



# **CUSTOMER RELATIONSHIP MANAGEMENT [CRM] WEBSITE**

**A MINI-PROJECT REPORT**

*Submitted by*

**M.LINGESH (211519205086)**

**D.MANOJ (211519205093)**

**V.LOKESHWARAN (211519205090)**

*in partial fulfilment for the award of the degree*

*of*

**BACHELOR OF TECHNOLOGY**

*in*

**INFORMATION TECHNOLOGY**

**PANIMALAR INSTITUTE OF TECHNOLOGY**

**ANNA UNIVERSITY: CHENNAI 600 025**

**APRIL 2022**

# PANIMALAR INSTITUTE OF TECHNOLOGY

ANNA UNIVERSITY: CHENNAI 600 025



## BONAFIDE CERTIFICATE

Certified that this project report “ **CUSTOMER RELATIONSHIP MANAGEMENT [CRM] WEBSITE** ” is the bonafide work of “ **LINGESH (211519205086) , MANOJ (21519205093) , LOKESHWARAN (211519205090 )** ” who carried out the mini-project work under my supervision.

### SIGNATURE

**Dr. KAVITHA SUBRAMANI M.E, Ph.D**

**HEAD OF THE DEPARTMENT,**

Department of Information Technology,

Panimalar Institute of Technology,Poonamallee,

Chennai 600 123.

### SIGNATURE

**Mr. R. VINSTON RAJA M.Tech.,(Ph.D)**

**ASSOCIATE PROFESSOR,**

Department of Information Technology,

Panimalar Institute of Technology, Poonamallee,

Chennai 600 123.

---

**Certified that the candidates were examined in the university mini-project viva-voce held on\_\_\_\_\_at Panimalar Institute of Technology, Chennai 600 123.**

**INTERNAL EXAMINER**

**EXTERNAL EXAMINER**

## ACKNOWLEDGEMENT

A project of this magnitude and nature requires kind co-operation and support from many, for successful completion. We wish to express our sincere thanks to all those who were involved in the completion of this project.

We seek the blessing from the **Founder** of our institution **Dr.JEPPIAAR, M.A, Ph.D**, for having been a role model who has been our source of inspiration behind our success in education in his premier institution. Our sincere thanks to the Honorable Chairman of our prestigious institution **Mrs.REMIBAI JEPPIAAR** for her sincere endeavor in educating us in her premier institution.

We would like to express our deep gratitude to our beloved **Secretary and Correspondent Dr.P.CHINNADURAI, M.A, Ph.D**, for his kind words and enthusiastic motivation which inspired us a lot in completing this project.

We also express our sincere thanks and gratitude to our dynamic **Directors Mrs.C.VIJAYA RAJESHWARI, Dr.C.SAKTHI KUMAR, M.E, Ph.D**, and **Mrs.S.SARANYA SREE SAKTHI KUMAR, B.E**, for providing us with necessary facilities for completion of this project.

We also express our appreciation and gratefulness to our respected **Principal Dr. T. JAYANTHY, M.E, Ph.D**, who helped us in the completion of the project. We wish to convey our thanks and gratitude to our **Head of the Department, Dr. KAVITHA SUBRAMANI, M.E(CSE), Ph.D(CSE)**, for her full support by providing ample time to complete our project.

Last, we thank our parents and friends for providing their extensive moral support and encouragement during the course of the project.

## **ABSTRACT**

This is a project to develop a Customer Relationship Management (CRM) system. CRM is one of the systems included in Enterprise Resource Planning (ERP) System. The main purpose of CRM is to improve the relationship with the customers by using different module like analysis, customer service and others.

The objective to do this project is to develop a system which can help the organizations to decrease their defection rate of customers. Because the lower defection rate means the bigger customer base, which lead to more profit for the organization.

CRM in this project is included few modules. The modules included are customer filtering, customer profiling and promotion tools. Customer filtering allows the user filter out a customer list from the customer database by using customer's demographic information. Customer profiling to makes each customer is having a profile; the user may see the customer's profile included analysis of customer. Promotion Tools allows the user can create new promotion base on the product, and filter list of customer to promote the promotion.

After that, the user can see the analysis of the performance of the promotion. The tools used to develop this CRM included PHP, Web server, Mysql, javascript, HTML, CSS, jquery and jquery's plugin.

## **TABLE OF CONTENTS**

<b>CHAPTER</b>	<b>TITLE</b>	<b>PAGE NO</b>
	<b>ABSTRACT</b>	<b>I</b>
	<b>LIST OF FIGURES</b>	<b>IV</b>
	<b>LIST OF ABBREVIATION</b>	<b>V</b>
<b>1</b>	<b>INTRODUCTION</b>	<b>1</b>
	1.1 OVERVIEW	1
	1.2 PROBLEM DEFINITION	2
	1.3 SCOPE	3
<b>2</b>	<b>LITERATURE SURVEY</b>	<b>4</b>
	2.1 SURVEY DESCRIPTION	4
<b>3</b>	<b>SYSTEM ANALYSIS</b>	<b>6</b>
	3.1 EXISTING SYSTEM	6
	3.2 PROPOSED SYSTEM	6
<b>4</b>	<b>SYSTEM SPECIFICATION</b>	<b>8</b>
	4.1 HARDWARE SPECIFICATION	8
	4.2 SOFTWARE SPECIFICATION	8
<b>5</b>	<b>SYSTEM IMPLEMENTATION</b>	<b>13</b>
	5.1 SYSTEM ARCHITECTURE	13
	5.2 LAYER OF CODING	
	5.3 MODULES DESCRIPTION	15
<b>6</b>	<b>SYSTEM DESIGN</b>	<b>26</b>
	6.1 DATA FLOW DIAGRAM	26
	6.1.1 LEVEL 0	26
	6.1.2 LEVEL 1	28

6.1.4	LEVEL 3	29
6.2	UML DIAGRAM	31
6.2.1	USE CASE DIAGRAM 1	31
6.2.2	USE DIAGRAM 2	32
6.2.3	ENTITY RELATIONSHIP MODEL	33
<b>7</b>	<b>SOFTWARE TESTING</b>	
7.1	SYSTEM IMPLEMENTATION	
7.2	PREPARE FOR SYSTEM	
7.3	SYSTEM MAINTENANCE	
7.4	COST ESTIMATION	
7.5	REPORTS	
<b>8</b>	<b>SYSTEM TESTING</b>	
8.1	TESTING OBJECTIVES	34
8.2	SYSTEM TESTING	35
8.3.	TESTING LEVEL	36
8.3.1	UNIT TESTING	36
8.3.2	INTEGRATION TESTING	37
8.3.3	VALIDATION TESTING	38

<b>9</b>	<b>CONCLUSION AND FUTURE ENHANCEMENT</b>	<b>41</b>
<b>10</b>	<b>APPENDIX A: SOURCE CODE</b>	<b>42</b>
<b>11</b>	<b>APPENDIX B: SNAP SHOTS</b>	<b>55</b>
<b>12</b>	<b>REFERENCES</b>	<b>67</b>





## **LIST OF FIGURES**

<b>FIGURE NO</b>	<b>NAME OF THE FIGURE</b>	<b>PAGE NO</b>
1	Architecture Diagram	11
2	Architecture of CNN	14
3	Architecture of Deep Neural Network	18
4	Confusion Matrix	20
5	Data Preprocessing DFD	21
6	Data Sampling DFD	22
7	CNN Algorithm DFD	23
8	DNN Algorithm DFD	23
9	Performance Comparison of CNN and DNN DFD	24

## **LIST OF ABBREVIATION**

<b>ABBREVIATION</b>	<b>EXPANSION</b>
CRM	CUSTOMER RELATIONSHIP MANAGEMENT
SFA	SALES FORCE AUTOMATION
HTML	HYPER TEXT MARKUP LANGUAGE
CSS	CASCADING STYLE SHEET
PHP	HYPER TEXT PREPROCESSOR
SQL	STRUCTURED QUERY LANGUAGE

# **CHAPTER 1**

## **INTRODUCTION**

### **[CUSTOMER RELATIONSHIP MANAGEMENT]**

#### **1.1 OVERVIEW**

A CRM is a collection of people, processes, software, and internet capabilities that helps an enterprise manage customer relationship effectively and systematically. The goal of CRM is to understand and anticipate the needs of current and potential customer to increase customer retention and loyalty while optimizing the way product and services are sold.

CRM stands for Customer Relationship Management. It is a strategy used to learn more about customers' needs and behaviors in order to develop stronger relationships with them. After all, good customer relationships are at the heart of business success. There are many technological components to CRM, but thinking about CRM in primarily technological terms is a mistake. The more useful way to think about CRM is as a process that will help bring together lots of pieces of information about customers, sales, marketing effectiveness, responsiveness and market trends

Marketing Autoamation is the most comprehensive campaign management solution available. It provides everything needed to turn raw, disparate customer data into profitable marketing campaigns – all the way through inception, execution and measurement.

Marketing Optimization applies sophisticated mathematical approaches to optimize marketing campaign ROI given limited budgets, channel capacities and other organizational constraints.

The idea of CRM is that it helps businesses use technology and human resources to gain insight into the behavior of customers and the value of those customers. If it works as hoped, a business can:

- 1 . Provide better customer service
- 2 . Make call centers more efficient
- 3 . Cross sell products more effectively
- 4 . Help sales staff close deals faster
- 5 . Simplify marketing and sales process

## **1.2 PROBLEM DEFINITION**

A CRM system gives everyone — from sales, customer service, business development, recruiting, marketing, or any other line of business — a better way to manage the external interactions and relationships that drive success. A CRM tool lets you store customer and prospect contact information, identify sales opportunities, record service issues, and manage marketing campaigns, all in one central location — and make information about every customer interaction available to anyone at your company who might need it.

With visibility and easy access to data, it's easier to collaborate and increase productivity. Everyone in your company can see how customers have been communicated with, what they've bought, when they last purchased, what they paid, and so much more. CRM can help companies of all sizes drive business growth, and it can be especially beneficial to a small business, where teams often need to find ways to do more with less.

## **1.3 SCOPE**

Traditionally and essentially CRM is management software for sales, marketing and customer service teams as they are the major touchpoints for any customer contact strategy.

### **CRM for sales management**

A good mobile enabled CRM will allow sales reps to manage their tasks, activities and meetings from wherever they are, reducing unwanted administration time and building in best practice into prospect management.

Sales managers who need real time transparency in reviewing their team's sales pipeline and KPIs can do so with the use of configurable CRM dashboards.

## **CRM for marketing**

CRM can provide both marketing and sales functionality within their own operational requirements; but should also enable better collaboration and transparency between teams; putting the lead, prospect and customer at the heart of the CRM strategy.

Marketing teams can be supported in lead generation efforts, planning and executing multi-channel marketing campaigns, segment audiences, deliver targeted messaging at the right time, analyse marketing spend ROI and carry out best practice testing efficiently.

CRM system review should include:

1. Multi-channel marketing campaign management
2. Database management
- 3 . Social media engagement

## **CRM for customer service**

Customer service teams need to be able to deliver a positive customer experience and to do this they need to be supported by a CRM system that gives them a good record of past customer contacts, support enquiries, technical incidents and product history.

Customer service operators need to handle calls, emails and social/web enquiries quickly and to the satisfaction of the customer; but also keeping to service KPIs such as response times, resolution percentages and SLA adherence.

A CRM system should provide a 360 degree view of the customer and support operatives with information they need at their fingertips so they can resolve issues and handle queries efficiently.

When considering CRM functionality for customer service, organisations should review:

1. Customer service and contact management
2. Knowledge sharing and document management
3. Computer Telephony Integration (CTI)
4. SLA and contract management
5. Social listening integration

**2.1 SURVEY DESCRIPTION**

The expression of Customer Relationship Management (CRM) was being started to use since early 1990s. According to Customer Relationship Management: Concepts and Technologies (Buttle 2008), there are many attempts to define the domain of CRM .

CRM is an information industry term for methodologies, software and usually internet capabilities that help an enterprise manage customer relationships in an organized way.

CRM also define as the process of managing all aspects of interaction a company has with its customers, including prospecting, sales and service.

CRM is an integrated approach to identifying, acquiring and retaining customers. By enabling organizations to manage and coordinate customer interactions across multiple channels, departments, lines of business and geographies, CRM helps organizations maximize the value of every customer interaction and drive superior corporate performance.

CRM is an integrated information system that is used to plan, schedule and control the pre-sales and post-sales activities in an organization. CRM embraces all aspects of dealing with prospects and customers, including the call centre, sales-force, marketing, technical support and field service. The primary goal of CRM is to improve long-term growth and profitability through a better understanding of customer behavior. CRM aims to provide more effective feedback and improved integration to better gauge the return on investment (ROI) in these areas.

## **CHAPTER 3**

### **APPLICATION ANALYSIS**

#### **3.1 EXISTING APPLICATION**

The existing system in the organization is not completely computerized. The system is not working smoothly; therefore the organization has decided to replace it with a completely computerized one. The problems, which the existing system faces, are:

#### **Low Functionality:**

With the existing system, the biggest problem was the low functionality of the department. The problem faced hampered the work of the department. For all the tasks like entering the customer data, salesman data, product data, taking the orders, making Bill, making reports etc a large number of employees were appointed who would have been utilized in some other useful tasks.

##### **Erroneous Input And Output:**

In the existing system, humans performed all tasks. As in the human tendency, errors are also a possibility. Therefore, the inputs entered by the salesman in the registers may not be absolutely foolproof and may be erroneous. As a result of wrong input, the output reports etc will also be wrong which would in turn affect the performance of home appliance company.

#### **Portability:**

System that existed previously was manual. As a result, the system was less portable. One has to carry the loads of so many registers to take the data from one place to another. A big problem was that the system was less flexible and if one wanted to make a change would need to change in all the registers that would also prove to be big headache.

#### **Security:**

Security concerns were also one of the motives of the department for the need of the software. In the registers, the data is not secure as anybody can tamper with the data written in the registers. Also for the security of the registers, lots of problems arise to store the registers in a secure place and the appointments of security personnel can also cost a bit.



### **Data Redundancy:**

In the case of manual system, the registers are maintained in which, a lot of data is written. Therefore, there is a problem in the registers that the same data may be repeated again and again. Against the customer id, a lot of data will be repeated which will cause a lot of problems at the time of query as well as at the time of preparing the reports because a single data that will be left mistakenly will largely affect the report and subsequently, the performance of the department. In the software the concept of primary key and foreign key is used very efficiently, which will prevent the redundancy of data that will prove to be very beneficial to the organization because it will nullify the human error completely.

