

INTRODUCTION

Objective

The shopping cart is mainly useful for who have not time to go to shopping, those are just entered into this website and bought what ever they want.

Even it is night or morning they entered into this site, and chosen different items like fruits, books, toys etc..

‘Customer is our god’ mainly this website is based on this formula. After chosen items he bought into Pay pal process like VISA or MASTER credit cards or any Debit cards are accepted in this website. Customer is happily shopping at his rest place.

Project Overview

Once customer entered with his own username and password, at that time automatically one shopping cart will be created, once user select an item it will add to cart. In case user think the selected item is not useful for me, then deleted that item from shopping cart.

Customer selected some items, but in his credit or debit cart haven’t that much balance, then he was logout from the website, the selected items are stored at cart with specific users with his allotted carts, after some days he bought those items then automatically deleted from the cart.

ANALYSIS

System analysis

1. Existing System

. Existing system is a manual one in which users are maintaining books to store the information like product details, purchases, sales details and accounts for every month. It is very difficult to maintain historical data.

Disadvantages

The following are the disadvantages of the existing system

- It is difficult to maintain important information in books.
- More manual hours need to generate required reports.
- It is tedious to manage historical data which needs much space to keep all the previous years' ledgers, books etc.
- Daily sales and purchases details must be entered into books are very difficult to maintain

2.Objective of the System

The objective of the Shopping Cart is to provide better information for the users of this system for better results for their maintainence in the product details that is sales, purchases and stock.

System Specifications

Hardware Requirements:-

- Dual Core(Processor).
- 256 MB Ram
- 512 KB Cache Memory
- Hard disk 10 GB
- Microsoft Compatible 101 or more Key Board

Software Requirements: -

Technology Implemented : Apache Server

Language Used : PHP 5.62 (Developed in Core PHP)

Database : My SQL

User Interface Design : HTML, AJAX,JQUERY,JAVASCRIPT

Web Browser : Mozilla, Google Chrome, IE8,OPERA

Software : XAMPP Server

DESIGN

Introduction

Design is the first step in the development phase for any techniques and principles for the purpose of defining a device, a process or system in sufficient detail to permit its physical realization.

Once the software requirements have been analyzed and specified the software design involves three technical activities - design, coding, implementation and testing that are required to build and verify the software.

The design activities are of main importance in this phase, because in this activity, decisions ultimately affecting the success of the software implementation and its ease of maintenance are made. These decisions have the final bearing upon reliability and maintainability of the system. Design is the only way to accurately translate the customer's requirements into finished software or a system.

Design is the place where quality is fostered in development. Software design is a process through which requirements are translated into a representation of software. Software design is conducted in two steps. Preliminary design is concerned with the transformation of requirements into data.

UML Diagrams:

Actor:

A coherent set of roles that users of use cases play when interacting with the use cases.



Use case:

A description of sequence of actions, including variants, that a system performs that yields an observable result of value of an actor.



UML stands for Unified Modeling Language. UML is a language for specifying, visualizing and documenting the system. This is the step while developing any product after analysis. The goal from this is to produce a model of the entities involved in the project which later need to be built. The representation of the entities that are to be used in the product being developed need to be designed.

There are various kinds of methods in software design:

They are as follows:

- Use case Diagram
- Sequence Diagram
- Collaboration Diagram
- Activity Diagram
- State chat Diagram

USECASE DIAGRAMS:

Use case diagrams model behavior within a system and helps the developers understand of what the user require. The stick man represents what's called an actor.

Use case diagram can be useful for getting an overall view of the system and clarifying who can do and more importantly what they can't do.

Use case diagram consists of use cases and actors and shows the interaction between the use case and actors.

- The purpose is to show the interactions between the use case and actor.
- To represent the system requirements from user's perspective.
- An actor could be the end-user of the system or an external system

A Use case is a description of set of sequence of actions. Graphically it is rendered as an ellipse with solid line including only its name. Use case diagram is a behavioral diagram that shows a set of use cases and actors and their relationship. It is an association between the use cases and actors. An actor represents a real-world object. Primary Actor – Sender, Secondary Actor Receiver

