

# **RKDF INSTITUTE OF SCIENCE & TECHNOLOGY**

## **Major Project Report**

Submitted in Partial fulfillment for the award of Graduate Degree in Bachelor of Engineering in  
Computer Science & Engineering

**Submitted to**

**SRK UNIVERSITY, BHOPAL (M.P)**



Submitted By

**Rajkumar Kannaujiya**

Under the Guidance of

**Rekha Kumari Kannaujiya**

(Department of Computer Science & Engineering)

**RKDF INSTITUTE OF SCIENCE AND TECHNOLOGY**  
**BHOPAL (M.P)**

# **R.K.D.F. INSTITUTE OF SCIENCE AND TECHNOLOGY BHOPAL (M.P.)**

## ***Department of Computer Science & Engineering***



### **DECLARATION**

I hereby declare that the work, which is being presented in the report, entitled “Online Service Management System ( OSMS)” partial fulfillment of the requirements for the award of degree of Bachelor of Engineering, submitted in the Computer Science & Engineering department of RKDF Institute of Science and Technology is an authentic record of my own work carried under the guidance Ms.. Rekha Kumari Kannaujiya . I have not submitted the matter embodied in this report for award of any other degree.

**Name of the Student**

**Rajkumar Kannaujiya**

**(Enrollment No.)**

**SRKUCS018028**

**R.K.D.F. INSTITUTE OF SCIENCE AND TECHNOLOGY  
BHOPAL (M.P.)**

***Department of Computer Science & Engineering***



**CERTIFICATE**

*This is to certify that the Major Project entitled “Bus ticket booking System” submitted to Sarvepalli Radhakrishnan University, Bhopal (M.P.) in the Department of Computer Science & Engineering by Rajkumar Kannaujiya (SRKUCS018028) is a partial fulfillment of the requirement for the award of the degree of Bachelor of Engineering in Computer Science & Engineering .*

***Rekha kumari kannaujiya***  
***Guide***

***Dr. Varsha Namdeo***  
***Head of the Department***

***Dr. M. K. Chopra***  
***Principal / RKDFIST***

# **R.K.D.F. INSTITUTE OF SCIENCE & TECHNOLOGY, BHOPAL (M.P.)**

***Department of Computer Science & Engineering***



## **APPROVAL CERTIFICATE**

This is to certify that the Major Project entitled “**Online Service Management System (OSMS)**” submitted to *Sarvepalli Radhakrishnan University*, Bhopal (M.P.) in the department of ***Computer Science & Engineering*** by **Rajkumar Kannaujiya (SRKUCS018028)** in a partial fulfillment of the requirement for the award of the degree of Bachelor of Engineering in Computer Science & Engineering.

**Internal Examiner**

**External Examiner**

## ACKNOWLEDGEMENT

It is my immense pleasure to express my deep sense of gratitude and indebtedness to my highly respected and esteemed guide **Rekha Kumari Kannaujiya** Associate Professor, and **Dr. M. K. Chopra**, Director, R.K.D.F. Institute of Science & Technology, Bhopal. Their invaluable guidance, inspiration, constant encouragement sincere criticism and sympathetic attitude could make this project possible.

I extend my deep sense of gratitude to **Dr. Varsha Namdeo**, Head of the Department of Computer Science and Engineering, R.K.D.F. Institute of Science & Technology, Bhopal for providing encouragement, guidance and support during my project work.

I am deeply grateful to **Ms. Shikha Choudhary**, Dean, R.K.D.F. Institute of Science & Technology, Bhopal for her constant encouragement and inspiration and providing me with the required resources from the college.

I extend my thanks to all the staff members of the department who extended their co-operation and help throughout the project work. And last but not least, I express my gratitude to my parents, who have been a constant source of inspiration.

**Rajkumar Kannaujiya**  
**SRKUCS018028**

## INDEX

<b>Chapter 1.....</b>	<b>1</b>
<b>Introduction.....</b>	<b>1</b>
1.1    Overview.....	1
1.2    Objectives.....	2
<b>Chapter 2.....</b>	<b>3</b>
<b>System Analysis.....</b>	<b>3</b>
2.1    Software Requirements Specification(SRS).....	4
2.1.1    Data Gathering .....	4
2.1.2    Feasibility Study .....	4
2.1.3    Software Process Model .....	7
2.2    Data Flow Digrams (DFD) .....	10
2.2.1    DFD 0 Level .....	10
2.2.2    DFD 1 Level .....	11
2.2.3    DFD 2 Level .....	12
2.3    Entity Relationship Digram (Er-Digram) .....	13
2.4    Flow Chart .....	14
2.5    Gantt Chart .....	17
<b>Chapter 3.....</b>	<b>18</b>
<b>System Design .....</b>	<b>18</b>
3.1    Input to Project .....	18
3.2    Output to Project .....	21
3.3    Modularization.....	21
3.4    Class Digram .....	26
3.5    Process Logic .....	27

3.6	Data Integrity .....	28
3.7	Data Dictionary .....	29
<b>Chapter 4</b> .....		36
<b>Tool /Environment</b> .....		36
4.1	Hardware Requirements .....	36
4.2	Software Requirements .....	36
<b>Chapter 5</b> .....		37
<b>Software Description</b> .....		37
5.1	Visual Studio Code .....	37
5.2	Languages .....	37
5.2.1	HTML .....	37
5.2.2	CSS .....	38
5.2.3	JAVA SCRIPT .....	38
5.2.4	Why Using PHP .....	38
5.3	Why Using MySQL .....	39
5.4	Yed Graph Editor .....	39
5.5	Star UML .....	40
<b>Chapter 6</b> .....		41
<b>Program Code</b> .....		41
<b>Chapter7</b> .....		107
<b>Testing</b> .....		107
7.1	Type Of Testing .....	108
7.2	Use Case .....	110
7.3	Test Cases .....	113

<b>Chapter 8.....</b>	<b>114</b>
<b>Implementation .....</b>	<b>114</b>
<b>Chapter 9.....</b>	<b>115</b>
<b>Limitations .....</b>	<b>115</b>
<b>Chapter 10.....</b>	<b>116</b>
<b>Future Scope .....</b>	<b>116</b>
<b>Chapter 11.....</b>	<b>117</b>
<b>Conclusion .....</b>	<b>117</b>
<b>Bibliography .....</b>	



## Chapter 1: Introduction

---

OSMS is India's leading chain of multi-brand Electronics and Electrical service workshops offering wide array of services. We focus on enhancing your uses experience by offering world-class Electronic Appliances maintenance services. Our sole mission is "To provide Electronic Appliances care services to keep the devices fit and healthy and customers happy and smiling": With well-equipped Electronic Appliances service centres and fully trained mechanics, we provide quality services with excellent packages that are designed to offer you great savings. Our state-of-art workshops are conveniently located in many cities across the country.

### 1.1 Overview

Through computers organization. IT Firms, Universites and Businesses etc.. can transact to their clients in a convenient way using advance technologies and specially the Web Application. The Web Application is created for Service Centre functions through this "Online Service Managment System" Admin (Manager) can lessen their errors and efforts in every Customer Supportprocessing and transactions and in making reports. It is Web Application which is developed in HTML, CSS, PHP & MySQL. The purpose of this Web Application is to manage the activities of service centre. Even a person can handle very easily; it means Web Interface is user friendly.

### 1.2 Objective

•**Practicality:** The software must be stable and can be operated by people with average intelligence.

- **Efficiency:** This involves accuracy, timeliness and comprehensiveness output.
- **Cost:** It is desirable to aim for a system with a minimum cost subject the condition that it must satisfy all the requirements.
- **Portability:** The Web Application should be portable to all environments.
- **Security:** This important aspect of design covers areas of physical security of data. This could be provided by a login facility enabling username and password for the user and administrator. Thus it makes the Admin work simple with 100% efficiently.

## Chapter 2: System Analysis

---

System Analysis is a problem solving technique that improves the system and ensures that all the components of the system work efficiently to accomplish their purpose.

A systems analyst researches problem, plans solutions, recommends software and systems, and coordinates development to meet business or other requirements. The main goal of this system analyst is to collect different data from different site, process these data and generate progress as well as daily report.

System analyst operates in a dynamic environment where change is a way of life. The environment may be a business firm, a business application, or a computer system. to construct a system the following key elements must be considered:-

**Input:** Input is what data the system receives to produce a certain output.

**Output:** What goes out from the system after being processed is known as

**Output. Processing:** The process involved to transform input into output is known as Processing

**Control:** In order to get the desired results it is essential to monitor and control the input, Processing and the output of the system. This job is done by the control.

**Feedback:** The Output is checked with the desired standards of the output set and the necessary steps are taken for achieving the output as per the standards, this process is called as Feedback. It helps to achieve a much better control in the system.

**Boundaries:** The boundaries are nothing but the limit of the system. Setting up boundaries helps for better concentration of the actives carried in the system.

**Environment:** The things outside the boundary of the system are known as environment. Change in the environment affects the working of the system.

**Interfaces:** The interconnections and the interactions between the sub-systems are known as the Interfaces. They may be inputs and outputs of the systems.

## **2.1 Software Requirements Specification (SRS)**

A software requirements specification is a document that captures complete description about how the system is expected to perform. It is usually signed off at the end of requirements engineering phase.

### **Product perspective**

The software product is a Web application. The application will be made up of two parts, one administrator who has all the rights and the other user who has limited rights to handle the application. The two users of the system, namely the Service Manager (Admin) and Customers (User) interact with the system in different ways.

### **Product Functions**

First of all it will authenticate the user whether he is Admin or User the unauthorized person can't get access to the application. The Admin will be able to Add, delete, and modify Product details. He can also Add, delete and modify Service Request made by Customers. He can use this application to check all reports related to Product Sell, and assign work order as well as he can manipulate the data of Repair Request.

### **Safety Requirement**

For preventing unauthorized access to the application, this application have login feature so only granted user can access with defined rights.

### **2.1.1 Data Gathering**

Data collection is the systematic approach to gathering and measuring information from a variety of sources to get a complete and accurate picture of an area of interest. Data collection enables a person or organization to answer relevant questions, evaluate outcomes and make predictions about future probabilities and trends. Accurate data collection is essential to maintaining the integrity of research making informed business decisions and ensuring quality assurance.

### **2.1.2 Feasibility study**

Feasibility study means to check whether the project is feasible or not.that means possible or not. Some feasibility study regarding this project is as follows:

### **Economic Feasibility**

The project has shown the economic feasibility by the study of the fact that by using this software the increased number of the customers can be given service effectively and efficiently and can save a lot time and saving time means saving money. The cost and benefit analysis has shown that cost that have incurred in developing the project is less than the benefits that the project is going to provide once it is developed, so this project has passed the feasibility test.

### **Technical Feasibility**

Technical feasibility centers on the existing computer system (Hardware, Software etc.) and to what extent it supports the existing system. As the existing system computer system is viable so there is no matter of technical feasibility that is the system is technically feasible. In this type of feasibility study it is checked whether there is a need of new hardware/software or not. What are the basic requirements of the project? If there is need then how it can be fulfilled. In this context, this project doesn't need any special hardware or software.

### **Behavioral Feasibility**

The User also interested in this project, as it will help them to do work with ease and efficiently without complexity, so they supported the development of this project with full enthusiasm. This shows the behavioral feasibility of the project.

### **Time Feasibility**

It is the determination of whether a proposed project can be implemented fully within stipulated time frame. The project was decided to be done in three months and was thought to be feasible

### **Operational Feasibility**

In this feasibility study it is determined whether there is need of well qualified operator or simple user. Is there need to train the operator or not? This project is supporting the Graphical User Interface; hence operating this project is so simple. Even a person who has a little knowledge of computer can easily handle this well. There is no need of trained operator.

### **2.1.3 Software Process model**

The Software Process Models are the various processes or methodologies that are being selected for the development of the project depending on the project's aims and goals. There are many development life cycle models that have been developed in order to achieve

different required objectives. The models specify the various stages of the process and the order in which they are carried out.

