CHAPTER 1 :

Synopsis

SYNOPSIS

**1.1Title of the Project:**

Online Vehicle Showroom(OVS)

**1.2Objective of the project:**

This project is aimed at developing a Web application that depicts Online Vehicle Showroom and booking vehicles through online. Customer can register to this site and he/she can book

vehicles by entering his login information. Administrator is main user of this system and he can add employees, and new vehicle details.

**1.3Project Category:**

RDBMS (Relational Database Management System)

**1.4Software Specifications:**

* Technology Implemented : Apache Server
* Language Used : PHP 5.3
* Database : My SQL 5.5
* User Interface Design : HTML, AJAX, Javascript
* Web Browser : Mozilla, Google Chrome,

Internet explorer

**1.5Hardware Requirements:**

* Processor :Pentium, AMD or HigherVersion.
* Operating System :Windows XP/ Windows 7/ Linux
* RAM :256 MB, 2GB recommended
* Hardware Devices : Keyboard with mouse
* Hard disk : 10GB or More
* Display : Standard Output Display

**1.6Modules:**

* Main and Login Page:
  + The main page, neatly designed page, here login option is provided to login. Also new signup option is provided to for new user to signup.
* Vehicles page:
  + This screen will display all the vehicle details, with exact cost, Features, Model details, etc.
* Vehicle Selection Screen:
  + This search screen Option must be provide to select the vehicles based on name. In this option customer can select vehicles based on its model.
* Booking page:
  + In this page user can book vehicles by entering cart information. After booking vehicles user will receive booked vehicles information via mail.
  + This page also includes the details about the payment must be done by the customer during the time of delivery.
* Admin Page:
  + Administrator has full permission to access this web site. Here administrator can add new vehicle details, and he/she can add his employees.
* Dealer Page:
  + Dealers can handle customer details, and he/she can communicate with customers through mail. If customer forgets his password dealer can reset his/her password.

**1.7Future Scope of the Project:**

The present system is developing as web application. In future we would like to develop it for portable devices like android cell phones or iphone cell phones.

Customer will get alert SMS’s when new vehicle model release, Booking date, Delivery date,etc.

CHAPTER 2:

Software

Requirement

Specification

SOFTWAREREQUIREMENTSPECIFICATION

# 

# 2.1 Introduction

The purpose of this SRS document is to provide a detailed overview of our software product, its parameters and goals. This project document describes the project's target audience and its user interface, hardware and software requirements.

The aim of this document is to gather and analyze and give an in-depth insight of the complete Online Vehicle Showroom by defining the problem statement in detail. The detailed requirements of the Online Vehicle Showroom are provided in this document.

## 

## 2.1.1 Purpose:

The purpose of the document is to collect and analyze all assorted ideas that have come up to define the system, its requirements with respect to consumers.The main purpose of this Vehicle showroom is to develop a web based interface for Vehicle showroom companies.

The main purpose of this Online Vehicle Showroom is that it provides provision to customers to buy or book vehicles through online. The current system is offline system, in this to purchase vehicle the customer should visit to showroom. So this current system is very difficult because its time consuming. So our project aims at creating an web application which tracks Customer records, Online booking, Online vehicle records, etc and it provides easy to use web based interface for customers where customers can search for vehicles, view a complete details, models, features, pricing of the vehicles and book the vehicles.

## 2.1.2 Scope:

The name of the project is “ONLINE VEHICLE SHOWROOM”. The main scope of this web application is that depicts online Vehicle showroom and booking vehicles through online. Customer can register to this site and he/she can book vehicles by entering his/her login information. Administrator is main user of thissystem and he/she can add employees, and new vehicle details.

* The current system can be extended to allow the customers to register accounts and save favourite vehicles in to wish list.
* The design of the web application involves the listing the vehicles, search for vehicles, display the complete details of vehicles, etc.
* It provides updated information about the vehicles of all the companies.
* Customer can view Purchase details and billing records any time.

**2.1.3 Definitions, Acronyms, and Abbreviations:**

|  |  |
| --- | --- |
| SRS | Software requirement specification |
| OVS | Online Vehicle Showroom |
| PHP | PHP: Hypertext pre-processor.  PHP is a server scripting language, and is a powerful tool for making dynamic and interactive Web pages quickly. PHP is a widely-used, free, and efficient alternative to competitors such as Microsoft's ASP. |
| MySQL | MySQL Database server.  MySQL is a most popular database system used on the web. It supports standard SQL Queries. The data in MySQL is stored in tables. A table is a collection of related data, and it consists of columns and rows. |

## References:

Book references:

* Software engineering by PankajJalote
* Learning PHP, MySQL, JavaScript, and CSS: A Step-by-Step Guide to Creating Dynamic Websites by Robin Nixon

Internet references:

* www.w3schools.com
* www.php.net
* www.mysql.com

## 2.2.1 Overview:

The following subsections provide complete overview of the SRS documentation for the product “Online Vehicle Showroom”. The entire SRS is documented in the view of customers, dealers and admin and the following subsections are arranged to complete outlook of the software, its perspective, features, System requirements.

# 2.2.2 Overall Description:

User Friendliness is provided in the web application with various controls provided by system Rich User Interface. This project which helps customers to buy or book vehicles through online. It keeps track of Customers records, Vehicle records, Payment and billing records, dealer records, etc. The system generates Invoice and bill after purchasing or booking of new vehicle. This is web application and it’s developing using PHP language. All the records stores in MySQL Database.

**2.3. Product Perspective:**

This totally self contained and works efficiently. It provides simple database rather then complex ones for high requirements and it provides good and easy graphical user interface new, naïve as well as experienced users of the computers.

**2.4. Product Function:**

The product functions will include the following areas:

1. Admin logs in into the application user name & password.
2. After successful login the main page appears.
3. The application is capable enough to store data and also perform some editng on them that is added.It will be having user friendly GUI’s that will guide the user to easily achive the same. This program is a menu driven program. When we click the main menu the different forms will be enabled.

**2.5.Functional Requirements**

**2.5.1 Modules:**

The program consists of the following modules:

* Main and Login module:
  + The main page, neatly designed page, here login option is provided to login. Also new signup option is provided for new user to signup.
* Vehicles module:
  + This screen will display all the vehicle details, with exact cost, Features, Model details, etc.
* Vehicle Selection module:
  + This search screen Option must be provide to select the vehicles based on name. In this option customer can select vehicles based on its model or name.
* Booking module:
  + In this page user can book vehicles by entering cart information. After booking vehicles user will receive booked vehicles information via mail.
  + This page also includes the details about the payment must be done by the customer during the time of delivery.
* Admin module:
  + Administrator has full permission to access this web site. Here administrator can add new vehicle details, and he/she can add his/herdealers.
* Dealer module:
  + Dealers can handle customer details, and he can communicate with customers through mail. If customer forgets his password dealer can reset his password.

### **2.6. System requirements**

### **2.6.1 User Interfaces:**

The user interface for the software shall be compatible to any browser such as Internet Explorer, Google chrome, or Mozilla Firefox.

## 

## Design Constraints:

* There are no memory requirements.
* The computers must be equipped with web browsers such as Internet explorer, Google chrome, Mozilla firefox , etc.
* A general knowledge of basic computer skills is required to use the product.

## Performance:

1. Good working computer with all requirements.
2. The product shall be based on web and has to be run from a web server.
3. The product shall take initial load time depending on internet connection strength.

### 

### **Data Storage:**

1. The customer’s web browser shall never display a customer’s password.
2. The customer’s web browser shall never display a customer’s credit card number after retrieving from the database.
3. The system’s back-end servers shall only be accessible to authenticated administrators.
4. The system’s back-end databases has login id and password.

## On-line User Documentation and Help System Requirements:

It shall provide specific guidelines to a customers for using the Online Vehicle Showroom. To implement online user help, link and search fields shall be provided.

### **2.6.2 Software Interfaces:**

* Technology Implemented : Apache Server
* Language Used : PHP 5.3
* Database : My SQL 5.5
* User Interface Design : HTML, AJAX, Javascript
* Web Browser : Mozilla, Google Chrome,

Internet explorer

**2.6.3 Hardware Interfaces:**

* Processor :Pentium,AMD or HigherVersion.
* Operating System :Windows XP/ Windows 7/ Linux
* RAM :256 MB, 2GB recommended
* Hardware Devices : Keyboard with mouse
* Hard disk : 10GB or More
* Display : Standard Output Display

### **2.6.4 Communications Interfaces:**

The Online Vehicle showroom shall use the HTTP protocol for communication over the internet and for the intranet communication will be through TCP/IP protocol suite.

