**1.INTRODUCTION**

**1.1 Problem Definition**

As the name specifies “HOSTEL MANAGEMENT SYSTEM “is a software developed for managing various activities in the hostel. For the past few years the number of educational institutes are increasing rapidly. Thereby the number of hostels are also increasing for the accommodation of the students studying in this institute. And hence there is a lot of strain on the person who are running the hostel and software’s are not usually used in this context. This particular project deals with the problems on managing a hostel and avoids the problems which occur when carried manually

**1.2 Project Description**

The online HOSTEL MANAGEMENT SYSTEM is web based software to provide college students accommodation to the university hostel more efficiently. This project also keeps details of the hostellers and applied students. It is headed by warden. He will be the administrator for accommodation of a large number of students into hostel

This document is intended to minimize human works and make hostel allocation is an easier job for coast students and hostel authorities by providing online application for hostel, automatically select the students from the waiting list and mess calculation, complaint registration, notice board etc. Hostellers can view notice board, hostel fee and mess menu by login into the online system.

**Functions**

Hostel Management System Provides the searching facilities based on various factors. Such as Beds, Rooms, Allotees, Rent Hostel Management System also manage the details online for students details, employees details, Rent.

It tracks all the information of Hostel, Payments, Allotees etc Manage the information of Hostel Shows the information and description of the Beds, Rooms To increase efficiency of managing the Beds, Hostel It deals with monitoring the information and transactions of Allotees.

Manage the information of Beds Editing, adding and updating of Records is improved which results in proper resource management of Hostel Management System data.

**Purpose**

The hostel management needs to create the hostel management system (HMS) to organize the rooms, mess, students record and the other information about the students. how many students can live in a room, and the students of the hostel can be recognized from their ID card number.

**Scope**

This software product the hostel management to improve their services for all the students of the hostel. This also reduce the manual work of the persons in admin penal and the bundle of registers that were search when to find the information of a previous student, because through this system you can store the data of those students who had leaved the hostel three years ago. Through this you can check the personal profile of all the current students within few minutes the data base of the system will help you to check a particular one. The system will help you to check the mess bills of every student and the student’s hostel dues. The students of the hostel will be recognized from the ID number allocated at the room rental time.

**Hostel management system is divided into two modules**.

* Admin module
* User module

#### Admin Module

* Login – This module is used for admin login.
* Dashboard – Admin dashboard related total students, total rooms and total courses.
* Course Creation- Admin can add and manage course(Add, edit and delete)
* Rooms- Admin can create rooms(seater, fees)(Add, edit and delete)
* Students Registration- Admin can register student profile.
* Manage Student- (View and Delete)
* Admin Profile
* Change password
* Access log : admin can see user login access details .
* Logout

#### User Module

* User Registration
* User login
* User forgot Password
* User Profile : User can update own profile after login
* Change Password : Allow user to change the password.
* Book Hostel : User can book hostel
* Hostel details : user can see hostel details
* Access log : User can see own login access and Logout

**2. System Study**

**2.1 Existing system**

The existing system is a manually maintained system.All the hostel records are to be maintained for the details of each student, fees details ,room allocation ,attendance etc.

All these details are entered and retrieved manually on paper work.

Disadvantages of the existing system are

* Time consuming
* Inaccuracy of data
* More human error
* Low security
* Difficult to handle data
* Difficult to update data

**2.2 Proposed system**

The proposed system is the computerized version of the existing system .

Provides easy and quick access over the data .

Advantages of the proposed system are :

* Good security for user information
* Easy to handle
* Easy data updating
* Authorization schemes
* Less human error

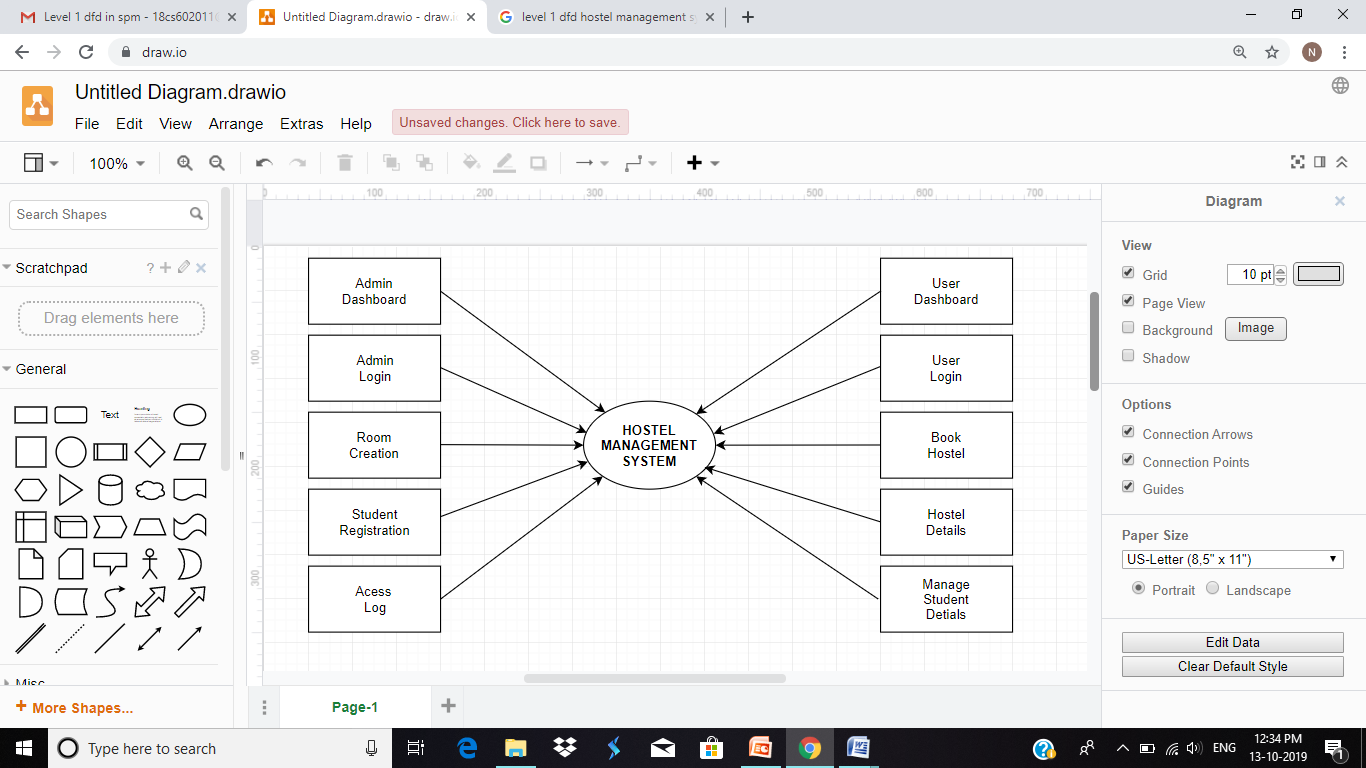
**2.3 DATA FLOW DIAGRAM :**

A Data Flow Diagram (DFD) is a graphical representation of the "flow" of data through an [information system](http://en.wikipedia.org/wiki/Information_system), modelling its process aspects. A DFD is often used as a preliminary step to create an overview of the system, which can later be elaborated. DFDs can also be used for the [visualization](http://en.wikipedia.org/wiki/Data_visualization) of [data processing](http://en.wikipedia.org/wiki/Data_processing) (structured design).

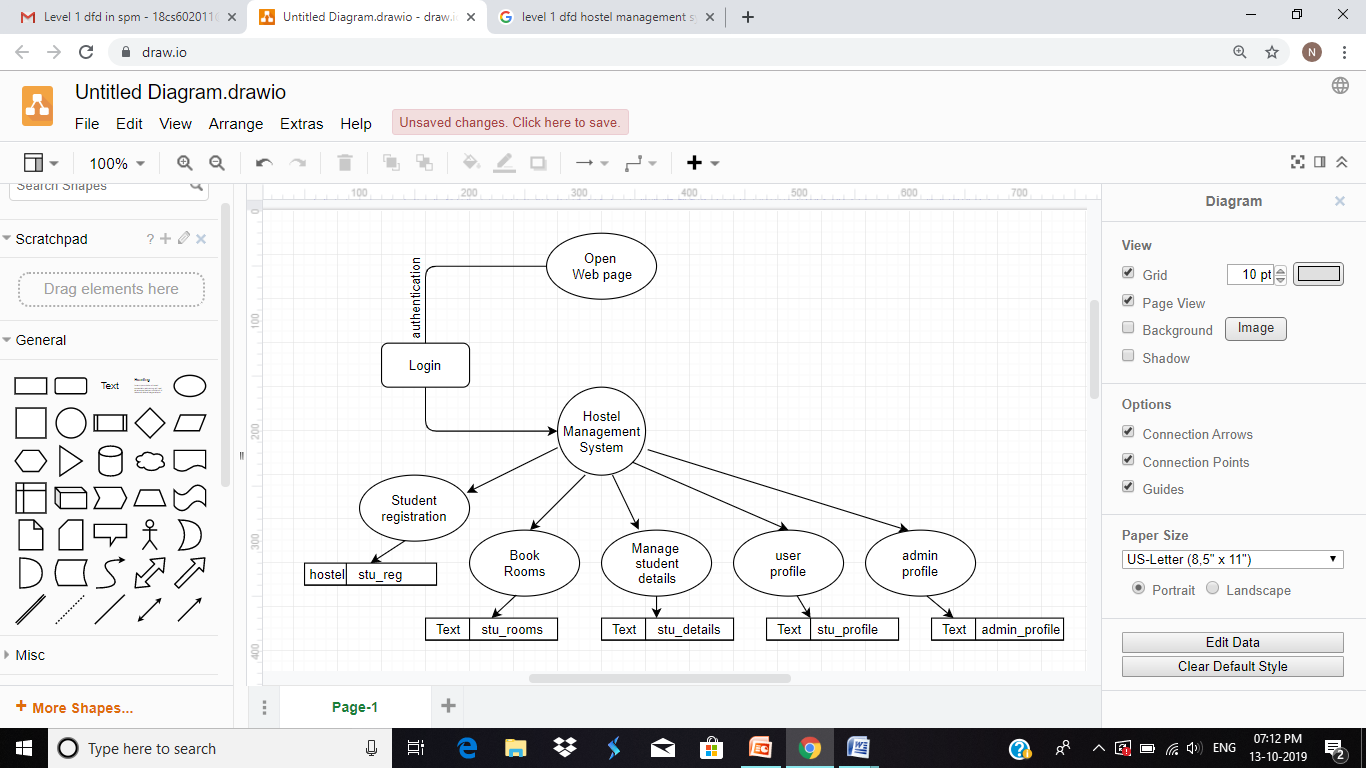
A DFD shows what kind of information will be input to and output from the system, where the data will come from and go to, and where the data will be stored. It does not show information about the timing of process or information about whether processes will operate in sequence or in parallel.

It is common practice to draw the [context-level data flow diagram](http://en.wikipedia.org/wiki/System_context_diagram) first, which shows the interaction between the system and external agents which act as data sources and data sinks. This helps to create an accurate drawing in the context diagram. The system's interactions with the outside world are modelled purely in terms of data flows across the system boundary. The context diagram shows the entire system as a single process, and gives no clues as to its internal organization.

