AUTOSHOP

(Online Used Vehicle Sales System)

Project Report Submitted by

JISHNUMON P B

Reg. No: LAJC16MCA042

In Partial fulfillment for the award of the degree

Of

MASTER OF COMPUTER APPLICATIONS (MCA)

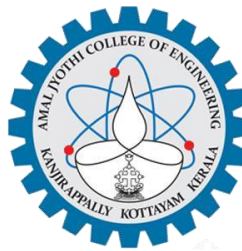**AMAL JYOTHI COLLEGE OF ENGINEERING**

KANJIRAPPALLY

APJ ABDUL KALAM TECHNOLOGICAL UNIVERSITY

2017-2019

DEPARTMENT OF COMPUTER APPLICATIONS

AMAL JYOTHI COLLEGE OF ENGINEERING

KANJIRAPPALLY

<u>CERTIFICATE</u>

This is to certify that the project entitled **"AUTOSHOP "** is a bonafide record of the project done by **JISHNUMON P B (LAJC16MCA042) ,** during the academic year **2017-2019** carried out under my supervision. Certified further, that to the best of my knowledge the work reported herein does not form part of any other project report or dissertation on the basis of which a degree or award was conferred on an earlier occasion on this or any other candidate.

Fr. Rubin Thottupuram	**Sr. Elsin Chakkalackal**	**Ms. Teenu V Therese**
Head of the Department	Project Coordinator	Internal Guide

DECLARATION

I hereby declare that the project report **"AUTOSHOP"** is a bonafide work done at Amal Jyothi College of Engineering, towards the partial fulfilment of the requirements for the award of the Degree of Master of Computer Applications (MCA) from APJ Abdul Kalam Technological University, during the academic year 2017-2019.

Date...................   **JISHNUMON P B**

KANJIRAPPALLY   **Reg.No : LAJC16MCA042**

ACKNOWLEDGEMENT

ABSTRACT

The project is an online Automobile shopping system that allows web users to purchase used vehicle online without visiting any physical location.

This project allows the user to purchase Automobile online and Bid vehicle in Auction. The Administrator updates the information about new Automobiles concurrently. Only registered customers can purchase Automobiles bid from the system. The system also maintains the details of new Automobiles, company details etc. The user must register in the site to access their accounts; after login they can buy used cars by bidding Automobiles which are available for auction. The user can bid a vehicle with a high rate than the latest Bid Rate, before the date expires. After date expires, the site sends the Confirmation letter to the user who bid the vehicle with highest rate through email-id which they specified.

This system allows the users to search items category wise, then Brand wise and Model wise. Thus this system provides all the basic functionalities to a user who would like to purchase used vehicle in online.

TABLE OF CONTENTS

List of Abbreviation

IDE   -   Integrated Development Environment

HTML   -   Hyper Text Markup Language.

CSS   -   Cascading Style Sheet

SQL   -   Structured Query Language

DFD   -   Data Flow Diagram

PTA   -   Pure Tone Average

CHAPTER-1
INTRODUCTION

1.INTRODUCTION

ABOUT THE PROJECT

The project entitled '**Autoshop'** is an online Automobile shopping system that allows web users to purchase used cars online without visiting any physical location. **'Autoshop'** allows the user to purchase Automobile online and Bid cars in Auction. The Site Administrator updates the information about new Automobiles concurrently. Only registered customers can purchase Automobiles bid from Autoshop. The user must register in the site to access their accounts; after login they can buy used cars by bidding Automobiles which are available for auction. The user can bid a car with a high rate than the latest Bid Rate, before the date expires. After date expires, the site sends the Confirmation letter to the user who bid the car with highest rate through email-id which they specified.

This system allows the users to search items category wise, then Brand wise and Model wise. Thus this system provides all the basic functionalities to a user who would like to purchase used cars in online.

The important modules of the project are:

- User Module
- Auction
- Administrator Module
- Purchasing

The project has been developed based on these modules.

1. **User Module**

   The main processes in this module are given below:

o Change Password

o Edit Details

o Purchase

o Bidding

o Product Review

o View Updations about new product

2.Auction

o   Select Cars

o   Add to Auction

o   Updating Details

o   Setting Initial Rate

o   Sending Confirmations

3. Administrator Module

o   Edit Profile

o   Confirm Employee account

o   Manage Customers

o   Checking Products

o   Add vehicle to Auction

4. Purchasing

o   Selecting a Model

o   Displaying Details

o   Purchase

1.1 PROJECT OVERVIEW

Niyoga Technologie

Niyoga Technologies is a full-Web cycle product development services leader that combines deep domain expertise and cross-industry experience to connect makers with markets worldwide.We use open source technologies [php, mysql, drupal, magento, wordpress, Zend] to reduce cost and improve qualities, that helps our clients to get better service at low cost.

Using insight gained from working on innovative products and disruptive technologies, we collaborate with customers to show them how strategic research and development can become a tool for managing their future. We build partnerships with market-defining business and technolog leaders who want to make amazing products, discover new revenue opportunities, and accelerate time to market.

We partner with influential companies–from disruptive start-ups to established leaders–including over eighty percent of the world's top technology brands. Our customers are found in every major industry–from software development to device manufacturing to consumer electronics–and our global presence allows us to help them get to market rapidly while managing costs at every stage of the product lifecycle.

Our product strategists, designers, and engineers take a shared product responsibility approach with our customers. We are committed to sharing the realization of a complete product roadmap– from creating industry–defining products with the latest technologies to getting them to market faster. Because many of our customers choose to work with us as an equal partner in the product development process, we do everything we can to make that partnership worthwhile.

We treat our customer relationships like true partnerships. We know client satisfaction is the base of a company, so we take maximum effort to keep industry standard in our project and give support after delivery. Feel free to contact us to complete your project. The global marketplace that is the web is now saturated with businesses all competing for the attention

of potential customers and it is more important than ever to have a website that is professionally designed and visually striking. Niyoga Technologies can help your company maximize the benefits of an online presence

1.2 PROJECT SPECIFICATION

Analysis is a detailed study of various operations performed   by a system and their relationships within and outside the system. During analysis, data are collected on the available files, decision points and transactions handled by the present system. Interviews, on-site observation and questionnaire are the tools used for system analysis. Using the following steps it becomes easy to draw the exact boundary of the new system under consideration:

 All procedures, requirements must be analyzed and documented in the form of detailed data flow diagrams (DFD's), data dictionary, logical data structures and miniature specifications. System Analysis also includes sub-dividing of complex process involving the entire system, identification of data store and manual processes. The main purpose of analysis activity is to clearly understand the exact requirements of the system and eliminates the entire problem: inconsistency and its incompleteness of the existing system.

For system analysis, first we visited National Institute of Speech and Hearing. Discussions were made with the Audiologist about the testing and various factors that consider for testing. The discussions were very useful and informative. Through the discussion we came to know that there are a number of external and internal factors that affect the testing.

CHAPTER-2
SYSTEM STUDY

2.1 INTRODUCTION

System analysis is a process of gathering and interpreting facts, diagnosing problems and the information to recommend improvements on the system. It is a problem solving activity that requires intensive communication between the system users and system developers. System analysis or study is an important phase of any system development process. The system is studied to the minute's detail and analysed. The system analyst plays the role of the interrogator and dwells deep into the working of the present system. The system is viewed as a whole and the input to the system are identified. The outputs from the organizations are traced to the various processes. System analysis is concerned with becoming aware of the problem, identifying the relevant and decisional variables, analysing and synthesizing the various factors and determining an optimal or at least a satisfactory solution or program of action.

A detailed study of the process must be made by various techniques like interviews, questionnaires etc. The data collected by these sources must be scrutinized to arrive to a conclusion. The conclusion is an understanding of how the system functions. This system is called the existing system. Now the existing system is subjected to close study and problem areas are identified. The designer now functions as a problem solver and tries to sort out the difficulties that the enterprise faces. The solutions are given as proposals. The proposal is then weighed with the existing system analytically and the best one is selected. The proposal is presented to the user for an endorsement by the user. The proposal is reviewed on user request and suitable changes are made. This is loop that ends as soon as the user is satisfied with proposal.

