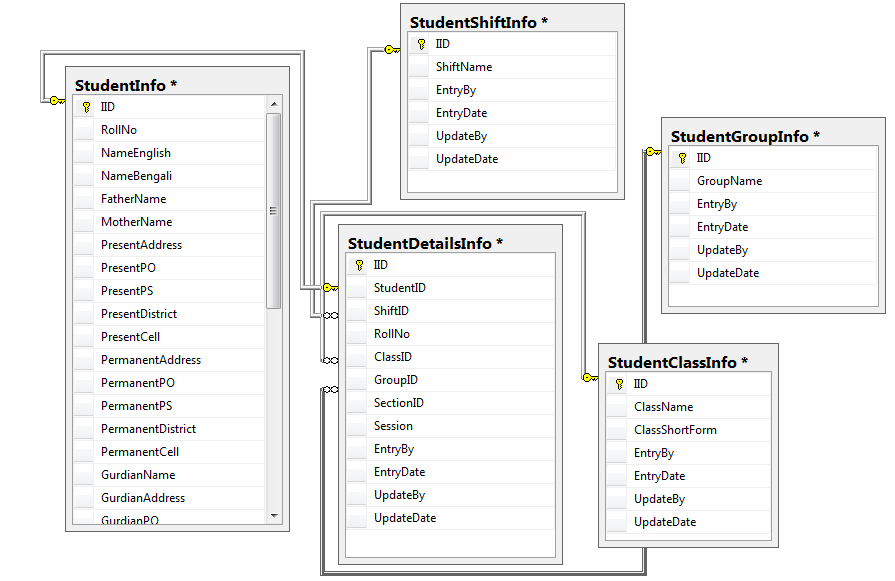
**Figure 3:** User Relational Database

### Administration



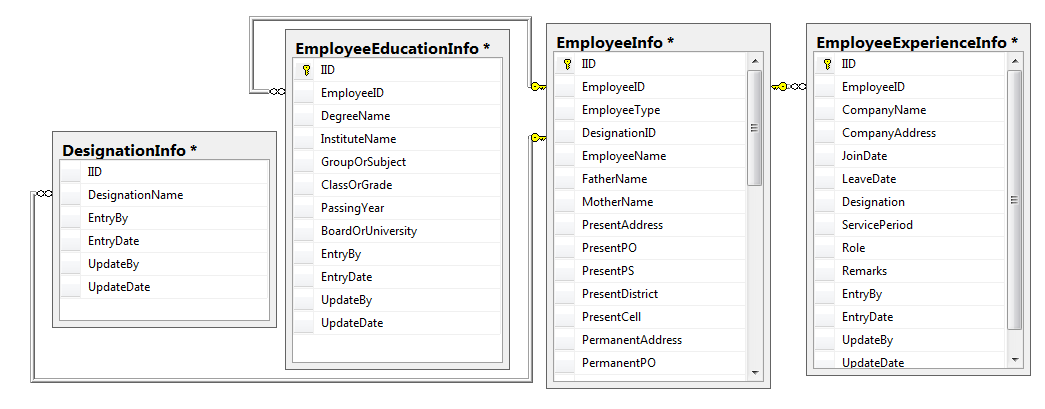
**Figure 4:** Administration Relational Database

### Student



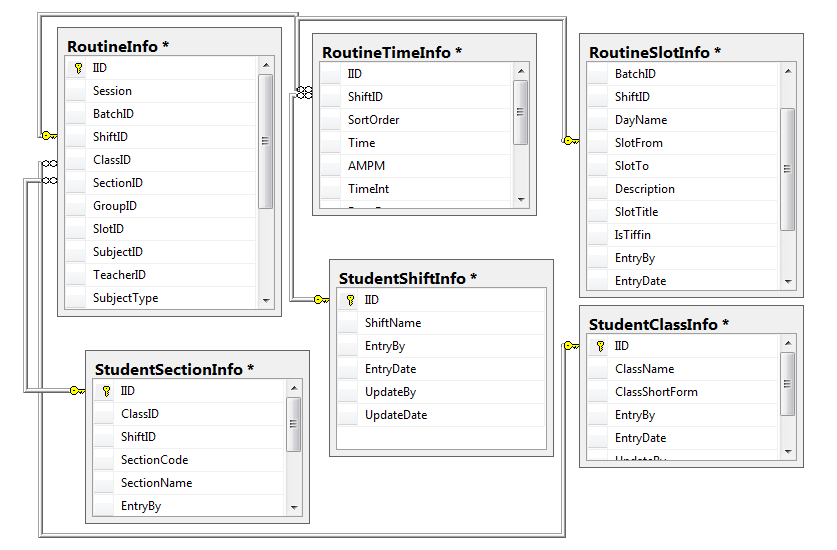
**Figure 5:** Student Relational Database

### Employee



**Figure 6:** Employee Relational Database

### Routine



**Figure 7:** Routine Relational Database

# Chapter 4

# System Design & Coding Structure

# Introduction

The [design phase](http://searchsoftwarequality.techtarget.com/definition/systems-development-life-cycle) is concerned with the physical construction of the system. Included are the design or configuration of the network (hardware, operating system, programming, etc.), design of user interfaces (forms, reports, etc.), design of system interfaces (for communication with other systems), and security issues. It is important that the proposed design be tested for performance, and to ensure that it meets the requirements outlined during the analysis phase. In other words, the main objective of this phase is to transform the previously defined requirements into a complete and detailed set of specifications which will be used during the next phase. Some of the activities that need to take place during the design phase are:

1. Design the application
2. Design and integrate the network
3. Design and integrate the database
4. Create a contingency plan
5. Start a Maintenance, Training and Operations plan
6. Review the design
7. Articulate the business processes and procedures
8. Establish a transition strategy
9. Deliver the System Design Document
10. Review final design

A database system is essentially nothing more than a computerized record keeping system the database itself can be regarded as kind of electronic filing cabinet. A database consists of same collection of some collection of persistent data that is used by the applications system of given some instituted. The term “instituted” here is simply a convenient generic term for any reasonable self- contained science, technical or other institution.

## Database Design

A database management system (DBMS) is a collection of programs that enables you to [store](http://www.webopedia.com/TERM/S/store.html), modify, and extract information from a [database](http://www.webopedia.com/TERM/D/database.html). There are many different types of database management systems, ranging from small [systems](http://www.webopedia.com/TERM/S/system.html) that [run](http://www.webopedia.com/TERM/R/run.html) on personal computers to huge systems that run on [mainframes](http://www.webopedia.com/TERM/M/mainframe.html).

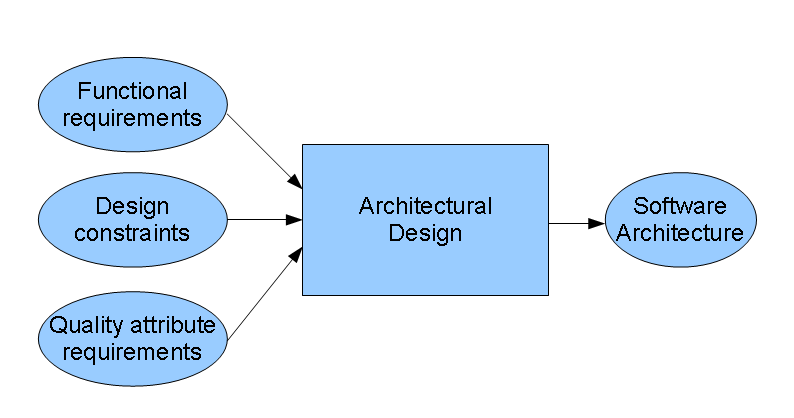
DBMS is a software that handles the storage, retrieval, and updating of data in a computer system.

Ex- SQL Server (Microsoft), MySQL (Freeware), Oracle (Oracle), NoSQL (Oracle), NonStop SQL (Hewlett Packard)

## System Architecture Design

A system architecture or systems architecture is the [conceptual model](http://en.wikipedia.org/wiki/Conceptual_model) that defines the [structure](http://en.wikipedia.org/wiki/Structure), [behavior](http://en.wikipedia.org/wiki/Behavior), and more [views](http://en.wikipedia.org/wiki/View_model) of a [system](http://en.wikipedia.org/wiki/System). An architecture description is a formal description and representation of a system, organized in a way that supports reasoning about the [structures](http://en.wikipedia.org/wiki/Structure) and [behaviors](http://en.wikipedia.org/wiki/Behavior) of the system.

System architecture can comprise system [components](http://en.wikipedia.org/wiki/System), the externally visible properties of those components, the relationships (e.g. the behavior) between them. It can provide a plan from which products can be procured, and systems developed, that will work together to implement the overall system. There have been efforts to formalize languages to describe system architecture, collectively these are called [architecture description languages](http://en.wikipedia.org/wiki/Architecture_description_languages).



**Figure 8:** Software Architecture Design

Software architecture refers to the high level structures of a [software system](http://en.wikipedia.org/wiki/Software_system), the discipline of creating such structures, and the documentation of these structures. It is the set of structures needed to reason about the software system. Each structure comprises software elements, relations among them, and properties of both elements and relations. The architecture of a software system is a metaphor, analogous to the [architecture](http://en.wikipedia.org/wiki/Architecture) of a building.

## Code Design

Design patterns are solutions to software design problems you find again and again in real-world application development. Patterns are about reusable designs and interactions of objects.

HTML stands for **H**yper **T**ext **M**arkup **L**anguage

A markup language is a set of **markup tags**

HTML documents are described by **HTML tags**

Each HTML tag **describes** different document content

This real-world code demonstrates the Singleton pattern as a Load Balancing objects. Only a single instance (the singleton) of the class can be created because servers may dynamically come on-or off-line and every request must go through the one object that has knowledge about the state of the (web) farm.