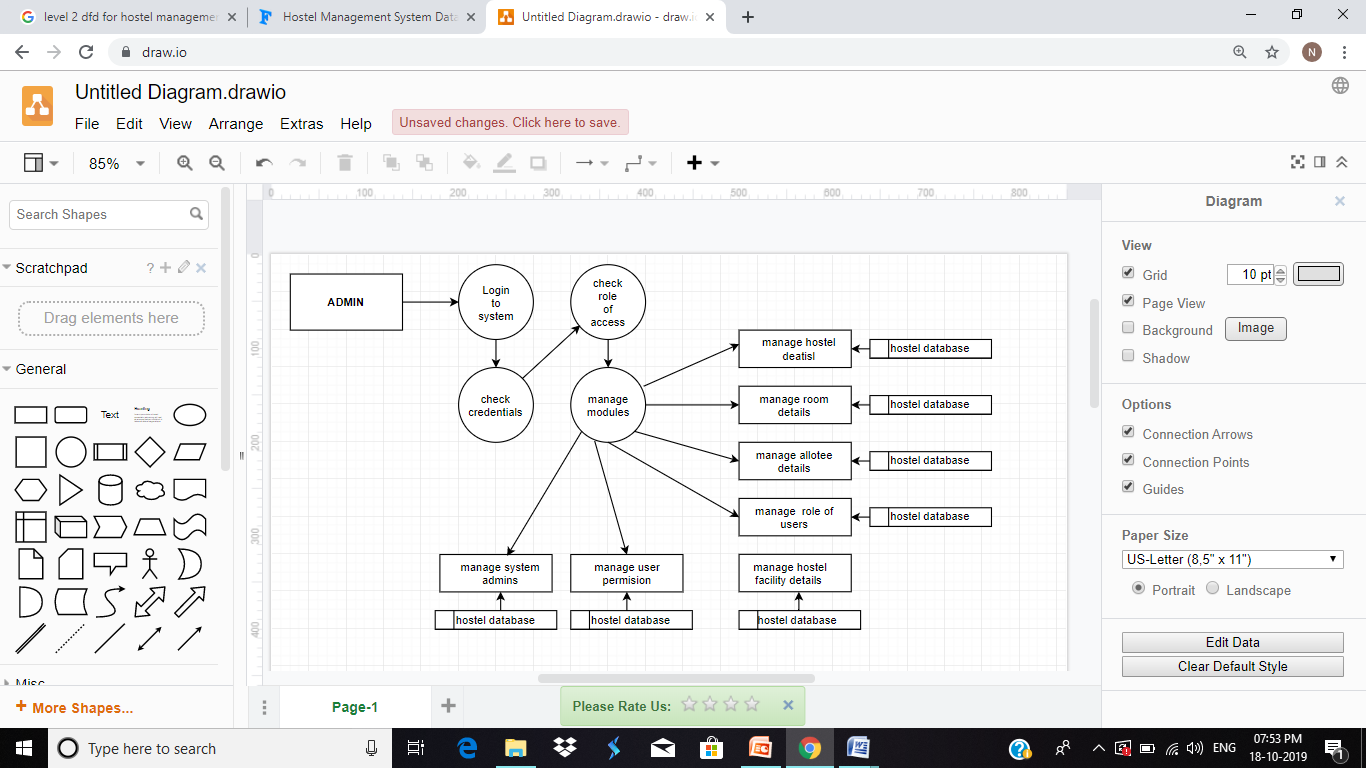
**Level 0 DFD :**

****

**LEVEL -1 DFD**

****

**LEVEL 2 DFD**

****

**3. SYSTEM REQUIREMENTS**

**3.1 Software Requirements:**

Framework : PHP 5.5

Front End : HTML, CSS, JAVASCRIPT

Database : XAMPP

Operating System : Windows 10

Servers : APACHE TOMCAT

**3.2 Hardware Requirements:**

Processor : Intel itanium i5

Processor Speed : 250 MHz to 833MHz

RAM : Min 1GB

Hard Disk : Min 40GB

**4. DETAILS OF SOFTWARE**

**4.1 Overview OF Front End**

**4.1.1 PHP**

The query cache plugin is implemented as a PHP extension. It is written in C and operates under the hood of PHP. During the startup of the PHP interpreter, it gets registered as a mysqlnd plugin to replace selected mysqlnd C methods. Hereby, it can change the behavior of any PHP MySQL extension (mysqli, PDO\_MYSQL, mysql) compiled to use the mysqlnd library without changing the extensions API. This makes the plugin compatible with each and every PHP MySQL application. Because existing APIs are not changed, it is almost transparent to use. Please, see the mysqlnd plugin API description for a discussion of the advantages of the plugin architecture and a comparison with proxy based solutions. At PHP run time PECL/ mysqlnd\_qc can proxy queries send from PHP to the MySQL server. It then inspects the statement string to find whether it shall cache its results. If so, result set is cached using a storage handler and further executions of the statement are served

What is PHP? PHP (Hypertext Pre-processor) is a widely-used open source generalpurpose scripting language that is especially suited for web development and can be embedded into HTML. The best things in using PHP are that it is extremely simple for a newcomer, but offers many advanced features for a professional programmer. Don't be afraid reading the long list of PHP's features. You can jump in, in a short time, and start writing simple scripts in a few hours. PHP is an acronym for "PHP: Hypertext Pre-processor". PHP is a widelyused, open source scripting language. PHP scripts are executed on the server. PHP is free to download and use PHP is a server-side scripting language designed primarily for web development but also used as a general-purpose programming language. Originally created by Rasmus Lerdorf in 1994, the PHP reference implementation is now produced by The PHP Development Team. PHP originally stood for Personal Home Page, but it now stands for the recursive acronym PHP: Hypertext Pre-processor. PHP code may be embedded into HTML or HTML5 mark-up, or it can be used in combination with various web template systems, web content management systems and web frameworks. PHP code is usually processed by a PHP interpreter implemented as a module in the web server or as a 15 Common Gateway Interface (CGI) executable. The web server software combines the results of the interpreted and executed PHP code, which may be any type of data, including images, with the generated web page. PHP code may also be executed with a command-line interface (CLI) and can be used to implement standalone graphical applications, What is a PHP File? PHP files can contain text, HTML, CSS, JavaScript, and PHP code. PHP code are executed on the server, and the result is returned to the browser as plain HTML. PHP files have extension ".php" What Can PHP Do? PHP can generate dynamic page content. PHP can create, open, read, write, delete, and close files on the server. PHP can collect form data. PHP can send and receive cookies. PHP can add, delete, modify data in your database. PHP can be used to control user-access. PHP can encrypt data. With PHP you are not limited to output HTML. You can output images, PDF files, and even flash movies. You can also output any text, such as XHTML and XML. Why PHP? PHP runs on various platforms (Windows, Linux, Unix, Mac OS X, etc.) PHP is compatible with almost all servers used today (Apache, IIS, etc.) PHP supports a wide range of databases. PHP is free. Download it from the official PHP resource: www.php.net. PHP is easy to learn and runs efficiently on the server side

**4.1.2 HYPER TEXT MARKUP LANGUAGE**

**HTML** is a cornerstone technology used to create web pages as well as to create user interfaces for mobile and web applications. Web browsers can read HTML files and render them into visible and audible web pages. By adding tags to the page, you give instructions to the web browser about how you want the page to display.

The Hyper Text in Hyper Text Markup Language stands for its capability to link a text of graphic HOTSPOT to other web pages. This powerful feature, the Hyperlink, is one of the main reasons why HTML was adopted as the language for the web.HTML is maintained by a standard body called the World Wide Web consortium (W3C).

The HTML language is a collection of tags that influence the display and formatting of a web page’s content. At a minimum, HTML document should contain basic sets of tags that identify it as a web page and define a simple structure for the page. The HTML tags <HTML></HTML> are the first and last tags in your page.

Everything else on the page goes between these two tags. These 2 tags tell the browser, or other program reading the page, that the file is a web page.

It provides a means to create structured documents by denoting structural semantics for text such as headings, paragraphs, lists, links, quotes and other items. HTML elements are delineated by tags, written using angle brackets.

**4.1.3 CASCADING STYLE SHEETS**

**CSS** is a [style sheet language](http://en.wikipedia.org/wiki/Style_sheet_language) used for describing the [look and formatting](http://en.wikipedia.org/wiki/Presentation_semantics) of a document written in a [markup language](http://en.wikipedia.org/wiki/Markup_language). Almost most often used to set the visual style of [web pages](http://en.wikipedia.org/wiki/Web_page) and user [interfaces](http://en.wikipedia.org/wiki/Interface_(computing)) written in [HTML](http://en.wikipedia.org/wiki/HTML) and [XHTML](http://en.wikipedia.org/wiki/XHTML), the language can be applied to any [XML](http://en.wikipedia.org/wiki/XML) document, including [plain XML](http://en.wikipedia.org/wiki/Plain_Old_XML), [SVG](http://en.wikipedia.org/wiki/Scalable_Vector_Graphics) and [XUL](http://en.wikipedia.org/wiki/XUL).

CSS is designed primarily to enable the separation of document content from document presentation, including aspects such as the [layout](http://en.wikipedia.org/wiki/Page_layout), [colors](http://en.wikipedia.org/wiki/Color), and [fonts](http://en.wikipedia.org/wiki/Typeface). This separation can improve content [accessibility](http://en.wikipedia.org/wiki/Accessibility), provide more flexibility and control in the specification of presentation characteristics, enable multiple HTML pages to share formatting, and reduce complexity and repetition in the structural content.

This separation of formatting and content makes it possible to present the same markup page in different styles for different rendering methods, such as on-screen, in-print, by voice and on braille-based, tactile devices. It can also be used to display the web page differently depending on the screen size or device on which it is being viewed.

Changes to the graphic design of a document can be applied quickly and easily, by editing a few lines in the CSS file they use, rather than by changing markup in the documents. The CSS specification describes a priority scheme to determine which style rules apply if more than one rule matches against a particular element.

**4.1.4 ADOBE DREAMWEAVER :**

**Adobe Dreamweaver** is a [proprietary](http://en.wikipedia.org/wiki/Proprietary_software) [web development](http://en.wikipedia.org/wiki/Web_development) tool developed by [Adobe Systems](http://en.wikipedia.org/wiki/Adobe_Systems). Dreamweaver was originally developed by [Macromedia](http://en.wikipedia.org/wiki/Macromedia) in 1997, and was maintained by them until Macromedia was acquired by Adobe Systems in 2005.Adobe Dreamweaver is available for both [OS X](http://en.wikipedia.org/wiki/OS_X) and [Windows](http://en.wikipedia.org/wiki/Windows).

Following Adobe's acquisition of the Macromedia product suite, releases of Dreamweaver subsequent to version 8.0 have been more compliant with [W3C](http://en.wikipedia.org/wiki/World_Wide_Web_Consortium) standards. Recent versions have improved support for [Web](http://en.wikipedia.org/wiki/World_Wide_Web) technologies such as [CSS](http://en.wikipedia.org/wiki/Cascading_Style_Sheets), [JavaScript](http://en.wikipedia.org/wiki/JavaScript), and various [server-side scripting](http://en.wikipedia.org/wiki/Server-side_scripting)[languages](http://en.wikipedia.org/wiki/Programming_language) and [frameworks](http://en.wikipedia.org/wiki/Software_framework) including [ASP](http://en.wikipedia.org/wiki/Active_Server_Pages) (ASP JavaScript, ASP VBScript, ASP.NET, C#, ASP.NET VB), [ColdFusion](http://en.wikipedia.org/wiki/ColdFusion), [Scriptlet](http://en.wikipedia.org/wiki/Scriptlet), and [PHP](http://en.wikipedia.org/wiki/PHP).

Adobe Dreamweaver is a web design and development application that combines a visual design surface known as Live View and a code editor with standard features such as syntax highlighting, code completion and code collapsing as well as more sophisticated features such as real-time syntax checking and code introspection for generating code hints to assist the user in writing code.

