

INTRODUCTION

1.1 PROJECT PROFILE

The “HOME SERVE HANDYMAN SERVICES” system has been developed to override the problems prevailing in the manual system. The Home serve handyman services project related to job at home, when a person want a electrician, plumber , etc... for his home, they can book through this website in a smooth and effective manner. It serves as a platform where you can hire professionals for all your household chores at your fingertips.

Mainly this is a Employee work management system. This system is very useful for everyone who wants to offer home services because now a days everyone wants to save time and shot out their problems, therefore this system is very useful for peoples. This system helps both employees and customers. This system provides the employee with an income and customer get an easy and convenient service. All professionals will be from your local area and verified by admin to assure the quality of the work. Professionals arrives on chosen date and time to take care of your chores.

There are 3 users:

- Admin
- Customer
- Employee

1.2 ORGANIZATION OVERVIEW

D'Soft Solutions is a Software Development and Research Centre of De Paul Education Trust, a registered charitable Trust of the Vincentian fathers who are committed to the integral development of the less privileged members' of the society. It has long years of sound experience in software development field.

D'Soft Solutions has the prime objective of supplying efficient software solutions to various enterprises to make the best use of the offerings of Information Technology. It is organized in such a way as to provide customized and tailored solutions to specific industries, business concerns and other service institutions. The qualified and experienced personnel will provide training for operating the software and our after sale service is guaranteed.

SYSTEM SPECIFICATION

2.1 RECOMMENDED SOFTWARE SPECIFICATION

Operating System : Windows 10

Runtime framework : PHP 7.2.14

Front End : PHP, HTML, JavaScript

Back End : MySQL

Server : Apache

Web browser : Microsoft Edge or any compatible browser

2.2 RECOMMENDED HARDWARE SPECIFICATION

Processor : Intel Pentium 5 with 3GHz or higher memory

RAM : 8.00 GB

Hard Disk : 1 TB

Mouse : Standard Mouse

Keyboard : Logitech Keyboard

DEVELOPING TOOL

3.1 FRONT END

3.1.1 OVERVIEW OF PHP

PHP (recursive acronym for PHP: Hypertext Pre-processor) is a widely-used open source general-purpose scripting language that is especially suited for web development and can be embedded into HTML.

PHP is a server-side scripting language designed for web development but also used as a general-purpose programming language. Originally created by Rasmus Lerdorf in 1994, the PHP reference implementation is now produced by The PHP Group. PHP originally stood for Personal Home Page, but it now stands for the recursive backronym PHP: Hypertext Pre-processor.

PHP code may be embedded into HTML code, or it can be used in combination with various web template systems, web content management system and web frameworks. PHP code is usually processed by a PHP interpreter implemented as a module in the web server or as a Common Gateway Interface (CGI) executable. The web server combines the results of the interpreted and executed PHP code, which may be any type of data, including images, with the generated web page. PHP code may also be executed with a command-line interface (CLI) and can be used to implement standalone graphical applications.

The standard PHP interpreter, powered by the Zend Engine, is free software released under the PHP License. PHP has been widely ported and can be deployed on most web servers on almost every operating system and platform, free of charge. The PHP language evolved without a written formal specification or standard until 2014, leaving the canonical PHP interpreter as a de facto standard. Since 2014 work has been on-going to create a formal PHP specification.

3.1.2 WAMP SERVER

WAMP stands for web development and internal testing, but may also be used to serve live websites. Stands for "Windows, Apache, MySQL, and PHP." WAMP is a variation of LAMP for Windows systems and is often installed as a software bundle (Apache, MySQL, and PHP). It is often used

The most important part of the WAMP package is Apache (or "Apache HTTP Server") which is used run the web server within Windows. By running a local Apache web server on a Windows machine, a web developer can test webpages in a web browser without publishing them live on the Internet.

WAMP also includes MySQL and PHP, which are two of the most common technologies used for creating dynamic websites. MySQL is a high-speed database, while PHP is a scripting language that can be used to access data from the database. By installing these two components locally, a developer can build and test a dynamic website before publishing it to a public web server.

While Apache, MySQL, and PHP are open source components that can be installed individually, they are usually installed together. One popular package is called "WampServer," which provides a user-friendly way to install and configure the "AMP" components on Windows.

.

3.2 BACK END

3.2.1 MYSQL

MySQL is the world's most popular open source database. With its proven performance, reliability and ease-of-use, MySQL has become the leading database choice for web-based applications, used by high profile web properties including Facebook, Twitter, YouTube, Yahoo! and many more. Oracle drives MySQL innovation, delivering new capabilities to power next generation web, cloud, mobile and embedded applications.

MySQL AB was a software company that was founded in 1995. It was acquired by Sun Microsystems in 2008; Sun was in turn acquired by Oracle Corporation in 2010. MySQL AB is the creator of MySQL, a relational database management system, as well as related products such as MySQL Cluster. The company was dually headquartered in Uppsala, Sweden and Cupertino, California with offices in other countries (Paris, Munich, Dublin, Milan, and Tokyo).

With around 400 employees in 25 countries, MySQL AB was one of the largest open source companies worldwide. Around 70% of the employees worked for MySQL from their home offices.

MySQL is an open source relational database management system (RDBMS) based on Structured Query Language (SQL). MySQL runs on virtually all platforms, including Linux, UNIX, and Windows. Although it can be used in a wide range of applications, MySQL is most often associated with web-based applications and online publishing and is an important component of an open source enterprise stack called LAMP. LAMP is a Web development platform that uses Linux as the operating system, Apache as the Web server, and MySQL as the relational database management system and PHP as the object-oriented scripting language. (Sometimes Perl or Python is used instead of PHP). MySQL, which was originally conceived by the Swedish company MySQL AB, was acquired by Oracle in 2008. Developers can still use MySQL under the GNU General Public License (GPL), but enterprises must obtain a commercial license from Oracle.

3.3 OPERATING SYSTEM

Windows 10

Windows 10 is an operating system produced by Microsoft for use on personal computers, including home and business desktops, laptops etc. Windows 10 was intended to be a more focused, incremental upgrade to the Windows line, with the goal of being compatible with applications and hardware with which Windows Vista was already compatible. Presentations given by Microsoft in 2008 focused on multi-touch support, a redesigned Windows shell with a new taskbar, referred to as the Super bar, a home networking system called Home Group, and performance improvement.

1. Speed

Even aside from incompatibilities and other issues that many people had with Vista, one of the most straightforward was speed – it just felt too sluggish compared to windows 8.1, even on pumped up hardware. Windows 10 brings a more responsive and sprightly feel and Microsoft has spent a lot of time and effort getting the Start Menu response just right. Microsoft has also recognized the need for improved desktop responsiveness, which gives the impression that the computer is responding to the user and that they are in control – something that was often lacking with Vista. You can also expect faster boot times. And the boot sequence is now not only prettier than it was with Vista, but it's speedier too.

2.Compatibility

In simple terms, compatibility on Windows 10 will be far better than it was with Vista. Many programs that individuals and companies used on Windows 7 did not work immediately and required updates, but with Windows 7 almost all applications that work on Vista should still run. In essence, Windows Vista has done most of the hard work for Windows 7 in this respect.

3. Lower hardware requirements

Vista gained a reputation for making even the beefiest hardware look rather ordinary. Windows 10, however, will run well on lower end hardware, making the transition from Window 8.1 less painful. Microsoft is even pushing Windows 10 for net books. This could provide a modern replacement for Windows 8.1, which has found a new lease of life as the OS of choice on net books, supplanting Linux.

SYSTEM ANALYSIS

4.1 INTRODUCTION

Information is a vital ingredient for the operations and management of any organization. Computers have become an essential part of organizational information processed. This project entitled “Home Serve Handyman Services” is developed for D'Soft Computer education, Angamaly. This project contains the following modules:-

- Admin
- Customer
- Employee

We have developed this project by using PHP as front end and MYSQL as back end. We hope that this system will make the task of user easier.

4.2 EXISTING SYSTEM

Existing System is more complex. When someone need aid with small but major household tasks, the trouble arises when service skilled persons are unavailable or the trusted providers are impossible to find who delivers consistently flawless service on instance. That is While using the existing system we are losing our valuable time, effort, cost and human resources. Proposed system can eliminate the drawbacks of the existing system to a great extent.

4.3 LIMITATION OF EXISTING SYSTEM

- Customer call an employee manually. It is time-consuming. Existing system is offline.
- There is no direct role for the higher officials.
- Chance for Damage the machine due to lack of attention.
- No guarantied service.
- The existing system may not provide service to the client due to the lack of availability of the service person.

4.4 FEASIBILITY STUDY

All projects are feasible when given unlimited resources and infinite time. It is both necessary and prudent to evaluate the feasibility of a project at the earliest possible time. A feasible study is not warranted for system in which economic justification is observed, technical risk is low, few legal problems are expected and no reasonable alternative exists. An estimate is made of whether the identified user needs may be satisfied using our recent software and hardware technologies. The study will decide if the Home serve Handyman services will be cost effective, from the business point of view and it can be developed in the existing budgetary. The feasibility study should be relatively sharp and quick. The gesture should inform the decision of whether to go ahead with a more detailed analysis.

Feasibility study may be documented as a separated report to higher officials of the top level management and can be included as appendices to the system specification. Feasibility and risk analysis is detailed in many worries. If there is project risk then the feasibility of producing the quality software is reduced. The study is done in three phases.

- Operational Feasibility
- Technical Feasibility
- Economical Feasibility

OPERATIONAL FEASIBILITY

The purpose of the operational feasibility study is to determine the whether the Home serve Handyman services system will be used if it is developed and installed. And whether there will be resistance from users that will undermine the possible application benefit. The first challenge was whether the system meets the organizational requirements. This is checked by the system requirement collected from the users and the management and the operational feasibility proved that the system is capable to meet its functional requirements.

The Home serve Handyman services is completely driven and user friendly. In Home serve Handyman services operational feasibility is dependent on human resources available for the project and involves projecting whether the system will

be used if it is developed and implemented. Operational feasibility is a measure of how Home serve Handyman services solves the problem and how it satisfies requirements identified in the requirement analysis phase of system development. This site is feasible with all aspects of operation.

TECHNICAL FEASIBILITY

The technical feasibility study is a study of function, performances and constraints and improve the ability to create an acceptable system. Technical feasibility is frequently the most difficult to achieve at the stage of product engineering process. The system must be evaluated from technical viewpoint first. The assessment of this feasibility must be based on the outline design of the system requirements in the terms of inputs, outputs program procedure and staffs. This project is said to be technically feasible. Technical feasibility centers on the existing computer systems and extend to which it can support the proposed system. This involves financial consideration to technical enhancements. This site is feasible with all aspects of technical.

ECONOMICAL FEASIBILITY

Economic analysis is the most frequently used method for evaluating the effectiveness of the Home serve Handyman services. It evaluates whether the Home serve Handyman services benefits greater than cost. The system is an effective one since the benefits of the software outweigh the cost incurred in installing it. It can be developed under optimal expenses with the available hardware and software. This site is economically feasible.

4.5 PROPOSED SYSTEM

Home serve Handyman services is very cost effective. This project can eliminate the drawbacks of the existing system to a great extent. The registration procedure is very simple in online and booking of handyman done through online instead of phone calls and also reduce the risk of hiring the unknown at your home. So, the registered service person will report or answer at your door step.

4.6 THE ADVANTAGES OF PROPOSED SYSTEM

- User friendly
- Time- consumption is less.
- A single click system describes booking highly skilled in-house professionals and gets your service done on time.
- Admin can control all employees.
- Customer can register a complaint against the service person.
- The proposed system provides the customers with a sure, fast and prompt service.

SYSTEM DESIGN

5.1 INTRODUCTION

System design is the solution to the creation of a new system. This phase is composed of several systems. This phase focuses on the detailed implementation of the feasible system. System design has two phases of development logical and physical design. During logical design phase the analyst describes inputs (sources), outputs (destinations), databases (data stores) and procedures (data flows) all in a format that meets the user's requirements. Design goes through the logical and physical stages of development. At an early stage in designing a new system, the system analyst must have a clear understanding of the objectives, which the design is aiming to fulfill. Second input data and master files (database) have to be designed to meet the requirements of the proposed output. The operational (processing) phases are handled through program construction and testing. The system design includes:

- Output design
- Database design
- Input design
- Form design
- Architectural design
- System modules

5.2 OUTPUT DESIGN

Computer output is the most important and direct information source to the user. Output design is a process that involves designing necessary outputs in the form of reports that should be given to the users according to the requirements. Efficient, intelligible output design should improve the system's relationship with the user and help in decision making. So, while designing output the following things are to be considered.

- Determine what information to present
- Arrange the presentation of information in an acceptable format
- Decide how to distribute the output to intended receipts
- **View Registration details**

Admin can view the registered employees and also customers along with their necessary details and then admin will verify the employees details.

- **View booked users**

Admin can view the booked users with their particular from date and to date also can view all the booked users. Employee can view his bookings from customers. Then he can generate accept or reject action.

- **View Cancelled users**

Admin can view the cancelled users.

- **View Tracking Details**

Customers can view their ordered book status either accept or reject.

- **View Complaint details**

Admin can view the complaints from customers.

- **View Review and Rating details**

Admin can view the reviews and ratings done by customers.

5.3 DATABASE DESIGN

Data Base design is the logical form of design of data storage in the form of records in a particular structure in the form of tables with fields which is not transparent to the normal user but it actually acts as the backbone of the system. As we know database is a collection of which helps the system to manage and store data is called database management system. Data base management system builds some form of constraints like integrity constraints, i.e., the primary key or unique key and referential integrity which help to keep data structure storage and access of data from tables efficiently and accurately and take necessary steps to concurrent access of data and avoid redundancy of data in tables by normalization criterions.

Normalization is the method of breaking down complex table structures into simple table structures by using certain rules thus reduce redundancy and inconsistency and disk space usage and thus increase the performance of the system or application which is directly linked to the database design and also solve the problems of anomalies. There are different forms of normalization, some are:

- First normal form (1NF)
- Second normal form (2NF)
- Third normal form (3NF)
- Boyce code normal form
- Forth normal form (4NF)
- Fifth normal form (5NF)

The database design of the new system is in Second normal form and every non-key attribute is functionally depend only on the primary key. The master and transaction tables and their structure are shown below.

5.3.1 Table Design

- Table Name: Job category
Table Description: Various Home services

Field	Datatype	Size	Constraints	Description
Job_id	Int	4	Primary key	Id of various jobs.
Job_title	Varchar	20	Not null	Title of jobs.
Job_desc	Varchar	50	Not null	Description about different jobs.
Job_image	Varchar	50	Not null	Image of jobs

- Table Name: Area
Table Description: Location Details

Field	Datatype	Size	Constraints	Description
Area_id	Int	2	Primary key	Id of area
Area_name	Varchar	20	Not null	Location
Area_image	Varchar	50	Not null	Location Map

- Table Name: Customer
Table Description: Customer Registration Details

Field	Datatype	Size	Constraints	Description
Cust_id	Int	5	Primary key	Id of the customer
Cust_firstname	Varchar	20	Not null	First name of the customer
Cust_lastname	Varchar	20	Not null	Last name of the customer
Cust_age	Int	2	Not null	Age of the customer
Cust_gender	Char	1	Not null	Gender of the customer
Cust_address	Varchar	50	Not null	Address of the customer
Cust_Email	Varchar	20	Not null	Email of customer
Password	Varchar	20	Not null	Password
Area_id	Int	2	Foreign key	Area_id is referenced from Area table
Mobile_no	Varchar	10	Not null	Phone number of the customer

- Table Name: Employee
Table Description: Registration Details of Employees

Field	Datatype	Size	Constraints	Description
Emp_id	Int	5	Primary key	Id of the employees
Emp_Firstname	Varchar	20	Not null	First name of employees
Emp_Lastname	Varchar	20	Not null	Last name of employees
Dob	Date		Not null	Date of birth of employees
Age	Int	2	Not null	Age of the employees
Gender	Char	1	Not null	Gender of the employees
Emp_address	Varchar	50	Not null	Address of employees
Area_id	Int	2	Foreign key	Area_id is referenced from Area table
City	Varchar	20	Not null	City of employees
Phone_no	Varchar	10	Not null	Phone of employees
Emp_email	Varchar	20	Not null	Email of employees
Emp_Password	Varchar	30	Not null	Password
Emp_Image	Varchar	50	Not null	Image of employees
Doj	Date		Not null	Date of joining
Job_id	Int	4	Foreign key	Job_id is referenced from Jobs table
Emp_desc	Varchar	30	Not null	Description about the employee
Emp_experience	Varchar	20	Not null	Experience about the employee
Emp_Avail	Varchar	20	Not null	Employee availability status
Status	Varchar	10	Not null	Admin can approve or Reject the employees

- Table Name: Leaves
Table Description: Leave requests of employee

Field	Datatype	Size	Constraints	Description
Leave_id	Int	10	Primary key	Id of leave
Emp_id	Int	5	Foreign key	Emp_id is referenced from Employee table.
Leave_desc	Varchar	30	Not null	Reason
From_date	Date		Not null	Leave starting date
To_date	Date		Not null	Leave ending date
Leave_status	Varchar	20	Not null	Status of leave (approve or reject)

- Table Name: Booking
Table Description: Booking requests of customer

Field	Datatype	Size	Constraints	Description
Book_id	Int	10	Primary key	Id of booking
Emp_id	Int	5	Foreign key	Emp_id is referenced from Employee Table
Cust_id	Int	5	Foreign key	cust_id is referenced from Customer Table
Book_date	Date		Not null	Booking date (current date)
Problem_desc	Varchar	20	Not null	Problem description
Requester_name	Varchar	20	Not null	Requester name
Address	Varchar	20	Not null	Address
City	Varchar	20	Not null	City
Area_id	Int	10	Foreign key	Area_id is referenced from Area table
Contact_no	Varchar	10	Not null	Phone number
From_date	Date		Not null	From date
To_date	Date		Not null	To date
Days	Int	10	Not null	No of days of booking
Cust_email	Varchar	20	Not null	Customer email
Book_status	Varchar	20	Not null	Status of booking if it is booked or cancel
Status	Varchar	20	Not null	Status made by employees like accept or reject or complete

- Table Name: Cancel
Table Description: cancel booking requests

Field	Datatype	Size	Constraints	Description
Cancel_id	Int	10	Primary key	Id of cancellation
Book_id	Int	10	Foreign key	Book_id is referenced from Booking Table
Cancel_date	Date		Not null	Date of cancellation

- Table Name: Complaint
Table Description: Complaint registration done by customers

Field	Datatype	Size	Constraints	Description
Complaint_id	Int	10	Primary key	Id of complaint
Cust_id	Int	5	Foreign key	Cust_id is referenced from Customer table
Emp_id	Int	5	Foreign key	Emp_id is referenced from Employee table
Book_id	int	10	Foreign key	Book_id is referenced from Booking table
Complaint_desc	Varchar	50	Not null	Describe the complaint
Complaint_date	Date		Not null	Complaint date
Reply	Varchar	50	Not null	Reply for complaints
Complaint_status	Varchar	20	Not null	Status of complaint (apply or reply)

- Table Name: Review
Table Description: Review Registration done by customers

Field	Datatype	Size	Constraints	Description
Review_id	Int	10	Primary key	Id of review
Emp_id	Int	10	Foreign key	Emp_id is referenced from Employee table
Cust_id	Int	5	Foreign key	Cust_id is referenced from Customer table
Comment	Varchar	50	Not null	Reviews done by customers
Rate_1	Int	5	Not null	Rating for overall performance
Rate_2	Int	5	Not null	Rating for particular employee
Re_date	Date		Not null	Current date

5.4 INPUT DESIGN

The input design is the link between the information system and the user. It comprises the developing specification and procedures for data preparation and those steps are necessary to put transaction data into a usable form for processing data entry. The activity of putting data into the computer for processing can be achieved by inspecting the computer to read data from a written or printed document or it can occur by having people keying the data directly into the system.