## 

## Licensing Requirements:

Not Applicable

CHAPTER 3:

System Design

systemdesign

**3.1 Introduction**

The purpose of the Design Phase is a solution of the problem specified in the requirements document. This is the first step in moving from the problem domain to the solution domain. The design of a system is perhaps the most critical factor affecting the quality of the software; it has a major impact on the later phases is the **Design Document.** This Document is similar to a blueprint or a plan for the solution and is used later during implementation, testing and maintenance.

The Design activity is often divided into separate phases – **System Design** and **Detailed Design.System Design is sometimes also called Top-Level Design.** This system design aims to identify and modules that should be in the system, the specifications of these modules, and how they interact with each other to produce the desired result. At the end of the System Design all the major data structures, file formats, and the major modules in the system and their specifications are decided.

**3.2 Applicable Documents :**

The document used in system design is Software Requirement Specification Document.

**3.3 Functional Decomposition :**

* Main and Login Page:
  + The main page, neatly designed page, here login option is provided to login. Also new signup option is provided to for new user to signup.
* Vehicles page:
  + This screen will display all the vehicle details, with exact cost, Features, Model details, etc.
* Vehicle Selection Screen:
  + This search screen Option must be provide to select the vehicles based on name. In this option customer can select vehicles based on its model.
* Booking page:
  + In this page user can book vehicles by entering cart information. After booking vehicles user will receive booked vehicles information via mail.
  + This page also includes the details about the payment must be done by the customer during the time of delivery.
* Admin Page:
  + Administrator has full permission to access this web site. Here administrator can add new vehicle details, and he/she can add his/herdealers.
* Dealer Page:
  + Dealer can handle customer details, and he/she can communicateWith customers through mail. If customer forgets his/her password employee can reset his password.

**3.4 DESCRIPTION OF THE PROGRAMS**

**3.4.1 Context Flow Diagram (CFD)**

A Context Flow Diagram is a top level (also known as level 0) data flow diagram. It only contains one process node (process 0) that generalizes the function of the entire system in relationship to external entities. In context diagram the entire system is treated as a single process and all its inputs, outputs, sinks and sources are identified and shown.

**CFD OF VEHICLE SHOWROOM MANAGEMENT:**

**CFD Level 0:**

Transaction report

Purchased vehicles

Customer

Admin/dealer

**dealer**

Add vehicles

Profile, order vehicles

Online Vehicle Showroom

**3.4.2 DATA FLOW DIAGRAM(DFD):**

**Symbols Used in DFD Diagram:**

|  |  |  |
| --- | --- | --- |
| **NOTATION** | **COMPONENT** | **DESCRIPTION** |
|  | Process | An oval represents a process or transform that is applied to data or control and changes it in some way. |
|  | External Entity | A rectangle is used to represent an external entity, that is, a system element that produces information for transformation by the software or receives information produced by the software. |
|  | Data Flows | An arrow represents one or more data items or data objects. |
|  | Data Store | The open box represents datastore-stored information that is used by the software. |

A Data Flow Diagram (DFD) is a graphical representation of the "flow" of data through an Information System. A data flow diagram can also be used for the visualization of Data Processing. It is common practice for a designer to draw a context-level DFD first which shows the interaction between the system and outside entities. This context-level DFD is then "exploded" to show more detail of the system being modell.

A DFD represents flow of data through a system. Data flow diagrams are commonly used during problem analysis. It views a system as a function that transforms the input into desired output. A DFD shows movement of data through the different transformations or processes in the system.

Dataflow diagrams can be used to provide the end user with a physical idea of where the data they input ultimately has an effect upon the structure of the whole system from order to dispatch to restock how any system is developed can be determined through a dataflow diagram. The appropriate register saved in database and maintained by appropriate authorities.

**Data Flow Diagram of the Vehicle showroom management**

**Level 1:**TOP LEVEL DFD

Add

vehicles

1.0

Dealer

Dealer details

Vehicles

Uploading

images

2.0

Vehicles details

Registration

3.0

Customer details

Admin

Customer

Vehicle

Sales

4.0

Sales

Sales report

**Level 2:** ADMIN MODULE

Login

Admin

Get dealer details

1.1

Tax price

Tax type

Dealer ID

New password

Update Tax details

1.4

Username

password

dealer

Create new password

1.3

Create account

1.2

Tax details

Login details

dealer details

Tax

Account login details

Dealer

Forget password

New login details

**Level 3**:DEALER MODULE

Vehicle name,

Model type, cost

Dealer

New vehicles

Registration

2.1

Vehicle details

images

Vehicles

Upload images

2.2

Vehicle id

Vehicle id,

Vehicle name

Image title, Image path

Vehicle price

Calculate vehicle price

2.3

Images

taxid

Tax

**Level 4:**CUSTOMER MODULE

Customer

Get Customer details

3.1

Login failed

Login authentication

3.3

Customer profile

Login details

Customer

Vehicle name,

Model type, cost

Create account

3.2

Login details

Vehicle

Purchase

report

Vehicle order request

3.4

Vehicle details

Vehicle

Vehicle name total amount

Vehicle name payment details

Make Payment

3.5

Sales

**3.5.DESCRIPTION OF THE COMPONENT:**

**3.5.1 LOGIN MODULE:**

**INPUT:** The username and password

**PROCESS**: The process is named as authentication the process check the accepted username and password

**OUTPUT**: If the inputs are valid the successful login is one and a new screen with links to respective page is displayed.

**3.5.2.ADMIN MODULE**

**INPUT:** inputs dealer details to create new account for dealer

**PROCESS:** validation of data.

**OUTPUT:**The system creates new dealer profilewith login ID and password.

**3.5.3.DEALER MODULE**

**INPUT:** inputs the vehicle details such as vehicle name, image, price, etc.

**PROCESS:**Calculate vehicle price and uploading image.

**OUTPUT:**The system uploads new vehicle details to database.

**3.5.4.CUSTOMER MODULE**

**INPUT:**The user creates new account and orders new vehicle.

**PROCESS:**Vehicle ordering and payment process

**OUTPUT:**The user orders vehicle and makes payment.

CHAPTER 4:

Database Design

## DATABASEDESIGN

**4.1.Introduction**

Database is an application that manages data and allow quick storage and retrieval of the data .The first step in designing the database is to understand what the data is to be stored in the database, what application is to be built on it and what operators must frequent and subject to perform requirements .

The schema of database system is a structure description in a formal language supported by database management system(DBMS).Database schema refers to the organisation of the data, to create a blue printof how a database to be constructed.

VOS database is a data structure that stores organized information .The database contains multiple tables which includes several different fields. For example the VOS database include table for customer, dealer and login. Each of these tables would have different fields that are relevant to the information stored in the table.

**The design of OVS database tables are shown below:**

**4.1.1.TableStructure**

**4.1.1.1.Tablename :** admin

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Sl.no** | **Attribute** | **Data Type** | **Description** | **Constraints** |
| **1.** | **adminid** | int(10) | Administrator id | Primary key |
| **2.** | adminname | varchar(25) | Administrator name |  |
| **3.** | **username** | varchar(25) | User name |  |
| **4.** | password | varchar(25) | Password |  |
| **5.** | contactnumber | varchar(25) | Phone number |  |
| **6.** | createdat | date | Created time |  |
| **7.** | lastlogin | datetime | Last login time |  |
| **8.** | status | varchar(10) | status |  |

## 4.1.1.2.Tablename: customer

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Sl.no** | **Attribute** | **Data Type** | **Description** | **Constraints** |
| **1.** | **custid** | int(10) | Customerid | Primary key |
| **2.** | fname | varchar(25) | First name |  |
| **3.** | lname | varchar(25) | Last name |  |
| **4.** | contactno | varchar(25) | Phone number |  |
| **5.** | **emailid** | varchar(25) | Email id |  |
| **6.** | password | varchar(25) | Password |  |
| **7.** | address | text | Address |  |
| **8.** | city | varchar(20) | City |  |
| **9.** | state | varchar(20) | Status |  |
| **10.** | country | varchar(50) | Country |  |
| **11.** | pincode | varchar(10) | Pincode |  |
| **12.** | gender | varchar(10) | Gender |  |
| **13.** | createdat | date | Created time |  |
| **14.** | lastlogin | datetime | Last login time |  |
| **15.** | status | varchar(10) | Status |  |

## 4.1.1.3.Tablename:dealer

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Sl.no** | **Attribute** | **Data Type** | **Description** | **Constraints** |
| **1.** | **dealerid** | int(10) | Dealer id | Primary key |
| **2.** | adminid | int(10) | Administrator id | Foreign key |
| **3.** | companyname | varchar(25) | Company name |  |
| **4.** | imgid | int(10) | Image id |  |
| **5.** | fname | varchar(25) | Fisrt name |  |
| **6.** | lname | varchar(25) | Last name |  |
| **7.** | username | varchar(25) | User name |  |
| **8.** | password | varchar(25) | Password |  |
| **9.** | contactnumber | varchar(15) | Phone number |  |
| **10.** | address | text | Address |  |
| **11.** | createdat | date | Created time |  |
| **12.** | lastlogin | datetime | Last login time |  |
| **13.** | status | varchar(10) | Status |  |