## Sample Code

* + 1. **Form Login**

Design:

Code:

|  |  |
| --- | --- |
|  | |
|  | <html> | |
|  |  | |
|  | <head> | |
|  | <meta charset="utf-8"> | |
|  | <meta name="viewport" content="width=device-width, initial-scale=1.0"> | |
|  | <title>school management system</title> | |
|  | <link rel="shortcut icon" href="[assets/img/title.gif](http://localhost/school%20management%20system/assets/img/title.gif)" type="image/x-icon"> | |
|  | <link rel="stylesheet" href="[assets/bootstrap/css/bootstrap.min.css](http://localhost/school%20management%20system/assets/bootstrap/css/bootstrap.min.css)"> | |
|  | <link rel="stylesheet" href="<http://fonts.googleapis.com/css?family=Roboto>"> | |
|  | <link rel="stylesheet" href="[assets/css/styles.css](http://localhost/school%20management%20system/assets/css/styles.css)"> | |
|  | <link rel="stylesheet" href="[assets/css/Google-Style-Login.css](http://localhost/school%20management%20system/assets/css/Google-Style-Login.css)"> | |
|  | <link rel="stylesheet" href="[assets/css/bootstrap.css](http://localhost/school%20management%20system/assets/css/bootstrap.css)"> | |
|  | <link rel="stylesheet" href="[assets/css/bootstrap.min.css](http://localhost/school%20management%20system/assets/css/bootstrap.min.css)"> | |
|  | <link rel="stylesheet" href="[assets/css/bootstrap-theme.css](http://localhost/school%20management%20system/assets/css/bootstrap-theme.css)"> | |
|  | <link rel="stylesheet" href="[assets/css/bootstrap-theme.min.css](http://localhost/school%20management%20system/assets/css/bootstrap-theme.min.css)"> | |
|  | <link rel="stylesheet" href="[assets/css/Hero-Technology.css](http://localhost/school%20management%20system/assets/css/Hero-Technology.css)"> | |
|  | <link rel="stylesheet" href="[assets/css/Mockup-MacBook-Pro.css](http://localhost/school%20management%20system/assets/css/Mockup-MacBook-Pro.css)"> | |
|  | <link rel="stylesheet" href="[assets/bootstrap/css/bootstrap.min.css](http://localhost/school%20management%20system/assets/bootstrap/css/bootstrap.min.css)"> | |
|  | <link rel="stylesheet" href="[assets/css/styles.css](http://localhost/school%20management%20system/assets/css/styles.css)"> | |
|  | <link rel="stylesheet" href="[assets/css/Google-Style-Login.css](http://localhost/school%20management%20system/assets/css/Google-Style-Login.css)"> | |
|  | </head> | |
|  |  | |
|  | <body background="assets/img/5.jpg" > | |
|  | <div style=" | |
|  | font-family:Nyala, Arial; | |
|  | text-align: left; | |
|  | background-color: #526F35; | |
|  | padding: 20px; | |
|  | color:white; | |
|  | width: 100%; | |
|  | height: 150px;"> | |
|  | <!--This codes to load the image loader--> | |
|  |  | |
|  | <!--this is the heading section--> | |
|  | <h2> | |
|  |  | |
|  | <div style="float:right; font-size:20px;text-align:right;"> | |
|  |  | |
|  | <img src="[assets/img/mail2.png](http://localhost/school%20management%20system/assets/img/mail2.png)">Email: info@info.com<br> | |
|  | <img src="[assets/img/call1.png](http://localhost/school%20management%20system/assets/img/call1.png)">Contact:+123456789<br> | |
|  | <img src="[assets/img/location.png](http://localhost/school%20management%20system/assets/img/location.png)">Location: Sample Address | |
|  | </div> | |
|  | </h2> | |
|  | </div> | |
|  |  | |
|  | <div class="login-card"> | |
|  | <p class="profile-name-card"> </p> | |
|  | <form class="form-signin" action="login.php" method="POST" enctype="multipart/form-data"><span class="reauth-email"> </span> | |
|  | <input class="form-control" type="text" required="" placeholder="username" name="username" autofocus="" id="username"> | |
|  | <input class="form-control" type="password" required="" placeholder="Password" name="password" id="password"> | |
|  | <div class="checkbox"> | |
|  | <div class="checkbox"> | |
|  | <label> | |
|  | <input type="checkbox">Remember me</label> | |
|  | </div> | |
|  | </div> | |
|  | <button class="btn btn-primary btn-block btn-lg btn-signin" type="submit" name="login">Login</button> | |
|  | </form><a href="[email.php](http://localhost/school%20management%20system/email.php" \t "_blank)" class="forgot-password">Forgot your password?</a></div> | |
|  | <script src="[assets/js/jquery.min.js](http://localhost/school%20management%20system/assets/js/jquery.min.js)"></script> | |
|  | <script src="[assets/bootstrap/js/bootstrap.min.js](http://localhost/school%20management%20system/assets/bootstrap/js/bootstrap.min.js)"></script> | |
|  | </body> | |
|  |  | |
|  | </html> | |

* + 1. **User Group Permission**

Design:

Code:

|  |  |
| --- | --- |
|  | |
|  | |
|  |  | |
|  | <!DOCTYPE html> | |
|  | <html> | |
|  |  | |
|  | <head> | |
|  | <meta charset="utf-8"> | |
|  | <meta name="viewport" content="width=device-width, initial-scale=1.0"> | |
|  | <title>school management system</title> | |
|  | <link rel="shortcut icon" href="[assets/img/title.gif](http://localhost/school%20management%20system/assets/img/title.gif)" type="image/x-icon"> | |
|  | <link rel="stylesheet" href="[assets/css/bootstrap.min.css](http://localhost/school%20management%20system/assets/css/bootstrap.min.css)"> | |
|  | <link href="[assets/css/loader.css](http://localhost/school%20management%20system/assets/css/loader.css)" rel="stylesheet" /> | |
|  | <script src="[assets/js/canvasjs.min.js](http://localhost/school%20management%20system/assets/js/canvasjs.min.js)"></script> | |
|  | <!--\*\*\*\*\*jquery -3.2.1.js file supports the use of dropdown\*\*\*--> | |
|  | <script src="[assets/js/jquery-3.2.1.js](http://localhost/school%20management%20system/assets/js/jquery-3.2.1.js)"></script> | |
|  |  | |
|  | <script type="text/javascript"> | |
|  | var class1=2; | |
|  | var class2=0; | |
|  | var class3=0; | |
|  | var class4=0; | |
|  | var class5=0; | |
|  | var class6=0; | |
|  | var class7=0; | |
|  | var class8=0; | |
|  | window.onload = function () { | |
|  | var chart = new CanvasJS.Chart("chartContainer", { | |
|  | title:{ | |
|  | text: "Number of students" | |
|  |  | |
|  | }, | |
|  | data: [ | |
|  | { | |
|  | // Change type to "doughnut", "line", "splineArea", etc. | |
|  | type: "spline", | |
|  |  | |
|  | dataPoints: [ | |
|  | { label: "Form 1", y: class1 }, | |
|  | { label: "Form 2", y: class2 }, | |
|  | { label: "Form 3", y: class3 }, | |
|  | { label: "Form 4", y: class4 }, | |
|  | ] | |
|  | } | |
|  | ] | |
|  | }); | |
|  | chart.render(); | |
|  | } | |
|  | </script> | |
|  |  | |

|  |  |
| --- | --- |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |

* + 1. **Section Entry**

Design:

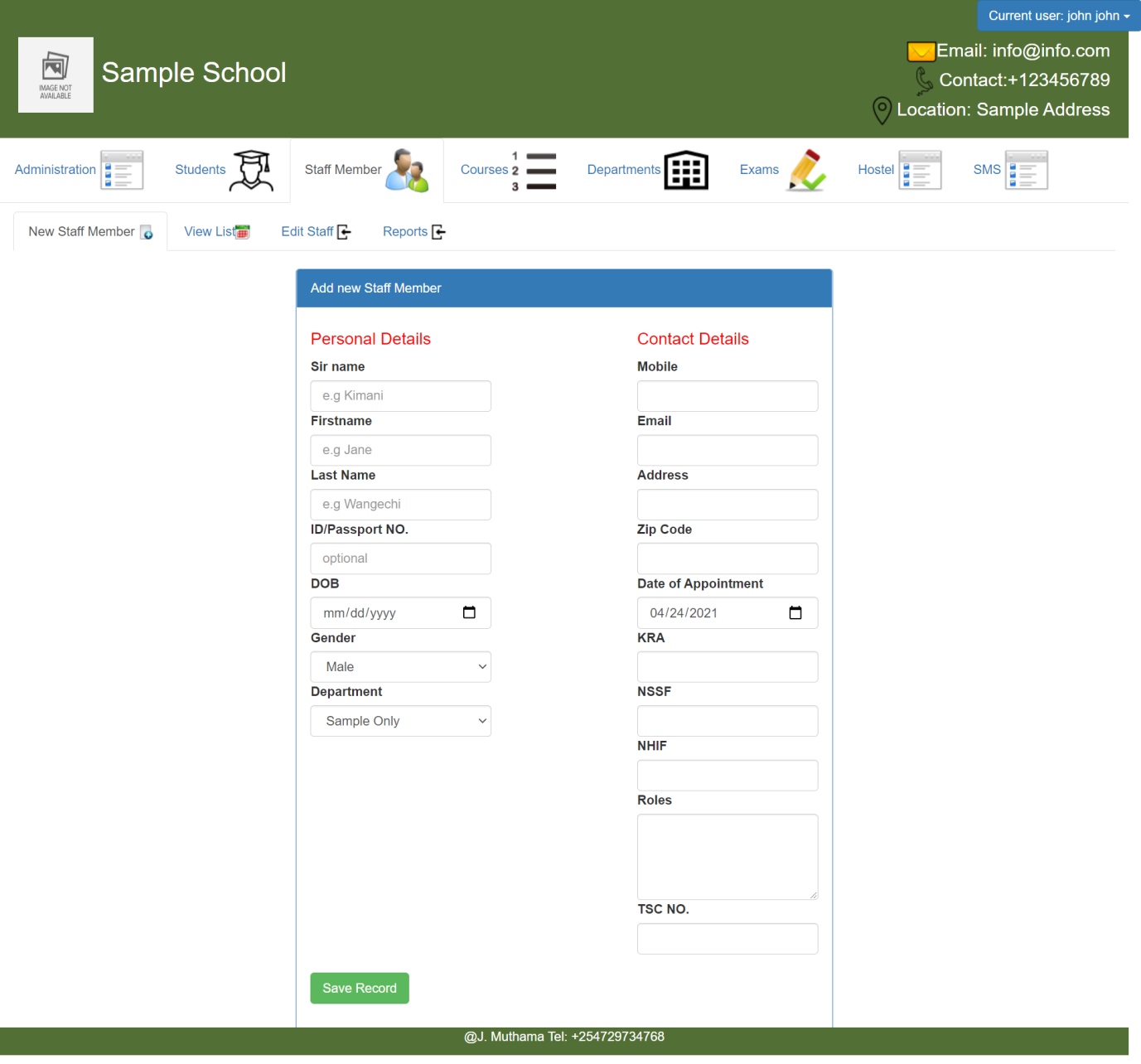
Code:

<head>

|  |
| --- |
|  |
|  | <!--<style type="text/css"> |
|  | body{ |
|  | margin-right: 10px; |
|  | margin-left: 10px; |
|  | } |
|  | </style> |
|  |  |
|  | --> |
|  | <style type="text/css"> |
|  | .panel-success{ |
|  | background-color: rgba(245, 245, 245, 0); |
|  | border:none; |
|  | } |
|  |  |
|  | </style> |
|  |  |
|  |  |
|  |  |
|  | <body background="assets/img/5.jpg" > |
|  | <!--end of heading section--> |
|  | <ul class="nav navbar-right top-nav"> |
|  | <div class="dropdown"> |
|  | <button class="btn btn-primary dropdown-toggle" type="button" data-toggle="dropdown" > |
|  | Current user: john&nbsp;john <span class="caret"></span></button> |
|  | <ul class="dropdown-menu"> |
|  | <li><a href="[manage\_account.php](http://localhost/school%20management%20system/manage_account.php" \t "_blank)"><i class="fa fa-users fa-lg"></i>&nbsp;View User</a></li> |
|  | <li><a href="[register\_form.php](http://localhost/school%20management%20system/register_form.php" \t "_blank)"><i class="fa fa-users fa-lg"></i>&nbsp;Add New User</a></li> |
|  | <li class="divider"></li> |
|  | <li><a href="[session\_logout.php](http://localhost/school%20management%20system/session_logout.php" \t "_blank)"><i class="fa fa-fw fa-power-off"></i>&nbsp;Log Out</a></li> |
|  | </ul> |
|  | </div> |
|  | </ul> |
|  | <!--\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*--> |
|  | <div style=" |
|  | font-family:Nyala, Arial; |
|  | text-align: left; |
|  | background-color: #526F35; |
|  | padding: 20px; |
|  | color:white; |
|  | width: 100%; |
|  | height: 150px;"> |
|  | <!--This codes to load the image loader--> |
|  | <div id="loading"> |
|  | <img id="loading-image" src="[assets/img/loader.gif](http://localhost/school%20management%20system/assets/img/loader.gif)" alt="Loading..." /> |
|  | </div> |
|  | <!--this is the heading section--> |
|  | <h2> |
|  |  |
|  | <div style="float:right; font-size:20px;text-align:right;"> |
|  |  |
|  | <img src="[assets/img/mail2.png](http://localhost/school%20management%20system/assets/img/mail2.png)">Email: info@info.com<br> |
|  | <img src="[assets/img/call1.png](http://localhost/school%20management%20system/assets/img/call1.png)">Contact:+123456789<br> |
|  | <img src="[assets/img/location.png](http://localhost/school%20management%20system/assets/img/location.png)">Location: Sample Address |
|  | </div> |
|  | </h2> |
|  | </div> |
|  | <div> |
|  | <ul class="nav nav-tabs"> |
|  | <li class="active"><a href="[homepage.php](http://localhost/school%20management%20system/homepage.php" \t "_blank)" >Administration <img src="[assets/img/details.png](http://localhost/school%20management%20system/assets/img/details.png)"></a></li> |
|  | <li ><a href="[students.php](http://localhost/school%20management%20system/students.php" \t "_blank)" >Students <img src="[assets/img/student48.png](http://localhost/school%20management%20system/assets/img/student48.png)"></a></li> |
|  | <li><a href="[staff.php](http://localhost/school%20management%20system/staff.php" \t "_blank)">Staff Member <img src="[assets/img/staff48.png](http://localhost/school%20management%20system/assets/img/staff48.png)"></a></li> |
|  | <li><a href="[course.php](http://localhost/school%20management%20system/course.php" \t "_blank)" >Courses <img src="[assets/img/course.png](http://localhost/school%20management%20system/assets/img/course.png)"></a></li> |
|  | <li><a href="[departments.php](http://localhost/school%20management%20system/departments.php" \t "_blank)" >Departments <img src="[assets/img/department.png](http://localhost/school%20management%20system/assets/img/department.png)"></a></li> |
|  | <li><a href="[markstep1.php](http://localhost/school%20management%20system/markstep1.php)" >Exams <img src="[assets/img/update.png](http://localhost/school%20management%20system/assets/img/update.png)"></a></li> |
|  | <li><a href="[hostel.php](http://localhost/school%20management%20system/hostel.php" \t "_blank)" >Hostel[<i>Premium</i>] <img src="[assets/img/details.png](http://localhost/school%20management%20system/assets/img/details.png)"></a></li> |
|  | <li><a href="[sms.php](http://localhost/school%20management%20system/sms.php" \t "_blank)">SMS <img src="[assets/img/details.png](http://localhost/school%20management%20system/assets/img/details.png)"></a></li> |
|  | <!--<li><a href="tab-8" role="tab" data-toggle="tab">Hostel <img src="assets/img/details.png"></a></li> |
|  | <li><a href="tab-7" role="tab" data-toggle="tab">Parents <img src="assets/img/details.png"></a></li>--> |
|  |  |
|  | </ul> |
|  | <div class="tab-content"> |
|  | <div class="tab-pane active" role="tabpanel" id="tab-1"> |
|  |  |
|  | <p> |
|  | <div class="table-responsive" > |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |

* + 1. **Students Daily Report**

Design:



Code:

|  |  |
| --- | --- |
|  | |
|  | <div class="row"> | |
|  | <div class="col-lg-6 col-lg-offset-3"> | |
|  | <div class="panel panel-primary"> | |
|  | <div class="panel-heading">Add new student</div> | |
|  |  | |
|  | <!-- /.panel-heading --> | |
|  | <div class="panel-body"> | |
|  | <div class="table"> | |
|  |  | |
|  | <form action="students.php" method="POST" enctype="multipart/form-data"> | |
|  | <label>Admission number</label> | |
|  | <input type="text" name="admission\_number" placeholder="Admission number" id="ssname" class="form-control"> | |
|  | <div style="float:left; position:relative"> | |
|  | <h4>Personal Details</h4> | |
|  |  | |
|  | <label>Sir name</label> | |
|  | <input type="text" name="sirname" placeholder="e.g Kimani" id="ssname" class="form-control"> | |
|  |  | |
|  | <label>Firstname</label> | |
|  | <input type="text" name="firstname" placeholder="e.g Jane" id="sfname" class="form-control" > | |
|  |  | |
|  | <label>Last Name</label> | |
|  | <input type="text" name="lastname" placeholder="e.g Wangechi" id="slname" class="form-control"> | |
|  |  | |
|  | <label>ID/Passport NO.</label> | |
|  | <input type="text" name="idno" class="form-control" placeholder="optional"> | |
|  | <label>DOB</label> | |
|  | <input type="date" name="dateofbirth" required class="form-control" max="2010-12-31"> | |
|  | <label>Gender</label> | |
|  | <select name="gender" id="gender" class="form-control"> | |
|  | <option>Male</option><option>Female</option></select> | |
|  |  | |
|  | <!-- <label>Country</label> | |
|  | <input type="text" name="country\_id" placeholder="kenya" value="Kenya" class="form-control" /> | |
|  |  | |
|  | <label>County</label> | |
|  | <select name="county\_id" id="county\_id" onchange="OnSelectionChange(this)" class="form-control"> | |
|  | </select> | |
|  |  | |
|  | <label> Constituency</label> | |
|  | <select name="constituency\_id" id="constituency\_id" onchange="onclick(this)" class="form-control"> | |
|  | </select> | |
|  | </div> --> | |
|  | <!--section two--> | |
|  | <div style="float:right; position:relative"> | |
|  | <h4>Contact Details</h4> | |
|  |  | |
|  | <label> Mobile</label> | |
|  | <input type="number" name="mobile" class="form-control"> | |
|  | <label> Email</label> | |
|  | <input type="email" name="email" class="form-control" placeholder="xyz@gmail.com"> | |
|  |  | |
|  | <label> Address</label> | |
|  | <input type="text" name="address" class="form-control" > | |
|  |  | |
|  | <label> Zip Code</label> | |
|  | <input type="text" name="zipcode" class="form-control"> | |
|  |  | |
|  | <label> Date of Admission</label> | |
|  | <input type="date" name="reg\_date" required class="form-control" value="2021-04-25" > | |
|  |  | |
|  | <label>Class. </label> | |
|  | <select name="course\_id" required class="form-control"> | |
|  | <option>Class 101</option><option>102</option></select> | |
|  | <font color="red"><i>2 Classes found [<a href='[course.php](http://localhost/school%20management%20system/course.php" \t "_blank)'>Add Class</a>] </i></font><br> | |
|  |  | |
|  | <!-- <label>Status</label> | |
|  | <select name="border" required class="form-control"> | |
|  | </select> | |
|  | NB:Additional charges to all borders | |
|  | </div> --> | |
|  | <!--this is section three--> | |
|  | <div style="float:left; position:relative; clear:both;"> | |
|  |  | |
|  | <h4>Guardian/Parent and other Details for student</h4> | |
|  |  | |
|  | <label> Sir name</label> | |
|  | <input type="text" name="psirname" id="sn" class="form-control"> | |
|  |  | |
|  | <label>Firstname Name</label> | |
|  | <input type="text" name="pfirstname" id="fn" class="form-control"> | |
|  |  | |
|  | <label>Last Name</label> | |
|  | <input type="text" name="plastname" class="form-control"> | |
|  |  | |
|  | <label>Mobile</label> | |
|  | <input type="number" name="pmobile" class="form-control"> | |
|  |  | |
|  | <label>Emergency Contact</label> | |
|  | <input type="number" name="emergency\_contact" class="form-control"> | |
|  |  | |
|  | <label>Relationship </label> | |
|  | <input type="text" name="prelationship" placeholder="Father" class="form-control"><br> | |
|  |  | |
|  |  | |
|  | <input type="submit" name="register" value="Save Record" class="btn btn-success"><br><br> | |
|  | </div> | |
|  | </form></div> | |
|  | </div> | |
|  |  | |
|  |  | |
|  |  | |