The selection of model has very high impact on the testing that is carried out. It will define the what, where and when of our planned testing, influence regression testing and largely determines which test techniques to use. I

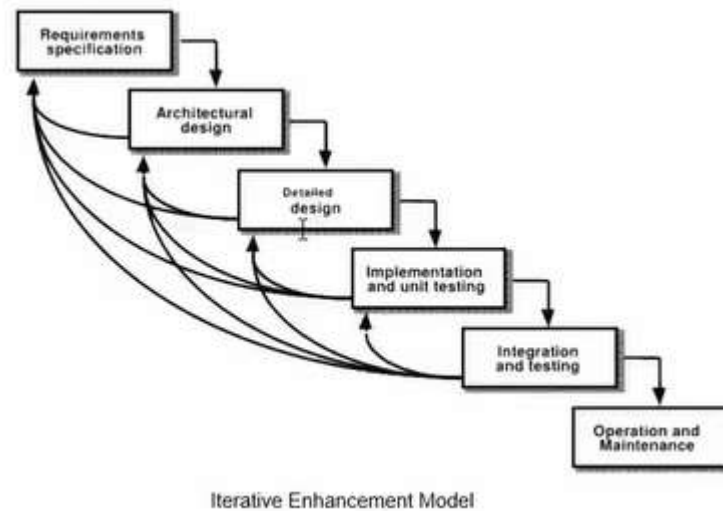
Choosing right model for developing of the software product or application is very important. Based on the model the development and testing processes are carried out.

A Process Model describes the sequence of phases for the entire lifetime of a product. Therefore it is sometimes also called Software Life Cycle. This covers everything from the initial commercial idea until the final de installation or disassembling of the product after its use.



In order to develop the project "Online Service Managment System" we have adopted the Iterative Enhancement Model also known as Incremental Model. This model removes the shortcoming of waterfall model. Since many facts of this system are already known. It is not a new concept and hence no research is required. A working version can be easily created and hence the system can start working. Rest of the functionalities can be implemented in the next iteration and can be delivered later. As the requirement analysis is also not required. It not being a new technology risk involved is also less. So one need not perform detailed risk analysis. If redevelopment Admin is less than development can be started with less number of people and in next increments others can be involved. As this model combines the advantage of waterfall model and prototyping, clients are always aware of the product being delivered and can always suggest changes and enhancements and can get them implemented. As less amount of customer communication is required one need not apply spiral model in

which all types of analysis is done in detail. As the deadline is affordable one need not to for Rapid Application Development model. Iterative enhancement model is useful when less manpower is available for software development and the release deadlines are specified. It is best suited for in house product development, where it is ensured that the user has something to start with. The complete product is divided into releases and the developer delivers the product release by release.

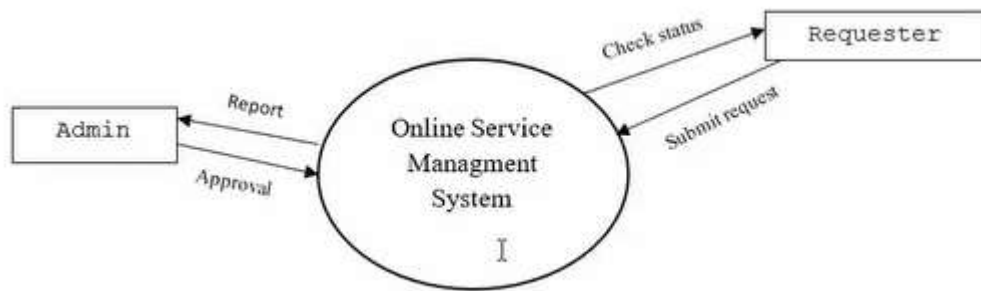


## 2.2 Data Flow Diagram (DFD)

Data flow diagram is graphical representation of flow of data in an information system. It uses defined symbols like rectangles, circles and arrows, plus short text labels, to show data inputs, outputs, storage points and the routes between each destination. Data flowcharts can range from simple, even hand-drawn process overviews, to in-depth, multi-level DFDs that dig progressively deeper into how the data is handled

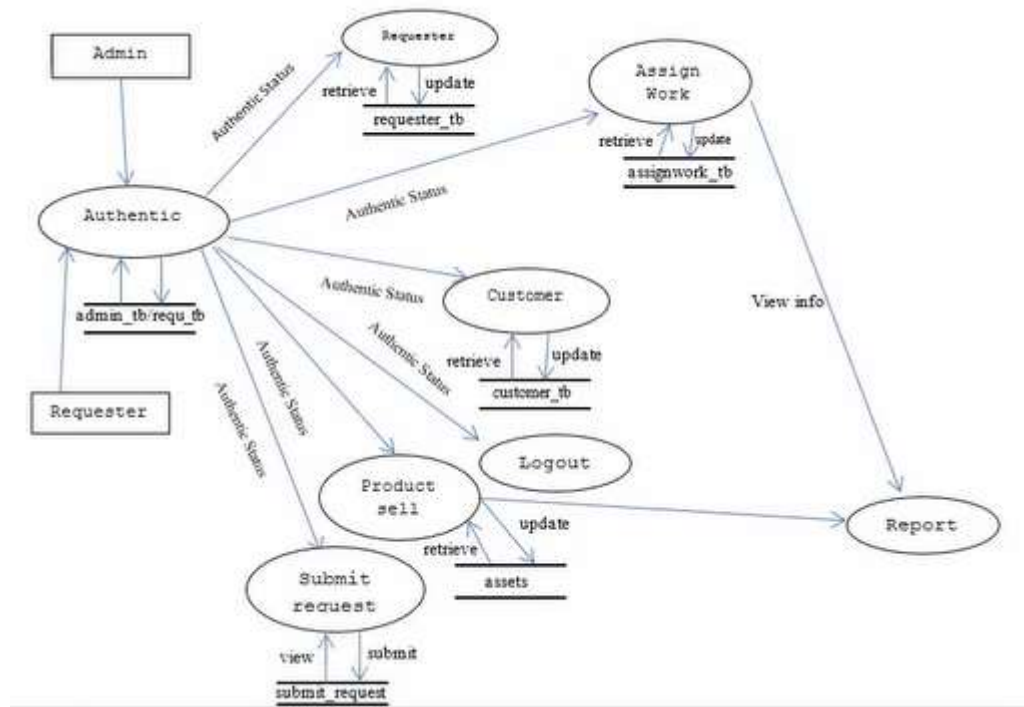
### 2.2.1 DFD 0 Level

The 0 Level DFD shows flow of data of application. DFD Level 0 is also called a Context Diagram. It's a basic overview of the whole system or process being analyzed or modeled.



### 2.2.2 DFD 1 Level

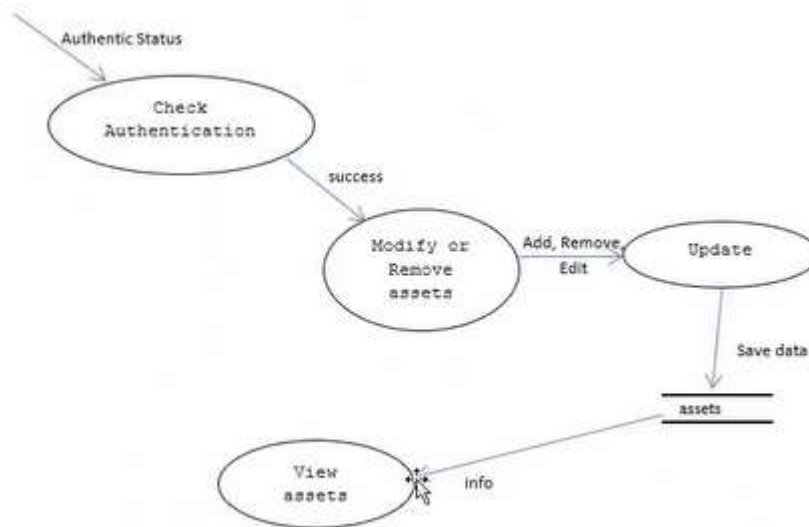
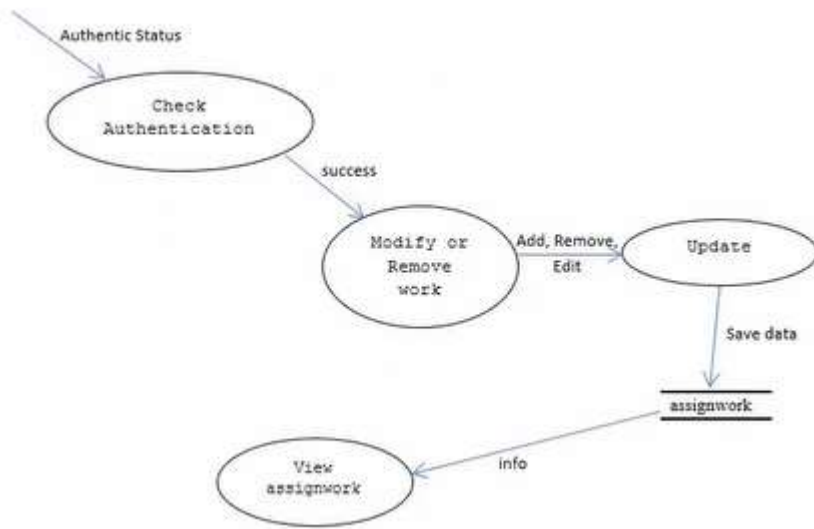
DFD Level 1 provides a more detailed breakout of pieces of the Context Level Diagram. This DFD describes main functions carried out by the system, as we break down the high-level process of the Context Diagram into its sub-processes.



### 2.2.3 DFD 2 Level

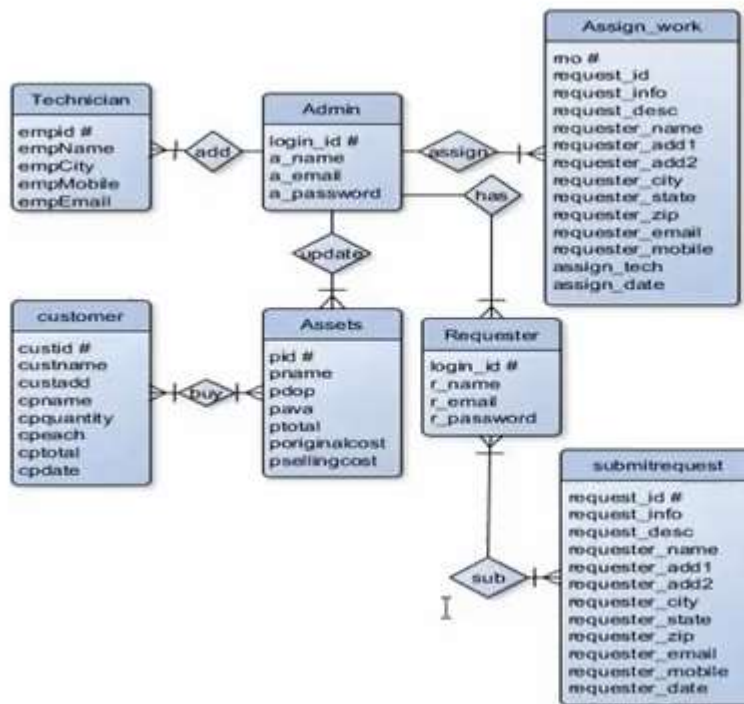
The DFD 2 Level describes flow of data in more detail. DFD Level 2 goes one step deeper into parts of Level 1. It may require more text to reach the necessary level of detail about the system's functioning.





## 2.3 Entity Relationship Diagram (ER-Diagram)

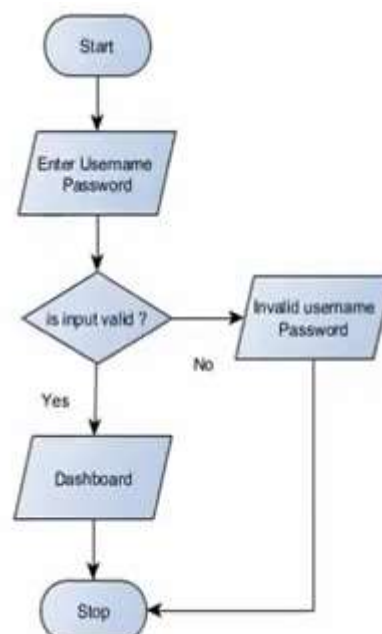
An Entity Relationship Diagram (ERD) is a visual representation of different entities within a system and how they relate to each other. Entity relationship diagrams are used in software engineering during the planning stages of the software project. They help to identify different system elements and their relationships with each other.