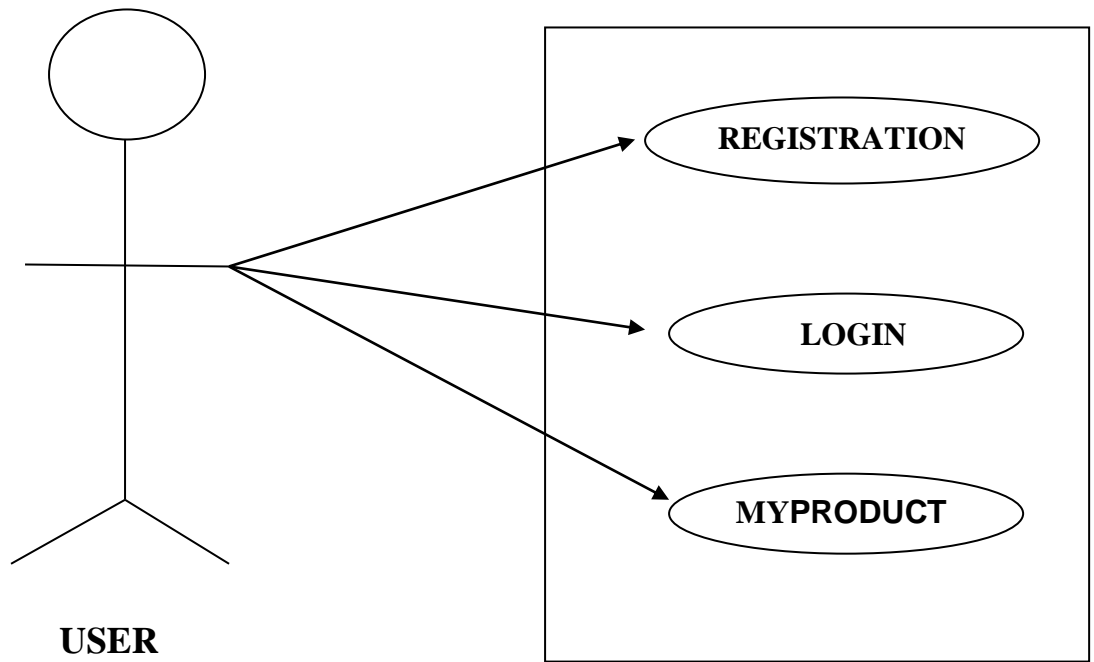


Figure 1:USECASE DIAGRAM FOR USER

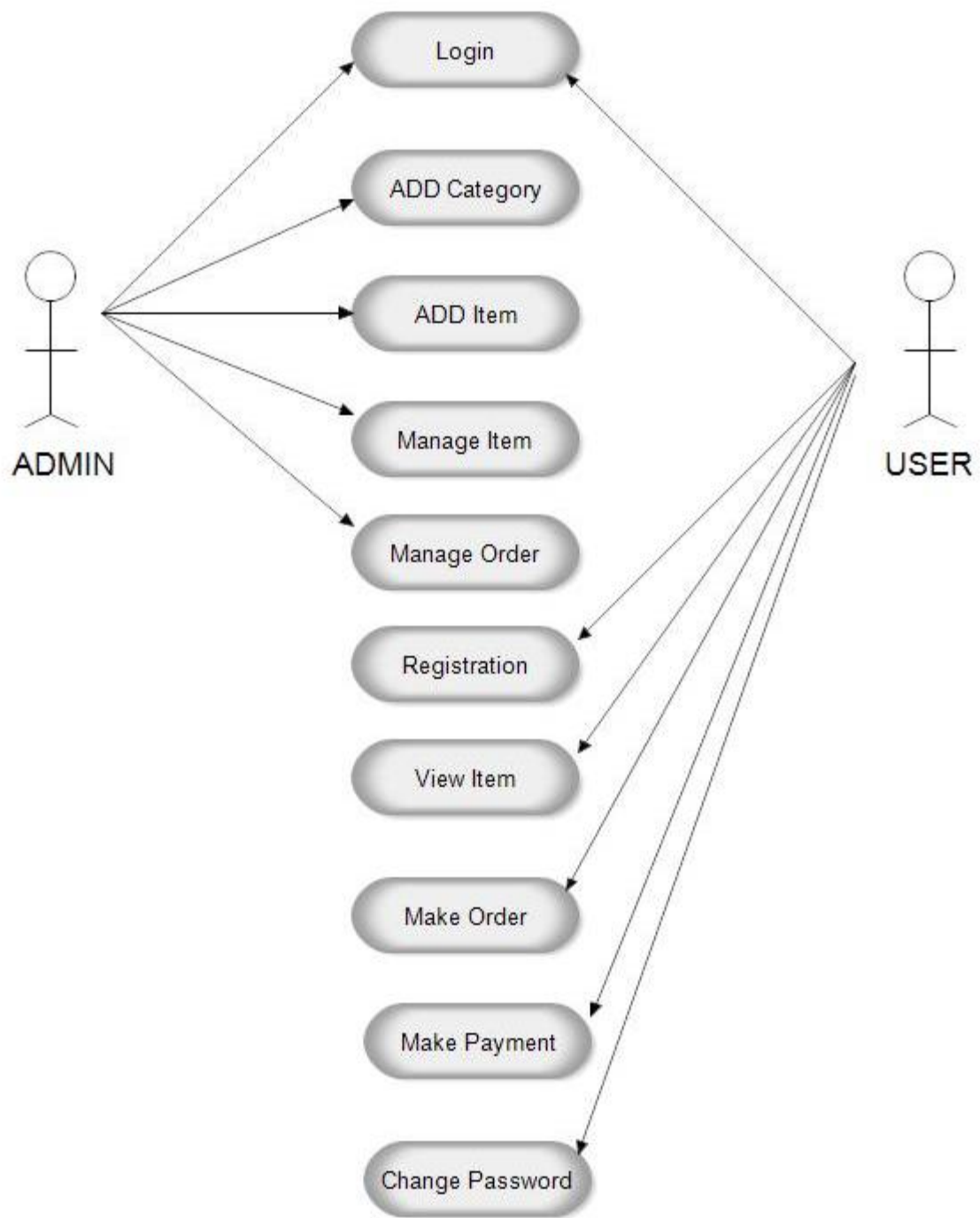


Figure 2. Use Case Diagram for User and admin

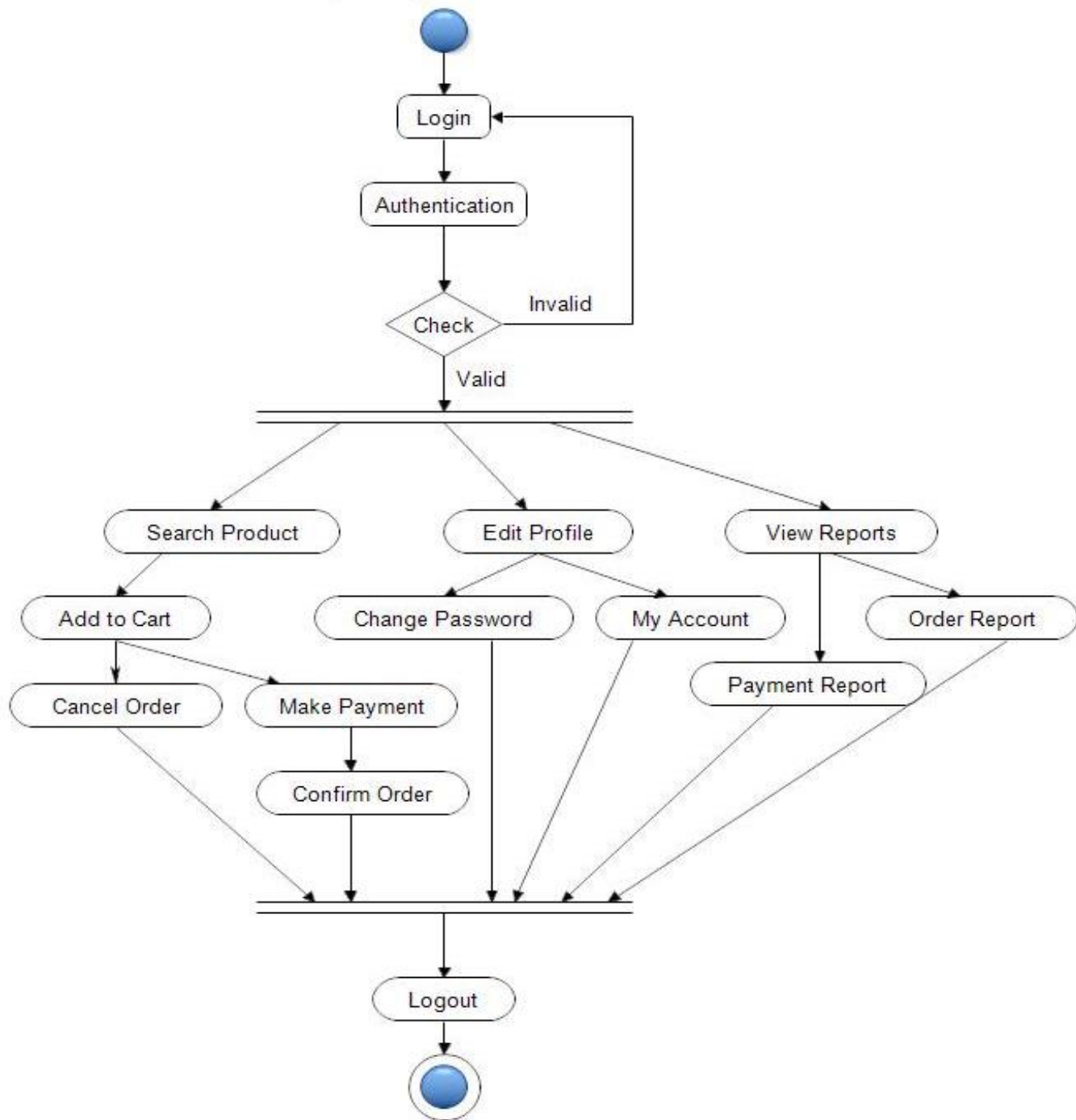


Figure 3 Activity diagram for user Side

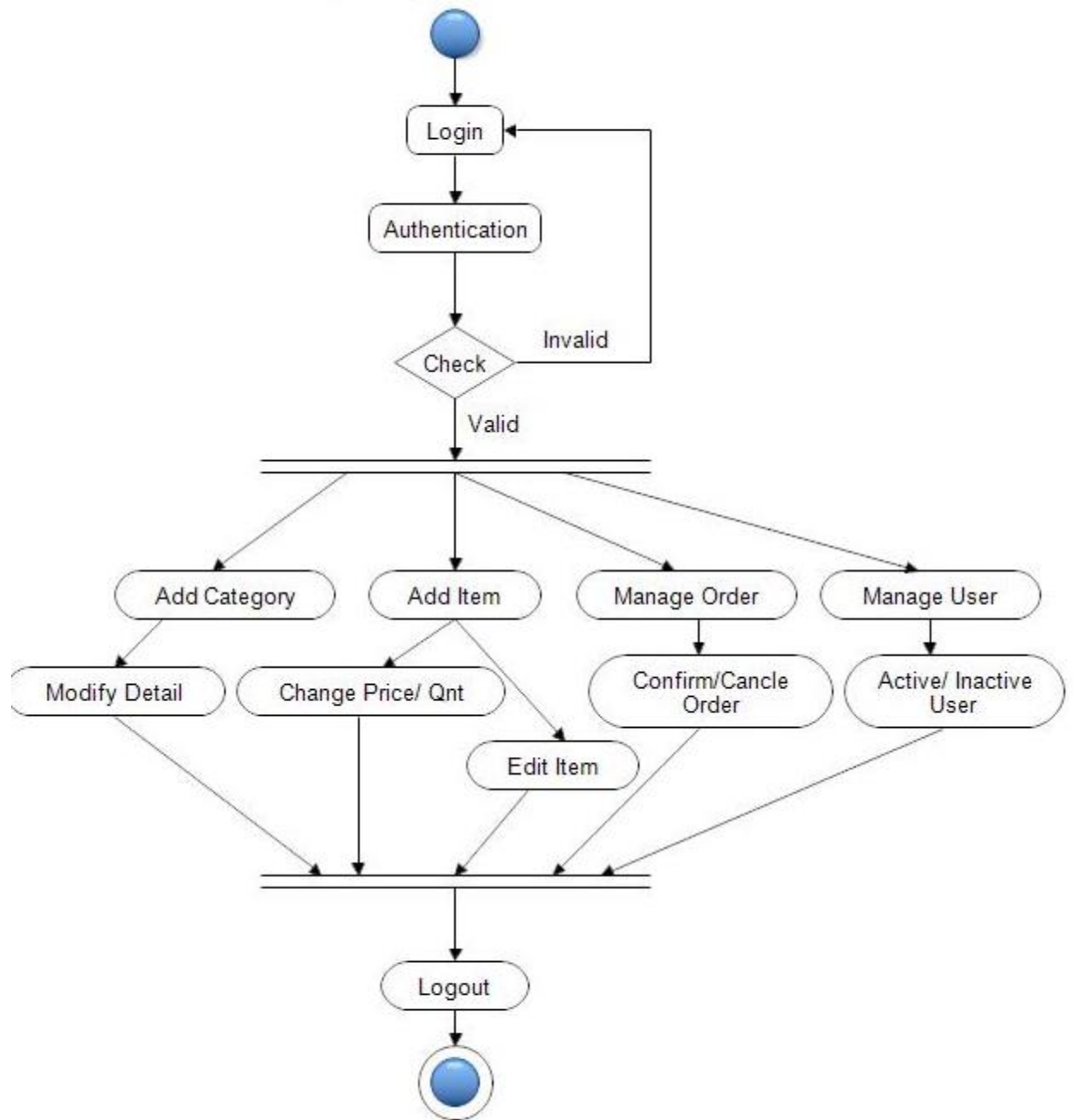


Figure 4 Activity Diagram for admin side

SEQUENCE DIAGRAM:

Sequence diagram and collaboration diagram are called INTERACTION DIAGRAMS. An interaction diagram shows an interaction, consisting of set of objects and their relationship including the messages that may be dispatched among them.