**4.2 OVERVIEW OF BACK END**

**4.2.1 APACHE TOMCAT :**

Inorder to develop JSP pages on your own computer there must be a JSP-enabled server installed. The standard reference server for JSP is the Tomcat server which is part of the Jakarta project. Apache Tomcat is an open source software implementation of the Java Servlet and Java Server Pages technologies. The Java Servlet and Java Server Pages specifications are developed under the Java Community Process.

Apache Tomcat is developed in an open and participatory environment and released under the Apache Software License. Apache Tomcat is intended to be a collaboration of the best-of-breed developers from around the world. Java Server Pages (JSP) technology enables Web developers and designers to rapidly develop and easily maintain, information-rich, dynamic Web pages that leverage existing business systems.

**Apache Tomcat 8.0.3.0** implements the servlet 3.1 and Java Server Pages 2.3 specifications from the Java Community Process and includes many additional features that make it a useful platform for developing and deploying web applications and web services. The notable changes include correct a false positive warning for Thread Local related memory leaks; update the packaged version of the Tomcat Native Library.

**4.2.2 XAMP**

It is a [free and open-source](https://en.wikipedia.org/wiki/Free_and_open-source) [cross-platform](https://en.wikipedia.org/wiki/Cross-platform) [web server](https://en.wikipedia.org/wiki/Web_server) [solution stack](https://en.wikipedia.org/wiki/Solution_stack) package developed by Apache Friends, consisting mainly of the [Apache HTTP Server](https://en.wikipedia.org/wiki/Apache_HTTP_Server), [MariaDB](https://en.wikipedia.org/wiki/MariaDB" \o "MariaDB) [database](https://en.wikipedia.org/wiki/Database), and [interpreters](https://en.wikipedia.org/wiki/Interpreter_(computing)) for scripts written in the [PHP](https://en.wikipedia.org/wiki/PHP) and [Perl](https://en.wikipedia.org/wiki/Perl) [programming languages](https://en.wikipedia.org/wiki/Programming_language). Since most actual web server deployments use the same components as XAMPP, it makes transitioning from a local test server to a live server possible.

XAMPP's ease of deployment means a [WAMP](https://en.wikipedia.org/wiki/WAMP) or [LAMP](https://en.wikipedia.org/wiki/LAMP_(software_bundle)) stack can be installed quickly and simply on an operating system by a developer. With the advantage a number of common add-in applications such as [WordPress](https://en.wikipedia.org/wiki/WordPress" \o "WordPress) and [Joomla!](https://en.wikipedia.org/wiki/Joomla!" \o "Joomla!) can also be installed with similar ease using [Bitnami](https://en.wikipedia.org/wiki/Bitnami" \o "Bitnami).

XAMPP is regularly updated to the latest releases of [Apache](https://en.wikipedia.org/wiki/Apache_HTTP_Server), [MariaDB](https://en.wikipedia.org/wiki/MariaDB" \o "MariaDB), [PHP](https://en.wikipedia.org/wiki/PHP) and [Perl](https://en.wikipedia.org/wiki/Perl). It also comes with a number of other modules including [OpenSSL](https://en.wikipedia.org/wiki/OpenSSL" \o "OpenSSL), [phpMyAdmin](https://en.wikipedia.org/wiki/PhpMyAdmin" \o "PhpMyAdmin), [MediaWiki](https://en.wikipedia.org/wiki/MediaWiki" \o "MediaWiki), [Joomla](https://en.wikipedia.org/wiki/Joomla" \o "Joomla), [WordPress](https://en.wikipedia.org/wiki/WordPress" \o "WordPress) and more. Self-contained, multiple instances of XAMPP can exist on a single computer, and any given instance can be copied from one computer to another. XAMPP is offered in both a full and a standard version (Smaller version).

The most obvious characteristic of XAMPP is the ease at which a [WAMP](https://en.wikipedia.org/wiki/WAMP) webserver stack can be deployed and instantiated. Later some common packaged applications that could be easily installed were provided by [Bitnami](https://en.wikipedia.org/wiki/Bitnami" \o "Bitnami).

Officially, XAMPP's designers intended it for use only as a development tool, to allow website designers and programmers to test their work on their own computers without any access to the Internet. To make this as easy as possible, many important security features are disabled by default. XAMPP has the ability to serve web pages on the [World Wide Web](https://en.wikipedia.org/wiki/World_Wide_Web). A special tool is provided to [password-protect](https://en.wikipedia.org/wiki/Password) the most important parts of the package.

XAMPP also provides support for creating and manipulating databases in [MariaDB](https://en.wikipedia.org/wiki/MariaDB" \o "MariaDB) and [SQLite](https://en.wikipedia.org/wiki/SQLite" \o "SQLite) among others.

Once XAMPP is installed, it is possible to treat a [localhost](https://en.wikipedia.org/wiki/Localhost" \o "Localhost) like a remote host by connecting using an [FTP](https://en.wikipedia.org/wiki/File_Transfer_Protocol) client. Using a program like [FileZilla](https://en.wikipedia.org/wiki/FileZilla" \o "FileZilla) has many advantages when installing a [content management system](https://en.wikipedia.org/wiki/Content_management_system) (CMS) like [Joomla](https://en.wikipedia.org/wiki/Joomla" \o "Joomla) or [WordPress](https://en.wikipedia.org/wiki/WordPress" \o "WordPress)It is also possible to connect to localhost via FTP with an [HTML editor](https://en.wikipedia.org/wiki/HTML_editor).

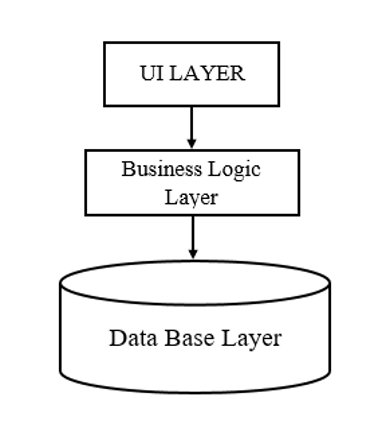
**5. SYSTEM DESIGN**

**5.1 ARCHITECTURAL DESIGN :**

The architectural design of a system emphasizes the design of the systems architecture that describes the structure, behavior and more views of that system and analysis. An idea that takes the parts of a structure and turns them into a whole system. This is done with objectives and limitations. Concept that focuses on the components or elements of a structure or system and unifies them into a coherent and functional whole, according to a particular approach in achieving the objective(s) under the given constraints or limitations. See also behavioral design.

Architecture is a team-working process and rarely a lone activity. There is always a client and there is always an interpreter of that client's needs. The relationship between client and architect is fundamental, and the establishment of a professional and trusting relationship between the two is the bedrock of every successful project.

Creating architecture involves art and beauty, science and engineering, values and beliefs, friendship and team-working. It is one of life's rewarding activities, bringing together a wide range of personalities, skills and expertise. It is an adventure for the client, the architect and their team.

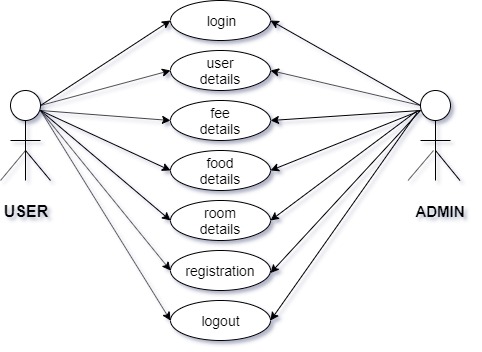


**UI LAYER:** In this layer the UI developer creates a UI for the users to use it, this basically have Website page with menu icons, images, buttons where user can click on each item to see the events. The goal of user interface design is to produce a user interface which makes it easy (self-explanatory), efficient, and enjoyable (user-friendly) to operate a machine in the way which produces the desired result. This generally means that the operator needs to provide minimal input to achieve the desired output, and also that the machine minimizes undesired outputs to the human. UI in Furnish are Home Page, Login Page, Registration Page, Cart Page, Shipping Details, Payment and Orders History.

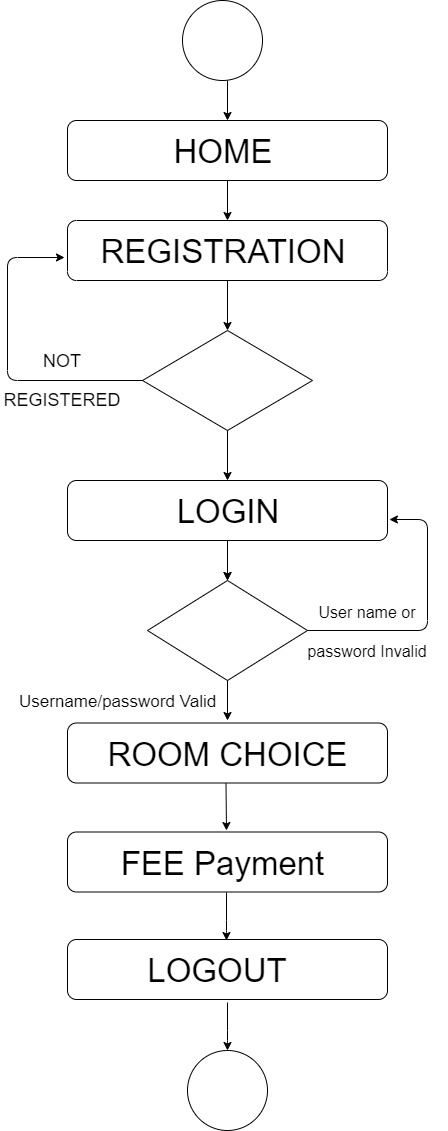
**BUSINESS LOGIC LAYER:** Business logic is the part of the program that encodes the real-world business rules that determine how data can be created, displayed, stored, and changed. It is contrasted with the remainder of the software that might be concerned with lower-level details of managing a database or displaying the user interface, system infrastructure, or generally connecting various parts of the program. Furnish CRUD- Create, Read, Update and Delete operations are done in this layer

**DATA LAYER:** The data layer, which sits in the middle, transfers visitor interaction data occurring at the experience layer to vendors at the application layer.

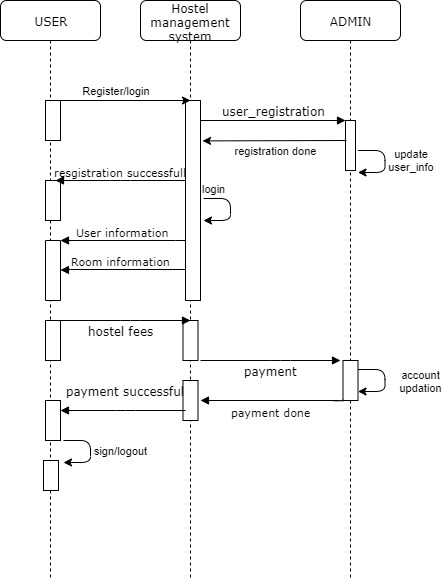
**5.2 USE CASE DIAGRAM**



**5.3 ACTIVITY DIAGRAM**



**5.4 SEQUENCE DIAGRAM**



**5.5 INPUT DESIGN**

The input data are collected and organized to make data entry easy, logical, and error free. Each area in the input form should be identified should be specified for the user what to write and where to write.

A screen is an actually a display station that has a buffer for storing data. The main objective of screen design is for simplicity, accurate and quick data capture or entry.

The objective in the input design is to ensure that the data which will be processed by the system is collected and inserted into the system efficiently according to the specified requirements, and with the minimum errors.

