

# 1.INTRODUCTION

## 1. About the Project

Hostel management system project is a software application designed to assist hostel managers and staff in their daily operations. One of the key features of this system is room allocation and management. With this system, the hostel can automate the process of room allocation by providing a centralized dashboard for hostel staff to view and manage room availability. The system can handle room booking in real-time. Hostel staff can allocate rooms to students based on their preferences, room type, and availability. The room management feature of this system enables hostel staff to monitor the status of each rooms . This helps the hostel staff to ensure that the students that the students have a comfortable and enjoyable stay. Furthermore, the hostel management system project can also help in managing room applications. The system can provide an online application form for students to fill out their details and requirements for their stay. This helps the hostel staff to manage and review applications effectively, ensuring that guests are assigned suitable rooms and are satisfied with their stay. Overall, the hostel management system's room allocation, room application, and room management features can help the hostel staff to manage the rooms more efficiently and provide the students with a seamless and enjoyable experience.

The languages used in this project are HTML, CSS, PHP and Database is MySQL.

### List of Modules :

- Login Module
- User Module
  - Dashboard
  - My Profile
  - Book Hostel
  - Room Details
  - Change Password

- Admin Module
  - Dashboard
  - Courses
    - Add Courses
    - Manage Courses
  - Rooms
    - Add Rooms
    - Manage Rooms
  - Manage Students
  - User Access Log

**Technologies Used in the Project :**

- **Front-End** : HTML,CSS,JS,PHP
- **Back-End** : MYSQL.

## **2. SYSTEM STUDY**

### **2.1 Existing System**

hostel management systems were largely manual and paper-based. Hostel managers used to keep records and manage operations using pen and paper or spreadsheets. Guest check-ins and check-outs were done manually, and there was little automation for managing reservations, billing, or inventory. This manual approach could be time-consuming and error-prone, making it difficult for hostel managers to keep up with the demands of running a hostel.

### **2.2 Disadvantages of Existing System**

- **Technical Issues:** Like all software, hostel management systems can have technical issues, such as bugs, glitches, or downtime. This can cause disruptions to hostel operations and lead to frustration for both admin and students.
- **Learning Curve:** Learning to use a new system can be challenging, especially for staff who may be used to a manual or paper-based system. It may take time and effort to train staff on how to use the new system, which could be a disadvantage for smaller hostels with limited resources.
- **Data Security:** Digital systems can be vulnerable to data breaches and other security threats. Hostel managers need to ensure that the system they use is secure and that they have the proper safeguards in place to protect student data.

### **2.3 Proposed System**

hostel management systems have evolved to become much more sophisticated and streamlined. They are largely digital, cloud-based, and provide a range of features that help hostel admin to manage their operations efficiently. Modern hostel management systems offer automated reservation management, billing and payment processing, and inventory management. They also provide detailed reporting and analytics, allowing hostel admin to make data-driven decisions. With the rise of mobile devices, many modern hostel management systems also offer mobile check-in and other features that make it easy for guests to manage their stay.

## **2.4 Advantages of Proposed System**

- Increased efficiency: Hostel managers can manage operations more efficiently, saving time and reducing errors.
- Enhanced security: Digital systems offer better security and data protection than traditional paper-based systems.
- Improved guest experience: Modern systems make it easy for guests to make reservations, check-in and check-out, and manage their stay, providing a seamless guest experience.
- Real-time reporting: Modern systems provide real-time reporting and analytics, allowing hostel managers to make informed decisions.

## **2.5 Problem Definition and Description**

A System analysis is the detailed study of the various operations performed by the system and their relationships within and outside the system. Analysis is the process of breaking something into its parts so that the whole may be understood. System Analysis is concerned with becoming aware of the problem, identifying the relevant and most decisional variables, analysing and synthesizing the various factors and determining an optimal or at least a satisfactory solution. During this a problem is identified, alternate system solutions are studied and recommendations are made about committing the resources used to design the system.

## 3. SYSTEM ANALYSIS

### 3.1 Packages Selected

- **Front-End:** HTML, CSS, JS, PHP
- **Back-End:** MYSQL.

### 3.2 Resource Required

#### 3.2.1 Hardware Specification


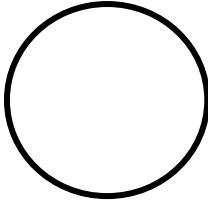


**Processor** : INTEL® CORE™ i5 – 7<sup>th</sup> Gen CPU 3.00 Ghz  
**RAM** : 8GB  
**Hard Disk** : 1TB  
**Keyboard** : Standard Keyboard

#### 3.2.2 Software Specification

**OS** : Windows 10  
**Frontend** : PHP  
**Backend** : MYSQL  
**Tool** : Sublime Text

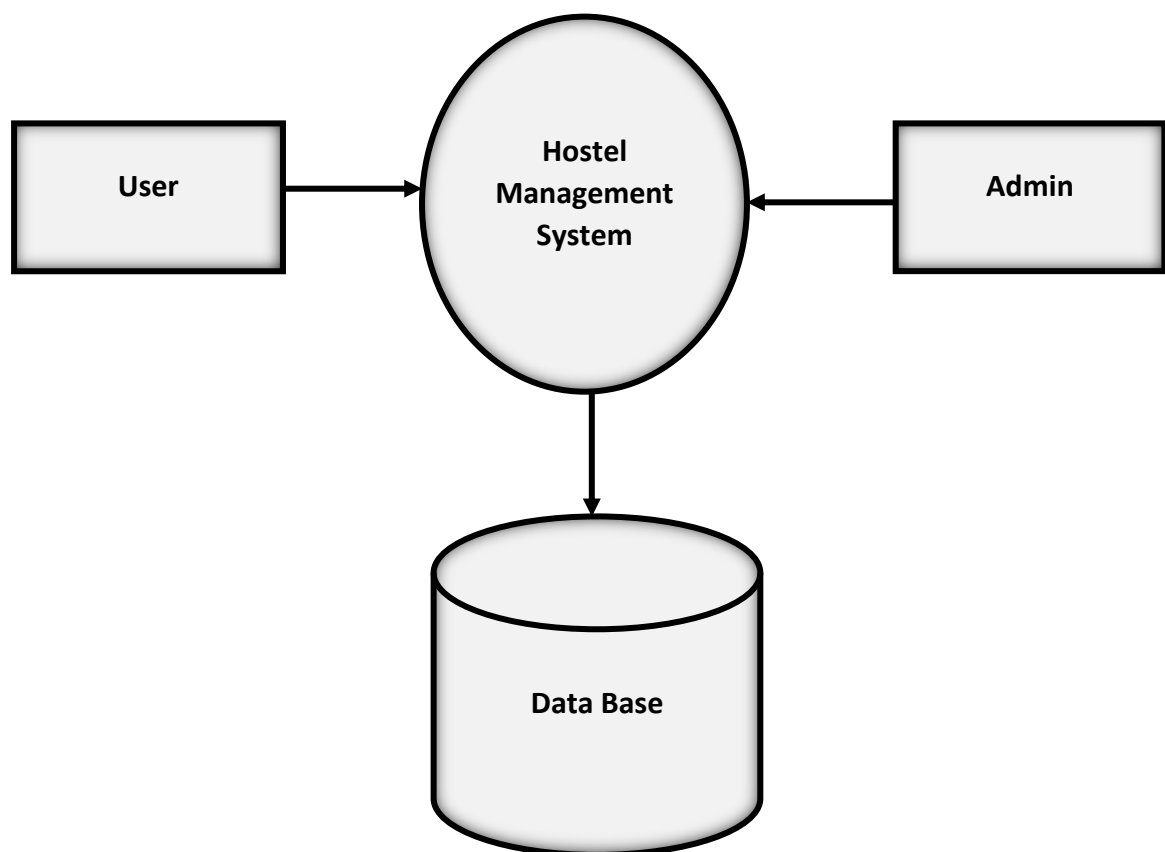
### 3.3 Data Flow Diagram

A data-flow diagram is a way of representing a flow of data through a system. It is a two-dimensional diagram. The DFD also provides information about the outputs and inputs of each entity and the process itself. A data-flow diagram has no control flow. It shows how data enters and leaves the system, what changes the information, and where data is stored. This type of diagram helps in business development.

Symbol	Description
	It represents an external entity. A source of data or a destination for data.
	It represents a process that is performed by the system.
	It indicates the data store, a place where the data is to be stored.
	It indicates data flow.