A sequence diagram is an introduction that empathizes the time ordering of messages. Graphically a sequence diagram is a table that shows objects arranged along the X-axis and messages ordered in increasing time along the Y-axis

Sequence Diagram

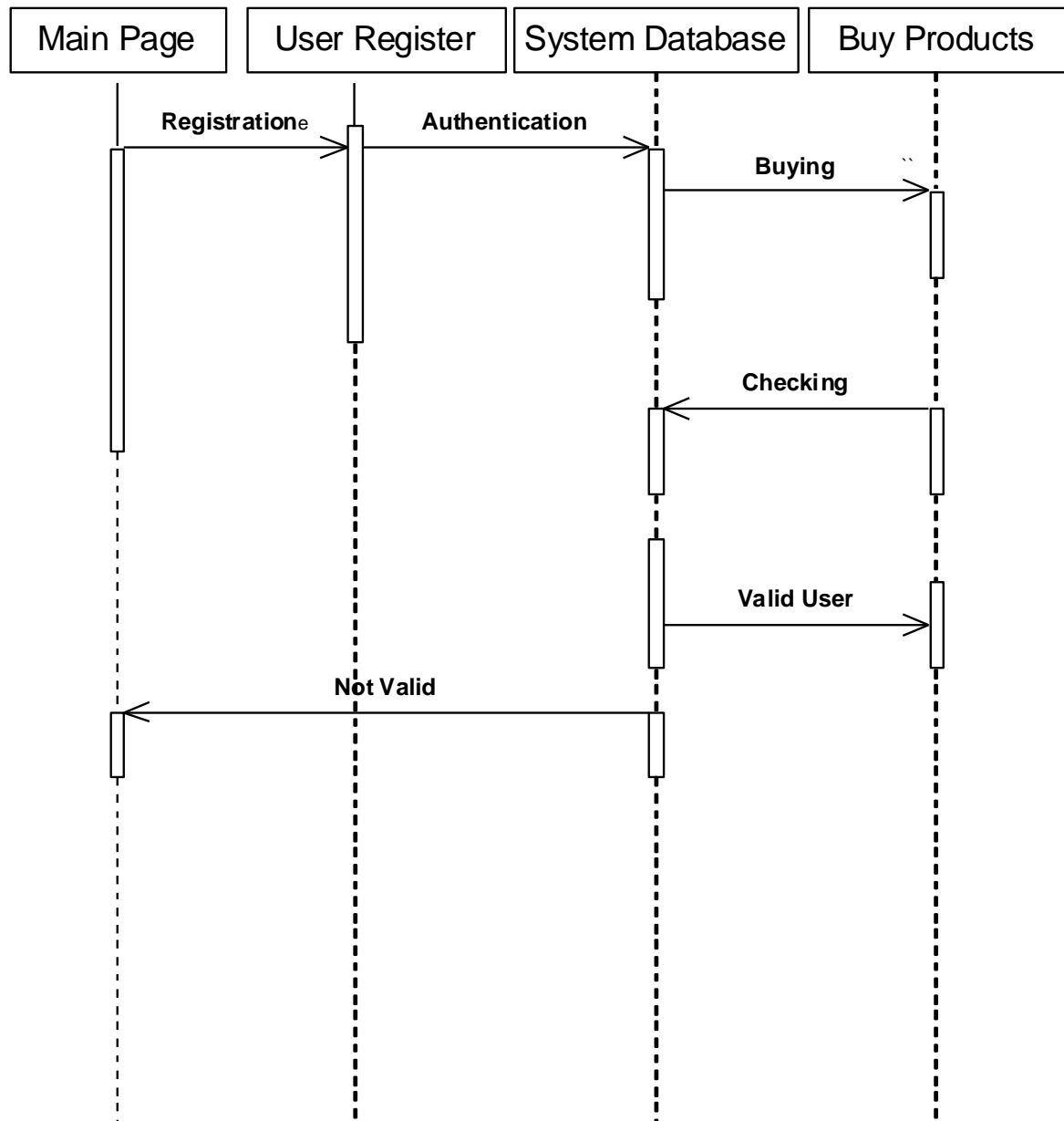


Figure 5 process

COLLABORATION DIAGRAM:

A collaboration diagram is an introduction diagram that emphasizes the structural organization of the objects that send and receive messages. Graphically a collaboration diagram is a collection of vertices and arcs.

Collaboration Diagram

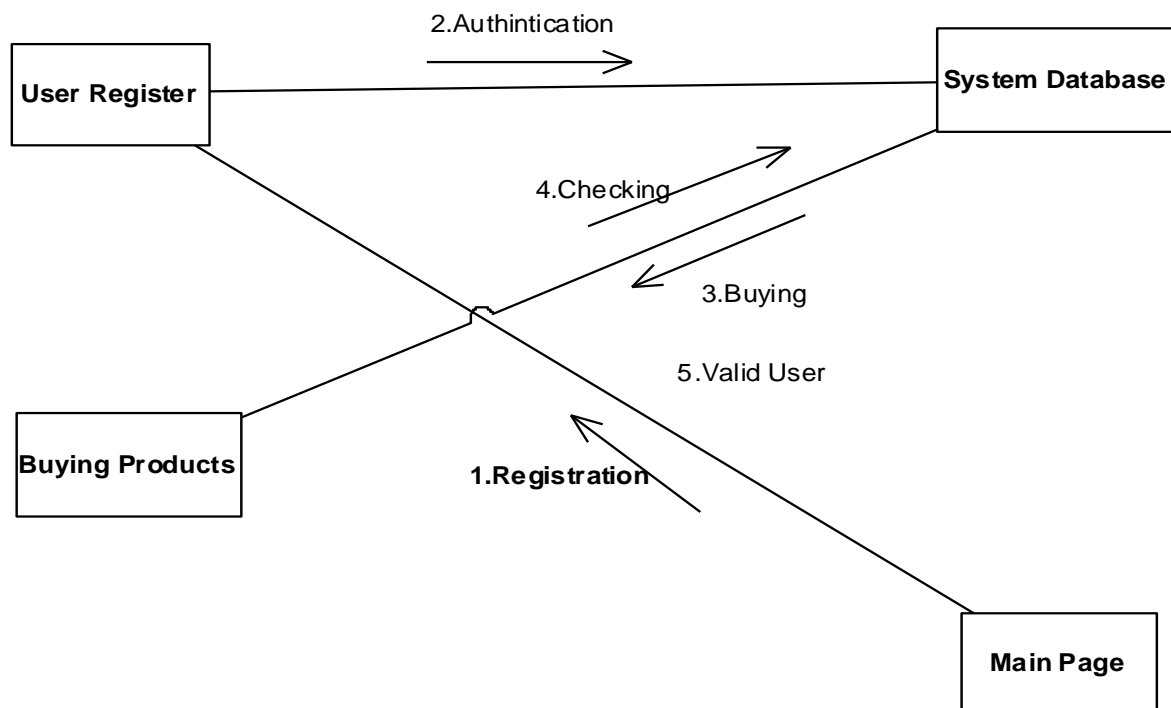


Figure .6 Collaboration Diagram

Class Diagram

Class is nothing but a structure that contains both variables and methods. The Class Diagram shows a set of classes, interfaces, and collaborations and their relating ships. There is most common diagram in modeling the object oriented systems and are used to give the static view of a system. It shows the dependency between the classes that can be used in our system.

The interactions between the modules or classes of our projects are shown below. Each block contains Class Name, Variables and Methods.