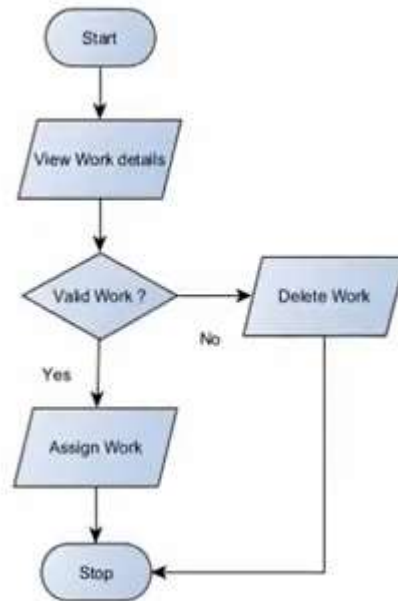
## 2.4 Flow Chart

A flowchart is a diagram that depicts a process, system or computer algorithm. They are widely used in multiple fields to document, study, plan, improve and communicate often complex processes in clear, easy to-understand diagrams. Flowcharts, sometimes spelled as flow charts, use rectangles, ovals, diamonds and potentially numerous other shapes to define the type of step, along with connecting arrows to define flow and sequence.

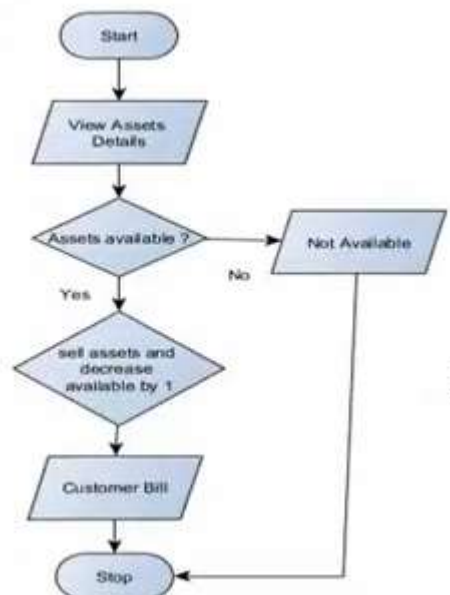
### Login



## Assign Work

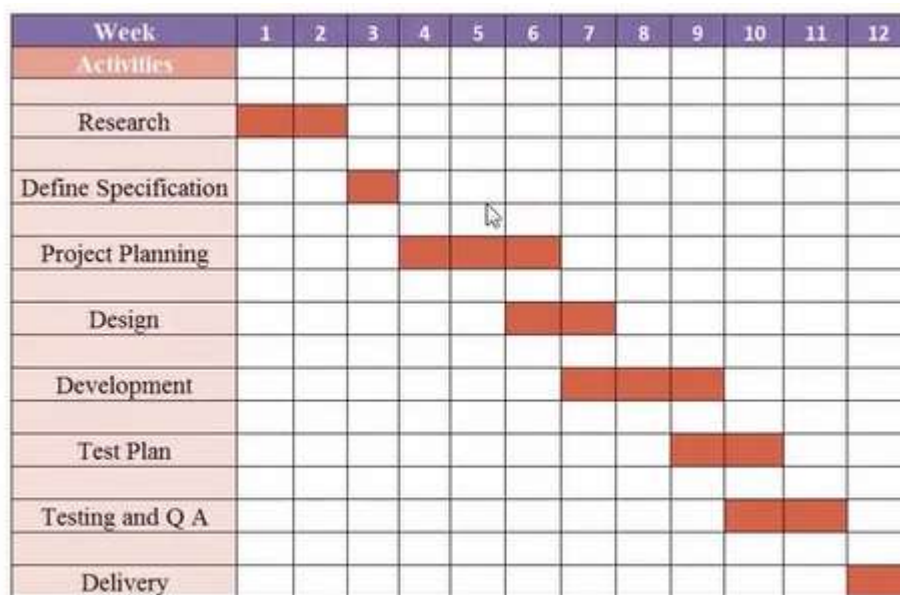


## Sell Product



## 2.5 Gantt Chart

Gantt chart was invented by a mechanical engineer named Henry Gantt in 1910. A gantt chart is simply a type of bar chart that visually represents a project plan over time. It shows start and end dates for tasks, displays milestones, and allows for dependencies between tasks. With all the features of Henry gantt's project management system, it's no wonder that even now, more than 100 years later, the gantt chart is still the preferred tool for managing projects of all sizes and types.



## Chapter 3: System Design

---

The systems design approach first appeared right before World War II, when engineers were trying to solve complex control and communications problems. They needed to be able to standardize their work into a formal discipline with proper methods, especially for new fields like information theory, operations research and computer science in general. System design is the process of defining the elements of a system such as the architecture, modules and components, the different interfaces of those components and the data that goes through that system. It is meant to satisfy specific needs and requirements of a business or organization through the engineering of a coherent and well-running system.

### 3.1 Input to the project

In order to complete the tasks of the Application and to get output by using this application work, there is need of some input based on the work that is to be carried out by using it. Different kinds of input are required for different purposes

**i. Input for new Registration**

Login ID# (Automatically generated)

Requester Name

Requester Email

Requester Password

**ii. Input for new Technician**

Emp ID# (Automatically generated)

Emp Name

Emp City

Emp Mobile

Emp Mobile

**iii. Input for Submit Request**

Request ID # (Automatically generated)

Request Info

Request Desc

Requester Name

Requester Addl.

Requester Add2

Requester City

Requester State

Requester Zip

Requester Email

Requester Mobile

Requester Date

#### **iv. Input for Assign Work**

Rno # (Automatically generated)

Request ID

Request Info

Request Desc

Requester Name.

Requester Add1

Requester Add2

Requester City

Requester Zip

Requester Email

Requester Mobile

Assign\_tech

Assign\_date

#### **v. Input for assets.**

pid #

Product Name

Product date of Purchase

Product available

Product Total

Product Original Cost

Product Selling Cost

#### **vi. Input for getting reports**

Date rang

#### **Vii. Input for login in the application**

Email #

Password

### **3.2 Output to the project**

The project named "**Online Service Managment System**" is being made keeping in mind to solve the activities that are carried out in the Service Centre. By using this Admin can easily do many things

like as:

- i. Maintain assets details easily
- ii. Maintain the requester details easily
- iii. Maintain the Technician details easily
- iv. Can check work order details
- v. Can assign work
- vi. Can check Service Request Status



vii.Can generate Bill

viii. Can generate hard copy output of all product sell

### **3.3 Modularization Detail**

#### **Home:**

When the user click on this button,it will display the other modules and pages of the website such as Services, Registration, Login, Contact, and Admin Login. This module will be used to display the brief introduction of the project and will show the title of the project as well as the name of the developer.

#### **Services:**

This modules describes which services company provides to its customers.

#### **Registration:**

This is the most important module of the Online Service Managment System which provides a Registration form where user/requesters can register themselves and submit Service Requests.

#### **Contact:**

This module contains a contact us form which can be used to send feedback or to communicate with the service provider.

#### **Login:**

This is user login form. When an user clicks on this link a user login form will be appear where user can enter their email id and password for logging in to the user panel.

#### **User Panel:**

#### **Profile:**

User can see their register email id and Name as well as if they wish to change the name, they can update new Name. The Registered Email ID is read only so it can't be altered.

### **Submit Request:**

Using this module user can submit service request. It is necessary to fill up all the details asked in the form. After submitting form user will get an receipt which he can print out.

### **Service Status:**

User can check their service request status by filling up service request id

### **Change Password:**

User can change his/her login password.

### **Logout:**

This Logout and Exit the Application.

Online Service Managment System

### **Admin Login:**

This is Admin login form. When Admin clicks on this link an Admin login form will be appear where admin can enter their email id and password for logging in to the

### **Admin panel.**

### **Dashboard:**

This screen displays overview of work and other stuff like Number of technician and list of requesters.

### **Work Oder:**

This page contains all the assigned request made by users. Admin can view or delete the assigned work as per their need.

### **Requestes:**

This is the most important module of admin panel where admin can assign the work/requests made by users/requesters. If there is any invalid request admin can delete that request

without assigning them.

### **Assets:**

The main work is to accomplish in this module is to add, modify or remove any assets of the Service centre. This contains few sub modules through which works are performed. These are as follows:

**New:** This is used to add new Product Part in the service centre. There is a Plus (+) sign button which is actually New Button.

**Edit:** This sub module is used to modify the existing details of the Product if anything goes changes in their record. There is a Pencil button which is actually Remove Button.

**Remove:** This is used to remove any product from the service

**Sell:** This is used when going to sell a product Admin can also print out a bill for customer.

### **Technician:**

The main work is to accomplish in this module is to add, modify or remove Technician of the Service centre. This contains few sub modules

through which works are performed. These are as follows:

- **New:** This is used to add new Technician details in the service centre. There is a Plus (+) sign button which is actually NewButton.

**Edit:** This sub module is used to modify the existing details of the

Technician if anything goes changes in their record. There is at Pencil button which is actually Remove Button.

**Remove:** This is used to remove Techncian from the service centre.

There is a Trash button which is Remove Button.

**Requester:**

The main work in this module is to add, modify or remove Requesters/Users. This contains few sub modules through which works are performed. These are as follows:

**New:** This is used to add new Requesters details in the service centre database. There is a Plus (+) sign button which is actually New Button

**Edit:** This sub module is used to modify the existing details of the Requester if anything goes changes in their record. There is a Pencil button which is actually Remove Button.

**Remove:** This is used to remove Requester from the service centre. There is a Trash button which is Remove Button.

**Sell Report:-** This module is used to view and print sell report.

**Work Report:-** This module is used to view and print Work report.

**Change Password:**

User can change his/her login password.

**Logout:**

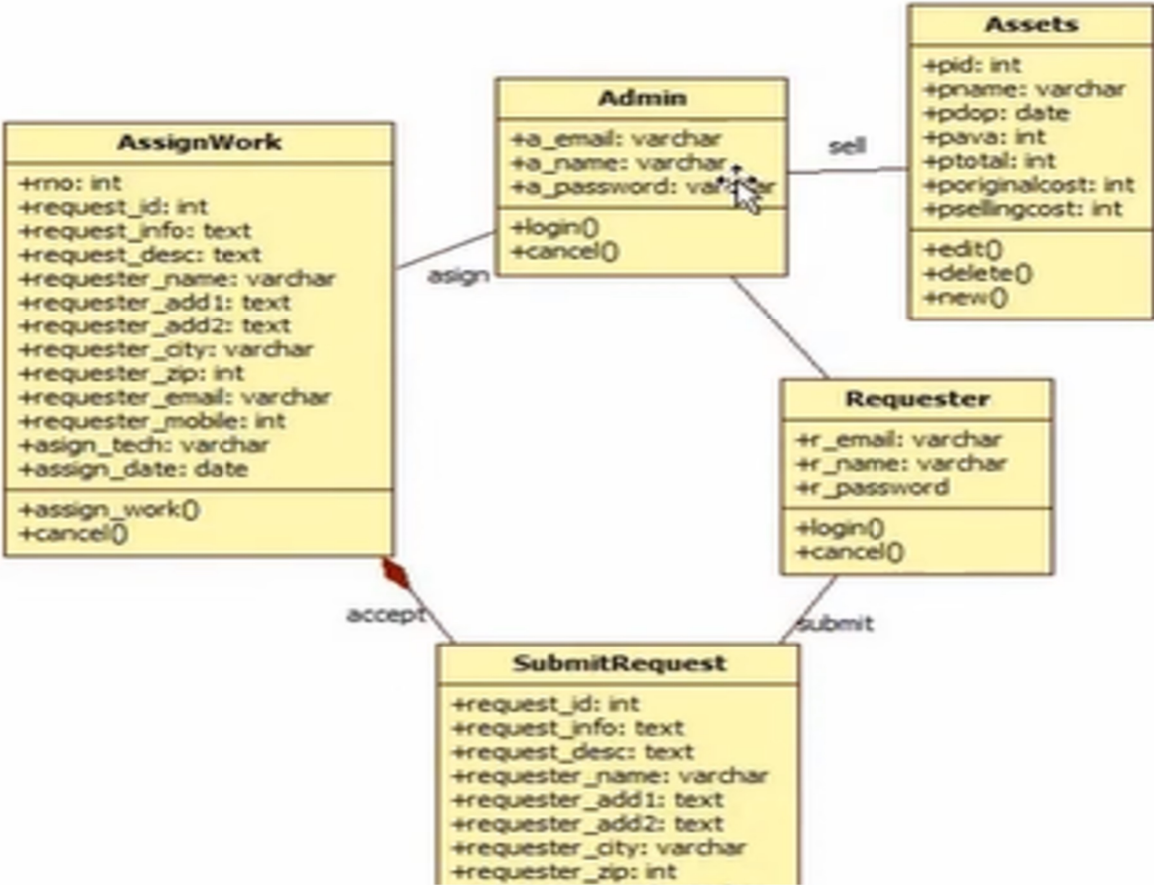
This Logout and Exit the Application.