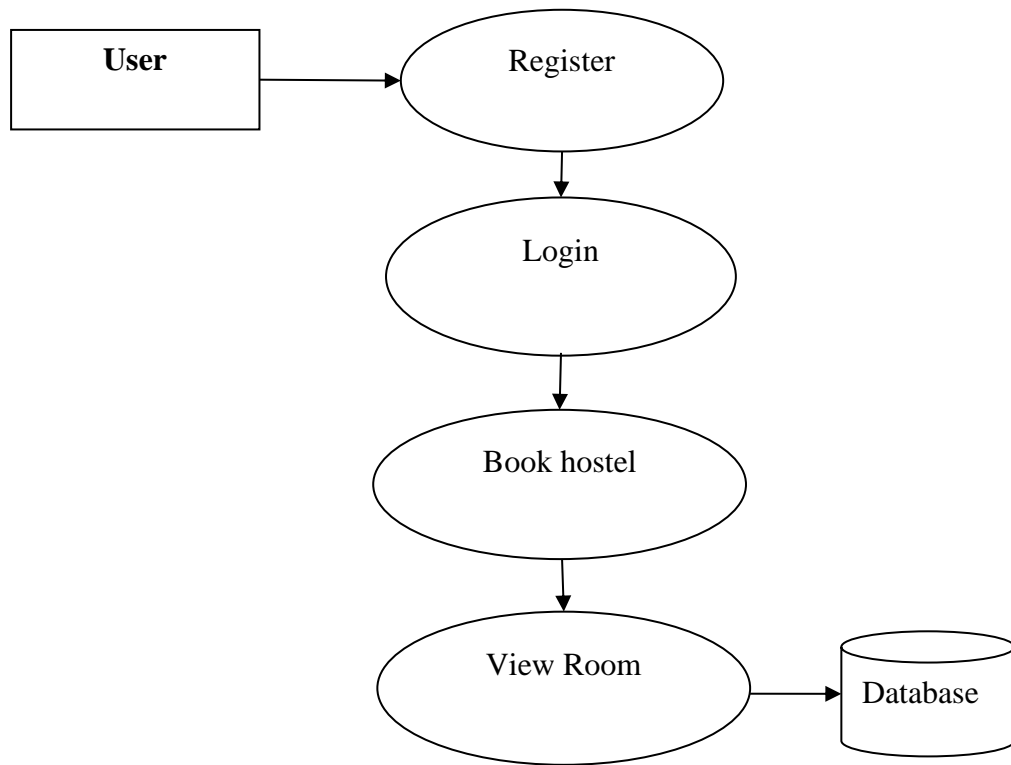
### **Level 0:**

The Level 0 DFD shows how the system is divided into subsystem. This subsystem shows only overall aspect of what the system going to do. It shows the basic functionality of the system



### Level 1:

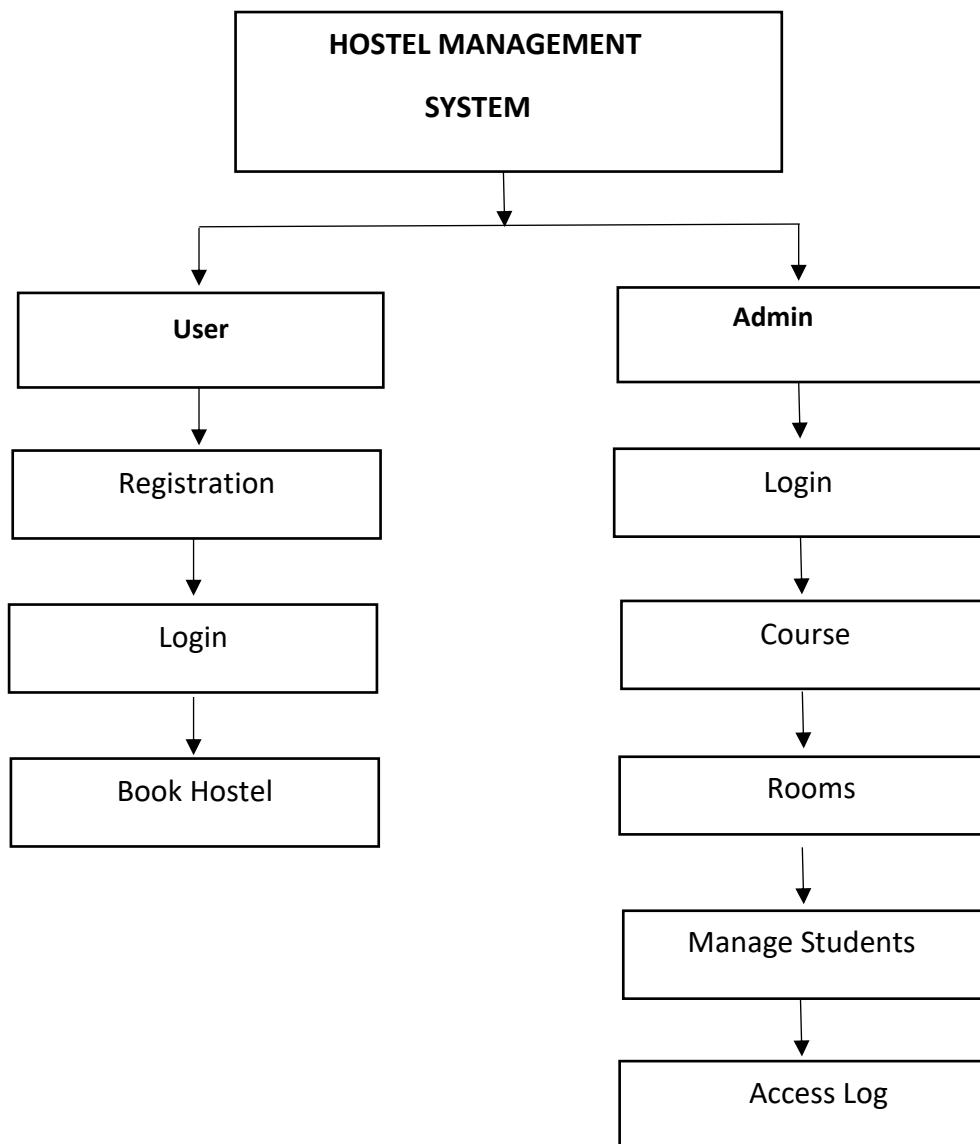
The level 1 DFD shows how the subsystem is divided into sub subsystem. It shows how the data flow through the sub subsystem and where it is going to store. It shows how the system should be accessed.





## 4. SYSTEM DESIGN

### 4.1 Architectural Design



## 4.2 I/O Form Design

### 1.Student Registration

Register number:

First Name :

Last Name :

Gender :

Email Id :

Password :

Confirm Password :

## I/O Form Design

### 1.Student Registration

Register number:

Localhost Says!

Student registered successfully

First Name

:

Last Name

:

Gender

:

Email Id

:

Password

:

Confirm Password

:

SUBMIT

### 2.Add Rooms

Select Seater :

5

Room No

:

101

Fees

:

5000

CREATE ROOM

### 3.Add Rooms

Select Seater :

Local Says!

Room Created succssfully

Room No :

101

Fees :

5000

CREATE ROOM

### 4.3 Tables

**Database Name : 20UCS130**

**Table Name : User\_Registration**

Field Name	Data Type	Width/Length	Constrains
Register No	Varchar	10	Primary Key
First Name	Varchar	15	Not Null
Middle Name	Varchar	20	Not Null
Last Name	Varchar	25	Not Null
Gender	Varchar	20	Not Null
Contact No	Int	10	Not Null
Email Id	Varchar	20	Not Null
Password	Varchar	15	Not Null
Confirm Password	Varchar	15	Not Null

**Table Name : User\_Login**

Field Name	Data Type	Width/Length	Constrains
EMAIL/REGISTER NO	varchar	50	Primary key
PASSWORD	varchar	10	Not null

**Table Name: Book\_hostel**

<b>Field Name</b>	<b>Datatype</b>	<b>Width/Length</b>	<b>Constraints</b>
Room no	int	10	NOT NULL
seater	int	20	NOT NULL
Fees per month	int	10	NOT NULL
Food status	varchar	15	NOT NULL
Stay from	int	40	NOT NULL
Duration	int	20	NOT NULL
Course	varchar	30	NOT NULL
First Name	varchar	15	NOT NULL
Last Name	varchar	10	NOT NULL
Gender	varchar	10	NOT NULL
Contact NO	int	30	NOT NULL
Email Id	varchar	15	NOT NULL
Emergency Contact	int	30	NOT NULL
Guardian Name	varchar	40	NOT NULL
Guardian Relation	varchar	50	NOT NULL
Guardian contact no	int	30	NOT NULL
Address	varchar	50	NOT NULL
State	varchar	20	NOT NULL
City	varchar	20	NOT NULL
Pincode	int	15	NOT NULL

**Table Name : Admin\_Login**

<b>Field Name</b>	<b>Data Type</b>	<b>Width/Length</b>	<b>Constrains</b>
EMAIL/REGISTER NO	varchar	50	Primary key
PASSWORD	varchar	10	Not null