- What data should be given as input?
- The dialogue to guide the operating personnel in providing input.
- Methods for preparing input validations and steps to follow when error Occur.
- **Login**

This feature used by the all the users to login into system. Users required username and password.

- **Services Details**

Admin add the home services details. It includes the name of services and description about service and image.

- **Location Details**

Admin add the area details such as area name, area image etc.

- **Registration Details**

Registration phase used by employee and customer. Customers should add his personal details and employee adds personal and work-experience details.

- **Leave Details**

In this phase employee can generate a leave requests.

- **Booking Details**

In this phase customers can generate the booking requests.

- **Complaint Details**

This phase is used by customers and here require complaint description, date, etc...

- **Review Details**

This phase is used by customers and here require review comment, ratings, date, etc..

5.5 FORM DESIGN

ADMIN

- Login Details: Admin can enter the username and password.
- Service Details: Admin can add, view, edit and delete the home services.
- Area Details: Admin can add, view, edit and delete the location details.
- Employee Details: Admin can view the registered employees, then admin can approve or reject the employees.
- Leave Details: Admin can view the leave requests from employees. Then he can verify it.
- Complaint Details: Admin can view the complaints from customer, then generate reply.
- Review Details: Admin can view the reviews from the customers .
- Report: Admin can generate reports such as booking reports, customer reports, employee reports, etc...

CUSTOMER

- Registration Details: customer can enter the details such as name, email, password for the registration form.
- Login Details: Customers can enter the email and password.
- Booking Details: Customers can book the available employee and the booking form contains problem description, address, date of bookings, etc...
- Cancel Details: Customers can cancel the bookings.
- Book Status Details: Customers can track the bookings.
- Complaint Details: Customers can do the complaints against a employee.
- Review Details: Customers can generate a feedback about the system and employees.

EMPLOYEE

- Registration Details: Employee can enter the details such as name, email, password and also his work-experience details for the registration form.
- Login Details: Employee can enter the email and password.
- Booking Details: Employee can view his bookings from employee. Then he can generate accept or reject action.
- Leave Details: Employee can generate a leave request form.

5.6 SYSTEM MODULES

A software system is always divided into several sub systems that makes it easier for the development. A software system that is structured into several subsystems makes it easy for the development and testing. The different subsystems are known as the modules and the process of dividing an entire system into subsystems is known as modularization or decomposition. The system under consideration has been divided into several modules taking in consideration the above-mentioned criteria. Modules are:

➤ ADMIN MODULE

- Login
Login with their Username and Password
- Set the jobs details
Admin can add and edit job details
- Set the location details.
Admin can add and edit location details
- Verification of registered employees.
- View all Booking Details.
- Approval of leave request from employees.
- View complaints from Customers.
- View Review and Rating from Customers
- Logout

➤ **CUSTOMER MODULE**

- **REGISTER**
Customer can register themselves.
- **Login**
Login with their Username and Password
- Search available employees in his area for home services.
- Add booking
- View their own bookings Status
- Cancel booking
- Sent complaints
- Make review and rating
- Logout

➤ **EMPLOYEE MODULE**

- **REGISTER**
Employee can register themselves ,Employee registration is verified by admin
- **Login**
Login with their Username and Password.
- Employees can upload their experience and Multi-skills and other details.
- View their own bookings From Customers
- Update their work status.
- Apply for leave
- Logout

DESIGN TOOL

6.1 UML Diagrams

UML stands for Unified Modeling Language. UML is a language for specifying, visualizing and documenting the system. This is the step while developing any product after analysis. The goal from this is to produce a model of the entities involved in the project which later need to be built. The representation of the entities that are to be used in the product being developed need to be designed. Software design is a process that gradually changes as various new, better and more complete methods with a broader understanding of the whole problem in general come into existence. There are various kinds of methods in software design. They are as follows:

- Use case diagram
- Activity diagram
- Sequence diagram
- Class diagram

Use case Diagrams:

Use case diagrams model behavior within a system and helps the developers understand of what the user require. The stick man represents what's called an actor. An actor represents an outside entity- either human or technological. Use case diagrams can be useful for getting an overall view of the system and clarifying who can do and more importantly what they can't do. Use case Diagram consists of use cases and actors and shows the interaction between the use case and actors. The purpose is to show the interactions between use cases and actor. To represent the system requirements from user's perspective. It must be remembered that the use-cases are the functions that are to be performed in the module. An actor could be the end-user of the system or an external system.

Activity Diagram:

The purpose is to show the activities which the users performed. Activities are shown parallel and sequentially in which order they are performed. Some activities are joined and split according to its flow. Flow of data is represented using arrows.

Sequence Diagram:

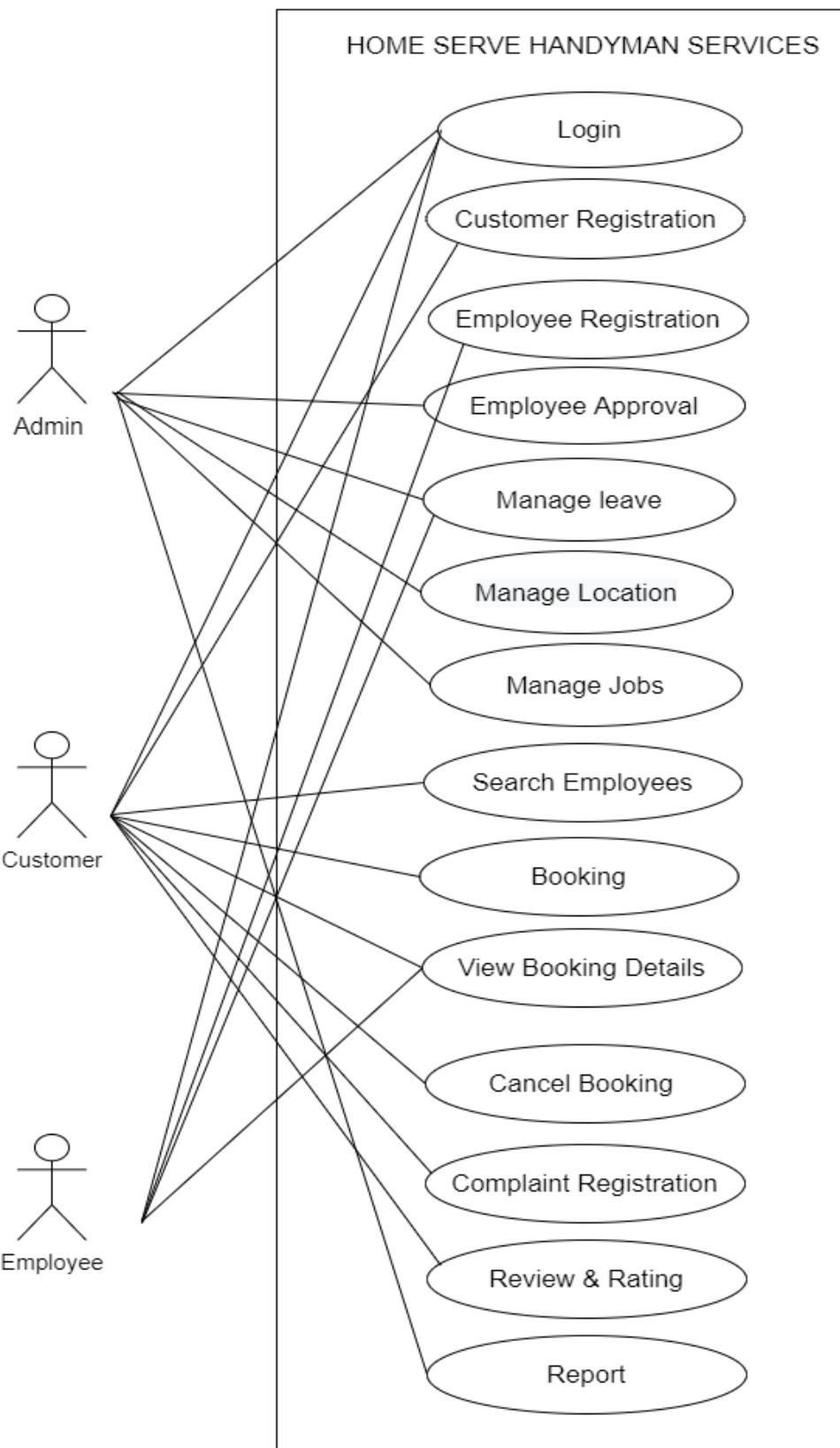
The purpose is to show the sequential flow through of activities. In other words, we call it mapping processes in terms of data transfers from the actor through corresponding objects. To represent the logical flow of data with respect to a process. It must be remembered that the sequence diagram displays objects and not the classes.

Class Diagram:

This is one of the most important of the diagrams in development. The diagram breaks the class into three layers. One has the name, the second describes its attributes and the third its methods. The private attributes are represented by a padlock to left of the name. The relationships are drawn between the classes. Developers use the Class Diagram to develop the classes. Analysts use it to show the details of the system. Architects look at class diagrams to see if any class has too many functions and see if they are required to be split.

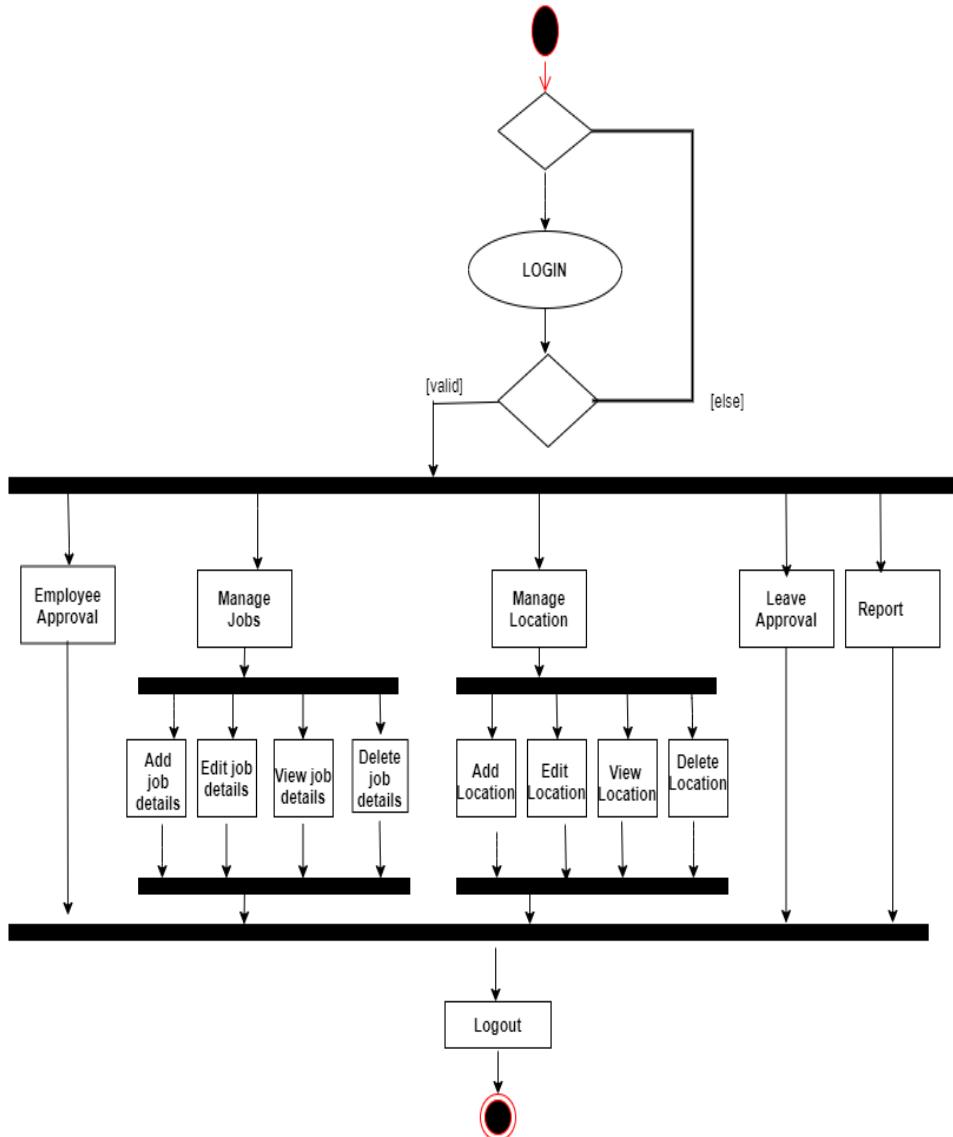
6.1.1 USECASE DIAGRAM:

use case diagram

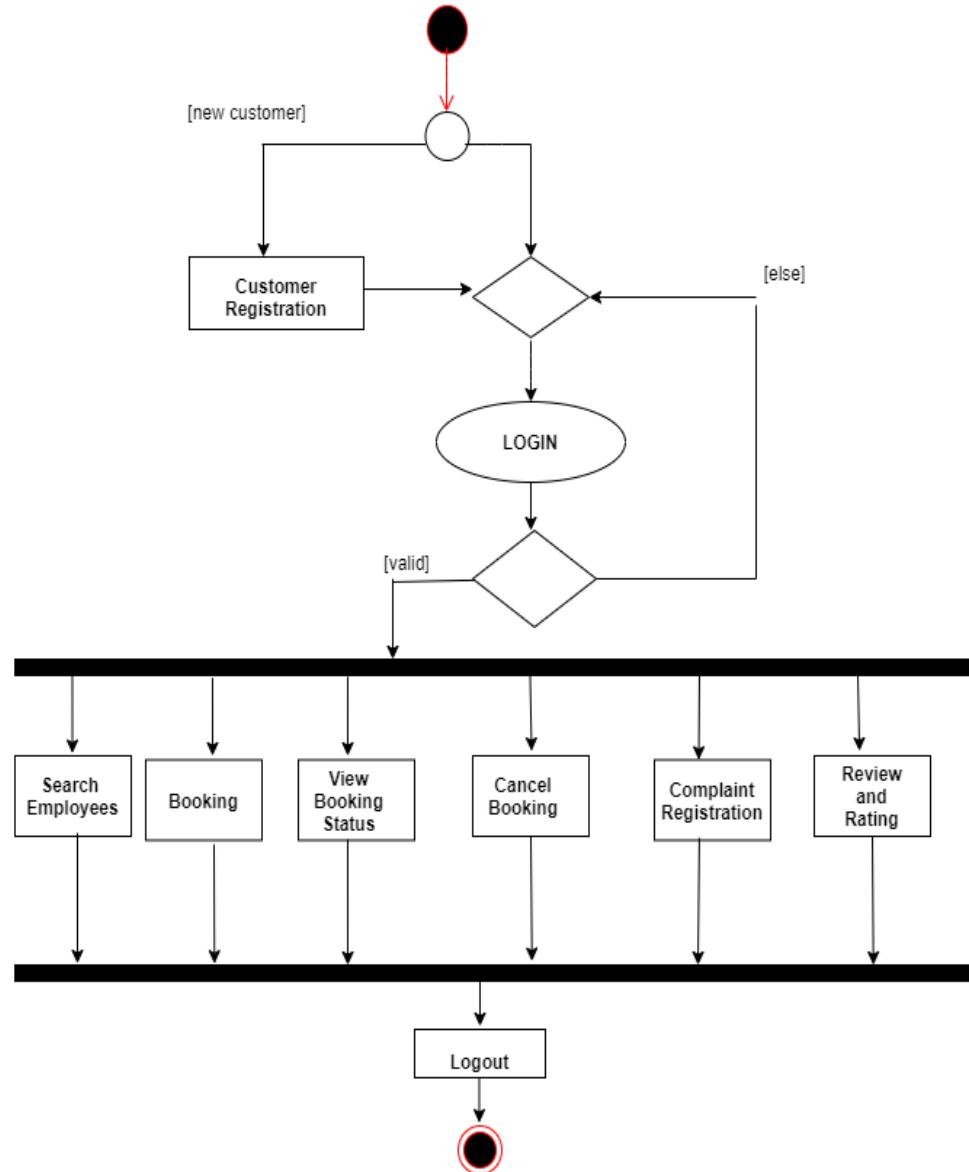


6.1.2 ACTIVITY DIAGRAM

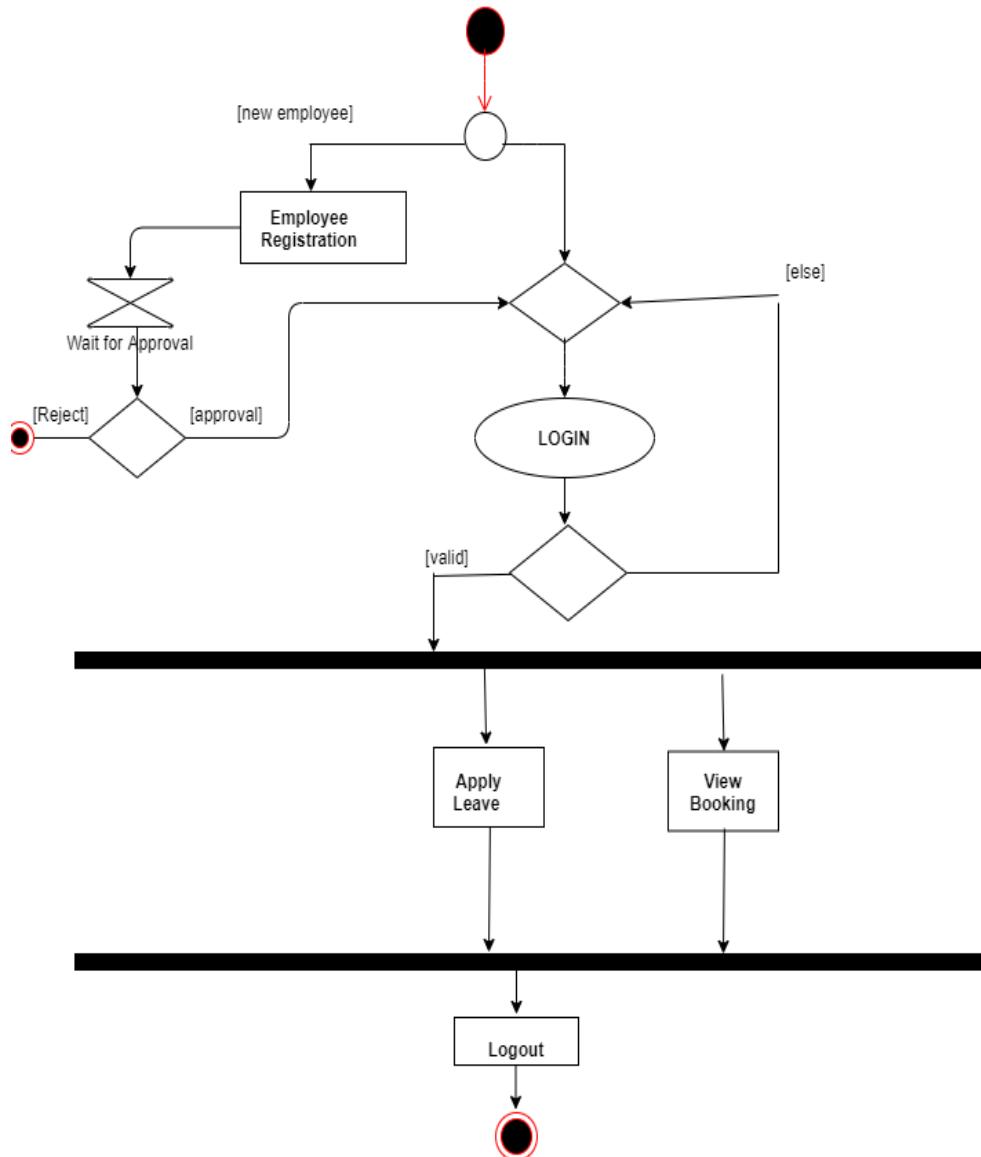
a) ADMIN



b) CUSTOMER

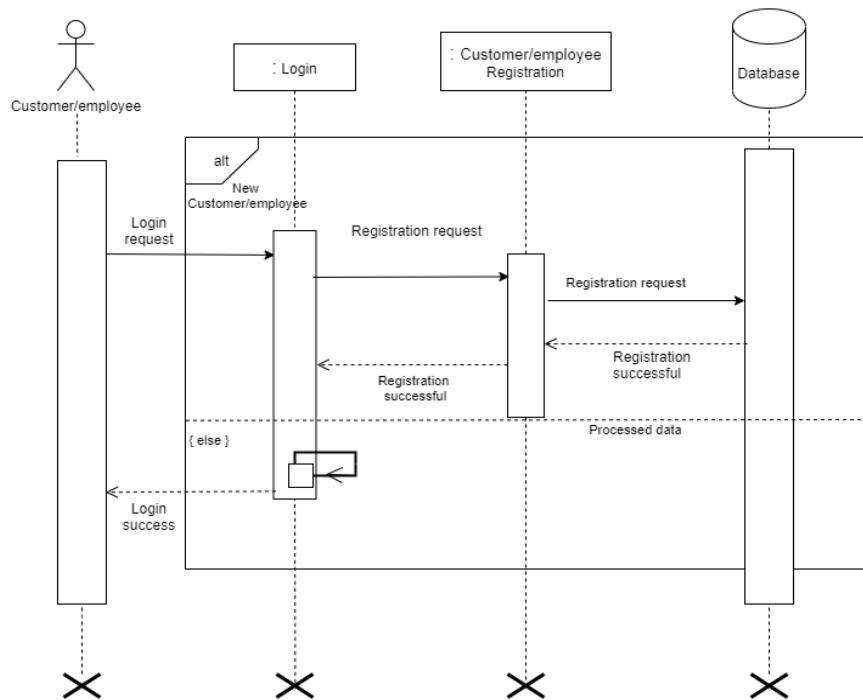


c) EMPLOYEE

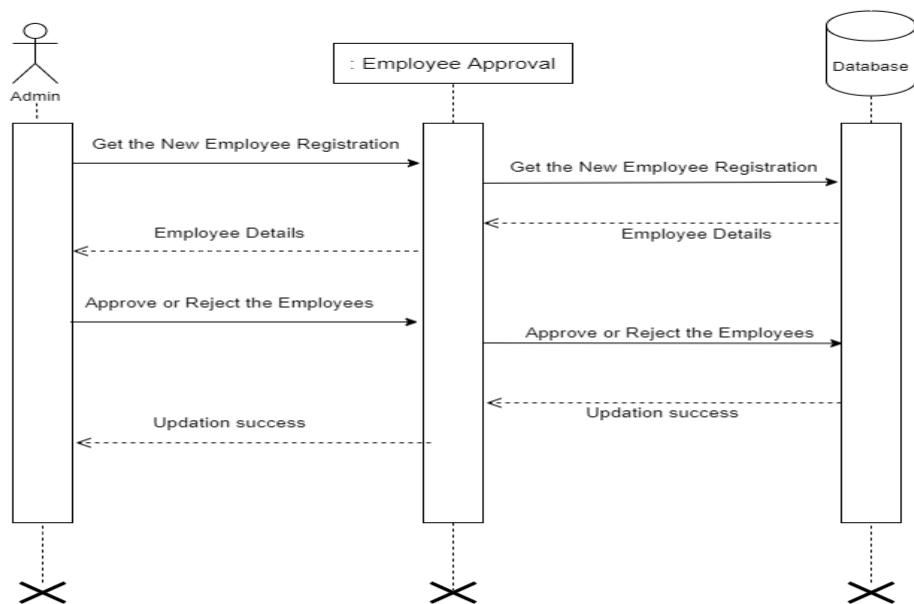


6.1.3 SEQUENCE DIAGRAM

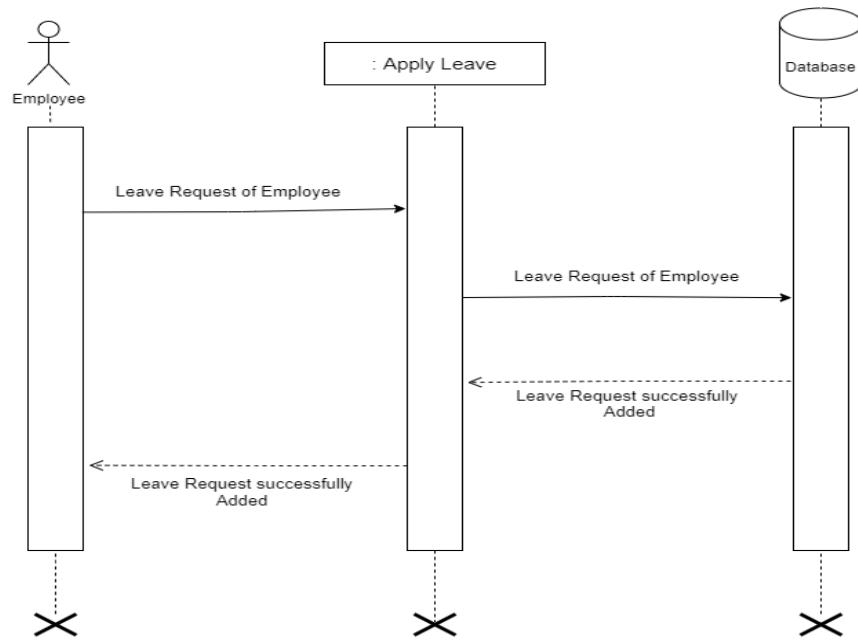
a) Login-Registration Process of Customers / Employees