### 3.4 Class Diagram

Class diagrams are the main building block in object-oriented modeling.

They are used to show the different objects in a system, their attributes, their operations and the relationships among them. Classes in class diagrams are represented by boxes that are partitioned into three:

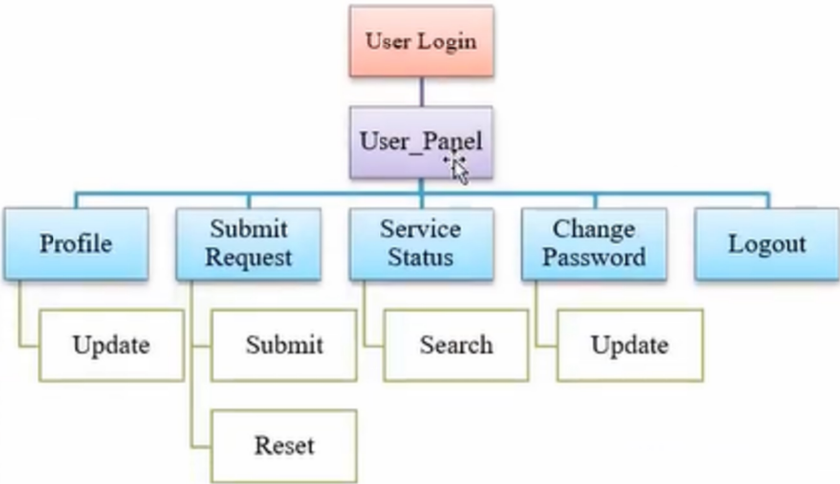
- (i) The top partition contains the name of the class.
- (ii) The middle part contains the class's attributes,
- (iii) The bottom partition shows the possible operations that are associated with the class.

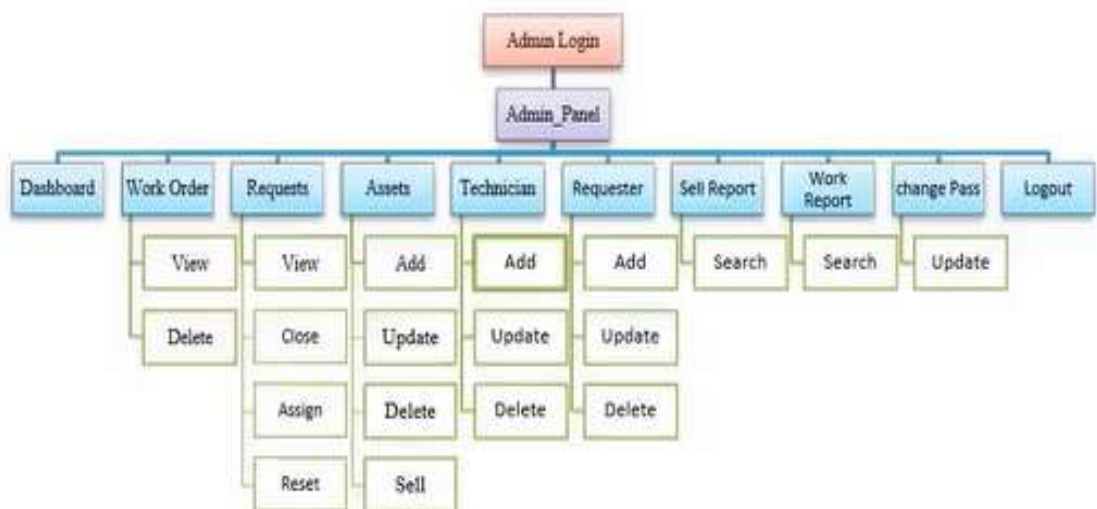


### 3.5

### Process Logic

:-





### 3.6 Data Integrity

Data integrity is the overall completeness, accuracy and consistency of data. This can be indicated by the absence of alteration between two instances or between two

updates of a data record, meaning data is intact and unchanged. Data integrity is usually imposed during the database design phase through the use of standard procedures and rules. The concept of data integrity ensures that all data in a database can be traced and connected to other data. This ensures that everything is recoverable and searchable. Having a single, well-defined and well-controlled data integrity system increases stability, performance, reusability and maintainability. Data values are standardized according to a data model and data type. All characteristics of the data must be correct including business rules, relations, dates and definitions for data to be complete. Data integrity is imposed within a database when it is designed and is authenticated through the ongoing use of error checking and validation routines. As a simple example, to maintain data integrity numeric columns/cells should not accept alphabetic data.

### 3.7 Data Dictionary

A data dictionary contains a list of all files in the database, the number of records in each file, and the names and types of each field. Most database management systems keep the data dictionary hidden from users to prevent them from accidentally destroying its contents. For most relational database management systems (RDBMS), the database management system software needs the data dictionary to access the data within a database. For example, the Oracle Database software has to read and write to an Oracle Database. However, it can only do this via the data dictionary created for that particular database.

Table Name: adminlogin_tb		
Attribute	Data Type	Description
a_login_id #	int	Stores login id (Automatically Generated)
a_name	varchar(60)	Stores Name
a_email	varchar(60)	Store Email
a_password	varchar(60)	Store Password

phpMyAdmin

Server: 127.0.0.1 » Database: osms\_db » Table: adminlogin\_tb

Browse Structure SQL Search Insert Export Import Privileges Operations Triggers

Table structure Relation view

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
1	a_login_id	int(11)			No	None		AUTO_INCREMENT	Change Drop More
2	a_name	varchar(60)	utf8_bin		No	None			Change Drop More
3	a_email	varchar(60)	utf8_bin		No	None			Change Drop More
4	a_password	varchar(60)	utf8_bin		No	None			Change Drop More

Check all With selected: Browse Change Drop Primary Unique Index Spatial Fulltext

Print Propose table structure Move columns Normalize

Add 1 column(s) after a\_password Go

Indexes

Action	Keyname	Type	Unique	Packed	Column	Cardinality	Collation	Null	Comment
Edit Rename Drop	PRIMARY	BTREE	Yes	No	a_login_id	0	A	No	

Create an index on 1 columns Go

Activate Windows

Online Service managin

Table Name: requesterlogin\_tb

Attribute	Data Type	Description
r_login_id #	int	Stores login id (Automatically Generated)
r_name	varchar(60)	Stores Name
r_email	varchar(60)	Store Email
r_password	varchar(60)	Store Password



phpMyAdmin

Server: 127.0.0.1 » Database: osms\_db » Table: requesterlogin\_tb

Browse Structure SQL Search Insert Export Import Privileges Operations Triggers

Table structure Relation view

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
<input type="checkbox"/>	1 r_login_id	int(11)			No	None		AUTO_INCREMENT	<a href="#">Change</a> <a href="#">Drop</a> <a href="#">More</a>
<input type="checkbox"/>	2 r_name	varchar(60)	utf8_bin		No	None			<a href="#">Change</a> <a href="#">Drop</a> <a href="#">More</a>
<input type="checkbox"/>	3 r_email	varchar(60)	utf8_bin		No	None			<a href="#">Change</a> <a href="#">Drop</a> <a href="#">More</a>
<input type="checkbox"/>	4 r_password	varchar(60)	utf8_bin		No	None			<a href="#">Change</a> <a href="#">Drop</a> <a href="#">More</a>

☐ Check all With selected: [Browse](#) [Change](#) [Drop](#) [Primary](#) [Unique](#) [Index](#) [Spatial](#) [Fulltext](#)

[Print](#) [Propose table structure](#) [Move columns](#) [Normalize](#)

[Add](#) 1 column(s) after r\_password [Go](#)

Indexes

Action	Keyname	Type	Unique	Packed	Column	Cardinality	Collation	Null	Comment
<a href="#">Edit</a> <a href="#">Rename</a> <a href="#">Drop</a>	PRIMARY	BTREE	Yes	No	r_login_id	3	A	No	

Create an index on 1 columns [Go](#)

**Table Name: customer\_tb**

Attributes	Data Type	Description
custid #	int	Customer ID (Automatically Generated)
custname	varchar(60)	Customer Name
custadd	varchar(60)	Customer Address
cpname	varchar(60)	Product Name
cpquantity	int	Product Quantity
cpeach	int	Each Quantity Price
cptotal	int	Total Price
cpdate	date	Selling Date

phpMyAdmin

Recent Favorites

- New
- information\_schema
- laundry\_website
- layundry
- mysql
- osms\_db
  - New
  - adminlogin\_tb
  - assets\_tb
  - assignwork\_tb
  - customer\_tb
  - requesterlogin\_tb
  - submitrequest\_tb
  - technician\_tb
- performance\_schema
- phpmyadmin
- shop\_inventory
- test

Server: 127.0.0.1 » Database: osms\_db » Table: customer\_tb

Browse Structure SQL Search Insert Export Import Privileges Operations Triggers

Table structure Relation view

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
<input type="checkbox"/>	1	custid	int(11)		No	None		AUTO_INCREMENT	<a href="#">Change</a> <a href="#">Drop</a> <a href="#">More</a>
<input type="checkbox"/>	2	custname	varchar(60) utf8_bin		No	None			<a href="#">Change</a> <a href="#">Drop</a> <a href="#">More</a>
<input type="checkbox"/>	3	custadd	varchar(60) utf8_bin		No	None			<a href="#">Change</a> <a href="#">Drop</a> <a href="#">More</a>
<input type="checkbox"/>	4	cpname	varchar(60) utf8_bin		No	None			<a href="#">Change</a> <a href="#">Drop</a> <a href="#">More</a>
<input type="checkbox"/>	5	cpquantity	int(11)		No	None			<a href="#">Change</a> <a href="#">Drop</a> <a href="#">More</a>
<input type="checkbox"/>	6	cpeach	int(11)		No	None			<a href="#">Change</a> <a href="#">Drop</a> <a href="#">More</a>
<input type="checkbox"/>	7	cptotal	int(11)		No	None			<a href="#">Change</a> <a href="#">Drop</a> <a href="#">More</a>
<input type="checkbox"/>	8	cpdate	date		No	None			<a href="#">Change</a> <a href="#">Drop</a> <a href="#">More</a>

☐ Check all With selected: [Browse](#) [Change](#) [Drop](#) [Primary](#) [Unique](#) [Index](#) [Spatial](#) [Fulltext](#)

[Print](#) [Propose table structure](#) [Move columns](#) [Normalize](#)

[Add](#) 1 column(s) after cpdate [Go](#)

Indexes

Action	Keyname	Type	Unique	Packed	Column	Cardinality	Collation	Null	Comment
--------	---------	------	--------	--------	--------	-------------	-----------	------	---------

Activate Windows  
Go to Settings to activate Windows.