Preliminary study is the process of gathering and interpreting facts, using the information for further studies on the system. Preliminary study is problem solving activity that requires intensive communication between the system users and system developers. It does various feasibility studies. In these studies, a rough figure of the system activities can be obtained, from which the decision about the strategies to be followed for effective system study and analysis can be taken

2.2 EXISTING SYSTEM

The Autoshop is that they have no centralised system for the running of their business. Most activities and processes of the business are paper based and independent of 2 each other. The only computer system currently used by the company is a website and a simple stock control system whereby they can add, edit and delete vehicles from their stock, in addition to a finance calculator for calculating the finance available to customers. These systems are not linked together. The current process of selling a vehicle requires the staff to write out all the details of the customer, vehicle, and finance on a paper invoice, and then file these away. This makes searching for the details of a customer or a vehicle sold several months ago a problem for the staff as they currently have to manually look through the filing cabinet and many invoices which can be a lengthy process. The company stores a range of vehicle offsite as well as onsite and therefore the customers that come to visit the showroom do not get to view the whole range of vehicles that   sell. This can deter away the customers as they may find that the company does not stock a vehicle that satisfies their requirements, when in fact they possibly may do. In the current system the booking of a car for a repair or service is simply noted down on paper, this can easily be misplaced in the office within the various piles of paperwork. The VAT costs for each car sold by the company have to be individually calculated by the staff which in some cases can be prone to error as it is a lengthy repetitive process, and again results to more paperwork being produced that has to be filed away. These problems result in increased workload for the staff and the possibility of data being lost as it is physically stored in several places.

2.4 PROPOSED SYSTEM

The proposed solution for solving the problems encountered by Steve Graves Autoshop is to develop a computerised system whereby most business processes can be carried out in a single

place, and all data relevant to the company is stored centrally. This can lead to a reduction on the workload for staff and improve the efficiency of running and managing the business. It is hoped that the new system will be able to bring most aspects of the business together so that they can all be carried out together. Therefore be able to manage stock control for vehicles on a computer system, and then use this data to process the sale of a vehicle by adding the customer and finance details to an invoice, and finally storing these in a database where all the details of the sale can easily be retrieved in the future directly from the system. Additionally it is hoped that the new system could solve the other problems encountered in the current system including the ability to search for vehicles with a customer's specific requirements as the input, schedule appointments for a repair or service of a vehicle on the system, automatically calculate the VAT on cars sold and calculate the total amount of VAT owed, along with many other capabilities to improve on the current system. Therefore the proposed solution is hoped to eradicate the problems encountered in the current system and aid the staff in their daily activities and processes of running and managing the business.

CHAPTER 3
REQUIREMENT ANALYSIS

3.1 FEASIBILITY STUDY

A feasibility analysis of the proposed online used car sales system was conducted. Feasibility analysis is performed to pick the system that meets the performance prerequisites at lower cost. It is a test of system proposed as indicated by work capacity, effect of the organization and powerful employments of assets. It likewise assesses the assets. It assesses the current systems and it methodology create a candidate system which are suitable to take care of the problem. A feasibility test finds answers to the following questions:

- What are the resources available?
- What are the needs of the user?
- How the candidate systems meet the needs?
- Whether the problem is worth solving?
- How the system works in the proposed organization?

After the initial stage of feasibility analysis of Autoshop the above questions are well answered. In the study phase the system is found to be highly feasible with the satisfaction of the users.

At first, the resources available were checked. After the analysis, it reached the conclusion that Autoshop can cope up with the available resources. The needs of the user were clearly identified. This was done with constant interviews with various people . Further study was made to check whether the candidate system is capable of satisfying the user needs. Next a study was made on whether Autoshop could actually solve the problem i.e. whether the system can perform its intended task. The predefined calculation methods were used for this purpose.

After the feasibility analysis it was found that Autoshop was feasible as it answered the above questions. But further more feasibility study is required as below.

3.1.1 Economical Feasibility

Economic feasibility is associated with, whether the system is economically feasible. It is the method used for evaluating the effectiveness of the candidate system and compare with the cost. This is judged by comparing the development cost against the income or benefit analysis, which is the basis for the economic justification of a system. In terms of benefits,

we have to consider both tangible and intangible benefits. Here it is seen that no new software and hardware is needed for the development of the system. Thus, this project is economically feasible. Autoshop doesn't have any operational cost associated with it. All it requires is a normally configured computer with internet connectivity. The procedure is to determine the benefits and savings that are expected from a candidate system and compare them with costs.

3.1.2 Technical Feasibility

A study of the resource availability may affect the ability to achieve an acceptable system. The technical feasibility is frequently the most difficult area encountered at this stage. It is essential that the process of analysis and definition be conducted in parallel with an assessment to technical feasibility. It centers on the existing computer system (hardware, software etc.) and to what extend it can support the proposed system.

The evaluation determines:

❖ Whether the technology needed for the proposed system is available or not?
❖ Can the work for the project be done with current equipment existing software technology with available personal?

❖ Can the system be upgraded if developed?
❖ If the new technology is needed, then what can be developed. This is the concerned with specifying equipment and software that will successfully satisfy the user requirement.

The technical needs of the system may include:

Front-end and back-end selection:

An important issue for the development of a project is the selection of suitable front-end and back-end. When we decided to develop the project we went through an extensive study to

determine the most suitable platform that suits the needs of the organization as well as help in development of the project. The aspects of our study included the following factors:

Front-end selection:

It must have a graphical user interface what assists users that are not from IT background. The factors considered are:

1. Scalability and extensibility

2. Flexibility

3. Robustness

4. Platform independent.

5. Easy to debug and maintain.

6. Event driven programming facility.

7. Front end must support some popular back end like MySQL.

According to the above stated features we selected PHP as the front-end for developing our project.

Back-end selection:

While selecting back-end the following factors are considered:

1. Multiple user support.

2. Efficient data handling.

3. Provide internet features for security.

4. Efficient data retrieval and maintenance.

5. Stored procedure.

6. Operating System compatible.

7. Easy to install.

8. Easy to implant with the Front-end.

   According to above stated features we selected MySQL as the backend.

3.1.3 Behavioural Feasibility

Getting the right manpower for the system to function successfully, should not be a cause to worry. The proposed Autoshop is very much user friendly. So it is very easy for the user so work on with Autoshop. It only needs basic knowledge. Therefore, it is operationally feasible. Since the software used is so simple, chance of technical assistance is also rare.

3.2 SYSTEM SPECIFICATION

3.2.1 HARDWARE SPECIFICATION

The selection of hardware configuration is a very important task related to software development. Particularly insufficient RAM may affect adversely on the speed and correspondingly on the efficiency of the entire system. The processor should be powerful to handle all the operations. Hard disk should have sufficient capacity to store database and application. The network should be well sufficient to handle the communication fast. Minimum hardware specification required to run this application:

Processor   : Dual core or compatible
Speed   : 800 MHz or above

Hard disk   : 180 GB

RAM   : 512 MB

Monitor   : Compatible monitor with 600 x 800 resolutions

Internet   : 56 kbps/above

Input   : Headset /Speaker

3.2.2 SOFTWARE SPECIFICATION

After the collection of all the required information regarding the software to be developed and has removed all incompleteness, inconsistencies and anomalies from the specification and then starts to systematically organize the requirements in the specification. Software's required to run this application:

Operating System   :Windows XP or higher/Linux

Front End   : PHP, Java Script, Jquery

Back End   :MYSQL

3.3 SOFTWARE DESCRIPTION

3.3.1 PHP

PHP, also known as the Hypertext Preprocessor, is a widely used computer programming language. It is used to create dynamic Web pages, or Web pages that update and display information depending on the user's activity. PHP is mainly a server side language, allowing it to process Web pages faster and easier, providing a better user experience.

Rasmus Ledorf, a Danish computer programmer, introduced the first version of PHP in 1995. He called the initial set of codes the "Personal Home Page Tools," which he used as a code to track visitors looking at his online resume. After some time, he added more functionality to the script and released the source code to everyone. Ledorf continued improving and developing PHP by himself until 1997.

During that same year, two more programmers, Andi Gutmans and Zeev Zuraski, coordinated with Ledorf to create PHP 3.0. This version was the first to resemble the current popular PHP most programmers use today. It has more functionality and capabilities and was officially called "PHP: Hypertext Preprocessor." The latest PHP is version 5, with several release updates. It still resembles the functionality of version 3.0, but with added features. It mainly runs through the "Zend Engine 2.0."

There are four major advantages when using PHP: accessibility, compatibility, simplicity and extensive community support. Because PHP is open source, access has no restrictions. Programmers interested in using PHP only need to download the scripts, without paying a single cent. It runs in all types of Operating Systems and is also compatible with the most popular Web servers, including Apache and IIS.Programmers also don't have to spend a lot of time learning PHP because its language is related to Perl and C, two of the most popular coding languages. Lastly, PHP has a large online community support. Since it's free, a lot of other programmers share scripts and give advice on how to improve the system.

