

CHAPTER 1

INTRODUCTION

The Hostel Management System is developed for automating the activities of hostel. The software will be great relief to the employees. This software will help user in case of registration, updating or searching the information about the rooms available in the hostel. The aim of the Hostel Management System is to carry out the activities of Hostel in an efficient way. It will take the operations of hostel to an upper level by providing faster access to data and allowing addition, modification, and deletion of data in a very systematic and reliable manner.

1.1 Preamble

A preamble is an introductory statement in a document that explains the document's purpose and underlying philosophy. When applied to the opening paragraphs of a statute, it may recite historical facts pertinent to the subject of the statute. It includes the introduction, literature survey, motivation to the project, problem statement, objective of the statement, limitations and enhancements of the study and organization of the document.

1.2 Problem statement

The problem is to use a database where we can store all the information which was earlier stored manually. We link the front end and back end to retrieve information and display it to the customer. To insert values into the tables such as registration of students and the individual facilities chosen by them. It is also to cancel registration or delete the student details as and when required. And to update the values if any changes are made and to retrieve the needed information from the database.

1.3 Proposed solution

This system is designed in favor of the hostel management which helps them to save the records of the students who have opted for hostel about their rooms and other facilities. This project will reduce the annual work of the people in the admin panel and the bundle of registers that are to be searched to find any information about the students. Through this one can check the personal profile of all the current students within a few minutes using the data base. It saves them from the manual work from which it is very difficult to find the records of the students, the food status of the students, and the information of those who have left the hostel. Thus is it used for automating the activities of hostel. The software will be a great relief to the employees who would have had to otherwise write and keep track of all the information manually. The software will help user in case of reporting, registration and searching the information about residents and rooms. The aim of the Hostel Management System is to carry out the activities of Hostel in an efficient way. It will take the operation of Hostel to a higher level by providing faster access to data and allowing addition, updating, modification, and deletion of data in a very simple and systematic manner. This project helps automate each and every activity of the manual system which increases its throughput. It provides a way to get quick response with very accurate information as and when required.

CHAPTER 2

ANALYSIS AND SYSTEM REQUIREMENTS

2.1 Existing Scenario

- All the work is done manually. Different copies of the student information are kept for different departments.
- Room is allotted according to the room requirements and other special facilities demanded by the student.
- Room categories: Single and Double.
- Hostel facilities and charges and other information are all kept in book.
- Student's information, fee records, student check-in and check-out, room status and all these information are kept in a register.
- All calculations relating to student's fees, fines and penalties, hostel funds are done manually.

2.1.1 Drawbacks:

- The existing system is highly manual involving a lot of paper work and calculation and therefore may be erroneous. This has lead to inconsistency and inaccuracy in the maintenance of data.
- The data which is stored on the paper only, may be lost, stolen or destroyed due to natural calamity like fire or water.
- The existing system is sluggish and consumes a lot of time causing inconvenience to students and the employees.
- Due to manual nature, it I difficult to update, delete, add or view the data.
- Since the number of students have drastically increased therefore maintaining and retrieving detailed record of every student is extremely difficult.

2.1.2 Features of proposed system:

- Long-term storage of records.
- High Accuracy in calculations.
- Efficiency in modification, storing and retrieval of data.
- Inexpensive modifications in facilities and terms of organization.
- Utilization of time and workforce.

2.2 Hardware and Software Requirements

The hardware and software requirements are as follows:

2.2.1 Hardware Requirements:

The most common set of requirements defined by any operating system or software application is the physical computer resources.

Following are the various aspects of hardware requirements

- Main processor : Intel Pentium Dual Core
- RAM size : 512Mb
- Hard Disk : 500Mb

2.2.2 Software Requirements:

Software requirements deal with defining software requirements and pre-requisites that need to be installed on a computer to provide optimal functioning of a software.

These requirements or pre-requisites are generally not included in the software installation package and need to be installed separately before the software is installed.

Following are the various aspects of software requirements

- Operating system : Windows 7
- Software environment : HTML5, CSS3, JavaScript, PHP7 and My SQL
- Hosting Server : Apache2 Server
- Database Server : MySQL Server
- Browsers used : Google Chrome, Mozilla Firefox

CHAPTER 3

SYSTEM DESIGN AND MODELING

This chapter specifies the design description of "Hostel Database Management System". This phase gives a clear idea of system to be implemented as we understand more with diagrams or pictorial representation than the textual matter written describing the project.

The design description consists of preliminary design and detailed design. Preliminary design specifies the low-level design.

3.1 Preliminary design

The purpose of the designed is to plan the solution of a problem specified by the system requirements. This phase is the first step in moving from problem to solution domain. In other words, starting with what is needed design takes us to work how to satisfy the needs.

The design of the system is perhaps the most critical factor affecting the quality of the software and has a major impact on the later phases, particularly testing and maintenance. The preliminary design phase may also be known as conceptual design or architectural design. During this phase, the high-level design concept is created.

This design concept maybe expressed as block diagrams, design and architectural descriptions, sketches and/or behavioral hardware description language.