The basic design consideration that would satisfy the user requirements were as follows.

**Guidelines are :**

Use the same format throughout the project.

Allow ample space to avoid data over crowding because it cause eyestrain and may reduce the interest of the user.

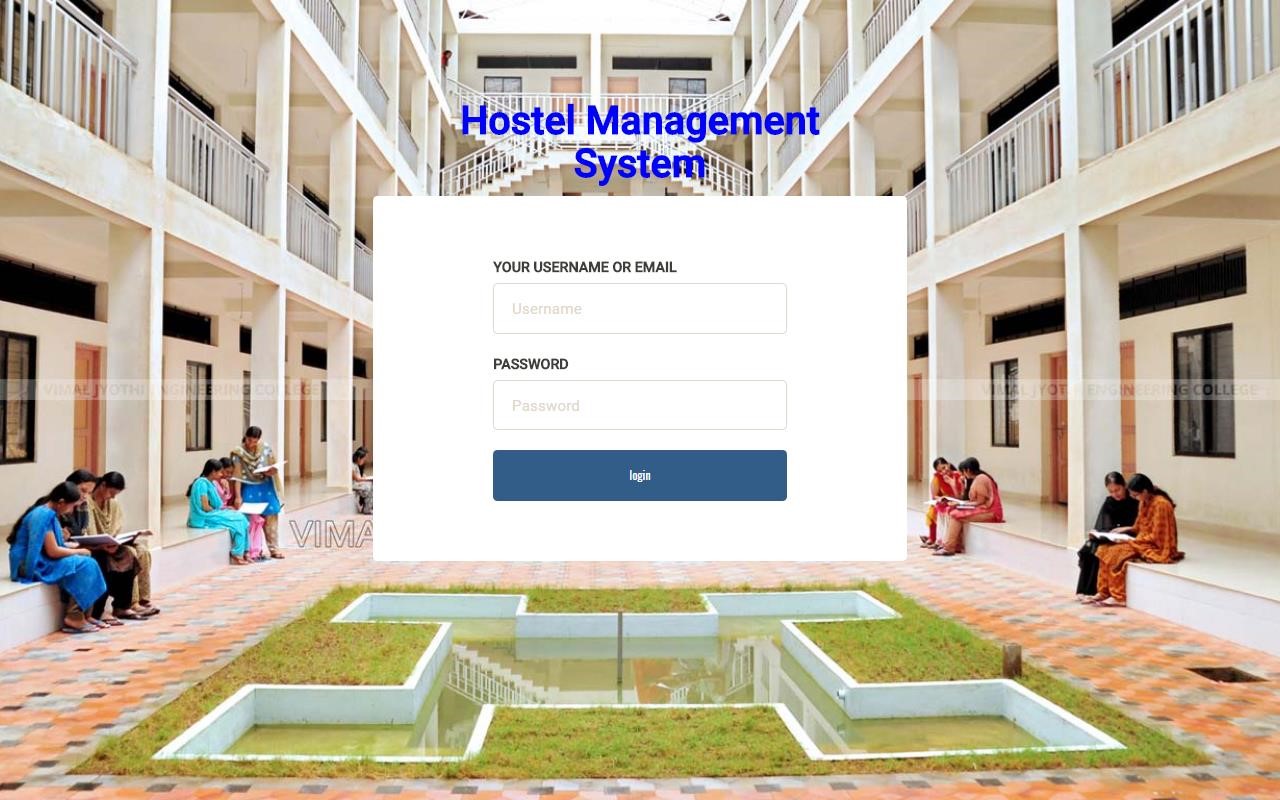
Use easy tolerant and consistent term such as “add”, “delete” and “close”.

**INPUT DESIGN IN THE SYSTEM**

Input design is used to provide a user-friendly interface to enter data. Input design is the processes of converting user originate inputs to a computer-based format. Inputs are important because in many instances they are the only contact a user has with a system. This project has following input form in which textboxes are used for value input:

**Input Screenshot :**

**ADMIN LOGIN**

****

**5.6 OUTPUT DESIGN :**

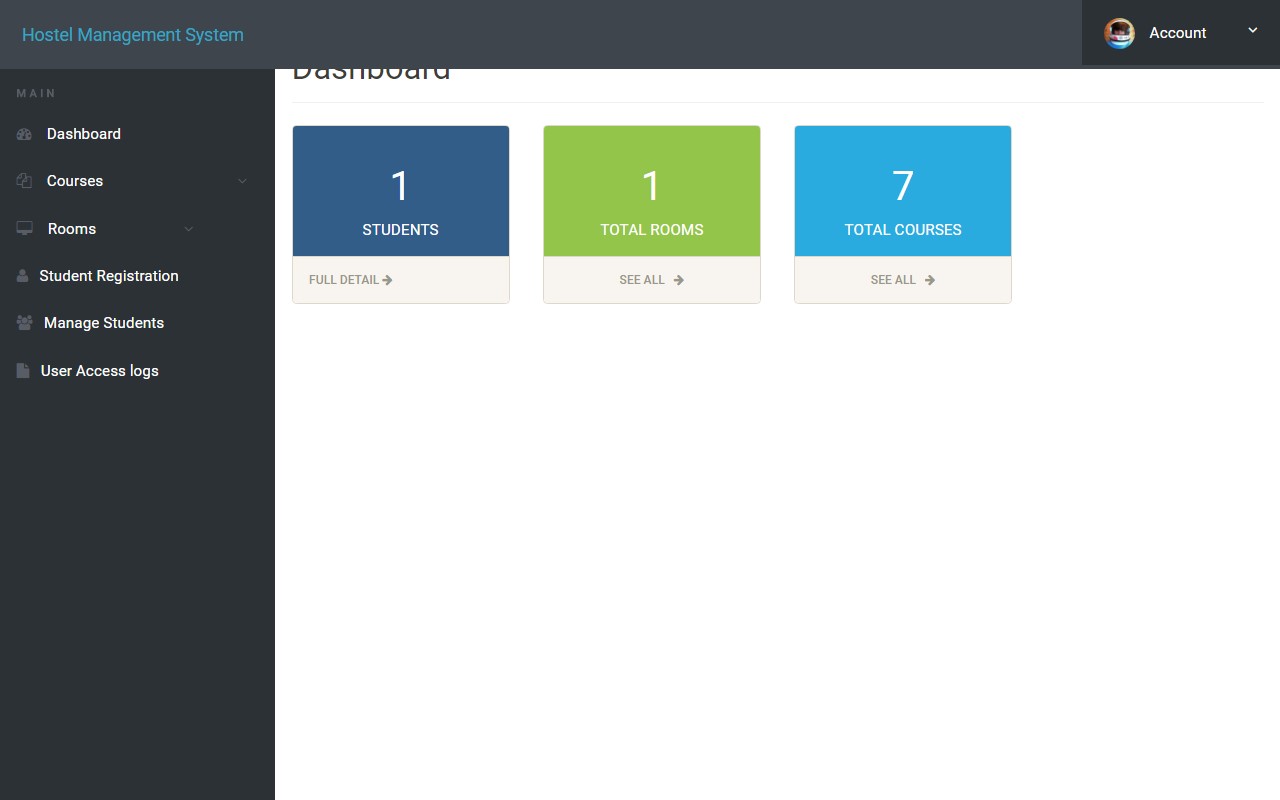
Computer output is the most important and direct information source to the user. Output design is a process that involves designing necessary outputs in the form of reports that should be given to the users according to their requirements. The reports generated should be simple, descriptive and clear to the user. So, while designing output, some points such as determining what information to present, arranging the presentation of information in an acceptable format, etc. should be considered.

The output design is an ongoing activity almost from the beginning of the project and follows the principles of form design. Efficient, intelligible and well-defined output design improves the relationship of the system and the user, thus facilitating decision-making.

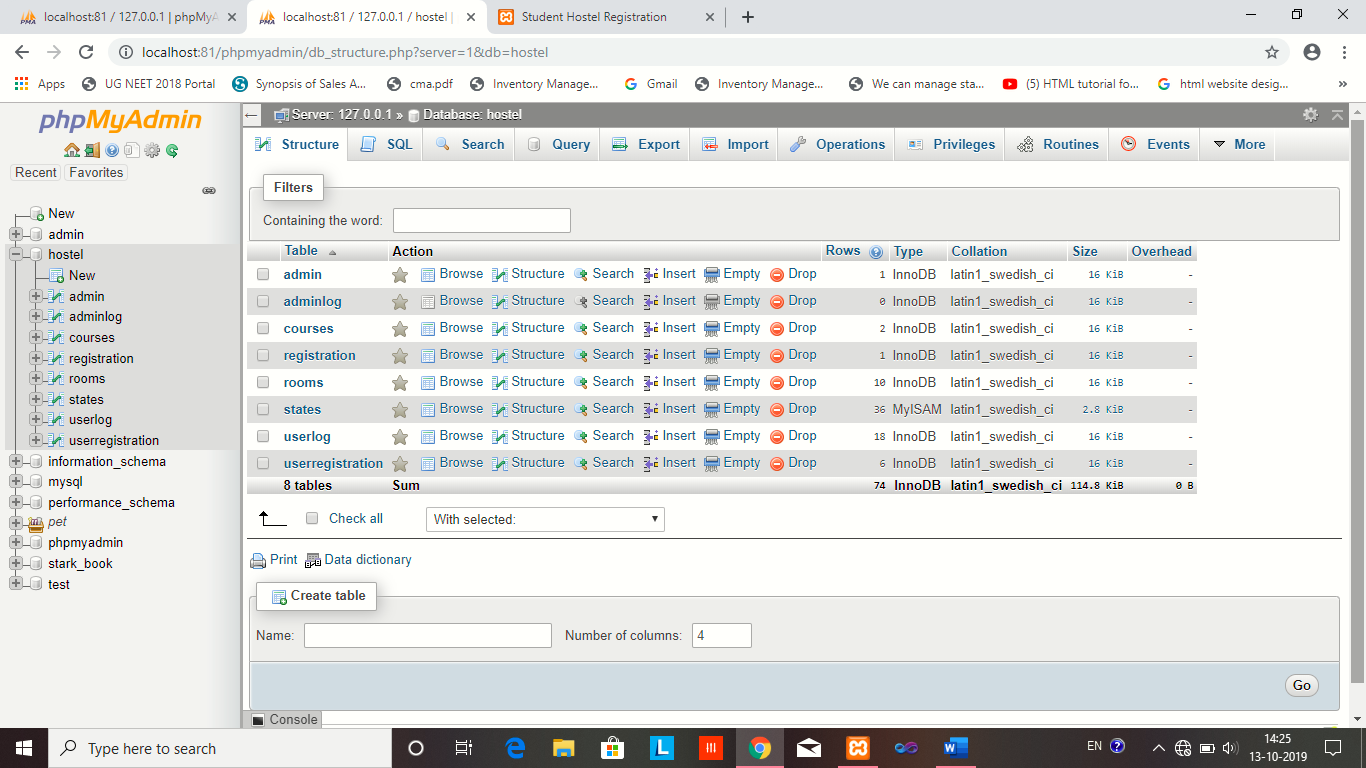
The objective of output design is to define the format of all printed documents and reports and of the screens that will be produced by the system. Computer output is the most important and direct source of information to the user. A major form of output, reports is the hardcopy from the printer. Printouts should be designed around the output requirements of the user. The proposed system has the capability to generate reports on the particular system for a particular period.