### **Processing Speed:**

In manual system, for a simple work, a number of employees are appointed and in case of keeping records of orders given by the customers, making Bill & reports, they take a lot of time, which may affect the performance of the organization as well as hamper the progress of the organization. It also affects the speed of working in the organization and the work that should have been performed in very short duration can take a large amount of time. But, in the case of software, all the tasks are performed at the touch of a key, which improves the performance of the organization, a great deal, thereby, improving the chances of progress of the organizations.

### **Manual Errors:**

When a number of tough tasks are prepared by the humans like preparation of reports, keeping records of all the customers, salesman & available stock in company etc then some human errors are obvious due to a number of factors like mental strain, tiredness etc. But as we all know that computer never gets tired irrespective of the amount of work it has to do. So, this software can nullify the probability of manual errors that improves the company performance.

### **Complexity In Work:**

In a manual system, whenever a record is to be updated or to be deleted a lot of cutting and overwriting needs to be done on all the registers that are concerned with the deleted or updated record, which makes the work very complex. However in the software, once a record is updated or deleted, all the concerned changes are made in the database automatically.

### **3.2 PROPOSED APPLICATION**

To overcome the existing system , there are 4 types of CRM implemented overall the business.

There are :

- 1.Strategic CRM ,
- 2.operational CRM ,
3. Analytical CRM and
4. Collaborative CRM .

Strategic CRM is use to development a business with customer centric culture. This culture is dedicated to winning and keeping customers by creating and delivering better value than the competitors. Customer centricity is compete with 3 others business logic, which is product, production and sales. Product-oriented businesses believe that the product with best quality, performance, design or features will be chosen by customer. Production-oriented businesses believe that customer more likely to choose low cost product. Sales-oriented businesses believe that customer will be persuaded to buy their product if they do enough for advertising, selling and public relation. A customer or market-oriented businesses share a set of belief putting the customer at first. Many managers say that customer centric must be right. However, sometime other orientation may stronger.

## **CHAPTER 4**

### **SYSTEM SPECIFICATION**

#### **4.1 HARDWARE SPECIFICATION**

SYSTEM : Computer and Server

Internet Connection : Mobile Data or WIFI

RAM : 2 GB

#### **4.2 SOFTWARE SPECIFICATION**

Operating System : windows 7 or above

Language : HTML,PHP AND SQL

Run base of : XAMPP

It is a modern but already mature programming language aimed to make developers happier. It's concise, safe, interoperable with Python and other languages, and provides many ways to reuse code between multiple platforms for productive programming.

Support for multiplatform programming is one of HTML, PHP and MY SQL benefits. It reduces time spent writing and maintaining the same code for different platforms while retaining the flexibility and benefits of native programming.

HTML AND PHP is a great fit for developing front - end applications. It allows you to write concise and expressive code while maintaining full compatibility with existing based technology stacks

### **4.3 SOFTWARE SPECIFICATIONS AND TOOLS**

The main tools or skill used to develop this project is Hyper Text Markup Language (HTML), Cascading style sheets (CSS), Javascript, PHP: Hypertext Preprocessor (PHP) and the Structured Query Language (SQL). HTML is a markup language and a markup language is a set of markup tags.

HTML documents contain HTML tags and plain text and it also called as a web page. CSS is use to define how to display HTML elements like font color, background color and others. CSS was added to HTML 4.0 and newer version to solve the problem of too many tags in HTML document. An external style sheet can help to save a lot of work and it is stored in CSS files.

Javascript is programming code that can be inserted into HTML pages and can be executed by modern web browser. Javascript is easy to learn and it's performance was powerful as JAVA and C++ language.

PHP is a server side scripting language and executed on server. PHP was supporting many databases like MySQL, Informix, Oracle, Sybase, Generic ODBC and others. PHP can be run on different platform like Windows and Linux, and it is also compatible with almost all servers used today like Apache and IIS.

SQL is a standard language use to accessing and manipulating databases. SQL can use to execute query against databases, retrieve data from databases, insert data to databases and other useful features.

Beside the language, the important technology to develop web application is the web server application. Web server application is application help to deliver the web content

## CHAPTER 5

### SYSTEM IMPLEMENTATION

#### 5.1 SYSTEM ARCHITECTURE

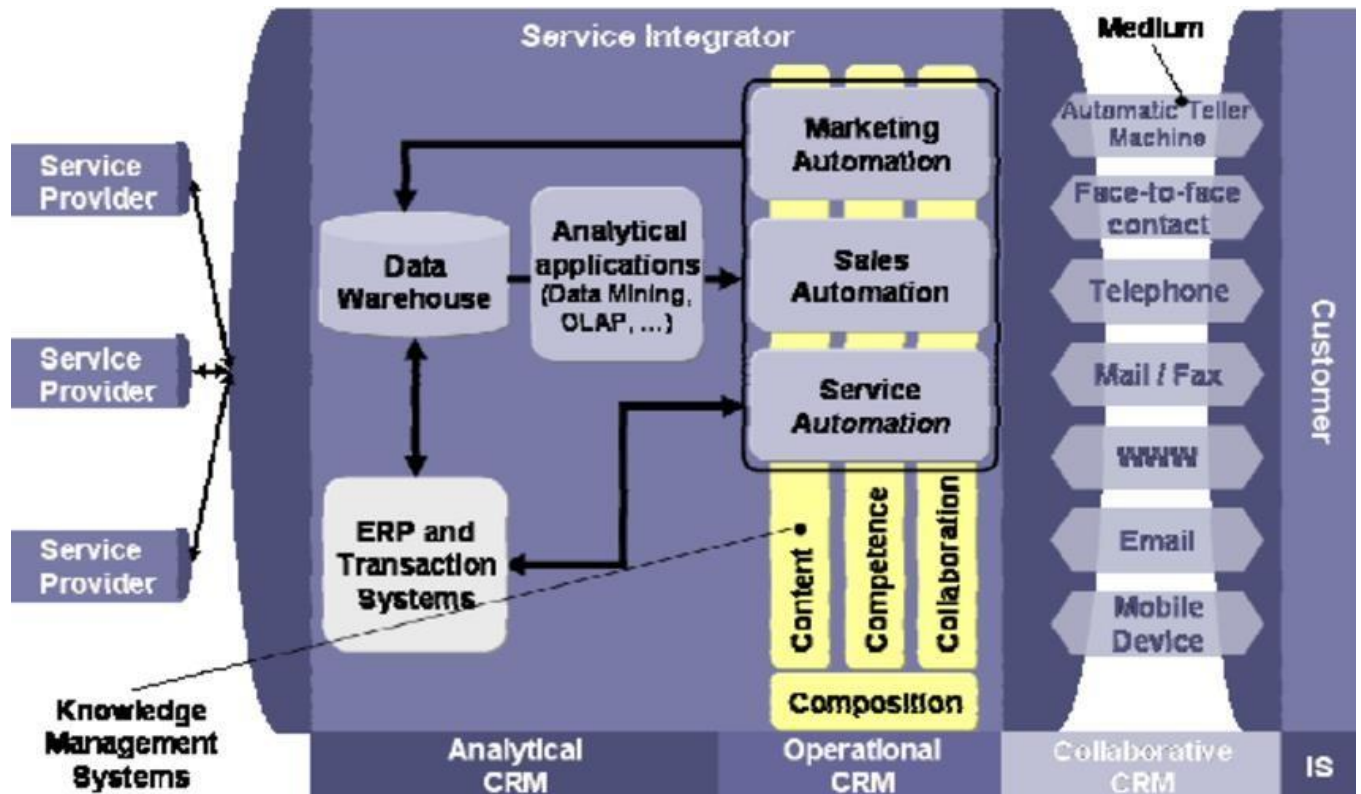


Fig.5.1 Architecture diagram

This CRM system is using nested include generating the dynamic content web pages. The system is using the index.php to include header, content, and footer. Beside this, the system also using jquery ajax to display some data generated by the PHP inside the Widget folder

## **Platform Independence**

The Write-Once-Run-Anywhere ideal has not been achieved (tuning for different platforms usually required), but closer than with other languages.

## **Object Oriented**

Object oriented throughout - no coding outside of class definitions, including main().  
An extensive class library available in the core language packages.

## **Compiler/Interpreter Combo**

Code is compiled to bytecodes that are interpreted by a xampp compiler

This provides portability to any machine for which a virtual machine has been written.

The two steps of compilation and interpretation allow for extensive code checking and improved security.

## **Automatic Memory Management**

1. Automatic garbage collection - memory management handled by database server .
- 2.No memory pointers
3. Programs runs inside the virtual machine sandbox.

## **Dynamic Binding**

1. The linking of data and methods to where they are located, is done at run-time.
2. New classes can be loaded while a program is running. Linking is done on the fly.
3. Even if libraries are recompiled, there is no need to recompile code that uses classes

## **Good Performance**

Interpretation of byte codes slowed performance in early versions, but advanced virtual machines with adaptive and just-in-time compilation and other techniques now typically provide performance up to 50% to 100% the speed of C++ programs.

## **Threading**

1. Lightweight processes, called threads, can easily be spun off to perform multiprocessing.
2. Can take advantage of multiprocessors where available
3. Great for multimedia displays.
4. PHP was designed with networking in mind and comes with many classes to develop sophisticated Internet communications.

## **CODING**

Coding is the phase of Software Life Cycle that produces the Actual code that will be delivered to the customer as the running System. The other phases of the life cycle may also develop code, such as prototypes, tests, and test drivers, but these are for the use by the developer. Individual modules developed in this system are also tested before being delivered to the next phase.

The design must be translated into a machine- readable form. This is what coding. Coding is basically translating the design into a machine-readable form. The code generation step performs this task. If design is performed in a detailed manner, code generation can be accomplished mechanistically.

While converting design into coding, following points are to be considered:

### **CODE EFFICIENCY**

The next thing, which we have to consider, is the “Code Efficiency”. The code that we write should be well efficient and error free. Because as much our code is efficient and error free, our software will be that much effective.

The Efficiency of the code or program depends on:

#### **Readability:**

Coding should be much and more readable as in such as Coding in HTML, PHP and MY SQL, etc.

**Easy Debugging:**

Coding should be such that errors could easily be removed or debugged. That is our coding should be as much as error free as possible.

**Easy Software Development:**

Software could easily be developed. Commands of programming language are similar to natural languages (English).

**Short & Precise Coding:**

Coding should not be lengthy. It should be short & precise. This factor is most important for the efficient coding.

**OPTIMIZATION OF CODE**

- Code Optimization is the last & final step in Code generation phase. Code optimization comes after the code generation.
- We have to optimize the code after generating it.
- Code optimization also plays a vital role as the efficiency of code plays in the successful development of the system.
- Code Optimization also effects the efficiency of code. Our code will be more efficient if it is more optimized.
- To produce faster and more compact, the code generator should include some form of “Code Optimization”
- This may exploit techniques such as the use of special purpose machine instructions or addressing modes, register optimization, etc.
- this code optimization may incorporate both machine dependent and machine independent techniques.



## **MODULES DESCRIPTION**

### **SALES MAN DATA MAINTENANCE MODULE**

This module is responsible for the updation of salesman\_master database. If any new salesman joins the organization or leave the organization or there is any change in the salesman data, then it can be done with the help of this module. Addition, deletion and updation of salesman data are done by this module.

### **PRODUCT DATA MAINTENANCE MODULE**

This module is responsible for the management of data of different product available in the database. Addition, deletion and updation of the product data is done by this module. At the time of adding product this module also add data in inventory and discount\_allowed table.

### **ORDER PROCESSING MODULE**

This module is responsible for taking order by the customer and updation of different tables such as bill\_details, transaction , inventory and salesman performance. Whenever any customer place order for any product firstly the salesman checks whether the product is available in the inventory or not. If there is sufficient amount of product in the inventory then a unique bill no. is generated and record is saved in bill\_details and transaction table. This module also generates bill.

### **STOCK PROCESSING MODULE**

Time to time updation of stock is necessary for the proper functioning of sales. Stock should be greater than reorder level, if stock is less than reorder level than addition of stock in the inventory is required. Time to time report of stock is generated and administrator and salesmen will get the information that which product is in stock and which one is not. If there is need of adding in the inventory than it can be done with the help of this module.

### **PRODUCT DATA MAINTENANCE**

This module is called whenever any new product is added , updated or deleted from the database. If any new record is added in the database than this module then user have to input details of product, inventory and price details. These details are saved in different tabled after performing validation checks. If there is any change in the features of the product than it also provide facility of editing. This is also responsible for providing

information related to different product on demand.

## **QUOTA ASSIGNMENT**

This module is responsible for assigning quota of different salesman for different products. Quota is assigned for one year. Quota assignment is very important for estimating performance of the company and comparing actual performance with standard performance.

## **ANALYSIS MODULE**

Analysis module is responsible for performing analysis on different sales, customer, product data to analyze sales performance, customer sales behavior and sales trend . Different modules responsible for performing analysis are

### **SALES DATA ANALYSIS**

This Module is responsible for analyzing sales data on the basis of territory, product and salesman. This module compares quota with sale. Also tells with the help of ratios that whether actual sale is above standards or below.

### **CUSTOMER DATA ANALYSIS**

This module analyzes data of customer on the basis of territory, occupation and income group. This module uses customer master, city details, transaction, bill details and complaint. It gives information related to customer satisfaction about particular product. It compares actual sale with complaint and gives ratio between sale and complaint. High ratio mean low customer satisfaction.