3.1.1 E-R Diagram:

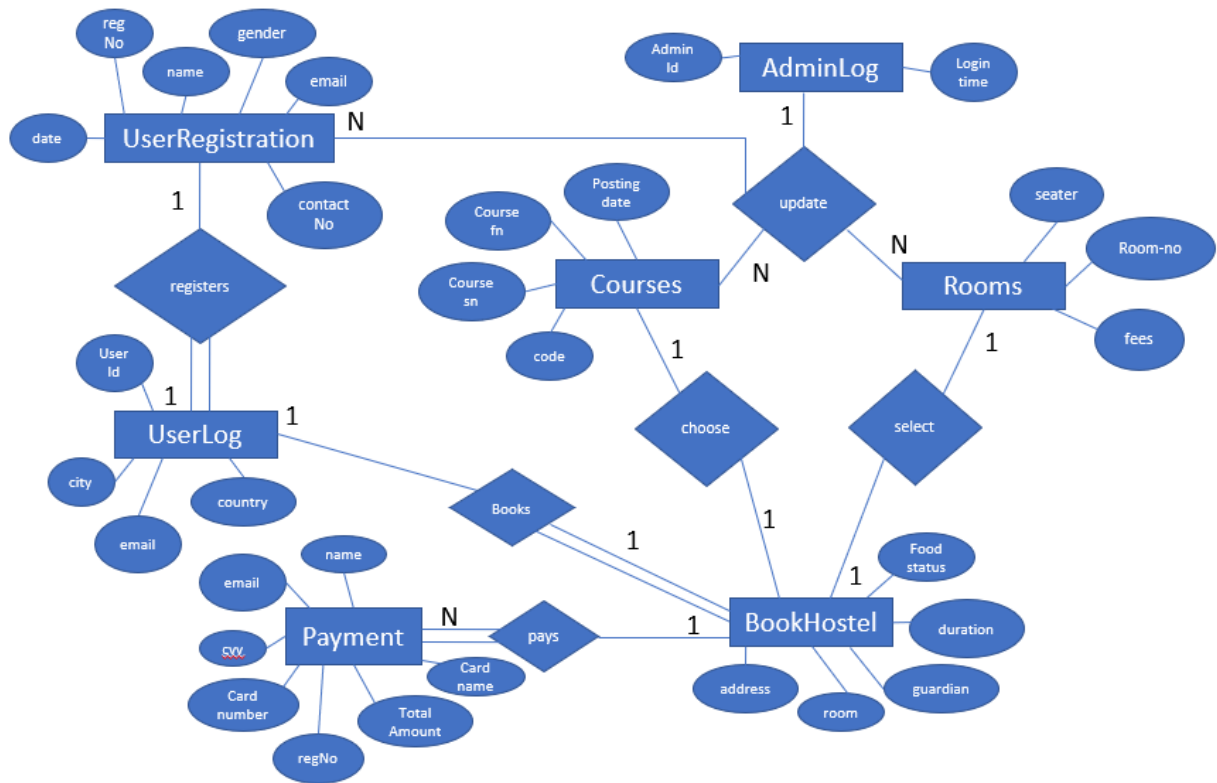


Fig: 3.1: ER Diagram of Hostel Management Database

It has seven entities: userRegistration, userLog, Courses, Rooms, BookHostel, Payment and AdminLog and six relationships: registers, booking, pays, choose, select and update. The userRegistration and userLog is related to with registers and has a cardinality ratio 1:1. userLog and BookHostel has dependent relation of booking having ratio 1: 1. BookHostel and Courses is related by choose with a ratio of 1:1. Courses and Rooms is related to AminLog by update relationship with ratio N:1. BookHostel entity has select relationship with Rooms by 1: 1 cardianality. BookHostel and Payment are related by a relation pays having cardinality 1: N.

3.1.2 Schema Diagram:

A schema is the structure behind data organization. It is a visual representation of how different table relationships enable the schema's underlying mission business for which the database is created.