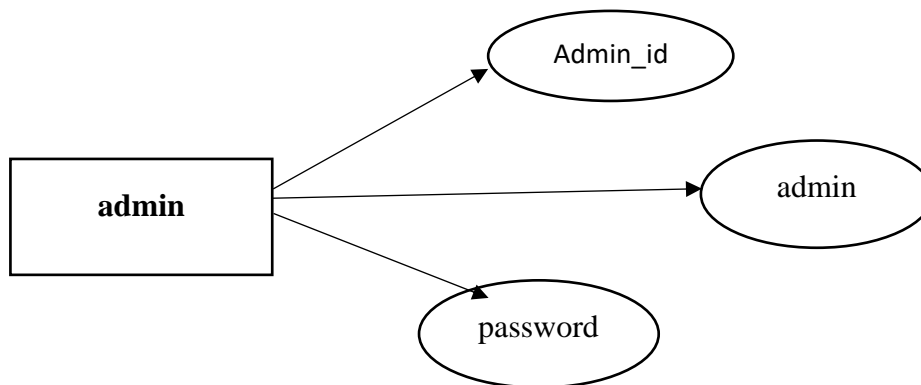
**Table Name : Add\_Room**

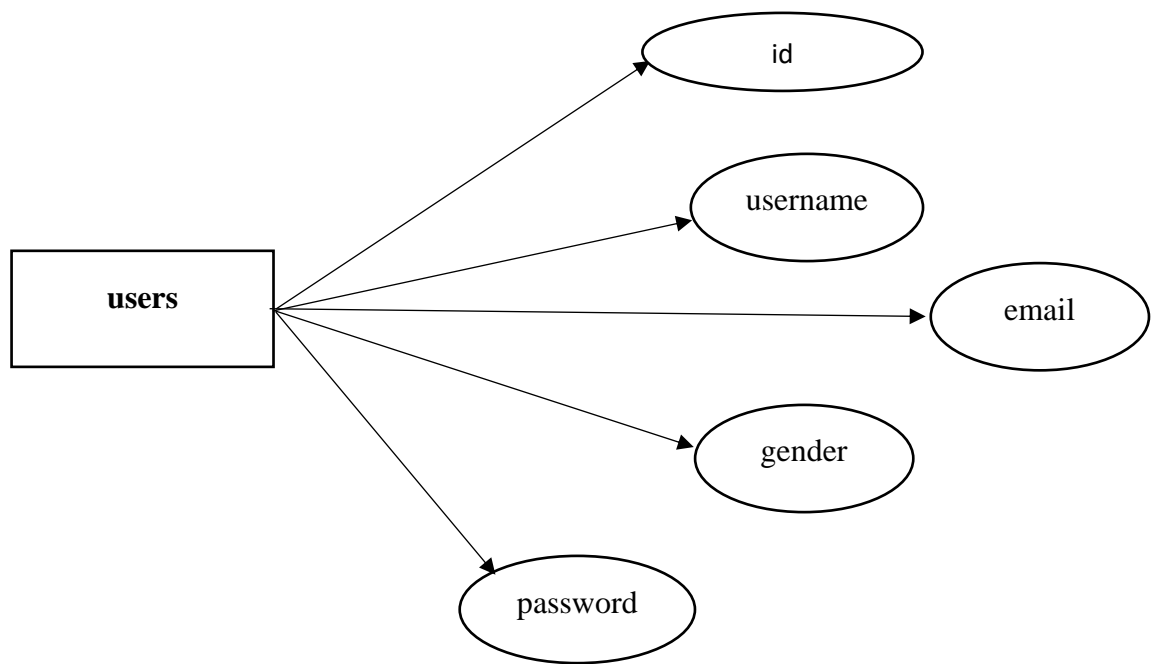
Field Name	Data Type	Width/Length	Constrains
Select Seater	Int	20	Not Null
Room NO	Int	15	Not Null
Fees	Int	10	Not Null

**Table Name : Add\_Course**

Field Name	Data Type	Width/Length	Constrains
Corse Code	Varchar	20	Not Null
Course Name(short)	Varchar	15	Not Null
Course Name(Full)	Varchar	20	Not Null

#### 4.4 E-R DIAGRAM





## 4.5 DATA DICTIONARY REGISTRATION

FIELD NAME	TABLE NAME	DESCRIPTION	SAMPLE VALUES
Register No	User Register	Specify The <b>id</b>	125
First Name	User Register	Specify The user name	Mari
Middle Name	User Register	Specify The Name	doss
Last Name	User Register	Specify The Nmae	A.
Gender	User Register	Specify The Gender	Male
Contact No	User Register	Specify The Number	6369868329
Email Id	User Register	Specify The Email	mari@gmail.com
Password	User Register	Specify The Password	*****
Confirm Password	User Register	Specify The Password	*****

### **ADD\_ROOMS**

<b>FIELD NAME</b>	<b>TABLE NAME</b>	<b>DESCRIPTION</b>	<b>SAMPLE VALUES</b>
Select Seater	ADMIN	Admin select to the seaters count	10
Room NO	ADMIN	Specify the room number	20
Fees	ADMIN	Admin can fix the fees	30

### **ADD\_COURSE**

<b>FIELD NAME</b>	<b>TABLE NAME</b>	<b>DESCRIPTION</b>	<b>SAMPLE VALUES</b>
Corse Code	Add course	Specify the code	20
Course Name(short)	Add course	Specify the name	30
Course Name(Full)	Add course	Specify the name	30



## **5. SYSTEM DEVELOPMENT**

### **5.1 FUNCTIONAL DOCUMENTATION**

#### **MODULES**

- Admin
- User

#### **ADMIN**

##### **Login**

This module used for admin login

##### **Dashboard**

Admin dashboard related total students, total room, Total course.

##### **Room management**

Admin can create rooms seater, fees (Add, edit and delete)

##### **Course management**

Admin can add and manage the course(Add, Edit and delete)

##### **Manage students**

Admin manage all details from the students

##### **Access log**

Admin can see user login access details

#### **USER**

##### **Login**

This module used for user login

##### **User profile**

User can update the profile after login

##### **Change Password**

Allow user to change the details

##### **Book Hostel**

User can book the hostel

Hostel Details

User can see the hostel details

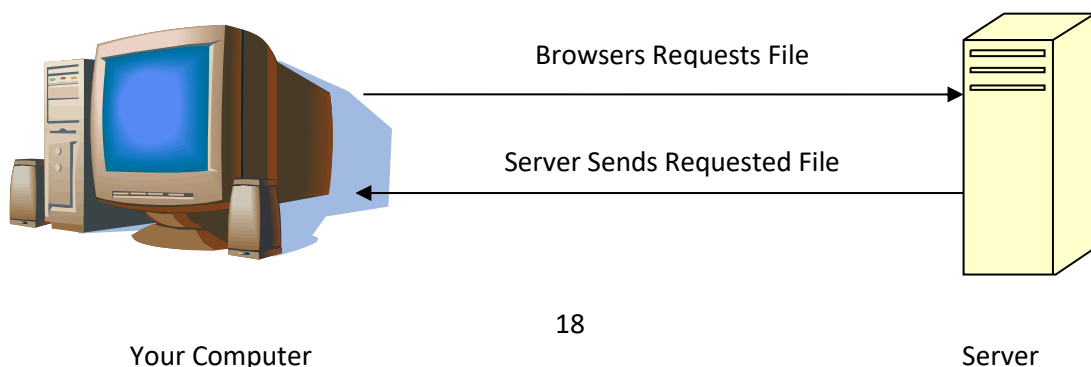
## 5.2 SPECIAL FEATURES OF LANGUAGE/UTILITY

### Front End

PHP: Hypertext Pre-processor (the name is a recursive acronym) is a widely used, general-purpose scripting language that was originally designed for web development to produce dynamic web pages. For this purpose, PHP code is embedded into the HTML source document and interpreted by a web server with a PHP processor module, which generates the web page document. As a general-purpose programming language, PHP code is processed by an interpreter application in command-line mode performing desired operating system operations and producing program output on its standard output channel. It may also function as a graphical application. PHP is available as a processor for most modern web servers and as standalone interpreter on most operating systems and computing platforms. PHP was originally created by Rasmus Lerdorf in 1995 and has been in continuous development ever since. The main implementation of PHP is now produced by The PHP Group and serves as the de facto standard for PHP as there is no formal specification. PHP is free software released under the PHP License, which is incompatible with the GNU General Public License (GPL) because restrictions exist regarding the use of the term PHP.

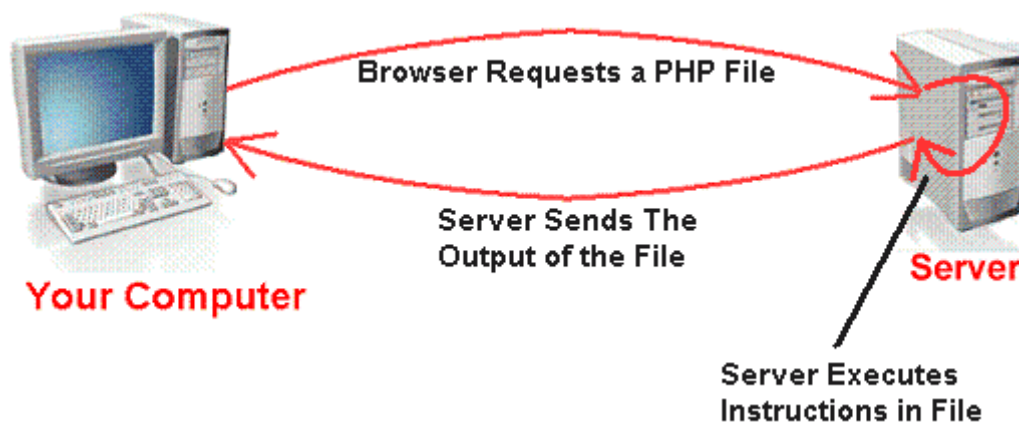
Hypertext refers to files linked together using hyperlinks, such as HTML (HyperText Markup Language) files. Pre-processing is executing instructions that modify the output. Below is a demonstration of the difference between HTML and PHP files.

### Accessing an HTML Page



1. Your browser sends a request to that web page's server (computer) for the file (HTML or image) you wish to view.
2. The web server (computer) sends the file requested back to your computer.
3. Your browser displays the file appropriately.
4. If you request a PHP file (ends with ".php"), the server handles it differently.

### **Accessing a PHP Page**



### **Accessing a PHP Page**

1. Your browser sends a request to that web page's server for the PHP file you wish to view.
2. The web server calls PHP to interpret and perform the operations called for in the PHP script.
3. The web server sends the output of the PHP program back to your computer.
4. Your browser displays the output appropriately.

### **Benefit of PHP**

Because the server does processing, the output of PHP files changes when its input changes. For example, most of the pages on the Horticulture site have only two (2) PHP commands:

1. Include the header file that defines the links on the left, the banner, and the quick links at the top.
2. Include the footer file that displays the mission statement and Horticulture contact information.