**Output Screenshot :**

**ADMIN DASHBOARD**

****

**5.7 DATABASE DESIGN**

**5.7.1 Hostel Database**

**6. SOURCE CODE**

**REGISTRATION PAGE**

<?php

session\_start();

include('includes/config.php');

if(isset($\_POST['submit']))

{

$regno=$\_POST['regno'];

$fname=$\_POST['fname'];

$mname=$\_POST['mname'];

$lname=$\_POST['lname'];

$gender=$\_POST['gender'];

$contactno=$\_POST['contact'];

$emailid=$\_POST['email'];

$password=$\_POST['password'];

$query="insert into userRegistration(regNo,firstName,middleName,lastName,gender,contactNo,email,password) values(?,?,?,?,?,?,?,?)";

$stmt = $mysqli->prepare($query);

$rc=$stmt->bind\_param('sssssiss',$regno,$fname,$mname,$lname,$gender,$contactno,$emailid,$password);

$stmt->execute();

echo"<script>alert('Student Successfully register');</script>";

}

?>

<!DOCTYPE html>

<html>

<head>

<meta charset="UTF-8">

<title>User Login</title>

<link rel="stylesheet" href="css1/style.css" type="text/css">

<link rel="stylesheet" href="css/font-awesome.min.css">

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

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

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

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

<link rel="stylesheet" href="css/fileinput.min.css">

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

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

<script type="text/javascript" src="js/jquery-1.11.3-jquery.min.js"></script>

<script type="text/javascript" src="js/validation.min.js"></script>

<script type="text/javascript" src="http://code.jquery.com/jquery.min.js"></script>

<script type="text/javascript">

function valid()

{

if(document.registration.password.value!= document.registration.cpassword.value)

{

alert("Password and Re-Type Password Field do not match !!");

document.registration.cpassword.focus();

return false;

}

return true;

}

</script>

</head>

<body>

<div id="background">

<div id="page">

<div id="header">

<div id="logo">

<a href="index.html" style="color:white; font-size:20px;">NATIONAL HOSTEL MANAGEMENT</a>

</div>

<div id="navigation">

<ul>

<li >

<a href="index.html">Home</a>

</li>

<li>

<a href="rooms.html">Rooms</a>

</li>

<li>

<a href="foods.html">Food</a>

</li>

<li>

<a href="login.php">Login</a>

</li>

</ul>

</div>

</div>

<div id="contents">

<div>

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

<h2 class="page-title" style="color:white">Student Registration </h2>

<div class="row">

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

<div class="panel panel-primary">

<div class="panel-heading">Fill all Info</div>

<div class="panel-body">

<form method="post" action="" name="registration" class="form-horizontal" onSubmit="return valid();">

<div class="form-group">

<label class="col-sm-2 control-label"> Registration No : </label>

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

<input type="text" name="regno" id="regno" class="form-control" required="required" >

</div>

</div>

<div class="form-group">

<label class="col-sm-2 control-label">First Name : </label>

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

<input type="text" name="fname" id="fname" class="form-control" required="required" >

</div>

</div>

<div class="form-group">

<label class="col-sm-2 control-label">Middle Name : </label>

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

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

</div>

</div>

<div class="form-group">

<label class="col-sm-2 control-label">Last Name : </label>

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

<input type="text" name="lname" id="lname" class="form-control" required="required">

</div>

</div>

<div class="form-group">

<label class="col-sm-2 control-label">Gender : </label>

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

<select name="gender" class="form-control" required="required">

<option value="">Select Gender</option>

<option value="male">Male</option>

<option value="female">Female</option>

<option value="others">Others</option>

</select>

</div>

</div>

<div class="form-group">

<label class="col-sm-2 control-label">Contact No : </label>

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

<input type="text" name="contact" id="contact" class="form-control" required="required">

</div>

</div>

<div class="form-group">

<label class="col-sm-2 control-label">Email id: </label>

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

<input type="email" name="email" id="email" class="form-control" onBlur="checkAvailability()" required="required">

<span id="user-availability-status" style="font-size:12px;"></span>

</div>

</div>

<div class="form-group">

<label class="col-sm-2 control-label">Password: </label>

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

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

</div>

</div>

<div class="form-group">

<label class="col-sm-2 control-label">Confirm Password : </label>

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

<input type="password" name="cpassword" id="cpassword" class="form-control" required="required">

</div>

</div>

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

<button class="btn btn-default" type="submit">Cancel</button>

<input type="submit" name="submit" Value="Register" class="btn btn-primary">

</div>

</form>

</div>

</div>

**MY PROFILE**

<?php

session\_start();

include('includes/config.php');

date\_default\_timezone\_set('Asia/Kolkata');

include('includes/checklogin.php');

check\_login();

$aid=$\_SESSION['id'];

if(isset($\_POST['update']))

{

$regno=$\_POST['regno'];

$fname=$\_POST['fname'];

$mname=$\_POST['mname'];

$lname=$\_POST['lname'];

$gender=$\_POST['gender'];

$contactno=$\_POST['contact'];

$udate = date('d-m-Y h:i:s', time());

$query="update userRegistration set regNo=?,firstName=?,middleName=?,lastName=?,gender=?,contactNo=?,updationDate=? where id=?";

$stmt = $mysqli->prepare($query);

$rc=$stmt->bind\_param('sssssisi',$regno,$fname,$mname,$lname,$gender,$contactno,$udate,$aid);

$stmt->execute();

echo"<script>alert('Profile updated Succssfully');</script>";

}

?>

<!doctype html>

<html lang="en" class="no-js">

<head>

<meta charset="UTF-8">

<meta http-equiv="X-UA-Compatible" content="IE=edge">

<meta name="viewport" content="width=device-width, initial-scale=1, minimum-scale=1, maximum-scale=1">

<meta name="description" content="">

<meta name="author" content="">

<meta name="theme-color" content="#3e454c">

<title>Profile Updation</title>

<link rel="stylesheet" href="css/font-awesome.min.css">

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

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

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

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

<link rel="stylesheet" href="css/fileinput.min.css">

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

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

<script type="text/javascript" src="js/jquery-1.11.3-jquery.min.js"></script>

<script type="text/javascript" src="js/validation.min.js"></script>

<script type="text/javascript" src="http://code.jquery.com/jquery.min.js"></script>

<script type="text/javascript">

function valid()

{

if(document.registration.password.value!= document.registration.cpassword.value)

{

alert("Password and Re-Type Password Field do not match !!");

document.registration.cpassword.focus();

return false;

}

return true;

}

</script>

</head>

<body>

<?php include('includes/header.php');?>

<div class="ts-main-content">

<?php include('includes/sidebar.php');?>

<div class="content-wrapper">

<div class="container-fluid">

<?php

$aid=$\_SESSION['id'];

$ret="select \* from userregistration where id=?";

$stmt= $mysqli->prepare($ret) ;

$stmt->bind\_param('i',$aid);

$stmt->execute() ;//ok

$res=$stmt->get\_result();

//$cnt=1;

while($row=$res->fetch\_object())

{

?>

<div class="row">

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

<h2 class="page-title"><?php echo $row->firstName;?>'s&nbsp;Profile </h2>

<div class="row">

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

<div class="panel panel-primary">

<div class="panel-heading">

Last Updation date : &nbsp; <?php echo $row->updationDate;?>

</div>

<div class="panel-body">

<form method="post" action="" name="registration" class="form-horizontal" onSubmit="return valid();">

<div class="form-group">

<label class="col-sm-2 control-label"> Registration No : </label>

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

<input type="text" name="regno" id="regno" class="form-control" required="required" value="<?php echo $row->regNo;?>" >

</div>

</div>

<div class="form-group">

<label class="col-sm-2 control-label">First Name : </label>

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

<input type="text" name="fname" id="fname" class="form-control" value="<?php echo $row->firstName;?>" required="required" >

</div>

</div>

<div class="form-group">

<label class="col-sm-2 control-label">Middle Name : </label>

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

<input type="text" name="mname" id="mname" class="form-control" value="<?php echo $row->middleName;?>" >

</div>

</div>

<div class="form-group">

<label class="col-sm-2 control-label">Last Name : </label>

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

<input type="text" name="lname" id="lname" class="form-control" value="<?php echo $row->lastName;?>" required="required">

</div>

</div>

<div class="form-group">

<label class="col-sm-2 control-label">Gender : </label>

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

<select name="gender" class="form-control" required="required">

<option value="<?php echo $row->gender;?>"><?php echo $row->gender;?></option>

<option value="male">Male</option>

<option value="female">Female</option>

<option value="others">Others</option>

</select>

</div>

</div>

<div class="form-group">

<label class="col-sm-2 control-label">Contact No : </label>

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

<input type="text" name="contact" id="contact" class="form-control" maxlength="10" value="<?php echo $row->contactNo;?>" required="required">

</div>

</div>

<div class="form-group">

<label class="col-sm-2 control-label">Email id: </label>

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

<input type="email" name="email" id="email" class="form-control" value="<?php echo $row->email;?>" readonly>

<span id="user-availability-status" style="font-size:12px;"></span>

</div>

</div>

<?php } ?>

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

<input type="submit" name="update" Value="Update Profile" class="btn btn-primary">

</div>

</form>

</div>

</div>

</div>

</div>

</div>

</div>

</div>

</div>

</div>

</div>

</div>

</div>

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

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

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

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

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

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

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

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

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

</body>

<script type="text/javascript">

$(document).ready(function(){

$('input[type="checkbox"]').click(function(){

if($(this).prop("checked") == true){

$('#paddress').val( $('#address').val() );

$('#pcity').val( $('#city').val() );

$('#pstate').val( $('#state').val() );

$('#ppincode').val( $('#pincode').val() );

}

});

});

</script>

<script>

function checkAvailability() {

$("#loaderIcon").show();

jQuery.ajax({

url: "check\_availability.php",

data:'emailid='+$("#email").val(),

type: "POST",

success:function(data){

$("#user-availability-status").html(data);

$("#loaderIcon").hide();

},

error:function (){}

});

}

</script>

</html>

**ROOM DETAILS**

<?php

session\_start();

include('includes/config.php');

include('includes/checklogin.php');

check\_login();

?>

<!doctype html>

<html lang="en" class="no-js">

<head>

<meta charset="UTF-8">

<meta http-equiv="X-UA-Compatible" content="IE=edge">

<meta name="viewport" content="width=device-width, initial-scale=1, minimum-scale=1, maximum-scale=1">

<meta name="description" content="">

<meta name="author" content="">

<meta name="theme-color" content="#3e454c">

<title>Room Details</title>

<link rel="stylesheet" href="css/font-awesome.min.css">

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

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

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

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

<link rel="stylesheet" href="css/fileinput.min.css">

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

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

<script language="javascript" type="text/javascript">

var popUpWin=0;

function popUpWindow(URLStr, left, top, width, height)

{

if(popUpWin)

{

if(!popUpWin.closed) popUpWin.close();

}

popUpWin = open(URLStr,'popUpWin', 'toolbar=no,location=no,directories=no,status=no,menubar=no,scrollbars=yes,resizable=no,copyhistory=yes,width='+510+',height='+430+',left='+left+', top='+top+',screenX='+left+',screenY='+top+'');

}

</script>

</head>

<body>

<?php include('includes/header.php');?>

<div class="ts-main-content">

<?php include('includes/sidebar.php');?>

<div class="content-wrapper">

<div class="container-fluid">

<div class="row">

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

<h2 class="page-title">Rooms Details</h2>

<div class="panel panel-default">

<div class="panel-heading">All Room Details</div>

<div class="panel-body">

<table id="zctb" class="table table-bordered " cellspacing="0" width="100%">

<tbody>

<?php

$aid=$\_SESSION['login'];

$ret="select \* from registration where emailid=?";

$stmt= $mysqli->prepare($ret) ;

$stmt->bind\_param('s',$aid);