**Table Name: assets\_tb**

Attributes	Data Type	Description
pid #	int	Product ID (Automatically Generated)
pname	varchar(60)	Product Name
pdop	date	Product Date
pava	int	Number of Available Product
ptotal	int	Number of Total Product
poriginalcost	int	Product Original Cost
psellingcost	int	Product Selling Price

**phpMyAdmin**

Recent | Favorites

- New
- information\_schema
- laundry\_website
- layundry
- mysql
- osms\_db
  - New
  - adminlogin\_tb
  - assets\_tb
  - assignwork\_tb
  - customer\_tb
  - requesterlogin\_tb
  - submitrequest\_tb
  - technician\_tb
- performance\_schema
- phpmyadmin
- shop\_inventory
- test

Server: 127.0.0.1 » Database: osms\_db » Table: assets\_tb

Browse Structure SQL Search Insert Export Import Privileges Operations Triggers

Table structure Relation view

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
<input type="checkbox"/>	1 pid	int(11)			No	None		AUTO_INCREMENT	<a href="#">Change</a> <a href="#">Drop</a> <a href="#">More</a>
<input type="checkbox"/>	2 pname	varchar(60)	utf8_bin		No	None			<a href="#">Change</a> <a href="#">Drop</a> <a href="#">More</a>
<input type="checkbox"/>	3 pdop	date			No	None			<a href="#">Change</a> <a href="#">Drop</a> <a href="#">More</a>
<input type="checkbox"/>	4 pava	int(11)			No	None			<a href="#">Change</a> <a href="#">Drop</a> <a href="#">More</a>
<input type="checkbox"/>	5 ptotal	int(11)			No	None			<a href="#">Change</a> <a href="#">Drop</a> <a href="#">More</a>
<input type="checkbox"/>	6 poriginalcost	int(11)			No	None			<a href="#">Change</a> <a href="#">Drop</a> <a href="#">More</a>
<input type="checkbox"/>	7 psellingcost	int(11)			No	None			<a href="#">Change</a> <a href="#">Drop</a> <a href="#">More</a>

☐ Check all
 With selected: [Browse](#) [Change](#) [Drop](#) [Primary](#) [Unique](#) [Index](#) [Spatial](#) [Fulltext](#)

[Print](#) [Propose table structure](#) [Move columns](#) [Normalize](#)

[Add](#) 1 column(s) after psellingcost [Go](#)

Indexes

Action	Keyname	Type	Unique	Packed	Column	Cardinality	Collation	Null	Comment
<a href="#">Edit</a> <a href="#">Rename</a> <a href="#">Drop</a>	PRIMARY	BTREE	Yes	No	pid	3	A	No	

Activate Windows  
Go to Settings to activate Windows.

**Table Name: submitrequest\_tb**

Attributes	Data Type	Description
request_id #	int	Request ID (Automatically Generated)
request_info	text	Request Info
request_desc	text	Request Description
requester_name	varchar(60)	Requester Name
requester_add1	text	Requester Address Line 1
requester_add2	text	Requester Address Line 2
requester_city	varchar(60)	Requester City
requester_state	varchar(60)	Requester State
requester_zip	int	Requester Zip
requester_email	varchar(60)	Requester Email
requester_mobile	bigint	Requester Mobile
request_date	date	Request Date

phpMyAdmin

Server: 127.0.0.1 Database: osms\_db Table: submitrequest\_tb

Recent Favorites

Table structure Relation view

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
<input type="checkbox"/>	1 request_id	int(11)			No	None		AUTO_INCREMENT	<a href="#">Change</a> <a href="#">Drop</a> <a href="#">More</a>
<input type="checkbox"/>	2 request_info	text	utf8_bin		No	None			<a href="#">Change</a> <a href="#">Drop</a> <a href="#">More</a>
<input type="checkbox"/>	3 request_desc	text	utf8_bin		No	None			<a href="#">Change</a> <a href="#">Drop</a> <a href="#">More</a>
<input type="checkbox"/>	4 requester_name	varchar(60)	utf8_bin		No	None			<a href="#">Change</a> <a href="#">Drop</a> <a href="#">More</a>
<input type="checkbox"/>	5 requester_add1	text	utf8_bin		No	None			<a href="#">Change</a> <a href="#">Drop</a> <a href="#">More</a>
<input type="checkbox"/>	6 requester_add2	text	utf8_bin		No	None			<a href="#">Change</a> <a href="#">Drop</a> <a href="#">More</a>
<input type="checkbox"/>	7 requester_city	varchar(60)	utf8_bin		No	None			<a href="#">Change</a> <a href="#">Drop</a> <a href="#">More</a>
<input type="checkbox"/>	8 requester_state	varchar(60)	utf8_bin		No	None			<a href="#">Change</a> <a href="#">Drop</a> <a href="#">More</a>
<input type="checkbox"/>	9 requester_zip	int(11)			No	None			<a href="#">Change</a> <a href="#">Drop</a> <a href="#">More</a>
<input type="checkbox"/>	10 requester_email	varchar(60)	utf8_bin		No	None			<a href="#">Change</a> <a href="#">Drop</a> <a href="#">More</a>
<input type="checkbox"/>	11 requester_mobile	bigint(11)			No	None			<a href="#">Change</a> <a href="#">Drop</a> <a href="#">More</a>
<input type="checkbox"/>	12 request_date	date			No	None			<a href="#">Change</a> <a href="#">Drop</a> <a href="#">More</a>

**Table Name: assignwork\_tb**

Attributes	Data Type	Description
rno #	int	Request Number (Automatically Generated)
request_id	int	Request ID
request_info	text	Request Info
request_desc	text	Request Description
requester_name	varchar(60)	Requester Name
requester_add1	text	Requester Address Line 1
requester_add2	text	Requester Address Line 2
requester_city	varchar(60)	Requester City
requester_state	varchar(60)	Requester State
requester_zip	int	Requester Zip
requester_email	varchar(60)	Requester Email
requester_mobile	bigint	Requester Mobile
assign_tech	varchar(60)	Assign Technician Name
assign_date	date	Assigned Date



phpMyAdmin

Server: 127.0.0.1 » Database: osms\_db » Table: assignwork\_tb

Browse Structure SQL Search Insert Export Import Privileges Operations Triggers

Table structure Relation view

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
<input type="checkbox"/> 1	rno	int(11)			No	None		AUTO_INCREMENT	Change  Drop  More
<input type="checkbox"/> 2	request_id	int(11)			No	None			Change  Drop  More
<input type="checkbox"/> 3	request_info	text	utf8_bin		No	None			Change  Drop  More
<input type="checkbox"/> 4	request_desc	text	utf8_bin		No	None			Change  Drop  More
<input type="checkbox"/> 5	requester_name	varchar(60)	utf8_bin		No	None			Change  Drop  More
<input type="checkbox"/> 6	requester_add1	text	utf8_bin		No	None			Change  Drop  More
<input type="checkbox"/> 7	requester_add2	text	utf8_bin		No	None			Change  Drop  More
<input type="checkbox"/> 8	requester_city	varchar(60)	utf8_bin		No	None			Change  Drop  More
<input type="checkbox"/> 9	requester_state	varchar(60)	utf8_bin		No	None			Change  Drop  More
<input type="checkbox"/> 10	requester_zip	int(11)			No	None			Change  Drop  More
<input type="checkbox"/> 11	requester_email	varchar(60)	utf8_bin		No	None			Change  Drop  More
<input type="checkbox"/> 12	requester_mobile	bigint(11)			No	None			Change  Drop  More
<input type="checkbox"/> 13	assign_tech	varchar(60)	utf8_bin		No	None			Change  Drop  More
<input type="checkbox"/> 14	assign_date	date			No	None			Change  Drop  More

Create Windows  
Go to Settings to activate Windows.

**Table Name: technician\_tb**

Attributes	Data Type	Description
empid #	int	Employee ID (Automatically Generated)
empname	varchar(60)	Employee Name
empcity	varchar(60)	Employee City
empmobile	bigint	Employee Mobile Number
empemail	varchar(60)	Employee Email ID

phpMyAdmin

Recent

Favorites

New

information\_schema

laundry\_website

layundry

mysql

osms\_db

- New
  - adminlogin\_tb
  - assets\_tb
  - assignwork\_tb
  - customer\_tb
  - requesterlogin\_tb
  - submitrequest\_tb
  - technician\_tb
- performance\_schema
- phpmyadmin
- shop\_inventory
- test

Server: 127.0.0.1 » Database: osms\_db » Table: technician\_tb

Browse

Structure

SQL

Search

Insert

Export

Import

Privileges

Operations

Triggers

Table structure

Relation view

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
<input type="checkbox"/>	1	empid	int(11)		No	None		AUTO_INCREMENT	<a href="#">Change</a> <a href="#">Drop</a> <a href="#">More</a>
<input type="checkbox"/>	2	empName	varchar(60)	utf8_bin	No	None			<a href="#">Change</a> <a href="#">Drop</a> <a href="#">More</a>
<input type="checkbox"/>	3	empCity	varchar(60)	utf8_bin	No	None			<a href="#">Change</a> <a href="#">Drop</a> <a href="#">More</a>
<input type="checkbox"/>	4	empMobile	bigint(11)		No	None			<a href="#">Change</a> <a href="#">Drop</a> <a href="#">More</a>
<input type="checkbox"/>	5	empEmail	varchar(60)	utf8_bin	No	None			<a href="#">Change</a> <a href="#">Drop</a> <a href="#">More</a>

☐ Check all    With selected: [Browse](#) [Change](#) [Drop](#) [Primary](#) [Unique](#) [Index](#) [Spatial](#) [Fulltext](#)

[Print](#)

[Propose table structure](#)

[Move columns](#)

[Normalize](#)

Add  column(s) after empEmail [Go](#)

Indexes

Action	Keyname	Type	Unique	Packed	Column	Cardinality	Collation	Null	Comment
<a href="#">Edit</a> <a href="#">Rename</a> <a href="#">Drop</a>	PRIMARY	BTREE	Yes	No	empid	0	A	No	

## Chapter 4 : Tools And Environment

---

### 4.1 Hardware Requirement :

Processor	1.6 GHz or Faster processor
RAM	1.5 GB
DISK Space	4 GB of Available Hard Disk
Graphic	DirectX 9-Capable Video Card
Display	1024 X 768 or Higher Resolution



## 4.2 Software Requirments:

Operating System	Windows 10
Front End	HTML,CSS,JAVA SCRIPT
Frameworks / Library	Bootstrap , Font awesome, Google font
Back end	PHP
Text editor	Visual Studio Code
Database	MySql
Web Browser	Google Chrome
Web Server	Apache
Drawing Tool	Star uml _ Yed Graph Editot

Online Service Managment System

## Chapter 5 : Software Description

---

### 5.1 Visual Studio Code

Visual Studio Code was announced on April 29, 2015 by Microsoft at the 2015 Build conference. A Preview build was released shortly thereafter. On November 18, 2015, Visual Studio Code was released under the MIT License and its source code posted to GitHub. Extension support was alsob announced .

On April 14, 2016, Visual Studio Code graduated the public preview stage and was released to web. Visual Studio Code is a source code editor developed by Microsoft for Windows, Linux and macOS. It includes support for debugging, embedded Git control, syntax highlighting. intelligent code completion, snippets, and code refactoring. It is also customizable, so users can change the editor's theme, keyboard shortcuts, and preferences. It is free and open-source, although the official download is under a proprietary license..

## **5.2 Languages :-**

### **5.2.1 HTML :-**

Hypertext Markup Language (HTML) is the standard markup language. for creating web pages and web applications. With Cascading Style Sheets (CSS) and JavaScript, it forms a triad of cornerstone technologies for the World Wide Web. Web browsers receive HTML documents from a web server or from local storage and render the documents into multimedia web pages. HTML describes the structure of a web page semantically and originally included cues for the appearance of the document.

### **5.2.2 CSS :-**

Cascading Style Sheets (CSS) is a style sheet language used for describing the presentation of a document written in a markup language

like HTML. CSS is a cornerstone technology of the World Wide Web. alongside HTML and JavaScript. CSS is designed to enable the separation of presentation and content, including layout, colors, and fonts. This separation can improve content accessibility, provide more flexibility and control in the specification of presentation characteristics, enable multiple web pages to share formatting by specifying the relevant CSS in a separate .css file, and reduce complexity and repetition in the structural content.

### **5.2.3 JAVASCRIPT**

JavaScript often abbreviated as JS, is a high-level, interpreted programming language. It is a language which is also characterized as dynamic, weakly typed, prototype-based and multi-paradigm.

### **5.2.4 Why using PHP**

Every PHP is an open source language and all its components are free to use and distribute. PHP is server-side scripting language. It is embedded in HTML source code. It is used to generate dynamic pages content. People find it useful to develop websites and dynamic web pages. It is platform independent. PHP supports all major web servers such as Apache, Microsoft IIS and Netscape etc. All the major database such as Mysql, PostgreSQL, Oracle, Sybase, Microsoft SQL Server is supported by PHP. The main reasons for using PHP language are:

(i) It collects form data and save data send by mail. It sends and receives cookies by accessing cookies variables.

- (ii) It sends and receives cookies by accessing cookies variables.
- iii) It provides add, delete and modify element function within our database.
- (iv) Through PHP, we can restrict users to access some pages of our website.
- (v) It can encrypt data, so that our data will become more secure.