Because including the files is performed every time the PHP file is accessed, when the header/footer files change, the new content will be immediately updated. In other words, if you add a new link, every page that includes the header will immediately display the new link.

## **Security**

About 30% of all vulnerabilities listed on the National Vulnerability Database are linked to PHP. These vulnerabilities are caused mostly by not following best practice programming rules: technical security flaws of the language itself or of its core libraries are not frequent (23 in 2008, about 1% of the total). Recognizing that programmers make mistakes, some languages include taint checking to detect automatically the lack of input validation which induces many issues. Such a feature is being developed for PHP, but its inclusion in a release has been rejected several times in the past. There are advanced protection patches such as Suhosin and Hardening-Patch, especially designed for Web hosting environments.

PHPIDS adds security to any PHP application to defend against intrusions. PHPIDS detects attacks based on cross-site scripting (XSS), SQL injection, header injection, directory traversal, remote file execution, remote file inclusion, and denial-of-service (DoS)

## **Syntax**

The PHP interpreter only executes PHP code within its delimiters. Anything outside its delimiters is not processed by PHP (although non-PHP text is still subject to control structures described in PHP code). The most common delimiters are `<?php` to open and `?>` to close PHP sections. `<script language="php">` and `</script>` delimiters are also available, as are the shortened forms `<?or<?=` (which is used to echo back a string or variable) and `?>` as well as ASP-style short forms `<%` or `<%=` and `%>`. While short delimiters are used, they make script files less portable as support for them can be disabled in the PHP configuration, and so they are discouraged. The purpose of all these delimiters is to separate PHP code from non-PHP code, including HTML. The first form of delimiters, `<?php` and `?>`, in XHTML and other XML documents, creates correctly formed XML 'processing instructions'. This means that the resulting

mixture of PHP code and other markup in the server-side file is itself well-formed XML.

Variables are prefixed with a dollar symbol, and a type does not need to be specified in advance. Unlike function and class names, variable names are case sensitive. Both double-quoted (") and here-doc strings provide the ability to interpolate a variable's value into the string. PHP treats newlines as whitespace in the manner of a free-form language (except when inside string quotes), and statements are terminated by a semicolon. PHP has three types of comment syntax: /\* \*/ marks block and inline comments; // as well as # are used for one-line comments. The echo statement is one of several facilities PHP provides to output text, e.g., to a Web browser. In terms of keywords and language syntax, PHP is similar to most high level languages that follow the C style syntax. if conditions, for and while loops, and function returns are similar in syntax to languages such as C, C++, Java and Perl.

## Data types

PHP stores whole numbers in a platform-dependent range, either a 64-bit or 32-bit signedinteger equivalent to the C-language long type. Unsigned integers are converted to signed values in certain situations; this behavior is different from other programming languages. Integer variables can be assigned using decimal (positive and negative), octal, and hexadecimal notations. Floating point numbers are also stored in a platform-specific range. They can be specified using floating point notation, or two forms of scientific notation. PHP has a native Boolean type that is similar to the native Boolean types in Java and C++. Using the Boolean type conversion rules, non-zero values are interpreted as true and zero as false, as in Perl and C++. The null data type represents a variable that has no value. The only value in the null data type is NULL. Variables of the "resource" type represent references to resources from external sources. These are typically created by functions from a particular extension, and can only be processed by functions from the same extension; examples include file, image, and database resources. Arrays can contain elements of any type that PHP can handle, including resources, objects, and even other arrays. Order is preserved in lists of values and in hashes with both keys and values, and the two can be intermingled. PHP also supports strings, which can be used with single quotes, double quotes, nowdoc or heredoc syntax.

## Functions

PHP has hundreds of base functions and thousands more via extensions. These functions are well documented on the PHP site; however, the built-in library has a wide variety of naming conventions and inconsistencies. PHP currently has no functions for thread programming, although it does support multi-process programming on POSIX systems.

## MYSQL SERVER

MySQL is the world's most used open source relational database management system (RDBMS) as of 2008 that run as a server providing multi-user access to a number of databases. The MySQL development project has made its source code available under the terms of the GNU General Public License, as well as under a variety of proprietary agreements. MySQL was owned and sponsored by a single for-profit firm, the Swedish company MySQL AB, now owned by Oracle Corporation.

MySQL is a popular choice of database for use in web applications, and is a central component of the widely used LAMP open source web application software stack—LAMP is an acronym for "Linux, Apache, MySQL, Perl/PHP/Python." Free-software-open source projects that require a full-featured database management system often use MySQL.

For commercial use, several paid editions are available, and offer additional functionality. Applications which use MySQL databases include: TYPO3, Joomla, Word Press, phpBB, MyBB, Drupal and other software built on the LAMP software stack. MySQL is also used in many high-profile, large-scale World Wide Web products, including Wikipedia, Google(though not for searches), ImagebookTwitter, Flickr, Nokia.com, and YouTube.

## Interimages

MySQL is primarily an RDBMS and ships with no GUI tools to administer MySQL databases or manage data contained within the databases. Users may use the included command line tools, or use MySQL "front-ends", desktop software and web applications that create and manage MySQL databases, build database structures, back

up data, inspect status, and work with data records. The official set of MySQL front-end tools, MySQL Workbench is actively developed by Oracle, and is freely available for use.

## **Graphical**

The official MySQL Workbench is a free integrated environment developed by MySQL AB, that enables users to graphically administer MySQL databases and visually design database structures. MySQL Workbench replaces the previous package of software, MySQL GUI Tools. Similar to other third-party packages, but still considered the authoritative MySQL frontend, MySQL Workbench lets users manage database design & modeling, SQL development (replacing MySQL Query Browser) and Database administration (replacing MySQL Administrator).

MySQL Workbench is available in two editions, the regular free and open source Community Edition which may be downloaded from the MySQL website, and the proprietary Standard Edition which extends and improves the feature set of the Community Edition.

## **Command line**

MySQL ships with some command line tools. Third-parties have also developed tools to manage a MySQL server, some listed below.

- Maatkit - a cross-platform toolkit for MySQL, PostgreSQL and Memcached, developed in Perl Maatkit can be used to prove replication is working correctly, fix corrupted data, automate repetitive tasks, and speed up servers. Maatkit is included with several GNU/Linux distributions such as CentOS and Debian and packages are available for Programming

MySQL works on many different system platforms, including AIX, BSDi, FreeBSD, HP-UX, eComStation, i5/OS, IRIX, Linux, Mac OS X, Microsoft Windows, NetBSD, Novell NetWare, OpenBSD, OpenSolaris, OS/2 Warp, QNX, Solaris, Symbian, SunOS, SCO OpenServer, SCO UnixWare, Sanos and Tru64. A port of MySQL to OpenVMS also exists.<sup>[32]</sup>

MySQL is written in C and C++. Its SQL parser is written in yacc, and a home-brewed lexical analyzer. Many programming languages with language-specific APIs include libraries for accessing MySQL databases. These include MySQL Connector/Net for integration with Microsoft's Visual Studio (languages such as C# and VB are most commonly used) and the JDBC driver for Java. In addition, an ODBCinterimage called MyODBC allows additional programming languages that support the ODBC interimage to communicate with a MySQL database, such as ASP or ColdFusion. The HTSQL - URL-based query method also ships with a MySQL adapter, allowing direct interaction between a MySQL database and any web client via structured URLs.

## Features

As of April 2009, MySQL offered MySQL 5.1 in two different variants: the open source MySQL Community Server and the commercial Enterprise Server. MySQL 5.5 is offered under the same licences. They have a common code base and include the following features:

- A broad subset of ANSI SQL 99, as well as extensions
- Cross-platform support
- Stored procedures
- Triggers
- Cursors
- Updatable Views
- Information schema
- Strict mode (ensures MySQL does not truncate or otherwise modify data to conform to an underlying data type, when an incompatible value is inserted into that type)
- X/Open XAdistributed transaction processing (DTP) support; two phase commit as part of this, using Oracle's InnoDB engine
- Independent storage engines (MyISAM for read speed, InnoDB for transactions and referential integrity, MySQL Archive for storing historical data in little space)
- Transactions with the InnoDB, and Cluster storage engines; savepoints with InnoDB
- SSL support
- Query caching



- Sub-SELECTs (i.e. nested SELECTs)
- Replication support (i.e. Master-Master Replication & Master-Slave Replication) with one master per slave, many slaves per master, no automatic support for multiple masters per slave.
- Full-text indexing and searching using MyISAM engine
- Embedded database library
- Unicode support (however prior to 5.5.3 UTF-8 and UCS-2 encoded strings are limited to the BMP, in 5.5.3 and later use utf8mb4 for full unicode support)
- ACID compliance when using transaction capable storage engines (InnoDB and Cluster)
- Partititoned tables with pruning of partitions in optimiser
- Shared-nothing clustering through MySQL Cluster
- Hot backup (via mysqlhotcopy) under certain conditions
- Multiple storage engines, allowing one to choose the one that is most effective for each table in the application (in MySQL 5.0, storage engines must be compiled in; in MySQL 5.1, storage engines can be dynamically loaded at run time): Native storage engines (MyISAM, Falcon, Merge, Memory (heap), Federated, Archive, CSV, Blackhole, Cluster, EXAMPLE, Maria, and InnoDB, which was made the default as of 5.5). Partner-developed storage engines (solidDB, NitroEDB, ScaleDB, TokuDB, Infobright (formerly Brighthouse), Kickfire, XtraDB, IBM DB2). InnoDB used to be a partner-developed storage engine, but with recent acquisitions, Oracle now owns both MySQL core and InnoDB.