Class

A description of set of objects that share the same attributes, operations, relationships, and semantics

users

| Column | Type |
|-----------------|--------------|
| id (Primary) | int(11) |
| name | varchar(255) |
| email | varchar(255) |
| contactno | bigint(11) |
| password | varchar(255) |
| shippingAddress | longtext |
| shippingState | varchar(255) |
| shippingCity | varchar(255) |
| shippingPincode | int(11) |
| billingAddress | longtext |
| billingState | varchar(255) |
| billingCity | varchar(255) |
| billingPincode | int(11) |
| regDate | timestamp |
| updateDate | varchar(255) |

products

| Column | Type |
|----------------------------|--------------|
| id (Primary) | int(11) |
| category | int(11) |
| subCategory | int(11) |
| productName | varchar(255) |
| productCompany | varchar(255) |
| productPrice | int(11) |
| productPriceBeforeDiscount | int(11) |
| productDescription | longtext |
| productImage1 | varchar(255) |
| productImage2 | varchar(255) |
| productImage3 | varchar(255) |
| shippingCharge | int(11) |
| productAvailability | varchar(255) |
| postingDate | timestamp |
| updateDate | varchar(255) |

State Chart Diagram

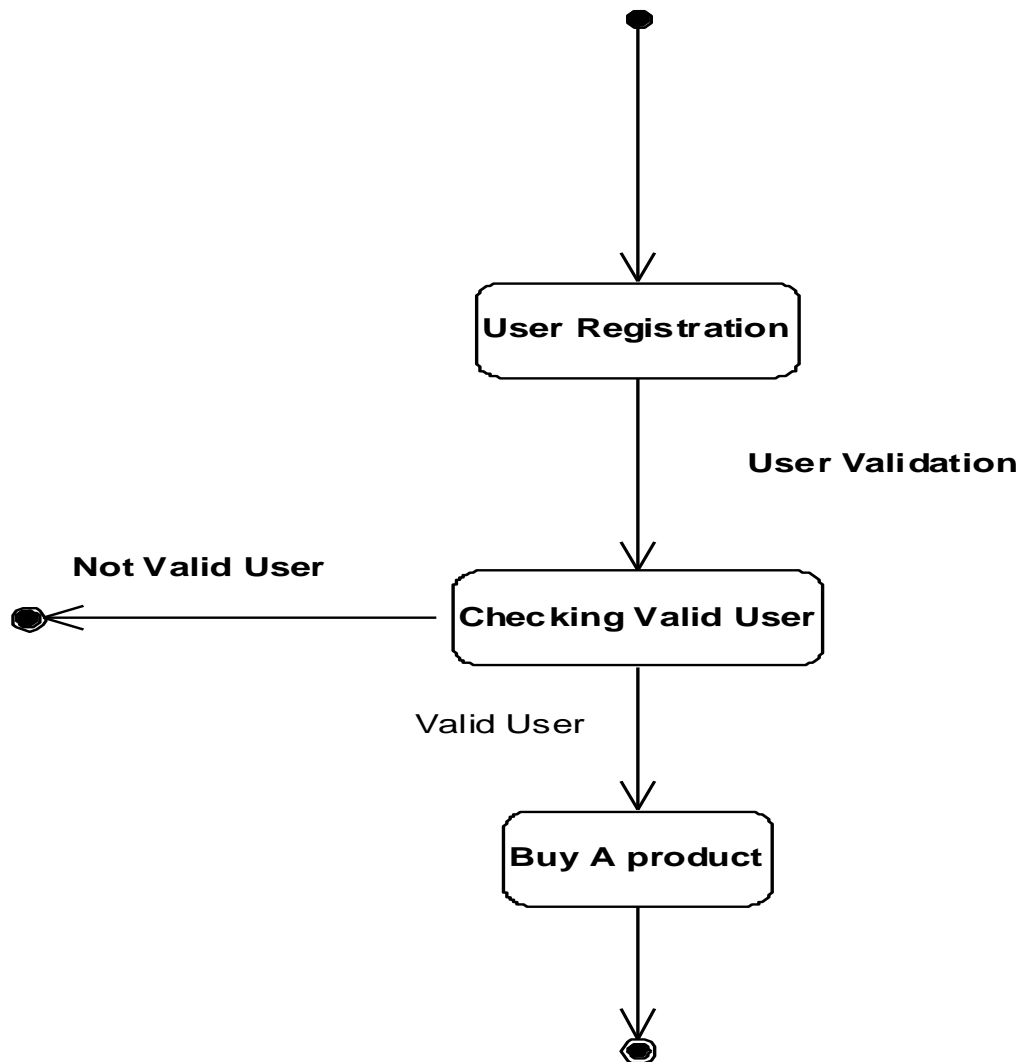


Figure.7 Validation Process

DATA FLOW DIAGRAMS:

The DFD takes an input-process-output view of a system i.e. data objects flow into the software, are transformed by processing elements, and resultant data objects flow out of the software.

Data objects represented by labeled arrows and transformation are represented by circles also called as bubbles. DFD is presented in a hierarchical fashion i.e. the first data flow model represents the system as a whole. Subsequent DFD refine the context diagram (level 0 DFD), providing increasing details with each subsequent level.

The DFD enables the software engineer to develop models of the information domain & functional domain at the same time. As the DFD is refined into greater levels of details, the analyst perform an implicit functional decomposition of the system. At the same time, the DFD refinement results in a corresponding refinement of the data as it moves through the process that embody the applications.

A context-level DFD for the system the primary external entities produce information for use by the system and consume information generated by the system. The labeled arrow represents data objects or object hierarchy.

Rule for DFD

- Fix the scope of the system by means of context diagrams.
- Organize the DFD so that the main sequence of the actions
- Reads left to right and top to bottom.
- Identify all inputs and outputs.
- Identify and label each process internal to the system with Rounded circles.
- A process is required for all the data transformation and Transfers. Therefore, never connect a data store to a data Source or the destinations or another data store with just a Data flow arrow.
- Do not indicate hardware and ignore control information.
- Make sure the names of the processes accurately convey everything the process is done.
- There must not be unnamed process.
- Indicate external sources and destinations of the data, with Squares.
- Number each occurrence of repeated external entities.
- Identify all data flows for each process step, except simple Record retrievals.
- Label data flow on each arrow.
- Use details flow on each arrow.
- Use the details flow arrow to indicate data movements.