As of May 2011, there are more than 25 million websites that run on PHP. Most webmasters prefer to use the language because PHP was created to work on Web development. PHP can also be integrated to HTML codes, allowing easier coding and flexibility when building websites. Although using PHP is simple, experienced programmers are usually required

when building complex and large websites. PHP has a lot of extensions and other add-ons, and an inexperienced programmer may get confused sorting through and checking all of the available scripts. Also, codes can sometimes become complex when programmers want to achieve a very specific function.

Lastly, because PHP is mainly a language designed for Web development, building large websites requires programmers to know HTML, JavaScript and CSS.

3.3.2 MYSQL

MySQL officially, but also called My Sequel is (as of July 2013) the world's second most widely used relational database management system (RDBMS) and most widely used open-source RDBMS. It is named after co-founder Michael Widenius's daughter, MY. The SQL acronym stands for Structured Query Language.

The MySQL development project has made its source code available under the terms of the GNU General Public License, as well as under a variety of proprietary agreements. MySQL was owned and sponsored by a single for-profit firm, the Swedish company MySQL AB, now owned by Oracle Corporation.

MySQL is a popular choice of database for use in web applications, and is a central component of the widely used LAMP open source web application software stack (and other 'AMP' stacks). LAMP is an acronym for "Linux, Apache, MySQL,Perl/PHP/Python." Free-software-open source projects that require a full-featured database management system often use MySQL

.For proprietary use, several paid editions are available, and offer additional functionality. Applications which use MySQL databases include:

TYPO3, MODx, Joomla, WordPress, phpBB, MyBB, Drupal and other software. MySQL is also used in many high-profile, large-scale websites, including Google (though not for searches), Facebook, Twitter, Flickr, and YouTube.

MySQL is a relational database management system (RDBMS), and ships with no GUI tools to administer MySQL databases or manage data contained within the databases. Users may use the included command line tools, or use MySQL "front-ends", desktop software and web applications that create and manage MySQL databases, build database structures, back up

data, inspect status, and work with data records. The official set of MySQL front-end tools, MySQL Workbench is actively developed by Oracle, and is freely available for use.

☐ The MySQL Database Server is very fast, reliable, scalable, and easy to use.

If that is what you are looking for, you should give it a try. MySQL Server can run comfortably on a desktop or laptop, alongside your other applications, web servers, and so on, requiring little or no attention. If you dedicate an entire machine to MySQL, you can adjust the settings to take advantage of all the memory, CPU power, and I/O capacity available. MySQL can also scale up to clusters of machines, networked together.

MySQL Server was originally developed to handle large databases much faster than existing solutions and has been successfully used in highly demanding production environments for several years. Although under constant development, MySQL Server today offers a rich and useful set of functions. Its connectivity, speed, and security make MySQL Server highly suited for accessing databases on the Internet.

- **MySQL Server works in client/server or embedded systems.**

    The MySQL Database Software is a client/server system that consists of a multi-threaded SQL server that supports different backends, several different client programs and libraries, administrative tools, and a wide range of application programming interfaces (APIs).

    We also provide MySQL Server as an embedded multi-threaded library that you can link into your application to get a smaller, faster, easier-to-manage standalone product.

CHAPTER 4
SYSTEM DESIGN

4.1 INTRODUCTION

The design phase concentrates on the itemized usage of the system suggested in the feasibility study. Attention is on making an interpretation of performance specification into design specification. The design stage is a move from a client situated document to a document arranged to the programmers or database administrators.

In the design phase of Autoshop, the solutions to the identified problems in the feasibility analysis are extracted. It also involves the creative design of the databases.
System design involves translating information requirements and conceptual design into technical specification and general flow of processing. After the user requirements are identified, related information is gathered to verify the problem and after evaluating the existing system, a new system is proposed.

For the Design of get settled software, care has been given for developing an efficient system, which is user friendly as well as high in performance. It has been assured that the system will have the functions and promises of the proposed system. In the design, the various techniques are used to present a simple efficient system. Design phase acts as a bridge between the software requirement specification and the implementation phase, which satisfies the requirements.

The major step in design is the preparation of input forms and the design of all major output forms in a manner acceptable to the user in all aspects. The base lies in the complete understanding of the system. The data flow diagrams explicitly specify the process flow. Table design or database design is the next major step. Extreme care has to be given here and several concepts of normalization have to be applied at many levels.

Program specification comes next. Here we specify various aspects of the program and also will in detail the major components used in the program. The overall process flow is also explained in much detail. Validation rules and checks come next. Several degrees of validation have to be applied to all inputs and various other operations made on the system. Deviation, if any, has to be checked from these validation rules, imposing the 'not null' constraint is one of the best examples. It has been used in many aspects. Various other constraints are also used.

Security checks refer to avoiding unnecessary access to data that is under use and guarding data from any malice.

4.3 MODULE DESIGN

System design experiences two phases of development: logical and physical design. At the point when analysis readies the logical framework design, they tag the user needs at the level of points of interest that basically decides the data stream into an out of the framework and the obliged information assets. Autoshop goes through both of these phases.

The design covers the following:

1. Reviews the current physical system.

2. Prepare output specification.

3. Prepares input specification

4. Prepares edit, security, and control specification.

5. Specifies the implementation plan.

6. Prepares a logical design walkthrough of the information flow, output, input, controls, and implementation plan.

7. Reviews benefits, costs, target dates, and system constraints.

Following the logical design is the physical design. This produces the working system by defining the design specification that tells programmers exactly what the system must do.

Physical system design consists of the following steps:

1. Design the physical system

2. Plan system implementation.

3. Devise a test and implementation plan and specify any hardware/software

4. Update benefits, costs, conversion date, and system constraints.

1.Registration

In Autoshop, we are providing a registration form, in which the user can give his/her personal details and create an account by giving user name and password.

2.Login

In our system we are provided a login for secured use of the system. The user is identified by his/her email id and password. Once the authenticated person is logged in he/she can manage the system as per their needs.

Output Design

Computer output is the most important and direct source of information to the user efficient output design should improve the system's relationships with the user and help in decision making. A major form of output is hardcopy from the printer. Printout should be designed around output requirement of the user. Output design concepts:

- Output is the most visible component of the information system.
- It is the basis of management's final assessment of the system
- It is designed by rapidly constructing prototypes
- During system design output are modeled as data flows

- Output may introduces new aspects to the system

Report Generation(Output)

This module helps the user to view or generate periodic report of the user hearing sensitivity. The user can generate report on certain time intervals for monitoring the variations. The report can be generated as a printable document.

4.4 DATA FLOW DIAGRAM

Data Flow Diagram (DFD) is a diagram that describes the flow of data and the processes that change data throughout a system. It's a structured analysis and design tool that can be used for flowcharting in place of or in association with information. Oriented and process oriented system flowcharts. When analysts prepare the Data Flow Diagram, they specify the user needs at a level of detail that virtually determines the information flow into and out of the system and the required data resources. This network is constructed by using a set of symbols that do not imply physical implementations. The Data Flow Diagram reviews the

current physical system, prepares input and output specification, specifies the implementation plan etc.

The purpose of the design is to create architecture for the evolving implementation and to establish the common tactical policies that must be used by desperate elements of the system. We begin the design process as soon as we have some reasonably completed model of the behaviour of the system. It is important to avoid premature designs, wherein develop designs before analysis reaches closer. It is important to avoid delayed designing where in the organization crashes while trying to complete an unachievable analysis model.

Throughout the project, the context flow diagrams, data flow diagrams and flow charts have been extensively used to achieve the successful design of the system. In our opinion, "efficient design of the data flow and context flow diagrams helps to design the system successfully without much major flaws within the scheduled time". This is the most complicated part in a project. In the designing process, our project took more than the activities in the software life cycle. If we design a system efficiently with all the future enhancements, the project will never become junk and it will be operational.

The data flow diagrams were first developed by Larry Constantine as way for expressing system requirements in graphical form. A data flow diagram also known as "bubble chart" has the purpose of clarifying system requirements and identifying major transformations that will become programs in system design. It functionally decomposes the requirement specification down to the lowest level. DFD depicts the information flow, the transformation flow and the transformations that are applied as data move from input to output. Data Flow Diagram is quite effective, especially when the required design is unclear and the user and analyst need a notational language for communication. It is used to model the system components such as the system process, the data used by the process, any external entities that interact with the system and information flows in the system.