### **5.3 How the form are executed ?**

**There are six forms they are executed as follow.**

#### **Admin Login page**

- Step 1: Start
- Step 2: Enter the username
- Step 3: Enter the password
- Step 4: Click the submit button
- Step 5: Stop

#### **User Registration page**

- Step 1: Start
- Step 2: Enter the username
- Step 3: Enter the email
- Step 4: Select the gender
- Step 5: Enter the password
- Step 6: Click the submit button
- Step 7: Stop

#### **User Login page**

- Step 1: Start
- Step 2: Enter the username
- Step 3: Enter the password
- Step 4: Click the submit button
- Step 5: Stop

## **BOOK HOSTEL FORM**

Step 1: Start

Step 2: Enter the room related information's.

Step 3: Enter the personal information's.

Step 4: Enter the correspondence address.

Step 5: Enter the personal address.

Step 6: Click the submit button.

## **ADD COURSE FORM**

Step 1: Start

Step2: Enter the course code.

Step 3: Enter the course name (short)

Step 4: Enter the course name (full)

Step 5: Click the Add course button.

## **ADD ROOM FORM**

Step 1: Start

Step 2: Select the seater

Step 3: Enter the room.no.

Step 4: Enter the fees per month

Step 5: Click the add room button.

## **6.TESTING**

### **6.1 TYPES OF TESTING DONE**

Testing is a set activity that can be planned and conducted systematically. Testing begins at the module level and work towards the integration of entire computers based system. Nothing is complete without testing, as it is vital success of the system.

Testing Objectives:

There are several rules that can serve as testing objectives, they are

1. Testing is a process of executing a program with the intent of finding an error
2. A good test case is one that has high probability of finding an undiscovered error.
3. A successful test is one that uncovers an undiscovered error.

If testing is conducted successfully according to the objectives as stated above, it would uncover errors in the software. Also testing demonstrates that software functions appear to the working according to the specification, that performance requirements appear to have been met.

There are three ways to test a program

1. For Correctness
2. For Implementation efficiency
3. For Computational Complexity.

Tests used for implementation efficiency attempt to find ways to make a correct program faster or use less storage. It is a code-refining process, which reexamines the implementation phase of algorithm development. Tests for computational complexity amount to an experimental analysis of the complexity of an algorithm or an experimental comparison of two or more algorithms, which solve the same problem.

The data is entered in all forms separately and whenever an error occurred, it is corrected immediately. A quality team deputed by the management verified all the necessary documents and tested the Software while entering the data at all levels.

## **TYPES OF TESTING**

The development process involves various types of testing. Each test type addresses a specific testing requirement. The most common types of testing involved in the development process are:

- Unit Test
- Integration Test
- System Test
- Validation Test

### **Unit Testing:**

The first test in the development process is the unit test. The is normally divided into modules, which in turn are divided into smaller units called units. These units have specific behavior. The test done on these units of code is called unit test. Unit test depends upon the language on which the project is developed. Unit tests ensure that each unique path of the project performs accurately to the documented specifications and contains clearly defined inputs and expected .

### **Integration Testing:**

In integration testing modules are combined and tested as a group. Modules are typically code modules, individual applications, source and destination applications on a network, etc. Integration Testing follows unit testing and precedes system testing. Testing after the product is code complete. Betas are often widely distributed or even distributed to the public at large in hopes that they will buy the final product when it is released.

### **System Testing**

System testing is defined as testing of a complete and fully integrated software product. This testing falls in black-box testing wherein knowledge of the inner design of the code is not a pre-requisite and is done by the testing team. It is the final test to verify

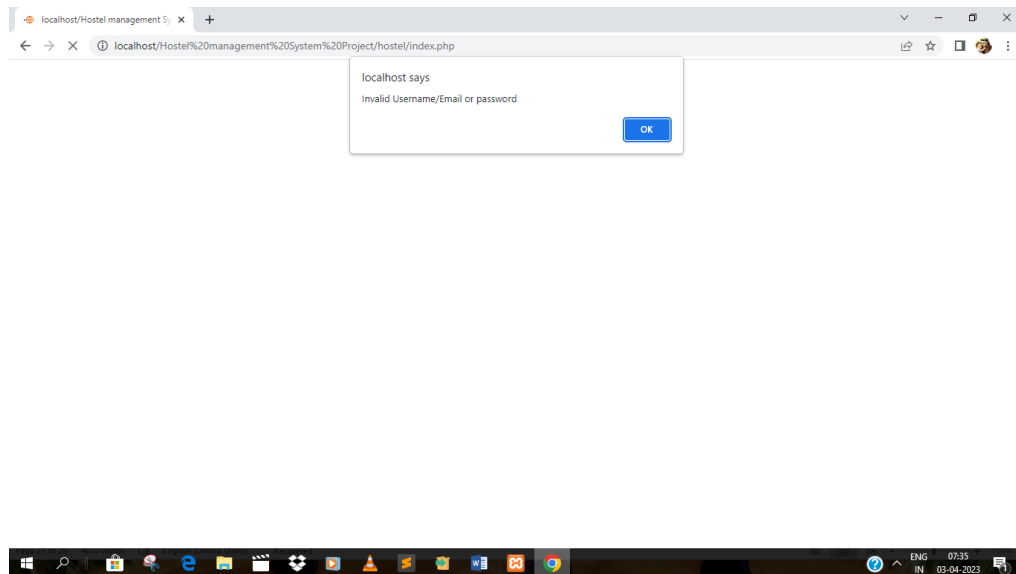
that the product to be delivered meets the specifications mentioned in the requirement document. It should investigate both functional and non-functional requirements.

## Validation Testing

The process of evaluating software during the development process or at the end of the development process to determine whether it satisfies specified business requirements. Validation Testing ensures that the product actually meets the client's needs. It can also be defined as to demonstrate that the product fulfills its intended use when deployed on appropriate environment. In a similar manner all the form input are validate

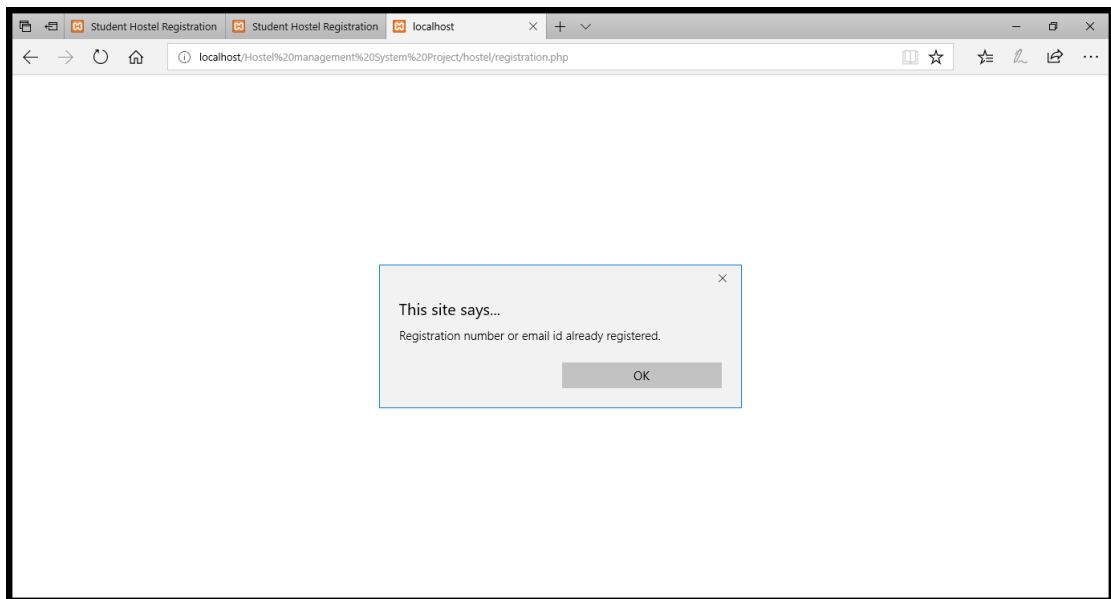
### Validating the username and password:

- Error message will appear if the username or password is incorrect.



### Validating the registration form:

- Error message will appear if the username is not entered.



## 6.2 TEST DATA & OUTPUT

### 1.Student Registration

Register number:	<input type="text" value="2oucs130"/>
First Name :	<input type="text" value="Mari"/>
Last Name :	<input type="text" value="Doss"/>
Gender :	<input type="text" value="Male"/>
Email Id :	<input type="text" value="Doss9946@gmail.com"/>
Password :	<input type="password" value="*****"/>
Confirm Password :	<input type="password" value="*****"/>
<input type="button" value="SUBMIT"/>	

## I/O Form Design

### 1.Student Registration

Register number:

Localhost Says!  
Student registered successfully

First Name

:

Last Name

:

Gender

:

Email Id

:

Password

:

Confirm Password

:

SUBMIT



## 2.Add Rooms

Select Seater :

5

Room No :

101

Fees :

5000

CREATE ROOM

## 2.Add Rooms

Select Seater :

Local Says!