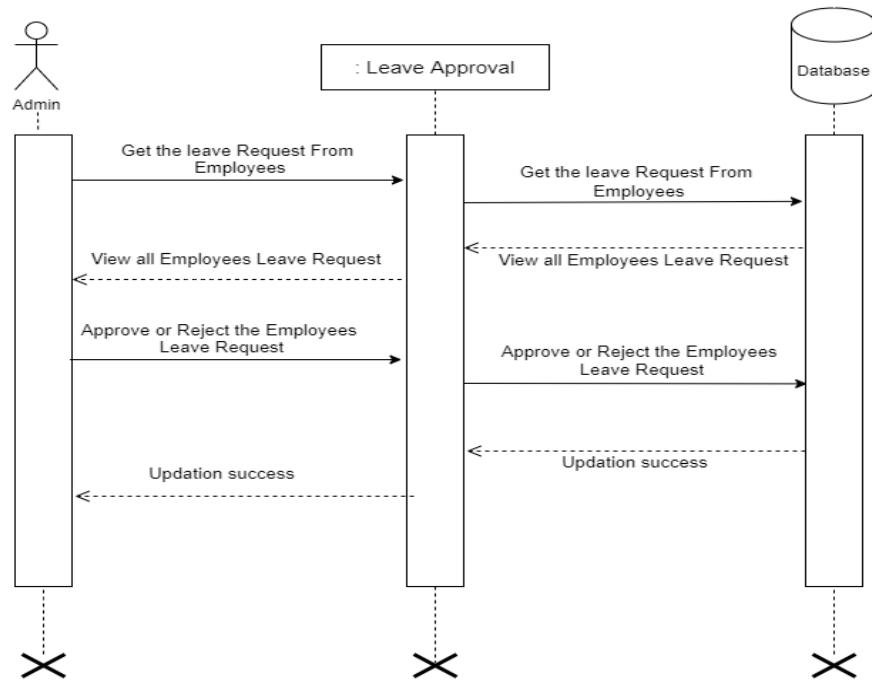
b) Employee Approval Process



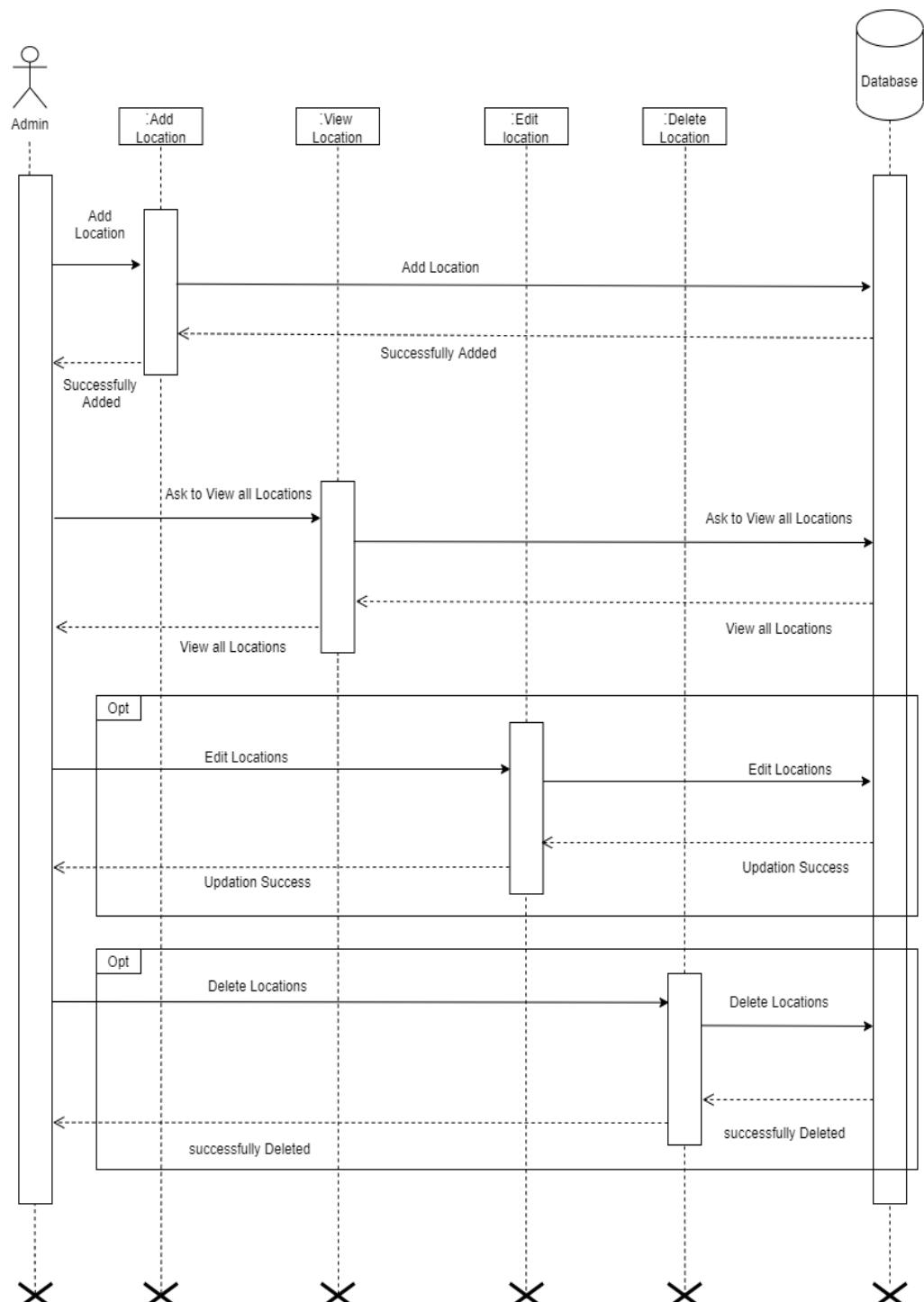
c) **Apply Leave Process**



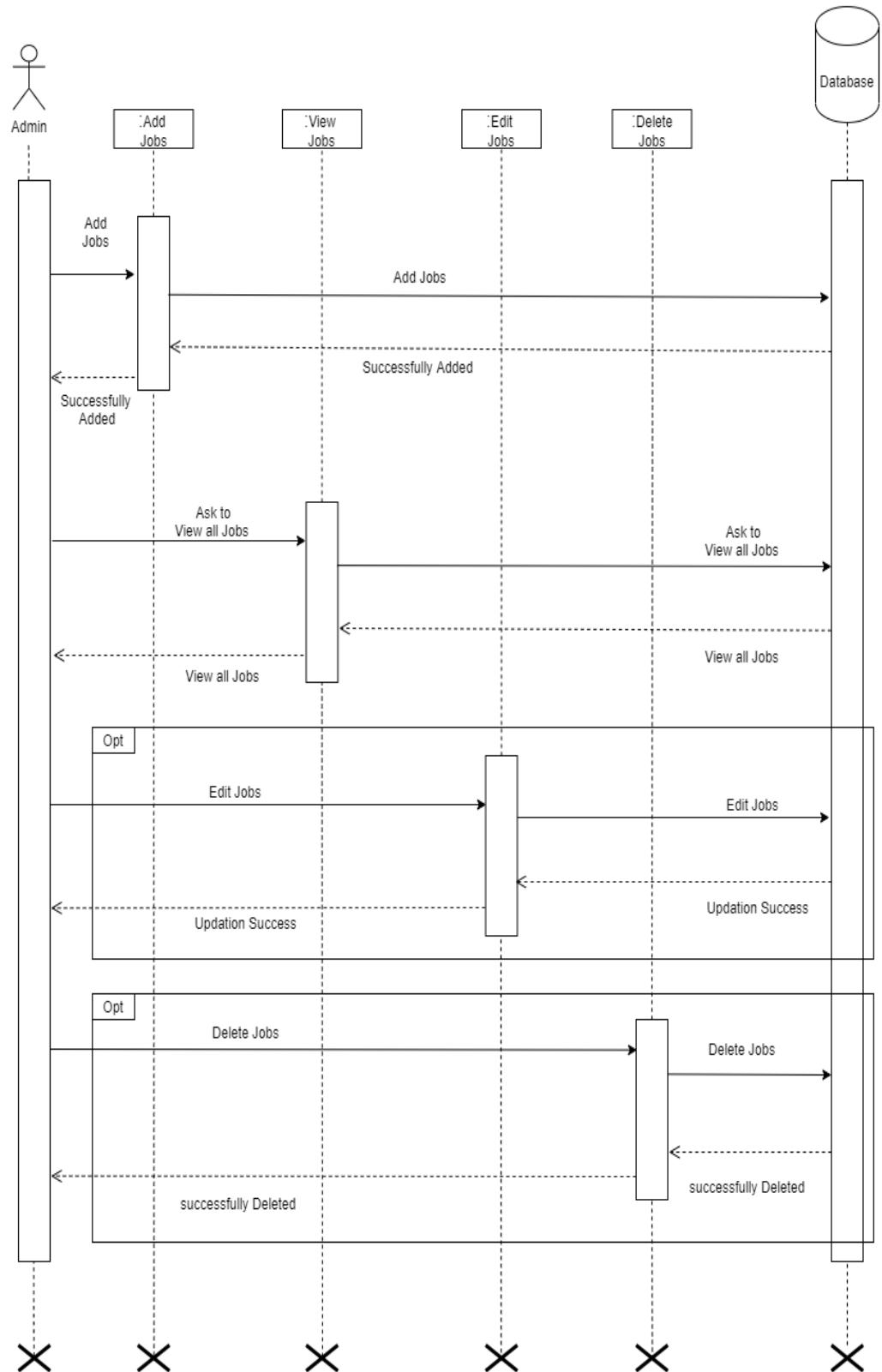
d) **Leave approval Process**



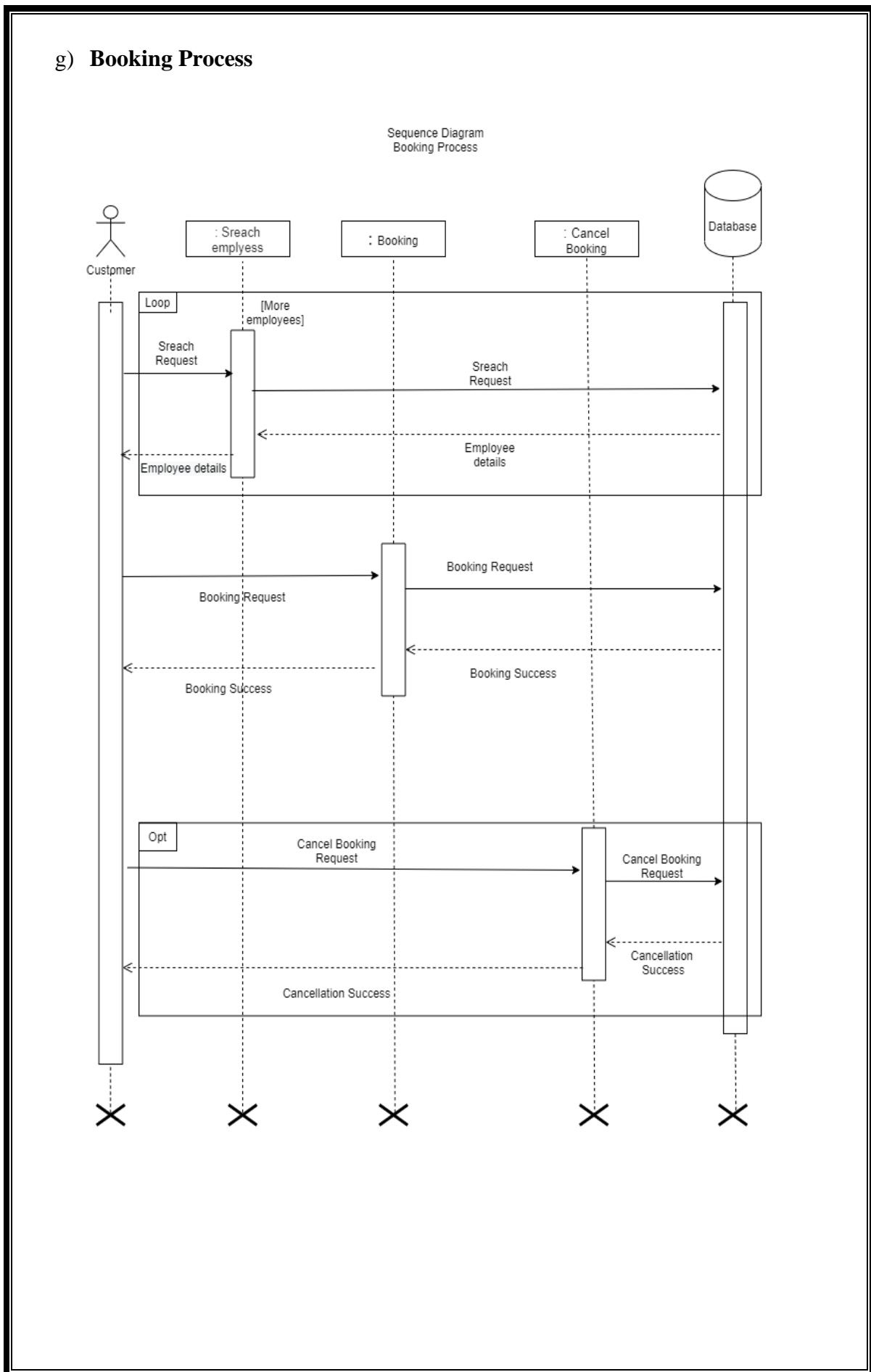
e) Manage Location Process



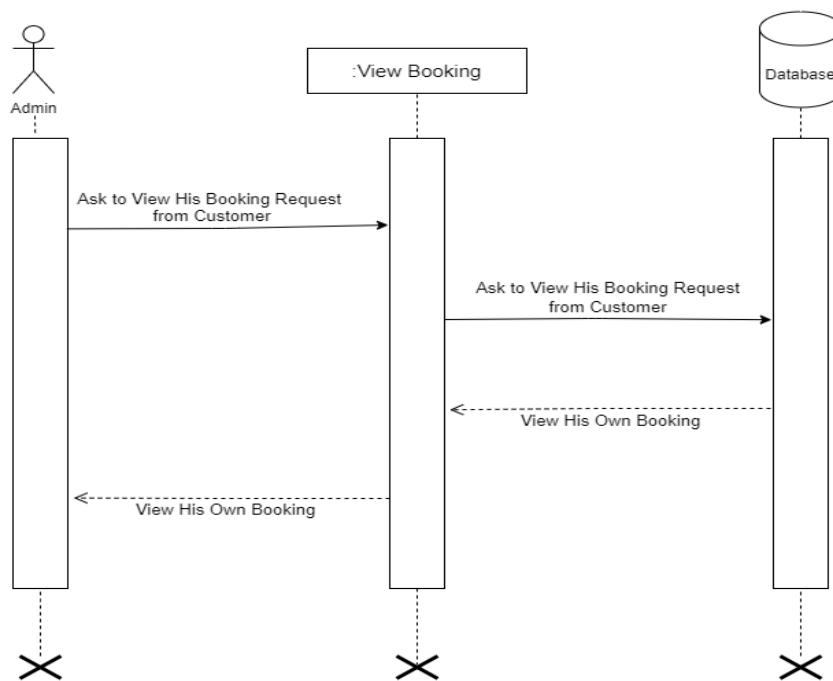
f) Manage Job Category Process



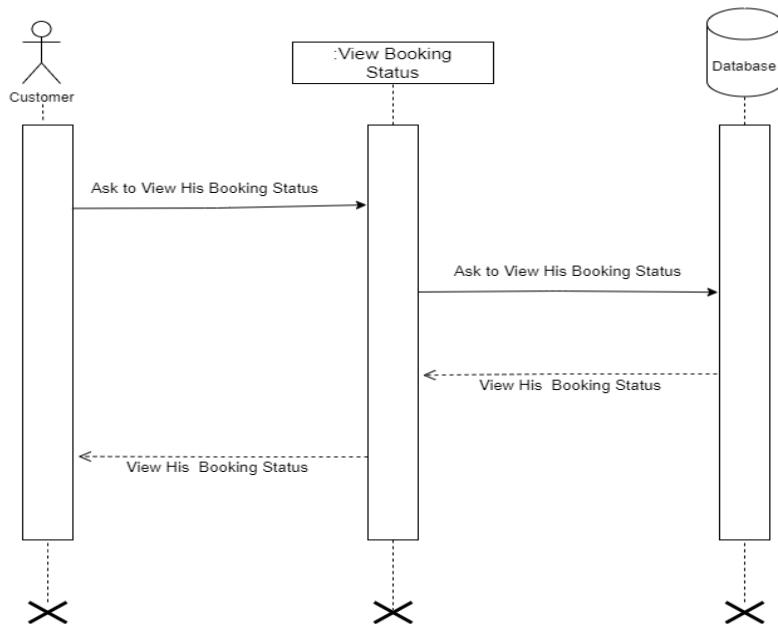
g) Booking Process

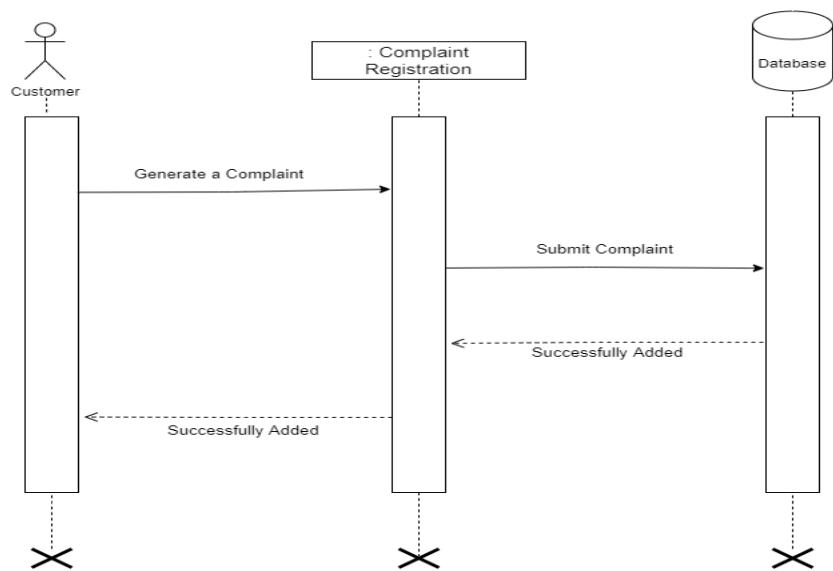
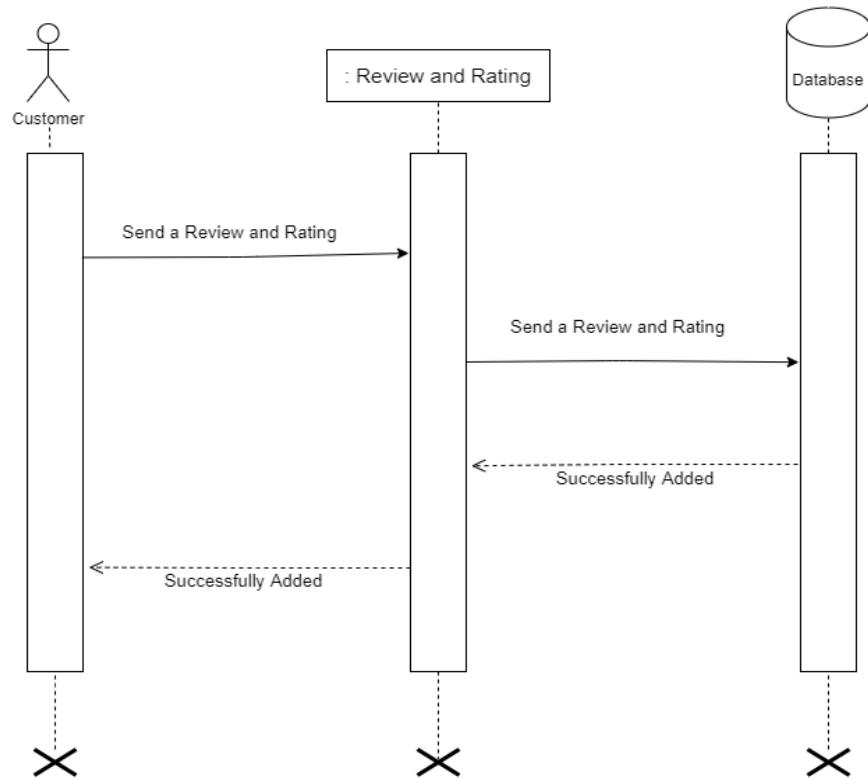


h) View Booking Process

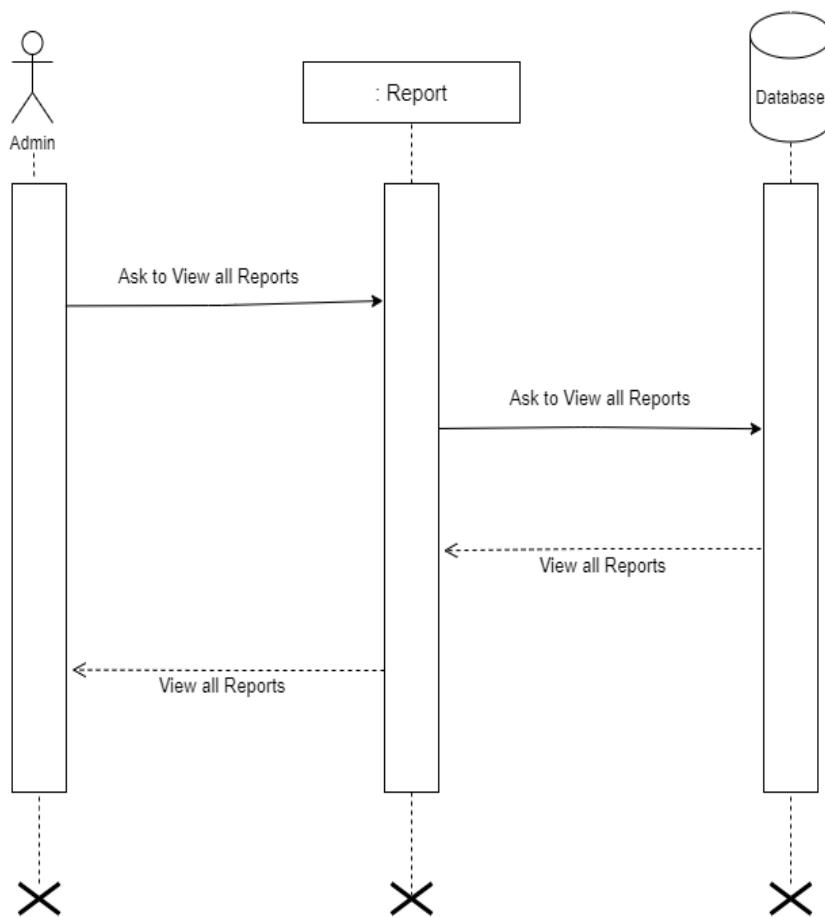


i) View Booking Status Process



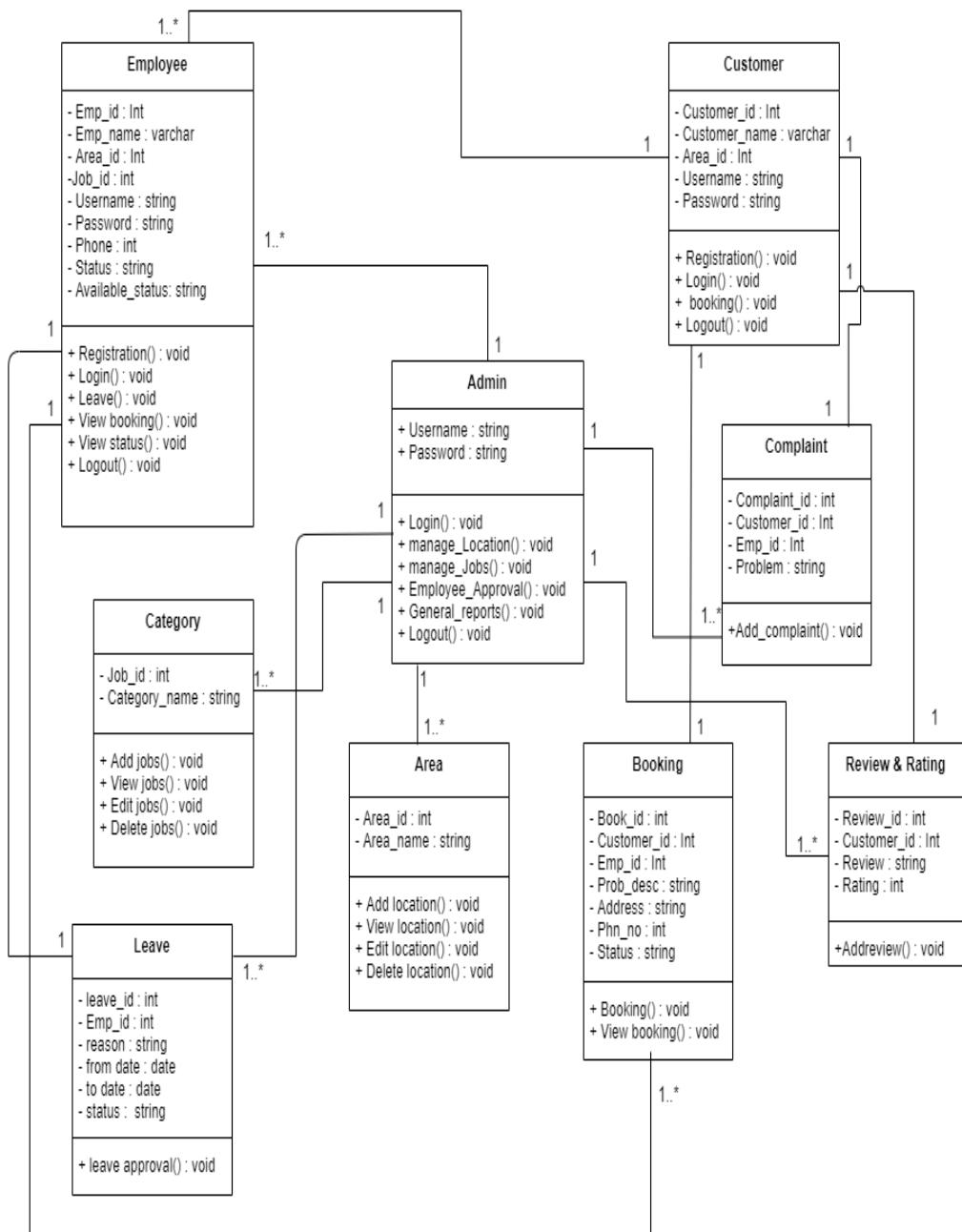
j) Complaint Registration**k) Review And Rating Process**

1) Report Process



6.1.4 CLASS DIAGRAM

CLASS DIAGRAM



SYSTEM TESTING

7.1 INTRODUCTION

Software testing is a critical element of software quality assurance and represents the ultimate previous of specification, design and coding. The testing phase involves the testing of the development system using test data. While testing the system using these test data, errors where found and corrected. Thus a series of test were performed to the system before the system was ready for implementation. Testing enhances the integrity of a system by identifying the deviations in the design and development of the expected end product.

It should focus more on the error-prone areas of the application. This helps in the prevention of errors in the system and builds confidence that the system will work without error testing. It is the process of executing a program with the intent of finding an error. Testing also adds value to the product by confirming to the user requirements. Testing verifies that software deliverable conforms precisely to the functional and design specifications that have been compiled during analysis and design phases. A good test case is the one that has the high probability of finding an as yet undiscovered error.

There are different types of system testing methods and some of them are:

7.2 UNIT TESTING

The unit test phase entails converting the design language in to program code and, most important, designing and carrying out tests of the individual units. Once individual modules or units have been tested and accepted, the integration and test phase begins. This initial part of structural testing corresponds to some quick checks that a developer performs before subjecting the code to more extensive code coverage testing or code complexity testing

In Home serve handyman services each modules are separated and tested. That means in the admin side, customer side and employee sides are separately tested. Check the duplication of data and the duplication is removed. And ensure that the updating are recorded correctly.

Test for the admin module

- Testing admin login form-This form is used for log in of administrator of the system. In this we enter the username and password. If both are correct admin home page will open otherwise if any of data is wrong it will get redirected back to the login page and again ask for correct or valid username and password.

Test for Customer login module

- Test for Customer login form-This form is used for log in for Customers. In this we enter the email and password. If all these are correct Customer homepage will open otherwise if any of data is wrong it will get redirected back to the login page and again ask for email and password.

Test for Employee login module

- Test for Employee login form-This form is used for log in for Employees. In this we enter the email and password. If all these are correct Employee homepage will open otherwise if any of data is wrong it will get redirected back to the login page and again ask for email and password.

7.3 INTEGRATION TESTING

It tests the integration of each module in the system. It also tests to find discrepancies between the system and its original objective. In this testing analysis we are trying to find areas where modules have been designed with different specification for data length, type etc. In Home serve handyman services integration testing is performed after unit testing. In unit testing it ensure that each module is working properly. In integration testing ensure that complete system working properly. In Home serve handyman services we ensure that the customer side, employee side and admin side together work properly.

First of all, the admin log in to the system and enters all necessary details such as home services, corresponding service name, service image, and also enter the location details, etc. These details are entered so that it is helpful for the user i.e. the customer's and employee registration.

Then the customer log in to the system and enters their personal details, email & password. They have a provision to edit their profiles and view their own account.

Then the employee log in to the system and enters their personal details, email & password. They have a provision to edit their profiles and view their own account.

7.4 VALIDATION TESTING

Validation refers to the process of using software in a live environment in order to find errors. During the course of validating the system, failure may occur and sometimes the coding has to be changed according to the requirement.

- Email entry check
 - Ensure that the user must enter a valid email.
- Password entry check
 - Ensure that the user must enter a valid password.
- Numeric and character check to corresponding fields
 - Ensures that the user must enter the numeric and character value in the specified field.
- Required Field Validator
 - Ensures that the user does not skip an entry.
- Date Check
 - Ensures that it is valid date or not.