Four basic symbols are used to construct data flow diagrams. They are symbols that represent data source, data flows, and data transformations and data storage. The points at which data are transformed are represented by enclosed figures, usually circles, which are called nodes.
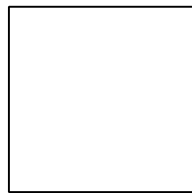
Main symbols used in the data flow diagram are:

1. Circle represents a process that transforms incoming data flows in to outgoing data flows.
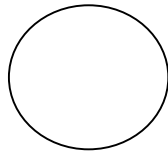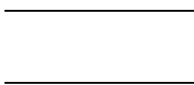
2. A square defines a source and destination of system data.

3. Arrow identifies data in motion.

4. An open rectangle defines a data store, data at rest or temporary repository of data.

Steps to Construct Data Flow Diagrams:-

Four steps are commonly used to construct a

DFD:

- Process should be named and numbered for easy reference. Each name should be representative of the process.
- The destination of flow is from top to bottom and from left to right.
- When a process is exploded in to lower level details they are numbered.

- The names of data stores, sources and destinations are written in capital letters.

Rules for constructing a Data Flow Diagram

- Arrows should not cross each other.

- Squares, circles and files must bear names.

- Decomposed data flow squares and circles can have same names.

- Draw all data flow around the outside of the diagram.

CONTEXT DIAGRAM

Context diagram is also called LEVEL-0 DFD. It represents the entire system in a single bubble. The various entities are with external which the system interacts and the data flows occurring between the systems and external entities are also represented. A system Context Diagram (SCD) is the highest level view of a system, showing the system as a whole and they should always be produced as DFDs. Context Diagrams show the interactions between a system and other factors with which the system is designed to face. SCD is very helpful in understanding the context in which the system will be part of Software engineering.

Context diagrams are used early in a project to get agreement on the scope under investigation. Context diagrams are typically included in a requirements document. These diagrams must be read by all project stakeholders and thus should be written in a plain language so the stakeholders can understand items within the document.The best System Context Diagrams are used to display how system interoperates at a very high level or how systems operate and interact logically. The System Context Diagram is a necessary tool in developing a baseline interaction between systems and actors; actors and systems and systems.

Context Diagram

Level 1 DFD (ADMIN)
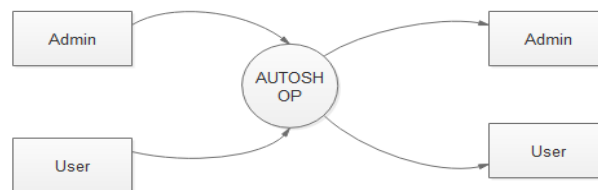
Level -2 DFD (Admin)
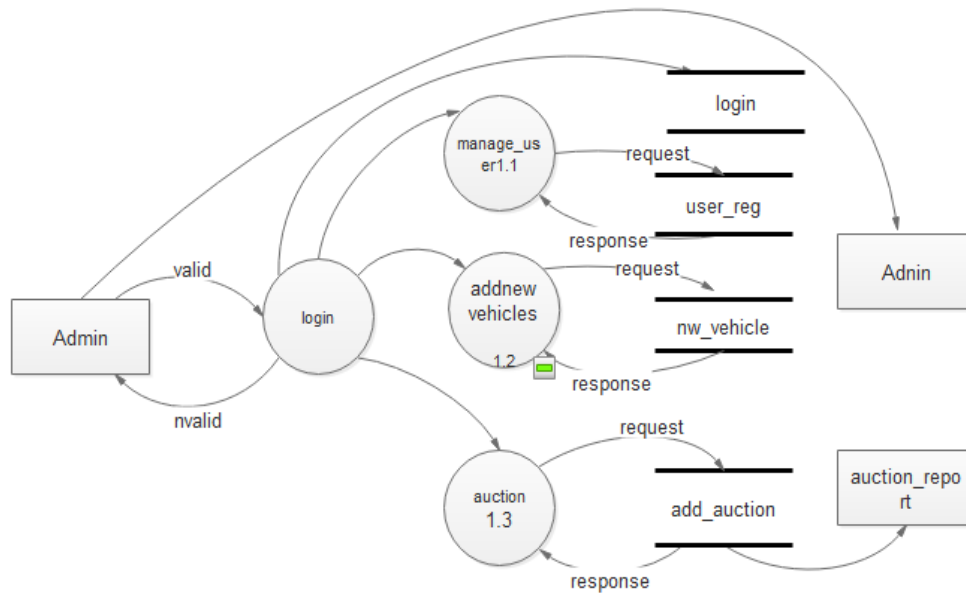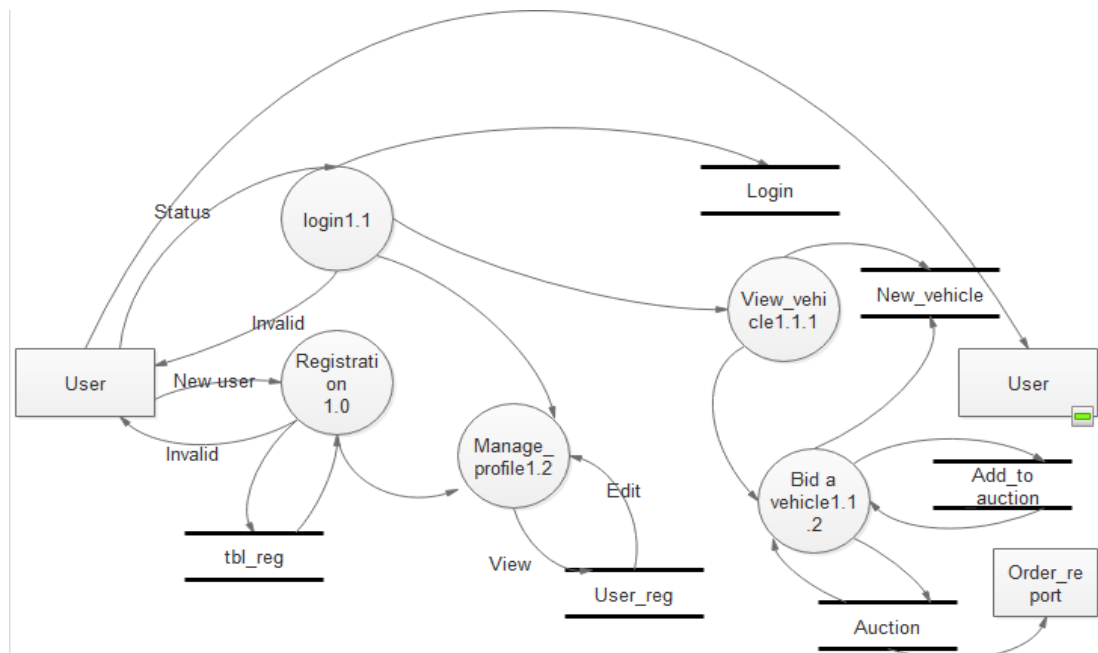

LEVEL 1 DFD (User)

4.5 UML DIAGRAM

A use case diagram is a graphic depiction of the interactions among the elements of a system. A use case is a methodology used in system analysis to identify, clarify, and organize system requirements. In this context, the term "system" refers to something being developed or operated, such as a mail-order product sales and service Web site. Use case diagrams are employed in UML (Unified Modeling Language), a standard notation for the modeling of realworld objects and systems.

System objectives can include planning overall requirements, validating a hardware design, testing and debugging a software product under development, creating an online help reference, or performing a consumer-service-oriented task.

For example, use cases in a product sales environment would include item ordering, catalog updating, payment processing, and customer relations. A use case diagram contains four components.