Room Created succssfully

Room No :

101

Fees :

5000

CREATE ROOM

## **7. USER MANUAL**

### **7.1 HARDWARE REQUIREMENTS**

- Processor : Dual core processor 2.6.0 GHz
- RAM : 1GB
- Hard disk : 160 GB
- Compact Disk : 650 MB
- Keyboard : Standard keyboard
- Monitor : 15 inch color monitor

### **7.2 SOFTWARE REQUIREMENTS**

- Operating system: Windows OS
- Front End :PHP
- Back end : MYSQL Server
- Tool : Visual Studio Code (VS code)

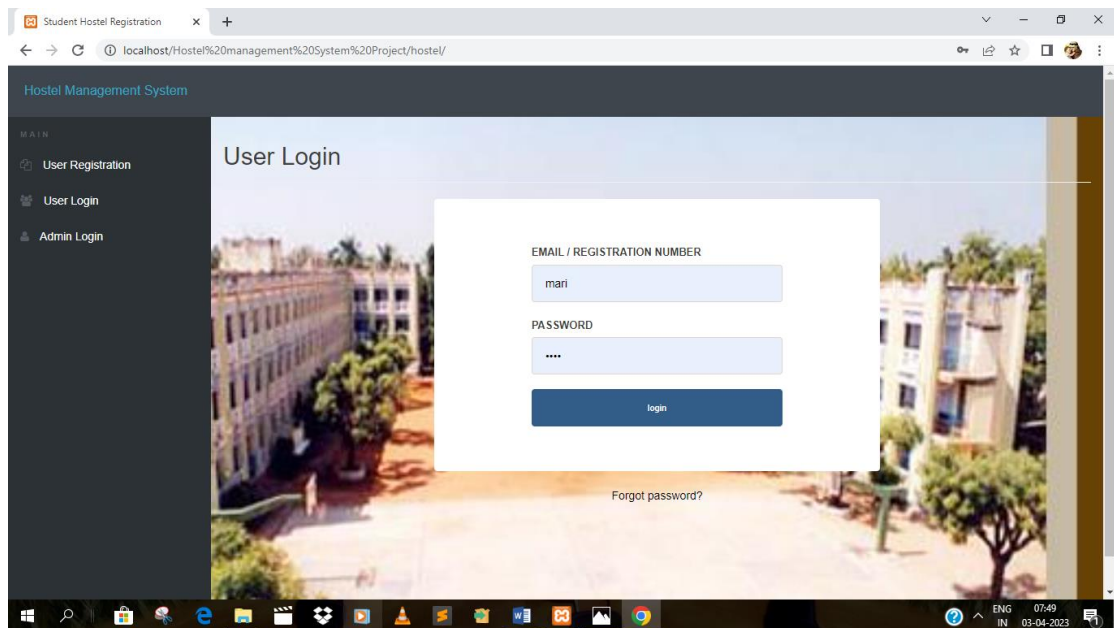
### **7.3 INSTALLATION PROCEDURE**

- Downloading Xampp Server Download the installer file for the latest version of Xampp Server, and save the file to your computer.
- Installing Xampp Server To start the installation process, you need to open the folder where you saved the file, and double-click the installer file. A security warning window will open, asking if you are sure you want to run this file. Click Run to start the installation process.
- Next you will see the Welcome to The Xampp Server Setup Wizard screen. Click Next to continue the installation.

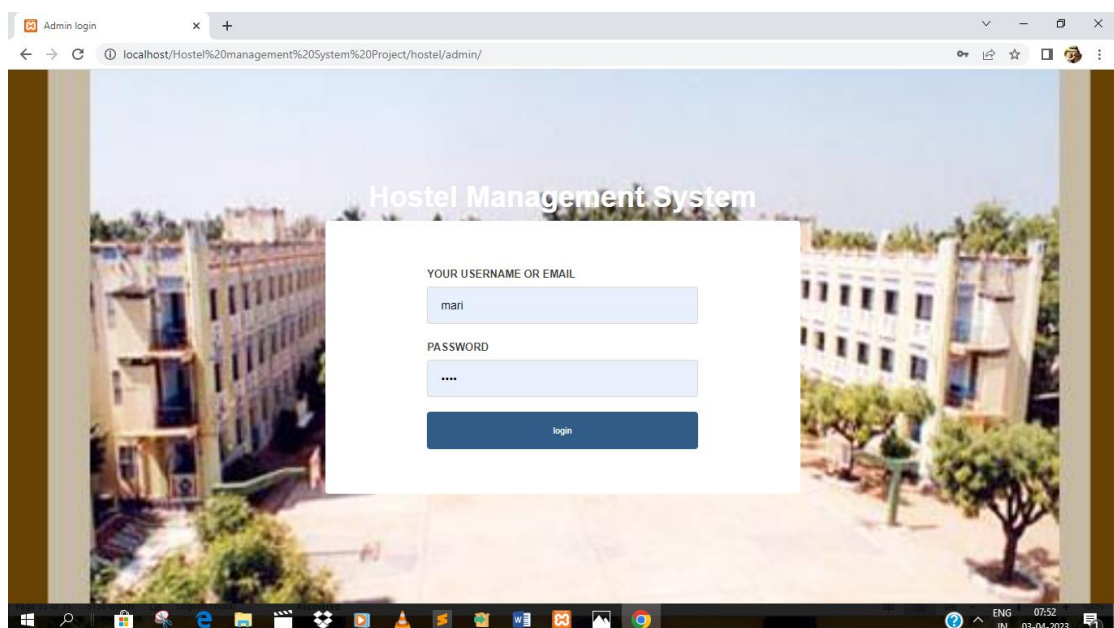
- The next screen you are presented with is the License Agreement. Read the agreement, check the radio button next to I accept the agreement, then click Next to continue the installation.
- Next you will see the Select Destination Location screen. Unless you would like to install Xampp Server on another drive, you should not need to change anything. Click Next to continue.
- Next you will see the Ready to install screen. You can review your setup choices, and change any of them by clicking back to the appropriate screen, if you choose to. Once you have reviewed your choices, click Install to continue.
- Xampp Server will begin extracting files to the location you selected.
- Windows Security Alert window will open, saying that Windows Firewall has blocked some features of the program. Check whether you want to allow Apache HTTP Server to communicate on a private or public network, then click Allow Access.
- You should see the Xampp Server icon appear in the system on the right side of your taskbar. If the icon is green, then everything is working properly. If the icon is orange, then there are issues with one of the services. If the icon is red, then both Apache and MySQL services aren't running. You will need to resolve those issues before continuing.

## 7.4 Sample I/O

### USER LOGIN



### ADMIN LOGIN



## ROOM DETAILS

**Room Realted Info**

Registration Number :	101	Apply Date :	2023-03-01 21:48:32		
Room no :	100	Seater :	5	Fees PM :	8000
Food Status:	Without Food	Stay From :	2023-03-17	Duration:	1 Months
Hostel Fee:	8000	Food Fee:	0 (You booked hostel without food).		
Total Fee :	8000				

**Personal Info**

Reg No. :	101	Full Name :	kevinjose	Email :	101@gmail.com
Contact No. :	6369868329	Gender :	male	Course :	Bachelor of Science
Emergency Contact No. :	6369868329	Guardian Name :	john	Guardian Relation :	brother
Guardian Contact No. :	8638861995				

**Addresses**

Correspondence Address	Nagapattinam Nagapattinam, 611103 Tamil Nadu	Permanent Address	Nagapattinam Nagapattinam, 611103 Tamil Nadu
------------------------	--	-------------------	--

## MANAGE REGISTER STUDENTS

**Manage Registered Students**

ALL ROOM DETAILS

Show: 10 entries Search:

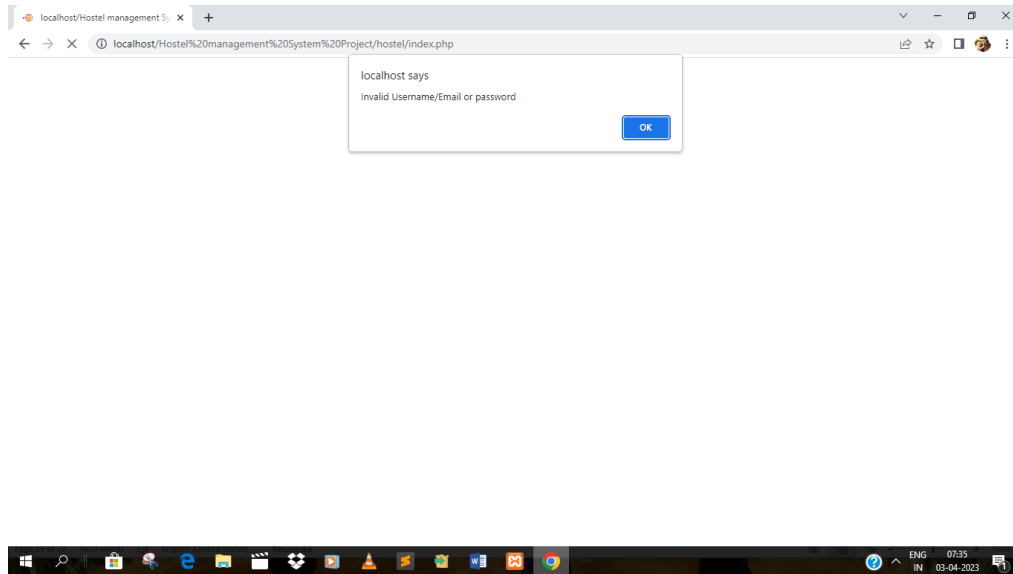
Sno.	Student Name	Reg no	Contact no	room no	Seater	Staying From	Action
1	Maridossdoss	20	6369868329	201	2	0000-00-00	<a href="#">edit</a> <a href="#">delete</a>
2	kevinjose	101	6369868329	100	5	2023-03-17	<a href="#">edit</a> <a href="#">delete</a>
3	jennissellenA	111	9748969388	112	3	2023-03-21	<a href="#">edit</a> <a href="#">delete</a>
4	jennissellen	20	6369868329	100	5		<a href="#">edit</a> <a href="#">delete</a>
Sno.	Student Name	Reg no	Contact no	Room no	Seater	Staying From	Action

Showing 1 to 4 of 4 entries

PREVIOUS 1 NEXT

## 7.5 ERROR MESSAGES

- This error message will appear If the username or password is incorrect.