7.5 TEST CASES

A test case is a set of sequential steps to execute a test operating on a set of predefined inputs to produce certain expected outputs. There are two types of test cases: - manual and automated. Manual test case is executed manually while an automated test case is executed using automation.

TEST CASE FOR LOGIN:

ID	TEST CASE	INPUT DATA	EXPECTED RESULT	ACTUAL RESULT	REMARK
1	Try to login	When no id is entered	When the valid username entered	An error message the id field is required	Success
2	Try to login	When no password is entered	When the valid password entered	An error message the password field is required	Success
3	Try to login	When no id and password entered	When the valid username and password entered	An error message please enter the details	Success

TEST CASE FOR ADDING SERVICES, AREA DETAILS:

ID	TEST CASE	INPUT DATA	EXPECTED RESULT	ACTUAL RESULT	REMARK
1	Try to add data to application	When no details entered	When the valid Details entered	An error message the field is required	succes
2	Try to add data to application	When already existin g data entered	This data is already existed	An error message please enter a valid data	Succes

TEST CASE FOR BOOKING:

ID	TEST CASE	INPUT DATA	EXPECTED RESULT	ACTUAL RESULT	REMARK
1	Try to book a Employee	When no details entered	When the valid Details entered	An error message the field is required	Success
2	Try to book a Employee	On clickin g button book	Displays a message details are save successfully	Displays a message details are save successfull y	Success

SYSTEM IMPLEMENTATION

8.1 SYSTEM IMPLEMENTATION

A crucial phase in the system life cycle is the successful implementation of the new system design. Implementation simply means converting a new system design into operation. This involves creating computer compatible files, training, and telecommunication network before the system is up and running. A crucial factor in conversion is not disrupting the functioning of organization. Actual data were input into the program and the working of the system was closely monitored. It is a process of converting a new or revised system into an operational one. It is the essential stage in achieving a successful new system because usually it involves a lot of upheaval in the user. It must therefore be carefully planned and controlled to avoid problems.

The implementation phase involves the following tasks:

- Careful planning.
- Investigation
- Design of methods
- Training of the staff in the changeover phase.
- Evaluation of changeover.

We implemented this new system in parallel run plan without making any disruptions to the ongoing system, but only computerizing the whole system to make the work, evaluation and retrieval of data easier, faster and reliable.

8.2 TRAINING

System implementation is the process of making the newly designed system fully operational and consistent in performance. The logical miss-working the system can be identified if any. Various combinations of test data were feed. Each process accuracy/reliability checking was made. After the approval, the system was implemented in the user department.

The preparation of implementation of documentation process is often viewed as total sum of the software documentation process. In a well-defined software development environment, however the presentation of implementation documents is essentially an interactive process that synthesis and recognizes document items that were produced during the analysis and design phase for the presentation to user. The following are the three types of implementation documents.

- Conversion Guide
- User Guide
- Operation Guide

Conversion Guide

The Conversion Guide phase of the implementation, process the tasks that are required to place the system into an operation mode. They amplify the conversion lane that was defined during the internal design phase and defines file conversion, file creation and data entry requirements.

User Guide

The system application and operation functions describes the overall performance capabilities of the system and define procedures the user must follow to operate the system. In the realm of information system, the content of a user guide must be developed to coincide with a criterion that defines the characteristics of one of the following methods of data processing

- Off-line processing
- Direct access processing

Operation Guide

The function of an operation is to define the control requirements of an online tour booking website and provide instruction for initializing, running and terminating the system. The items contained in an operation guide may be grouped as follows.

- General information
- System overviews
- Run description

8.3 POST IMPLEMENTATION REVIEW

A post implementation review (PIR) is an assessment and review of the completed working solution. It will be performed after a period of live running. Sometime after the project is completed, there are three purposes for a post implementation review:

- To ascertain the degree of success from the project. In particular, the extent to which it met its objectives delivered planned levels or benefit and addressed the specific requirements as originally defined.
- To examine the efficiency of all elements of the working business solution to see if further improvements can be made to optimize the benefit delivered.
- To learn lessons from their project, lessons which can be used by the team members and by the organization to improve further project work and solutions.

SYSTEM MAINTENANCE

9.1 SYSTEM MAINTENANCE

Maintenance is making adaptation of the software for external changes (requirements changes or enhancements) and internal changes (fixing bugs). When changes are made during the maintenance phase all preceding steps of the model must be revisited.

9.2 TYPES OF MAINTENANCE

In my project Home Serve Handyman Services website there are three types of maintenance:

- Corrective (Fixing Bugs/errors)
- Adaptive (Updates due to environment changes)
- Perfective (Enhancements, requirements changes)

Corrective maintenance: Removes software faults.

Perfective maintenance: Improves the system without changing its functionality. The objective of perfective maintenance should be to prevent failures and optimize the software.

Adaptive maintenance: Modifies the software to keep it up to date with its operative environment. It may be needed because of changes in the user requirements, changes in target platform, or changes in external interfaces. Minor adaptive changes should be handled by normal maintenance process. Major adaptive changes should be carried out as a separate development project.

SYSTEM EVALUATION

System evaluation is the process of assessing the performance of a complete all kinds of home service booking in online website to discover how it is likely to perform in live scenarios. Systems developers spend a proportion of their trading time trying to improve their existing systems and looking for new and different ideas. It is essential that any proposed system or change is fully evaluated before being used in live situation. System evaluation brings your complete system into play.

CONCLUSION

HOME SERVE HANDYMAN SERVICES system which solves all the problems and demerits of the existing system. At the present situation it will be an affective web application that can be used in our daily life. The professionals can repair and fix everything around your home in an efficient manner. It is a one-stop solutions for all your home care and appliance maintenance needs. To reduce burden in finding in-house solutions for the services, the proposed system provides several services by providing service specialists at your doorstep in one click

This website provides a computerized version of Home service Booking in Online which will benefit the customers. It makes entire process online where the customers can register themselves, employees can upload their profiles and admin can generate and view reports...

There is a future scope of this facility that many more features such as tutor services, virus sanitization services, etc... can be added to this project which makes the users more attracted to the website and thus making it more interactive more user friendly and project which fulfills each user's need in the best way possible.

APPENDIX

12.1 SAMPLE CODE

- **Add Home Services:**

```
<?php

require('../config/autoload.php');
include("head.php");
$file=new FileUpload();

$elements=array(
    "jtit"=>"","jdesc"=>"","jimg"=>"");

$form=new FormAssist($elements,$_POST);

$dao=new DataAccess();

$labels=array('jtit'=>"Home Service Name","jdesc"=>"Description","jimg"=>"Home service Image",);

$rules=array(
    "jtit"=>array("required"=>true,"minlength"=>3,"maxlength"=>20,"alphaspaceonly"=>true,"unique"=>array("field"=>"job_title","table"=>"jobs")),
    "jdesc"=>array("required"=>true,"minlength"=>3,"maxlength"=>500),
    "jimg"=> array('filerequired'=>true),
);

$validator = new FormValidator($rules,$labels);

//$no=1234;
if(isset($_POST["btn_insert"]))
{
if($validator->validate($_POST))
{
if($fileName=$file-
>doUploadRandom($_FILES['jimg'],array('.jpg','.png','.jpeg'),1000000,5,'..uploads'))
    {

$data=array('job_title'=>$_POST['jtit'],
    'job_desc'=>$_POST['jdesc'],
    'job_image'=>$fileName,
```

```

);

if($dao->insert($data,"jobs"))
{
    //echo "<script> alert('New record created successfully');</script> ";
    $msg="Registration Success";
header('location:manage_add_jobs.php');
}
else
{$msg="Registration failed";} ?>

<span style="color:red;"><?php echo $msg; ?></span>

<?php

}

else
echo $file->errors();
}

}

elseif (isset($_POST["btn_view"])) {
    header('location:viewmanagejobs.php');
}

?>
<html>
<head>
</head>
<body>
<br><br><br>
<form action="" method="POST" enctype="multipart/form-data">
<div class="container">
<br>
<H1 style="" class="text-center p-5"><b> Home Services </b></H1>

<div class="row">
    <div class="col-md-6">
<span style="">Home Service Name:</span>

<?= $form->textBox('jtit',array('class'=>'form-control')); ?>
<span style="color:red;"><?= $validator->error('jtit'); ?></span>

</div>
</div>
<br>
<div class="row">
    <div class="col-md-6">

```

```

<span style="">Description:</span>

<?= $form->textArea('jdesc',array('class'=>'form-control')); ?>
<span style="color:red;"><?= $validator->error('jdesc'); ?></span>

</div>
</div>
<br>
<div class="row">
    <div class="col-md-6">
<span style="">Home Service Image:</span>

<?= $form->fileField('jimg',array('class'=>'form-control')); ?>
<span style="color:red;"><?= $validator->error('jimg'); ?></span>

</div>
</div>
<br>
<br>
<br>
<input type="submit" name="btn_insert" class="btn btn-success" value="Add Services"/>
<input type="submit" name="btn_view" class="btn btn-success" value="View Services"/>
</form>
</div></div></div>

</body>

</html>
<?php include("ad_foot.php"); ?>

```

- **Add Location :**

```

<?php

require('../config/autoload.php');
include("head.php");
$file=new FileUpload();

$elements=array(
    "area_name"=>"", "area_image"=>"");

$form=new FormAssist($elements,$_POST);

```

```

$dao=new DataAccess();

$labels=array('area_name'=>"Area Name","area_image"=>"Map");

$rules=array(
    "area_name"=>array("required"=>true,"minlength"=>3,"maxlength"=>20,"alphaspaceonly"=>true,"unique"=>array("field"=>"area_name","table"=>"area")),
    "area_image"=> array('filerequired'=>true),
);

$validator = new FormValidator($rules,$labels);

//$_no=1234;
if(isset($_POST["btn_insert"]))
{
if($validator->validate($_POST))
{
if($fileName=$file-
>doUploadRandom($_FILES['area_image'],array('.jpg','.png','.jpeg'),100000,5,'..uploads'))
{
}

$data=array(
    'area_name'=>$_POST['area_name'],
    'area_image'=>$fileName,
);

if($dao->insert($data,"area"))
{
    //echo "<script> alert('New record created successfully');</script> ";
    $msg="Registration Success";
header('location:manage_area.php');
}
else
    {$msg="Registration failed";} ?>
}

```

```

<span style="color:red;"><?php echo $msg; ?></span>

<?php

}

else
echo $file->errors();
}

}

elseif (isset($_POST["btn_view"])) {
    header('location:viewmanagearea.php');
}

?>
<html>
<head>
</head>
<body>
<br><br><br>
<form action="" method="POST" enctype="multipart/form-data">
<div class="container"><br>
<H1 style="" class="text-center p-5"><b> Location Details </b></H1>
<br>
<div class="row">
    <div class="col-md-6">
<span style="">Area Name:</span>

<?= $form->textBox('area_name',array('class'=>'form-control')); ?>
<span style="color:red;"><?= $validator->error('area_name'); ?></span>

</div>
</div>

<br>
<div class="row">
    <div class="col-md-6">
<span style="">Map:</span>

<?= $form->fileField('area_image',array('class'=>'form-control')); ?>
<span style="color:red;"><?= $validator->error('area_image'); ?></span>

</div>
</div>

<br>

```

```

<input type="submit" name="btn_insert" class="btn btn-success" value="Add Area"/>
<input type="submit" name="btn_view" class="btn btn-success" value="View Area"/>
</form>
</div>

</body>

</html>
<?php include("ad_foot.php"); ?>

```

- **Customer Profile:**

```

<?php $conn = new mysqli("localhost","root", "", "project"); ?>
<?php include("customer_profile.php");      ?>
<div class="breadcrumbs-w3l">
    <div class="container">
        <span class="breadcrumbs">
            <a href="index.php">Home</a> |
            <span>Profile</span>
        </span>
    </div>
</div>
<?php
session_start();
if($_SESSION['email']){
$email = $_SESSION['email'];
}
else
{
echo "<script> location.href='customer_login.php'</script>";
}
$sql
                =
                "SELECT
cust_fristname,cust_lastname,cust_age,cust_gender,cust_housename,cust_ph,email,password
FROM customer WHERE email = '$email'";
$result = $conn->query($sql);
if($result->num_rows==1)
{
$row = $result->fetch_assoc();
$cust_fristname = $row['cust_fristname'];
$cust_lastname = $row['cust_lastname'];
$cust_age = $row['cust_age'];
$cust_gender = $row['cust_gender'];
$cust_housename = $row['cust_housename'];
$cust_ph = $row['cust_ph'];
$password = $row['password'];

}

```

```

if(isset($_POST['delete']))
{
$sql1 ="DELETE FROM customer WHERE email='$email'";
if ($conn->query($sql1) === TRUE) {
    echo "<script> alert(' Deleted Your Account Successfully');</script> ";
    echo"<script> location.replace('customer_login.php'); </script>";

} else {
    echo "Error: " . $sql1 . "<br>" . $conn->error;
}
}

if(isset($_POST["update"]))
{
//echo $POST["ename"];
$firstname = $_POST["cust_frstname"];
$lastname = $_POST["cust_lastname"];
//$dob = $_POST['dob'];
$age = $_POST['client_age'];
//$gender = $_POST['gender'];
$housename = $_POST['client_housename'];
//$city = $_POST['city'];
//$pincode = $_POST['pincode'];
$phone = $_POST['client_ph'];
//$doj = $_POST['doj'];
//$emp_password = $_POST['emp_password'];
//$emp_desc = $_POST['emp_desc'];
//$emp_exp = $_POST['emp_exp'];
//$emp_exp = $_POST['emp_exp'];
//$emp_quali = $_POST['emp_quali'];
//register_fee = 1000;

$sql2          ="UPDATE           customer      SET
cust_frstname='$firstname',cust_lastname='$lastname',cust_age='$age',cust_housename='
$housename',client_ph='$phone' WHERE email='$email'";
if ($conn->query($sql2) === TRUE) {
    echo "<script> alert(' Updated Successfully');</script> ";
    echo"<script> location.replace('profile.php'); </script>";

} else {
    echo "Error: " . $sql2 . "<br>" . $conn->error;
}
}

?>
<!DOCTYPE html>
<html lang="en">
<!DOCTYPE html>
<html lang="en">

```

```

<head>

</head>

<body>
<div class="box">

    <div class="container">
        <div id="news" class="w3ls-section">
            <div class="container">
                <h4 class="main-title" class="text-center p-5"> Your Profile </h4><br>
                <hr><hr>
                <form action="" method="POST">
                    <div class="row">
                        <div class="col-md-4">
                            <div class="form-group">
                                <label for="">First Name:</label>

                                <input type="text" name="cust_fristname" id="cust_fristname" class="form-control" value="<?php echo $cust_fristname ?>"></div></div>
                        <div class="col-md-4">
                            <div class="form-group">
                                <label for="">Last Name:</label>
                                <input type="text" name="cust_lastname" id="cust_lastname" class="form-control" value="<?php echo $cust_lastname ?>"></div></div>
                        <div class="col-md-4">
                            <div class="form-group">
                                <label for="">Age:</label>
                                <input type="text" name="client_age" id="client_age" class="form-control" value="<?php echo $client_age ?>"></div></div></div><br>
                            <div class="row">
                                <div class="col-md-4">
                                    <div class="form-group">
                                        <label for="">Gender:</label>
                                        <input type="text" name="client_gender" id="client_gender" class="form-control" value="<?php echo $client_gender ?>"></div></div>
                                <div class="col-md-6">
                                    <div class="form-group">
                                        <label for="">House Name:</label>
                                        <input type="text" cols="2" rows="4" name="client_housename" id="client_housename" value="<?php echo $client_housename ?>" class="form-control"></textarea></div></div></div><br>
                                <div class="row">
                                    <div class="col-md-4">
                                        <div class="form-group">
                                            <label for="">Contact No:</label>

```

```

<input type="text" name="client_ph" id="client_ph" class="form-control"
value=<?php echo $client_ph ?>></div></div>
<div class="col-md-4">
    <div class="form-group">
        <label for="">Email :</label>
        <input type="text" name="email" id="email" class="form-control" value=<?php
echo $email ?>></div></div>
<div class="col-md-4">
    <div class="form-group">
        <label for="">Password:</label><br>
        <input type="text" name="password" id="password" class="form-control"
value=<?php echo $password ?>" ></div></div></div><br>
<div class="row">
    <div class="col-md-6">
        <div class="form-group">
            <input type="submit" name="delete" value="Delete
Acoount" class="btn btn-primary btn-block" class="text-center p-5">
        </div>
    </div>
    <div class="col-md-6">
        <div class="form-group">
            <input type="submit" name="update" value="Update
Profile" class="btn btn-success btn-block" class="text-center p-5">
        </div>
    </div></div>
</form></div></div></div></div></div></body></html>

<?php
include("footer.php");
?>

```

- **Employee Login :**

```

<!DOCTYPE html>
<html lang="en">
<head>

<!--
=====
=====-->
</head>
<body>
<?php $conn = new mysqli("localhost","root", "", "project"); ?>
<?php require('../config/autoload.php'); ?>

```

```

<?php
include("header.php"); ?>
<div class="breadcrumbs-w3l">
    <div class="container">
        <span class="breadcrumbs">
            <a href="index.php">Home</a> |
            <span>Login </span>
        </span>
    </div>
</div>

<?php
$dao=new DataAccess();

$elements=array("emp_email"=>"","emp_password"=>"");
$form=new FormAssist($elements,$_POST);
$rules=array(
    'emp_email'=>array('required'=>true),
    'emp_password'=>array('required'=>true),
);
$validator=new FormValidator($rules);

if(isset($_POST['login']))
{
    if($validator->validate($_POST))
    {
        $mail=$_POST['emp_email'];
        $pass=$_POST['emp_password'];
        echo "$mail";
        echo "$pass";
        $sql="SELECT * FROM employee WHERE emp_email='".$mail."' and emp_password='".$pass."' and approve_status='approve'";

        $result=$conn->query($sql);

        if ($result->num_rows > 0)
        {
            echo "inside";
            $row = $result->fetch_assoc();
            $_SESSION['emp_email']=$row['emp_email'];
            $_SESSION['job_id']=$row['job_id'];
            $_SESSION['emp_id']=$row['emp_id'];
        }
    }
}

```

```

$_SESSION['area_id']=$row['area_id'];
$_SESSION['firstname']=$row['firstname'];
//$_SESSION['job_title']=$row['job_title'];
$a=$_SESSION['emp_email'];
$c=$_SESSION['job_id'];
$e=$_SESSION['emp_id'];
$c1=$_SESSION['area_id'];
$c2=$_SESSION['firstname'];

//$o=$_SESSION['job_title'];

echo "<script> alert('$a');</script> ";

echo"<script> location.replace('employee_profile.php'); </script>";

}

else{
    $msg="invalid username or password";

        echo "<script> alert('Invalid username or password');</script> ";
    }

}

?>

<div class="jumbotron">
    <div class="container">
        <center>
            </center>
            <div id="news" class="w3ls-section">
                <div class="container">
                    <h4 class="main-title" class="text-center p-5"> Account Login </h4><br>
                    <form action="" method="POST">
                        <div class="row">
                            <div class="col-md-6">
                                <div class="form-group">
                                    <label for="">User name</label>