$stmt->execute() ;

$res=$stmt->get\_result();

$cnt=1;

while($row=$res->fetch\_object())

{

?>

<tr>

<td colspan="4"><h4>Room Realted Info</h4></td>

<td><a href="javascript:void(0);" onClick="popUpWindow('http://localhost/hostel/full-profile.php?id=<?php echo $row->emailid;?>');" title="View Full Details">Print Data</a></td>

</tr>

<tr>

<td colspan="6"><b>Reg no. :<?php echo $row->postingDate;?></b></td>

</tr>

<tr>

<td><b>Room no :</b></td>

<td><?php echo $row->roomno;?></td>

<td><b>Seater :</b></td>

<td><?php echo $row->seater;?></td>

<td><b>Fees PM :</b></td>

<td><?php echo $fpm=$row->feespm;?></td>

</tr>

<tr>

<td><b>Food Status:</b></td>

<td>

<?php if($row->foodstatus==0)

{

echo "Without Food";

}

else

{

echo "With Food";

}

;?></td>

<td><b>Stay From :</b></td>

<td><?php echo $row->stayfrom;?></td>

<td><b>Duration:</b></td>

<td><?php echo $dr=$row->duration;?> Months</td>

</tr>

<tr>

<td colspan="6"><b>Total Fee :

<?php if($row->foodstatus==1)

{

$fd=2000;

echo (($dr\*$fpm)+$fd);

}

else

{

echo $dr\*$fpm;

}

?></b></td>

</tr>

<tr>

<td colspan="6"><h4>Personal Info Info</h4></td>

</tr>

<tr>

<td><b>Reg No. :</b></td>

<td><?php echo $row->regno;?></td>

<td><b>Full Name :</b></td>

<td><?php echo $row->firstName;?><?php echo $row->middleName;?><?php echo $row->lastName;?></td>

<td><b>Email :</b></td>

<td><?php echo $row->emailid;?></td>

</tr>

<tr>

<td><b>Contact No. :</b></td>

<td><?php echo $row->contactno;?></td>

<td><b>Gender :</b></td>

<td><?php echo $row->gender;?></td>

<td><b>Course :</b></td>

<td><?php echo $row->course;?></td>

</tr>

<tr>

<td><b>Emergency Contact No. :</b></td>

<td><?php echo $row->egycontactno;?></td>

<td><b>Guardian Name :</b></td>

<td><?php echo $row->guardianName;?></td>

<td><b>Guardian Relation :</b></td>

<td><?php echo $row->guardianRelation;?></td>

</tr>

<tr>

<td><b>Guardian Contact No. :</b></td>

<td colspan="6"><?php echo $row->guardianContactno;?></td>

</tr>

<tr>

<td colspan="6"><h4>Addresses</h4></td>

</tr>

<tr>

<td><b>Correspondense Address</b></td>

<td colspan="2">

<?php echo $row->corresAddress;?><br />

<?php echo $row->corresCIty;?>, <?php echo $row->corresPincode;?><br />

<?php echo $row->corresState;?>

</td>

<td><b>Permanent Address</b></td>

<td colspan="2">

<?php echo $row->pmntAddress;?><br />

<?php echo $row->pmntCity;?>, <?php echo $row->pmntPincode;?><br />

<?php echo $row->pmnatetState;?>

</td>

</tr>

<?php

$cnt=$cnt+1;

} ?>

</tbody>

</table>

</div>

</div>

</div>

</div>

</div>

</div>

</div>

<!-- Loading Scripts -->

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

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

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

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

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

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

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

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

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

</body>

</html>

**7 .TESTING**

**7.1 TEST CASES**

The purpose of testing is to discover errors. Testing is the process of trying to discover every conceivable fault or weakness in a work product. It provides a way to check the functionality of components, sub assemblies, assemblies and/or a finished product It is the process of exercising software with the intent of ensuring that the Software system meets its requirements and user expectations and does not fail in an unacceptable manner. There are various types of test. Each test type addresses a specific testing requirement.

**White Box Testing:**

White Box Testing is a testing in which in which the software tester has knowledge of the inner workings, structure and language of the software, or at least its purpose. It is purpose. It is used to test areas that cannot be reached from a black box level.

**Black Box Testing:**

Black Box Testing is testing the software without any knowledge of the inner workings, structure or language of the module being tested. Black box tests, as most other kinds of tests, must be written from a definitive source document, such as specification or requirements document, such as specification or requirements document. It is a testing in which the software under test is treated, as a black box you cannot “see” into it. The test provides inputs and responds to outputs without considering how the software works.

**Unit Testing:**

Unit testing is usually conducted as part of a combined code and unit test phase of the software lifecycle, although it is not uncommon for coding and unit testing to be conducted as two distinct phases.

Field testing will be performed manually and functional tests will be written in detail.

Pages must be activated from the identified link.

The entry screen, messages and responses must not be delayed.

Verify that the entries are of the correct format

No duplicate entries should be allowed

All links should take the user to the correct page

**Integration Testing:**

Software integration testing is the incremental integration testing of two or more integrated software components on a single platform to produce failures caused by interface defects. The task of the integration test is to check that components or software applications, e.g. components in a software system or one step up software applications at the company level interact without error.

**Acceptance Testing:**

User Acceptance Testing is a critical phase of any project and requires significant participation by the end user. It also ensures that the system meets the functional requirements.

**8. IMPLEMENTATION:**

Implementation is the stage of the project when the theoretical design is turned out into a working system. Thus it can be considered to be the most critical stage in achieving a successful new system and in giving the user, confidence that the new system will work and be effective. The implementation stage involves careful planning, investigation of the existing system and it’s constraints on implementation, designing of methods to achieve changeover and evaluation of changeover methods.

There are several activities involved while implementing a new project they are

* + - End user training
    - End user Education
    - Training on the application software
    - System Design
    - Parallel Run and to New System

**End user Training**

The successful implementation of the new system will purely upon the involvement of the officers working in that department. The officers will be imparted the necessary training on the new technology.

**End User Education**

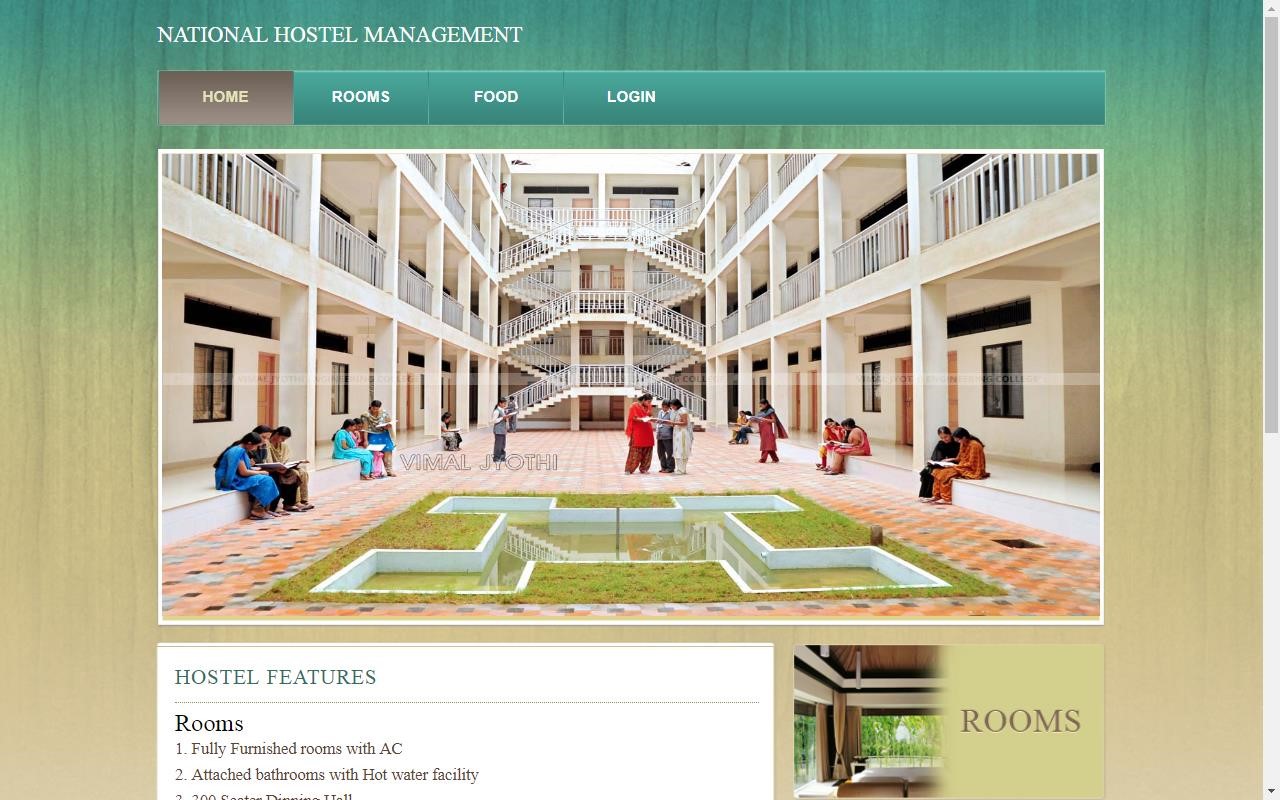
The education of the end user start after the implementation and testing is over. When the system is found to be more difficult to understand and complex, more effort is put to educate the end used to make them aware of the system, giving them lectures about the new system and providing them necessary documents and materials about how the system can do this.

**Training of application software**

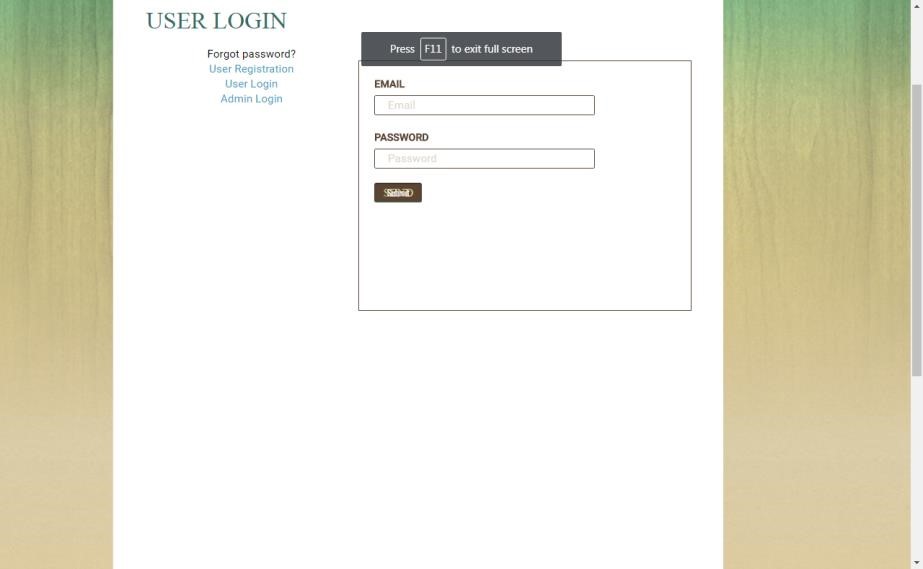
After providing the necessary basic training on the computer awareness, the users will have to be trained upon the new system such as the screen flows and screen design type of help on the screen, type of errors while entering the data, the corresponding validation check at each entry and the way to correct the data entered. It should then cover information needed by the specific user or group to use the system.