- The boundary, which defines the system of interest in relation to the world around it.
- The actors, usually individuals involved with the system defined according to their roles.
- The use cases, which are the specific roles played by the actors within and around the system.
- The relationships between and among the actors and the use cases

Use Case Diagram
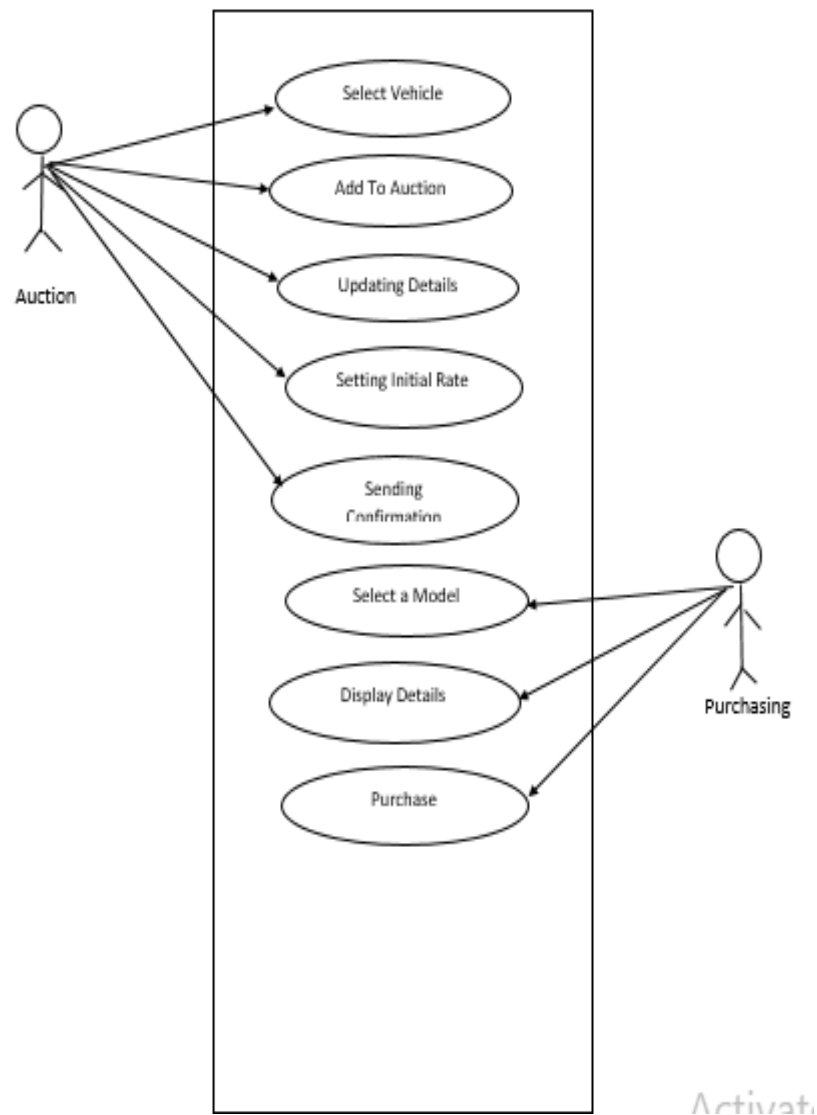
4.5.1 USER INTERFACE DESIGN

Registration

Personal details

4.6 Database Design

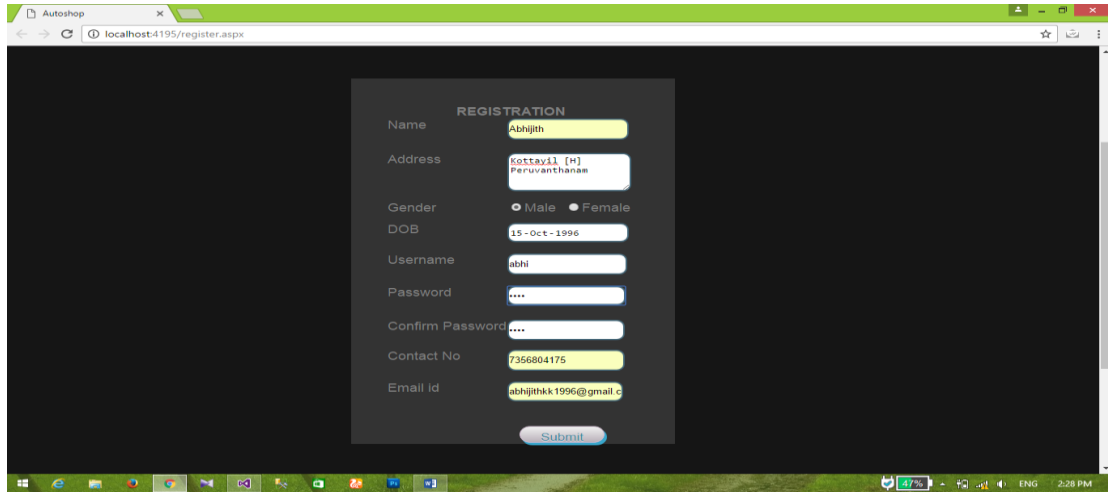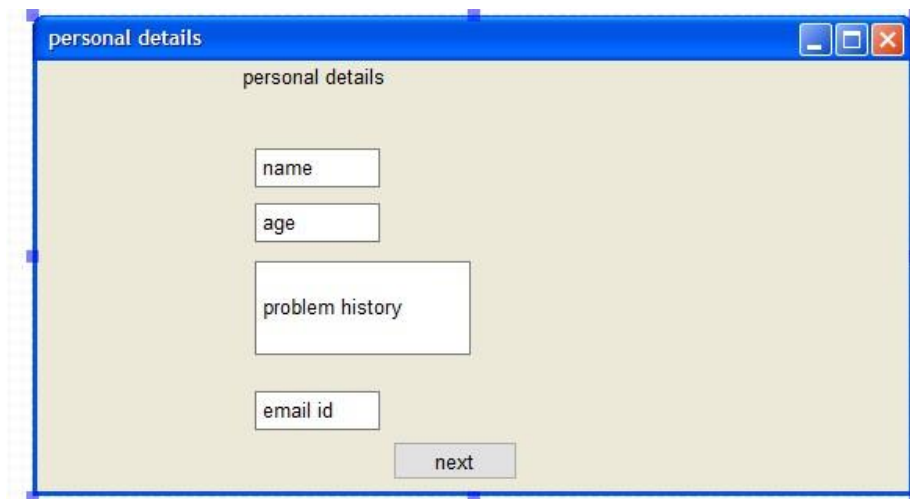A database is an organized mechanism that has the capability of storing information through which a user can retrieve stored information in an effective and efficient manner. The data is the purpose of any database and must be protected.

The database design is a two level process. In the first step, user requirements are gathered together and a database is designed which will meet these requirements as clearly as possible. This step is called Information Level Design and it is taken independent of any individual DBMS.

In the second step, this Information level design is transferred into a design for the specific DBMS that will be used to implement the system in question. This step is called Physical Level Design, concerned with the characteristics of the specific DBMS that will be used. A database design runs parallel with the system design. The organization of the data in the database is aimed to achieve the following two major objectives.

- Data Integrity
- Data independence

4.6.1 Relational Database Management System (RDBMS)

A relational model represents the database as a collection of relations. Each relation resembles a table of values or file of records. In formal relational model terminology, a row is called a tuple, a column header is called an attribute and the table is called a relation. A relational database consists of a collection of tables, each of which is assigned a unique name. A row in a tale represents a set of related values.

Relations, Domains & Attributes

A table is a relation. The rows in a table are called tuples. A tuple is an ordered set of n elements. Columns are referred to as attributes. Relationships have been set between every table in the database. This ensures both Referential and Entity Relationship Integrity. A domain D is a set of atomic values. A common method of specifying a domain is to specify a data type from which the data values forming the domain are drawn. It is also useful to specify a name for the domain to help in interpreting its values.

Every value in a relation is atomic, that is not decomposable.

Relationships

- Table relationships are established using Key. The two main keys of prime importance are Primary Key & Foreign Key. Entity Integrity and Referential Integrity Relationships can be established with these keys.

- Entity Integrity enforces that no Primary Key can have null values.

- Referential Integrity enforces that no Primary Key can have null values.

- Referential Integrity for each distinct Foreign Key value, there must exist a matching Primary Key value in the same domain. Other key are Super Key and Candidate Keys.

4.6.2 Normalization

Data are grouped together in the simplest way so that later changes can be made with minimum impact on data structures. Normalization is formal process of data structures in manners that eliminates redundancy and promotes integrity. Normalization is a technique of separating redundant fields and breaking up a large table into a smaller one. It is also used to avoid insertion, deletion, and updating anomalies. Normal form in data modelling use two concepts, keys and relationships. A key uniquely identifies a row in a table. There are two types of keys, primary key and foreign key. A primary key is an element or a combination of elements in a table whose purpose is to identify records from the same table. A foreign key is a column in a table that uniquely identifies record from a different table. All the tables have been normalized up to the third normal form.

As the name implies, it denotes putting things in the normal form. The application developer via normalization tries to achieve a sensible organization of data into proper tables and columns and where names can be easily correlated to the data by the user. Normalization eliminates repeating groups at data and thereby avoids data redundancy which proves to be a great burden on the computer resources. These include:

- Normalize the data.
- Choose proper names for the tables and columns. Choose the proper name for the data.