## **CUSTOMER DATA MAINTENANCE**

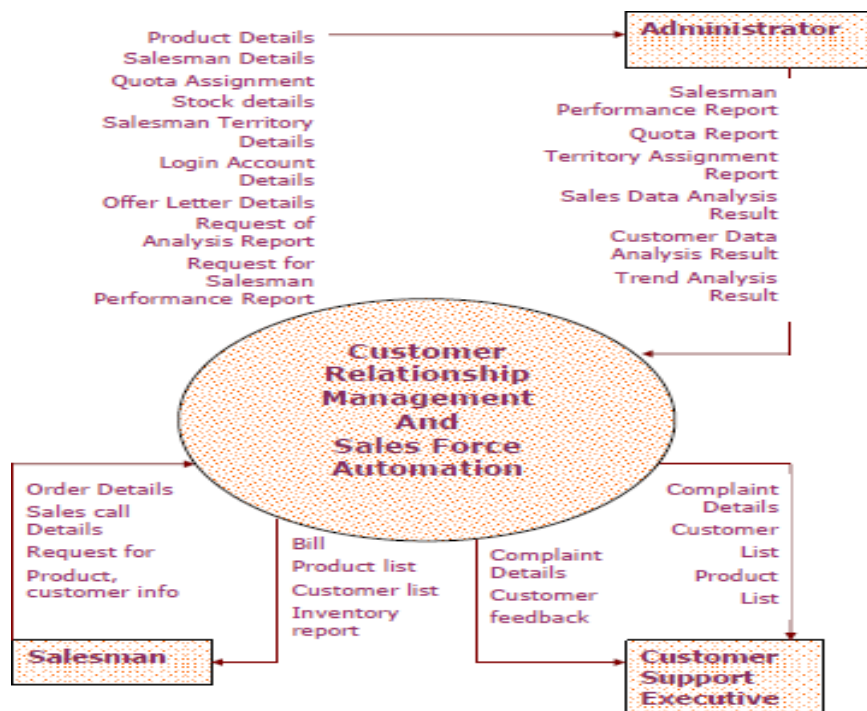
This module is responsible for management of customer data. This data is accessed and maintained by the salesman. If product is sold to a new customer then data of new customer can be added in the database. If there is any change in the customer record such as change of address etc then the salesmen can edit their records with the help of this module.

## CHAPTER 6

### SYSTEM DESIGN

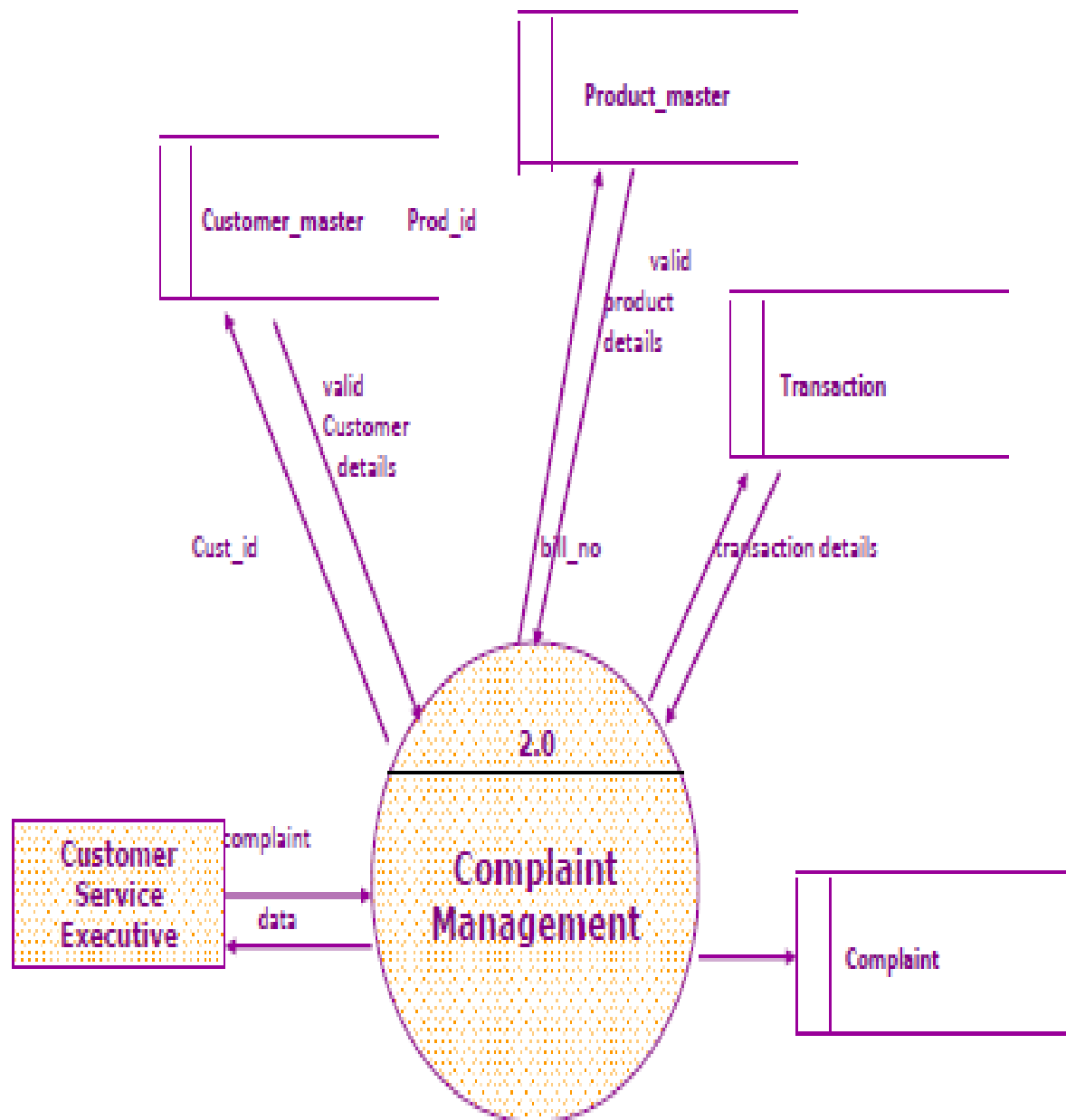
#### 6.1 DATA FLOW DIAGRAM

Data flow diagram (DFD) is a graphical representation of the "flow" of data through an information system. It differs from the flowchart as it shows the data flow instead of the control flow of the program. A data flow diagram can also be used for the visualization of data processing. The DFD is designed to show how a system is divided into smaller portions and to highlight the flow of data between those parts.



**Context Level DFD Of Customer Relationship Management and Sales Force Automation**

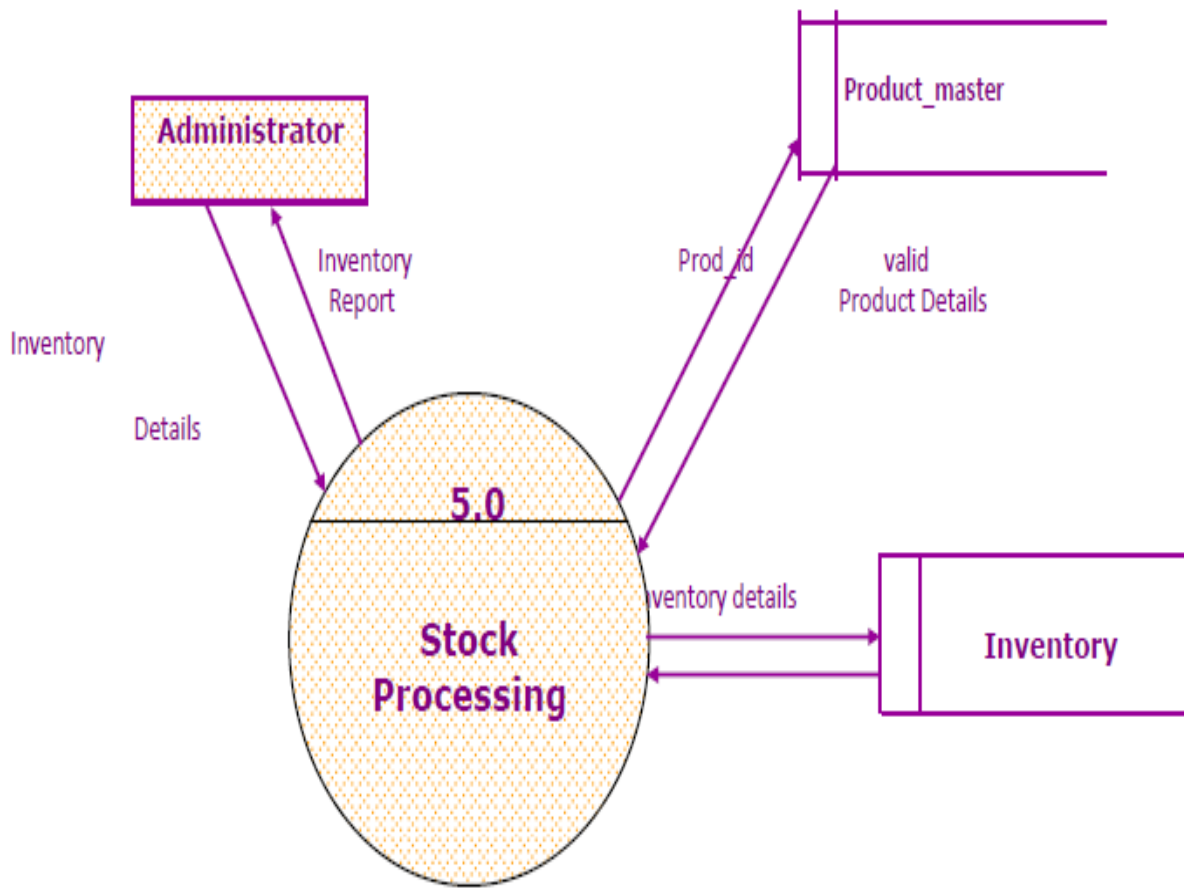
By context level dfd of customer relationship management and sales force automation we can maintained by customer relationship website to browse the my admin [www.https:\\crm\\admin](http://www.https:\\crm\\admin)



1st Level DFD of Complaint Management

### 6.1.1 LEVEL 1

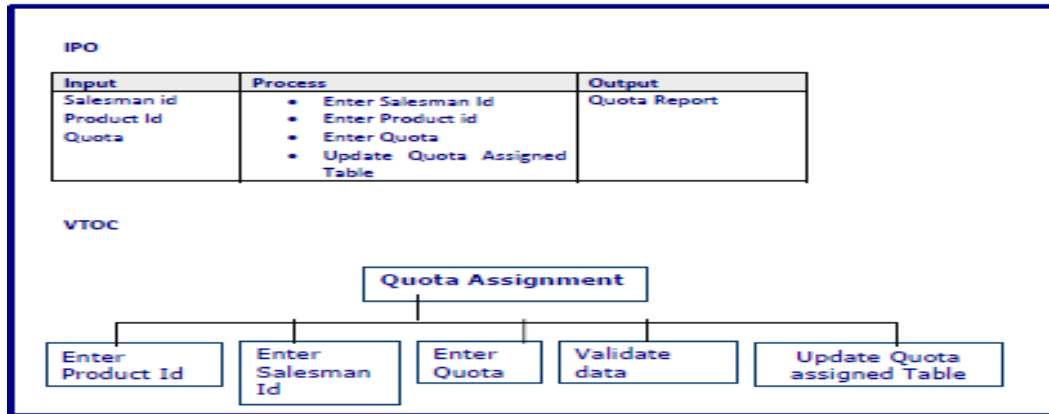
In fig 6.1.2, Preprocessed training data and validation data are passed on to the CRM algorithm to create the model. This model will generate the accuracy and loss with respect to the training data and it can be visualized. Using test data, CRM model can be evaluated



1st Level DFD of Stock Processing Module

ed.

HIPO Diagram of Quota Assignment

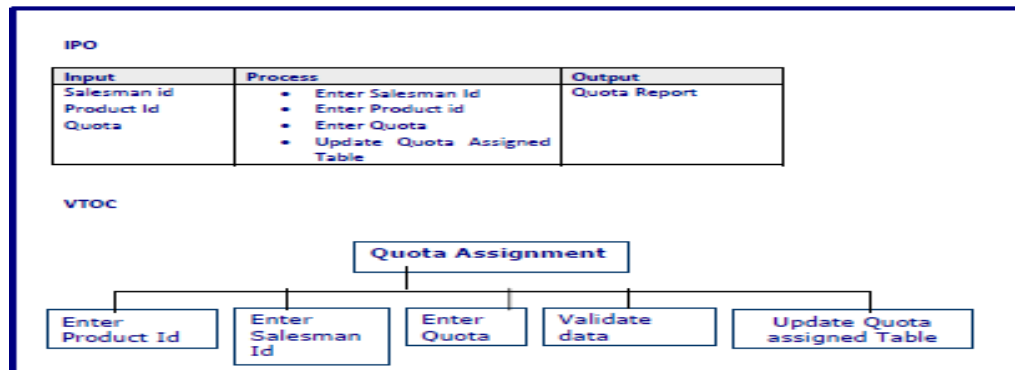


HIPO algorithm flow chart analysis

## 6.1.2 LEVEL 2

In fig 6.1.3, Preprocessed training data and validation data are passed on to the algorithm to create the model. This model will generate the accuracy and loss with respect to the training data and it can be visualized. Using test data model can be evaluated.