### **5.3 Why using MySQL**

Mysql is the most popular open source relational database management system. It is one of the best RDBMS being used to develop web-based software applications. It is easy to use and fast RDBMS. There are many good reasons which help us to develop website using this RDBMS:

- (i) It is open-source, so available for free.
- (ii) It works on many operating system and with many languages including PHP, PERL, C, C++ etc.
- (iii) Mysql is customizable.
- (iv) Mysql works very friendly to PHP.
- (v) Mysql works very quickly and works well even with large data sets.

### **5.4 yEd Graph Editor**

yEd is a powerful Free Desktop Application that can be used to quickly and effectively generate high-quality diagrams. yEd can be used to draw many different types of diagrams, including flowcharts, network diagrams, UMLs, BPMN, mind maps, organization charts, and entity relationship diagrams. yEd can automatically arrange diagram elements using a variety of graph layout algorithms. The program works much like many similar applications. yEd can export diagrams to various raster and vector formats, including GIF, JPEG, PNG, EMF, BMP, PDF, EPS, and SVG.

### **5.5 StarUML**

StarUML is an open source project to develop fast, flexible, extensible and featureful diagrams. With StarUML it is very easy to make Class Diagram. StarUML is implemented to provide many user-friendly features such as Quick dialog, Keyboard manipulation, Diagram overview, etc.

Online Service Managment System

## Chapter 6 : Program Code

---

**Index.php**



**<!DOCTYPE html>**

<html lang="en">

<head>

<meta charset="UTF-8">

<meta name="viewport" content="width=device-width, initial-scale=1.0">

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

<!-- Bootstrap CSS -->

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

<!-- Font Awesome CSS -->

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

<!-- Google Font -->

<link href="https://fonts.googleapis.com/css?family=Ubuntu" rel="stylesheet">

<!-- Custom CSS -->

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

<title>OSMS</title>

</head>

```
<body>

<!-- Start Navigation -->

<nav class="navbar navbar-expand-sm navbar-dark bg-danger pl-5 fixed-top">

  <a href="index.php" class="navbar-brand">OSMS</a>

  <span class="navbar-text">Customer's Happiness is our Aim</span>

  <button type="button" class="navbar-toggler" data-toggle="collapse" data-target="#myMenu">

    <span class="navbar-toggler-icon"></span>

  </button>

  <div class="collapse navbar-collapse" id="myMenu">

    <ul class="navbar-nav pl-5 custom-nav">

      <li class="nav-item"><a href="index.php" class="nav-link">Home</a></li>

      <li class="nav-item"><a href="#Services" class="nav-link">Services</a></li>

      <li class="nav-item"><a href="#registration" class="nav-link">Registration</a></li>

      <li class="nav-item"><a href="Requester/RequesterLogin.php" class="nav-link">Login</a></li>

      <li class="nav-item"><a href="#Contact" class="nav-link">Contact</a></li>

    </ul>

  </div>

</nav> <!-- End Navigation -->
```

```
<!-- Start Header Jumbotron-->

<header class="jumbotron back-image" style="background-image: url(images/Banner1.jpeg);">

  <div class="myclass mainHeading">

    <h1 class="text-uppercase text-danger font-weight-bold">Welcome to OSMS</h1>

    <p class="font-italic">Customer's Happiness is our Aim</p>

    <a href="Requester/RequesterLogin.php" class="btn btn-success mr-4">Login</a>

    <a href="#registration" class="btn btn-danger mr-4">Sign Up</a>

  </div>
```

</header> <!-- End Header Jumbotron -->

<div class="container">

<!--Introduction Section-->

<div class="jumbotron">

<h3 class="text-center">OSMS Services</h3>

<p>

OSMS Services is India’s leading chain of multi-brand Electronics and Electrical service

workshops offering

wide array of services. We focus on enhancing your uses experience by offering world-class

Electronic

Appliances maintenance services. Our sole mission is “To provide Electronic Appliances care

services to

keep the devices fit and healthy and customers happy and smiling”.

With well-equipped Electronic Appliances service centres and fully trained mechanics, we

provide quality

services with excellent packages that are designed to offer you great savings.

Our state-of-art workshops are conveniently located in many cities across the country. Now you

can book

your service online by doing Registration.

</p>

</div>

</div>

<!--Introduction Section End-->

<!-- Start Services -->

```
<div class="container text-center border-bottom" id="Services">
```

```
<h2>Our Services</h2>
```

```
<div class="row mt-4">
```

```
<div class="col-sm-4 mb-4">
```

```
<a href="#"><i class="fas fa-tv fa-8x text-success"></i></a>
```

```
<h4 class="mt-4">Electronic Appliances</h4>
```

```
</div>
```

```
<div class="col-sm-4 mb-4">
```

```
<a href="#"><i class="fas fa-sliders-h fa-8x text-primary"></i></a>
```

```
<h4 class="mt-4">Preventive Maintenance</h4>
```

```
</div>
```

```
<div class="col-sm-4 mb-4">
```

```
<a href="#"><i class="fas fa-cogs fa-8x text-info"></i></a>
```

```
<h4 class="mt-4">Fault Repair</h4>
```

```
</div>
```

```
</div>
```

```
</div> <!-- End Services -->
```

```
<!-- Start Registration -->
```

```
<?php include('userRegistration.php') ?>
```

```
<!-- End Registration -->
```

```
<!-- Start Happy Customer -->
```

```
<div class="jumbotron bg-danger" id="Customer">
```

```
<!-- Start Happy Customer Jumbotron -->
```

```
<div class="container">
```

```
<!-- Start Customer Container -->
```



```
<h2 class="text-center text-white">Happy Customers</h2>
```

```
<div class="row mt-5">
```

```
<div class="col-lg-3 col-sm-6">
```

```
<!-- Start Customer 1st Column-->
```

```
<div class="card shadow-lg mb-2">
```

```
<div class="card-body text-center">
```

```

```

```
<h4 class="card-title">Rahul Kumar</h4>
```

```
<p class="card-text">Itaque illo explicabo voluptatum, saepe libero rerum, ad
```

```
ducimus voluptas
```

```
nesciunt debitis numquam.</p>
```

```
</div>
```

```
</div>
```

```
</div> <!-- End Customer 1st Column-->
```

```
<div class="col-lg-3 col-sm-6">
```

```
<!-- Start Customer 2nd Column-->
```

```
<div class="card shadow-lg mb-2">
```

```
<div class="card-body text-center">
```

```

```

```
<h4 class="card-title">Sonam Sharma</h4>
```

```
<p class="card-text">Itaque illo explicabo voluptatum, saepe libero rerum, ad
```

```
ducimus voluptas
```

```
nesciunt debitis numquam.</p>
```

```
</div>
```

```
</div>
```

```
</div> <!-- End Customer 2nd Column-->
```

```
<div class="col-lg-3 col-sm-6">

  <!-- Start Customer 3rd Column-->

  <div class="card shadow-lg mb-2">

    <div class="card-body text-center">

      <h4 class="card-title">Sumit Vyas</h4>

      <p class="card-text">Itaque illo explicabo voluptatum, saepe libero rerum, ad  

        ducimus voluptas  

        nesciunt debitis numquam.</p>

    </div>

  </div>

</div> <!-- End Customer 3rd Column-->
```

```
<div class="col-lg-3 col-sm-6">

  <!-- Start Customer 4th Column-->

  <div class="card shadow-lg mb-2">

    <div class="card-body text-center">

      <h4 class="card-title">Jyoti Sinha</h4>

      <p class="card-text">Itaque illo explicabo voluptatum, saepe libero rerum, ad  

        ducimus voluptas  

        nesciunt debitis numquam.</p>

    </div>

  </div>

</div> <!-- End Customer 4th Column-->

</div> <!-- End Customer Row-->
```

</div> <!-- End Customer Container -->

</div> <!-- End Customer Jumbotron -->

<!--Start Contact Us-->

<div class="container" id="Contact">

<!--Start Contact Us Container-->

<h2 class="text-center mb-4">Contact US</h2> <!-- Contact Us Heading -->

<div class="row">

<!--Start Contact Us Row-->

<?php include('contactform.php'); ?>

<!-- End Contact Us 1st Column -->

<div class="col-md-4 text-center">

<!-- Start Contact Us 2nd Column-->

<strong>Headquarter:</strong> <br>

OSMS Pvt Ltd, <br>

Sec IV, Bokaro <br>

Jharkhand - 834005 <br>

Phone: +00000000 <br>

<a href="#" target="\_blank">www.osms.com</a> <br>

<br><br>

<strong>Delhi Branch:</strong> <br>

OSMS Pvt Ltd, <br>

Ashok Nagar, Delhi <br>

Delhi - 804002 <br>

```
Phone: +000000000 <br>

<a href="#" target="_blank">www.osms.com</a> <br>

</div> <!-- End Contact Us 2nd Column-->

</div> <!-- End Contact Us Row-->

</div> <!-- End Contact Us Container-->

<!-- End Contact Us -->


<!-- Start Footer-->

<footer class="container-fluid bg-dark text-white mt-5" style="border-top: 3px solid #DC3545;">

<div class="container">

<!-- Start Footer Container -->

<div class="row py-3">

<!-- Start Footer Row -->

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

<!-- Start Footer 1st Column -->

<span class="pr-2">Follow Us: </span>

<a href="#" target="_blank" class="pr-2 fi-color"><i class="fab fa-facebook-f"></i></a>

<a href="#" target="_blank" class="pr-2 fi-color"><i class="fab fa-twitter"></i></a>

<a href="#" target="_blank" class="pr-2 fi-color"><i class="fab fa-youtube"></i></a>

<a href="#" target="_blank" class="pr-2 fi-color"><i class="fab fa-google-plus-g"></i></a>

<a href="#" target="_blank" class="pr-2 fi-color"><i class="fas fa-rss"></i></a>

</div> <!-- End Footer 1st Column -->


<div class="col-md-6 text-right">

<!-- Start Footer 2nd Column -->

<small> Designed by Someone &copy; 2018.

</small>
```

```

        <small class="ml-2"><a href="Admin/login.php">Admin Login</a></small>

    </div> <!-- End Footer 2nd Column -->

</div> <!-- End Footer Row -->

</div> <!-- End Footer Container -->

</footer> <!-- End Footer -->


<!-- Bootstrap JavaScript -->

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

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

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

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

</body> </html>

```

## db connection.php :-

```

<?php
$db_host = "localhost";

$db_user = "root";

$db_password = "";

$db_name = "osms_db";


// Create Connection

$conn = new mysqli($db_host, $db_user, $db_password, $db_name);


// Check Connection

if($conn->connect_error) {

    die("connection failed");

}

```

?>

## User Registration.php :-

<?php

```
include('dbConnection.php');

if(isset($_REQUEST['rSignup'])){

    // Checking for Empty Fields

    if(($_REQUEST['rName'] == "") || ($_REQUEST['rEmail'] == "") || ($_REQUEST['rPassword'] == "")){

        $regmsg = '<div class="alert alert-warning mt-2" role="alert"> All Fields are Required </div>';

    } else {

        $sql = "SELECT r_email FROM requesterlogin_tb WHERE r_email='".$_REQUEST['rEmail']."'";

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

        if($result->num_rows == 1){

            $regmsg = '<div class="alert alert-warning mt-2" role="alert"> Email ID Already Registered </div>';

        } else {

            // Assigning User Values to Variable

            $rName = $_REQUEST['rName'];

            $rEmail = $_REQUEST['rEmail'];

            $rPassword = $_REQUEST['rPassword'];

            $sql = "INSERT INTO requesterlogin_tb(r_name, r_email, r_password) VALUES ('$rName','$rEmail', '$rPassword')";

            if($conn->query($sql) == TRUE){

                $regmsg = '<div class="alert alert-success mt-2" role="alert"> Account Succesfully Created </div>';

            } else {

                $regmsg = '<div class="alert alert-danger mt-2" role="alert"> Unable to Create Account </div>';

            }

        }

    }

}
```

```

}
?>

<div class="container pt-5" id="registration">

    <h2 class="text-center">Create an Account</h2>

    <div class="row mt-4 mb-4">

        <div class="col-md-6 offset-md-3">

            <form action="" class="shadow-lg p-4" method="POST">

                <div class="form-group">

                    <i class="fas fa-user"></i><label for="name" class="pl-2 font-weight-bold">Name</label><input type="text"
                        class="form-control" placeholder="Name" name="rName">

                </div>

                <div class="form-group">

                    <i class="fas fa-user"></i><label for="email" class="pl-2 font-weight-bold">Email</label><input type="email"
                        class="form-control" placeholder="Email" name="rEmail">

                    <!--Add text-white below if want text color white-->

                    <small class="form-text">We'll never share your email with anyone else.</small>

                </div>

                <div class="form-group">

                    <i class="fas fa-key"></i><label for="pass" class="pl-2 font-weight-bold">New
                        Password</label><input type="password" class="form-control" placeholder="Password" name="rPassword">

                </div>

                <button type="submit" class="btn btn-danger mt-5 btn-block shadow-sm font-weight-bold"
name="rSignup">Sign Up</button>

                <em style="font-size:10px;">Note - By clicking Sign Up, you agree to our Terms, Data
                    Policy and Cookie Policy.</em>

                <?php if(isset($regmsg)) {echo $regmsg; } ?>

            </form>

        </div>

    </div>