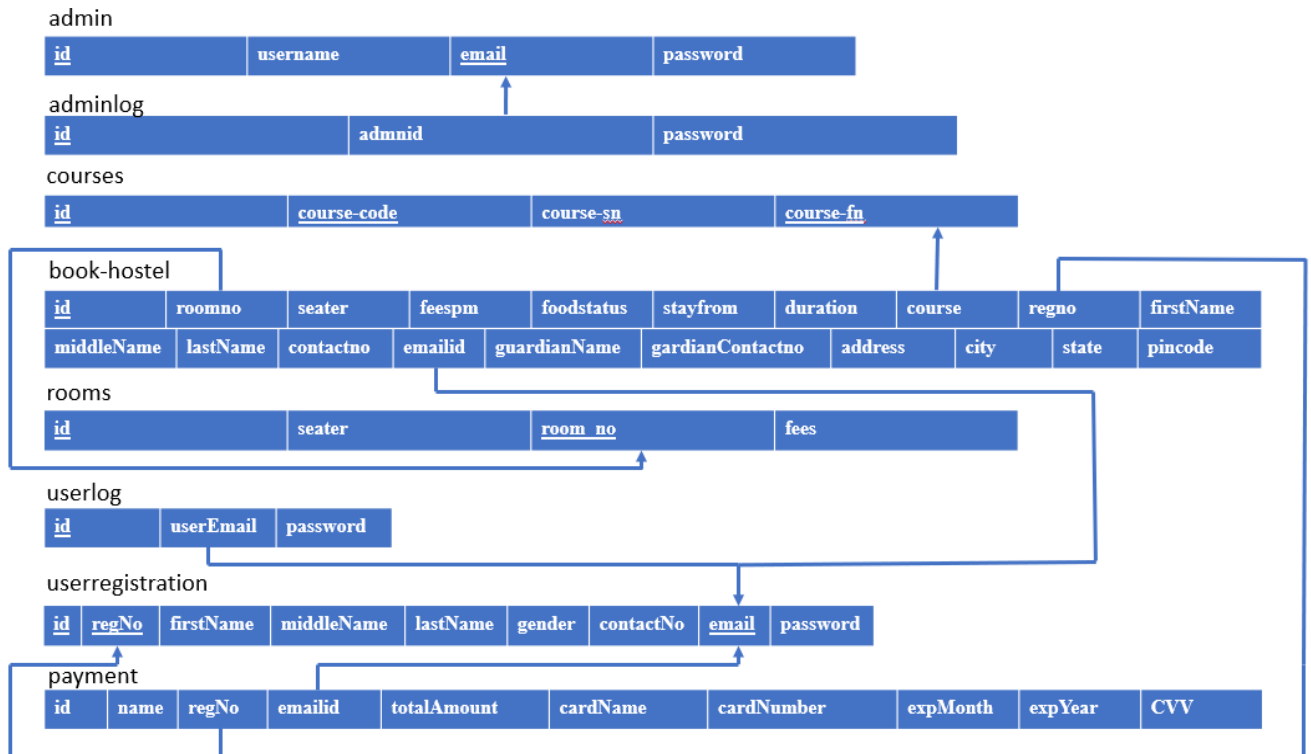


Fig: 3.2: Schema Diagram of Hostel Management Database

This is our schema diagram representing Hostel Database highlighting the primary keys, foreign keys along with the relationship of one table with another.

Consider three tables, **registration**, **userregistration** and **payment**. **userregistration** table has 'regNo' and 'email' as primary keys. The **userregistration** table is related to **registration** table using 'email' as the foreign key. It is also related to the **payment** table using 'regNo' as well as 'email' as foreign keys. And in turn **payment** is related to **registration** using 'regNo' as foreign key.

3.2 Normalization

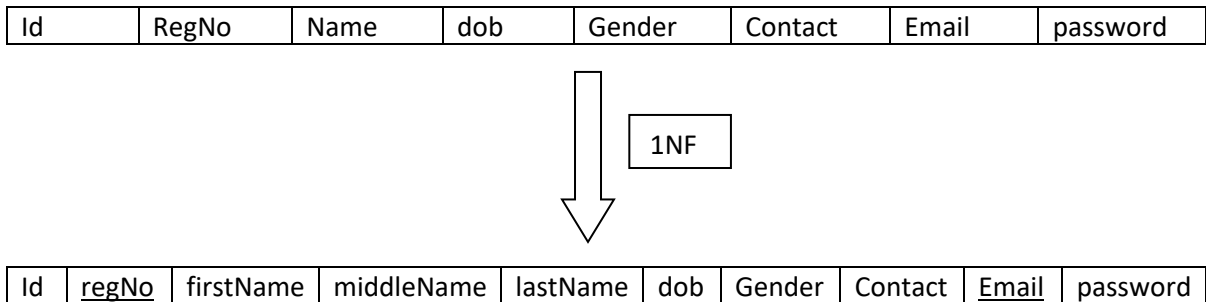
Normalization is a process of organizing the data in database to avoid data redundancy, insertion anomaly, update anomaly & deletion anomaly [2]

3.2.1. First normal form (1NF)

As per the rule of first normal form, an attribute (column) of a table cannot hold multiple values. It should hold only atomic values.

Example: **userregistration** table

Table 3.1 First Normal Form



The above table is now in first normal form.

3.2.2. Second normal form (2NF)

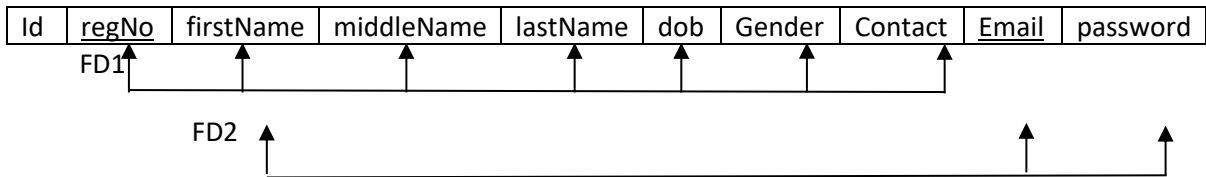
A table is said to be in 2NF if both the following conditions hold:

- Table is in 1NF (First normal form)
- No non-prime attribute is dependent on the proper subset of any candidate key of table.

An attribute that is not part of any candidate key is known as non-prime attribute.

Example: **userregistration** table

Table 3.2 Second Normal Form



FD1

<u>regNo</u>	firstName	MiddleName	lastName	dob	Gender	contact
--------------	-----------	------------	----------	-----	--------	---------

FD2

<u>Email</u>	firstName	password
--------------	-----------	----------

The above table is now in second normal form.