### Exception

IT has permeated in every sphere of life and schools across the world are no exception. In fact, the schools worldwide are using school management system software to make their teaching and administration sharper and more efficient. These institutions have been very quick to adapt to the changing dynamics of "good service" and hence did not waste any time in getting what turned their management into more well-ordered and nimbler one

### User Interface Design

User interface design (UI) or user interface engineering is the [design](http://en.wikipedia.org/wiki/Design) of [user interfaces](http://en.wikipedia.org/wiki/User_interface) for machines and software, such as [computers](http://en.wikipedia.org/wiki/Computer), [home appliances](http://en.wikipedia.org/wiki/Home_appliance), [mobile devices](http://en.wikipedia.org/wiki/Mobile_device), and other [electronic devices](http://en.wikipedia.org/wiki/Electronics), with the focus on maximizing the [user experience](http://en.wikipedia.org/wiki/User_experience). The goal of user interface design is to make the user's interaction as simple and efficient as possible, in terms of accomplishing user goals ([user-centered design](http://en.wikipedia.org/wiki/User-centered_design)).

Good user interface design facilitates finishing the task at hand without drawing unnecessary attention to it. [Graphic design](http://en.wikipedia.org/wiki/Graphic_design) and [typography](http://en.wikipedia.org/wiki/Typography) are utilized to support its [usability](http://en.wikipedia.org/wiki/Usability), influencing how the user performs certain interactions and improving the aesthetic appeal of the design; design aesthetics may enhance or detract from the ability of users to use the functions of the interface.

**Chapter 5**

# System implementation

# System implementation

Implementation refers to the final process of moving the solution from development status to production status. Depending on your project, this process is often called deployment, go-live, rollout or installation. For the purposes of Life cycle Step, all of these terms are synonymous with "implementation."

There is no single way to implement an application. It depends on the characteristics of your project and the solution. Some implementations are as easy as saying “we are now live.” This type of implementation can work when the solution is brand new and you are developing and testing in what will become the production environment. In these cases, implementation is just a state of mind. One day the solution is in development, and the next day it is in production.

## User Training

The initial training classes for users are held, and training materials are delivered at the classes. Some help desk personnel should attend the initial user training class. More training classes can be scheduled later, as new personnel start using the application. Training is done on the user acceptance test system, accessing the test database or a special training database.

## Distributed User Documentation

User documentation that was finalized in User Acceptance Testing is now distributed and in the users' possession.

### Finalized System Documentation

System documentation corrected with all updates from the testing phases is handed over to production support.

### Installed Production System

The production system is installed in the appropriate production environment or on the appropriate production server, and on any client workstations that require it.

### Post-Implementation Review Summary

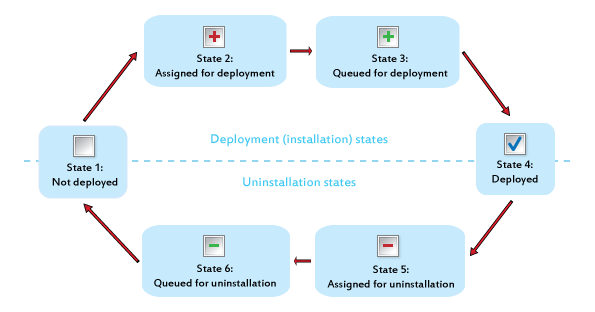
This report is produced after the post-implementation review meeting, and contains a summary of the project success criteria that were met, success criteria that were not met and reasons for the problem, what we can learn from the project to improve practices for the next project. In particular, the report should identify any techniques or practices used in this project that worked extremely well, and which the project team feels would benefit current and future projects.

### Methodology Compliance Form

This form is initialized by the project team, and completed by a methodology representative who has reviewed the project documentation and found it acceptable. It is completed in Word.

## Software deployment

Software deployment is all of the activities that make a [software system](https://en.wikipedia.org/wiki/Software_system) available for use. The general deployment process consists of several interrelated activities with possible transitions between them. These activities can occur at the [producer](https://en.wikipedia.org/wiki/Software_developer) side or at the [consumer](https://en.wikipedia.org/wiki/User_(computing)) side or both. Because every [software system](https://en.wikipedia.org/wiki/Software_system) is unique, the precise [processes](https://en.wikipedia.org/wiki/Process_(computing)) or [procedures](https://en.wikipedia.org/wiki/Algorithm) within each activity can hardly be defined. Therefore, "deployment" should be interpreted as a general process that has to be customized according to specific requirements or characteristics. A brief description of each activity will be presented later.



**Figure 10:** Deployment Process

## Deployment Activities

### Release

The [release](https://en.wikipedia.org/wiki/Software_release) activity follows from the completed [development](https://en.wikipedia.org/wiki/Software_development_process) process. It includes all the operations to prepare a system for [assembly](https://en.wikipedia.org/wiki/Compiler) and transfer to the customer site. Therefore, it must determine the [resources](https://en.wikipedia.org/wiki/Resource_(computer_science)) required to operate at the customer site and collect information for carrying out subsequent activities of deployment process.

### Install and activate

Activation is the activity of starting up the [executable](https://en.wikipedia.org/wiki/Executable) component of software. For simple systems, it involves establishing some form of [command](https://en.wikipedia.org/wiki/Command_(computing)) for execution. For complex systems, it should make all the supporting systems ready to use. (Not to be confused with the common use of the term activation concerning a software license, which is a function of [Digital Rights Management](https://en.wikipedia.org/wiki/Digital_Rights_Management#Limited_install_activations) systems.)

In larger software deployments, the working copy of the [software](https://en.wikipedia.org/wiki/Software) might be installed on a production server in a production environment. Other versions of the deployed software may be installed in a [test environment](https://en.wikipedia.org/w/index.php?title=Test_environment&action=edit&redlink=1), [development environment](https://en.wikipedia.org/wiki/Development_environment_(software_development_process)) and disaster recovery environment.

Further information: [Installation (computer programs)](https://en.wikipedia.org/wiki/Installation_(computer_programs))

### Deactivate

Deactivation is the inverse of activation, and refers to shutting down any executing components of a system. Deactivation is often required to perform other deployment activities, e.g., a software system may need to be deactivated before an update can be performed. The practice of removing infrequently used or obsolete systems from service is often referred to as [application retirement](https://en.wikipedia.org/wiki/Application_retirement) or application decommissioning.

### Adapt

The [adaptation](https://en.wikipedia.org/wiki/Adaptation_(computer_science)) activity is also a process to modify a software system that has been previously installed. It differs from updating in that adaptations are initiated by local events such as changing the [environment](https://en.wikipedia.org/wiki/Runtime_environment) of customer site, while updating is mostly started from remote software producer.

### Update

The update process replaces an earlier version of all or part of a software system with a newer release.

### Built-In

Mechanisms for installing updates are built into some software systems. Automation of these update processes ranges from fully automatic to user initiated and controlled. [Norton Internet Security](https://en.wikipedia.org/wiki/Norton_Internet_Security) is an example of a system with a semi-automatic method for retrieving and installing updates to both the antivirus definitions and other components of the system. Other software products provide query mechanisms for determining when updates are available.

### 

### Version tracking

Version tracking systems help the user find and install updates to software systems installed on PCs and local networks.

* Web based version tracking systems notify the user when updates are available for software systems installed on a local system. For example: Version Tracker Pro checks software versions on a user's computer and then queries its database to see if any updates are available.
* Local version tracking system notifies the user when updates are available for software systems installed on a local system. For example: Catalog stores version and other information for each software package installed on a local system. One click of a button launches a browser window to the upgrade web page for the application, including auto-filling of the user name and password for sites that require a login.
* Browser based version tracking systems notify the user when updates are available for software packages installed on a local system. For example: [wfx-Versions](https://en.wikipedia.org/w/index.php?title=Wfx-Versions&action=edit&redlink=1" \o "Wfx-Versions (page does not exist)) is a Firefox extension which helps the user find the current version number of any program listed on the web.

### Uninstall

Uninstallation is the inverse of installation. It is the removal of a system that is no longer required. It also involves some reconfiguration of other software systems in order to remove the uninstalled system’s [files](https://en.wikipedia.org/wiki/Computer_file) and [dependencies](https://en.wikipedia.org/wiki/Coupling_(computer_science)).

### Retire

Ultimately, a software system is marked as [obsolete](https://en.wikipedia.org/wiki/Obsolete) and [support](https://en.wikipedia.org/wiki/Technical_support) by the producers is withdrawn. It is the end of the [life cycle of a software product](https://en.wikipedia.org/wiki/Product_life_cycle_management).

# 

# Chapter 6

# System Test

# Software Testing

Software testing is an investigation conducted to provide stakeholders with information about the quality of the product or service under test. Software testing can also provide an objective, independent view of the [software](https://en.wikipedia.org/wiki/Software) to allow the business to appreciate and understand the risks of software implementation. Test techniques include the process of executing a program or application with the intent of finding [software bugs](https://en.wikipedia.org/wiki/Software_bug) (errors or other defects).

It involves the execution of a software component or system component to evaluate one or more properties of interest. In general, these properties indicate the extent to which the component or system under test:

1. Meets the requirements that guided its design and development,
2. Responds correctly to all kinds of inputs,
3. Performs its functions within an acceptable time,
4. Is sufficiently usable,
5. Can be installed and run in its intended [environments](https://en.wikipedia.org/wiki/Operating_environment), and
6. Achieves the general result its stakeholder’s desire.