## 4.1.1.3.Tablename:image

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Sl.no** | **Attribute** | **Data Type** | **Description** | **Constraints** |
| **1.** | **imgid** | int(10) | Image id | Primary key |
| **2.** | imagename | varchar(25) | Imagename |  |
| **3.** | vehid | int(10) | Vehicle id | Foreign key |
| **4.** | imagepath | varchar(50) | Image path |  |
| **5.** | defaultimg | int(1) | Default image |  |

## 

## 4.1.1.3.Tablename:sales

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Sl.no** | **Attribute** | **Data Type** | **Description** | **Constraints** |
| **1.** | **salesid** | int(10) | Sales id | Primary key |
| **2.** | vehid | int(10) | Vehicle id | Foreign key |
| **3.** | custid | int(10) | Customer id | Foreign key |
| **4.** | showroomid | int(10) | Showroom id | Foreign key |
| **5.** | vehcost | float(10,2) | Vehicle cost |  |
| **6.** | taxid | int(10) | Tax id | Foreign key |
| **7.** | ord\_date | date | Ordered date |  |
| **8.** | del\_date | date | Delivery date |  |
| **9.** | description | text | Description of vehicle |  |
| **10.** | status | varchar(10) | Status |  |

## 4.1.1.3.Tableshowroom:showroom

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Sl.no** | **Attribute** | **Data Type** | **Description** | **Constraints** |
| **1.** | **showroomid** | int(10) | Showroom id | Primary key |
| **2.** | dealerid | int(10) | Dealer id | Foreign key |
| **3.** | showroomname | varchar(50) | Showroom name |  |
| **4.** | imagepath | varchar(100) | Image path |  |
| **5.** | contactno | varchar(15) | Contact number |  |
| **6.** | address | text | Address |  |
| **7.** | city | varchar(25) | City |  |
| **8.** | state | varchar(25) | Status |  |
| **9.** | pincode | varchar(10) | Pincode |  |
| **10.** | status | varchar(10) | Status |  |

## 4.1.1.3.Tablename:tax

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Sl.no** | **Attribute** | **Data Type** | **Description** | **Constraints** |
| **1.** | **taxid** | int(10) | Tax id | Primary key |
| **2.** | taxdescription | text | Tax description |  |
| **3.** | tax | float(10,2) | Total Tax |  |
| **4.** | status | varchar(10) | Status |  |

## 4.1.1.3.TABLENAME:vehicle

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Sl.no** | **Attribute** | **Data Type** | **Description** | **Constraints** |
| **1.** | vehid | int(10) | Vehicle id | Primary key |
| **2.** | dealerid | int(10) | Dealer id | Foreign key |
| **3.** | vehname | varchar(50) | Vehicle name |  |
| **4.** | vehmodel | varchar(50) | Vehicle model |  |
| **5.** | vehtype | varchar(15) | Vehicle type |  |
| **6.** | vehdescription | text | Vehicle description |  |
| **7.** | vehcost | decimal(10,2) | Vehicle cost |  |
| **8.** | createdat | date | Create time |  |
| **9.** | status | varchar(10) | status |  |

**4.2. Entity Relationship Diagram (ER-Diagram):**

Entity relationship diagram is used in modern database software. Software engineering is to illustrate logical structure of database.iit is a relational schema database, modelling method, used to model a system and approach .this approach is commonly used in database design. The diagrams created using this method are called Entity-Relationship Diagram.

Entity-Relationship Diagram depicts the various relationships among entities, considering each objective as an entity. Entity is represented as rectangle and relationship between data objects .the ER diagram is a notation that is used to conduct the data modelling activity.

**Entity:**

Entity is the thing in which we want to store the information it is an elementary basic building block of storing information about business process. An entity represents an object defined within the information system about which you want to store the information .entities are distinct things in the enterprise.

**Relationships:**

A relationship is named connection or association between entities or used to relate two or more entities with common attributes or meaningful interaction between objects.

**Attributes:**

Attributes are the properties of the entities and the relationships are the descriptions of the entity. Attributes are elementary pieces of information attached to an entity.

|  |  |  |
| --- | --- | --- |
| **Name** | **Notation** | **Description** |
| **Entity** |  | An entity is a single object about which data can be stored it is subjected of a table. Entity and their relationships are modelled through the use of the entity relationships diagram. |
| **Relationship** |  | A relationship is named connection or association between entities or used to relate two or more entities with common attributes or meaningful interaction between objects. |
| **Attributes** |  | Attributes are the properties of the entities and the relationships are the descriptions of the entity. Attributes are elementary pieces of information attached to an entity. |



CHAPTER 5:

Detailed Design

DETAILEDDESIGN

**5.1. Introduction:**

The detailed design starts after the system design phase is complete. The main goal of detailed design is to specify the logic for the different modules that have been specified during the system design or coding for the module can be developed such that specification of the module may be given precisely. Once the module is precisely specified the internal logic for the module that will implement the given specification as decided.

The specification should be such that they are complete unambiguous and precise and they are not suggested any particular implementation. A well defined design language like PDL(Process Design Language)is used for detailed design.

In this project the front end consists of PHP forms. The form design is done using Dreamweaver.We are also using MySQL as backend tool for computerization of Online vehicle showroom.

**5.2. Applicable Documents :**

The detailed design refines the system design document hence the first applicable document here is system design. Also we are referring the data structures. Hence the second applicable document is database design. Since this project is user friendly, all users can operate it efficiently. This project helps the user to know the details of the dealers working in the showroom, and also about the customers. The admin can have the up to date information about the ordered vehicles made by customer. All work that was done by writing in papers can now be stored in database through the computer.

**5.3. Structure of software package :**

The various functional components used are:

* Functional component 1: Login module
* Functional component 2: Vehicle module
* Functional component 3: Vehicle selection module
* Functional component 4: Booking module
* Functional component 5*:* Admin module
* Functional component 6: Dealer module

**5.4 .Modular Decomposition of the System:**

**5.4.1. Login module:**

**5.4.1.1.Design assumptions:**

The module of this system are design to authenticate a user before he/she enters into the system. The user must have valid username and password registered. If the user is not valid than the user must login again with the correct username and password.

**5.4.1.2. Identification of module:**

The module identified in this sytem:

* Get username
* Get password
* Login

**5.4.1.3. Structure Chart showing hierarchy of modules:**

It is graphical representation of the control logic of processing function or modules representation of system

It is one of the most common methods used for system design. In a structure chart each program module is represented by a rectangular box. Modules at the top level of the structure chart call the modules at the lower levels. The connection between modular are represented by lines between the rectangular boxes. The connection describes the data flows between the called and calling module.

login

User info

Username

Email id

Customer

id

Customer

info

User id

User name

Online

registration

Password

**5.4.2.Vehicle module:**

**5.4.2.1.Design assumptions:**

The module of this system are used to add new vehicles and to update vehicle details and to delete vehicles.

**5.4.2.2.Identification of module:**

The module identified in this sytem:

* Add vehicle
* update
* delete

**5.4.2.3. Structure Chart showing hierarchy of modules:**

vehicles

Delete

Vehicle

record

Update

Vehicle

record

new

vehicle

id

delete

update

add

**5.4.3. Vehicle Selection module:**

**5.4.3.1.Design assumptions:**

This search screen Option must be provide to select the vehicles based on name. In this option customer can select vehicles based on its model.

**5.4.3.2. Identification of module:**

The module identified in this sytem:

* search

**5.4.3.3. Structure Chart showing hierarchy of modules:**

Home

Customer

id

search

**5.4.4.Booking module:**

**5.4.4.1.Design assumptions:**

In this page user can book vehicles by entering cart information. After booking vehicles user will receive booked vehicles information via mail.

**5.4.4.2. Identification of module:**

The module identified in this sytem:

* register
* submit
* edit

**5.4.4.3. Structure Chart showing hierarchy of modules:**

Register

Failed to

register

Customer

info

register

update

Edit

Submit

**5.4.5. Admin module:**

**5.4.5.1.Design assumptions:**

Administrator has full permission to access this web site. Here administrator can add new vehicle details, and he can add his employees.

**5.4.5.2. Identification of module:**

The module identified in this sytem:

* login
* add new dealer
* change password
* logout

**5.4.5.3. Structure Chart showing hierarchy of modules:**

Admin

New dealer details

Update new

password

Add dealer

New password

Logout

Change password

New dealer

Login

Dealer add or delete

Logs in

User

exit

add

username

exit

delete

password

**5.4.6. Dealer module:**

**5.4.6.1.Design assumptions:**

Dealers can handle customer details, and he can communicate with customers through mail. If customer forgets his password dealer can reset his password.