Dataflow diagrams

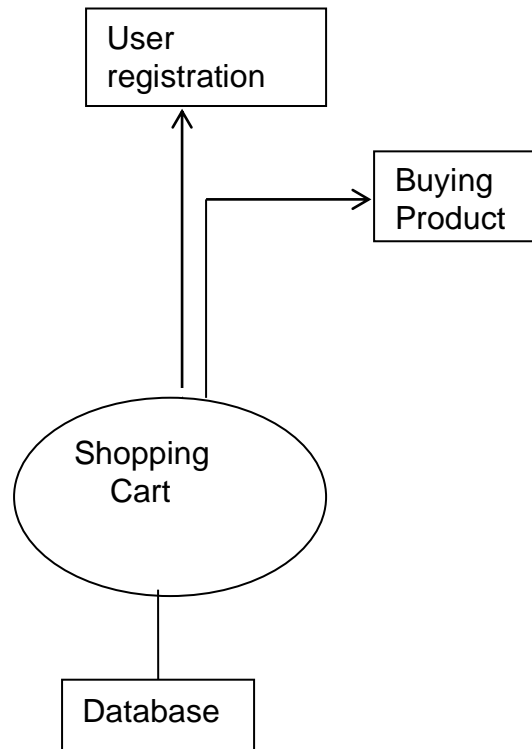


Figure 8 Database

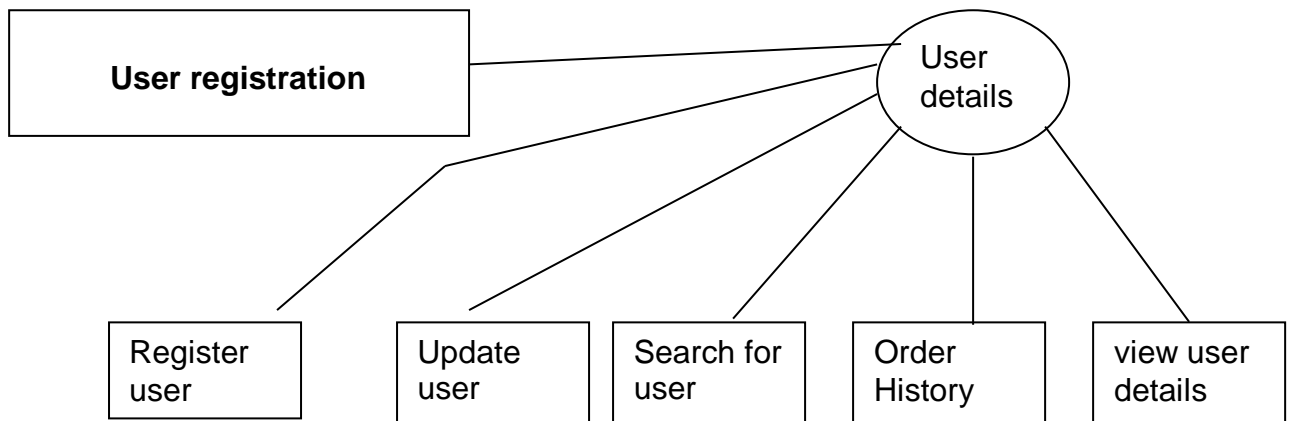


Figure9 User registration

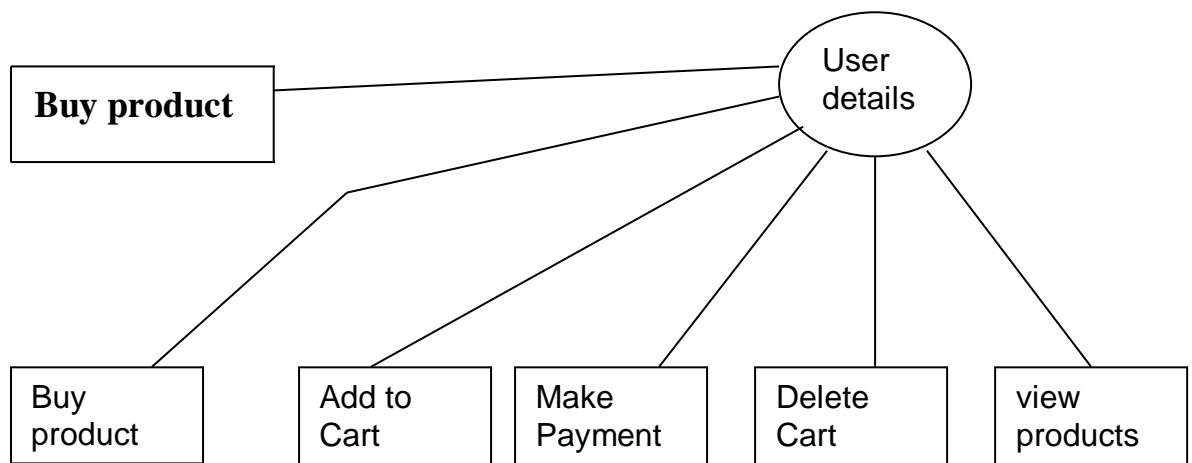


Figure 10 Buying Product

Dataflow diagram full:

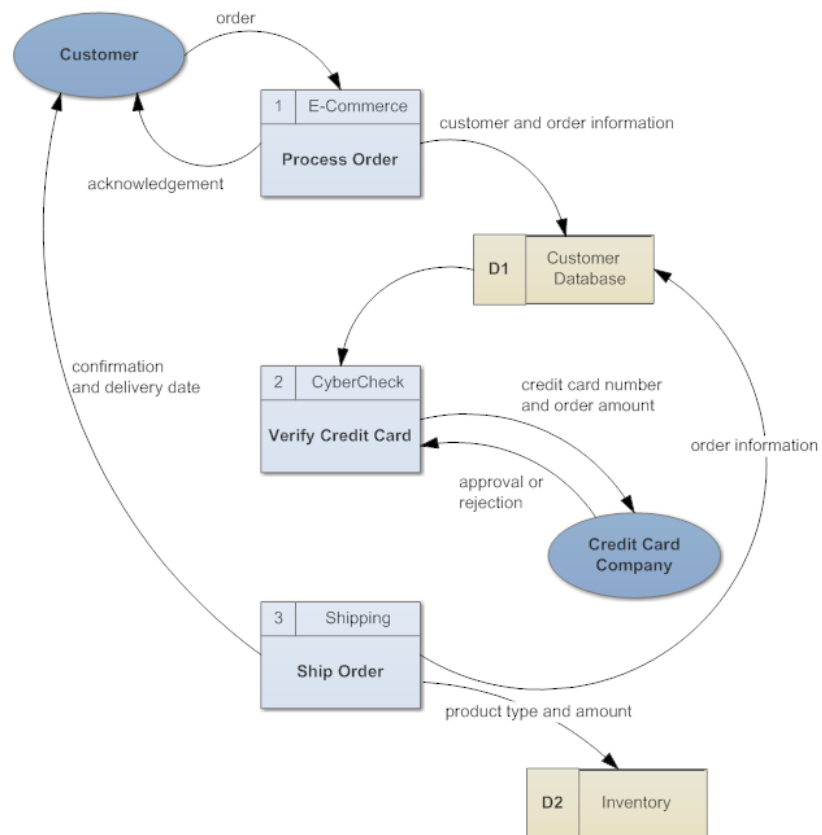


Figure 11 Ordering Process

E-R Diagrams:

The Entity-Relationship (ER) model was originally proposed by Peter in 1976 [Chen76] as a way to unify the network and relational database views. Simply stated the ER model is a conceptual data model that views the real world as entities and relationships. A basic component of the model is the Entity-Relationship diagram which is used to visually represents data objects. Since Chen wrote his paper the model has been extended and today it is commonly used for database design For the database designer, the utility of the **ER model is:**

- it maps well to the relational model. The constructs used in the ER model can easily be transformed into relational tables.
- it is simple and easy to understand with a minimum of training. Therefore, the model can be used by the database designer to communicate the design to the end user.
- In addition, the model can be used as a design plan by the database developer to implement a data model in a specific database management software.

Connectivity and Cardinality

The basic types of connectivity for relations are: one-to-one, one-to-many, and many-to-many. A *one-to-one* (1:1) relationship is when at most one instance of a entity A is associated with one instance of entity B. For example, "employees in the company are each assigned their own office. For each employee there exists a unique office and for each office there exists a unique employee.

A one-to-many (1:N) relationships is when for one instance of entity A, there are zero, one, or many instances of entity B, but for one instance of entity B, there is only one instance of entity A. An example of a 1:N relationships is a **department has many employee search employee is assigned to one department** A many-to-many (M:N) relationship, sometimes called non-specific, is when for one instance of entity A, there are zero, one, or many instances of entity B and for one instance of entity B there are zero, one, or many instances of entity A. The connectivity of a relationship describes the mapping of associated

ER Notation

There is no standard for representing data objects in ER diagrams. Each modeling methodology uses its own notation. The original notation used by Chen is widely used in academics texts and journals but rarely seen in either CASE tools or publications by non-academics. Today, there are a number of notations used, among the more common are Bachman, crow's foot, and IDEFIX.

All notational styles represent entities as rectangular boxes and relationships as lines connecting boxes. Each style uses a special set of symbols to represent the cardinality of a connection. The notation used in this document is from Martin. The symbols used for the basic ER constructs are:

- entities are represented by labeled rectangles. The label is the name of the entity. Entity names should be singular nouns.
- relationships are represented by a solid line connecting two entities. The name of the relationship is written above the line. Relationship names should be verbs
- attributes, when included, are listed inside the entity rectangle. Attributes which are identifiers are underlined. Attribute names should be singular nouns.
- cardinality of many is represented by a line ending in a crow's foot. If the crow's foot is omitted, the cardinality is one.
- existence is represented by placing a circle or a perpendicular bar on the line. Mandatory existence is shown by the bar (looks like a 1) next to the entity for an instance is required. Optional existence is shown by placing a circle next to the entity that is optional

ER DIAGRAM :