First Normal Form

The First Normal Form states that the domain of an attribute must include only atomic values and that the value of any attribute in a tuple must be a single value from the domain of that attribute. In other words 1NF disallows "relations within relations" or

"relations as attribute values within tuples". The only attribute values permitted by
1NF are single atomic or indivisible values. The first step is to put the data into First Normal
Form. This can be donor by moving data into separate tables where the data is of similar
type in each table. Each table is given a Primary Key or Foreign Key as per requirement of
the project. In this we form new relations for each non-atomic attribute or nested relation.
This eliminated repeating groups of data.A relation is said to be in first normal form if only
if it satisfies the constraints that contain the primary key only.

TABLES

1.**Registration**

Table no : 1

Table name : registration

Primary key :user_id

Table Description : To store User details

COLUMN NAME	DATA TYPE	DESCRIPTION
User_id	Varchar	Primary Key
Lid	Int	Foreign Key
Fname	Varchar	First Name
Mname	Varchar	Middle Name
Lname	Varchar	Last Name
Addr	Varchar	Address
Gender	Varchar	Gender
Dob	Varchar	Date of birth
Contact_no	Numeric	Contact no
Email_id	Varchar	Email-id

2.Login

Table no : 2

Table Name:-login_autoshop

Primary key :user_id

COLUMN NAME	DATA TYPE	DESCRIPTION
Lid	Int	Primary Key
Username	Varchar	User Name
Pass	Varchar	Password
Rol	Varchar	Role
Status	Varchar	Status

3.AUCTION

Table No:-3
Table Name:-auction
Primary Key:-auction_id

COLUMN NAME	DATA TYPE	DESCRIPTION
Auction_id	Int	Primary Key
User_id	Int	User_id
Bid_amt	Numeric	Bd_amount
Bid_date	Varchar	Bid_date
Vehicleid	Int	Vehicle_id

4.NEWVEHICLE

Table No:-4

Table Name:-addvehicle

Primary Key:-vehicleid

COLUMN NAME	DATA TYPE	DESCRIPTION
Vehicleid	Int	Primary Key
Vname	Varchar	Vehicle Name
Bprice	Numeric	Basic Price
Description	Varchar	Description
Model	Varchar	Model
Color	Varchar	Color
Fueltype	Varchar	Fueltype
Year	Numeric	Year
Mileage	Numeric	Mileage
Transmission	Varchar	Transmission
Registered	Varchar	Registered
Status	Varchar	Status

5.ADD_TO_AUCTION

Table No:-5

Table Name:-addtoauction

Primary Key:-aid

COLUMN NAME	DATA TYPE	DESCRIPTION
Aid	Int	Primary Key Auction Id
Adding_date	Varchar	Adding Date
B_price	Numeric	Basic_price
Last_date	Varchar	Last Date
Bid_rate	Int	Bid rate
Vehicleid	Int	vehicle_id

CHAPTER 5
TESTING

5.1 INTRODUCTION

No program or system design is perfect; communication between the user and the designer is not always complete or clear, and time is usually short. Inadequate testing or absence of testing, will lead to errors that may not appear, until months later. Thus, testing is vital to the success of the system. Therefore, Autoshop also has undergone testing. System testing makes the logical assumption that if all the parts of the system are correct then the goal will be successfully achieved.

5.2 TEST PLAN

A test plan implies a series of desired course of action to be followed in accomplishing various testing methods. The Test Plan acts as a blue print for the action that is to be followed. The software engineers create a computer program, its documentation and related data structures. The software developers is always responsible for testing the individual units of the programs, ensuring that each performs the function for which it was designed. There is an independent test group (ITG) which is to remove the inherent problems associated with letting the builder to test the thing that has been built. The specific objectives of testing should be stated in measurable terms. So that the mean time to failure, the cost to find and fix the defects, remaining defect density or frequency of occurrence and test work-hours per regression test all should be stated within the test plan. The levels of testing include:

❖ Unit testing
❖ Integration Testing
❖ Data validation Testing
❖ Output Testing

5.2.1 Unit Testing

Unit testing focuses verification effort on the smallest unit of software design – the software component or module. Using the component level design description as a guide, important control paths are tested to uncover errors within the boundary of the module.

The relative complexity of tests and uncovered scope established for unit testing. The unit testing is white-box oriented, and step can be conducted in parallel for multiple components. The modular interface is tested to ensure that information properly flows into and out of the program unit under test. The local data structure is examined to ensure that data stored temporarily maintains its integrity during all steps in an algorithm's execution. Boundary conditions are tested to ensure that all statements in a module have been executed at least once.

Finally, all error handling paths are tested.

Tests of data flow across a module interface are required before any other test is initiated. If data do not enter and exit properly, all other tests are moot. Selective testing of execution paths is an essential task during the unit test. Good design dictates that error conditions be anticipated and error handling paths set up to reroute or cleanly terminate processing when an error does occur. Boundary testing is the last task of unit testing step. Software often fails at its boundaries.

5.2.2 Integration Testing

Integration testing is systematic technique for constructing the program structure while at the same time conducting tests to uncover errors associated with interfacing. The objective is to take unit tested components and build a program structure that has been dictated by design. The entire program is tested as whole. Correction is difficult because isolation of causes is complicated by vast expanse of entire program. Once these errors are corrected, new ones appear and the process continues in a seemingly endless loop.

After performing unit testing in the System all the modules were integrated to test for any inconsistencies in the interfaces. Moreover differences in program structures were removed and a unique program structure was evolved.

5.2.3 Validation Testing or System Testing

This is the final step in testing. In this the entire system was tested as a whole with all forms, code, modules and class modules. This form of testing is popularly known as Black Box testing or System tests.

Black Box testing method focuses on the functional requirements of the software. That is, Black Box testing enables the software engineer to derive sets of input conditions that will fully exercise all functional requirements for a program.

Black Box testing attempts to find errors in the following categories; incorrect or missing functions, interface errors, errors in data structures or external data access, performance errors and initialization errors and termination errors.

5.2.4 Output Testing or User Acceptance Testing

Unit testing was done in Sell-Soft System by treating each module as separate entity and testing each one of them with a wide spectrum of test inputs. Some flaws in the internal logic of the modules were found and were rectified. After coding each module is tested and run individually. All unnecessary code where removed and ensured that all modules are working, and gives the expected result.

The system considered is tested for user acceptance; here it should satisfy the firm's need. The software should keep in touch with perspective system; user at the time of developing and making changes whenever required. This done with respect to the following points:

> ➢ Input Screen Designs,
> ➢ Output Screen Designs,

The above testing is done taking various kinds of test data. Preparation of test data plays a vital role in the system testing. After preparing the test data, the system under study is tested using that test data. While testing the system by which test data errors are again uncovered and corrected by using above testing steps and corrections are also noted for future use.

CHAPTER 6
IMPLEMENTATION

6.1 INTRODUCTION

Implementation is the stage of the project where the theoretical design is turned into a working system. It can be considered to be the most crucial stage in achieving a successful new system gaining the users confidence that the new system will work and will be effective and accurate. It is primarily concerned with user training and documentation. Conversion usually takes place about the same time the user is being trained or later. Implementation simply means convening a new system design into operation, which is the process of converting a new revised system design into an operational one.

At this stage the main work load, the greatest upheaval and the major impact on the existing system shifts to the user department. If the implementation is not carefully planned or controlled, it can create chaos and confusion.

Implementation includes all those activities that take place to convert from the existing system to the new system. The new system may be a totally new, replacing an existing manual or automated system or it may be a modification to an existing system. Proper implementation is essential to provide a reliable system to meet organization requirements. The process of putting the developed system in actual use is called system implementation. This includes all those activities that take place to convert from the old system to the new system. The system can be implemented only after through testing is done and if it is found to be working according to the specifications. The system personnel check the feasibility of the system. The more complex the system being implemented, the more involved will be the system analysis and design effort required to implement the three main aspects: education and training, system testing and changeover.

The implementation state involves the following tasks:

- Careful planning.
- Investigation of system and constraints.
- Design of methods to achieve the changeover.
- Training of the staff in the changeover phase.

6.2 IMPLEMENTATION PROCEDURES

Implementation of software refers to the final installation of the package in its real environment, to the satisfaction of the intended uses and the operation of the system. In many organizations someone who will not be operating it, will commission the software development project. In the initial stage people doubt about the software but we have to ensure that the resistance does not build up, as one has to make sure that:

o The active user must be aware of the benefits of using the new system.
o Their confidence in the software is built up.