**9. SCREENSHOTS**

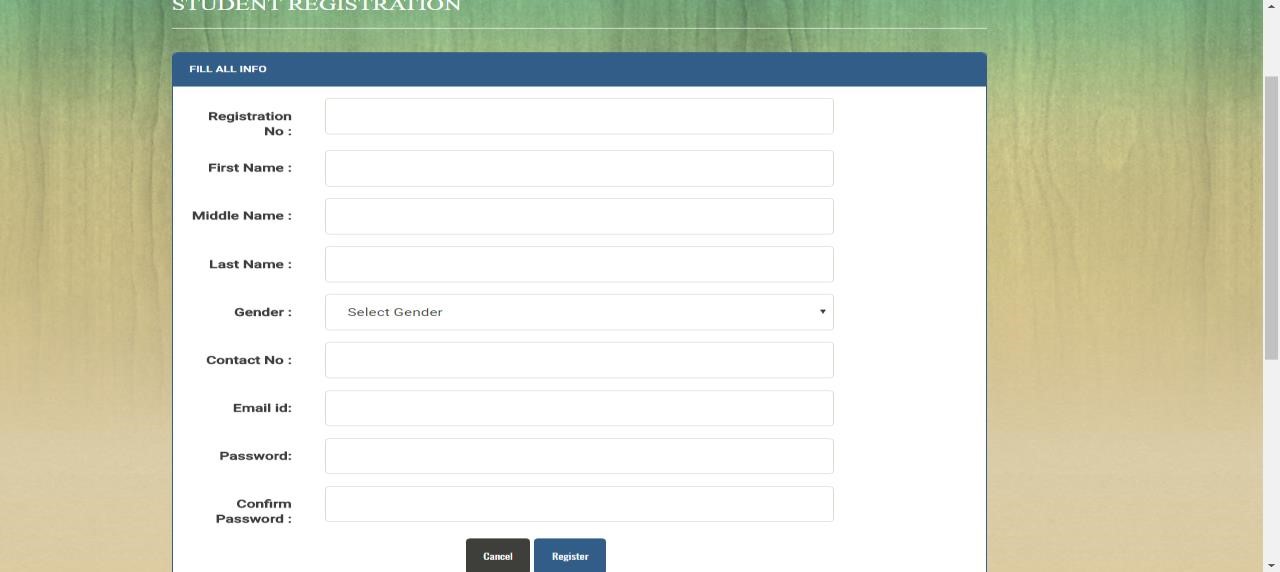
**HOME PAGE**

****

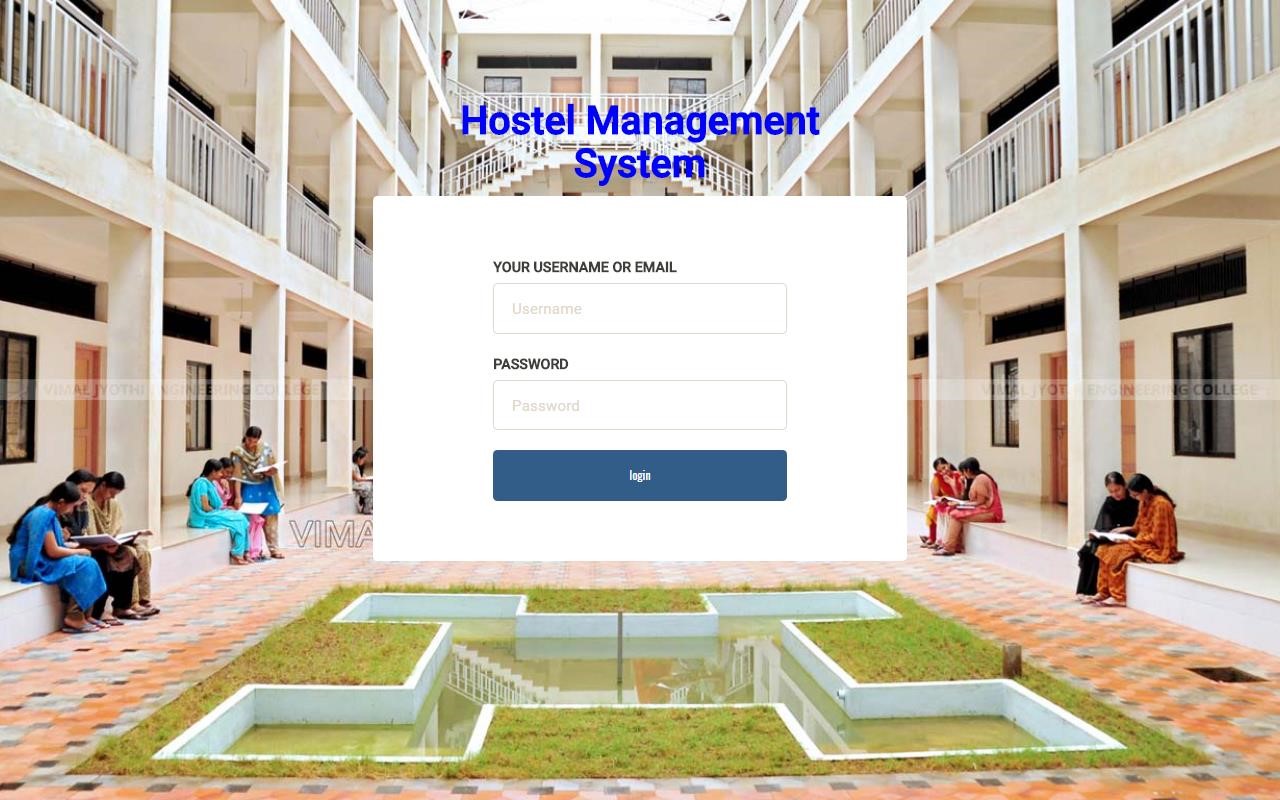
**REGISTRATION PAGE**

****

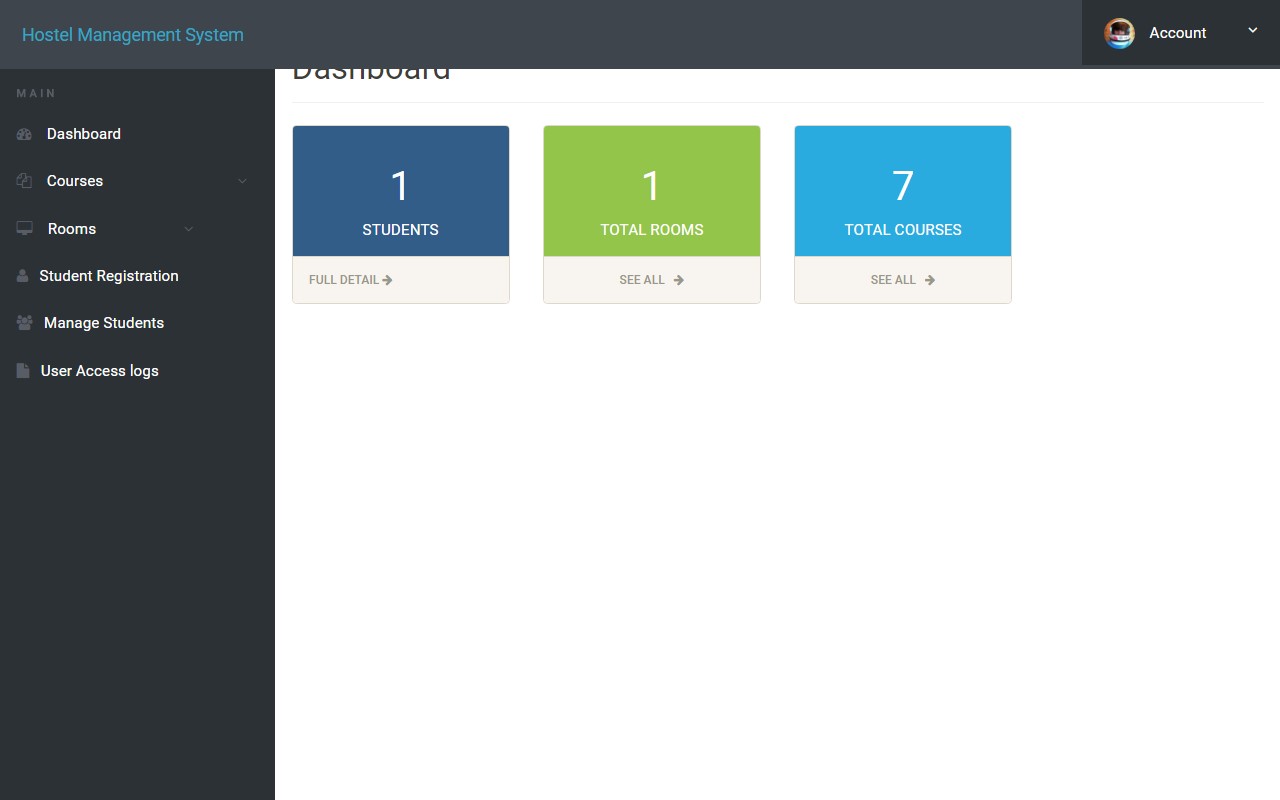
**STUDENT REGISTRATION**

****

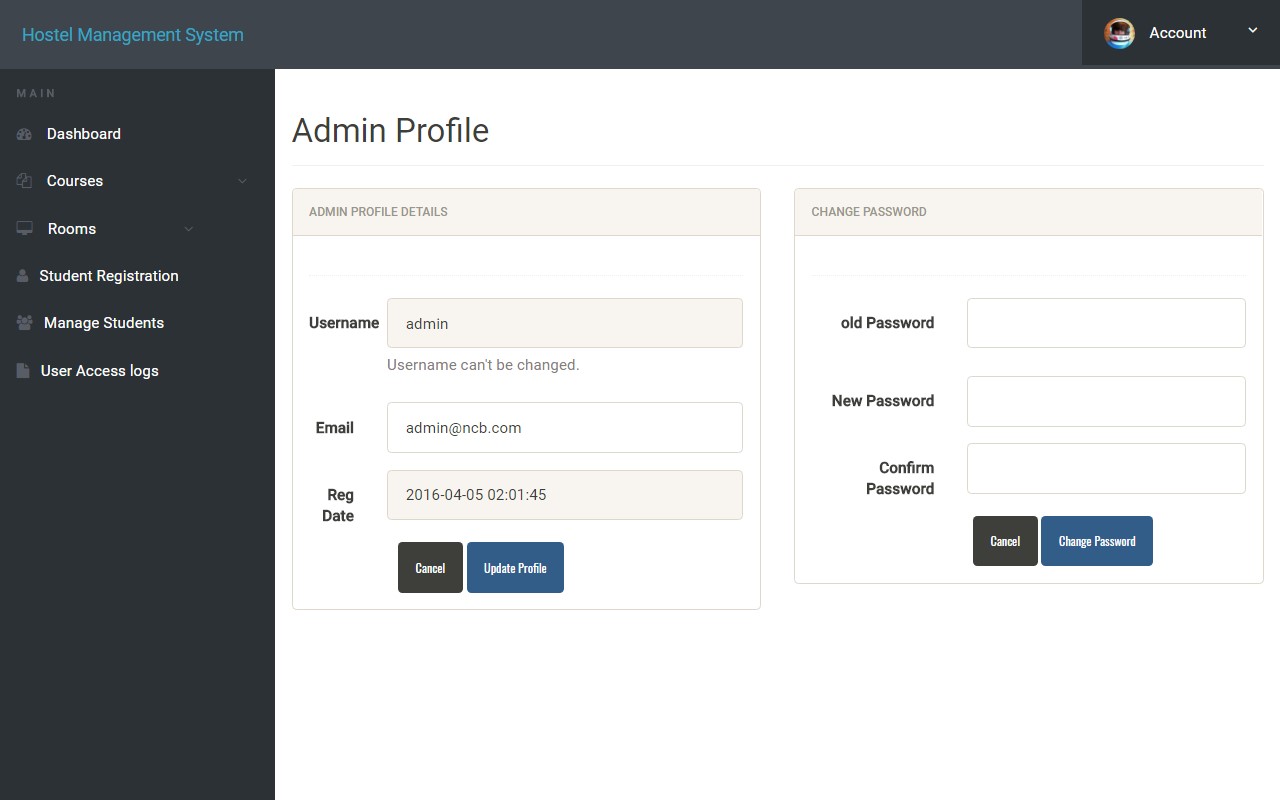
**ADMIN LOGIN PAGE**

****

**ADMIN DASH BOARD**

****

**ADMIN PROFILE**

****

## 

## **10. CONCLUSION**

**HOSTEL MANAGEMENT SYSTEM**  enhances the overall performance of the system. It removes the necessity of maintaining all the registers in turn helps to remove all kinds of manual errors. Work on employees will be very minimal.