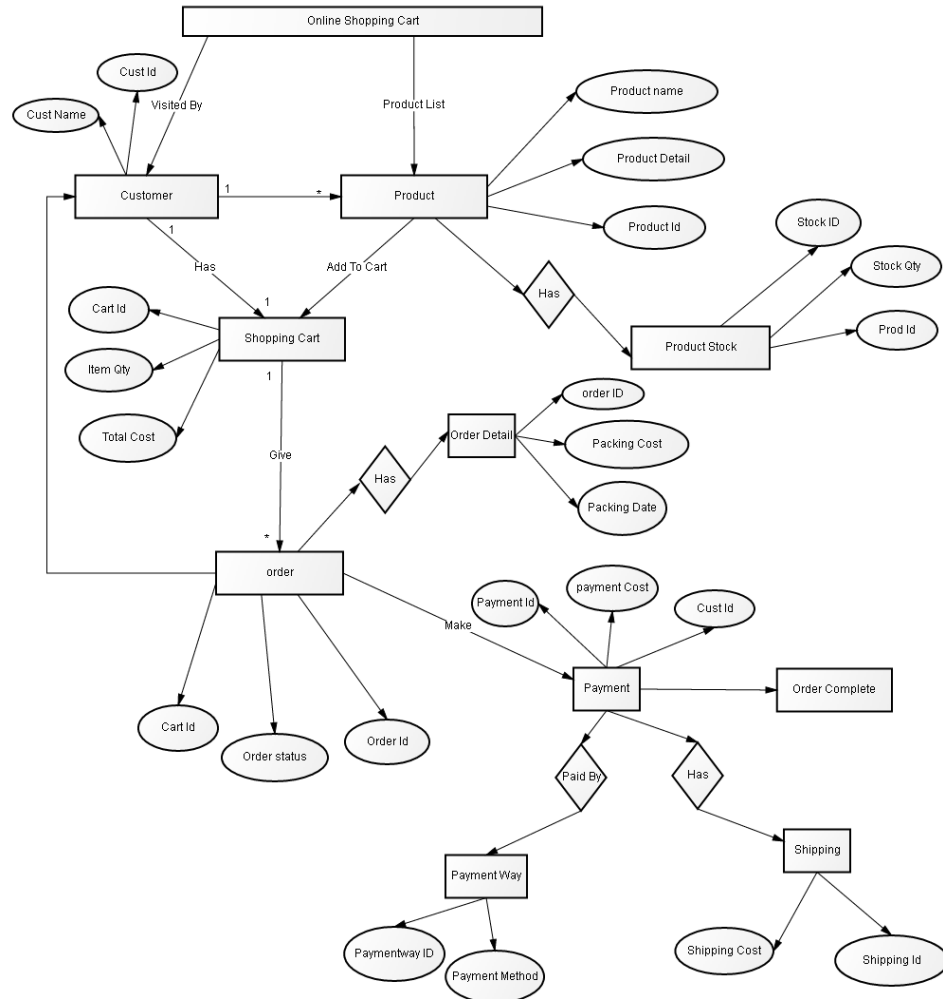


Figure 12 *Architecture Overflow*

PROJECT MODULES

MODULES :

This project contains 3 modules, those are

- **Admin**
- **Products**
- **User**

MODULES DESCRIPTION:

Admin:-

When admin login, he saw the customer's database, means how many users are authenticated to this website and how many users are transact everyday, and newly items are inserting into products.

Products:-

This module contains product name, and related image, and cost of its. Like toys, books, furniture, gold items, etc.. Whatever customer wants from the shopping cart.

User:-

User entered into with his username and password, when he entered into this, he saw what items are available today, this facility is available for this site. Chosen different items from website get those through door delivery

SOURCECODES

About intranet and internet

Technologically, the Internet is network of computers. Not just a few special Computers, but over nine million of all kinds of computers. Similarly it is not just a network, but a network of networks hence the name and using TCP/IP (transmission control protocol and internet protocol).

Internet is the name for a vast, worldwide system consisting of people, information and computers. Internet is global communication system of diverse, INTER connected computer NETWORK for the exchange of information of virtually every conceivable topic known to man.

Internet is not just one thing. It is a lot of things to lot of people. In today's world it is one of the most important commodity of life. The Internet is more important in what it enables than what it is, more of a phenomenon than fact.

Intranet

The classical definition of Intranet is the application of the Internet technologies to the internal business applications media most refer to the Intranet in terms of applying web technologies to information systems in the organization.

Introduction to JavaScript

JavaScript:

JavaScript is a new scripting language for WebPages. Scripts written with java script can be embedded into your HTML pages. With java script you have many possibilities for enhancing your HTML page with interesting elements. For example you are able to respond to user-initiated events quite easily. Some effects that are now possible with java script were some time ago only possible with CGI. So you can create really sophisticated pages with the helps of java script on the Internet.

Difference between java and Java Script

Although the names are almost the same Java is not the same as Java Script. These are two different techniques for Internet programming. Java is programming language. JavaScript is a scripting language as the name implies. The difference is that we can create real programs with java. But java script is not real programming. Java Script is meant to be easy to understand and easy to use. JavaScript authors should not have to care too much about programming. We could say that Java Script is rather an extension to HTML than a separate computer language. Of course this is not the official definition but it makes it easier to understand the difference between java and java script.

How can Java Script scripts run?

The first browser to support java script was the Netscape Navigator 2.0 of course the higher versions do have java script as well. You might know that java does not run on all Netscape Navigators 2.0 (or higher versions) versions. But this is not true for java script -although there are some problems with the different versions.

The Mac version for example seems to have many bugs. In the near future there are going to be some other browsers, which support java script. The Microsoft Internet explorer 3.0 is going to support java script. JavaScript enabled browsers are going to spread soon - it is worth learning this new technique now. You might realize that is really easy to write Java Script scripts. We have to know is some basic techniques and some work-around for problems you might encounter. Of course we need a basic. Understanding HTML before reading this tutorial you can find many really good online resources about HTML. Best you make an online search about 'html' at yahoo if you want to get informed about HTML. Now I want to show some small scripts so you can learn how they are implemented into HTML-documents and to show which possibilities you have with the new scripting language. The following is a very small script, which will only print a text into an HTML document.

```
<html>
```

```
<head>
```

```
My first JavaScript
```

```
</head>
```

```
<body><br>
```

```
This is a normal HTML document
```

```
<br>
```

```
<script language="JavaScript">
```

```
Document.write ("this is a java script")
```

```
</script><b r>
```

```
Backing HTML again
```

```
</body>
```

```
</html>
```

If you are using a java script enabled-browser at the moment then you will have the possibility to see this script working. If your browser doesn't support Java Script then this output might be some kind of strange...

This is a normal HTML document

This is java script!

Back in HTML again.

Functions

Functions are bet declared between the <Head> tag of HTML page. Functions are called by user-initiated events. Seems reasonable to keep the functions between the <Head> tags. They are loaded first before a user can do anything that might call a function. Scripts can be placed between inside comment fields to ensure that older browser do not display the script itself.

```
<html>
<head>
<script language="JavaScript">
function pushbutton (){
alert ("Hello!");
}
</script>
</head>
<body>
<form>
<input type="button" name="Button1" value="push me" onclick="pushbutton ()">
</form>
</body>
</html>
```

If we want to test this one immediately and you are using a Java Script enabled browser then please go ahead and push the button.

This script will create a button and when you press it a window will pop up saying "hello!". In fact we have a lot of possibilities just by adding functions to our scripts.

The common browsers transmit the form information by either method: here's the complete tag including the GET transmission method attribute for the previous form

Example

```
<Form method =GET action=http://phpgurukul.com/>
.....
</form>
```

Input elements.

Use the `<input>` tag to define any one of a number of common form elements including text fields multiple choice lists click able images and submission buttons. There are many attributers for this tag only that types and name attributes are required for each element, each type of input element uses only a subset of the followed attributes. Additional `<input>` attributes may be required based upon which type of the form element you specify

Submit button:

The submit button (`<input type=submit>`) does what its name implies, settings in motion the form's submission to the server from the browser. We many have more than submit buttons will be added to the parameter list the browser sends along to the server.

Example

```
< Input type ="submit">
```

```
<Input type="submit" value="submit" name="name">
```

Reset button:

The reset button if firm `<input>` button is nearly self- explanatory; it lets the user reset erase or set to some default value all elements in the form. By default the browser displays a reset button worth the label "reset". We can change that by specifying a value attribute with tour own button label.

Coding Pattern of Project(login.php):

```
<?php
session_start();
error_reporting(0);
include('includes/config.php');
// Code user Registration
if(isset($_POST['submit']))
{
$name=$_POST['fullname'];
$email=$_POST['emailid'];
$contactno=$_POST['contactno'];
$password=md5($_POST['password']);
$query=mysql_query("insert into users(name,email,contactno,password)
values('$name','$email','$contactno','$password')");
if($query)
{
echo "<script>alert('You are successfully register');</script>";
}
else{
echo "<script>alert('Not register something went worng');</script>";
}
}
// Code for User login
if(isset($_POST['login']))
{
$email=$_POST['email'];
$password=md5($_POST['password']);
$query=mysql_query("SELECT * FROM users WHERE email='$email' and
password='$password'");
$num=mysql_fetch_array($query);
if($num>0)
{
$extra="my-cart.php";
$_SESSION['login']=$_POST['email'];
$_SESSION['id']=$num['id'];
$_SESSION['username']=$num['name'];
$uip=$_SERVER['REMOTE_ADDR'];
$status=1;
$log=mysql_query("insert into userlog(userEmail,userip,status)
values('".$_SESSION['login']."','$uip','$status')");
$host=$_SERVER['HTTP_HOST'];
$url=rtrim(dirname($_SERVER['PHP_SELF']),'/\\');
header("location:http://$host$url/$extra");
exit();
}
else
{
$extra="login.php";
```

```

$email=$_POST['email'];
$uip=$_SERVER['REMOTE_ADDR'];
$status=0;
$log=mysql_query("insert into userlog(userEmail,userip,status)
values('$email','$uip','$status')");
$host = $_SERVER['HTTP_HOST'];
$uri = rtrim(dirname($_SERVER['PHP_SELF']),'/\');
header("location:http://$host$uri/$extra");
$_SESSION['errmsg']="Invalid email id or Password";
exit();
}
}

?>

<!DOCTYPE html>
<html lang="en">
<head>
<!-- Meta -->
<meta charset="utf-8">
<meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
<meta name="viewport" content="width=device-width, initial-scale=1.0, user-
scalable=no">
<meta name="description" content="">
<meta name="author" content="">
<meta name="keywords" content="MediaCenter, Template, eCommerce">
<meta name="robots" content="all">

<title>Shopping Portal | Signi-in | Signup</title>

<!-- Bootstrap Core CSS -->
<link rel="stylesheet" href="assets/css/bootstrap.min.css">

<!-- Customizable CSS -->
<link rel="stylesheet" href="assets/css/main.css">
<link rel="stylesheet" href="assets/css/green.css">
<link rel="stylesheet" href="assets/css/owl.carousel.css">
<link rel="stylesheet" href="assets/css/owl.transitions.css">
<!--<link rel="stylesheet" href="assets/css/owl.theme.css">-->
<link href="assets/css/lightbox.css" rel="stylesheet">
<link rel="stylesheet" href="assets/css/animate.min.css">
<link rel="stylesheet" href="assets/css/rateit.css">
<link rel="stylesheet" href="assets/css/bootstrap-select.min.css">

<!-- Demo Purpose Only. Should be removed in production -->
<link rel="stylesheet" href="assets/css/config.css">

<link href="assets/css/green.css" rel="alternate stylesheet" title="Green color">
<link href="assets/css/blue.css" rel="alternate stylesheet" title="Blue color">
<link href="assets/css/red.css" rel="alternate stylesheet" title="Red color">