## **8. CONCLUSION**

### **8.1 SUMMARY OF THE PROJECT**

So the Farming Management System is an e-commerce website which is used for farmers to add their products in their respective account and the buyers can buy them using their account. The application is also serve as a useful site to know what is going on the real world agricultural business, so the users can be benefited in various forms by using this e-commerce website.

This website can be further expanded by following the future enhancements mentioned below. I also have a plan to expand it from website to a mobile application. If it comes true, it would be a time saving one and it can increase the interaction between the user and the database. Everything in this project is simplified to its very bottom level. So it can not be a complicated one while interacting with our website. Its design is very simple so that the rural people can understand and use it for their purpose.

### **8.2 FUTURE DEVELOPMENT**

In the future, we can develop this website into an mobile application (for both android and apple ios if possible) with extra features. By converting this website into an application, we can overcome many of our disadvantages. We can get a Consistent user experience, user doesn't have to wait, rich functionality, personalization, scalability, security and useful in critical times.

# **BIBLIOGRAPHY**

## **BOOK REFERENCE**

- James Lee and Brent Lee “Open Source Development with LAMP Using Linux, Apache, MySQL, Perl and PHP”, Pearson Education, 2009.  
Book for Reference
- Jason Gerner, Elizabeth Naramore, Morgan Owens and Matt Warden,  
“Professional LAMP - Using Linux , Apache, My SQL and PHP5 Web development”, Wiley Publisher, 2006.

## **WEBSITE REFERENCE**

- [www.freetechbooks.com](http://www.freetechbooks.com)
- [www.slideshare.com](http://www.slideshare.com)
- [www.w3schools.com](http://www.w3schools.com)
- [www.programmersheaven.com](http://www.programmersheaven.com)
- [www.phpreferencebook.com](http://www.phpreferencebook.com)



## A. SOURCE CODE

### 1. Admin login

```
<?php
session_start();
include('includes/config.php');
if(isset($_POST['login']))
{
$username=$_POST['username'];
$password=$_POST['password'];
$stmt=mysqli->prepare("SELECT username,email,password,id FROM admin WHERE
(userName=? || email=?) and password=? ");
$stmt->bind_param('sss',$username,$username,$password);
$stmt->execute();
$stmt -> bind_result($username,$username,$password,$id);
$rs=$stmt->fetch();
$_SESSION['id']=$id;
$uip=$_SERVER['REMOTE_ADDR'];
$date=date('d/m/Y h:i:s', time());
if($rs)
{

header("location:dashboard.php");
}

else
{
echo "<script>alert('Invalid Username/Email or password');</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="">

<title>Admin login</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">
</head>
<body>

<div class="login-page bk-img" style="background-image: url(img/login-bg.jpg);">
<div class="form-content">
<div class="container">
<div class="row">
<div class="col-md-6 col-md-offset-3" style="margin-top:4%">
<h1 class="text-center text-bold text-light mt-4x">Hostel Management System</h1>
<div class="well row pt-2x pb-3x bk-light">
<div class="col-md-8 col-md-offset-2">

<form action="" class="mt" method="post">
<label for="" class="text-uppercase text-sm">Your Username or Email</label>
<input type="text" placeholder="Username" name="username" class="form-control mb">
<label for="" class="text-uppercase text-sm">Password</label>
<input type="password" placeholder="Password" name="password" class="form-control mb">

<input type="submit" name="login" class="btn btn-primary btn-block" value="login"
>
</form>
</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>
</html>

```

## 2. Add Course

```
<?php
session_start();
include('includes/config.php');
include('includes/checklogin.php');
check_login();
//code for add courses
if(isset($_POST['submit']))
{
    $coursecode=$_POST['cc'];
    $coursesn=$_POST['cns'];
    $coursefn=$_POST['cnf'];

    $query="insert into courses (course_code,course_sn,course_fn)
    values(?,?,?)";
    $stmt = $mysqli->prepare($query);
    $rc=$stmt->bind_param('sss',$coursecode,$coursesn,$coursefn);
    $stmt->execute();
    echo"<script>alert('Course has been added successfully');</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>Add Courses</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>
</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">Add Courses </h2>
<div class="row">
<div class="col-md-12">
<div class="panel panel-default">
<div class="panel-heading">Add courses</div>
<div class="panel-body">
<form method="post" class="form-horizontal">
<div class="hr-dashed"></div>
<div class="form-group">
<label class="col-sm-2 control-label">Course Code </label>
<div class="col-sm-8">
<input type="text" value="" name="cc" class="form-control"> </div>
</div>
<div class="form-group">
<label class="col-sm-2 control-label">Course Name (Short)</label>
<div class="col-sm-8">
<input type="text" class="form-control" name="cns" id="cns" value=""
required="required">
</div>
</div>
<div class="form-group">
<label class="col-sm-2 control-label">Course Name(Full)</label>
<div class="col-sm-8">
<input type="text" class="form-control" name="cnf" value="" >
</div>
</div>
<div class="col-sm-8 col-sm-offset-2">
<input class="btn btn-primary" type="submit" name="submit" value="Add
course">
</div>
</div>
</div>
</form>

```

```

</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>

</script>
</body>

</html>

```

### 3. Create room

```

<?php
session_start();
include('includes/config.php');
include('includes/checklogin.php');
check_login();
//code for add courses
if(isset($_POST['submit']))
{
    $seater=$_POST['seater'];
    $roomno=$_POST['rmno'];
    $fees=$_POST['fee'];
    $sql="SELECT room_no FROM rooms where room_no=?";
    $stmt1 = $mysqli->prepare($sql);
    $stmt1->bind_param('i',$roomno);
    $stmt1->execute();
    $stmt1->store_result();
    $row_cnt=$stmt1->num_rows;;

```

```

if($row_cnt>0)
{
echo"<script>alert('Room already exist');</script>";
}
else
{
$query="insert into rooms (seater,room_no,fees) values(?,?,?);";
$stmt = $mysqli->prepare($query);
$rc=$stmt->bind_param('iii',$seater,$roomno,$fees);
$stmt->execute();
echo"<script>alert('Room has been added successfully');</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>Create Room</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>

```

```

</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">Add a Room </h2>

<div class="row">
<div class="col-md-12">
<div class="panel panel-default">
<div class="panel-heading">Add a Room</div>
<div class="panel-body">
<?php if(isset($_POST['submit']))
{ ?>
<p style="color: red"><?php echo htmlentities($_SESSION['msg']); ?><?php
echo htmlentities($_SESSION['msg']=""); ?></p>
<?php } ?>
<form method="post" class="form-horizontal">

<div class="hr-dashed"></div>

<div class="form-group">
<label class="col-sm-2 control-label">Select Seater </label>
<div class="col-sm-8">
<Select name="seater" class="form-control" required>
<option value="">Select Seater</option>
<option value="1">Single Seater</option>
<option value="2">Two Seater</option>
<option value="3">Three Seater</option>
<option value="4">Four Seater</option>
<option value="5">Five Seater</option>
</Select>
</div>
</div>
<div class="form-group">

```

[illegible]



#### 4. Student Register

```
<?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'];
    $result="SELECT count(*) FROM userRegistration WHERE email=? ||
    regNo=?";
    $stmt = $mysqli->prepare($result);
    $stmt->bind_param('ss',$email,$regno);
    $stmt->execute();
    $stmt->bind_result($count);
    $stmt->fetch();
    $stmt->close();
    if($count>0)
    {
        echo"<script>alert('Registration number or email id already
        registered.');
```

```

<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>User Registration</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">
<div class="row">
<div class="col-md-12">
<h2 class="page-title">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" onBlur="checkRegnoAvailability()">
<span id="user-reg-availability" style="font-size:12px;"></span>
</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="reset">Reset</button>
<input type="submit" name="submit" Value="Register" 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>
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 ()
{
event.preventDefault();
alert('error');
}
});
}
</script>
<script>
function checkRegnoAvailability() {
$("#loaderIcon").show();
jQuery.ajax({
url: "check_availability.php",
data:'regno='+$("#regno").val(),
type: "POST",
success:function(data){

```

```

$("#user-reg-availability").html(data);
$("#loaderIcon").hide();
},
error:function ()
{
    event.preventDefault();
    alert('error');
}
});
}
</script>
</html>

```

## 5.Student Login

```

<?php
    session_start();

    include('includes/config.php');

    if(isset($_POST['login']))
    {
        $emailreg=$_POST['emailreg'];
        $password=$_POST['password'];

        $stmt=$mysqli->prepare("SELECT email,password,id FROM userregistration WHERE
        (email=? || regNo=?) and password=? ");

        $stmt->bind_param('sss',$emailreg,$emailreg,$password);

        $stmt->execute();

        $stmt -> bind_result($email,$password,$id);

        $rs=$stmt->fetch();

        $stmt->close();

        $_SESSION['id']=$id;

        $_SESSION['login']=$emailreg;

        $uip=$_SERVER['REMOTE_ADDR'];

        $ldate=date('d/m/Y h:i:s', time());
    }
}