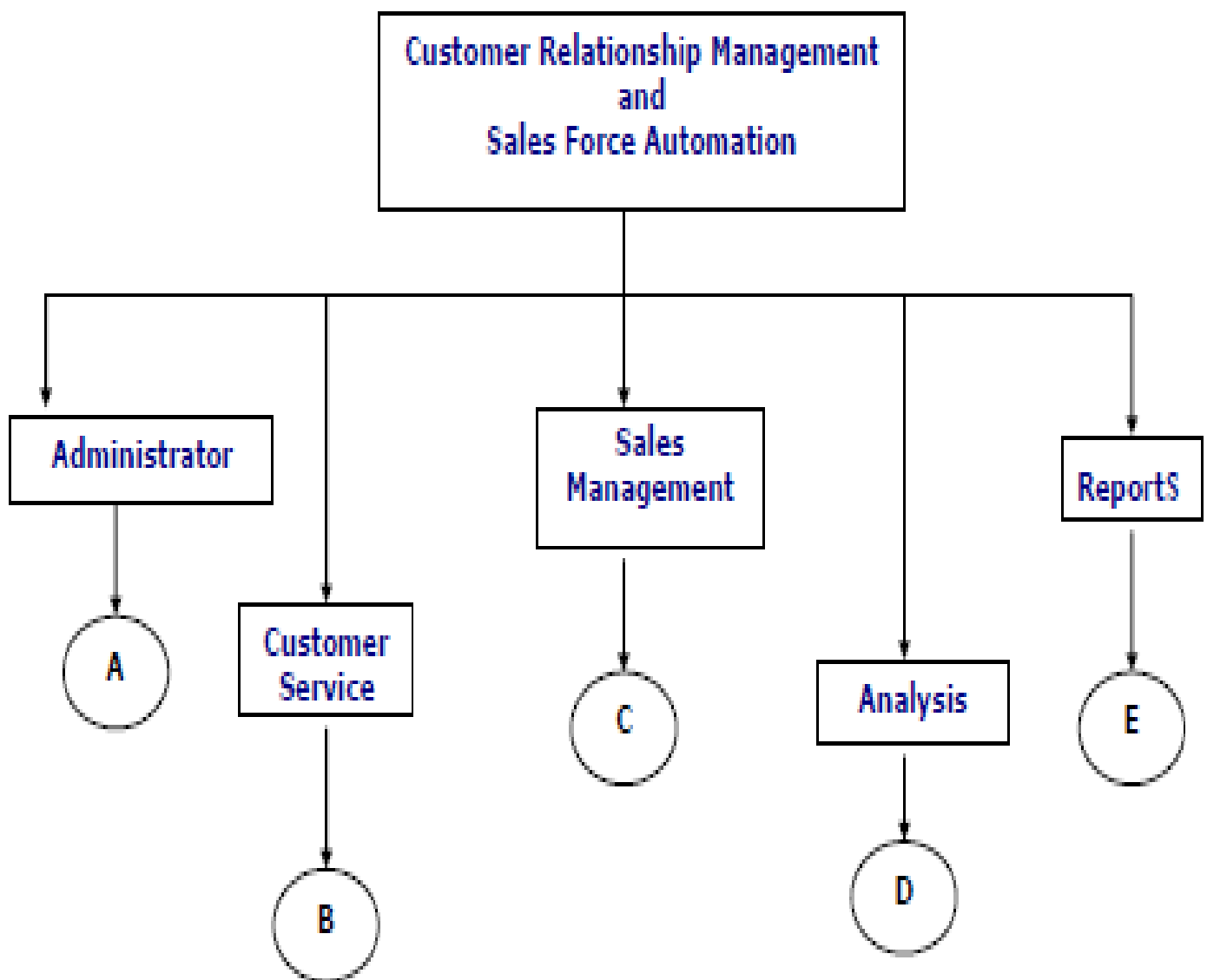
HIPO Diagram of Quota Assignment



Amount estimation by validating

### 6.1.3 LEVEL 3

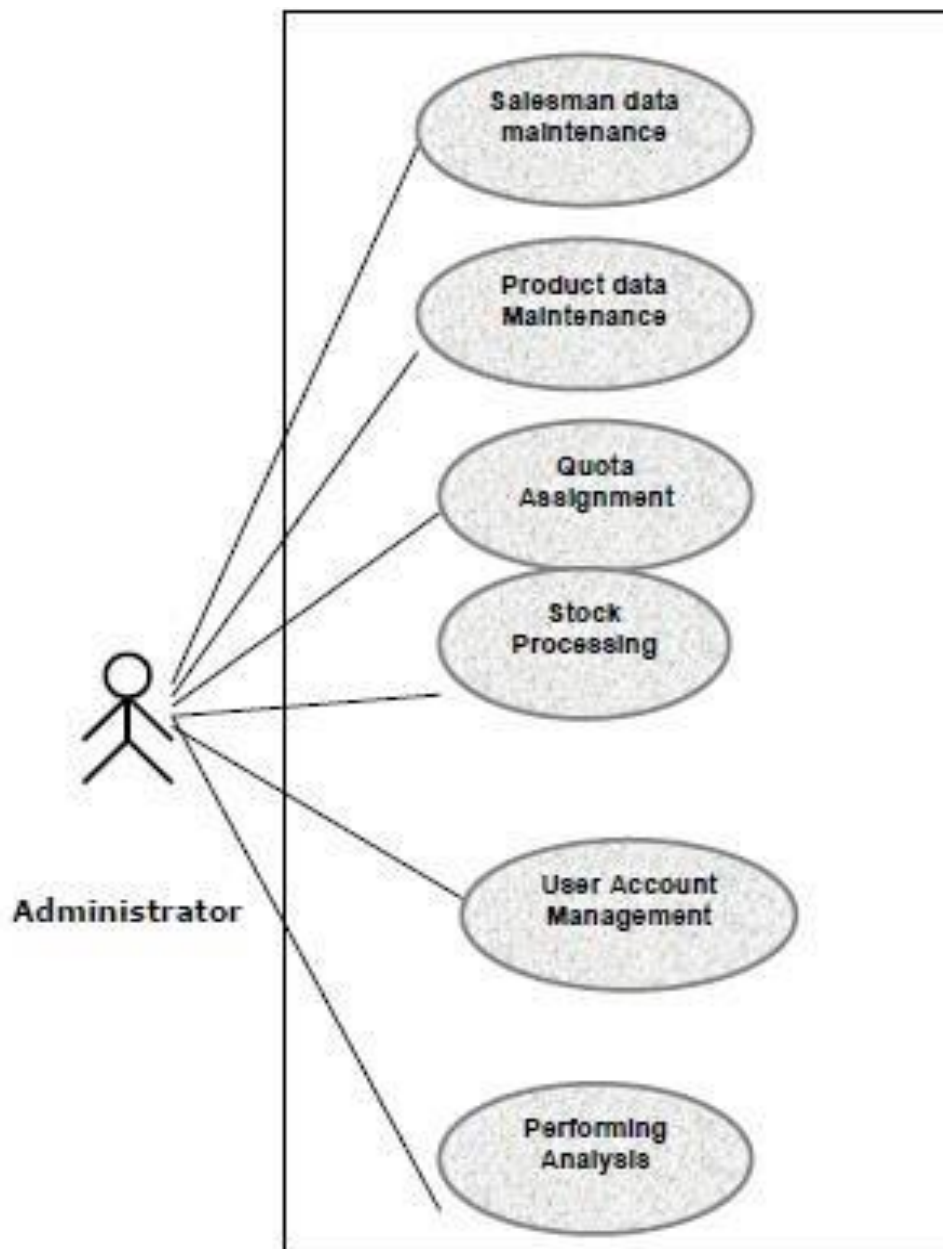
System Chart

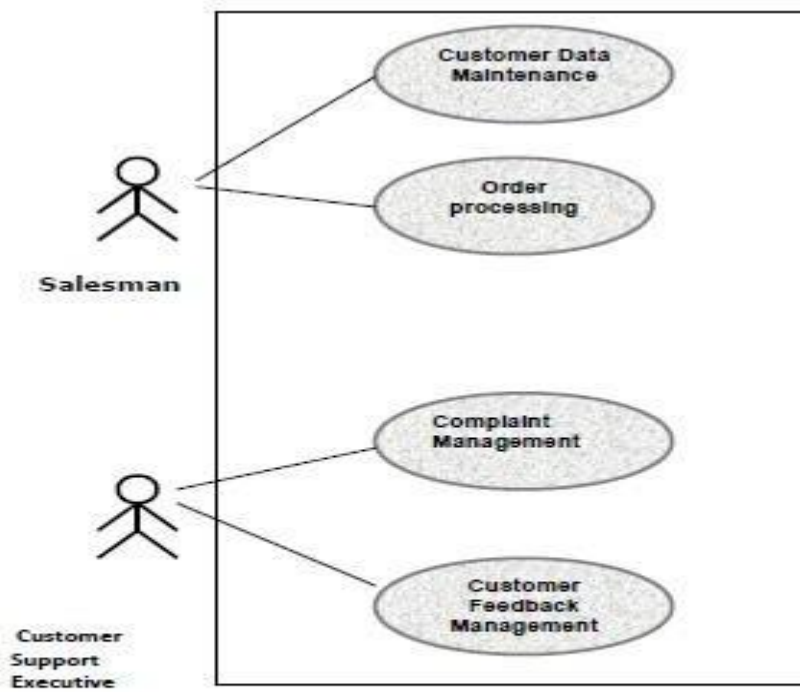




## 6.21 UML DIAGRAMS

### 6.22 USE CASE DIAGRAMS

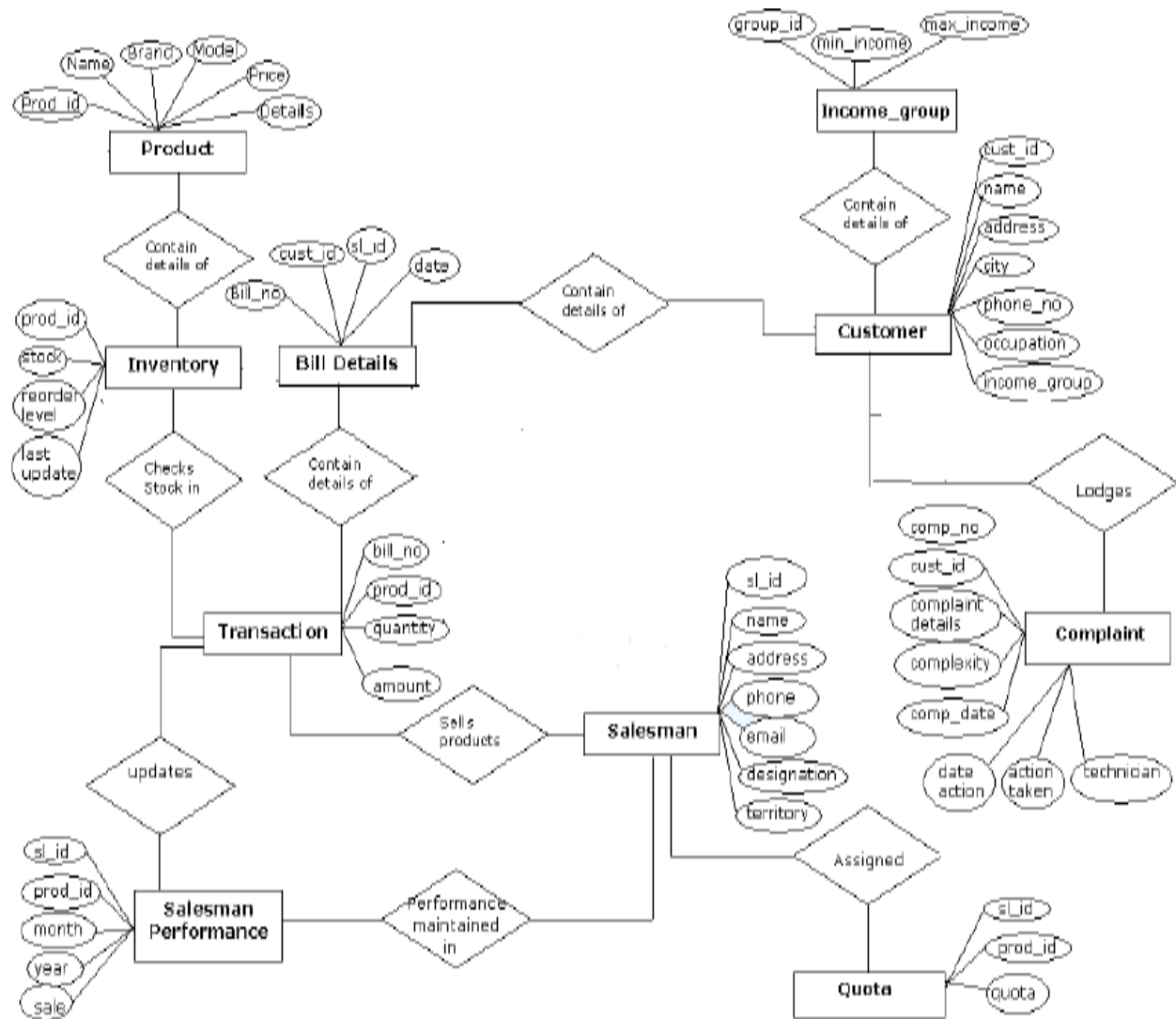




**Use Case Diagram of Customer Relationship Management and sales force Management System**

## **USE DIAGRAMS FOR CRM AND SALES FORCE**

## ENTITY RELATIONSHIP MODEL FOR SALES FORCE USAGE



### ER Diagram of CRM: Sales Force Automation

## **SOFTWARE TESTING**

Software testing is the process used to help identify the correctness, completeness, security, and quality of developed computer software. Testing is a process of executing a program or application with the intent of finding errors. With that in mind, testing can never completely establish the correctness of arbitrary computer software. In other words, testing is criticism or comparison that is comparing the actual value with an expected one. An important point is that software testing should be distinguished from the separate discipline of software quality assurance, which encompasses all business process areas, not just testing.

### **TESTING OBJECTIVES :**

Testing is a process of executing a program with the intent of finding an error.

A good test case is one that has a high-probability of finding an as-yet-undiscovered error.

A successful test is one that uncovers an as-yet-undiscovered error.

### **Software Testing Techniques**

The importance of software testing and its impact on software cannot be underestimated.

Software testing is a fundamental component of software quality assurance and represents

a review of specification, design and coding. The greater visibility of software systems and the cost associated with software failure are motivating factors for planning, through testing. It is not uncommon for a software organization to spent 40% of its effort on testing.

## **White Box Testing**

White box testing is a test case design approach that employs the control architecture of the procedural design to produce test cases. Using white box testing approaches, the software engineering can produce test cases that

- (1) Guarantee that all independent paths in a module have been exercised at least
- (2) Exercise all logical decisions
- (3) Execute all loops at their boundaries and in their operational bounds
- (4) Exercise internal data structures to maintain their validity.

## **Various white box testing**

### **Basis Path Testing**

Basic path testing is a white box testing techniques that allows the test case designer to produce a logical complexity measure of procedural design and use this measure as an approach for outlining a basic set of execution paths. Test cases produced to exercise each statement in the program at least one time during testing.

## **Control Structure Testing**

Although basis path testing is simple and highly effective, it is not enough in itself. Next we consider variations on control structure testing that broaden testing coverage and improve the quality of white box testing. Different control structure techniques are

- a. Condition testing
- b. Data flow testing
- c. Loop testing

## **Black Box Testing**

Black Box Testing is not a type of testing; it instead is a testing strategy, which does not need any knowledge of internal design or code etc. As the name "black box" suggests, no knowledge of internal logic or code structure is required. The types of testing under this strategy are totally based/focused on the testing for requirements and functionality of the work product/software application.

## **Various black box techniques**

**Functional Testing:**

In this type of testing, the software is tested for the functional requirements. The tests are written in order to check if the application behaves as expected.

**Smoke Testing:**

This type of testing is also called sanity testing and is done in order to check if the application is ready for further major testing and is working properly without failing up to least expected level.

**Recovery Testing:**

Recovery testing is basically done in order to check how fast and better the application can recover against any type of crash or hardware failure etc. Type or extent of recovery is specified in the requirement specifications.

**Alpha Testing:**

In this type of testing, the users are invited at the development center where they use the application and the developers note every particular input or action carried out by the user. Any type of abnormal behavior of the system is noted and rectified by the developers.

**Beta Testing:**

In this type of testing, the software is distributed as a beta version to the users and users test the application at their sites. As the users explore the software, in case if any

exception/defect occurs that is reported to the developers.