```

</div>

</div>

## Contact form.php :-

<?php

// The contact Us Form wont work with Local Host but it will work on Live Server

if(isset(\$\_REQUEST['submit'])) {

// Checking for Empty Fields

if(((\$\_REQUEST['name'] == "") || (\$\_REQUEST['subject'] == "") || (\$\_REQUEST['email'] == "") ||  
(\$\_REQUEST['message'] == "")){

// msg displayed if required field missing

\$msg = '<div class="alert alert-warning col-sm-6 ml-5 mt-2" role="alert"> Fill All Fileds </div>';

} else {

\$name = \$\_REQUEST['name'];

\$subject = \$\_REQUEST['subject'];

\$email = \$\_REQUEST['email'];

\$message = \$\_REQUEST['message'];

\$mailTo = "contact@geekyshows.com";

\$headers = "From: ". \$email;

\$txt = "You have received an email from ". \$name. ".\n\n".\$message;

mail(\$mailTo, \$subject, \$txt, \$headers);

\$msg = '<div class="alert alert-success col-sm-6 ml-5 mt-2" role="alert"> Sent Successfully </div>';

}

}

?>



```

<!--Start Contact Us Row-->

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

<!--Start Contact Us 1st Column-->

<form action="" method="post">

    <input type="text" class="form-control" name="name" placeholder="Name"><br>

    <input type="text" class="form-control" name="subject" placeholder="Subject"><br>

    <input type="email" class="form-control" name="email" placeholder="E-mail"><br>

    <textarea class="form-control" name="message" placeholder="How can we help you?"
    style="height:150px;"></textarea><br>

    <input class="btn btn-primary" type="submit" value="Send" name="submit"><br><br>

    <?php if(isset($msg)) {echo $msg; } ?>

</form>


</div> <!-- End Contact Us 1st Column-->

```


## Requester Login.php :-

### Online Maintenance Managment System

 Requester Area(Demo)

 **Email**

We'll never share your email with anyone else.

 **Password**

Activate Windows  
Go to Settings to activate Windows.

<?php

```
include('../dbConnection.php');
```

```

session_start();

if(!isset($_SESSION['is_login'])){

    if(isset($_REQUEST['rEmail'])){

        $rEmail = mysqli_real_escape_string($conn,trim($_REQUEST['rEmail']));

        $rPassword = mysqli_real_escape_string($conn,trim($_REQUEST['rPassword']));

        $sql = "SELECT r_email, r_password FROM requesterlogin_tb WHERE r_email='".$rEmail.'" AND r_password='".$rPassword.'" limit 1";

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

        if($result->num_rows == 1){

            $_SESSION['is_login'] = true;

            $_SESSION['rEmail'] = $rEmail;

            // Redirecting to RequesterProfile page on Correct Email and Pass

            echo "<script> location.href='RequesterProfile.php'; </script>";

            exit;

        } else {

            $msg = '<div class="alert alert-warning mt-2" role="alert"> Enter Valid Email and Password </div>';

        }

    }

} else {

    echo "<script> location.href='RequesterProfile.php'; </script>";

}

?>

<!DOCTYPE html>

<html lang="en">

<head>

```

```
<meta charset="UTF-8">
```

```
<meta name="viewport" content="width=device-width, initial-scale=1.0">
```

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

```
<!-- Bootstrap CSS -->
```

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

```
<!-- Font Awesome CSS -->
```

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

```
<style>
```

```
.custom-margin {  
    margin-top: 8vh;  
}
```

```
</style>
```

```
<title>Login</title>
```

```
</head>
```

```
<body>
```

```
<div class="mb-3 text-center mt-5" style="font-size: 30px;">
```

```
<i class="fas fa-stethoscope"></i>
```

```
<span>Online Maintenance Managment System</span>
```

```
</div>
```

```
<p class="text-center" style="font-size: 20px;"><i class="fas fa-user-secret text-danger"></i> <span>Requester  
Area(Demo)</span>
```

```
</p>
```

```
<div class="container-fluid mb-5">
```

```

<div class="row justify-content-center custom-margin">

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

    <form action="" class="shadow-lg p-4" method="POST">

      <div class="form-group">

        <i class="fas fa-user"></i><label for="email" class="pl-2 font-weight-bold">Email</label><input type="email"
          class="form-control" placeholder="Email" name="rEmail">

        <!--Add text-white below if want text color white-->

        <small class="form-text">We'll never share your email with anyone else.</small>

      </div>

      <div class="form-group">

        <i class="fas fa-key"></i><label for="pass" class="pl-2 font-weight-bold">Password</label><input
type="password"
          class="form-control" placeholder="Password" name="rPassword">

      </div>

      <button type="submit" class="btn btn-outline-danger mt-3 btn-block shadow-sm
font-weight-bold">Login</button>

      <?php if(isset($msg)) {echo $msg; } ?>

    </form>

    <div class="text-center"><a class="btn btn-info mt-3 shadow-sm font-weight-bold" href="../index.php">Back
      to Home</a></div>

  </div>

</div>

</div>

</div>

<!-- Bootstrap JavaScript -->

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

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

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

```

```
<script src="../js/all.min.js"></script>

</body>
```

```
</html>
```

## **Header.php :-**

```
<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

<meta name="viewport" content="width=device-width, initial-scale=1.0">

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

<title>

<?php echo TITLE ?>

</title>

<!-- Bootstrap CSS -->

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

<!-- Font Awesome CSS -->

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

<!-- Custome CSS -->

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

</head>

<body>

<!-- Top Navbar -->
```

```
<nav class="navbar navbar-dark fixed-top bg-danger flex-md-nowrap p-0 shadow">

<a class="navbar-brand col-sm-3 col-md-2 mr-0" href="RequesterProfile.php">OSMS</a>

</nav>
```

```
<!-- Side Bar -->
```

```
<div class="container-fluid mb-5 " style="margin-top:40px;">
```

```
<div class="row">
```

```
<nav class="col-sm-2 bg-light sidebar py-5 d-print-none">
```

```
<div class="sidebar-sticky">
```

```
<ul class="nav flex-column">
```

```
<li class="nav-item">
```

```
<a class="nav-link <?php if(PAGE == 'RequesterProfile') { echo 'active'; } ?>" href="RequesterProfile.php">
```

```
<i class="fas fa-user"></i>
```

```
Profile <span class="sr-only">(current)</span>
```

```
</a>
```

```
</li>
```

```
<li class="nav-item">
```

```
<a class="nav-link <?php if(PAGE == 'SubmitRequest') { echo 'active'; } ?>" href="SubmitRequest.php">
```

```
<i class="fab fa-accessible-icon"></i>
```

```
Submit Request
```

```
</a>
```

```
</li>
```

```
<li class="nav-item">
```

```
<a class="nav-link <?php if(PAGE == 'CheckStatus') { echo 'active'; } ?>" href="CheckStatus.php">
```

```
<i class="fas fa-align-center"></i>
```

```
Service Status
```

```
</a>
```

```

</li>

<li class="nav-item">

    <a class="nav-link <?php if(PAGE == 'Requesterchange'pass') { echo 'active'; } ?>"
href="Requesterchange'pass.php">

        <i class="fas fa-key"></i>

        Change Password

    </a>

</li>

<li class="nav-item">

    <a class="nav-link" href=" ../logout.php">

        <i class="fas fa-sign-out-alt"></i>

        Logout

    </a>

</li>

</ul>

</div>

</nav>

```

## Footer.php :-

```

<!-- Bootstrap JavaScript -->

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

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

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

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

</body>

</html>

```

## Requesterchange'pass.php

```

<?php

define('TITLE', 'Change Password');

define('PAGE', 'RequesterchangePASS');

include('includes/header.php');

include('../dbConnection.php');

session_start();

if($_SESSION['is_login']){

    $rEmail = $_SESSION['rEmail'];

} else {

    echo "<script> location.href='RequesterLogin.php'; </script>";

}


$rEmail = $_SESSION['rEmail'];

if(isset($_REQUEST['passupdate'])){

    if(($_REQUEST['rPassword'] == "")){

        // msg displayed if required field missing

        $passmsg = '<div class="alert alert-warning col-sm-6 ml-5 mt-2" role="alert"> Fill All Fileds </div>';

    } else {

        $sql = "SELECT * FROM requesterlogin_tb WHERE r_email='$rEmail'";

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

        if($result->num_rows == 1){

            $rPass = $_REQUEST['rPassword'];

            $sql = "UPDATE requesterlogin_tb SET r_password = '$rPass' WHERE r_email = '$rEmail'";

            if($conn->query($sql) == TRUE){

                // below msg display on form submit success

                $passmsg = '<div class="alert alert-success col-sm-6 ml-5 mt-2" role="alert"> Updated Successfully </div>';

            } else {

```



```

// below msg display on form submit failed

$passmsg = '<div class="alert alert-danger col-sm-6 ml-5 mt-2" role="alert"> Unable to Update </div>';

}

}

}

}

?>

<div class="col-sm-9 col-md-10">

<div class="row">

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

<form class="mt-5 mx-5" method="POST">

<div class="form-group">

<label for="inputEmail">Email</label>

<input type="email" class="form-control" id="inputEmail" value=" <?php echo $rEmail ?>" readonly>

</div>

<div class="form-group">

<label for="inputnewpassword">New Password</label>

<input type="password" class="form-control" id="inputnewpassword" placeholder="New Password"
name="rPassword">

</div>

<button type="submit" class="btn btn-danger mr-4 mt-4" name="passupdate">Update</button>

<button type="reset" class="btn btn-secondary mt-4">Reset</button>

<?php if(isset($passmsg)) {echo $passmsg; } ?>

</form>

</div>

</div>

```

</div>

</div>

</div>

<?php

include('includes/footer.php');

?>

## RequesterProfile.php

<?php

define('TITLE', 'Change Password');

define('PAGE', 'Requesterchangeepass');

include('includes/header.php');

include('../dbConnection.php');

session\_start();

if(\$\_SESSION['is\_login']){

    \$rEmail = \$\_SESSION['rEmail'];

} else {

    echo "<script> location.href='RequesterLogin.php'; </script>";

}

\$rEmail = \$\_SESSION['rEmail'];

if(isset(\$\_REQUEST['passupdate'])){

    if((\$\_REQUEST['rPassword'] == "")){

        // msg displayed if required field missing

        \$passmsg = '<div class="alert alert-warning col-sm-6 ml-5 mt-2" role="alert"> Fill All Fileds </div>';

    } else {

        \$sql = "SELECT \* FROM requesterlogin\_tb WHERE r\_email='\$rEmail'";

```

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

if($result->num_rows == 1){

    $rPass = $_REQUEST['rPassword'];

    $sql = "UPDATE requesterlogin_tb SET r_password = '$rPass' WHERE r_email = '$rEmail'";

    if($conn->query($sql) == TRUE){

        // below msg display on form submit success

        $passmsg = '<div class="alert alert-success col-sm-6 ml-5 mt-2" role="alert"> Updated Successfully </div>';

    } else {

        // below msg display on form submit failed

        $passmsg = '<div class="alert alert-danger col-sm-6 ml-5 mt-2" role="alert"> Unable to Update </div>';

    }

}

}

}

}

}

?>

<div class="col-sm-9 col-md-10">

    <div class="row">

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

            <form class="mt-5 mx-5" method="POST">

                <div class="form-group">

                    <label for="inputEmail">Email</label>

                    <input type="email" class="form-control" id="inputEmail" value=" <?php echo $rEmail ?>" readonly>

                </div>

                <div class="form-group">

                    <label for="inputnewpassword">New Password</label>

                    <input type="password" class="form-control" id="inputnewpassword" placeholder="New Password"

                    name="rPassword">