```

```

        <input type="text" name="emp_email" class="form-control">
            </div>
        </div>
        <div class="col-md-6">
            <div class="form-group">
                <label for="">Password</label>
                <input type="password" name="emp_password" class="form-control">
            </div>
        </div>
        </div><br>
        <br>
<div class="row">

    <div class="col-md-6">
        <div class="form-group">

            <input type="submit" name="login" value="Sign in" class="btn btn-success btn-block" style="text-align: center; padding: 5px">
        </div>
    </div>

<div class="col-md-6">
    <div class="form-group">
        <label for="">Forgot</label>
        <a href="reset.php">Your Password ? </a>
    </div>
</div></div><br>
<div class="row">

    <div class="col-md-6">
        <div class="form-group">
            <label for="">Are You a new Handyman?</label>
            <a href="employee_reg.php">Sign Up </a>
        </div>
    </div></div><br>

        </form></div></div>
    </div>
</div><?php include("footer.php"); ?>
    • Booking Form

<?php require('../config/autoload.php'); ?>
<?php $conn = new mysqli("localhost", "root", "", "project"); ?>
<?php include("customer_profile.php"); ?>
<div class="breadcrumbs-w3l">

```

```

<div class="container">
<span class="breadcrumbs">
    <a href="index.php">Home</a> | 
    <span>Book Now!</span>
</span>
</div>
</div>
<?php
$dao=new DataAccess();
//session_start();
$b=$_SESSION['client_id'];
$fromdate=$_SESSION['fromdate'];

        $todate=$_SESSION['todate'];
        $days=$_SESSION['days'];
        //echo $days;
//$a=$_SESSION['email'];

//$f=$_SESSION['book_date'];
//$p=$_SESSION['todate'];
//$b=$_SESSION['client_id'];

//$d=$_SESSION['client_fristname'];

$info=$dao->getData('*', 'employee', 'emp_id='.$_GET['id4']);
$file=new FileUpload();
$elements=array(
"emp_id"=>$info[0]['emp_id'], "firstname"=>$info[0]['firstname'], "book_date"=>"", "problem_desc"=>"", "requester_name"=>"", "housename"=>"", "city"=>"", "landmark"=>"", "pincode"=>"", "contact_no"=>"", "fromdate"=>"", "todate"=>"", "email"=>"email", "client_id"=>"");
//$d=$_SESSION['client_fristname'];
$form = new FormAssist($elements,$_POST);

$labels=array("emp_id"=>"ID      of      Employee", "firstname"=>"Name      of      the Handyman", "book_date"=>"Booking      Date", "problem_desc"=>"Description      about Problem", "requester_name"=>"Requester Name", "housename"=>"Housename", "city"=>"City", "landmark"=>"Landmark", "pincode"=>"Pincode", "contact_no"=>"Contact      Number", "fromdate"=>"Booking Date", "todate"=>"To Date", "email"=>"email", "client_id"=>"customer Id",);

$rules=array(
"emp_id"=>array("required"=>true),
"firstname"=>array("required"=>true),

```

```

"book_date"=>array("required"=>true),

"problem_desc"=>array("required"=>true,"minlength"=>3,"maxlength"=>50,"alphaspace
only"=>true),

"requester_name"=>array("required"=>true,"minlength"=>3,"maxlength"=>30,"alphaspa
ceonly"=>true),


"housename"=>array("required"=>true,"minlength"=>3,"maxlength"=>50,"alphaspaceon
ly"=>true),

"city"=>array("required"=>true,"minlength"=>3,"maxlength"=>20,"alphaspaceonly"=>tr
ue),

"landmark"=>array("required"=>true,"minlength"=>3,"maxlength"=>30,"alphaspaceonly
"=>true),


"contact_no"=>array("required"=>true,"minlength"=>10,"maxlength"=>10,"integeronly"
=>true),
    "fromdate"=>array("required"=>true),
    "todate"=>array("required"=>true),
    "email"=>array("required"=>true),
    "client_id"=>array("required"=>true),


);

$validator = new FormValidator($rules,$labels);
//$_POST['client_id']="";

/*if(isset($_POST["btn_insert"]))
{
    $client_id=$_POST['client_id'];

    //$_POST['client_fristname']=>$client_fristname,
    //$_POST['emp_id']=>$emp_id;
    //$_POST['firstname']=>$firstname,
    //$_POST['problem_desc']=>$problem_desc;
    //$_POST['requester_name']=>$requester_name;
    //$_POST['housename']=>$housename;
    //$_POST['landmark']=>$landmark;
    //$_POST['contact_no']=>$contact_no;
}
*/

```

```

$book_date=$_POST['book_date'];
$todate=$_POST['todate'];
$email=$_POST['email'];

$status='pending';
$sql = "INSERT INTO booking(client_id,emp_id,problem_desc,requester_name,housename,landmark,pincode,c
ontact_no,book_date,todate,email,status) VALUES
('{$client_id}', '{$emp_id}', '{$problem_desc}', '{$requester_name}', '{$housename}', '{$landmark}', '{$pin
code}', '{$contact_no}', '{$book_date}', '{$todate}', '{$email}', '{$status}')";

if ($conn->query($sql) === TRUE) {
    echo "<script> alert('New record created successfully');</script> ";
} else {
    echo "Error: " . $sql . "<br>" . $conn->error;
}
}/*
if(isset($_POST["btn_insert"]))
{
if(!isset($_SESSION['email']))
{
// header('location:login1.php');
//echo"<script >location.href = 'customer_login.php'</script>";

}
else
{
$name=$_SESSION['email'];
$b=$_SESSION['client_id'];
$fromdate=$_SESSION['fromdate'];
$todate=$_SESSION['todate'];
$days=$_SESSION['days'];
//$_tid = $_GET['id'];
//$_q1="select * from subcat where room_id=".$_tid;

//$_info1=$dao->query($_q1);
$_emp_id=$_POST['emp_id'];
//$_iname=$_info1[0]["room_no"];
$_book_date=date('Y-m-d',time());
$_problem_desc=$_POST['problem_desc'];
$requester_name=$_POST['requester_name'];
$housename=$_POST['housename'];
$_city=$_POST['city'];
$_landmark=$_POST['landmark'];
$_pincode=$_POST['pincode'];
$_contact_no=$_POST['contact_no'];
//$_fromdate = $_POST["fromdate"];
//$_todate = $_POST["todate"];
}
}

```

```

//$itemname = $iname;
//$fees = $info1[0]["fees"];
//$advance = $_POST["advance"];
//$_SESSION['advance']=$advance;
$email=$_POST["email"];
$book_status='booked';
$status='pending';

//date1=date('Y-m-d',time());

//balance=$_POST["balance"];
//diffDays= $_SESSION['date2'];
//$_SESSION['days']=diffDays;

//total=$_POST["total"]  ;

$sql          =      "INSERT           INTO
booking(client_id,emp_id,book_date,problem_desc,requester_name,housename,city,land
mark,pincode,contact_no,fromdate,todate,days,email,book_status,status)      VALUES
('$b','$emp_id','$book_date','$problem_desc','$requester_name','$housename','$city','$lan
dmark','$pincode','$contact_no','$fromdate','$todate','$days','$name','$book_status','$statu
s')";

$conn->query($sql);

// echo $sql;
//header('location:viewcart.php');
echo"<script >location.href = 'bookstatus.php'</script>";
}

?>
<html>
<head>
<style>
.form{
border:3px solid blue;
}
</style>
</head>
<body>

<div class="container">
<div class="boxed">
<div id="news" class="w3ls-section">
<div class="container">
<h4 class="main-title" class="text-center p-5"> Booking Request </h4><br>

```

```

<form action="" method="POST">
    <?php
if(isset($_SESSION['email']))
{
    //$_book_date=$_SESSION['book_date'];
//$_todate=$_SESSION['todate'];
//$_tdate=$_SESSION['date2'];

    $name=$_SESSION['email'];
?>

<?php } ?>

<table>
<tr>

<div class="row"><div class="form-group">
    <div class="col-md-6">
<span>Selected Employee ID</span>

<?= $form->textBox('emp_id',array('class'=>'form-control')); ?>
<?= $validator->error('emp_id'); ?>

</div>
</div>
<div class="col-md-6">
    <div class="form-group">
        <label for="">Selected Employee Name:</label>
        <input type="text" name="firstname" id="email" class="form-control" value="<?php
echo $info[0]['firstname']; ?>" disabled></div></div></div>

<br>
<div class="row"><div class="form-group">
    <div class="col-md-4">
        <label for="">From Date :</label>
        <input type="date" name="fromdate" id="email" class="form-control" value="<?php
echo $fromdate ?>" disabled></div></div>

<div class="col-md-4">
    <div class="form-group">
        <label for="">To Date :</label>
        <input type="date" name="todate" id="email" class="form-control" value="<?php
echo $todate ?>" disabled></div></div>
<div class="col-md-4">

```

```

<div class="form-group">
    <label for="">No of days :</label>
    <input type="text" name="days" id="email" class="form-control" value=<?php echo
$days ?>" disabled></div></div></div>

<br>
<div class="row">
<div class="col-md-4">
    <div class="form-group">
        <label for="">Booking Date:</label>
        <input type="date" name="book_date" id="email" class="form-control"
value=<?php echo date('Y-m-d'); ?>" disabled></div></div></div><br>

<div class="row"><div class="form-group">
    <div class="col-md-6">

<span>Description About Problem:</span>
<?= $form->textArea('problem_desc',array('class'=>'form-control')); ?>
<?= $validator->error('problem_desc'); ?>

</div>
</div></div>
<br>

<div class="row"><div class="form-group">
    <div class="col-md-6">

<span>Requester Name:</span>
<?= $form->textBox('requester_name',array('class'=>'form-control')); ?>
<?= $validator->error('requester_name'); ?>

</div>
</div></div>
<br>
<div class="row"><div class="form-group">
    <div class="col-md-12">

<span>House Name:</span>
<?= $form->textArea('housename',array('class'=>'form-control')); ?>
<?= $validator->error('housename'); ?>

</div>
</div></div>
<br>
<div class="row"><div class="form-group">
    <div class="col-md-6">

<span>City:</span>

```

```

<?= $form->textArea('city',array('class'=>'form-control')); ?>
<?= $validator->error('city'); ?>

</div>
</div>
<div class="form-group">
    <div class="col-md-6">

<span>Landmark:</span>
<?= $form->textBox('landmark',array('class'=>'form-control')); ?>
<?= $validator->error('landmark'); ?>

</div>
</div>
</div><br>
<div class="row"><div class="form-group">
    <div class="col-md-6">

<span>Pincode:</span>
<?= $form->textBox('pincode',array('class'=>'form-control')); ?>
<?= $validator->error('pincode'); ?>

</div>
</div>

<div class="form-group">
    <div class="col-md-6">

<span>Contact No:</span>
<?= $form->textBox('contact_no',array('class'=>'form-control')); ?>
<?= $validator->error('contact_no'); ?>

</div>
</div></div>
<br><div class="row">
<div class="col-md-6">
    <div class="form-group">
        <label for="">Email :</label>
        <input type="text" name="email" id="email" class="form-control" value="<?php
echo $name ?>" disabled></div></div>

<div class="col-md-6">
    <div class="form-group">
        <label for="">customer Id :</label>
        <input type="text" name="client_id" id="email" class="form-control" value="<?php
echo $b ?>" disabled></div></div></div>

```

```

<br><br>
<div class="row">
    <div class="col-md-12">
        <div class="form-group">

            <input type="submit" name="btn_insert" value="Book Now!" class="btn btn-success btn-block" style="text-align: center; padding: 5px;">
        </div>
    </div></div>

</tr></table>

</form>
</div></div></div></div>
</body>
</html>
<?php include("footer.php"); ?>

```

- **View Booking Request**

```

<?php $conn = new mysqli("localhost", "root", "", "project"); ?>
<?php include("employee_pro.php"); ?>
<div class="breadcrumbs-w3l">
    <div class="container">
        <span class="breadcrumbs">
            <a href="index.php">Home</a> |
            <span>Booking requests!</span>
        </span>
    </div>
</div>
<?php
session_start();

$a=$_SESSION['emp_email'];
$e=$_SESSION['emp_id'];

//echo $e;

$sql = "SELECT * FROM booking WHERE status='pending' and book_status='booked' and emp_id = ".$e."
ORDER BY fromdate DESC";
$result = $conn->query($sql);
if($result->num_rows==1)
{
$row = $result->fetch_assoc();
$problem_desc = $row['problem_desc'];
$requester_name = $row['requester_name'];

```

```

$houseName = $row['houseName'];
$city = $row['city'];

$contact_no = $row['contact_no'];
$fromdate = $row['fromdate'];
$todate = $row['todate'];
$days = $row['days'];
$book_status = $row['book_status'];
}

if(isset($_POST['okay']))
{
echo "<script> location.replace('employee_profile.php'); </script>";
}
?>
<!DOCTYPE html>
<html lang="en">
<!DOCTYPE html>
<html lang="en">
<head>

</head>

<body>

<div class="container">
<div id="news" class="w3ls-section">
<div class="container">
    <h4 class="main-title" class="text-center p-5"> My Bookings! </h4><br>
    <h2 class="text-center p-5">Welcome <?php echo $a ?></h2><br>
    <form action="" method="POST">
        <table border="1" class="table" style="background:white">
            <tr>

                <th>Problem_desc</th>

                <th>Requester</th>

                <th>House Name</th>
                <th>City</th>
                <th>Contact No</th>
                <th>From Date</th>
                <th>To Date</th>
                <th>No of Days </th>
                <th>Booking status </th>

            </tr>

```

```

        </tr>
<?php
foreach ($result as $row) {
?>
<tr>
    <td><?php echo $row['problem_desc']; ?></td>
    <td><?php echo $row['requester_name']; ?></td>
    <td><?php echo $row['housename']; ?></td>
    <td><?php echo $row['city']; ?></td>
    <td><?php echo $row['contact_no']; ?></td>
    <td><?php echo $row['fromdate']; ?></td>
    <td><?php echo $row['todate']; ?></td>
    <td><?php echo $row['days']; ?></td>
    <td><?php echo $row['book_status']; ?></td>
    <td><a href="accept.php?id1=<?php echo $row['book_id']; ?>"><h7 class="btn btn-success btn-block" class="text-center p-5"> Accept</a></h7>
        <td><a href="reject.php?id2=<?php echo $row['book_id']; ?>"><h7 class="btn btn-success btn-block" class="text-center p-5"> Reject</a></h7>

        </tr>
<?php } ?>
</table>
<?php
if ($result->num_rows <= 0) {

echo "<script> alert('Sorry No Records Found');</script> ";

echo "<script >location.href = 'employee_profile.php'</script>";
} ?>
<div class="col-md-4">
    <div class="form-group">
        <label for="">Your ID:</label>
        <input type="text" name="email" id="email" class="form-control" readonly
value=<?php echo $e ?>></div></div>

</form></div></div></div></body></html>

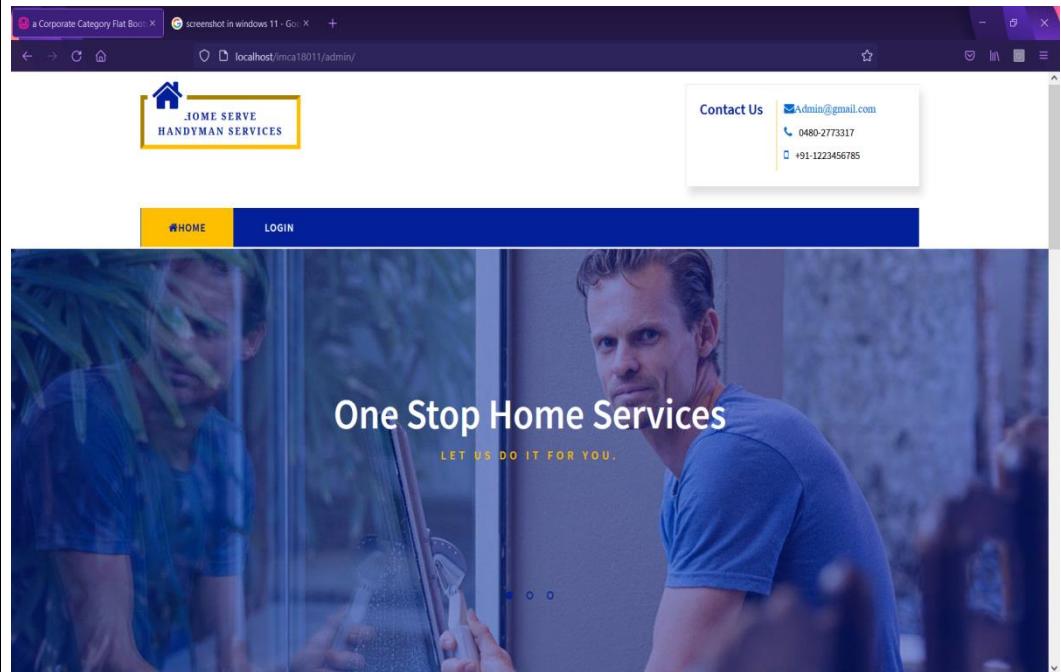
<?php
include("footer.php");
?>

```

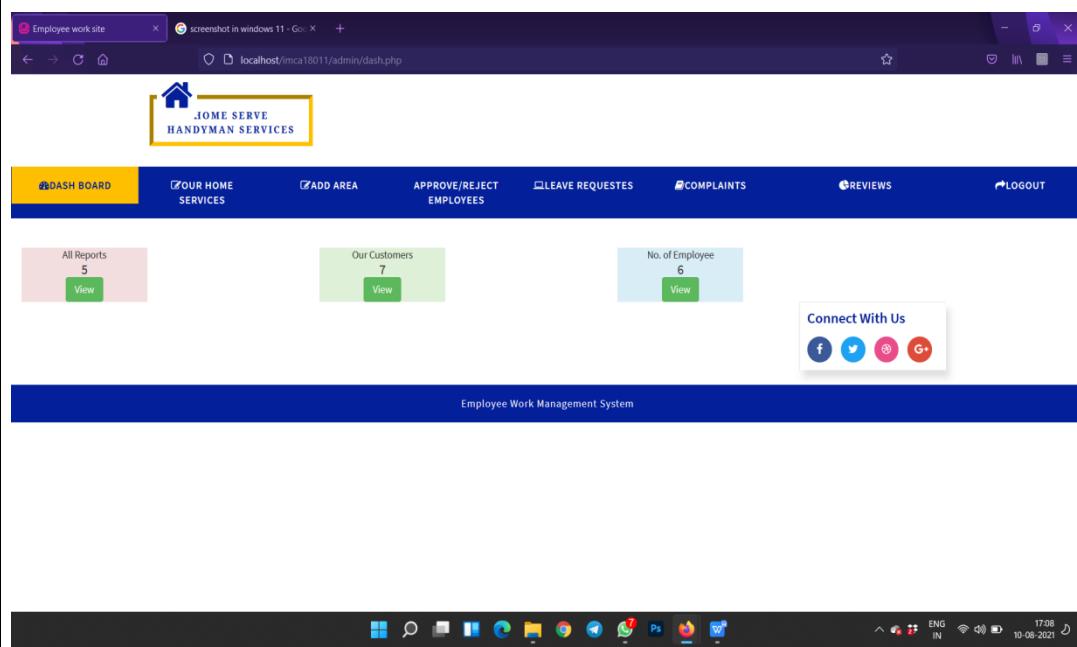
12.2 SCREEN SHOTS

ADMIN PAGES

- Home Page



- Dash Board



- Add Home Services

Add Jobs

Job Name: Electrician

Job Description: a person who installs and maintains electrical equipment.