3.2.3. Third Normal form (3NF)

A table design is said to be in 3NF if both the following conditions hold:

- Table must be in 2NF
- Transitive functional dependency of non-prime attribute on any super key should be removed.
- An attribute that is not part of any candidate key is known as non-prime attribute.

Example: **userregistration** table

Table 3.3 Third Normal Form

Id	<u>regNo</u>	firstName	middleName	lastName	dob	Gender	Contact	<u>Email</u>	password
----	--------------	-----------	------------	----------	-----	--------	---------	--------------	----------

The above table is in third normal form.

CHAPTER 4

IMPLEMENTATION

4.1 Operations:

1. **Registering a New User:** Make sure you're eligible to open an account. Visit the bank and ask to open an account. Ask important questions before you finalize your account. Supply the necessary information to create your account.
2. **Book Room:** When you want to put more money into your bank accounts you need to make a deposit. This is by provide your account numbers (or aadhar number).
3. **Update Profile:** You will usually need your account number (or aadhar number)
4. **View Room Details and make Payment:** Look up your bank account closing procedure. Request verification that your account is closed and collect your cash.
5. **Change Password:** A passbook or bankbook is a paper book used to record bank transactions on a deposit account and withdraw account.

4.2 SQL statements:

1. Insert statement: The INSERT INTO statement is used to insert new records in a table.

The INSERT INTO syntax will be as follows:

```
INSERT INTO table_name VALUES (value1, value2, value3, ...);
```

The following SQL statement insert's a new record in the 'userregistration' table:

```
INSERT INTO userregistration VALUES
```

```
('1ox16cs011','Brinda','Narayan','Murthy','Female',95675667657, 'brinda@gmail.com');
```

[1]

2. Update statement: An SQL UPDATE statement changes the data of one or more records in a table. Either all the rows can be updated, or a subset may be chosen using a condition.

The UPDATE syntax would be as follows:

```
UPDATE table_name SET column_name = value [WHERE condition]
```

The following SQL statement update's a record in the "userregistration" table:

```
UPDATE amount set fname='Kiran' where regno='1ox16cs011'; [5]
```

3. Delete statement: The DELETE statement is used to delete existing records in a table. The DELETE syntax would be as follows:

```
DELETE FROM table_name WHERE condition;
```

The following SQL statement delete's a record in the "userregistration" table:

```
DELETE FROM userregistration WHERE regno='1ox16cs011'; [5]
```

4. Create statement: The CREATE TABLE Statement is used to create tables to store data. Integrity Constraints like primary key, unique key, foreign key can be defined for the columns while creating the table. The CREATE syntax would be as follows:

```
CREATETABLE table_name(column1 datatype,column2 datatype,column3 datatype,...  
columnN datatype, PRIMARY KEY( one or more columns ));
```

The following SQL statement creates a table "userregistration":

```
Create table userregistration (regno int(5) PRIMARY KEY, fname varchar(20), mname  
varchar(20), lname varchar(20), contactno bigint, email varchar(100), password varchar(20));
```

5. Trigger statement: A trigger is a special kind of stored procedure that automatically executes when an event occurs in the database server. The trigger syntax would be as follows:

```
CREATE TRIGGER trigger_name BEFORE INSERT OR UPDATE OF <parameters> ON  
<table_name> FOR EACH ROW SET <condition>;
```

The following SQL statement triggers records in the "registration" table:

```
CREATE TRIGGER setexpiry BEFORE INSERT ON registration FOR EACH ROW BEGIN  
SET NEW.expiry= DATE_ADD( NEW.stayfrom, INTERVAL NEW.duration MONTH);  
END [3]
```

6. Stored procedure: A stored procedure is a set of SQL statements with an assigned name, which are stored in a RDBMS as a group, so it can be reused and shared by multiple programs.

The stored procedure syntax would be as follows:

```
CREATE PROCEDURE procedure_name(input parameters, output parameters)AS BEGIN  
<sql statement used in stored procedure> END
```

The following SQL statement creates a procedure:

Delimiter //

```
CREATE PROCEDURE insertdata(IN name VARCHAR, IN email VARCHAR, IN message  
VARCHAR) BEGIN INSERT INTO feedback(name, email, message) VALUES(name, email,  
message);  
END // [3]
```

4.3 Algorithms

Algorithm to Create and Show Trigger “CAPITALIZE” in SQL server

1. START
2. CREATE TRIGGER CAPITALIZE
3. {BEFORE AND INSERT}
4. [ON PAYMENT]
5. [FOR EACH ROW]
6. [SET NEW.CARDNAME=UPPER (NEW.CARDNAME)]
7. SHOW TRIGGERS
8. END

Algorithm to Create and Show Trigger “SETEXPIRY” in SQL Server

1. START
2. CREATE TRIGGER SETEXPIRY
3. {BEFORE AND INSERT}
4. [ON REGISTRATION]
5. [FOR EACH ROW]
6. [SET NEW.EXPIRY=DATE_ADD (NEW.STAYFROM, INTERVAL NEW.DURATION MONTH)]
7. SHOW TRIGGERS
8. END