### **Software Testing Strategies in used in the project**

A strategy for software testing integrates software test case design techniques into a well-planned set of steps that cause the production of software. A software test strategy provides a road map for the software developer, the quality assurance organization, and the customer

#### **Unit testing**

Unit testing concentrates verification on the smallest element of the program – the module. Using the detailed design description important control paths are tested to establish errors within the bounds of the module.

Firstly the unit testing on various modules and sub modules is performed in the project. Different modules are tested with different correct and incorrect data. For example in the order processing module order of 0 product is not allowed so in this case different methods are used to find out whether the modules is performing all processes correctly. All modules are tested and check it is running working properly.



### **Integration testing :-**

Once all the individual units have been tested there is a need to test how they were put together to ensure no data is lost across interface, one module does not have an adverse impact on another and a function is not performed correctly. Integration testing is a systematic approach that produces the program structure while at the same time producing tests to identify errors associated with interfacing.

In this project bottom up integration testing is used. Firstly lower level modules are tested. As modules are integrated bottom up, processing required for modules subordinates to a given level is always available and the need for stubs is eliminated.

### **Validation testing :-**

As a culmination of testing, software is completely assembled as a package, interfacing errors have been identified and corrected, and a final set of software tests validation testing are started. Validation can be defined in various ways, but a basic one is valid succeeds when the software functions in a fashion that can reasonably expected by the customer.

- In the first phase of alpha testing, developers test the software using white box techniques. Additional inspection is then performed using black box techniques. This is usually done by a dedicated testing team.
- This is often known as the second stage of alpha testing.

- Unit and integrated tests concentrate on functional verification of a component and incorporation of components into a program structure.
- Validation testing demonstrates tractability to software requirements, and system testing validates software once it has been incorporated into a larger system.

### **Test Cases Specification for System Testing**

The different conditions that need to be tested, along with the test cases used for testing those conditions and expected output are given. The goal is to test different functional requirements, as specified in requirement document. Test cases have been selected for both valid and invalid inputs.

#### **Test Cases for Adding New Customer Data:**

Seq. No	Test Fields	Condition Being Checked	Expected Output
1.	TCustName	Cust_name=""	Customer name is blank
2.	TAddress	Address=""	Address is blank
3.	CCity	City' s selected index is 0	Select city
4.	TPhone	Check whether given phone contain only numbers or not	Phone number should contain number only
5.	TMobile	Check whether given phone contain only numbers or not	Mobile number should contain number only
6.	Tmobile	Length is greater than 11	Length of number should be 10
7.	CSalary	Selected index is 0	Select salary
8.	COccupation	Selected index is 0	Select Occupation
9.	TDOB	Check whether date is in correct format	Input Date in correct format

## Test Cases for Order Processing Module

Seq. No	Test Fields	Condition Being Checked	Expected Output
1.	TBillDate	Format of Date is correct or not	Input date in correct format
2.	TbillDate	Date is blank	Enter Bill Date
3.	CCustId	Customer Id's selected index is 0	Select Customer Id
4.	CSId	Salesman id's selected index is 0	Select salesman Id
5.	CProdId	Product Id's selected index is 0	Select product id

6.	TQuantity	Quantity is blank or 0	Enter Quantity correctly
7.	Count	Count=0	Enter product details

### Test Cases for Stock Processing Module

Seq. No	Test Fields	Condition Being Checked	Expected Output
1.	CProdId	Product id's selected index is 0	Select product id
2.	TStock	Stock is 0	Enter stock
3.	TStock	Stock field contain alphabet or special char	Enter stock correctly

### Test Cases for Quota Assignment Module:

Seq. No	Test Fields	Condition Being Checked	Expected Output
1.	CProdId	Product id's selected index is 0	Select product id
2.	CSId	Salesman id's selected index is 0	Select salesman Id
3.	TQuota	Quota is blank	Enter Quota
4.	TQuota	Quota Contain alphabets	Enter Quota Correctly

### Test Cases for New Completed Module

#### Test Cases for Quota Assignment Module:

Seq. No	Test Fields	Condition Being Checked	Expected Output
1.	CProdId	Product id's selected index is 0	Select product id
2.	CSId	Salesman id's selected index is 0	Select salesman Id
3.	TQuota	Quota is blank	Enter Quota
4.	TQuota	Quota Contain alphabets	Enter Quota Correctly

#### Test Cases for New Complaint Module

Seq. No	Test Fields	Condition Being Checked	Expected Output
1.	CProdId	Product id's selected index is 0	Select product id
2.	TCustId	Customer Id's selected index is 0	Select customer Id
3.	TDescription	Description is blank	Enter Complaint Descriptions
4.	TDOC	Date of complaint is blank	Enter complaint date
5.	TDOC	Date of complaint is in wrong format	Enter date in correct format

#### Test Cases for Pending Complaint module

Seq. No	Test Fields	Condition Being Checked	Expected Output
1.	CCompld	Complaint Id is not selected	Select Complaint id
2.	TDateAction	Date is in wrong format	Enter Date of action correctly
3.	TEctnician	Technician name is blank	Enter technician name
4.	TActionTaken	Action taken is blank	Enter action taken to solve the complaint

## SYSTEM IMPLEMENTATION

The purpose of **System Implementation** can be summarized as follows: making the new system available to a prepared set of users (the deployment), and positioning on-going support and maintenance of the system within the Performing Organization (the transition). At a finer level of detail, deploying the system consists of executing all steps necessary to educate the Consumers on the use of the new system, placing the newly developed system into production, confirming that all data required at the start of operations is available and validating that business functions that interact with the system are functioning properly. Transitioning the system support responsibilities involves changing from a system development to a system support and maintenance mode of operation, with ownership of the new system moving from the Project Team to the Performing Organization.

Once the design model of “**CRM: Sales System**” is created, it is implemented as a prototype, examined by end-users and modified by developers, i.e. us, based on their comments. To accommodate this iterative design approach, a broad class of interface design and prototyping tools has evolved.

This phase consists of the following processes:

### **Prepare for System Implementation :**

Where all steps needed in advance of actually deploying the application are performed, including preparation of both the production environment and the Consumer communities.

### **Deploy System:**

Where the full deployment plan, initially developed during System Design and evolved throughout subsequent lifecycle phases, is executed and validated.

### **Transition to Performing Organization:**

Where responsibility for and ownership of the application are transitioned from the Project Team to the unit in the Performing Organization that will provide system support and maintenance.

### **Prepare for system implementation**

- In the implementation of any new system, it is necessary to ensure that the Consumer community is best positioned to utilize the system once deployment efforts have been validated. Therefore, all necessary training activities must be scheduled and coordinated. As this training is often the first exposure to the system for many individuals, it should be conducted as professionally and competently as possible. A positive training experience is a great first step towards Customer acceptance of the system.
- Often the performance of deployment efforts impacts many of the Performing Organization's normal business operations. Examples of these impacts include:
  - Consumers may experience a period of time in which the systems that they depend on to perform their jobs are temporarily unavailable to them.
  - Technical Services personnel may be required to assume significant implementation responsibilities while at the same time having to continue current levels of service on other critical business systems.

- Technical Support personnel may experience unusually high volumes of support requests due to the possible disruption of day-to-day processing.

## **DEPLOY SYSTEM**

Deploying the system is the culmination of all prior efforts where all of the meetings, planning sessions, deliverable reviews, prototypes, development, and testing pay off in the delivery of the final system.

It is also the point in the project that often requires the most coordination, due to the breadth and variety of activities that must be performed. Depending upon the complexity of the system being implemented, it may impact technical, operational, and cultural aspects of the organization.

A representative sample of high-level activities might include the installation of new hardware, increased network capabilities, deployment and configuration of the new system software, a training and awareness campaign, activation of new job titles and responsibilities, and a completely new operational support structure aimed at providing Consumer-oriented assistance during the hours that the new system is available for use (to name a few).



## **TRANSITION TO PERFORMING ORGANIZATION**

In many organizations, the team of individuals responsible for the long-term support and maintenance of a system is different from the team initially responsible for designing and developing the application. Often, the two teams include a comparable set of technical skills. The responsibilities associated with supporting an operational system, however, are different from those associated with new development. In order to affect this shift of responsibilities, the Project Team must provide those responsible for system support in the Performing Organization with

a combination of technical documentation, training, and hands-on assistance to enable them to provide an acceptable level of operational support to the Consumers. This system transition is one element

## **SYSTEM MAINTENANCE**

Software once designed & developed needs to be maintained till end of software / system life. When the system is fully implemented, analyst must take precautions to ensure that the need for maintenance is controlled through design and testing and the ability to perform it is provided through proper design practices.

Maintenance is necessary to eliminate errors in the working system during its working life and to tune the system to any variations in its working environment.

Often small system deficiencies are found as a system is brought into operations and changes are made to remove them. System planners must always plan for resource availability to carry out these maintenance functions. The importance of maintenance is to bring the new system to standards.

The term “Software Maintenance” is commonly used to refer to the modification that are made to a software system in its initial release.

Maintenance requirements for information systems

(a) From 60% to 90% of the overall cost of software during the life of a system is spend on maintenance.

(b) Often maintenance is not done very efficiently.

(c) Software demand is growing at a faster rate than supply. Many programmers spending more time on systems maintenance than on new software development.

The keys to reduce the need of maintenance , while making it possible to do essential tasks more efficiently, are as follows :

- More accurately defining the user’s requirements during systems development.
- Making better Systems documentation.
- Using proper methods of designing processing logic and communicating it to project team members.

- Utilising the existing tools and techniques in an effective way.
- Managing the systems engineering process in a better and effective way.

## **SYSTEM SECURITY MEASURES**

The System designed & developed is to be kept secure from various persons or users who are not allowed to use the software. Various Security measures are taken to prevent the Software / System from other persons or users.

In this System, users are divided in three category-

- Administrator has authority of maintaining salesman data, product data, assigning territory to the salesman, stock processing, offer letter processing, quota assignment, creating user account. Administrator can also provide facility of analyzing customer, sales, and complaint data.
- Salesman has authority of adding customer details, order processing, sales call data management.
- Customer Support executive has authority of complaint processing and entering customer feedback.

Different authority is assigned different user. Each user has a user name and password to access the system. Every time when user will open the system, user has to give his/her username and password in the login window. Access category is assigned to every user.

## **COST ESTIMATION OF THE PROJECT**

The most fundamental calculation in the COCOMO model is the use of the Effort Equation to estimate the number of Person-Months required to develop a project. Most of the other COCOMO results, including the estimates for Requirements and Maintenance, are derived from this quantity.

The COCOMO calculations are based on your estimates of a project's size in Source Lines of Code (SLOC). SLOC is defined such that:

- ✚ Only Source lines that are DELIVERED as part of the product are included -- test drivers and other support software is excluded
- ✚ SOURCE lines are created by the project staff -- code created by applications generators is excluded
- ❖ One SLOC is one logical line of code
- ❖ Declarations are counted as SLOC
- ❖ Comments are not counted as SLOC

In the COCOMO II model, some of the most important factors contributing to a project's duration and cost are the Scale Drivers. You set each Scale Driver to describe your project; these Scale Drivers determine the exponent used in the Effort Equation.

The 5 Scale Drivers are:

- Precedentedness
- Development Flexibility
- Architecture / Risk Resolution
- Team Cohesion
- Process Maturity

It is analyzed that the project will benefit the organization in terms of time saving as well as cost wise. It is going to improve accuracy at each phase of the system. If we consider the Benefits gained by the system we can see that the software is feasible in economical point of view.

## **REPORTS**

Various Reports are generated in this project-

1. Salesman Performance Report

2. Customer Report

3. Product Report

4. Inventory report

5. Complaint Reports

6. Quota Report

7. Offer Letter

8. Bill

9. Analysis Reports

### **1. Salesman Performance Report-**

This report will display performance of the sales person. It will check display both standard performance and actual performance of sales person. With the help of this report administrator can get information that what quota assigned to salesman and what is the actual sale done by the salesman.

### **2. Customer Report**

This report generate list of customer on the basis of demand submitted by the user. For example if user want list of particular city than he can submit it and he will obtain it. Report generation on the basis of multiple criteria is also possible such as on the basis of city, income group and Occupation.

### **3. Product Report**

This report will generate detail product description and list of product list on the demand of customer.

### **4. Inventory Report**

This report provides information related to inventory of different products.

## **5. Quota Report**

This report will display details of how much quota assigned to which sales person.

Administrator has to provide salesman Id the report will generate accordingly.

## **6. Complaint Report**

This report will generate detail of complains and their solution for a particular period.

This report will show details of different types of complain, which product has maximum number of complains and which product has no complain for particular period.

## **7. Offer Letter**

Offer Letter is generated on the basis of condition provided. This report will display details of offer letter along with details of customer to whom it will be send.

## **8. Bill**

Bill is generated on the time of sale of any product. Bill contains information related to customer, salesman, and product details.

## **9. Analysis Reports**

Administrator can get various analysis report on the basis of sales data, customer data.



## **CHAPTER 8**

## **SYSTEM TESTING**

### **8.1 TESTING OBJECTIVES**

Testing is the stage of implementation, which is aimed at ensuring that the system works accurately and efficiently before live operation commences. The logical design and physical design is thoroughly and continually examined on paper to ensure that they will work when implemented. Thus the system test in implementation was a confirmation that all is correct and an opportunity to show the users that the system works.

Testing of the online classified system was performed in three stages which are as follows:-

- Unit Testing
- Integration Testing
- System Testing

### **8.2 SYSTEM TEST**

System tests are designed to validate a fully developed system with a view to assuring that it meets its requirements. There are three types of system testing which are as follows:-

### **8.3 TESTING LEVEL**

#### ***Alpha Testing:***

- The initial testing of a computer program or system under actual usage conditions, it can be done in-house by the vendor, or outside by a customer or third party tester.
- Acceptance Testing performed by the customer in a controlled environment.

- The software used by the customer in a setting approximating the target environment with the developer observing and recording errors and usage problems.

### ***Beta Testing:***

Beta Testing is done after alpha testing. The main purpose of Beta Testing is as follows:-

- Testing done by the potential or existing users, customers and end users at the external site without developers involvement is known as beta testing.
- It is operation testing i.e. it tests if the software satisfies the business or operational needs of the customers and end users.
- Beta Testing is done for external acceptance testing of COTS (Commercial off the Shelf) software.

### ***Test Case Design:***

Any engineered product (and most other things) can be tested in one of the following two ways. Knowing the specified function that a product has been designed to perform, tests can be conducted to demonstrate each function is fully operational.

Knowing the internal working of a product, tests can be conducted to ensure that —all gears mesh, that is the internal operation of the product performs according to the specification and all internal components have been adequately exercise.