Job Image:

a1fb1ef9a76d84f72322a6dd0e27e92_50773e13103dc1cc5d.jpg

- View Services

Sno	Home Service Name	Description	Image	
1	plumber	fits and repairs the pipes, fitting		<input type="button" value="Edit"/> <input type="button" value="Delete"/>
2	carmechnic	a skilled worker who repairs and maintains vehicle engines and other machinery		<input type="button" value="Edit"/> <input type="button" value="Delete"/>
3	painter	a person whose job is painting buildings.		<input type="button" value="Edit"/> <input type="button" value="Delete"/>
4	welder	a person who welds metal.		<input type="button" value="Edit"/> <input type="button" value="Delete"/>

- Add Area

Add Area

Location: Anganaly

area Image:

Browse... aef3f07dc0a2d5ea48628cf9c72a791_89d94166a7ebf0b1.jpg

Submit View Area

- View Registered Employees

OUR EMPLOYEES

Sno	Employee Name	employee age	Employee House name	
1	ebin	22	zzxsxsad	View More...
2	rojin	20	sdhsjhsah	View More...
3	hithesh	21	dhdhwk	View More...
4	mariya	21	sadiah	View More...
5	dominic	22	skhakjhas	View More...
6	nevin	22	wdjjjdgwjegdwj	View More...
7	raju	21	sdsdsadasd	View More...

Back

Employee work site screenshot in windows 11 - Go! X + localhost/mca18011/admin/view_approved_employees1.php?id=52

Employee Personal Details



First Name: ebin Last Name: baby

Date of Birth: 1999-03-16 Age: 22 Gender: m

Housename: zzaxasad City: sdhahd

Contact No: 7898786544

Email: ebinbaby@gmail.com Location Name: Chalakkudy

Home service Name: plumber

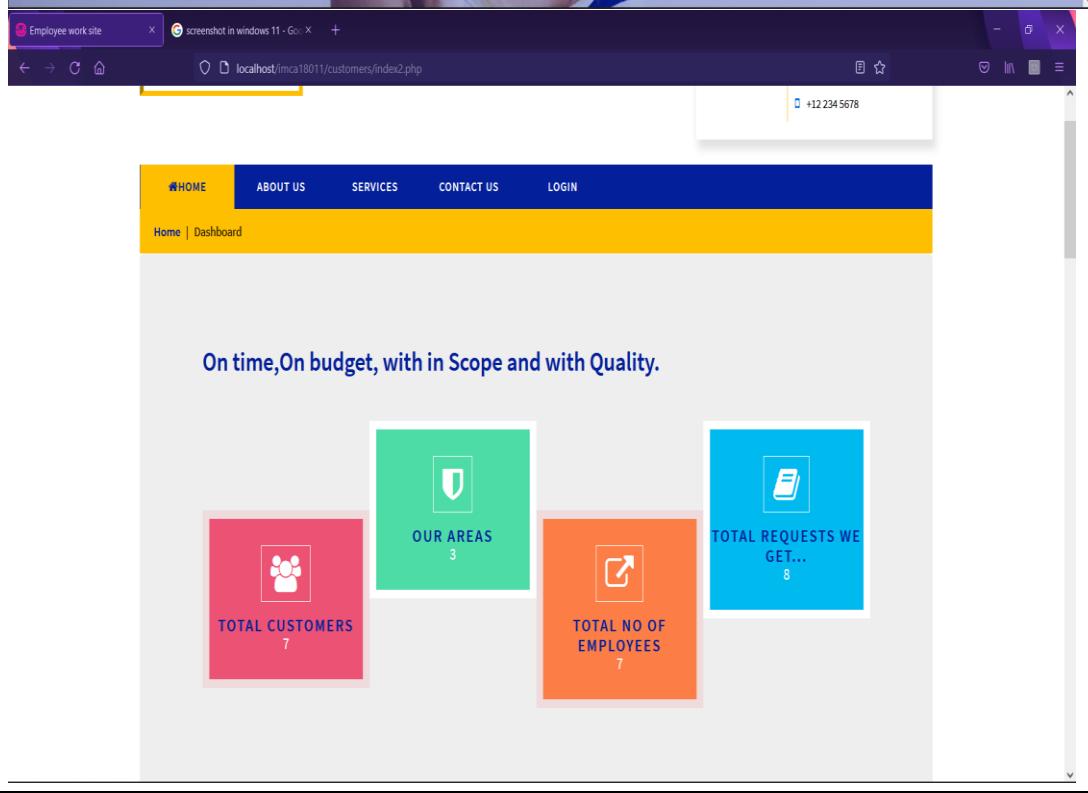
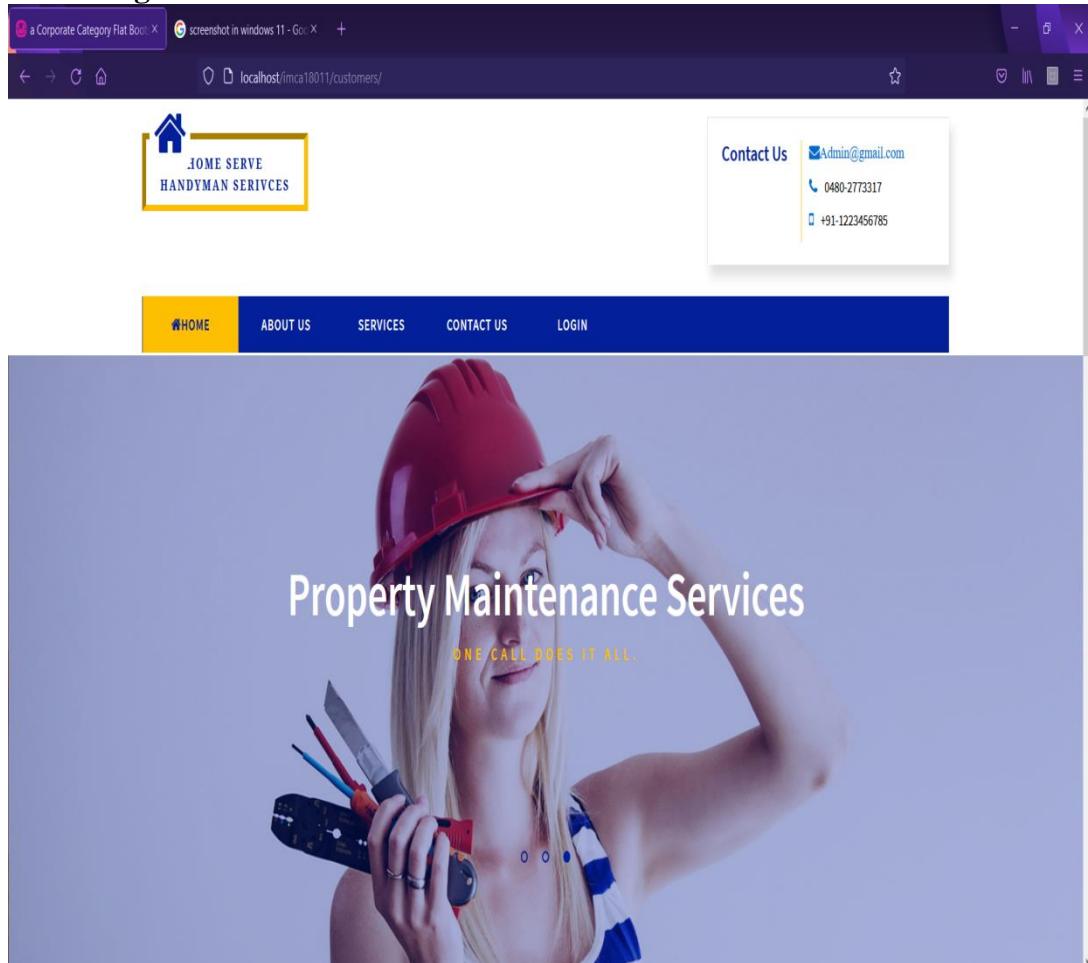
Description About Him: ziczzasa

Experience: 1

Back Print

CUSTOMER

- **Home Page**



The image shows two screenshots of a website, likely a local handyman service provider.

Top Screenshot: Our Services

The title "Our Services" is centered at the top. Below it are five service categories, each with an image and a brief description:

- plumber**: A person working on a kitchen sink.
- carmechanic**: A person working under the hood of a car.
- painter**: A person painting a wall.
- welder**: A person welding metal.
- Electrician**: A person working on electrical wiring inside a building.

Bottom Screenshot: Our Areas

The title "Our Areas" is centered at the top. Below it are three area names, each with an image and a brief description:

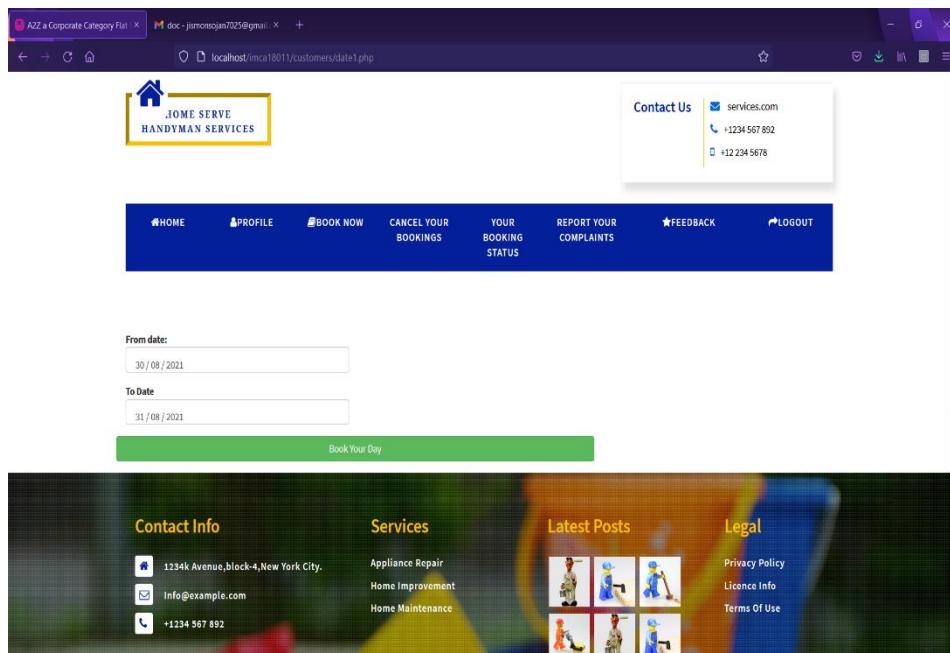
- Chalakkudy**: An aerial view of a river flowing through a lush green landscape.
- Angamaly**: An aerial view of a town with red-roofed buildings and surrounding greenery.
- Aluva**: An aerial view of a coastal area with a bridge over water and green surroundings.

- **Login Page**

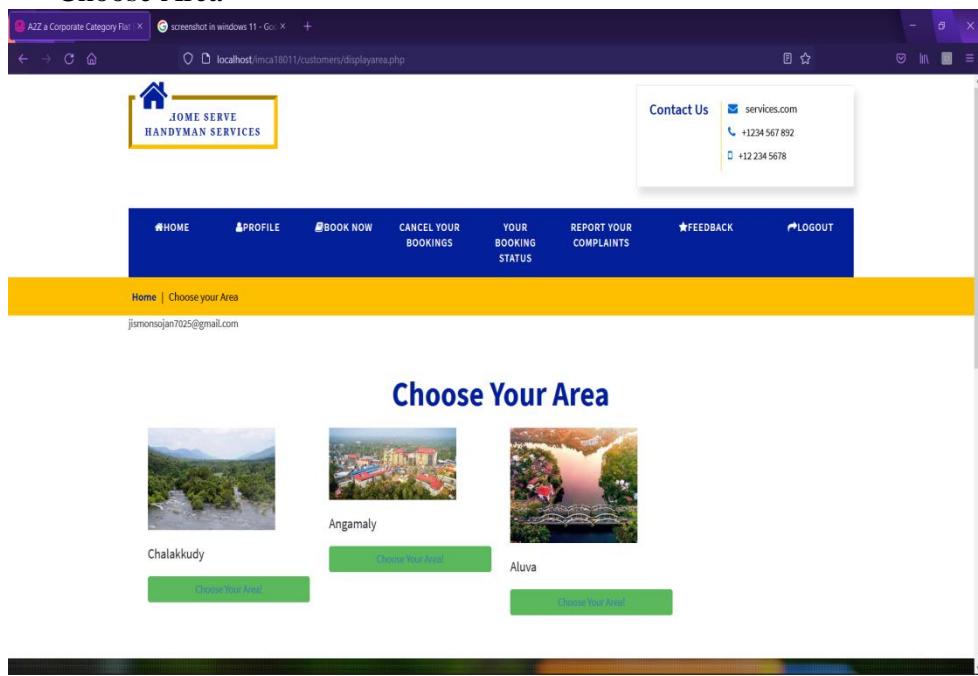
- **Registration Page**

- **Booking Page**

- Selecting Date



- Choose Area



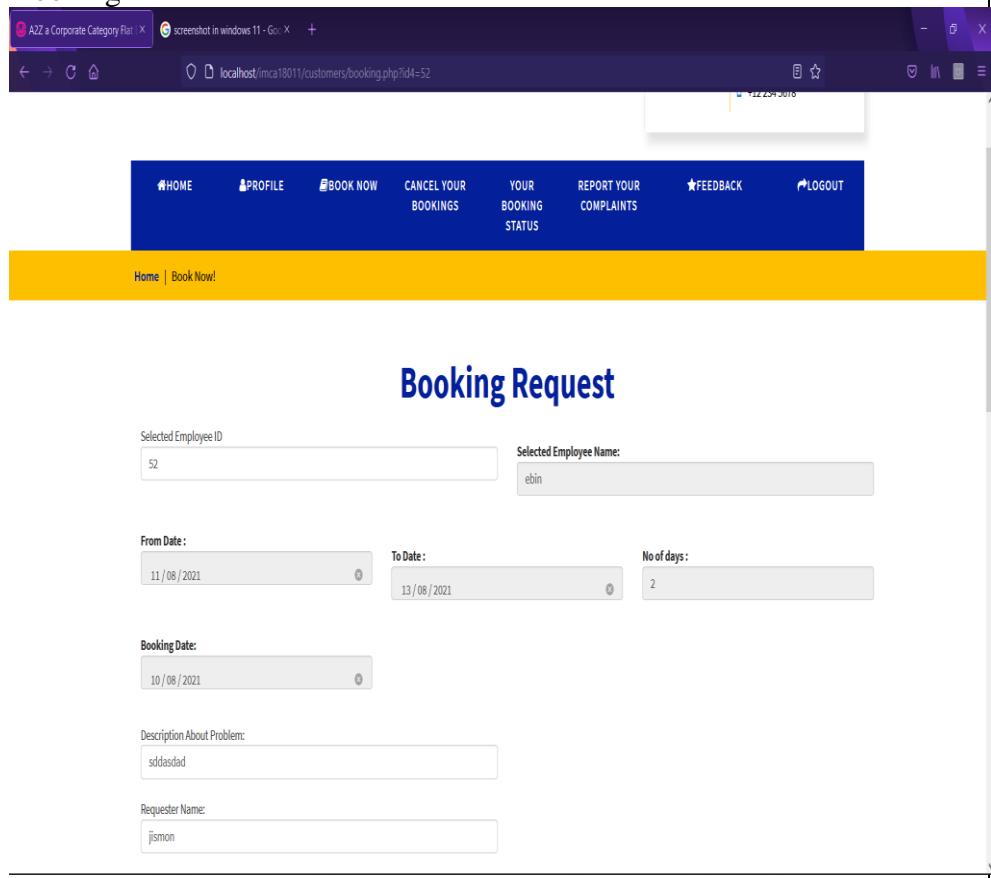
- Choose Services

The screenshot shows a web browser window with the URL localhost/mca18011/customers/displayservices.php?id=11. The page has a blue header bar with navigation links: HOME, PROFILE, BOOK NOW, CANCEL YOUR BOOKINGS, YOUR BOOKING STATUS, REPORT YOUR COMPLAINTS, FEEDBACK, and LOGOUT. A yellow navigation bar below it says "Home | Choose Your Service" and shows the email jismorsiyan7025@gmail.com. On the right, there's a "Contact Us" sidebar with an envelope icon, the email services.com, and two phone numbers: +1234567892 and +12345678. The main content area features a title "Choose Your Service!" above four service categories: plumber, car mechanic, painter, and welder. Each category includes a small image, a brief description, and a green "Choose Your Services!" button.

- Choose Available Employee

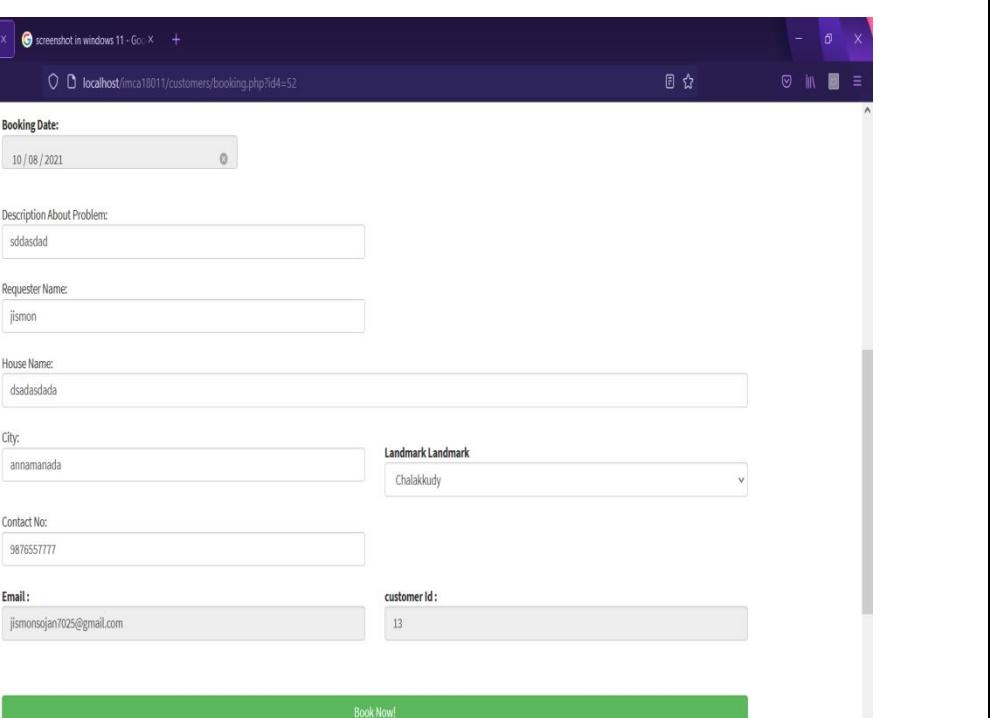
The screenshot shows a web browser window with the URL localhost/mca18011/customers/bookaser.php?id2=118. The page has a blue header bar with links: BOOKINGS, BOOKING STATUS, and COMPLAINTS. A yellow navigation bar below it says "Home | Choose Your employee". The main content area features a title "Choose Your Employee!" above a profile picture of a man named ebin. Below the picture, his details are listed: Name: ebin, Age: 22, Description: zxzxaxsa, and Experience: 1 years. A green "Now it's Time For Booking!" button with a "BOOK NOW" link is at the bottom. The page has a decorative footer banner at the bottom.