## Principles of software testing

**Principle 1:** Testing shows that there are defects present in the software  
A self-explanatory point, it states that when a project is tested, it is checked for possible defects or bugs by creating different **software testing strategies**.

**Principle 2:** Testing software exhaustively is impossible.  
This means that testing software is not possible exhaustively and instead, testers need optimum time to test an application, which is based on the risk assessment of the same.

**Principle 3:** Testing software early.  
It is imperative to start testing software as early as possible. This ensures that the defects can be captured and fixed within the stipulated time-frame, thereby allowing developers to deliver the software to the clients on time.

**Principle 4**: Clustering the defects.  
Defect clustering simply state that a small number of modules in an application contains maximum defects detected.

**Principle 5**: The Pesticide Paradox.  
When the same tests are repeated over time and again, then the test cases do not find any new bugs. This situation gives rise to a new principle known as the Pesticide Paradox. However, this can be overcome by reviewing and revising the test cases and adding new and different test cases.

**Principle 6**: Testing is dependent on context.  
This means that when you test a mobile application, it will be on different grounds than while testing a web application. Similarly, testing a Mac application will be different than testing an Android application and the likes.

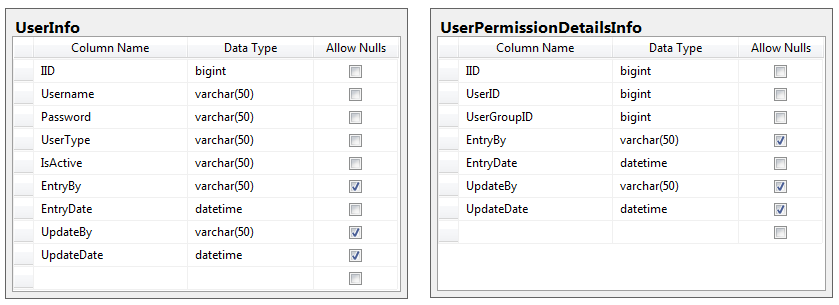
**Principle 7**: Absence of errors – fallacy.  
This principle merely points out to the fact that finding and fixing defects in a software system is of no use if the system build in itself is unusable and is unable to meet the users’ needs and requirements.

# Chapter 7

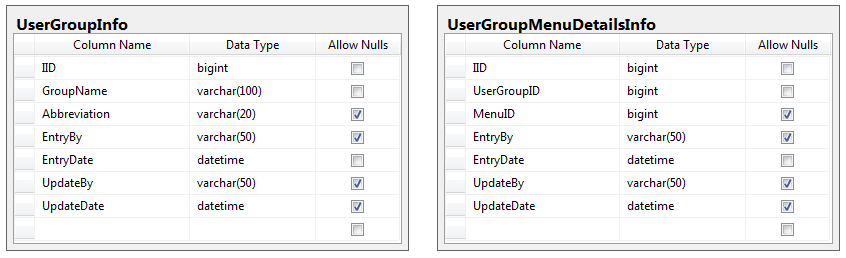
# User Manual

# Information of Table

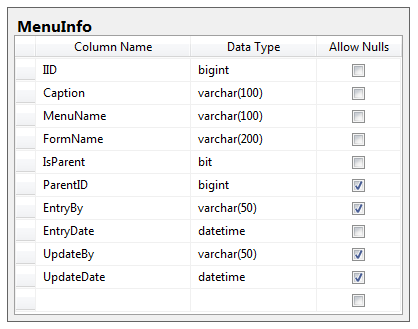
## User Table

****

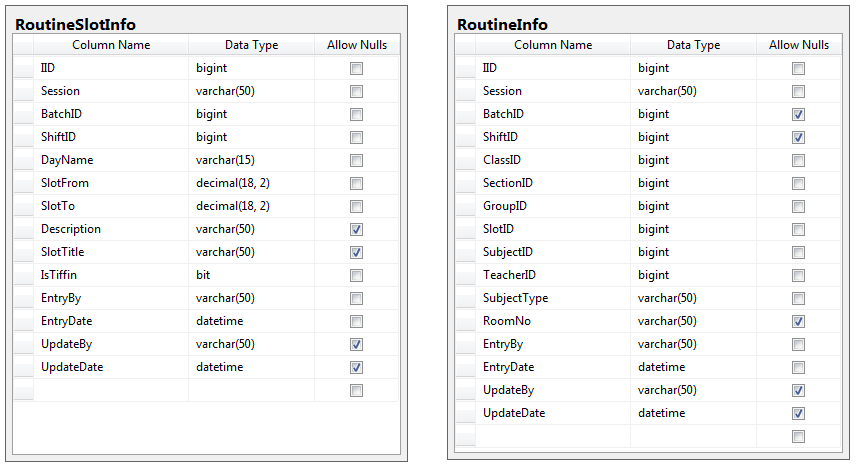
## User Group Table

****

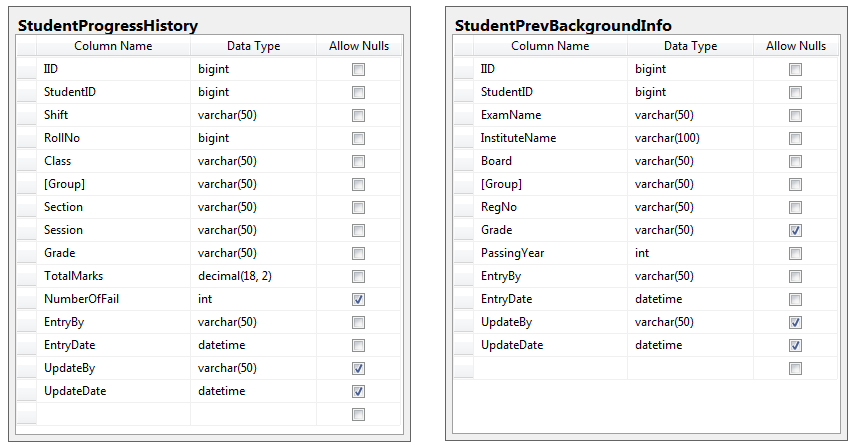
## Menu Table

****

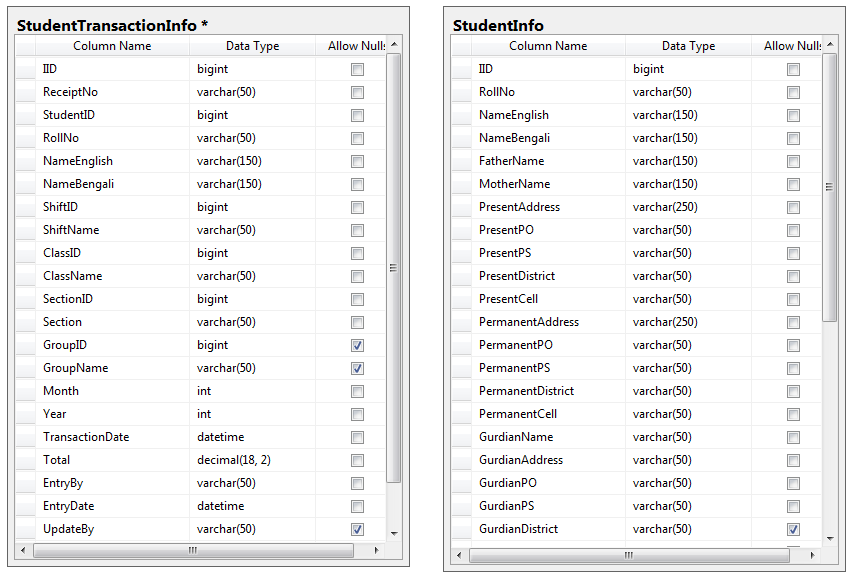
## Routine Table



## Student Progress Table



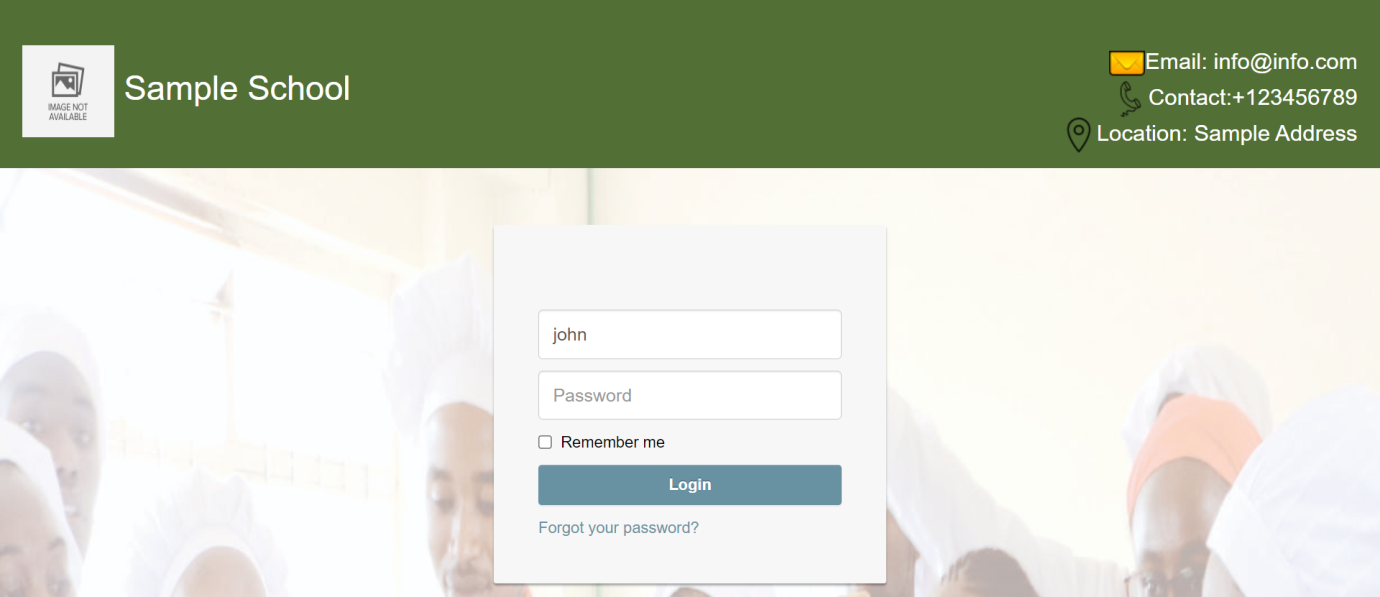
## Student Table



# Manual

## Login

After installing the software system user might be treated as a public user. The user only can see three functional tools bar named File, View and help. When the user clicks into option Login, A Login form appears there into interface. User has to enter the User-ID and Password to login. After typing all required info’s user have to enter login.