Proper guidance is imparted to the user so that he is comfortable in using the application. Before going ahead and viewing the system, the user must know that for viewing the result, the server program should be running in the server. If the server object is not up running on the server, the actual process won't take place.

6.2.1 User Training

User training is designed to prepare the user for testing and converting the system. To achieve the objective and benefits expected from compute based system, it is essential for the people who will be involved to be confident of their role in the new system. As system becomes more complex, the need for training is more important. By user training the user comes to know how to enter data, respond to error messages,interrogate the database and call up routine that will produce reports and perform other necessary functions.

6.2.2 Training on the Application Software

After providing the necessary basic training on computer awareness the user will have to be trained on the new application software. This will give the underlying philosophy of the use of the new system such as the screen flow, screen design type of help on the screen, type of errors while entering the data, the corresponding validation check at each entry and the ways to correct the date entered. It should then cover information needed by the specific user/ group to use the system or part of the system while imparting the training of the program on the application. This training may be different across different user groups and across different levels of hierarchy.

6.2.3 System Maintenance

Maintenance is the enigma of system development. The maintenance phase of the software cycle is the time in which a software product performs useful work. After a system is successfully implemented, it should be maintained in a proper manner. System maintenance is an important aspect in the software development life cycle. The need for system maintenance is for it to make adaptable to the changes in the system environment. Software maintenance is of course, far more than "Finding Mistakes".

CHAPTER 7
CONCLUSION
AND
FUTURESCOE

7.1 CONCLUSION

The project is an online Automobile shopping system that allows web users to purchase used cars online without visiting any physical location. This system allows the user to purchase Automobile online and Bid cars in Auction. Thus, Autoshop provides all the basic functionalities to a user who would like to purchase used cars in online.

7.2 FUTURE SCOPE

The system is designed to meet the modifications required in the future. So most of the required modifications can be done without much re-work.

       □ Suggestion of suitable hearing aids according to the hearing loss.

       □ Online purchase

CHAPTER 8
BIBLIOGRAPHY

BOOKS/REFERENCES:

- Gary B. Shelly, Harry J. Rosenblatt, "*System Analysis and Design*", 2009.
- Roger S Pressman, "*Software Engineering*", 1994.
- PankajJalote, "S*oftware engineering*: a precise approach", 2006.
- James lee and Brent ware Addison, "Open source web development with LAMP", 2003
- IEEE Std 1016 Recommended Practice for Software Design Descriptions.

WEBSITES:

- www.w3schools.com
- www.jquery.com
- http://homepages.dcc.ufmg.br/~rodolfo/es-1-03/IEEE-Std-830-1998.pdf
  www.agilemodeling.com/artifacts/useCaseDiagram.htm

CHAPTER 9
APPENDIX

9.1 SAMPLE CODE

1.LOGIN FORM CODE

```php
<?php

if(isset($_SESSION['lid'])){

header("location:../adminhome1.php");

}



require "connect.php";

if(isset($_POST["submit"]))

{

$username=$_POST['username'];

$password=$_POST['password'];

//$password = md5($password);

echo $sql="select status from login_autoshop where uname='$username' and password='$password'";

  $res=mysqli_query($conn,$sql) or die("eror");

   while($fetch=mysqli_fetch_array($res))
```

```php
  {

   if($fetch['status']==0)

     {

     $_SESSION["lid"]=$fetch['lid'];

   echo 'rrrrr';

      // header("location:../adminhome1.php");

      }



    if($fetch['status']==1)

     {

     $_SESSION["lid"] =$fetch['lid'];

      header("location:userhome1.php");

   }

   }

}

?>

<!DOCTYPE html>

<html lang="en">

<head>
```

```
<title>Login</title>

<meta charset="UTF-8">

<meta name="viewport" content="width=device-width, initial-scale=1">

<!--
===================================================================
=============================-->

<link rel="icon" type="image/png" href="images/icons/favicon.ico"/>

<!--
===================================================================
=============================-->

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

<!--
===================================================================
=============================-->

<link rel="stylesheet" type="text/css" href="fonts/font-awesome-4.7.0/css/font-awesome.min.css">

<!--
===================================================================
=============================-->

<link rel="stylesheet" type="text/css" href="fonts/iconic/css/material-design-iconic-font.min.css">

<!--
===================================================================
=============================-->

<link rel="stylesheet" type="text/css" href="vendor/animate/animate.css">
```

```html
<!--
===================================================================
=============================-->

<link rel="stylesheet" type="text/css" href="vendor/css-hamburgers/hamburgers.min.css">

<!--
===================================================================
=============================-->

<link rel="stylesheet" type="text/css" href="vendor/animsition/css/animsition.min.css">

<!--
===================================================================
============================-->

<link rel="stylesheet" type="text/css" href="vendor/select2/select2.min.css">

<!--
===================================================================
============================-->

<link rel="stylesheet" type="text/css" href="vendor/daterangepicker/daterangepicker.css">

<!--
===================================================================
=============================-->

<link rel="stylesheet" type="text/css" href="css/util.css">

<link rel="stylesheet" type="text/css" href="css/main.css">

<!--
===================================================================
============================-->

</head>

<body>
```

```
<div class="limiter">

<div class="container-login100" style="background-image: url('images/bg-01.jpg');">

<div class="wrap-login100 p-l-55 p-r-55 p-t-65 p-b-54">

<form class="login100-form validate-form" method="post">

<span class="login100-form-title p-b-49">

Login

</span>


<div class="wrap-input100 validate-input m-b-23" data-validate = "Username is reauired">

  <span class="label-input100">Username</span>

  <input class="input100" type="text" name="username" placeholder="Type your username">

  <span class="focus-input100" data-symbol="&#xf206;"></span>

  </div>


  <div class="wrap-input100 validate-input" data-validate="Password is required">

  <span class="label-input100">Password</span>

  <input class="input100" type="password" name="password" placeholder="Type your
password">

  <span class="focus-input100" data-symbol="&#xf190;"></span>
```

```
</div>



<div class="text-right p-t-8 p-b-31">

  <a href="#">

  Forgot password?

  </a>

  </div>



<div class="container-login100-form-btn">

  <div class="wrap-login100-form-btn">

  <div class="login100-form-bgbtn"></div>

  <button class="login100-form-btn">

  Login

  </button>

  </div>

  </div>



  <div class="txt1 text-center p-t-54 p-b-20">
```


Or Sign Up Using

</div>

<div class="flex-c-m">

<i class="fa fa-facebook"></i>

<i class="fa fa-twitter"></i>

<i class="fa fa-google"></i>

</div>

```
<div class="flex-col-c p-t-155">

<span class="txt1 p-b-17">

Or Sign Up Using

</span>


<a href="#" class="txt2">

Sign Up

</a>

</div>

</form>

</div>

</div>

</div>



<div id="dropDownSelect1"></div>



<!--
================================================================
=============================-->
```

```html
<script src="vendor/jquery/jquery-3.2.1.min.js"></script>

<!--
===================================================================
===============================-->

<script src="vendor/animsition/js/animsition.min.js"></script>

<!--
===================================================================
===============================-->

<script src="vendor/bootstrap/js/popper.js"></script>

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

<!--
===================================================================
===============================-->

<script src="vendor/select2/select2.min.js"></script>

<!--
===================================================================
===============================-->

<script src="vendor/daterangepicker/moment.min.js"></script>

<script src="vendor/daterangepicker/daterangepicker.js"></script>

<!--
===================================================================
===============================-->

<script src="vendor/countdowntime/countdowntime.js"></script>

<!--
===================================================================
===============================-->
```

```php
<script src="js/main.js"></script>
```

```html
</body>

</html>
```

2.ADD VEHICLE FORM CODE

```php
<?php

require "connect.php";

if(isset($_POST["submit"]))

{

$vehicle=$_POST["vehicle"];

$basic_price=$_POST["basic_price"];


$description=$_POST["description"];

$model=$_POST["model"];

$color=$_POST["color"];

$fuel=$_POST["fuel"];

$registred=$_POST["registred"];

$year=$_POST["year"];

$milage=$_POST["milage"];
```

```php
$transmission=$_POST["transmission"];

$target_dir = "uploads/";

$target_file = $target_dir . basename($_FILES["image"]["name"]);

move_uploaded_file($_FILES["image"]["tmp_name"], $target_file);$image=$target_file;

echo$suc=mysqli_query($conn,"insertintoaddvehicle(vehicle,basic_price,image,description,
model,color,fuel,registred,year,milage,transmission)
values('$vehicle','$basic_price','$target_file','$description','$model','$color','$fuel','$registred
','$year','$milage','$transmission')") or die (mysqli_error());

}

?>
```

```html
<html>

<head>

<link href="css\style1.css" rel="stylesheet">

</head>

<body background="images/123.PNG">

<div id="main">

<center>

<div id="top"><b>ADD VEHICLES</b></div>

<div id="menu"><center><a href="adminhome1.php">HOME &nbsp&nbsp </a>

</a><a href="addvehicles.php">ADD CARS &nbsp&nbsp</a>

</a><a href="login_autosh.php">CONFIRM &nbsp&nbsp</a>
```