- Booking



The screenshot shows the 'Booking Request' page with the following fields filled:

- Selected Employee ID: 52
- Selected Employee Name: ebin
- From Date: 11/08/2021
- To Date: 13/08/2021
- No of days: 2
- Booking Date: 10/08/2021
- Description About Problem: sddasdad
- Requester Name: jismon



The screenshot shows the 'Booking Request' page with the following fields filled:

- Booking Date: 10/08/2021
- Description About Problem: sddasdad
- Requester Name: jismon
- House Name: dsadasdada
- City: annamanada
- Landmark Landmark: Chalakkudy
- Contact No: 9876557777
- Email: jismonsojan7025@gmail.com
- customer Id: 13

A large green button at the bottom right is labeled 'Book Now!'

- Booking Report

The screenshot shows a web application interface for managing bookings. At the top, there's a header with a logo for 'HOME SERVE HANDYMAN SERVICES', navigation links like 'HOME', 'PROFILE', 'BOOK NOW', 'CANCEL YOUR BOOKINGS', 'YOUR BOOKING STATUS', 'REPORT YOUR COMPLAINTS', 'FEEDBACK', and 'LOGOUT'. Below the header is a yellow navigation bar with 'Home | Track your Status'. The main content area is titled 'Your Bookings' and displays a table of bookings:

Customer Id	Problem_desc	Requester	Employee Name	From Date	To Date	Status	
13	sddasdad	jismon	ebin	2021-08-11	2021-08-13	pending	<button>Booking Report</button>
13	sdsa	sdsd	hithesh	2021-05-27	2021-05-29	pending	<button>Booking Report</button>
13	wdsdsad	asdadasd	ebin	2021-05-12	2021-05-13	pending	<button>Booking Report</button>
13	sdfsfssddf	jismons	hithesh	2021-04-27	2021-04-30	accept	<button>Booking Report</button>
13	asdg	kdsakh	mariya	2021-04-19	2021-04-27	accept	<button>Booking Report</button>

The screenshot shows a 'Booking Request' page. At the top, it says 'localhost/imca18011/customers/bookeReport.php?id=16'. The main content includes a heading 'Home Services' and a section for 'Booking Request'. It displays the following customer information:

Dear Customer,
Thank you for getting your booking at our website. We would like to know how we performed.

Customer Name : jismon
Problem Description : sddasdad
Housename : dsadasdada
City : annamanada
Contact Number : 9876557777
From Date : 2021-08-11
To Date : 2021-08-13
No of Days of Work : 2

Buttons for 'print' and 'Back' are at the bottom.

- Add Review and rating

Dear Customer,
Thank you for getting your booking at our website. We would like to know how we performed. Please spare some moments to give us your valuable feedback as it will help us in improving our service.

Please rate your service experience for the following parameters

1. Your overall experience with us ?

2. Experience with our Employee

3. Any Other suggestions:

improve service

EMPLOYEE

- Registration Page

First Name: ebin

Last Name: baby

Address: kottakkal

Gender: Male Female

Date of Birth: 10 / 04 / 2000

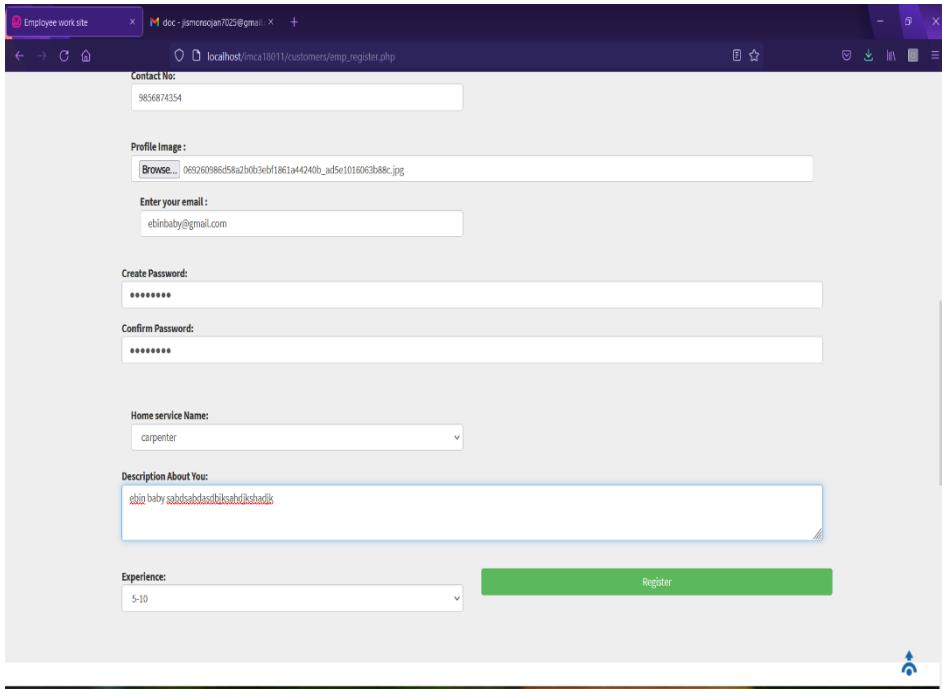
Age: 23

Location Name: Alva

City: Dhesham

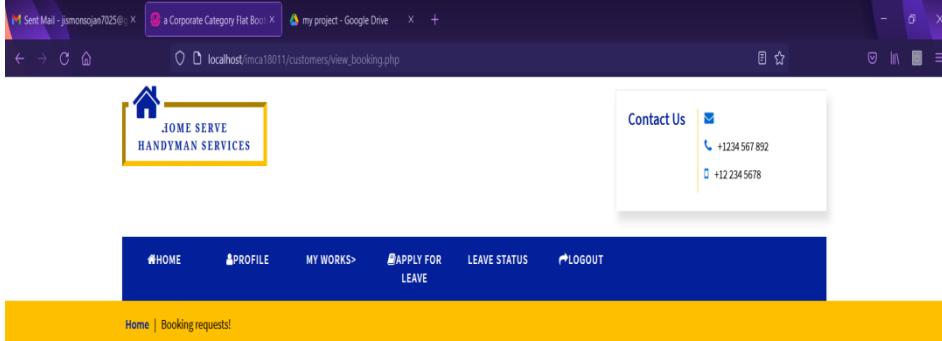
Contact No: 9856874354

Profile Image: 069260986d58a2b0b3ebf1861a44240b_ad5e1016063b88c.jpg



The screenshot shows a web browser window titled "Employee work site" with the URL "localhost/mca18011/customers/emp_register.php". The form fields include:

- Contact No: 9856874354
- Profile Image: A file input field containing the path "069260986d58a2b0b3ebf1861a4240b_af5e1016063b88c.jpg".
- Enter your email: ebinbaby@gmail.com
- Create Password: A password input field containing "*****".
- Confirm Password: A password input field containing "*****".
- Home service Name: A dropdown menu showing "carpenter".
- Description About You: A text area containing "john baby saldsabdasitkashifishadik".
- Experience: A dropdown menu showing "5-10".
- A green "Register" button.



The screenshot shows a web browser window titled "my project - Google Drive" with the URL "localhost/mca18011/customers/view_booking.php". The page features a logo for "HOME SERVE HANDYMAN SERVICES" and a "Contact Us" section with phone numbers "+1234567892" and "+122345678". The navigation bar includes links for HOME, PROFILE, MY WORKS>, APPLY FOR LEAVE, LEAVE STATUS, and LOGOUT. The main content area displays a yellow banner "Home | Booking requests!" and the heading "My Bookings!". It shows a welcome message "Welcome ebinbaby@gmail.com" and a table of booking requests:

Customer id	Problem_desc	Requester	House Name	City	Contact No	From Date	To Date	No of Days	Booking status	Accept	Reject
13	sddasdad	jismon	dsadasdada	annamanada	9876557777	2021-08-11	2021-08-13	2	booked	<button>Accept</button>	<button>Reject</button>
13	wdfsdsad	asdsadasd	adsadasd	dsadasd	4444444444	2021-05-12	2021-05-13	1	booked	<button>Accept</button>	<button>Reject</button>

Below the table is a text input field labeled "Your ID:" with the value "52".

The screenshot shows a web application interface for managing work status. At the top, there's a navigation bar with links for HOME, PROFILE, MY WORKS, APPLY FOR LEAVE, LEAVE STATUS, and LOGOUT. A yellow banner below the navigation bar displays the text "Home | Work Status!". The main content area is titled "My Work Status" and welcomes the user "ebinbaby@gmail.com". Below this, there's a table showing booking details:

Customer id	Problem_desc	Requester	House Name	City	Contact No	From Date	To Date	No of Days	Booking status
13	sddasdad	jismon	dsadasdada	annamanada	9876557777	2021-08-11	2021-08-13	2	booked

Below the table, there's a search bar labeled "Your ID:" containing the value "52".

12.3 REPORTS

Booking Report

- Received Booking Report

The screenshot shows a report page titled "Received Booking Report". It features a table with booking history data:

Customer id	Problem_desc	Requester Name	Employee id	From Date	To Date	Current work status
13	sddasdad	jismon	52	2021-08-11	2021-08-13	accept
13	sdsa	sdasd	54	2021-05-27	2021-05-29	pending
13	wdsdsad	asdsadasd	52	2021-05-12	2021-05-13	pending
18	fsfsdfsd	martin	58	2021-05-11	2021-05-14	pending
13	sdfsdfdf	jismons	54	2021-04-27	2021-04-30	accept
13	wererw	anjaly	52	2021-04-24	2021-04-26	pending
13	asd	kdsakh	55	2021-04-19	2021-04-27	accept
13	not work	mary	57	2021-04-19	2021-04-21	complete
13	dasd	sdsada	52	2021-04-16	2021-04-19	pending

At the bottom of the report table, there are two buttons: "print" and "excel".

- **Accept Booking Report**

The screenshot shows a web browser window with three tabs: 'Sent Mail - jismonsojan7025@gmail.com', 'Employee work site', and 'my project - Google Drive'. The main content area has a header 'HOME SERVE HANDYMAN SERVICES' with a logo. Below it is a navigation bar with 'DASHBOARD', 'COMPLAINTS REPORTS', 'LEAVE REPORTS', and 'Booking Reports' (which is highlighted). The main section is titled 'Accepted Booking Reports' and contains a table with the following data:

Customer Name	Problem description	Requester name	Employee Name	From Date	To Date	Status
jismon	asdg	ksdsakh	mariya	2021-04-19	2021-04-27	accept
jismon	sdfsfssdff	jismons	hithesh	2021-04-27	2021-04-30	accept
jismon	sddasdad	jismon	ebin	2021-08-11	2021-08-13	accept

At the bottom left are 'Back' and 'print' buttons.

- **Reject Booking Report**

The screenshot shows a web browser window with three tabs: 'Sent Mail - jismonsojan7025@gmail.com', 'Employee work site', and 'my project - Google Drive'. The main content area has a header 'HOME SERVE HANDYMAN SERVICES' with a logo. Below it is a navigation bar with 'DASHBOARD', 'COMPLAINTS REPORTS', 'LEAVE REPORTS', and 'Booking Reports' (which is highlighted). The main section is titled 'Rejected Booking Reports' and contains a table with the following data:

Customer Name	Problem description	Requester	Employee Name	From Date	To Date	Status
jismon	wdsdsad	asdsadasd	ebin	2021-05-12	2021-05-13	reject

At the bottom left are 'Back' and 'print' buttons.

- **Completed Booking Report**

The screenshot shows a web browser window with three tabs: "Sent Mail - jismonojan7025@gmail.com", "Employee work site", and "my project - Google Drive". The main content area has a header "HOME SERVE HANDYMAN SERVICES" with a house icon. Below it is a navigation bar with "DASHBOARD", "COMPLAINTS REPORTS", "LEAVE REPORTS", and "Booking Reports" (which is highlighted). The main title is "Completed Booking Reports". A table displays one row of data:

Customer Name	Problem description	Requester	Employee Name	From Date	To Date	Status
jismon	not work	mary	nevin	2021-04-19	2021-04-21	complete

Buttons at the bottom are "print" (blue) and "Back" (green).

- **Pending Booking Report**

The screenshot shows a web browser window with three tabs: "Sent Mail - jismonojan7025@gmail.com", "Employee work site", and "my project - Google Drive". The main content area has a header "HOME SERVE HANDYMAN SERVICES" with a house icon. Below it is a navigation bar with "DASHBOARD", "COMPLAINTS REPORTS", "LEAVE REPORTS", and "Booking Reports" (which is highlighted). The main title is "Pending Booking Reports". A table displays one row of data:

Customer Name	Problem description	Requester	Employee Name	From Date	To Date	Status
jismon	sdsd	sdsd	hithesh	2021-05-27	2021-05-29	pending

Buttons at the bottom are "Back" (green) and "print" (blue).

- **Cancel Booking Report**

The screenshot shows a web browser window with three tabs: 'Sent Mail - jismonojan7025@gmail.com', 'Employee work site', and 'my.project - Google Drive'. The main content area has a header 'HOME SERVE HANDYMAN SERVICES' with a house icon. Below it is a navigation bar with 'DASHBOARD', 'COMPLAINTS REPORTS', 'LEAVE REPORTS', and 'Booking Reports' (which is highlighted). The main section is titled 'Cancel Booking Reports' and contains a table with the following data:

Customer Name	Problem description	Requester	Employee Name	From Date	To Date	Status
jismon	dasd	sdsada	ebin	2021-04-16	2021-04-19	cancel
jismon	wererw	anjaly	ebin	2021-04-24	2021-04-26	cancel
martin	fsfsdfsdfsd	martin	raju	2021-05-11	2021-05-14	cancel

At the bottom left are two buttons: 'Back' and 'print'.

- **Datewise Booking Report**

The screenshot shows a web browser window with three tabs: 'Employee work site', 'doc - jismonojan7025@gmail.com', and 'localhost/imca18011/admin/datewise.php'. The main content area has a header 'HOME SERVE HANDYMAN SERVICES' with a house icon. Below it is a navigation bar with 'DASHBOARD', 'COMPLAINTS REPORTS', 'LEAVE REPORTS', and 'Booking Reports' (which is highlighted). The main section is titled 'CUSTOMER BOOKING DATES REPORT' and contains a form with the following fields:

From: To: Sort:

The screenshot shows a web browser window with the URL `localhost/imca18011/admin/datewise.php`. The page title is "HOME SERVE HANDYMAN SERVICES". The navigation bar includes links for DASHBOARD, COMPLAINTS REPORTS, LEAVE REPORTS, and Booking Reports. The main content area is titled "CUSTOMER BOOKING DATES REPORT" and contains a table with the following data:

Customer Name	Problem description	Requester name	Employee Name	From date	To date	status
jismon	asdgi	ksdsakh	mariya	2021-04-19	2021-04-27	accept
jismon	not work	mary	nevin	2021-04-19	2021-04-21	complete
jismon	sdffsfddf	jismons	hithesh	2021-04-27	2021-04-30	accept
jismon	wdsdsad	asdsadasd	ebin	2021-05-12	2021-05-13	reject
jismon	sdsd	sdsd	hithesh	2021-05-27	2021-05-29	pending

Leave Report

The screenshot shows a web browser window with the URL `localhost/imca18011/admin/leavereports.php`. The page title is "HOME SERVE HANDYMAN SERVICES". The navigation bar includes links for DASHBOARD, COMPLAINTS REPORTS, LEAVE REPORTS, and Booking Reports. The main content area is titled "Employee leave Reports" and contains a table with the following data:

Sl No	Employee name	From Date	To Date	Reason	Status
1	mariya	2021-05-11	2021-05-15	fever	cancel
2	mariya	2021-05-04	2021-05-07	aaaa	reject
3	mariya	2021-05-07	2021-05-10	marriage	approve

Employee leave Reports

Sl No	Employee name	From Date	To Date	Reason	Status
1	mariya	2021-05-11	2021-05-15	fever	cancel
2	mariya	2021-05-04	2021-05-07	aaaa	reject
3	mariya	2021-05-07	2021-05-10	marriage	approve

Back print

Customer Complaint Report

The screenshot shows a web browser window with the URL localhost/mca18011/admin/complaintreports.php. The page title is "Customer Complaint Reports". At the top, there is a navigation bar with tabs: DASHBOARD, COMPLAINTS REPORTS (which is active), LEAVE REPORTS, and Booking Reports. Below the navigation bar is a table titled "Customer Complaint Reports" with the following data:

Sno	Customer name	Employee name	Complaint Description	Date	Reply
1	jismon	nevin	bad behaviour	2021-05-04	xaasdas
2	jismon	nevin	lgfgfjg	2021-05-05	sorry for this
3	jismon	nevin	sadsad	2021-05-05	we will take an action of this
4	jismon	nevin	vgvnbv	2021-05-05	ok we solve it
5	jismon	nevin	AAAA	2021-05-05	
6	jismon	nevin	bad behaviour	2021-05-16	

At the bottom left of the table, there are two buttons: "Back" and "print".

BIBLIOGRAPHY

❖ Web Sites

[1] UML Diagrams

<http://en.wikipedia.org/wiki/umldiagrams>

[2] Basics of PHP

<http://javatpoint.com>

[3] Feasibility Study

<http://en.wikipedia.org/wiki/Feasibilitystudy>

[4] System Testing

<http://en.wikipedia.org/wiki/Systemtesting>

[5] MySQL

<http://www.mysql.com>

❖ Books

[1] (Mario Larig) Php Reference : Beginner to intermediate PHP5

[2] (Luke Welling) Php and MySQL development

[3] (Lee) Introduction to system analysis and design ,Second edition

[4] (Roger S. Pressman) Software Engineering