### 8.3.1 UNIT TESTING

Unit testing is undertaken when a module has been coded and successfully reviewed.

This can be done by two methods:

*a) Black Box testing*

*b) Equivalence Class Partitioning*

***a) Black Box Testing***

Test cases are designed from an examination of the input/output values only and no knowledge of designing or coding is required the following are the two main approaches of designing black-box test cases.

***b) Equivalence Class Partitioning***

The domain of input values to a program is partitioned into a set of equivalence classes. This partitioning is done on such way that the behaviour of the program is similar to every boundary value analysis. Boundary value analysis leads to selection of the test cases at the boundaries of different equivalence classes.

Testing done by: Team Member

### 8.3.2 INTEGRATION TESTING

During integration testing different modules of the system are integrated using integration plan. The integration plan specifies the steps and the order in which modules are combined to realize the full system.

Purpose:

- To test whether the module performs its intended task.
- Once all the modules have been integrated and tested, system testing can start.

### **8.33 VALIDATION TESTING**

The output generated is validated with the minimum support count and confidence. The frequent patterns generated by Apriori is validated with Eclat algorithm. In the same way, the association rules are also validated.

There are also different types of testing. They are

#### ***White Box Testing:***

White-box testing is a methodology used to ensure and validate the internal framework, mechanisms, objects and components of a software application. White-box testing verifies code according to design specifications and uncovers application vulnerabilities.

White-box testing is also known as transparent box testing, clear box testing, structural testing and glass box testing. Glass box and clear box indicate that internal mechanisms are visible to a software engineering team.

White-box testing advantages include:

- Enables test case reusability and delivers greater stability
- Facilitates code optimization
- Facilitates finding of the locations of hidden errors in early phases of development
- Facilitates effective application testing
- Removes unnecessary lines of code

### ***Regression Testing :***

It is a type of software testing i.e. carried out by software testers as functional regression tests & developers as Unit Regression Tests.

- Objective of regression tests are to find defects that got introduced to detect fixes or introduction of new features.
- Regression tests are ideal candidate for automation

### ***Accessibility Testing :***

This is a formal type of software testing that helps to determine whether the software can be used by people with disability.

There are also companies & consultants that provide website accessibility audits.

### ***Ad-hoc Testing :***

Ad hoc testing is an informal and improvisational approach to assessing the viability of a product.

- An ad-hoc is usually only conducted once unless a defect is found.
- Commonly used in software development, ad hoc testing is performed without a plan of action and any actions taken are not typically documented.
- Testers may not have detailed knowledge of product requirements.
- Ad hoc testing is also referred to as random testing and monkey testing.

## **CHAPTER 8**

### **FUTURE SCOPE OF THE PROJECT**

- This application can be implement in those organizations where there is need for automation of sales work.
- This application minimize manual work in the organization so work load on organization will decrease and efficiency and effectiveness of the organization will increase.
- This application helps in reducing redundancy of the data and provides security so that unauthorized person cannot access the application.
- The database management in the organization becomes more reliable. This application provide facility to analyze data of customer, sale and complains.
- This can be highly useful in planning marketing and sales strategy of the organization.
- It attempts to integrate and automate the various customer-serving processes within a company.
- CRM is a strategy used to learn more about customers' needs and behaviors in order to develop stronger relationships with them.
- Good customer relationships are at the heart of business success. So this application will help in building good customer relation with customer.
- With the help of this project, the organization will be able to managing customer data in effective way.
- Proper management of sales and customer data will provide facility of obtaining customer details time to time on the basis of which organizatiocan make marketing and sales plan effectively.

## **CONCLUSION AND FUTURE ENHANCEMENT**

There is always room for improving in any software package, however good efficient it may be. But the important thing is that the system should be flexible enough for future modification/ alteration whenever and by whomsoever it may be. Keeping in consideration this important factor, the system is designed in such a way. The software is developed in modules are efficient enough to introduce any change in the software to get more information.

- ❖ Similarly, the present system can be implemented on Internet and software can be connected to the various branches of this home appliance company of course with more security constraints added to it.
- ❖ This project can be attached to website of the company which may provide information related to products and also provide facility of registering product etc. It may also help in finding new prospective customer.
- ❖ The backend can be improved using Oracle; it will provide better database management and securities.
- ❖ More modules can be added in the system such as it can provide facility of direct email so that organization can generate offer letter and send it directly to the customer.

## APPENDIX A:

### SOURCE CODE

#### # Source Code for HOME PAGE

```
<?php
session_start();
include("dbconnection.php");
include("checklogin.php");
check_login();
?>
<!DOCTYPE html>
<html>
<head>
<meta http-equiv="content-type" content="text/html; charset=UTF-8" />
<meta charset="utf-8" />
<title>CRM | Admin Dashboard</title>
<meta name="viewport" content="width=device-width, initial-scale=1.0, maximum-
scale=1.0, user-scalable=no" />
<meta content="" name="description" />
<meta content="" name="author" />
    <link href="../assets/plugins/jquery-metrojs/MetroJs.min.css" rel="stylesheet"
type="text/css" />
<link rel="stylesheet" type="text/css" href="../assets/plugins/shape-hover/css/demo.css"
/>
<link rel="stylesheet" type="text/css" href="../assets/plugins/shape-
hover/css/component.css" />
<link rel="stylesheet" type="text/css" href="../assets/plugins/owl-
carousel/owl.carousel.css" />
<link rel="stylesheet" type="text/css" href="../assets/plugins/owl-carousel/owl.theme.css"
/>
<link href="../assets/plugins/pace/pace-theme-flash.css" rel="stylesheet" type="text/css"
media="screen"/>
<link href="../assets/plugins/jquery-slider/css/jquery.sidr.light.css" rel="stylesheet"
type="text/css" media="screen"/>
<link rel="stylesheet" href="../assets/plugins/jquery-ricksaw-chart/css/rickshaw.css"
type="text/css" media="screen" >
<link rel="stylesheet" href="../assets/plugins/Mapplic/mapplic/mapplic.css"
type="text/css" media="screen" >
```



```

<link href="../assets/plugins/bootstrapv3/css/bootstrap.min.css" rel="stylesheet"
type="text/css"/>
<link href="../assets/plugins/bootstrapv3/css/bootstrap-theme.min.css" rel="stylesheet"
type="text/css"/>
<link href="../assets/plugins/font-awesome/css/font-awesome.css" rel="stylesheet"
type="text/css"/>
<link href="../assets/css/animate.min.css" rel="stylesheet" type="text/css"/>
<link href="../assets/plugins/jquery-scrollbar/jquery.scrollbar.css" rel="stylesheet"
type="text/css"/>
<link href="../assets/css/style.css" rel="stylesheet" type="text/css"/>
<link href="../assets/css/responsive.css" rel="stylesheet" type="text/css"/>
<link href="../assets/css/custom-icon-set.css" rel="stylesheet" type="text/css"/>
<link href="../assets/css/magic_space.css" rel="stylesheet" type="text/css"/>
    <link href="../css/bootstrap.min.css" rel="stylesheet">
    <link href="../css/plugins/morris.css" rel="stylesheet">

    <script type="text/javascript"
src="http://ajax.googleapis.com/ajax/libs/jquery/1.8.2/jquery.min.js"></script>
</head>
<body class="">
<?php include("header.php");?>
<div class="page-container row">

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

    <div class="clearfix"></div>
    <!-- END SIDEBAR MENU -->
</div>
</div>
<!-- BEGIN PAGE CONTAINER-->
<div class="page-content">
    <!-- BEGIN SAMPLE PORTLET CONFIGURATION MODAL FORM-->
    <div id="portlet-config" class="modal hide">
        <div class="modal-header">
            <button data-dismiss="modal" class="close" type="button"></button>
            <h3>Widget Settings</h3>
        </div>
        <div class="modal-body"> Widget settings form goes here </div>
        </div>
    <div class="clearfix"></div>

```

```

        <div class="content sm-gutter">
            <div class="page-title">
                </div>
            <!-- BEGIN DASHBOARD TILES -->
                <div class="row">
                    <div class="col-md-3 col-vlg-3 col-sm-6">
                        <div class="tiles green m-b-10">
                            <div class="tiles-body">
                                <div class="controller"> <a href="javascript:;" class="reload"></a> <a
                                    href="javascript:;" class="remove"></a> </div>
                                <div class="tiles-title text-black">Overall Visitors </div>
                                <div class="widget-stats">
                                    <div class="wrapper transparent">
                                        <?php $ov=mysqli_query($con,"select * from usercheck");
                                            $num=mysqli_num_rows($ov);
                                                ?>
                                        <span class="item-title">Overall Visitors</span> <span class="item-count animate-
                                            number semi-bold" data-value="<?php echo $num;?>" data-animation-
                                                duration="700">0</span>
                                            </div>
                                        </div>
                                    </div>
                                <div class="widget-stats ">
                                    <div class="wrapper last">
                                        <?php
                                            $tdate=date("Y/m/d");

                                            $tv1=mysqli_query($con,"select * from usercheck where logindate='$tdate'");
                                                $num11=mysqli_num_rows($tv1);
                                                    ?>

                                        <span class="item-title">Today</span> <span class="item-count animate-number semi-
                                            bold" data-value="<?php echo $num11;?>" data-animation-duration="700">0</span>
                                            <?php
                                                ?>
                                            </div>
                                        </div>
                                    </div>
                                </div>
                            </div>
                        </div>
                    </div>
                </div>
            </div>
        </div>

```

</div>

</div>

</div>

</div>

```
<div class="col-md-3 col-vlg-3 col-sm-6">
  <div class="tiles blue m-b-10">
    <div class="tiles-body">
      <div class="controller"> <a href="javascript:;" class="reload"></a> <a
        href="javascript:;" class="remove"></a> </div>
      <div class="tiles-title text-black">Registered Users </div>
      <div class="widget-stats">
        <div class="wrapper transparent">
          <?php $rt=mysqli_query($con,"select * from user");
            $rw=mysqli_num_rows($rt);?>
        <span class="item-title">Registered Users</span> <span class="item-count animate-
          number semi-bold" data-value="<?php echo $rw;?>" data-    animation-
            duration="700">0</span>
        </div>
      </div>
    </div>
    <div class="widget-stats ">
      <div class="wrapper last">
        <?php
          $utd=date('Y-m-d');
          $rt1=mysqli_query($con,"select * from user where posting_date='$utd'");
          $rw1=mysqli_num_rows($rt1);?>
        <span class="item-title">Today's</span> <span class="item-count animate-number
          semi-bold" data-value="<?php echo $rw1;?>" data-animation-duration="700">0</span>
        </div>
      </div>
    </div>
  </div>
  <div class="col-md-4 col-vlg-3 col-sm-6">
    <div class="tiles purple m-b-10">
      <div class="tiles-body">
        <div class="controller"> <a href="javascript:;" class="reload"></a> <a
```

```

        href="javascript:;" class="remove"></a> </div>
<div class="tiles-title text-black">Quote Requests </div>
    <div class="widget-stats">
        <div class="wrapper transparent">
            <?php
                $qr=mysqli_query($con,"select * from prequest");
                $oq=mysqli_num_rows($qr);
            ?>
            <span class="item-title">Overall Quotes</span> <span class="item-count animate-
            number semi-bold" data-value="<?php echo $oq?>" data-animation-
            duration="700">0</span>
        </div>
    </div>

    <div class="widget-stats">
        <div class="wrapper transparent">
            <?php
                $qr1=mysqli_query($con,"select * from prequest where status='0'");
                $oq1=mysqli_num_rows($qr1);
            ?>
            <span class="item-title">New Quotes's</span> <span class="item-count animate-number
            semi-bold" data-value="<?php echo $oq1;?>" data-animation-duration="700">0</span>
        </div>
    </div>

    <div class="widget-stats ">
        <div class="wrapper last">
            <?php
                $qr2=mysqli_query($con,"select * from prequest where status='1'");
                $oq2=mysqli_num_rows($qr2);
            ?>
            <span class="item-title">In Progress</span> <span class="item-count animate-number
            semi-bold" data-value="<?php echo $oq2;?>" data-animation-duration="700">0</span>
        </div>
    </div>
</div>
</div>
</div>

```

```

        <div class="row">
            <div class="col-md-3 col-vlg-3 col-sm-6">
                <div class="tiles red m-b-10">
                    <div class="tiles-body">
                        <div class="controller"> <a href="javascript:;" class="reload"></a> <a
                            href="javascript:;" class="remove"></a> </div>
                        <div class="tiles-title text-black">Overall Tickets </div>
                        <div class="widget-stats">
                            <div class="wrapper transparent">
                                <?php $vt=mysqli_query($con,"select * from ticket");
                                    $ovt=mysqli_num_rows($vt);
                                    ?>
                                <span class="item-title">All Tickets</span> <span class="item-count animate-number
                                    semi-bold" data-value="<?php echo $ovt;?>" data-animation-duration="700">0</span>
                                </div>
                            </div>
                        </div>
                    </div>
                </div>
            </div>
        </div>

```

```

        <div class="widget-stats ">
            <div class="wrapper last">
                <?php
                    $tdate=date("Y/m/d");

                    $otv=mysqli_query($con,"select * from ticket where status='Open'");
                    $otv1=mysqli_num_rows($otv);
                    ?>

```

```

        <span class="item-title">Pending Tickets</span> <span class="item-count animate-
            number semi-bold" data-value="<?php echo $otv1;?>" data-animation-
            duration="700">0</span> <?php
            ?>
        </div>
    </div>

```

</div>

</div>

</div>

</div>

</div>

<!-- END DASHBOARD TILES -->

<!-- START DASHBOARD CHART -->

<div class="col-lg-12" style="min-height:280px;">

<div class="panel panel-red">

<div class="panel-heading">

<h3 class="panel-title"><i class="fa fa-long-arrow-right"></i> All User Visit </h3>

<script type="text/javascript">

var visitorsCount = [];

var myCat = [];

</script>

<?php

\$totaldays = cal\_days\_in\_month(CAL\_GREGORIAN, date("m"), date("Y"));

\$month\_array=array();

for(\$i=1; \$i<=\$totaldays; \$i++)

{

if(!array\_key\_exists(\$i,\$month\_array))

{

\$key = ";

if(\$i<10)

{

\$key = '0'.\$i;

\$month\_array[\$key] = 0;

}

else

{

\$month\_array[\$i]