Algorithm to Create a Stored Procedure “INSRTDATA” in SQL server

1. START
- 2.1 Create procedure in MySQL
- 2.2 Declare DELIMITER //
- 2.3 CREATE PROCEDURE INSERTDATA
- 2.4 Pass parameter (IN name VARCHAR, IN email VARCHAR, IN message VARCHAR)
- 2.5 BEGIN
- 2.6 INSERT INTO feedback (name, email, message) VALUES (name, email, message);
- 2.7 END//
3. END

Algorithm to Create a Stored Procedure “FINDAGE” in SQL server

1. START

2.1 Create procedure in MySQL

2.2 Declare DELIMITER //

2.3 CREATE PROCEDURE FINDAGE

2.4 Pass parameter (IN dob DATE)

2.5 BEGIN

2.6 INSERT INTO userregistration (age) VALUES ((SELECT FLOOR (DATEDIFF (NOW
(), dob) / 365)));

2.7 END//

3. END

4.4 Flowchart of Hostel Management System

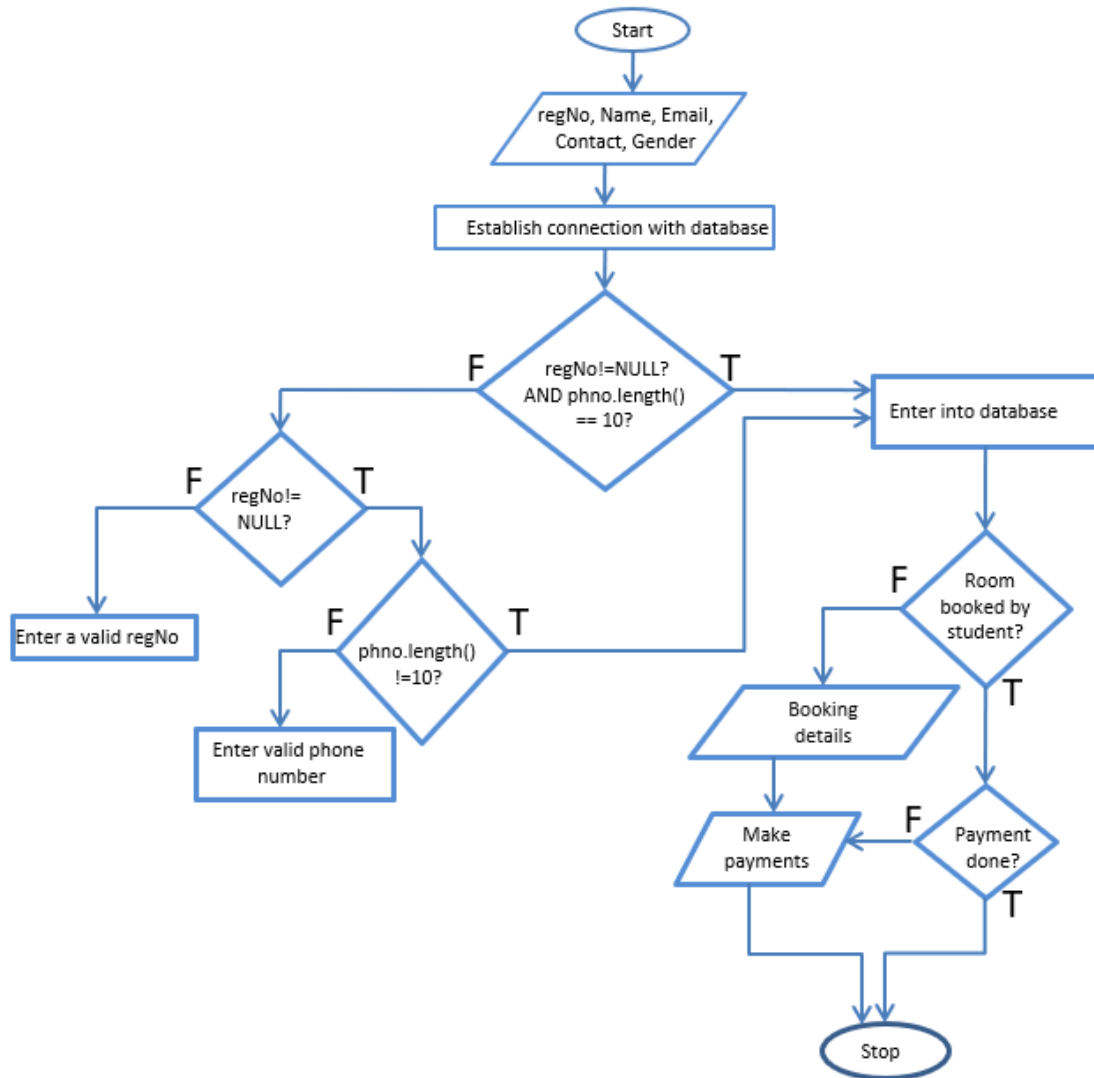


Figure 4.1 Flowchart of Operation

CHAPTER 5

TESTING

This chapter gives the outline of all the testing methods that are carried out to get a bug free

application. Quality can be achieved by testing the product using different techniques at different phases of the project development.