```

```

<link href="assets/css/orange.css" rel="alternate stylesheet" title="Orange color">
<link href="assets/css/dark-green.css" rel="alternate stylesheet" title="Darkgreen
color">
<!-- Demo Purpose Only. Should be removed in production : END -->

```

```

<!-- Icons/Glyphs -->
<link rel="stylesheet" href="assets/css/font-awesome.min.css">

```

```

<!-- Fonts -->
<link href='http://fonts.googleapis.com/css?family=Roboto:300,400,500,700'
rel='stylesheet' type='text/css'>

```

```

<!-- Favicon -->
<link rel="shortcut icon" href="assets/images/favicon.ico">
<script type="text/javascript">
function valid()
{
if(document.register.password.value!= document.register.confirmpassword.value)
{
alert("Password and Confirm Password Field do not match !!");
document.register.confirmpassword.focus();
return false;
}
return true;
}
</script>
</head>
<body class="cnt-home">

```

```

<!--HEADER-->
<header class="header-style-1">

```

```

<!-- TOP MENU-->
<?php include('includes/top-header.php');?>
<!-- TOP MENU : END -->
<?php include('includes/main-header.php');?>
<!-- NAVBAR -->
<?php include('includes/menu-bar.php');?>
<!-- NAVBAR : END -->

```

```

</header>

```

```

<!-- HEADER : END -->
<div class="breadcrumb">
<div class="container">
<div class="breadcrumb-inner">
<ul class="list-inline list-unstyled">
<li><a href="home.html">Home</a></li>
<li class='active'>Authentication</li>

```



```

</ul>
</div><!-- /.breadcrumb-inner -->
</div><!-- /.container -->
</div><!-- /.breadcrumb -->

<div class="body-content outer-top-bd">
<div class="container">
<div class="sign-in-page inner-bottom-sm">
<div class="row">
<!-- Sign-in -->
<div class="col-md-6 col-sm-6 sign-in">
<h4 class="">sign in</h4>
<p class="">Hello, Welcome to your account.</p>
<form class="register-form outer-top-xs" method="post">
<span style="color:red;" >
<?php
echo htmlentities($_SESSION['errmsg']);
?>
<?php
echo htmlentities($_SESSION['errmsg']="");
?>
</span>
<div class="form-group">
<label class="info-title" for="exampleInputEmail1">Email Address
<span>*</span></label>
<input type="email" name="email" class="form-control unicast-form-control text-
input" id="exampleInputEmail1" >
</div>
<div class="form-group">
<label class="info-title" for="exampleInputPassword1">Password
<span>*</span></label>
<input type="password" name="password" class="form-control unicast-form-control
text-input" id="exampleInputPassword1" >
</div>
<div class="radio outer-xs">
<a href="forgot-password.php" class="forgot-password pull-right">Forgot your
Password?</a>
</div>
<button type="submit" class="btn-upper btn btn-primary checkout-page-button"
name="login">Login</button>
</form>
</div>
<!-- Sign-in -->

<!-- create a new account -->
<div class="col-md-6 col-sm-6 create-new-account">
<h4 class="checkout-subtitle">create a new account</h4>
<p class="text title-tag-line">Create your own Shopping account.</p>
<form class="register-form outer-top-xs" role="form" method="post"
name="register" onSubmit="return valid();">

```

```

<div class="form-group">
<label class="info-title" for="fullname">Full Name <span>*</span></label>
<input type="text" class="form-control unicast-form-control text-input"
id="fullname" name="fullname" required="required">
</div>

```

```

<div class="form-group">
<label class="info-title" for="exampleInputEmail2">Email Address
<span>*</span></label>
<input type="email" class="form-control unicast-form-control text-input"
id="exampleInputEmail2" name="emailid" required >
</div>

```

```

<div class="form-group">
<label class="info-title" for="contactno">Contact No. <span>*</span></label>
<input type="text" class="form-control unicast-form-control text-input"
id="contactno" name="contactno" maxlength="10" required >
</div>

```

```

<div class="form-group">
<label class="info-title" for="password">Password. <span>*</span></label>
<input type="password" class="form-control unicast-form-control text-input"
id="password" name="password" required >
</div>

```

```

<div class="form-group">
<label class="info-title" for="confirmpassword">Confirm Password.
<span>*</span></label>
<input type="password" class="form-control unicast-form-control text-input"
id="confirmpassword" name="confirmpassword" required >
</div>

```

```

<button type="submit" name="submit" class="btn-upper btn btn-primary checkout-
page-button">Sign Up</button>
</form>
<span class="checkout-subtitle outer-top-xs">Sign Up Today And You'll Be Able To
: </span>
<div class="checkbox">
<label class="checkbox">
Speed your way through the checkout.
</label>
<label class="checkbox">
Track your orders easily.
</label>
<label class="checkbox">
Keep a record of all your purchases.
</label>
</div>

```

```

</div>
<!-- create a new account --></div><!-- /.row -->
</div>
<?php include('includes/brands-slider.php');?>
</div>
</div>
<?php include('includes/footer.php');?>
<script src="assets/js/jquery-1.11.1.min.js"></script>

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

<script src="assets/js/bootstrap-hover-dropdown.min.js"></script>
<script src="assets/js/owl.carousel.min.js"></script>

<script src="assets/js/echo.min.js"></script>
<script src="assets/js/jquery.easing-1.3.min.js"></script>
<script src="assets/js/bootstrap-slider.min.js"></script>
<script src="assets/js/jquery.rateit.min.js"></script>
<script type="text/javascript" src="assets/js/lightbox.min.js"></script>
<script src="assets/js/bootstrap-select.min.js"></script>
<script src="assets/js/wow.min.js"></script>
<script src="assets/js/scripts.js"></script>
<script src="switchstylesheet/switchstylesheet.js"></script>
<script>
$(document).ready(function(){
$(".change-color").switchstylesheet( { separator:"color" } );
$('.show-theme-options').click(function(){
$(this).parent().toggleClass('open');
return false;
});
});

$(window).bind("load", function() {
$('.show-theme-options').delay(2000).trigger('click');
});
</script>

</body>
</html>

```

OVERVIEW OF TECHNOLOGIES USED

3.1 Front End Technology

PHP

PHP is a server-side scripting language designed specifically for the web. Within an HTML page, you can embed PHP code that will be executed each time the page is visited. Your PHP code is interpreted at the web server and generates HTML or other output that the visitor will see.

PHP was introduced in 1994. As of November 2007, it was installed on more than 21 million domains worldwide, and this number is growing rapidly. You can see the current number at <http://www.php.net/usage.php>

PHP is an Open Source project. PHP originally stood for Personal Home Page and now stands for PHP Hypertext Preprocessor.

Unique Features

If you are familiar with other server side language like ASP.NET or JSP you might be wondering what makes PHP so special, or so different from these competing alternatives well, here are some reasons:

- Performance
- Portability(Platform Independent)
- Ease Of Use
- Open Source
- Third-Party Application Support
- Community Support

Performance

Scripts written in PHP executes faster than those written in other scripting language, with numerous independent benchmarks, putting the language ahead of competing alternatives like JSP, ASP.NET and PERL. The PHP 5.0 engine was completely redesigned with an optimized memory manager to improve performance, and is noticeable faster than previous versions. In addition, third party accelerators are available to further improve performance and response time.