So we can tell that proposed system is more efficient & convenient than existing system.

The objective of software planning is to provide a frame work that enables the manager to make reasonable estimates made within a limited frame at the beginning of the software project and should be updated regularly as the project progresses. It is concluded that the application works well and satisfy the users. The application is tested very well and errors are properly debugged. The site is simultaneously accessed from more than one system. Simultaneous login from more than one place is tested. The site works according to the restrictions provided in their respective browsers. Further enhancements can be made to the application, so that the web site functions very interactive and useful to existing application. We feel that we have been successful in providing the application a user-friendly look. The quality features like correctness, efficiency, usability, maintainability, portability, accuracy, errors free, tolerance and expandability are successfully done.

On completion I believe that the application completes the purpose it was made for.

**11. BIBLIOGRAPHY**

**REFERENCES**

**1. BOOKS**

**A.** P.J Deitel, H.M Deitel, Web Programming, Fourth Edition, PHI publishers. Kevin Loney, Oracle Database 10g, Tata Mcgraw – Hill publishers, New Delhi.

**B.** Elisabeth Freeman and Eric Freeman, Head First HTML with CSS and XHTML, Shroff publishers.

**C.** Ivor Horton’s, Java Programming, Seventh Edition, Wiley India Pvt. Ltd, New Delhi.

**D.** Ramakrishnan, Gehre, Data Base Management System, Third Edition , Mcgraw- Hill publishers.

**E.** Sommerville, Software Engineering, Eighth Edition, Pearson Education publishers.

**F.** Thomas Powell, Fritz Scheneider, JavaScript, Second Edition, Tata Mcgraw- Hill publishers, New Delhi.

**2. WEBSITES**

[www.google.com](http://www.google.com/)

[www.w3school.com](http://www.w3school.com/)

[www.vortex.com](http://www.vortex.com/)

[www.wikipedia.com](http://www.wikipedia.com/)

[www.tizag.com](http://www.tizag.com/)

[www.killersites.com](http://www.killersites.com/)

**12 . USER MANUAL**

The general instructions to register as a user and avail services is as follows:

* First if a user is not registered he must register using registration page where he needs to enter the basic details such as Registration number , first name , last name , contact , email id, password
* Then next he must login to enter the website. The user has a unique id of the college called register number through which he can always login as a user.
* He then must enter all his personal details in the dashboard section of the user profile .
* Next he must select the amount of seater room he wants in the booking hostel section whether he wants 4 seater or 3 or 2 or 1.
* Then he must also enter his personal details such as fathers name , occupation,

Phone number and then guardian name , address , phone number , he must also enter whether he wants the mess facilities or not .

**1. Introduction**

* + 1. **Background**

This user manual is to help you maintain, update and use our new Hostel management website easily and quickly. You’ll find detailed screenshots, explanations and instructions on how to manage our new site. After you’ve tried it a couple of times, you’ll find it’s easy to use and you may not need this guide any longer, but it is here as a reference.

* + 1. **About**

Our website is built using a PHP , HTML , CSS , java script , is an online website which helps organizations in maintaining and managing hostel rooms student details who have enrolled, food requirements and rental systems etc .

* 1. **Basics: Accessing our website and Admin**
     1. **Accessing our website**
* You can access our website through the following link

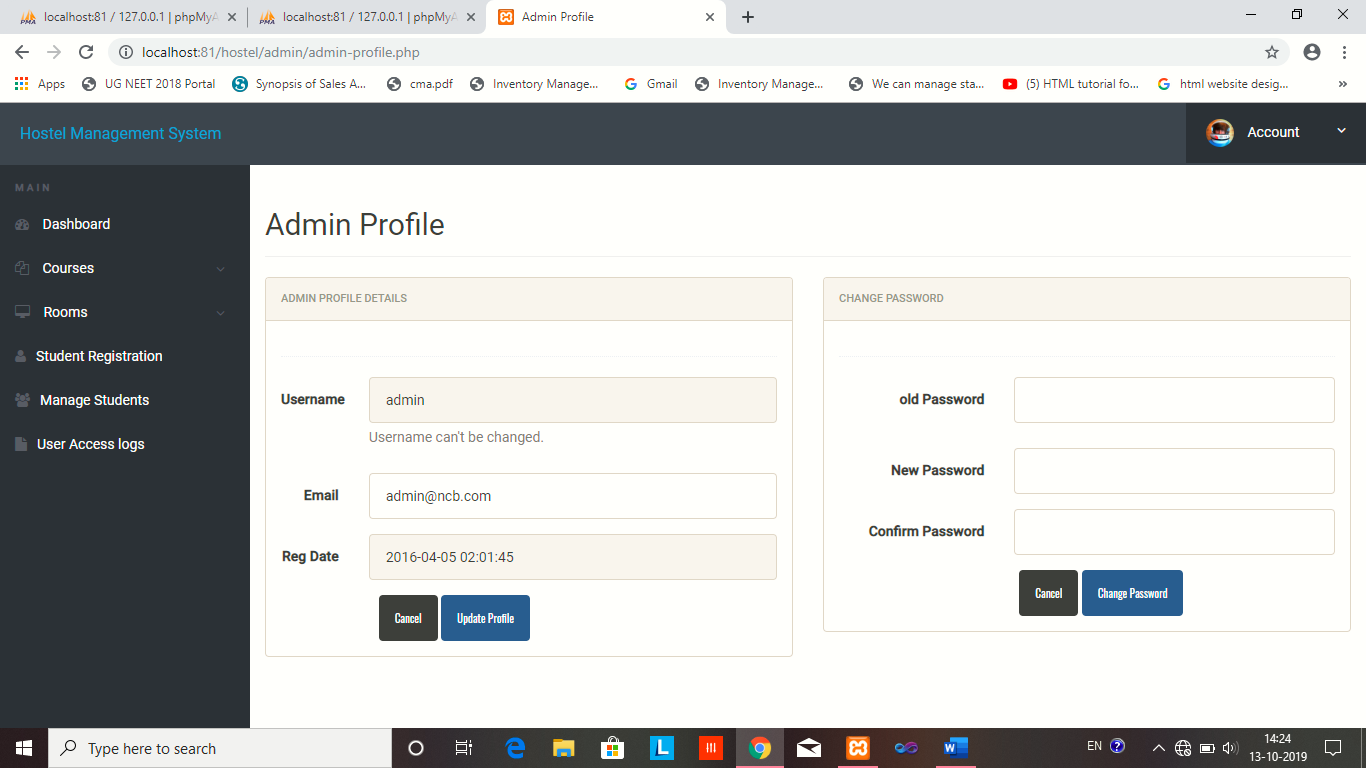
<http://localhost/pet/Pet>

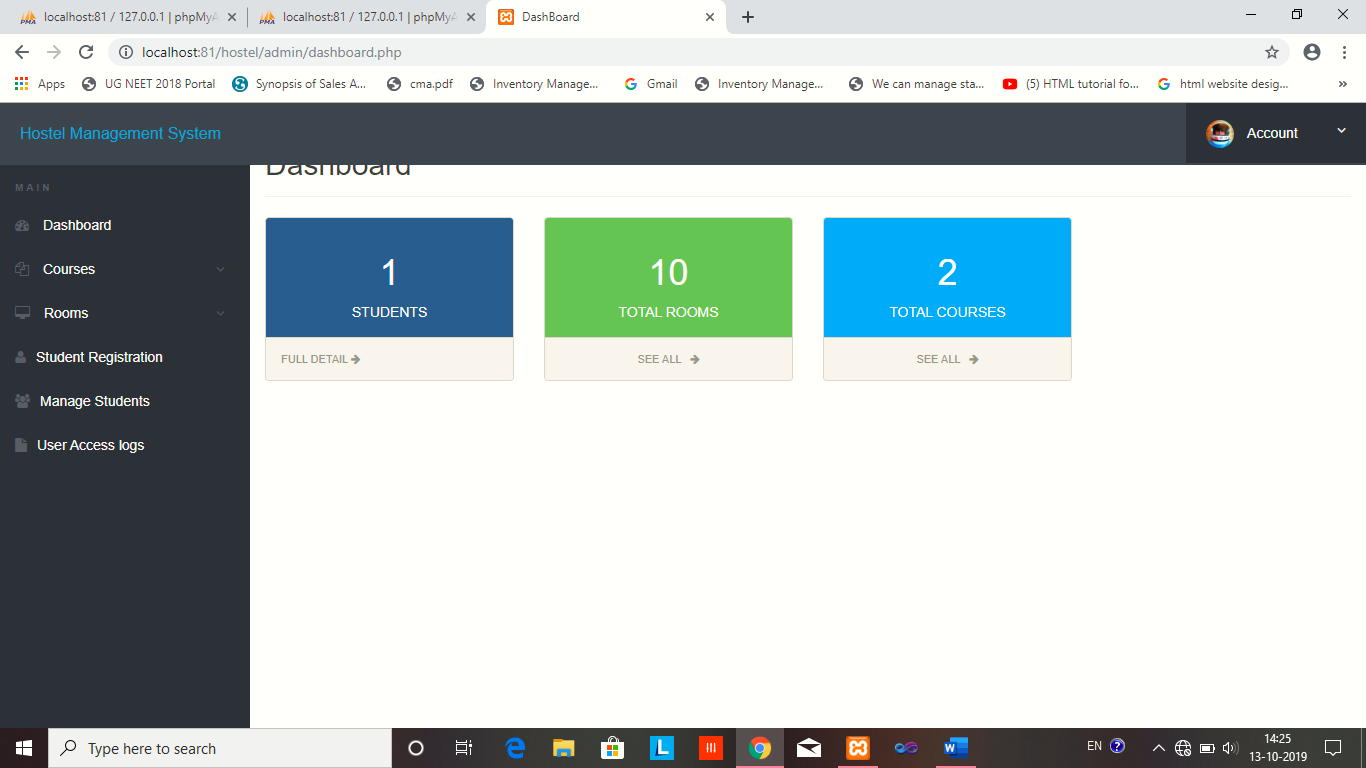
* The admin panel where u can add or delete or manage things is through the following link

<http://localhost/pet/Pet>/admin

You will need to enter your username and password to use the site or book rooms or for admin to manage student details .

**2.2 Admin area**





Admin area is where you can make changes .

You can manage student details , u can view student who have accessed the log through which you can know when did they register and their enrolling time .

You can view the rooms that have been filled already or how many more rooms are available that are empty.

You can also view to which course each student belongs to

You can view student details through their unique student register numbers

You can also view students personal information as in emergency to contact their guardians etc .

All their guardians number their family details are stored which the admin can use to view and manage anytime