5.1 Testing process

Testing is an integral part of software development. Testing process, in a way certifies, whether the product, that is developed, complies with the standards, that it was designed to. Testing process involves building of test cases, against which, the product has to be tested. In some cases, test cases are done based on the system requirements specified for the product/software, which is to be developed.

5.2 Testing objectives

The main objectives of testing process are as follows:

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

5.3 Levels of Testing

Different levels of testing are used in the testing process; each level of testing aims to test different aspects of the system. The basic levels are unit testing, integration testing, system testing and acceptance testing.

5.3.1 Unit Testing

Unit testing focuses verification effort on the smallest unit of software design the module. The software built, is a collection of individual modules.

In this kind of testing exact flow of control for each module was verified. With detailed design consideration used as a guide, important control paths are tested to uncover errors within the boundary of the module.

Table 5.1: Negative test case for Input First Name

Function Name	Input	Expected Output	Error	Resolved
Registration	Brinda123 as First name	Expected output not seen	Numbers are being taken as input for name	Display an error message requesting only alphabets to be entered

Table 5.2: Positive test case for Input First Name

Function Name	Input	Expected Output	Error	Resolved
Create account	Brinda as First name	Expected output is seen	-	-

Table 5.3: Negative test case for Input Phone Number

Function Name	Input	Expected Output	Error	Resolved
Input phone number	9663487888abc	Expected output not seen	alphabets are being taken as input for phone number	Display an error message requesting only 10 digits to be entered

Table 5.4: Positive test case for Input Phone Number

Function Name	Input	Expected Output	Error	Resolved
Input phone number	9663487888	Expected output is seen	-	-

Table 5.5: Negative test case for Input Register No.

Function Name	Input	Expected Output	Error	Resolved
Register Number	1OX16 CS011	Expected output not seen	Space is being accepted.	Error message saying Register number should not contain spaces

Table 5.6: Positive test case for Input Register No.

Function Name	Input	Expected Output	Error	Resolved
Register number	1OX16CS011	Expected output is seen	-	-

5.3.2 Integration testing

The second level of testing is called integration testing. In this, many class-tested modules are combined into subsystems, which are then tested. The goal here is to see if all the modules can be integrated properly. We have identified all and debugged.

Table 5.7: Test case on basics of Checking Room Details

Function Name	Input	Expected Output	Error	Resolved
Negative-Checking for room details using regno.	1ox16 34	Must display that the reg. number is not valid	Output not seen	room_details(regnono)
Positive-Checking for room details using regno	1ox16cs011	Must display the room details	-	-

5.3.3 System testing

Here the entire application is tested. the reference document for this process is the requirement document, and the goal is to see IF the application meets its requirements. Each module and component of ethereal was thoroughly tested to remove bugs through a system testing strategy. Test cases were generated for all possible input sequences and the output was verified for its correctness.

Table 5.8: Test cases for the project

Steps	Action	Expected output
Step1:	The screen appears when the users runs the program	A page with different tabbed panes appears.
Step2: Selection1	The screen appears when the user selects any one of the tabbed panes from the click of the mouse. 1.Ameneties 2.Contact 3.Admin 4.User Registration 5.User Login 6.Queries	A window for viewing Hostel facilities with option '1', a window of contact details and Admin Login for the hostel with option '2' and '3' and a window for user to register and fill details with option '4' and option '5' for user to login to his profile and finally option '6' for displaying output of the queries
Step3: Selection2	The screen appears when the user logs in with respective email and password. 1.Update Profile 2.Book Room 3.Room Details & Payment 4.Change Password	The pages that facilitate the user to update his personal details and book a room in hostel and later view its details and make appropriate payment. He can also change the password if h wishes.

CONCLUSION

The application development is very flexible and much functionality can be added to it, to enhance performance of this project titled " HOSTEL DATABASE MANAGEMENT SYSTEM".

It was great challenge for us to use HTML, CSS and PHP for the development for our project. This hostel database management system is very useful for the hostel's management to insert the user information, their room details and their academic information.

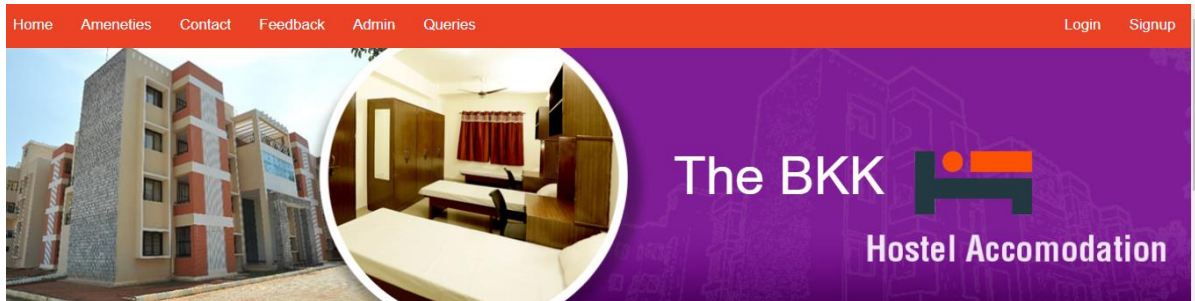
By using a variety of queries, we can determine various useful details about the hostel. This project helps the hostel management in many ways. This project has been developed for the management's and users convince. We have developed this project for various hostels to make use of. The system is very much user friendly and can be implemented with ease. Hence the database is easily implementable and very much user friendly. It is also easy to update and can adapt to any hostel requirements.