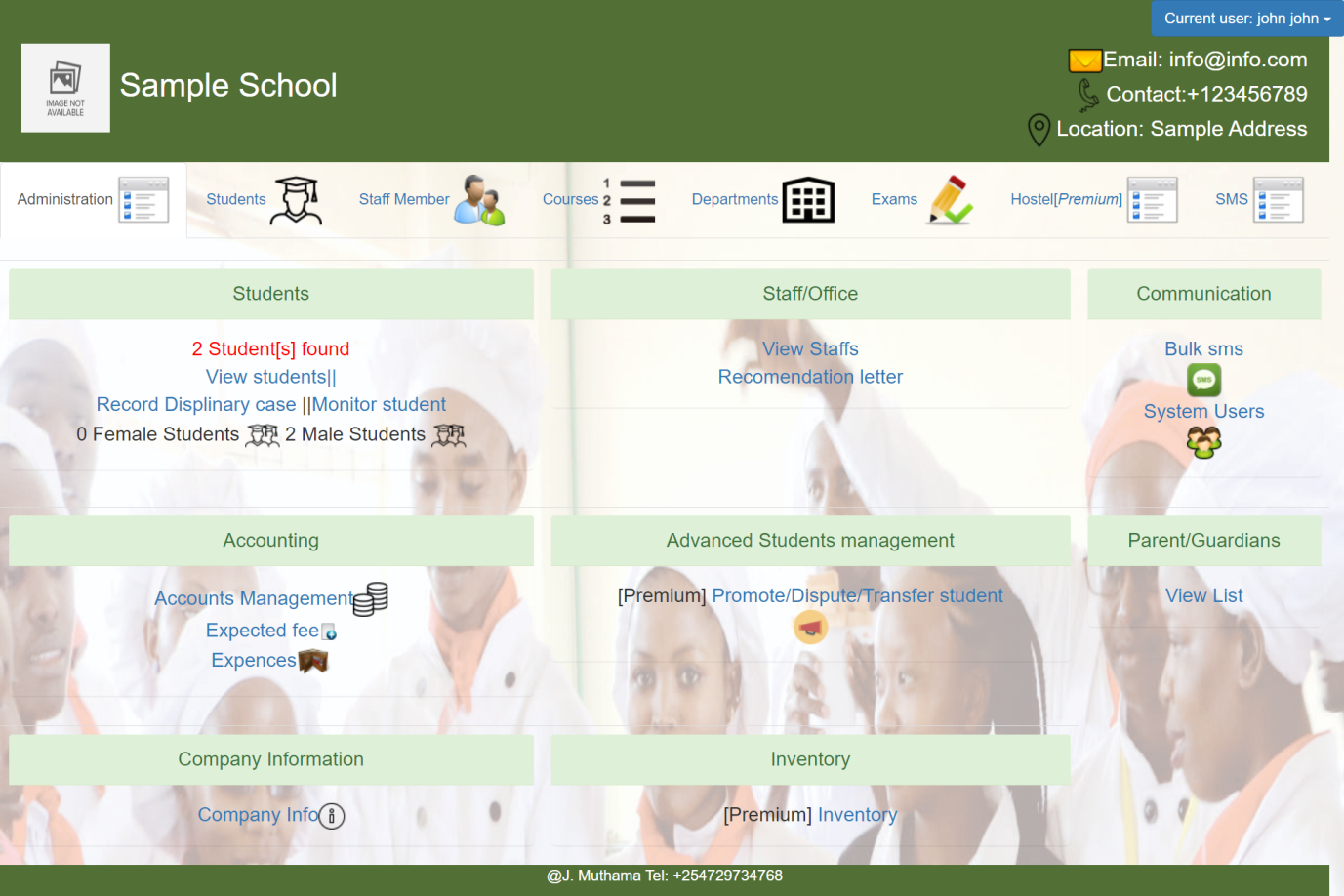
**

**Figure 11:** Login Process

The user must have logged in into there to have the full access. Into File tools bar there are six objective options, where the first option is Login. After Login user can see and access with all other functionalities.

## User Management

This system shows all the functionalities of a User. They Are user Group, Group Permission and User Entry.



**Figure 12:** Administration

### User Group

The User can Create or Delete any User Groups. User can also Reset the existing user Groups. User needs just a Group name and abbreviation for Creating a User Group. When a User finished creating a User Group, It appears at the right side of the User Group Form with its Entry by and Entry Date.

**Figure 13:** User Group Entry

### Group permission

The Groups, user created into the User group can’t access into all the sections. A User Group can access only those sections whose are given. All User Groups cannot access all sections. So How many sections and which sections are permitted to be accessed by a User Group is Decided here.

**Figure 14:** Group Permission

### User Entry

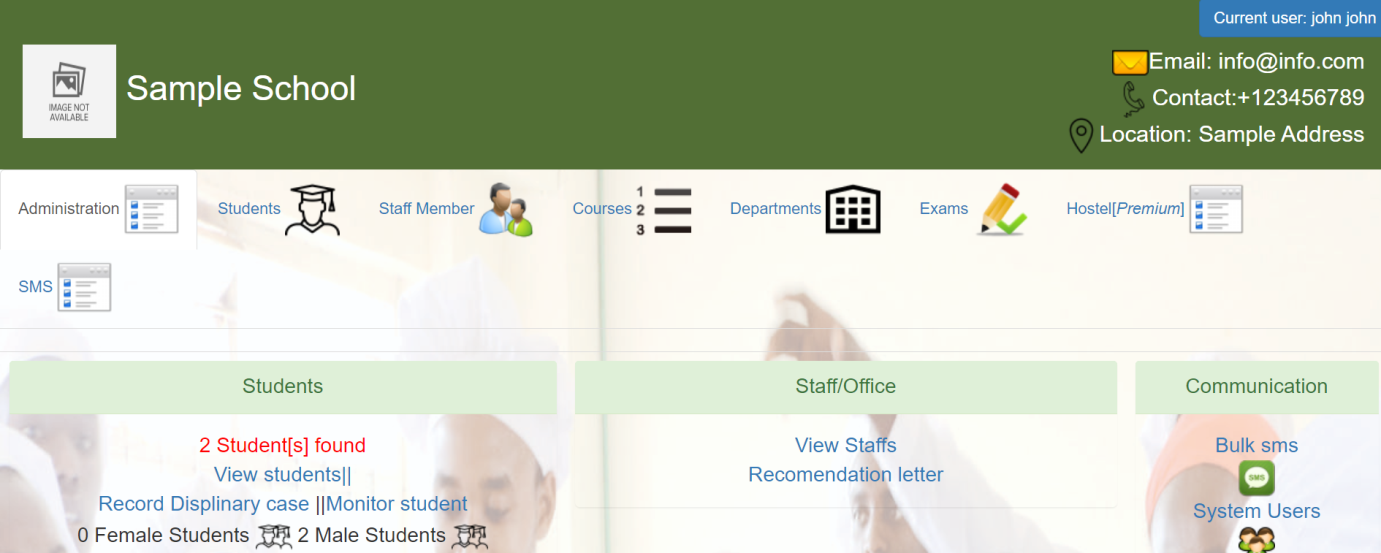
The Super admin can add other users into this system if necessary. It is quite foggy to manage all the systems alone, so user can add several users. There just needs a valid User-name and a password. User must select the Sections which are permitted to that New User.

**Figure 15:** User Entry

The User can be active or non-active. After creating the User-Ids, it will appear into the right side of the form with some relational information’s active, group permission, Entry By and Entry date. We can also reset/Update the info’s if needed.

## Administration

The system management shows all the functionalities of arranging and organizing classes, shifts and sections in a polite and easy way. It shows shifts, classes, sections, subject groups, group subjects and designation when necessary to know.



### Shift

It shows the shifts of any school management system. It May b Day or Morning. Nowadays some Institutions are now started night shifts also.

### Class

This UI is for enter all existing class in an educational institute. We also assign a Class Short for easy handling.

### Subject

This is Input screen for enlistment of subject with subject code and subject type alike – subjective/ selective.

### Section

User can see all the sections of all classes with their **igure 19:** SubjectEntry

individual section codes. If the user wants to add a new section for any identical class and shifts it needs just a section name with its section codes. After saving it will appear into the right side of user interface. User also can delete or reset any section data if needed.

### Student Group

Student group shows the group of any student. User can create any user group with just only a Group-name. Usually there are 3 student groups named Science, Business Communication and Humanities.

### 8.3.6. Group Subjects

We assign subject name for under subjective, Selective and optional just select particular class and group.

### Designation

Designation means those, with whose strong monitoring and love a institution runs properly. With the help of this, User could see the entire designations name at a time. User can add other designation name according to necessity. User also can reset or delete the previous data/record if needed.

## Attendance

### Student Attendance

This system shows the student attendance. If the User just want to check any required class’s and section’s students attendance of a desired day, User only have to enter the class name, section name and that date, Ant may b the attendance report of a huge number of students would be shown in a couple of minute. User can save, delete or reset the data’s if needed.

### Employee Attendance

This functionality shows all of the employee’s attendance as like students and input attendance.