**5.4.6.2. Identification of module:**

The module identified in this sytem:

* login
* add new customer
* change password
* logout

**5.4.4.3. Structure Chart showing hierarchy of modules:**

Dealer

User

exit

New

password

Customer

registered

Customer

register

Uuupdate

password

login

New customer

Logout

Change password

Dealer add or delete

Logs in

username

add

delete

password

CHAPTER 6:

Testing

TESTING

**6.1 Introduction:**

Software testing is a critical element of software quality assurance and represents the ultimate review of specification, design and coding. Testing presents an interesting of a system using various test data. Preparation of the test data plays a vital role in the system testing. After preparation the test data, the system under study is tested those test data. Error were found and corrected by using the following testing steps and corrections are recorded for future references. Thus, series of testing is performed on the system before it is already for implementation.

The development of software system involves a series of production activities where opportunities for injection of human errors are enormous. Error may begin to occur at the very inception of the process where the objectives may be erroneously or imperfectly specified as well as in later design and development stages. Because of human in ability to perform and communicate with perfection, software development if followed by assurance activities.

Quantity assurance is the review of software products and related documents products of completeness, correctness, reliability and maintainability. And of course it includes assurance that the system meets the specification and the requirements for its intended use and performance. The various levels of qualities assurances are described in following sub section.

**6.2Objectives of testing:**

The objectives of testing are:

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

**6.3System Testing:**

Software testing is a critical element of software quality assurance and represents the ultimate review of specification, design and coding. The testing phase involves the testing of the system using various test data; Preparation of test data plays a vital role in system testing. After preparation the test data, the system under study is tested.

Those test data, error were found and correct it by following testing steps and correction are recorded for feature reference. Thus a series is testing performed on the system before is ready for implementation.

**6.4Testing methods:**

System testing is the stage of implementation. This is to check whether the system after works accurately and efficiently before live operation commences. Testing is the vital to the success of the system. The candidate system is subject to variety of testes: online response, volume, stress, recovery, security and usability tests. A series of tests are performed for the proposed system is ready for user acceptance testing.

**The various type of testing on the system is:**

* Unit testing
* Integrated testing
* Validation testing
* Output testing
* User acceptance testing

**6.4.1 Unit testing:**

Unit testing focuses on verification effort on the smallest unit of software design module. Using the unit test plans. Prepared in the design phase of the system as a guide important control paths are tested to uncover errors within the boundary of the modules. The interfaces of each of the modules under consideration are also tested. Boundary conditions were checked.

All independents paths were exercised to ensure that all statements in the module executed at least once and all error-handling paths were tested. Each unit was thoroughly tested to check if it might fall in any possible situation. This testing was carried out during the programming itself. At the end of this testing phase each unit was found to be working satisfactorily as regarded to the expected out tom the module.

**6.4.2. Integration Testing:**

Data can be across an interface one module can have an adverse effect on another’s Sub functions when combined may not produce the desired major function; global data structures can present problems. Integration testing is a symmetric technique for constructing tests to uncover errors associated with the interface. All modules are combined in this testing step. Then the entire program was tested as a whole.

**6.4.3.Validation Testing:**

At the culmination of integration testing software is completely assemble as a package. Interfacing errors have been uncovered and corrected and fin; series of software test-validation testing begins. Validation testing can be defined in many ways but a simple definition is that validation succeeds when software functions in manner that is reasonably expected by the consumer.

Software validation is achieved through a series of black box tests that demonstrate conformity with requirement after validation test has been conducted one of two conditions exists.

* The function or performance Characteristics confirm to specification that are accepted.
* A validation from specification is uncovered and a deficiency created.

Deviation or errors discovered at this step in this project is corrected prior to completion of the project with the help of user by negotiating to establish a method for resolving deficiencies. Thus the proposed system under consideration has been tested by using validation testing and found to be working satisfactorily.

**6.4.4.Output Testing:**

After performing the validation testing the next step is output testing of the proposed system since a system is useful if it does not produce the required output in the specific format required by them tests the output generator displayed on the system under consideration. Here the output is considered in the two ways- one is the onscreen and the other is printed format. The output formation the screen is found to be correct as the format was designed in the system design phase according to the user needs. As far as hardcopies are considered it goes in terms with the user requirement. Hence output testing does not result any correction in the system.

**6.5 User acceptance testing:**

User acceptance of the system is a key factor for success of any system. The system under consideration is tested for user acceptance by constantly keeping in touch with prospective system and user at the time of developing and making changes whenever required.

**6.6Test cases:**

* Test Unit: Admin

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Serial No.** | **Module Name** | **Condition to be tested** | **Expected Output** | **Remarks** |
| 1 | Admin Login | If any field in the form is empty | Alert the admin to enter all the fields and then proceed | Success full |
| If user name and password is incorrect | Alert the admin to enter valid login and then proceed | Success full |
| If user name and password is correct | Then proceed to the admin home page | Success full |
| 2 | Admin Change Password | If any field in the form is empty | Alert the admin to enter all the fields and then proceed | Success full |
| If old, new and retype password is correct | Password should be successfully changed | Success full |
| If old, new and retype password is incorrect | Alert the admin to enter correct password | Success full |
| 3 | Admin add dealers | If any field in the form is empty | Alert the admin to enter all the fields and then proceed | Success full |
| If same dealer is added | Alert the admin to not enter the same dealer | Success full |

* Test Unit: Customer

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Serial No.** | **Module Name** | **Condition to be tested** | **Expected Output** | **Remarks** |
| 1 | Customer Registration | If any field in the form is empty | Alert the user to enter all the fields and then proceed | Success full |
| If the user enters non-character values in name field | Alert the user to enter character values and then proceed | Success full |
| If the user enters non-numeric values in name field | Alert the user to enter numeric values and then proceed | Success full |
| If the user enters invalid format for email Id | Alert the user to enter the valid format and then proceed | Success full |
| If pin code is less than or greater than 6 | Alert the user to enter the pin code is equal to 6 | Success full |
| If phone no is less than or greater than 12 | Alert the user to enter the phone no is equal to 12 | Success full |
| 2 | Customer Login | If any field in the form is empty | Alert the user to enter all the fields and then proceed | Success full |
| If user name and password is incorrect | Alert the user to enter valid login and then proceed | Success full |
| If user name and password is correct | Then proceed to the candidate home page | Success full |
| 3 | Customer Mail | If any field in the form is empty | Alert the user to enter all the fields and then proceed | Success full |
| If mail is incorrect | Alert the user to enter correct to\_id and then proceed | Success full |
| 4 | Customer Change Password | If any field in the form is empty | Alert the user to enter all the fields and then proceed | Success full |
| If old, new and retype password is correct | Password should be successfully changed | Success full |
| If old, new and retype password is incorrect | Alert the user to enter correct password | Success full |

* Test Unit: Dealer

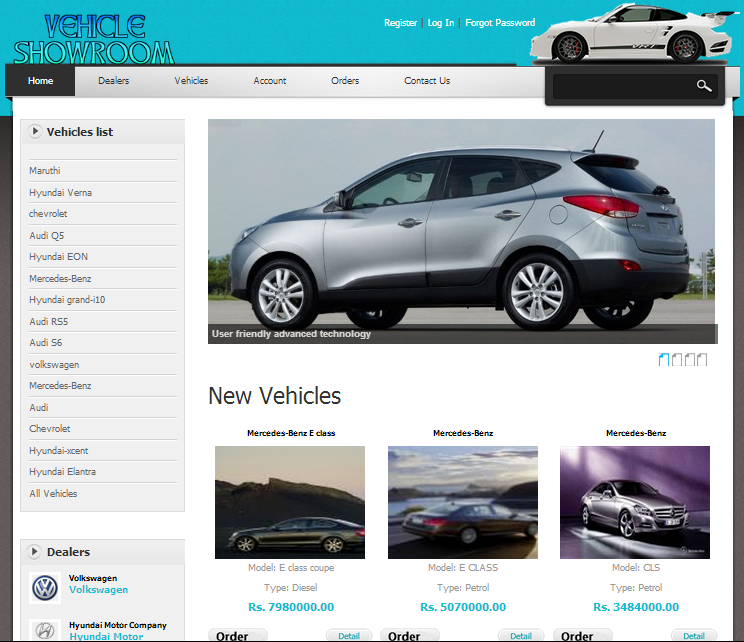
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Serial No.** | **Module Name** | **Condition to be tested** | **Expected Output** | **Remarks** |
| 1 | Dealer Registration | If any field in the form is empty | Alert the user to enter all the fields and then proceed | Success full |
| If the user enters non-character values in name field | Alert the user to enter character values and then proceed | Success full |
| If the user enters non-numeric values in name field | Alert the user to enter numeric values and then proceed | Success full |
| If the user enters invalid format for email Id | Alert the user to enter the valid format and then proceed | Success full |
| If pin code is less than or greater than 6 | Alert the user to enter the pin code is equal to 6 | Success full |
| If phone no is less than or greater than 12 | Alert the user to enter the phone no is equal to 12 | Success full |
| 2 | Dealer Login | If any field in the form is empty | Alert the user to enter all the fields and then proceed | Success full |
| If user name and password is incorrect | Alert the user to enter valid login and then proceed | Success full |
| If user name and password is correct | Then proceed to the employer home page | Success full |
| 3 | Dealer Mail | If any field in the form is empty | Alert the user to enter all the fields and then proceed | Success full |
| If mail is incorrect | Alert the user to enter correct mail and then proceed | Success full |
| 4 | Dealer Change Password | If any field in the form is empty | Alert the user to enter all the fields and then proceed | Success full |
| If old, new and retype password is correct | Password should be successfully changed | Success full |
| If old, new and retype password is incorrect | Alert the user to enter correct password | Success full |
| 5 | Dealer Add showrooms | If any field in the form is empty | Alert the dealer to enter all the fields and then proceed | Success full |
| If the dealer enter invalid details of vehicle | Alert the dealer to enter valid details and then proceed | Success full |
| If the dealer enter non-character value for showroom | Alert the dealer to enter the character value and then proceed | Success full |
| If the dealer enter non-numeric value for cost of  vehicles | Alert the dealer to enter the numeric value and then proceed | Success full |