The system can be made better by adding more options and extending it further towards broader aspects. Our project becomes useful, as it is user-friendly and easy to understand. The project was successfully completed within the time durations.

REFERENCES

- [1] Fundamentals of database systems” 7th edition, Ramez Elmasri and Shamkant B. Navathe, Pearson Education, Asia
- [2] <http://www.youtube.com/watch?v=xoTyrdT9SZI&list=PLLGlmW7jT-nTr1ory9o2MgsOmmx2w8FB3>
- [3] https://www.w3schools.com/trigger_storedprocedure.doc
- [4] <https://www.stackoverflow.com/cssproperties.pdf>
- [5] <https://sqlite.org/basiccommands.doc>

Appendix A: Snapshots



Our Infrastructure

Hostels provide budget-oriented, sociable accommodation where students can register and book a bed. This new hostel building comprises of basement, ground and three upper floors which was completed in the year 2002 to accommodate around 200 students. Rooms with 2,3 and 4 sharing with attached toilet and bathrooms with internet facility. Our hostel along with out college is located in the centre of the city making it the best choice for the students. And the benefit – it's easy to find, so all the entertainment is within easy walking distance and you don't have to spend money on public transportation and hence not a problem to find food even at 4:00am.

Figure A.1: Home Page

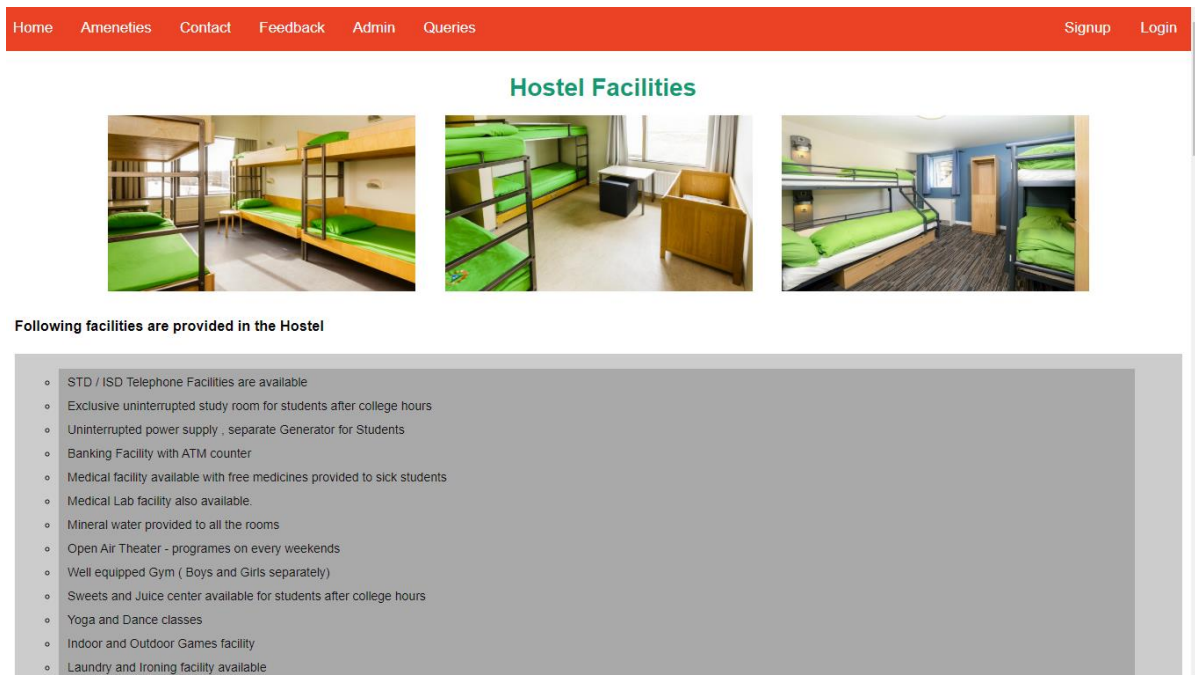


Figure A.2: Amenities Page

Home Ameneties Contact Feedback Admin Queries Login Signup

Feedback Form

Enter Name
Enter Your Name

Email address
Enter Email Address

Enter Your Message
Your feedback

Submit

TheBKKHostel 10th Milestone, Hosur Road Bangalore, India About the hostel
"A happy place for your ward to feel like"

Figure A.3: Feedback Page

Home Ameneties Contact Feedback Admin Queries Login Signup

General Queries Relating to Hostel Database

- List of Students who booked hostel between '2018-09-01' and '2018-12-31' and also view the expiry date of stay by using Triggers.
Get List
- List of Students who opted with and without Food.
With Food Get List
Without Food Get List
- List of Students who pay atleast 2000 more than, those who pay least amount for the hostel.