= 0;

```
    }
    ?>
    <script
var myKey = "Day " + '<?php echo $i; ?>';

    myCat.push(myKey);
    </script>
type="text/javascript">        <?php

    }

    }
    //print_r($month_array);
$results = mysqli_query($con,"SELECT logindate FROM usercheck");
    //$f2=mysqli_num_rows($a2);

    if(mysqli_num_rows($results) >0)
    {
    while($row = mysqli_fetch_row($results))
    {
        $user_date = $row[0];
        $dateArray = explode('/', $user_date);
        $year = $dateArray[0];
        $monthName = date("M", mktime(0, 0, 0, $dateArray[1], 10));
        $currentMonth = date('m',mktime(0, 0, 0, $dateArray[1], 10));
        // echo $monthName;
        //$month = date("M", strtotime($user_date));
        //echo $month;

        //echo $month.'<br/>';

        //$day = date('d',$dateArray));

        if($year == date("Y") && $currentMonth == date("m"))
        {

            if(array_key_exists($dateArray[2],$month_array))
```

```

        {
$month_array[$dateArray[2]] = $month_array[$dateArray[2]] + 1;
        }
    }
}
}

//print_r($month_array);
foreach($month_array as $key=>$value)
{
    ?>
    <script type="text/javascript">
visitorsCount.push(<?php echo $value;?>);
    </script>
    <?php
    }
    ?>

```

```

<script type="text/javascript">
    var d = new Date();
    var month = new Array();
    month[0] = "January";
    month[1] = "February";
    month[2] = "March";
    month[3] = "April";
    month[4] = "May";
    month[5] = "June";
    month[6] = "July";
    month[7] = "August";
    month[8] = "September";
    month[9] = "October";
    month[10] = "November";
    month[11] = "December";
    var n = month[d.getMonth()];
    $(function () {
        $('#container').highcharts({
            title: {
text: 'Daily Visitors Chart of ' + n,

```



```

        x: -20 //center
        },
        subtitle: {
            text: "",
            x: -20
        },
        xAxis: {
            categories: myCat
        },
        yAxis: {
            min: 0,
            title: {
text: 'Visitors Count'
            },
            plotLines: [{
                value: 0,
                width: 1,
                color: '#808080'
            }]
        },
        tooltip: {
            valueSuffix: ' Users'
        },
        legend: {
            layout: 'vertical',
            align: 'right',
verticalAlign: 'middle',
            borderWidth: 0
        },
        series: [{
            name: 'visitorsCount',
            data: visitorsCount
        }]
    });
    });
</script>

```

```

<div id="container" style="min-width: 310px; height: 400px; margin: 0 auto"></div>

```

```

</div>

```

```

<div class="panel-body">

```

```

        <div id="morris-line-chart"></div>
        <div class="text-right">
<a href="#">View Details <i class="fa fa-arrow-circle-right"></i></a>
        </div>
        </div>
        </div>
        </div>

<!-- END DASHBOARD CHART -->

</div></div>
<!-- BEGIN CHAT -->

</div>
<script src="../../assets/plugins/jquery-1.8.3.min.js" type="text/javascript"></script>
<script src="../../assets/plugins/jquery-ui/jquery-ui-1.10.1.custom.min.js"
        type="text/javascript"></script>
<script src="../../assets/plugins/bootstrap3/js/bootstrap.min.js"
        type="text/javascript"></script>
<script src="../../assets/plugins/breakpoints.js" type="text/javascript"></script>
<script src="../../assets/plugins/jquery-unveil/jquery.unveil.min.js"
        type="text/javascript"></script>
<script src="../../assets/plugins/jquery-block-ui/jqueryblockui.js"
        type="text/javascript"></script>
<script src="../../assets/plugins/jquery-lazyload/jquery.lazyload.min.js"
        type="text/javascript"></script>
<script src="../../assets/plugins/jquery-scrollbar/jquery.scrollbar.min.js"
        type="text/javascript"></script>
<!-- END CORE JS FRAMEWORK -->
<!-- BEGIN PAGE LEVEL JS -->
<script src="../../assets/plugins/jquery-slider/jquery.sidr.min.js"
        type="text/javascript"></script>
<script src="../../assets/plugins/jquery-slimscroll/jquery.slimscroll.min.js"
        type="text/javascript"></script>
<script src="../../assets/plugins/pace/pace.min.js" type="text/javascript"></script>
<script src="../../assets/plugins/jquery-numberAnimate/jquery.animateNumbers.js"

```

```

        type="text/javascript"></script>
<script src="../../assets/plugins/jquery-ricksaw-chart/js/raphael-min.js"></script>
    <script src="../../assets/plugins/jquery-ricksaw-chart/js/d3.v2.js"></script>
<script src="../../assets/plugins/jquery-ricksaw-chart/js/rickshaw.min.js"></script>
    <script src="../../assets/plugins/jquery-sparkline/jquery-sparkline.js"></script>
        <script src="../../assets/plugins/skycons/skycons.js"></script>
    <script src="../../assets/plugins/owl-carousel/owl.carousel.min.js"
        type="text/javascript"></script>
    <script type="../../text/javascript" src="
        "></script>
<script src="../../assets/plugins/jquery-gmap/gmaps.js" type="text/javascript"></script>
<script src="../../assets/plugins/Mapplic/js/jquery.easing.js" type="text/javascript"></script>
    <script src="../../assets/plugins/Mapplic/js/jquery.mousewheel.js"
        type="text/javascript"></script>
<script src="../../assets/plugins/Mapplic/js/hammer.js" type="text/javascript"></script>
    <script src="../../assets/plugins/Mapplic/mapplic/mapplic.js"
        type="text/javascript"></script>
<script src="../../assets/plugins/jquery-flot/jquery.flot.js" type="text/javascript"></script>
    <script src="../../assets/plugins/jquery-flot/jquery.flot.resize.min.js"
        type="text/javascript"></script>
<script src="../../assets/plugins/jquery-metrojs/MetroJs.min.js" type="text/javascript"
    ></script>
    <script src="../../assets/js/core.js" type="text/javascript"></script>
    <script src="../../assets/js/chat.js" type="text/javascript"></script>
    <script src="../../assets/js/demo.js" type="text/javascript"></script>
    <script src="../../assets/js/dashboard_v2.js" type="text/javascript"></script>
    <script type="text/javascript" src="js/highcharts.js"></script>
    <script type="text/javascript" src="js/exporting.js"></script>
    <script type="text/javascript">
$(document).ready(function () {
    $(".live-tile,.flip-list").liveTile();
});
</script>
</body>
</html>

```

## # SOURCE CODE FOR SQL

```
-- phpMyAdmin SQL Dump
-- version 4.9.0.1
-- https://
--
-- Host: 127.0.0.1
-- Generation Time: Apr 22, 2021 at 06:34 PM
-- Server version: 10.3.15-MariaDB
-- PHP Version: 7.2.19

SET SQL_MODE = "NO_AUTO_VALUE_ON_ZERO";
SET AUTOCOMMIT = 0;
START TRANSACTION;
SET time_zone = "+00:00";

/*!40101 SET
 @OLD_CHARACTER_SET_CLIENT=@ @CHARACTER_SET_CLI
 ENT */;
/*!40101 SET
 @OLD_CHARACTER_SET_RESULTS=@ @CHARACTER_SET_R
 ESULTS */;
/*!40101 SET
 @OLD_COLLATION_CONNECTION=@ @COLLATION_CONNE
 CTION */;
/*!40101 SET NAMES utf8mb4 */;

--
-- Database: `crm`
--

--
-- Table structure for table `admin`
--

CREATE TABLE `admin` (
  `id` int(11) NOT NULL,
```

```
`name` varchar(255) NOT NULL DEFAULT "",  
`password` varchar(255) NOT NULL DEFAULT ""  
) ENGINE=MyISAM DEFAULT CHARSET=latin1;
```

```
--  
-- Dumping data for table `admin`  
--
```

```
INSERT INTO `admin` (`id`, `name`, `password`) VALUES  
(1, 'admin', 'admin');
```

```
-----
```

```
--  
-- Table structure for table `prequest`  
--
```

```
CREATE TABLE `prequest` (  
  `id` int(11) NOT NULL,  
  `name` varchar(255) DEFAULT NULL,  
  `email` varchar(255) DEFAULT NULL,  
  `contactno` varchar(11) DEFAULT NULL,  
  `company` varchar(255) DEFAULT NULL,  
  `wdd` varchar(255) DEFAULT NULL,  
  `cms` varchar(255) DEFAULT NULL,  
  `seo` varchar(255) DEFAULT NULL,  
  `smo` varchar(255) DEFAULT NULL,  
  `swd` varchar(255) DEFAULT NULL,  
  `dwd` varchar(255) DEFAULT NULL,  
  `fwd` varchar(255) DEFAULT NULL,  
  `dr` varchar(255) DEFAULT NULL,  
  `whs` varchar(255) DEFAULT NULL,  
  `wm` varchar(255) DEFAULT NULL,  
  `ed` varchar(255) DEFAULT NULL,  
  `wta` varchar(255) DEFAULT NULL,  
  `opi` varchar(255) DEFAULT NULL,  
  `ld` varchar(255) DEFAULT NULL,  
  `da` varchar(255) DEFAULT NULL,  
  `osc` varchar(255) DEFAULT NULL,  
  `nd` varchar(255) DEFAULT NULL,
```

```

`others` varchar(255) DEFAULT NULL,
`query` longtext DEFAULT NULL,
`posting_date` date DEFAULT NULL,
`remark` longtext DEFAULT NULL,
`status` int(11) DEFAULT NULL
) ENGINE=InnoDB DEFAULT CHARSET=latin1;

```

--

-- Dumping data for table `prequest`

--

```

INSERT INTO `prequest` (`id`,`name`,`email`,`contactno`,
`company`,`wdd`,`cms`,`seo`,`smo`,`swd`,`dwd`,`fwd`,`dr`,
`whs`,`wm`,`ed`,`wta`,`opi`,`ld`,`da`,`osc`,`nd`,`others`,`query`,
`posting_date`,`remark`,`status`) VALUES
(1, 'Anuj Kumar', 'phpgurukulteam@gmail.com', '1234567890', 'Test',
'Website Design & Development', '', '', '', 'Dynamic Website Design',
'', 'Web Hosting Services', '', 'Ecommerce Development', 'Walk
Through Animation', '', '', '', 'This is for testing', '2021-04-22',
'This is for test', NULL);

```

-----

--

-- Table structure for table `ticket`

--

```

CREATE TABLE `ticket` (
`id` int(11) NOT NULL,
`ticket_id` varchar(11) DEFAULT NULL,
`email_id` varchar(300) DEFAULT NULL,
`subject` varchar(300) DEFAULT NULL,
`task_type` varchar(300) DEFAULT NULL,
`prioprity` varchar(300) DEFAULT NULL,
`ticket` longtext DEFAULT NULL,
`attachment` varchar(300) DEFAULT NULL,
`status` varchar(300) DEFAULT NULL,
`admin_remark` longtext DEFAULT NULL,
`posting_date` date DEFAULT NULL,

```

```

`admin_remark_date` timestamp NULL DEFAULT
current_timestamp() ON UPDATE current_timestamp()
) ENGINE=InnoDB DEFAULT CHARSET=latin1;

--
-- Dumping data for table `ticket`
--

INSERT INTO `ticket` (`id`, `ticket_id`, `email_id`, `subject`,
`task_type`, `priority`, `ticket`, `attachment`, `status`,
`admin_remark`, `posting_date`, `admin_remark_date`) VALUES
(12, '5', 'phpgurukulteam@gmail.com', 'Test Ticket', 'billing', 'important',
'This ticket for testing purpose.', '', 'closed', 'Ticket resolved. Solution
provided', '2021-04-22', '2021-04-21 18:30:00');

```

-----

```

--
-- Table structure for table `user`
--

```

```

CREATE TABLE `user` (
`id` int(11) NOT NULL,
`name` varchar(255) DEFAULT NULL,
`email` varchar(255) DEFAULT NULL,
`alt_email` varchar(255) DEFAULT NULL,
`password` varchar(255) DEFAULT NULL,
`mobile` varchar(255) DEFAULT NULL,
`gender` varchar(255) DEFAULT NULL,
`address` varchar(500) DEFAULT NULL,
`status` int(11) DEFAULT NULL,
`posting_date` timestamp NULL DEFAULT current_timestamp()
) ENGINE=MyISAM DEFAULT CHARSET=latin1;

```

```

--
-- Dumping data for table `user`
--

```

```

INSERT INTO `user` (`id`, `name`, `email`, `alt_email`, `password`,
`mobile`, `gender`, `address`, `status`, `posting_date`) VALUES

```

```

-----

--
-- Table structure for table `usercheck`
--

CREATE TABLE `usercheck` (
  `id` int(11) NOT NULL,
  `logindate` varchar(255) DEFAULT "",
  `logintime` varchar(255) DEFAULT "",
  `user_id` int(11) DEFAULT NULL,
  `username` varchar(255) DEFAULT NULL,
  `email` varchar(255) DEFAULT "",
  `ip` varbinary(16) DEFAULT NULL,
  `mac` varbinary(16) DEFAULT NULL,
  `city` varchar(255) DEFAULT NULL,
  `country` varchar(255) DEFAULT NULL
) ENGINE=MyISAM DEFAULT CHARSET=latin1;

--
-- Dumping data for table `usercheck`
--

INSERT INTO `usercheck` (`id`, `logindate`, `logintime`, `user_id`,
  `username`, `email`, `ip`, `mac`, `city`, `country`) VALUES
(1, '2021/04/22', '05:59:18pm', 1, 'Anuj Kumar',
  'phpgurukulteam@gmail.com', 0x3a3a31,
  0x31322d46342d38442d31322d39392d39, "", ""),
(2, '2021/04/22', '10:00:15pm', 1, 'Anuj Kumar',
  'phpgurukulteam@gmail.com', 0x3a3a31,
  0x31322d46342d38442d31322d39392d39, "", "");

--
-- Indexes for dumped tables
--

--
-- Indexes for table `admin`
--

```



```

ALTER TABLE `admin`
  ADD PRIMARY KEY (`id`);

--
-- Indexes for table `prequest`
--
ALTER TABLE `prequest`
  ADD PRIMARY KEY (`id`);

--
-- Indexes for table `ticket`
--
ALTER TABLE `ticket`
  ADD PRIMARY KEY (`id`);

--
-- Indexes for table `user`
--
ALTER TABLE `user`
  ADD PRIMARY KEY (`id`);

--
-- Indexes for table `usercheck`
--
ALTER TABLE `usercheck`
  ADD PRIMARY KEY (`id`);

--
-- AUTO_INCREMENT for dumped tables
--

--
-- AUTO_INCREMENT for table `admin`
--
ALTER TABLE `admin`
  MODIFY `id` int(11) NOT NULL AUTO_INCREMENT,
  AUTO_INCREMENT=2;

--
-- AUTO_INCREMENT for table `prequest`