```

```
</div>
```

```
<button type="submit" class="btn btn-danger mr-4 mt-4" name="passupdate">Update</button>
```

```
<button type="reset" class="btn btn-secondary mt-4">Reset</button>
```

```
<?php if(isset($passmsg)) {echo $passmsg; } ?>
```

```
</form>
```

```
</div>
```

```
</div>
```

```
</div>
```

```
</div>
```

```
</div>
```

```
<?php
```

```
include('includes/footer.php');
```

```
?>
```

## **submitrequestsuccess.php**

```
<?php
```

```
define('TITLE', 'Success');
```

```
include('includes/header.php');
```

```
include('../dbConnection.php');
```

```
session_start();
```

```
if($_SESSION['is_login']){
```

```
    $rEmail = $_SESSION['rEmail'];
```

```
} else {
```

```
    echo "<script> location.href='RequesterLogin.php'; </script>";
```

```
}
```

```
$sql = "SELECT * FROM submitrequest_tb WHERE request_id = {$_SESSION['myid']}";
```

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

```
if($result->num_rows == 1){
```

```
    $row = $result->fetch_assoc();
```

```
    echo "<div class='ml-5 mt-5'>
```

```
    <table class='table'>
```

```
    <tbody>
```

```
    <tr>
```

```
        <th>Request ID</th>
```

```
        <td>".$row['request_id']. "</td>
```

```
</tr>
```

```
<tr>
```

```
    <th>Name</th>
```

```
    <td>".$row['requester_name']. "</td>
```

```
</tr>
```

```
<tr>
```

```
    <th>Email ID</th>
```

```
    <td>".$row['requester_email']. "</td>
```

```
</tr>
```

```
<tr>
```

```
    <th>Request Info</th>
```

```
    <td>".$row['request_info']. "</td>
```

```
</tr>
```

```
<tr>
```

```
    <th>Request Description</th>
```

```
    <td>".$row['request_desc']. "</td>
```

```
</tr>
```

```
<tr>

    <td><form class='d-print-none'><input class='btn btn-danger' type='submit' value='Print'
onClick='window.print()'></form></td>

</tr>

</tbody>

</table> </div>

";
```

```
} else {

    echo "Failed";

}

?>
```

```
<?php

include('includes/footer.php');

$conn->close();

?>
```

## Chapter 7 : Testing

---

Software testing is a process used to identify the correctness, completeness and quality of developed computer software. It includes a set of activities conducted with the intent of finding errors in software so that it could be corrected before the product is released to the end users. In other word software testing is an activity to check that the software system is defect free. Software testing is primarily a broad process that is composed of several interlinked processes. The primary objective of software testing is to measure software health along with its completeness in terms of core requirements. Software testing involves examining and checking software through different testing processes. The objectives of these processes can include:

- (i) Completeness - Verifying software completeness in regards to  
functional/business requirements
- (ii) Errors Free - Identifying technical bugs/errors and ensuring the software  
is error-free
- (iii) Stability Assessing usability, performance, security, localization,  
compatibility and installation

This phase determine the error in the project. If there is any error then it must be removed before delivery of the project.

### 7.1 Type of Testing

For determining errors various types of test action are performed: -

**Unit Testing:** -Unit testing focuses verification effort on the smallest unit of software design - the module. Using the detail design description as a guide, important control paths are

tested to uncover errors within the boundary of the module. The relative complexity of tests and the errors detected as a result is limited by the constrained scope established for unit testing. The unit test is always white box oriented, and the step can be conducted in parallel for multiple modules.

Unit testing is normally considered an adjunct to the coding step. After source level code has been developed, reviewed, and verified for correct syntax, unit test case design begins.

**Integration Testing** - A level of the software testing process where

individual units are combined and tested as a group. The purpose of this level of testing is to expose faults in the interaction between integrated units.

**System Testing:** -Software is only one element of a larger computer

based system. Ultimately, software is incorporated with other system

elements (e.g. new hardware, information), and a series of system

integration and validation tests are conducted. Steps taken during software design and testing can greatly improve the probability of successful software integration in the larger system. A classic system testing problem is "finger pointing". This occurs when a defect is uncovered, and one system element developer blames another for the problem. The software engineer should anticipate potential interfacing problems and design error handling paths that test all information coming from other elements of the system, conduct a series of tests that simulate bad data or other potential errors at the software interface, record the results or tests to use as "evidence" if finger pointing does occur, participate in the planning and design of system test to ensure that software is adequately tested.

There are many types of system tests that are worthwhile for software based systems:

**Usability Testing** - Usability Testing is a type of testing done from an end-user's perspective to determine if the system is easily usable. Tests all functionalities of the software against

the requirement.

**Performance testing**- is designed to test the run time performance of software within the context of an integrated system Security testing -



**Security testing** -attempts to verify that protection

mechanisms built into a system will protect it from improper penetration

**Stress tests** - Stress tests are designed to confront programs with

abnormal situations.

## 7.2 Use Case :-

A use case diagram is essentially a picture showing system behavior

along with the key actors that interact with the system. The use case represents complete functionality. Use case diagram can be imagined as a black box where only the input, output, and the function of the black box are known. Use Case elements are used to make test cases when performing the testing. The use case should contain all system activities that have significance to the users. A use case can be thought of as a collection of possible scenarios related to a particular goal, indeed. Use cases can be employed during several stages of software development. such as planning system requirements, validating design and testing software.

Use case Diagram Objects

Use case diagrams mostly consist of 3 objects: -

**Actor** :-

Actor is a use case diagram is any entity that performs a role in one given system. This could be a person, organization or an external system

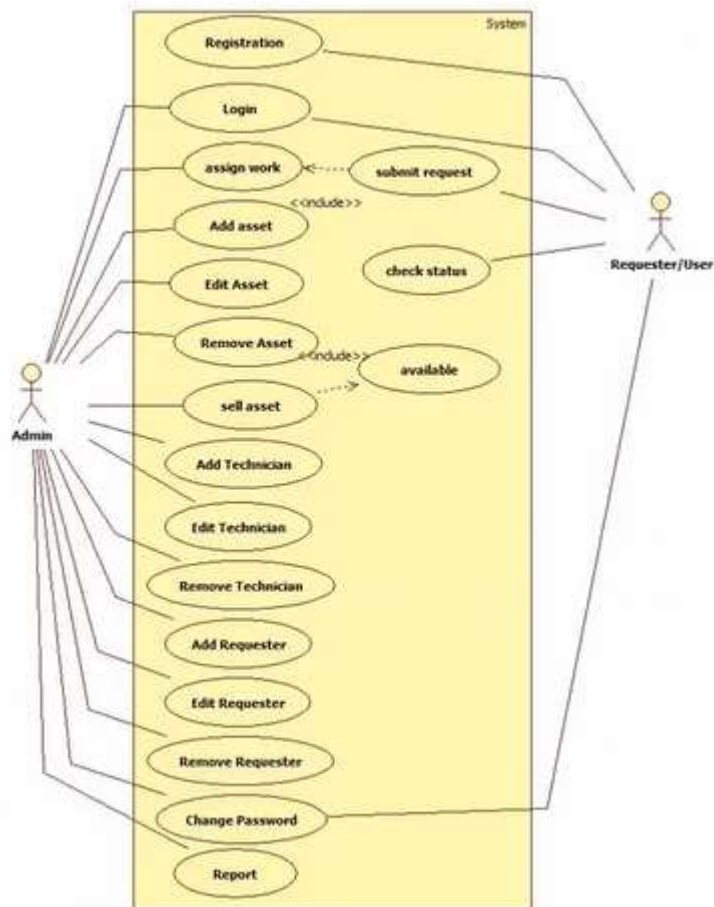
**Use Case** - A Use case represents a function or an action within the system. its drawn as an oval and named with the function.

**System** - System is used to define the define the scope of the use case and

-

**Include** - This represents required. Symbol of this function is dashed arrow and arrow is labeled with the keyword <<include>>

**Extend** - This represents optional and it is also shown with dashed arrow the arrow is labeled with the keyword <<extend>>



### 7.3 Test Case :-

A test case is a set of conditions or variables under which a tester will determine whether an application, software system or one of its features is working as it was originally established for it to do.

#### ⊕ Login:

Test Case ID	Test Scenario	Test Case	Pre-Condition	Test Steps	Test Data	Expected Result	Actual Result	Status Pass/Fail
TC_Login_1	Verify Login	Enter Valid username and valid password	Need a valid username and password to do login	1. Enter username 2. Enter Password 3. Click Login	Valid username Valid password	Successful login, Main screen of application should displayed	Successful login, Main screen of application displayed	Pass
TC_Login_2	Verify Login	Enter Valid username and invalid password	Need a valid username and password to do login	1. Enter username 2. Enter Password 3. Click Login	Valid username Invalid Password	No Matched Username/ Password	No Matched Username/ Password	Pass
TC_Login_3	Verify Login	Enter Invalid username and valid password	Need a valid username and password to do login	1. Enter username 2. Enter Password 3. Click Login	Invalid username Valid Password	No Matched Username/ Password	No Matched Username/ Password	Pass
TC_Login_4	Verify Login	Enter Invalid username and invalid password	Need a valid username and password to do login	1. Enter username 2. Enter Password 3. Click Login	Invalid username Invalid Password	No Matched Username/ Password	No Matched Username/ Password	Pass

## Registration

Test Case ID	Test Scenario	Test Case	Pre-Condition	Test Steps	Test Data	Expected Result	Actual Result	Status Pass/Fail
TC_REG_1	Verify Registration Detail	Enter Valid and correct data	Need valid text and number Data to be entered	1. Enter Valid Data in appropriate fields 2. Click Submit	Valid Text and Number Data	Successful, Member Added Successfully	Successful, Member Added Successfully	Pass
TC_REG_2	Verify Registration Detail	Enter invalid and incorrect data	Need text and number Data to be entered	Enter invalid Data in fields	Invalid Text and Number Data	Enter Valid Data	Enter Valid Data	Pass
TC_REG_3	Verify Registration Detail	Entering Nothing, Required Fields are blank	-	Click Submit	Nothing to enter Required fields are blank	Fill required field	Fill required field	Pass

## Chapter 8 : Implementation

---

Our dedication to our Clients goes well beyond the deployment of our software.

We are committed to providing our Client with a positive experience that starts with a successful implementation. Implementation is the stage in the project where the theoretical design is turned into a working system. The implementation phase constructs, installs and operates the new system. The most stage is achieving a new successful system is that it will work efficiently and effectively. Security and integrity of database are very important for any software system because databases are the backbone of the system. Security need to be implanted at every level of the system so that only authorized user can access the system for updation and other significance process.

## Chapter 9 : Limitations

---

- (i) SMS alert facility is not available.
- (ii) Portal is not SEO friendly
- (iii) Registration Email Verification Not available .
- (iv) Risk unauthorized accessibility

## Chapter 10: Future Scope

---

The various things can be made it simple and user friendly. As by increasing some of the coding we can improve it functionality. online payment system is yet not integrated to the system which can be featured in the near future. Till now it does not have the facility of back up the database. By as the next advancement we can make it able to bundle the backup facility so that one can perform operation based on previous records. As the technology emerges, it is possible to upgrade the system and can be

adaptable to desired environment. Based on the future security issued, security can be improved using emerging technologies.

## Chapter 11: Conclusion

---

The Online Service Managment System has been computed successfully and was also tested successfully by taking "Test Cases". It is user friendly, and has required options, which can be utilized by the user to perform the desired operations.

The Software is developed using HTML, CSS, JS as front end and PHP, MySql as back end in windows environment.

The goals that are achieved by the software are:

- (i) Simplification of the operations
- (ii) Less processing time and getting required information
- (iii) User friendly
- (iv) Portable and flexible for further enhancement

# Bibliography

---

The following reference has been used to develop the project "Online Service Managment System":

## **Books:**

- (i) The Complete Reference PHP
- (ii) Head First SQL: Your Brain on SQL by Lynn Beighley

## **Web Source: -**

- (I) [www.google.co.in](http://www.google.co.in)
- (ii) [www.wikipedia.org](http://www.wikipedia.org)
- (iii) [www.tutorialspoint.com](http://www.tutorialspoint.com)
- (iv) [www.stackoverflow.com](http://www.stackoverflow.com)
- (v) [www.docs.microsoft.com](http://www.docs.microsoft.com)