CUSTOMERS

SALES

PROFILE

LOGOUT

<form action="#" method="post" enctype="multipart/form-data">

<div id="signup">

<h2 class="smp">UPLOAD DETAILS<h2>

<center>

<font size="4px">

<label>Vehicles:</label>

<input id="vehicle" name="vehicle"   placeholder="Vehicles" type="text" required pattern="^[a-zA-Z0-9]+$">

<label>Basic Price:</label>

<input id="basic_price" name="basic_price" placeholder="Basic Price" type="number" pattern="\d+(\.\d{2})?" required>

<label>Upload Image:
</label>

```
<input type="file" id ="image" name="image" accept="image/*" placeholder="Upload
Vehicles Image"><br><bR>
```

```
<label>Description:</label><br>
```

```
<textarea placeholder="description" name="description" required></textarea><br>
```

```
<h2 class="smp">FEATURES<h2>
```

```
<label>Models:</label>
```

```
<input id="model" name="model"  placeholder="Model" type="text" required><br>
```

```
<label>Color:</label>
```

```
<input id="color" name="color" pattern="^[a-zA-Z0-9]+$" placeholder="Color" type="text"
required><br>
```

```
<label>Fuel Type:</label>
```

```
<select id="fuel" name="fuel"  pattern="^[a-zA-Z0-9]+$" placeholder="Fuel" type="text"
required>
```

```
<option value="Petrol">Petrol</option>
```

```
<option value="Diesel">Diesel</option>
```

```
</select><br>
```

```
<label>Registerd:</label><br>
```

```
Yes<input  id="registred"  name="registred"      placeholder="Registred"  type="radio"
value="yes" required><br>
```

```
No<input  id=""  name="registred"    placeholder="Registred"  type="radio"  value="no"
required><br>
```

```
<label>Year:</label>

<input id="year" name="year" pattern="[0-9]{4}" placeholder="Year" type="text"
required><br>

<label>Milage:</label>

<input id="milage" name="milage" placeholder="Milage" type="text" required><br>

<label>Transmission:</label>

<input id="transmission" name="transmission"  placeholder="Transmission" type="text"
required><br>

<br><br>

<input  name="submit" type="submit" value=" SUBMIT ">

<br>

</font>

</center>

</div>

</form>

<body>

</html>
```
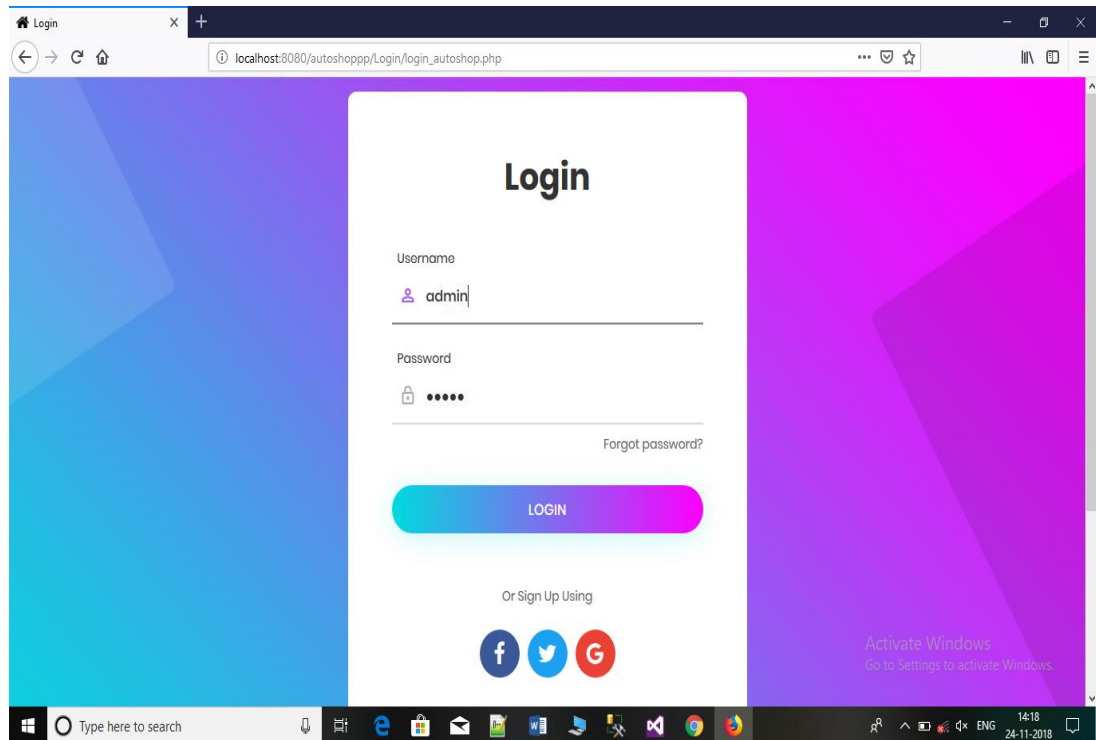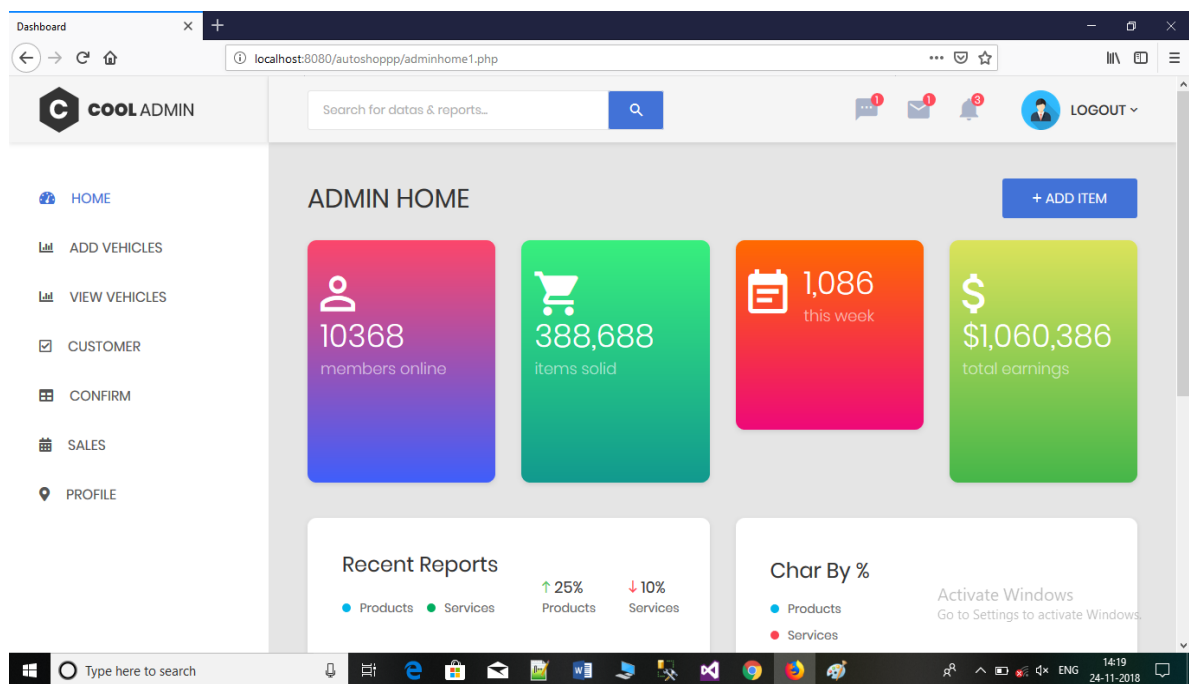
10.2 SAMPLE SCREEN

Home Page

Home Page

Login Page

Admin Home

Add Vehicles

View Customer

User Home

View Vehicle & Buy