```

```

--
ALTER TABLE `prequest`
  MODIFY `id` int(11) NOT NULL AUTO_INCREMENT,
  AUTO_INCREMENT=2;

--
-- AUTO_INCREMENT for table `ticket`
--
ALTER TABLE `ticket`
  MODIFY `id` int(11) NOT NULL AUTO_INCREMENT,
  AUTO_INCREMENT=13;

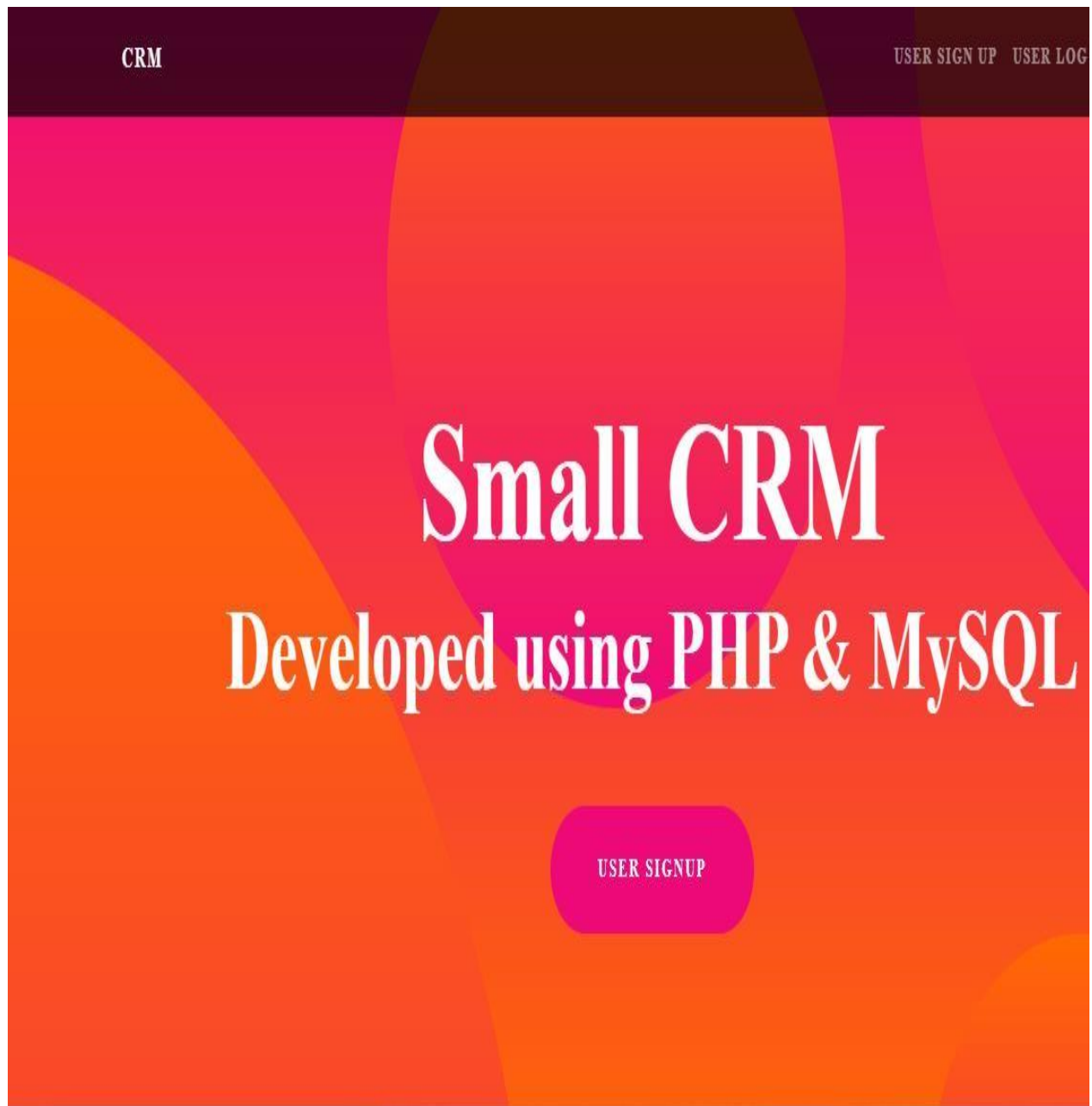
--
-- AUTO_INCREMENT for table `user`
--
ALTER TABLE `user`
  MODIFY `id` int(11) NOT NULL AUTO_INCREMENT,
  AUTO_INCREMENT=2;

--
-- AUTO_INCREMENT for table `usercheck`
--
ALTER TABLE `usercheck`
  MODIFY `id` int(11) NOT NULL AUTO_INCREMENT,
  AUTO_INCREMENT=3;
COMMIT;

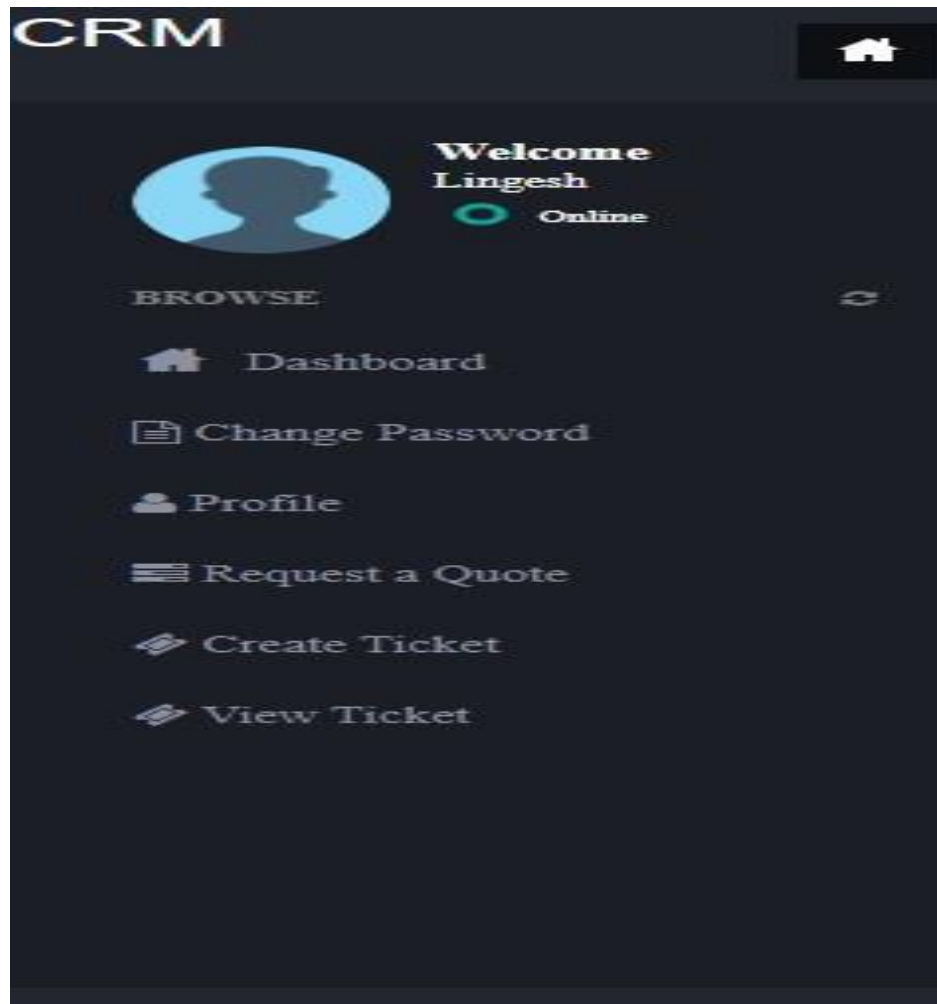
/*!40101 SET
  CHARACTER_SET_CLIENT=@OLD_CHARACTER_SET_CLIEN
  T */;
/*!40101 SET
  CHARACTER_SET_RESULTS=@OLD_CHARACTER_SET_RESU
  LTS */;
/*!40101 SET
  COLLATION_CONNECTION=@OLD_COLLATION_CONNECTI
  ON */;

```

**SNAPSHOTS OF PROJECT OUTPUT:**



## HOME PAGE OF WEBSITE



## Create ticket

Subject



Task Type

Select your Task Type



Priority

Choose your Priority



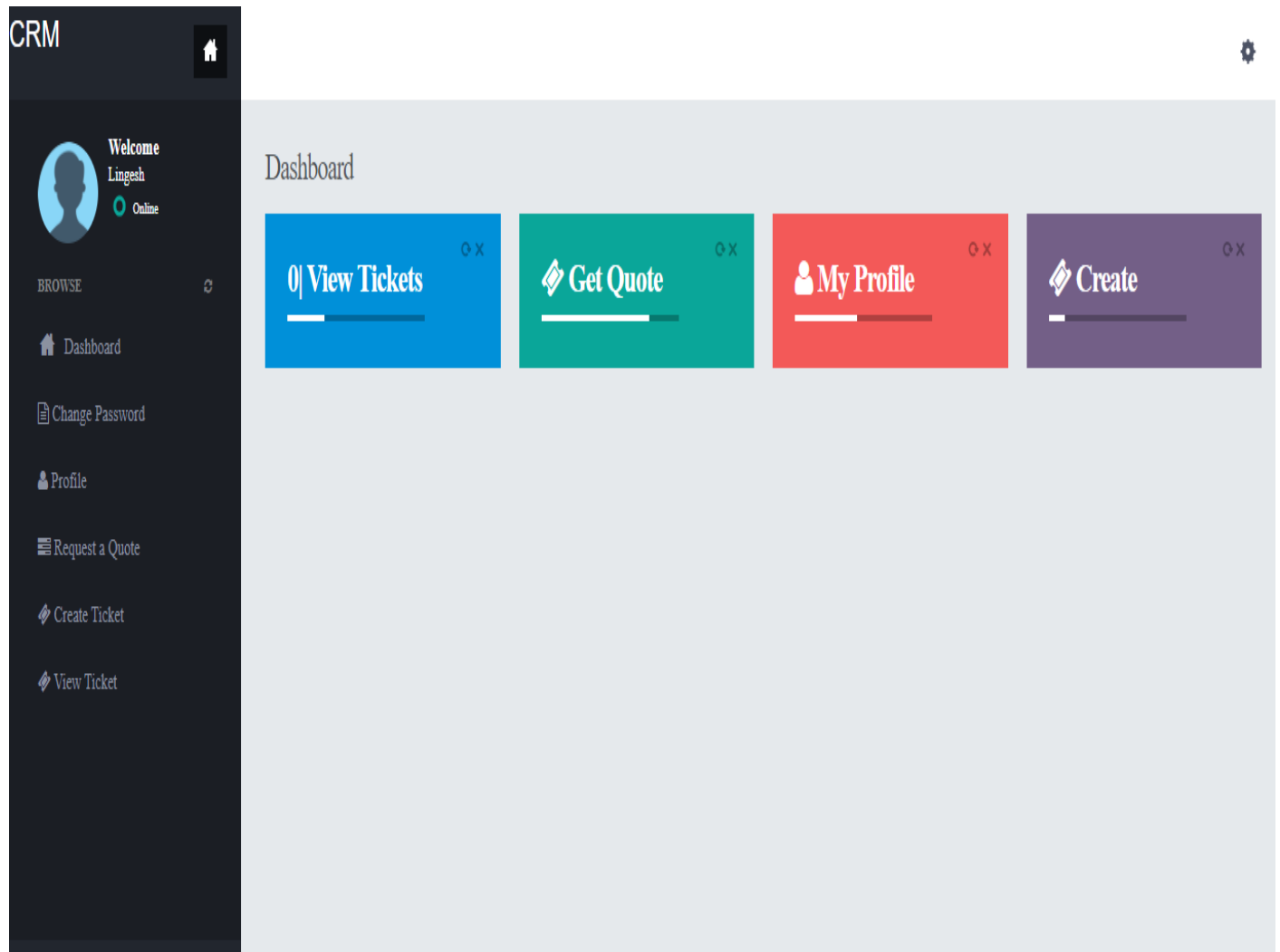
Description



Clear Form

Send

## DASHBOARD OF WEBSITE



## Request a Quote

Please click below mention services of your interest to receive quotation for the same:

Name  Linges

Email  lingesm22@gmail.com

Contact no  9080643206

Company 

Service Required : ☒ Website Design & Development

☐ CMS (Content Management System)

☐ SEO (Search Engine Optimization)

☐ SMO (Social Media Optimization)

☐ Static Website Design

☐ Dynamic Website Design

☐ Flash Website Development

☐ Domain Registration

☐ Web Hosting Services

☐ Website Maintenance

☐ Ecommerce Development

☐ Walk Through Animation

☐ Online Payment Integration

☐ Logo Design

## CREATING A TICKET

|   |  |
|---|--|
| <input type="checkbox"/> Static Website Design      | <input type="checkbox"/> Dynamic Website Design    |
| <input type="checkbox"/> Flash Website Development  | <input type="checkbox"/> Domain Registration       |
| <input type="checkbox"/> Web Hosting Services       | <input type="checkbox"/> Website Maintenance       |
| <input type="checkbox"/> Ecommerce Development      | <input type="checkbox"/> Walk Through Animation    |
| <input type="checkbox"/> Online Payment Integration | <input type="checkbox"/> Logo Design               |
| <input type="checkbox"/> Dash board Application     | <input type="checkbox"/> Open Source Customization |
| <input type="checkbox"/> NewsLetter Design          | <input type="checkbox"/> Others                    |

Query

Clear Form

Submit



## REFERENCES

### Websites

- [http://en.wikipedia.org/wiki/Customer\\_relationship\\_management](http://en.wikipedia.org/wiki/Customer_relationship_management)
- <http://www.zeelabs.com/CRM.html>
- <http://www.crmnewz.com/>
- <http://www.microsoft.com/crm>
- <http://www.crystaldecisions.com/>
- [www.crmsolution.com](http://www.crmsolution.com)

## **BOOKS**

- “Analysis And Design Of Information Systems” By V. Rajaraman
- “Principles Of Information Systems” By Ralph M.Stair And George W. Reylond
- “MFC Programming (Unleased)” By SAMS Publication
- “Programming of PHP By Kruglinski, Wingo
- “The Complete Reference OF CRM By TMH George
- “Crystal Report Developer Guide” By Businness.com