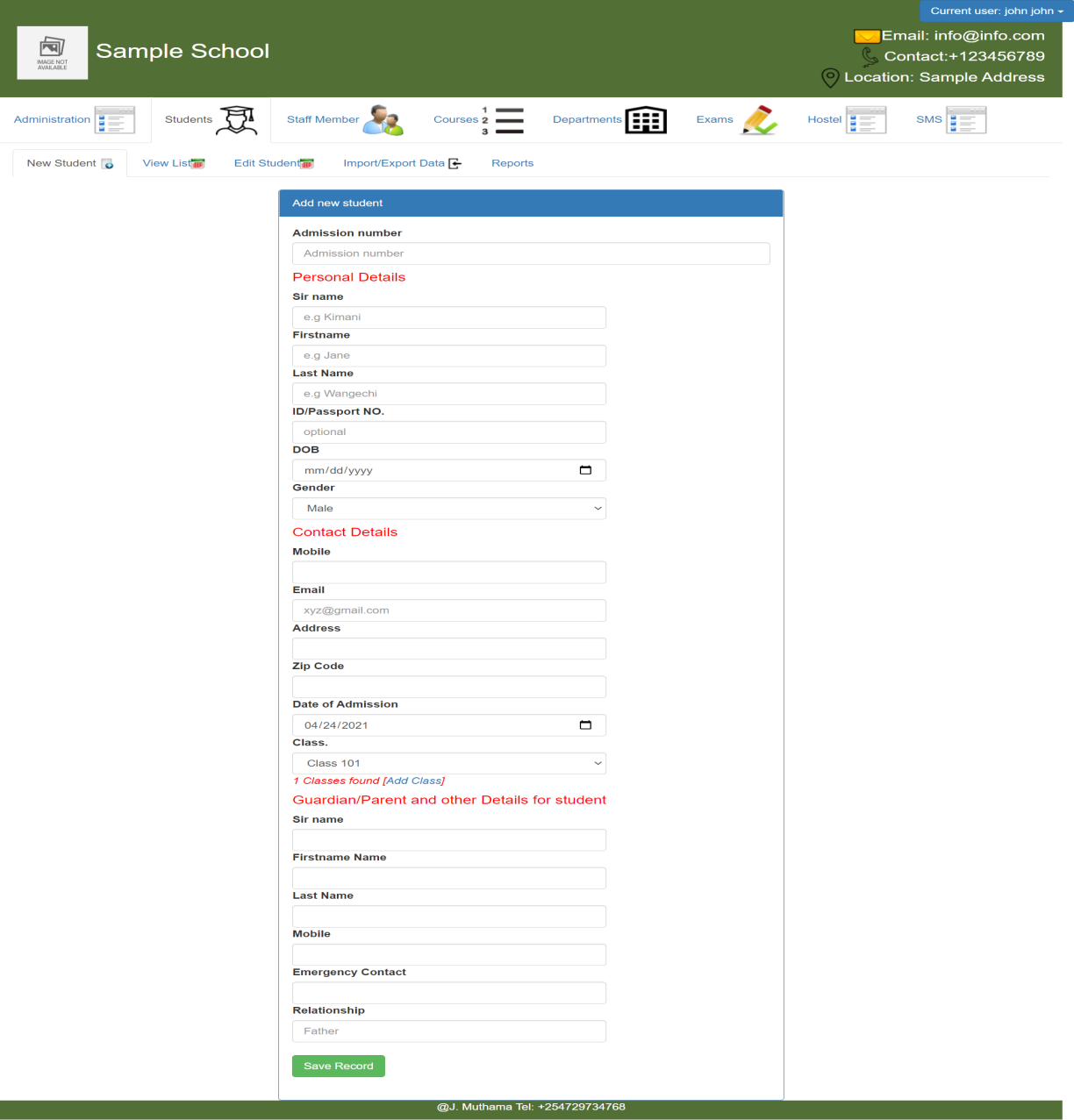
### Holidays

Holiday input screen is for institute can inputted their festival time and weekend for easy to concern anybody with short notes about it like remarks. Friday is by default holiday.

## PIMS (Personal Information Module System)

This Functionality represents Personal Information Module System. It includes/shows all kind of student and employee in information, personal or professional. Student Admission, Student promotion, Employee Registration, Student List, Employee List, Student Details report, student list Report, Employee Details Report and Employee List report all are given here if needed.

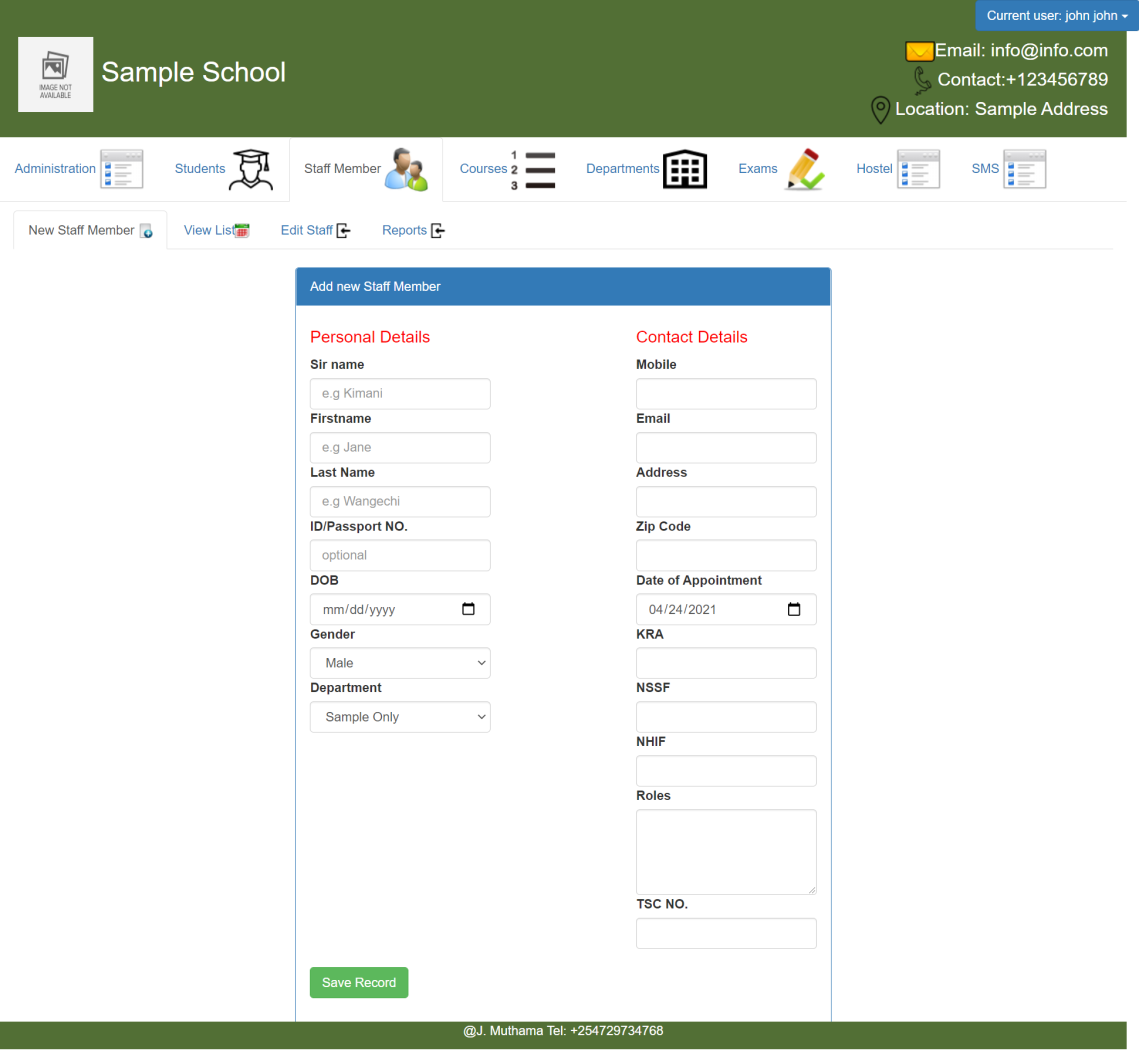
### Student Admission

As this is a School management System there must be students, and also there must have an organized way to admit them. At here, User can admit students with fulfilling this form. After getting into all information user can save them as students of that institutions. It also include Students photo on it. It’s easy to identify. Guardian’s information is available with mobile number and photo.

**Figure 28:** Student Admission

### Employee Registration

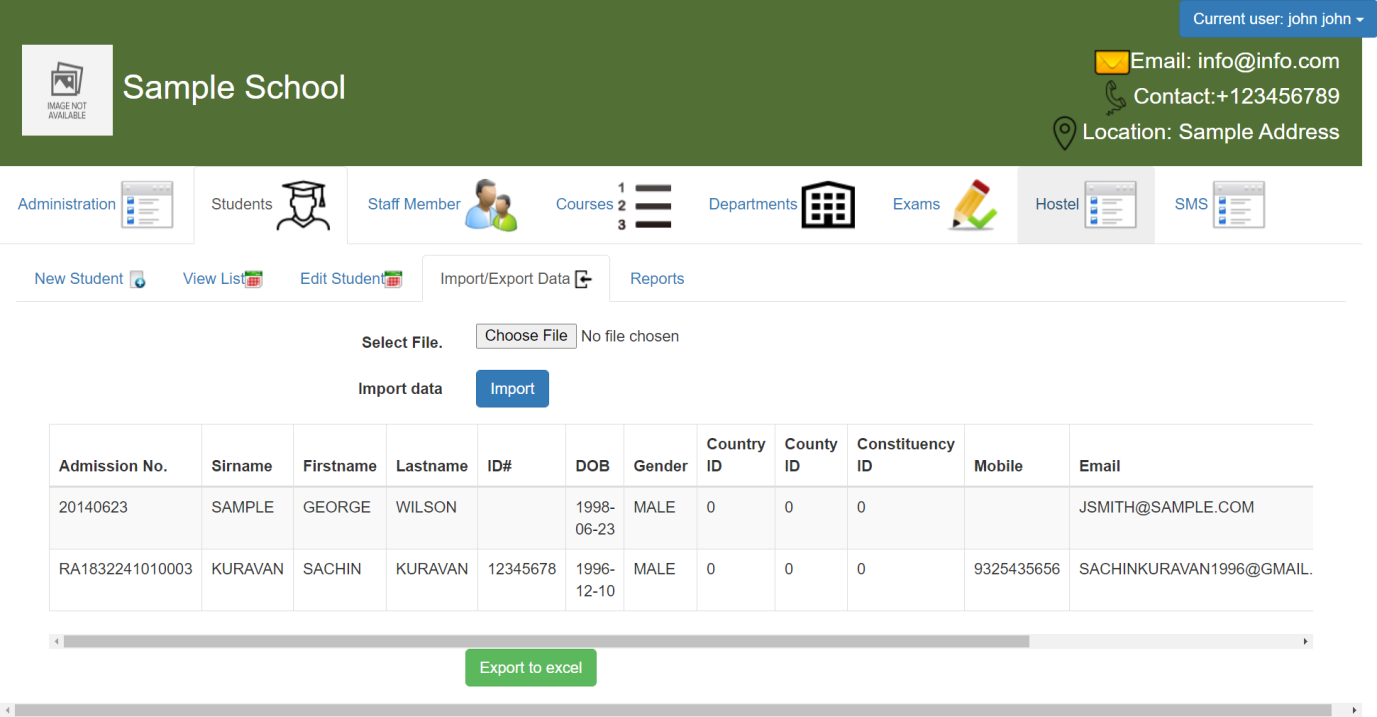
Admin can Register Employs according to necessity of management system. User can also update/Reset the old employ’s data’s if necessary. It saves the employs photo and digital signature which is more secure and identical.