TheBKKHostel 10th Milestone, Hosur Road Bangalore, India About the hostel
"A happy place for your ward to feel like"

Figure A.4: Queries Page

Administrators

The Admin Login Portal displays three administrator cards, each with a name, ID, and a Login button:

- Brinda N Murthy** (ID: 10X16CS020) - Login button
- Kiran Saravana** (ID: 10X16CS035) - Login button
- Kishan Shetty** (ID: 10X16CS037) - Login button

Figure A.5: Admin Login Portal

The Admin Dashboard for BKK Hostel features a sidebar menu and a main dashboard area. The sidebar menu includes:

- Dashboard
- Students (dropdown)
- Manage Students
- Student Access Log
- Student Payment
- Rooms (dropdown)
- Add Rooms
- Manage Rooms
- Courses (dropdown)
- Add Courses
- Manage Courses
- Feedback Details

The main dashboard area displays three summary cards:

- 6 STUDENTS** - FULL DETAIL →
- 5 TOTAL ROOMS** - SEE ALL →
- 4 TOTAL COURSES** - SEE ALL →

Figure A.6: Admin Dashboard

BKK Hostel Logout

Menu

- Dashboard
- Students**
 - Manage Students
 - Student Access Log
 - Student Payment
- Rooms
- Courses
- Feedback Details

Manage Students

ALL ROOM DETAILS

Show 10 entries Search:

Sno.	Student Name	Reg no	Contact no	room no	Seater	Staying From	Action
1	brinda narayan murthy	1529555	123456	201	2	2018-10-08	edit delete
2	arun totad kumar	97	6449	600	5	2018-10-10	edit delete
3	power puff girls	8	456789	132	5	2018-10-11	edit delete
4	mahider akepatti reddy	1	8768657886	200	2	2018-10-18	edit delete
5	micky mouse mini	34567	789456123	132	5	2018-05-08	edit delete
6	Zaid Abdul Jabbar	1	3456787654	200	2	2018-11-06	edit delete

Showing 1 to 6 of 6 entries

PREVIOUS 1 NEXT

Figure A.7: Manage Students

BKK Hostel Logout

Menu

- Dashboard
- Students
- Rooms**
 - Add Rooms
 - Manage Rooms
- Courses
- Feedback Details

Manage Rooms

ALL ROOM DETAILS

Show 10 entries Search:

Sno.	Seater	Room No.	Fees (PM)	Posting Date	Action
1	2	201	6000	2016-04-12 01:30:47	edit delete
2	2	200	6000	2016-04-12 01:30:58	edit delete
3	3	112	4000	2016-04-12 01:31:07	edit delete
4	5	132	2000	2016-04-12 01:31:15	edit delete
5	5	600	5000	2018-10-21 11:56:33	edit delete

Showing 1 to 5 of 5 entries

PREVIOUS 1 NEXT

Figure A.8: Manage Rooms

The screenshot shows a web browser window with a dark blue header containing navigation links: Home, Ameneties, Contact, Feedback, Admin, Queries, Signup, and Login. The main content area is dark blue and features a central white registration form titled "Registration" with a document icon. The form includes the following fields: Register Number, First Name, Middle Name, Last Name, Date of Birth, a dropdown menu for "Select Gender", Contact, and Email Address.

Figure A.9: Student Registration

The screenshot shows a web browser window with a dark blue header containing navigation links: Home, Ameneties, Contact, Feedback, Admin, Queries, Signup, and Login. The main content area is dark blue and features a central white login form titled "User Login" with a user icon. The form includes the following fields: Email, Password, a "Submit" button, and links for "Forgot Password?" and "New User".

Figure A.10: Student Login

BKK Hostel Logout

Zaid's Profile

LAST UPDATION DATE :

Registration No : 1ox16cs011

First Name : Zaid

Middle Name : Abdul

Last Name : Jabbar

Gender : Male

Contact No : 3456787654

Email id : zaid@yahoo.com

Update Profile

Figure A.11: Update Profile

BKK Hostel Logout

Registration

FILL ALL INFO

Hostel Already Booked By You
[Click here to view your Room details and Payment information](#)

Room Related Info

Room no. Select Room

Seater

Fees Per Month

Check Availability

Food Status Without Food With Food(Rs 2000.00 Per Month Extra)

Stay From mm/dd/yyyy

Figure A.12 Booking a Room

BKK Hostel Logout

Room Details Make Payment

ALL ROOM DETAILS

Room Realited Info				Print Data	
Reg date :2018-11-12 17:05:22					
Room no :	200	Seater :	2	Fees PM :	6000
Food Status:	With Food	Stay From :	2018-11-06	Duration:	10 Months
Expiry :	2019-09-06 Total Fee : 80000				
Personal Info Info					
Reg No. :	1	Full Name :	ZaidAbdulJabbar	Email :	zaid@yahoo.com
Contact No. :	3456787654	Gender :	Male	Course :	Bachelor of Technology
Emergency Contact No. :	6789678906	Guardian Name :	Brinda	Guardian Relation :	Sister
Guardian Contact No. :	6789789789				
Addresses					

Figure A.13: Room Details after Booking

Registration

1ox16cs 037|

Please match the requested format.
Spaces are not allowed

Kishan

Shetty

02-12-1998

Male

8765787654

k@gmail.com

.....

.....

Figure A.14: Register No. validation for Registration Page