```

```

if($rs)
{
    $uid=$_SESSION['id'];
    $uemail=$_SESSION['login'];
    $ip=$_SERVER['REMOTE_ADDR'];
    $geopluginURL='http://www.geoplugin.net/php.gp?ip='.$ip;
    $addrDetailsArr = unserialize(file_get_contents($geopluginURL));
    $city = $addrDetailsArr['geoplugin_city'];
    $country = $addrDetailsArr['geoplugin_countryName'];
    $log="insert into userLog(userId,userEmail,userIp,city,country)
values('$uid','$uemail','$ip','$city','$country')";
    $mysqli->query($log);
    if($log)
    {
        header("location:dashboard.php");
    }
}
else
{
    echo "<script>alert('Invalid Username/Email or password');</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>Student Hostel Registration</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">
<div class="row">
<div class="col-md-12">
<h2 class="page-title">User Login </h2>
<div class="row">
<div class="col-md-6 col-md-offset-3">
<div class="well row pt-2x pb-3x bk-light">
<div class="col-md-8 col-md-offset-2">

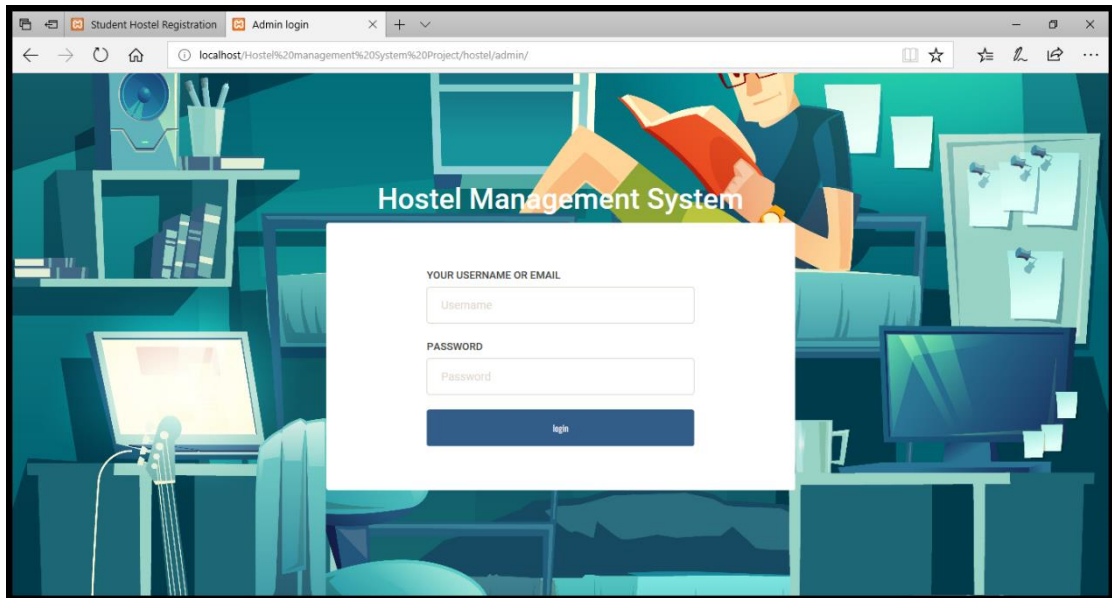
<form action="" class="mt" method="post">
<label for="" class="text-uppercase text-sm">Email / Registration Number</label>
<input type="text" placeholder="Email / Registration Number" name="emailreg"
class="form-control mb" required="true">
<label for="" class="text-uppercase text-sm">Password</label>
<input type="password" placeholder="Password" name="password" class="form-
control mb" required="true">
<input type="submit" name="login" class="btn btn-primary btn-block" value="login"
>
</form>
</div>
</div>
<div class="text-center text-light" style="color:black;">
<a href="forgot-password.php" style="color:black;">Forgot password?</a>
</div>
</div>

```

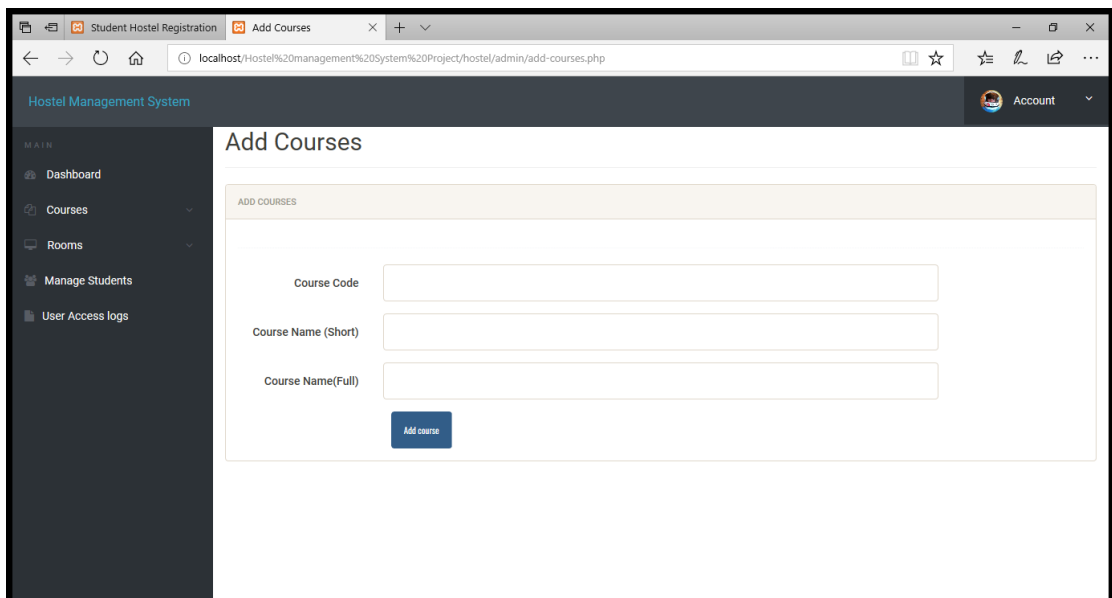
```
</div>
</div>
</div>
</div>
</div>
</div>
</div>
</div>
</div>
<script src="js/jquery.min.js"></script>
<script src="js/bootstrap-select.min.js"></script>
<script src="js/bootstrap.min.js"></script>
<script src="js/jquery.dataTables.min.js"></script>
<script src="js/dataTables.bootstrap.min.js"></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>
</html>
```

## B. Sample Output

### 1.Admin login page:



### 2.Add Course:



### 3.Add Rooms:

The screenshot shows a web browser window with the title 'Student Hostel Registration' and the page 'Create Room'. The URL is 'localhost/Hostel%20management%20System%20Project/hostel/admin/create-room.php'. The page has a dark sidebar with the 'Hostel Management System' logo and a menu with 'Dashboard', 'Courses', 'Rooms', 'Manage Students', and 'User Access logs'. The main content area is titled 'Add a Room' and contains a form with the following fields: 'Select Seater' (a dropdown menu), 'Room No.' (a text input), and 'Fee(Per Student)' (a text input). A blue 'Create Room' button is at the bottom of the form.

### 4.Student Register:

The screenshot shows a web browser window with the title 'Student Hostel Registration' and the page 'User Registration'. The URL is 'localhost/Hostel%20management%20System%20Project/hostel/registration.php'. The page has a dark sidebar with the 'Hostel Management System' logo and a menu with 'User Registration', 'User Login', and 'Admin Login'. The main content area is titled 'Student Registration' and contains a form with the following fields: 'Registration No.' (a text input), 'First Name' (a text input), 'Middle Name' (a text input), 'Last Name' (a text input), 'Gender' (a dropdown menu), 'Contact No.' (a text input), 'Email id' (a text input), and 'Password' (a text input). A blue header bar at the top of the form says 'FILL ALL INFO'.

## 5.Student Login:

The screenshot shows a web browser window with the URL `localhost/Hostel%20management%20System%20Project/hostel/index.php`. The page is titled "Hostel Management System" and features a dark sidebar with navigation links: "User Registration", "User Login", and "Admin Login". The main content area is titled "User Login" and contains a login form with the following fields:

- EMAIL / REGISTRATION NUMBER**: A text input field with placeholder text "Email / Registration Number".
- PASSWORD**: A text input field with placeholder text "Password".
- login**: A blue button to submit the login form.
- Forgot password?**: A link below the login button.

## 6.Book Hostel:

The screenshot shows a web browser window with the URL `localhost/Hostel%20management%20System%20Project/hostel/book-hostel.php`. The page is titled "Hostel Management System" and features a dark sidebar with navigation links: "Dashboard", "Book Hostel", "Room Details", "My Profile", and "Change Password". The main content area is titled "Registration" and contains a form with the following fields:

- FILL ALL INFO**: A header for the registration form.
- Room Related info**: A section header for room-related information.
- Room no.**: A dropdown menu with the placeholder text "Select Room".
- Seater**: A text input field.
- Fees Per Month**: A text input field.
- Food Status**: A radio button group with two options: "Without Food" (selected) and "With Food(Rs 2000.00 Per Month Extra)".
- Stay From**: A text input field with placeholder text "mm/dd/yyyy".
- Duration**: A dropdown menu with the placeholder text "Select Duration in Month".