CHAPTER 7:

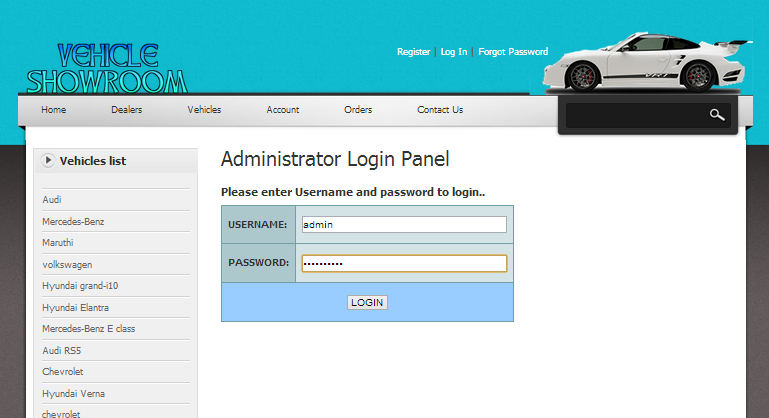
Screen shots

SCREEN SHOTS

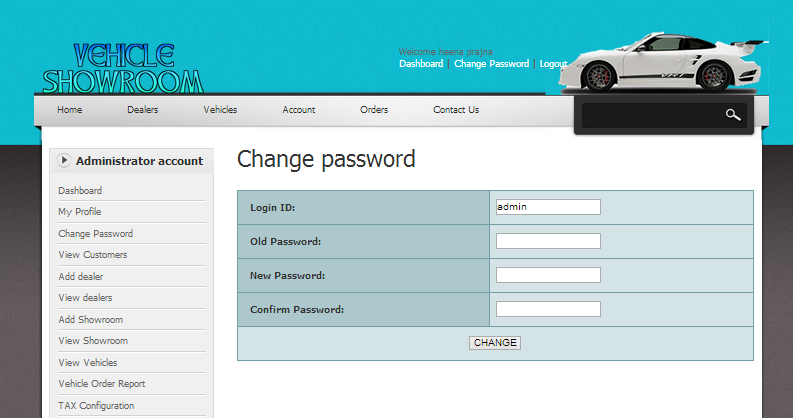
7.1 HOME PAGE OF ONLINE VEHICLE SHOWROOM:



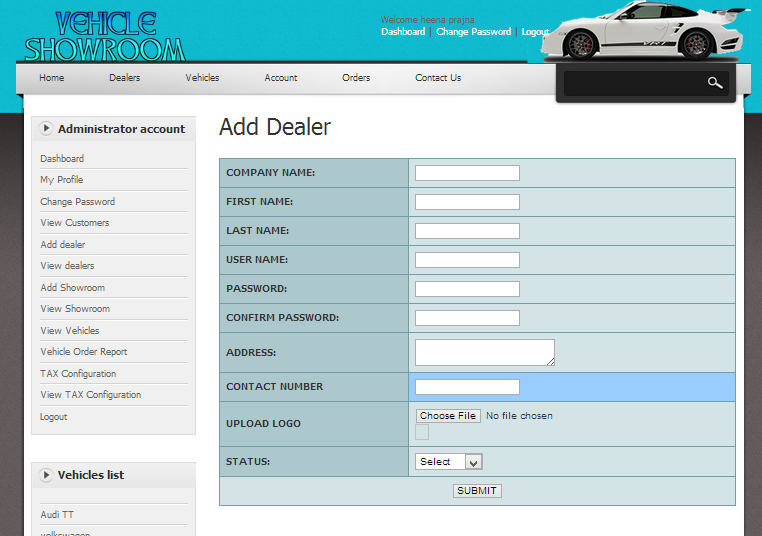
7.2 ADMIN LOGIN:



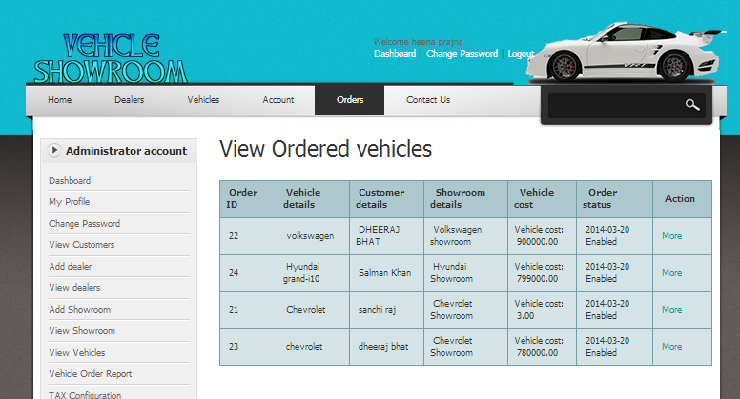
7.3 ADMIN CHANGEPASSWORD:



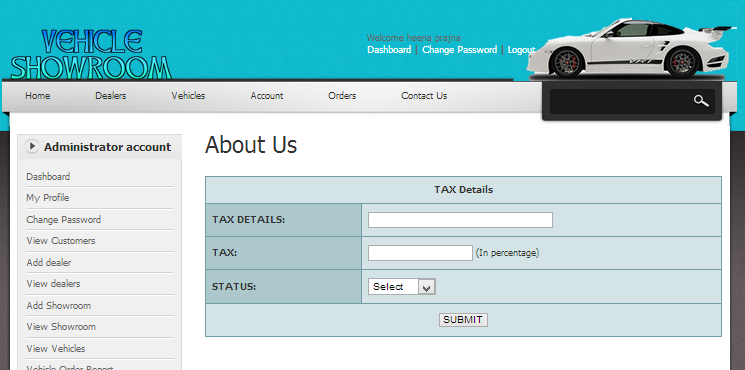
7.4 ADD DEALER:



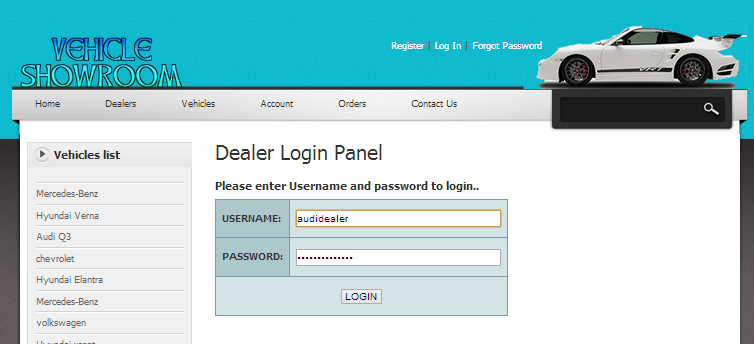
7.5 VIEW ORDERED VEHICLES:



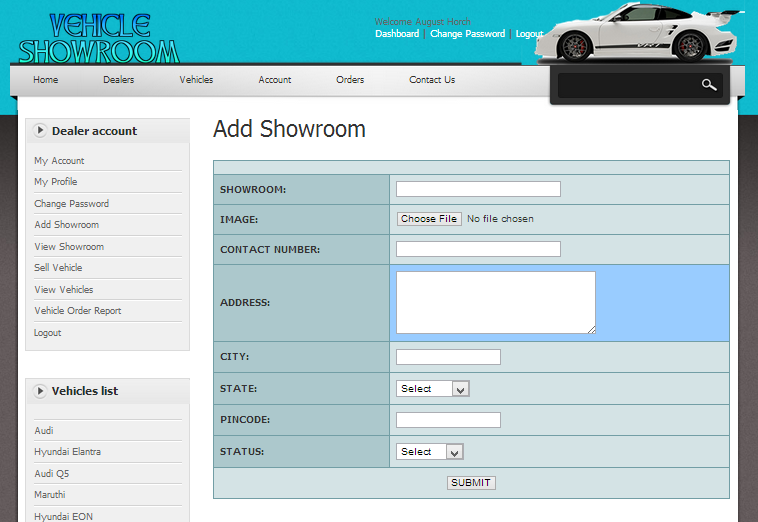
7.6 TAX CONFIGURATION:



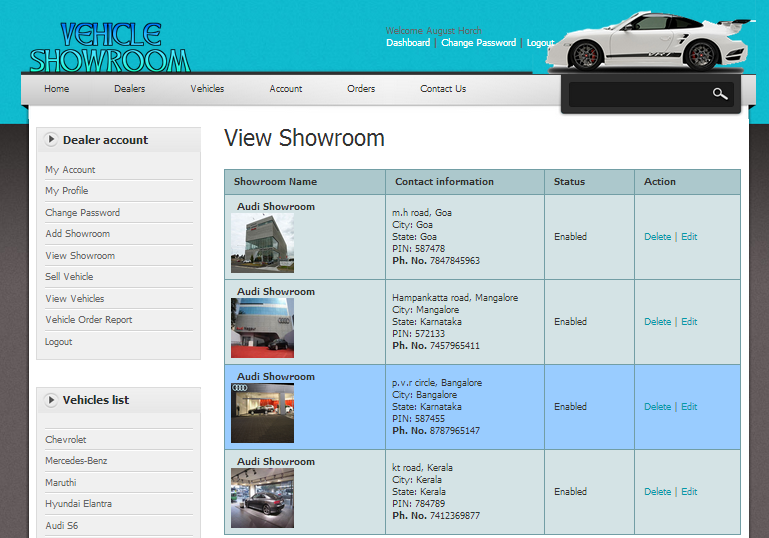
7.7 DEALER LOGIN:



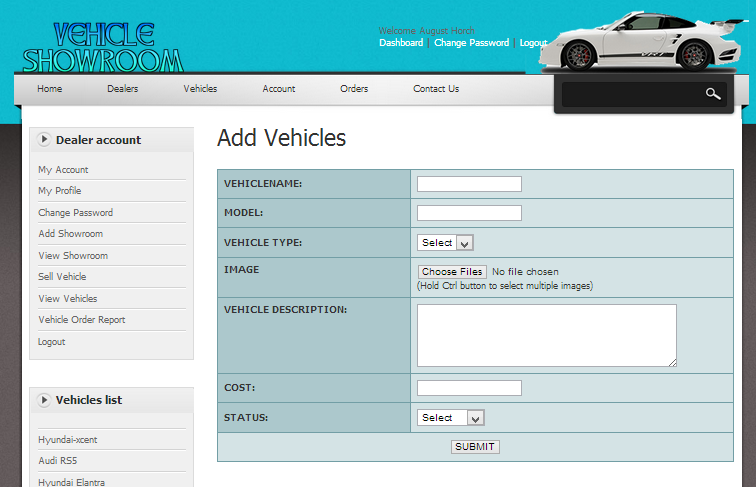
7.8 ADD SHOWROOM:



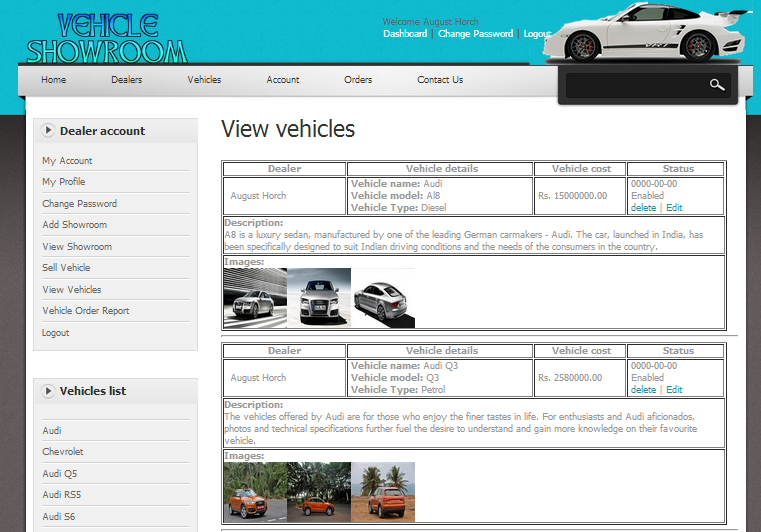
7.9 VIEW SHOWROOM:



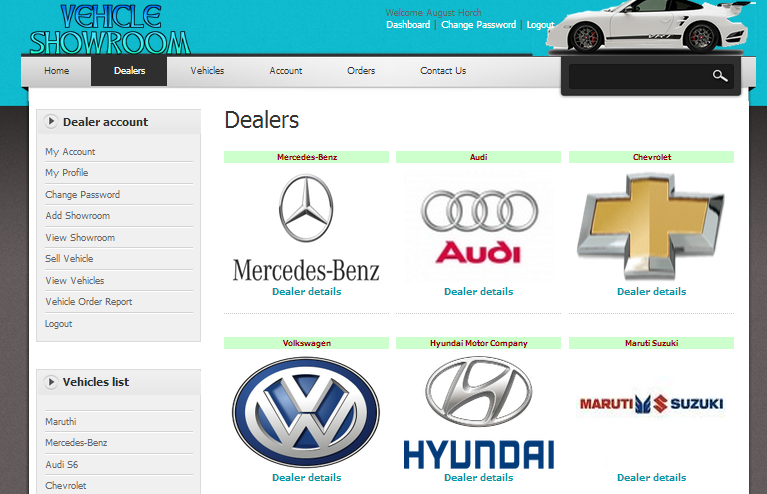
7.10 ADD VEHICLES:



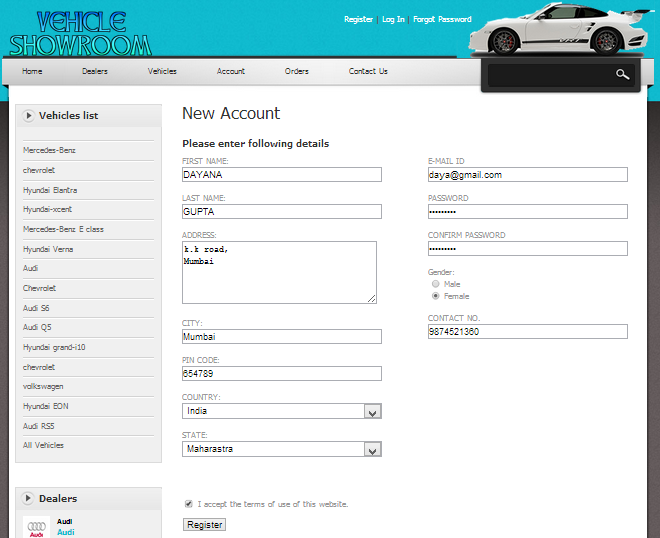
7.11 VIEW VEHICLES:

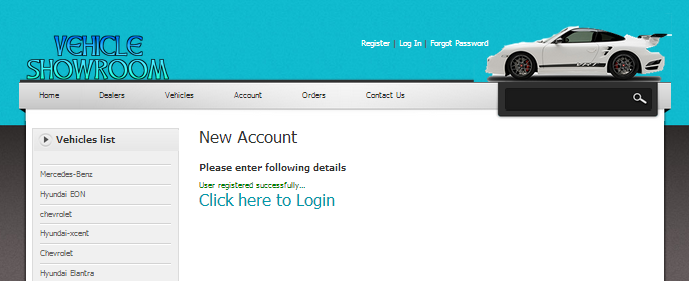


7.12 DEALERS:

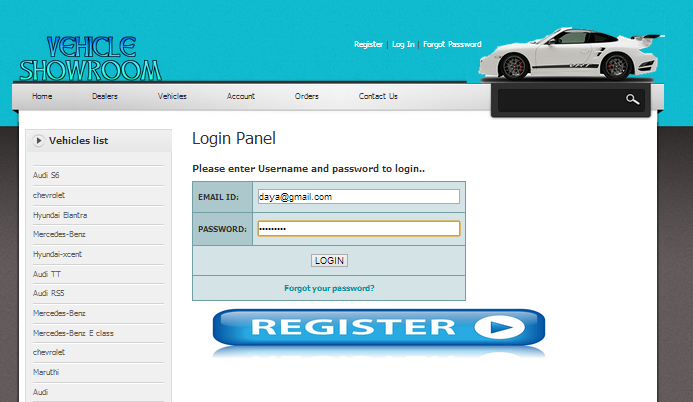


7.13 CUSTOMER REGISTRATI0N:

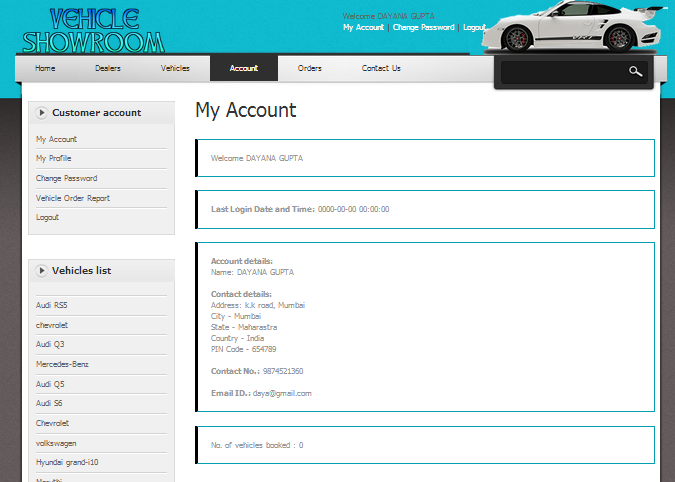




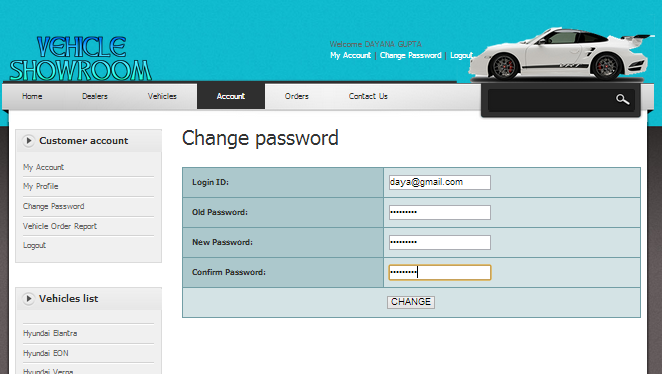
7.14 CUSTOMER LOGIN:



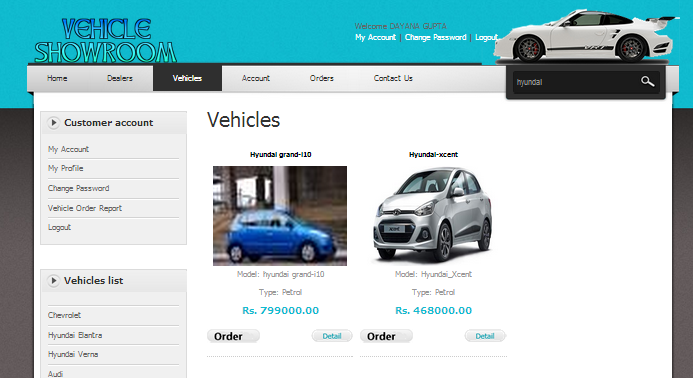
7.15 CUSTOMER DETAILS:



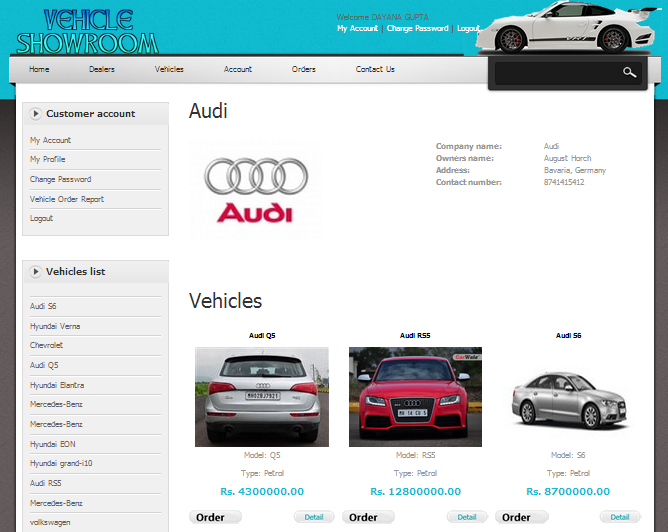
7.16 CUSTOMER CHANGEPASSWORD:



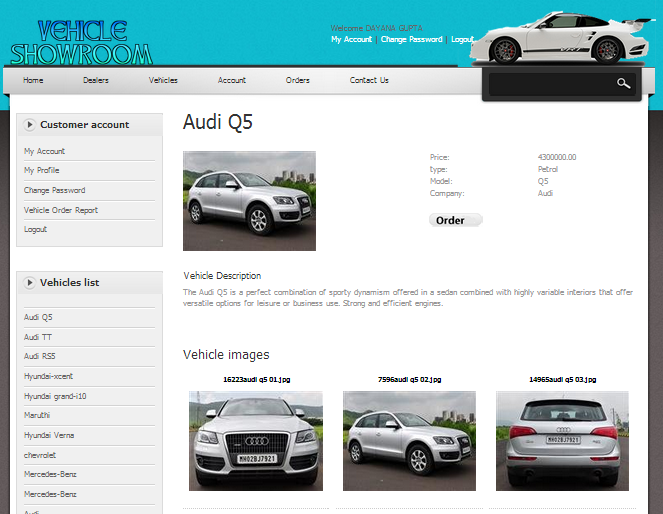
7.17 VEHICLE SEARCH:



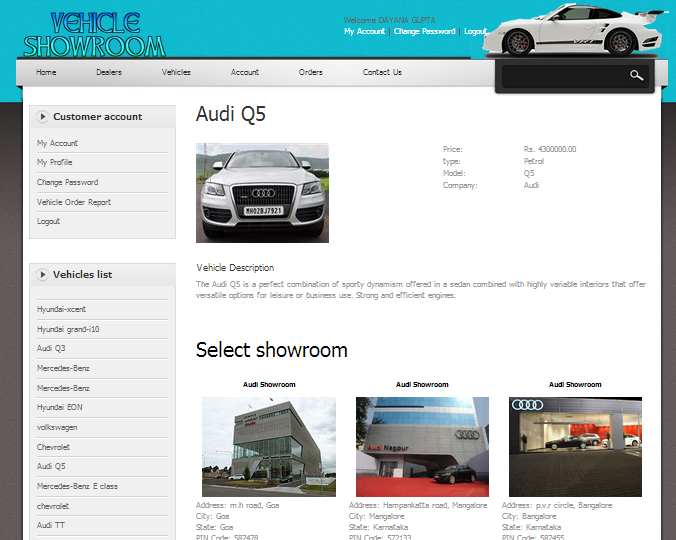
7.18 VEHICLE SELECTION:



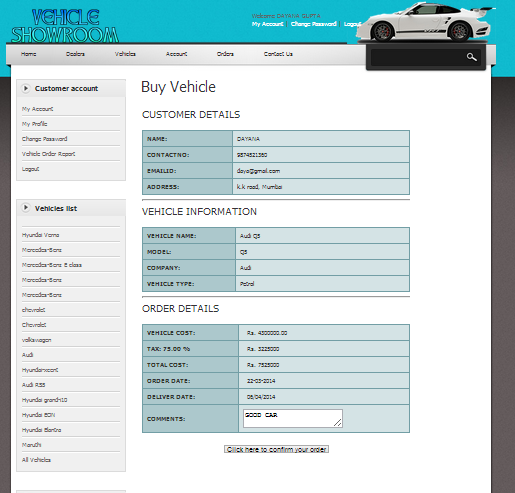
7.19 VEHICLE DETAILS:



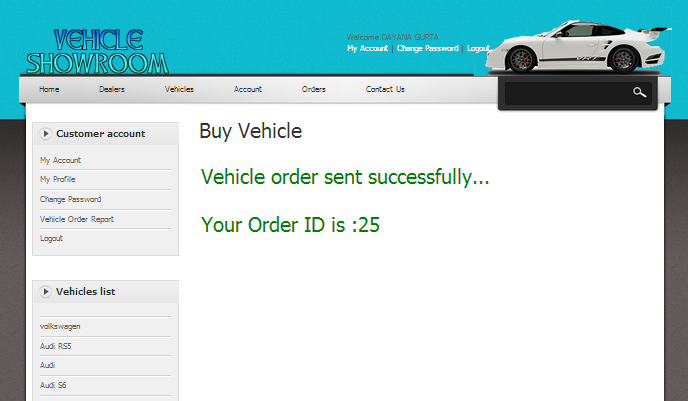
7.20 SELECT SHOWROOM:



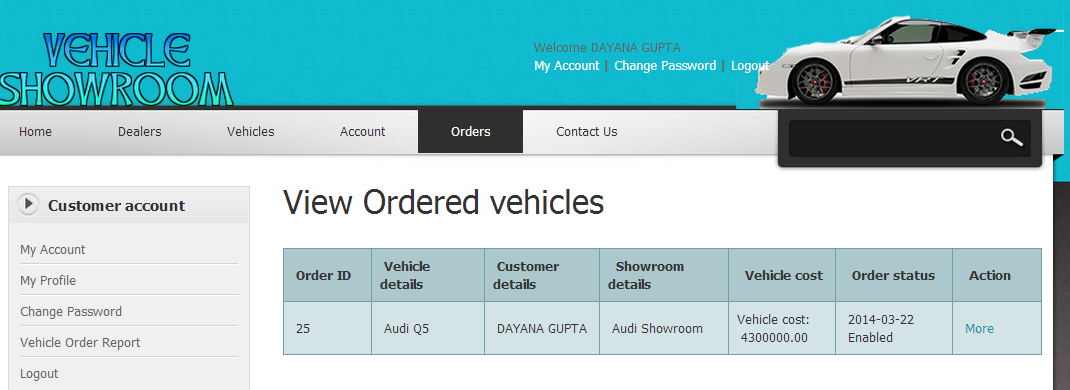
7.21 CUSTOMER DETAILS:



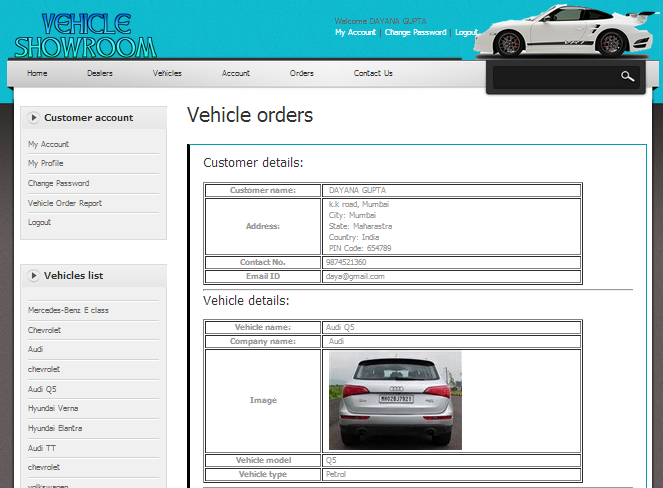
7.22 ORDERED SUCCESSFULLY:

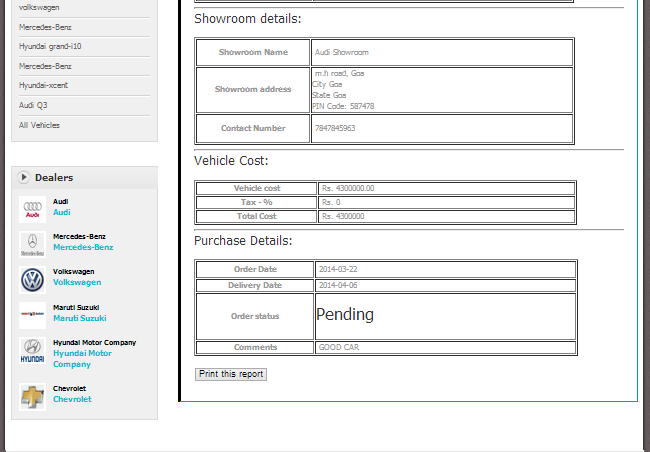


7.23 VIEW ORDERED VEHICLES:



7.24 VEHICLE ORDER DETAILS:





CHAPTER 8:

User Manual

USERMANUAL

**8.1. Introduction:**

The project “Online Vehicle showroom” is user friendly software which is developed using PHP as the front end and MYSQL as the back end. This software is developed under windows operating system. This software runs on Intel core i3 or faster with 2GB Ram.

Online Vehicle showroom provides a web interface that is powerful, flexible and easy to use along with all the critical information, easily accessible. The system provides three major screen pages. That is for admin, customer and other is dealer. Before the customer or dealer uses the system he/she has to register him or herself by providing necessary information. After the customer or dealer registration their information is send to admin. When the admin can approve customer or dealer registration then only they can login to their home pages. Here customer, dealer and admin are having separate home pages. Customer home page allows only the customer to view, modify, delete their profile and also customer can order the vehicles through online, customer can have the mail option to send the messages to the admin and dealer. Dealer home page allows only the dealer to view, modify, delete their profile; dealer can have the mail option to send the messages to the admin and customer, dealer can add the showroom and also view and add the vehicles and view the customer who are ordered for vehicles. Admin home page allows only the admin to approve/reject the customer and dealer; admin can have the mail option to send the messages to the dealer and customer, admin can add the dealer.

**8.2.Major Screen Pages and descriptions**

|  |  |
| --- | --- |
| **Major screen page of admin** | **Description** |
| Login Page | Here the Admin will login |
| Home page | This is main page for admin |
| Approve customer | Here admin can approve/reject the candidate |
| Approve dealer | Here admin can approve/reject the employer |
| Change password | Here admin can change his password |
| Add dealer | Here admin can add the dealer |
| Reporting | Here admin can searches customer or dealer |

|  |  |
| --- | --- |
| **Major screen page of customer** | **Description** |
| Login Page | Here the candidate will login |
| Home page | This is main page for candidate |
| View profile | Here customer can view their profile |
| Modify | Here customer can modify their details |
| Delete | Here customer can delete their account |
| Change password | Here customer can change his password |
| Online vehicle order | Here customer can order the vehicles |

|  |  |
| --- | --- |
| **Major screen page of**  **dealer** | **Description** |
| Login Page | Here the dealer will login |
| Home page | This is main page for employer |
| View profile | Here dealer can view their profile |
| Modify | Here dealer can modify their details |
| Delete | Here dealer can delete their account |
| Change password | Here dealer can change his password |
| Add showroom | Here dealer can add the showroom |
| View customer | Here dealer can view the customer details |
| Add vehicle | Here dealer can add the vehicles |

**8.3.Installation Guidelines:**

To run the system the user has to check whether the system satisfies the requirements given below.

a) The system should run on any windows platform (windows XP, windows 7)

b) MY SQL should exists in the system for backend construction

c) Hardware requirements

* 2GB of RAM
* Intel core i3 400 MHZ
* 14” color monitor

The Online vehicle showroom system covers the following features.

* Attractive WebPages
* User friendly interface
* Complete details about the vehicles
* Data security is maintained.

d) Software requirements

* Front-end-tools: Apache, PHP (Hypertext Preprocessor), Java Script, HTML (Hyper Text Markup Language).
* Back-end-tool: My SQL database.

**8.4.Method of Installation:**

I) Open web-Application directory which is present in installed folder.

II) Now copy the Web-Application into this folder

III) Along with Web information folder which intern having classes SRC, lib folders

IV) Open any internet browser and type the URL as <http://localhost/> and present

V) Click on Apache manager

VI) This will display your web-Application with status as true or false

VII) Now open your Application with the following URL’S.

CHAPTER 9:

Conclusion

CONCLUSION

The project “ONLINE VEHICLE SHOWROOM” holds details about the new vehicles that is added by the dealer who should register with this site.and make customer book vehicles through online. It keeps track of Customers records, Vehicle records, Payment and billing records, dealer records, etc.

The system generates Invoice and Bill after purchasing or booking of new vehicle. This is web application and it’s developing using PHP language. All the records stores in MySQL Database. It provides features like adding vehicle,deletevehicle,searching vehicle and updated all ready existed records.

This totally self contained and works efficiently. It provides simple database rather then complex ones for high requirements and it provides good and easy graphical user interface new, naïve as well as experienced users of the computers .

This project has fulfilled all the objectives identified.the module

has been developed in an attractive manner keeping in the mind about

customer convenience.so customer with common computer language can

handle it very easily.

CHAPTER 10:

Bibliography

Bibliography

Book references:

* Software engineering by PankajJalote
* Learning PHP, MySQL, JavaScript, and CSS: A Step-by-Step Guide to Creating Dynamic Websites by Robin Nixon

Internet references:

* [www.w3schools.com](http://www.w3schools.com)
* [www.php.net](http://www.php.net)
* [www.mysql.com](http://www.mysql.com)