**Figure 29:** Employee Registration

### Student List

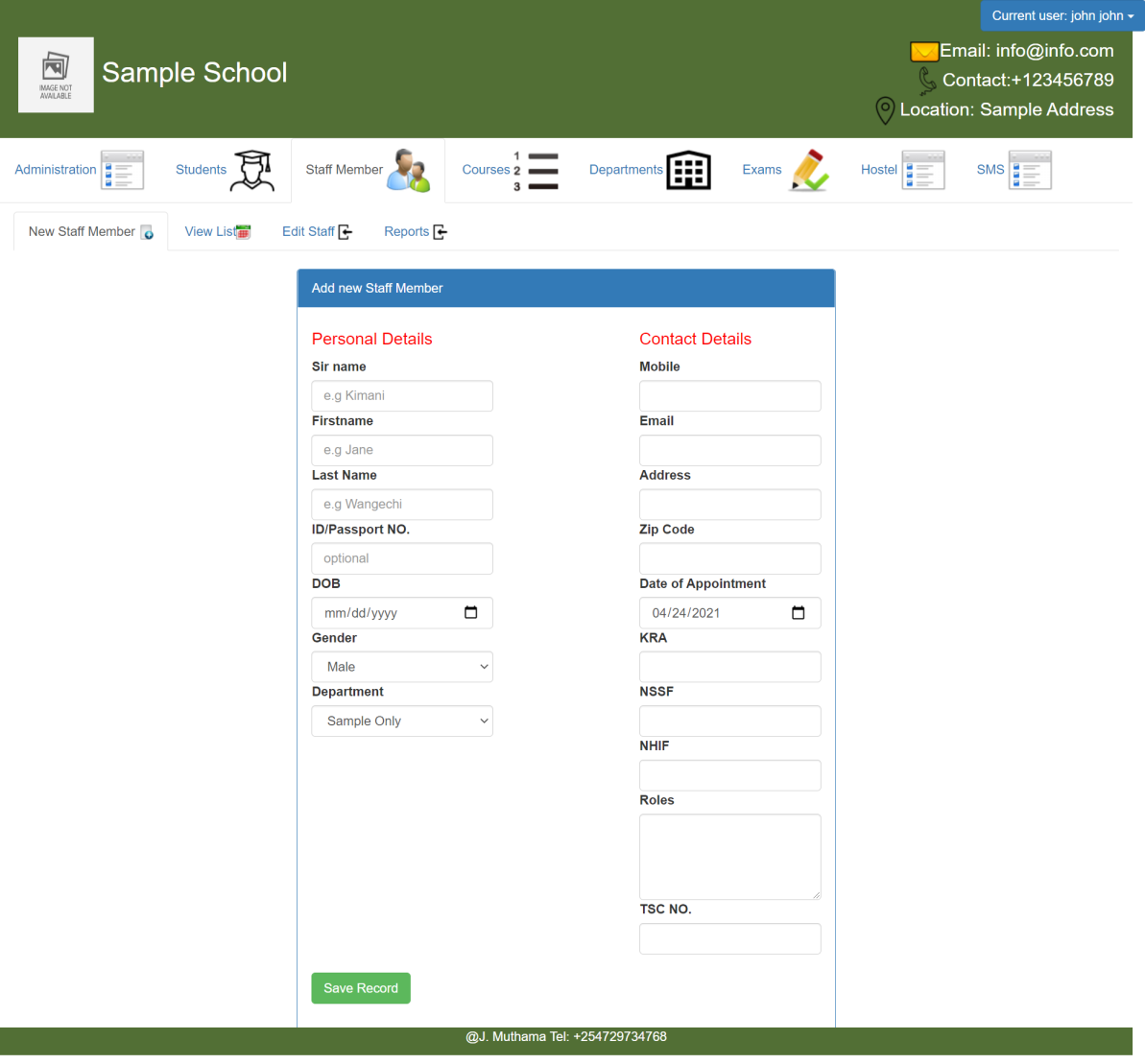
Admin / User can watch the student list. According to a School Management system, there are a lot of Students. So, may be User want to watch only some selective students list. As it, users have just input the required class and section onto it and here result comes. Admin can edit any information from here just click edit button and update data.



**Figure 30:** Student List

### Employee List

It shows the entire employee’s data and information with their photo and digital signatures.



### Student Details Report

User can see the full details report of any particular student if needed from student list. It is depicted into figure 22.

### Student list Report

User can also see the Student list Report from student list report class wise.

### Employee Details Report

User can watch a particular employ’s details report here like their Educational Information, Professional Information, Skill and training Info’s.

### Employee List Report

Admin/ User can view employee list report from Employee List. It can be filtered by Designation, Marital Status or Blood group.

## Result

**Figure 34:** Result Module

### Students Marks Policy

Every Educational Institute has its own marks policy. So they can input their marks policy by this UI. Simply Select class, session and groups then show. Under all subject they can assign marks policy like – have class test or not, class test total marks, pass marks, is marks in hand writing/ spelling, written pass marks, total pass marks etc.

**Figure 35:** Students Marks Policy

### Subject Wise Marks Entry

Subject Wise Marks Entry is marks entry for individual subjects marks entry. When select particular shift, class, year, group, exam-term and subject then its show all student list in grid view. It’s an easy UI for user who entry the marks. He simply input the marks for particular student and save it. If anybody absent in exam then it should be marks in checkbox. When result is published those student result shows automatically absent in those particular subject

## Routine

**Figure 37:** Routine Module

### Routine Slot Entry

It’s a slot for routine entry. It’s just easy to entry the routine. Just select Shift, Year, Day and Time From (7:00 AM) – Time To (7:30 AM) and Slot Title (1st period). Time Slot is individual period time (7:00 AM- 7:30 AM). Almost these period are generated in routine which we entry in Routine Slot Entry UI. Break Time is also assignable by mark the checkbox is break.

### E:\school management system 1\staff add page.pngRoutine Entry

Routine Entry view’s as day and period wise routine for particular Class, Shift and Year. We just select Subject Name, Subject Type, Teacher’s Name and assigns room number for each and every period and individual day. It’s a weekly view. When all period’s and days are inputted correctly then save the data for next step.

### Student Class Routine

It generated the student Class Routine. Just simply select Shift, Class, Section, Year and Group and then show. We can print this report from here for student use and also export to pdf and excel.

### Teacher’s Schedule

This UI is generated Teacher’s Schedule. Just Select Shift, Year and Teacher Name and show the routine. It’s a class schedule for individual teachers with date –time and room number. Teacher’s can print their schedule for further use and also make pdf or excel file from here.

## Accounts

We can store the user information through database to revise bill for future reference, after paying the amount they will received the receipt

### Debit Head

We can create all expense head as Debit Head. We create debit head as parent head. If we have a child head under parent head then mark is parent head. If this head is linked with class then select linked with class and select class.

### Credit Head

We can create all income head as Credit Head. We create credit head as parent head. If we have a child head under parent head then mark is parent head and under parent head we create child head. If this head is linked with class then select linked with class and select class.

## 

### Bank Deposit

When an institute deposits any amount of money into a bank they can input deposit information into this system like bank name, branch name, deposit amount with time and date. Institute can use this information for any further query.

### Bank Withdraw

When an institute withdraws any amount of money from a bank they can input withdraw information into this system like bank name, branch name, deposit amount with time and date. Institute can use this information for any further query.

### Income

Income UI is most crucial in Accounts module. By income UI we save income source with credit head and amount of income. We simply click into head code and select credit head code for income source. Head name is automatically inputted into head name. Income is need for all accounts report.

### Expense

Expense UI is also crucial in Accounts module. By expense UI we save expense source with debit head and amount of income. We simply click into head code and select debit head code for income source. Head name is automatically inputted into head name. Expense is need for all accounts report.

**Chapter 8**

# Future Works and conclusion

# Future Expandability

We are analysis User Management, Student Profile Management, Routine Management, Result Management, Employee Management and Accounts Management etc. We are complete software that has included this entire requirement.

All good software may have some limitations this software may be some limitations. We are trying to fix it as per possible.

## Conclusion

The application Education Management Software is the total package for the school or college management system requirements. With the help of menu bar users can interact with software very easily almost every object has been provided with tool bar.

This application is also supports terminal services so that database will be more secure by centralized the database.  
In fine we want to say we are not yet an experienced system analyst, but tried our best to fulfill this project.

**Reference**

[**https://en.wikipedia.org/wiki/PHP**](https://en.wikipedia.org/wiki/PHP)

[**https://www.phptpoint.com/php-tutorial/**](https://www.phptpoint.com/php-tutorial/)

[**https://www.tutorialspoint.com/php/index.htm**](https://www.tutorialspoint.com/php/index.htm)

[**https://www.w3schools.com/php/**](https://www.w3schools.com/php/)