Portability

PHP is available for UNIX, MICROSOFT WINDOWS, MAC OS, and OS/2. PHP Programs are portable between platforms. As a result, a PHP application developed on, say, Windows will typically run on UNIX without any significant issues. This ability to easily undertake cross-platform development is a valuable one, especially when operating in a multi platform corporate environment or when trying to address multiple market segments.

Ease Of Use

“Simplicity is the ultimate sophistication”, Said Leonardo da Vinci, and by that measure, PHP is an extremely sophisticated programming language. Its syntax is clear and consistent, and it comes with exhaustive documentation for the 5000+ functions included with the core distributions. This significantly reduces the learning curve for both novice and experienced programmers, and it’s one of the reasons that PHP is favored as a rapid prototyping tool for Web-based applications.

Open Source

PHP is an open source project – the language is developed by a worldwide team of volunteers who make its source code freely available on the Web, and it may be used without payment of licensing fees or investments in expensive hardware or software. This reduces software development costs without affecting either flexibility or reliability. The open-source nature of the code further means that any developer, anywhere, can inspect the code tree, spit errors, and suggest possible fixes, this produces a stable, robust product wherein bugs, once discovered, are rapidly resolved – sometimes within a few hours of discovery !.

Third-Party Application Support

One of PHP’s Strengths has historically been its support for a wide range of different databases, including MySQL, PostgreSQL, Oracle, and Microsoft SQL Server. PHP 5.3 Supports more than fifteen different database engines, and it includes a common API for database access. XML support makes it easy to read and write XML documents though they were native PHP data structures, access XML node collections using Xpath, and transform XML into other formats with XSLT style sheets.

Community Support

One of the nice things about a community-supported language like PHP is the access it offers to the creativity and imagination of hundreds of developers across the world. Within the PHP community, the fruits of this creativity may be found in PEAR, the PHP Extension and Application Repository and PECL, the PHP Extension Community Library, which contains hundreds of ready-made widgets and extensions that developers can use to painlessly add new functionality to PHP. Using these widgets is often a more time- and cost-efficient alternative to rolling your own code.

PHP Server

The PHP Community Provides Some types of Software Server solution under The GNU (General Public License).

These are the following:

- WAMP Server
- LAMP Server
- MAMP Server
- XAMPP Server

All these types of software automatic configure inside operating system after installation it having PHP, MySQL, Apache and operating system base configuration file, it doesn't need to configure manually.

WAMP----- Microsoft window o/s,Apache Mysql PHP

LAMP---- Linux Operating System Apache Mysql PHP

MAMP---- Mac os Apache Mysql PHP

XAMPP---- x-os (cross operating system) Apache Mysql PHP Perl

Introduction to HTML

The hypertext markup language (HTML) is a simple markup language. Used to create a hypertext documents that are portable from one platform to another HTML documents are SGML (Standard generalized mark up language) documents with generic semantics that are appropriate for representing information from a wide range of applications. This specification defines HTML version 3.2. HTML 3.2 aims to capture recommended practice as of early '96 and as such a replacement for HTML2.0 (RFC 1866).

A set of instructions embedded in a document is called mark up language. These instructions describe what the document text means and how it should look like in a display. Hyper Text Mark Up language (HTML) is the language used to encode World Wide Web documents.

WHY TO USE HTML?

Website is a collection of pages, publications, and documents that reside on web server. While these pages publications and a document as a formatted in a single format, you should use HTML for home page and all primary pages in the site. This will enable the millions of web users can easily access and to take advantage of your website.

HTML is considered first for formatting any new material you plan to publish on the web. HTML documents are platform independent, meaning that they don't conform to any standard. If they are created properly you can move home page to any server platform or you can access them with any complaint www browser.

STRUCTURE OF HTML

HTML elements perform a defined task. HTML uses two types of elements

- . Empty Tags

- . Container Tags

These tags differ because of what they represent. Empty tags represent formatting constricts such as line breaks and horizontal rules. Container tags define a section of text, formats and dot all of the selected text. A container tag has both a beginning and an ending.

HTML LAYOUT:

An HTML document consists of text, which comprises the content of the document and tags, which, defines the structure, and appearance of the document. The structure of an HTML document is simple, consists of outer.

<HTML>tag enclosing the document header and body

<HTML>

<HEAD>

<TITLE>the title of HTML document</TITLE>

</HEAD>

<BODY>

This is where the actual HTML documents

Text lies, which is displayed in the browser

</BODY></HTML>

Each document has a head and body delimited by the <HEAD> and <BODY> tag. The head is where you give your HTML document a title and where you indicate other parameters the browser may use when displaying the document. This includes the text for displaying the text. Tag also references special and indicates the hot spots that link your document to other documents

HTML FORMS:

Creating a form usually involves two independent steps: Creating the layout for the form itself and then writing a script program on the server side to process the formation you get back from a form.

To create a form, You use the `<FORM>` tag. Inside the opening and closing FORM tags are each of the individual form elements plus any other HTML content to create a layout for that form.

The opening tag of the FORM element usually includes the attributes: METHOD and ACTION. The METHOD attributes can be either GET or POST which determines how your form data is sent to the script to process it.

The ACTION attribute is a pointer to the script that processes the form on the server side. The ACTION can be included by a relative path or by a full URL to a script on your server or somewhere else. For example, the following `<FORM>` tag would call a script called form-name in cgi-bin directory on server `www.myservser.com`

```
<FORM Method= post action=http://www.mytservser.com/cgi-bin/form-name.pl>
.....
</FORM>
```

METHOD ATTRIBUTE:

The other required attribute for the `<form>` tag sets the methods by which the browser form's data to the server for processing. There are two ways: the POST method and GET method. With POST method, the browser sends the data in two steps: the browser first contacts the form-processing server specified in the action attributes, and once contact is made, sends the data.

BACK END TECHNOLOGY:

MySQL Introduction

There are a large number of database management systems currently available, some commercial and some free.

Some of them : **Oracle, Microsoft Access, Mysql and PostgreSQL.**

These database systems are powerful, feature-rich software, capable of organizing and searching millions of records at very high speeds.

Understanding Databases, Records, and Primary Keys

Every Database is composed of one or more tables.

These Tables, which structure data into rows and columns, Impose organization on the data.

The records in a table(below) are not arranged in any particular order.

SQL, statements fall into one of three categories.(Types of SQL)

Data Definition Language(DDL) : DDL Consists of statements that define the structure and relationships of a database and its table.

These Statements are used to Create, drop and modify databases and tables.

Data Manipulation Language(DML) : DML statements are related to altering and extracting data from a database.

These statements are used to add records to, update records in, and delete records from, a database; perform queries; retrieve table records matching one or more user specified criteria; and join tables together using their common fields.

Data Control Language(DCL) : DCL statements are used to define access levels and security privileges for a database.

You would use these statements to grant or deny user privileges; assign roles; change passwords; view permissions; and create rule sets to protect access to data.

The Syntax of SQL is quite intuitive. every SQL statement begins with an “action word”, like DELETE, INSERT, ALTER etc.

it ends with a semicolon. whitespace, tabs, carriage returns are ignored.

Some example of valid SQL statements :

CREATE DATABASE employee;

```
SELECT name FROM users where email ="anuj.lpu1@gmail.com";  
DELETE FROM cars WHERE year_of_manufacture < 1980;
```

PHP Mysql connectivity

Use the mysql connect() function to establish connection to the MySQL server. To access the database functionality we have to make a connection to database using Php.

mysql_connect() function is used to establish the connection to mysql server. Four arguments need to be passed to mysql_connect() function.

hostname : if you are working on local system , you can use localhost or you can also provide ip address or server name.

username : if there is a existing user , you can provide username. default username is 'root'.

password - by default password is blank or null.

dbname - it is a optional field . it is basically a name of the database that need to be connected.

mysql_connect-(host, username, password, dbname);

host(Server name) Either a host name(server name) or an IP address

username-The MySQL user name

password-The password to log in with

dbname-Optional. The database to be used when performing queries

Note:

There are more available parameters, but the ones listed above are the most important.

In the following example we store the connection in a variable (\$con) for later use in the script.

```
<?php
```

```
// Create connection
```

```
$con=mysql_connect("localhost","root","") or die(mysql_error());
```

```
?>
```

Here localhost is server name. root is MySQL default user name. default password is blank and database name is my_db. **mysql_error()** function provides mysql connectivity error message.

MySQL Close Connection

```
<?php
```

```
// Create connection
```

```
$con=mysql_connect("localhost","root","","my_db") or die(mysql_error());
```

```
//code to be executed.
```

```
// Close connection
```

```
mysql_close($con);
```

```
?>
```

after work with the database is done we have to close the connection using `mysql_close()` function in which the connection to the database is passed.

Client-side Script(JAVASCRIPT):-

JavaScript:

JavaScript is a new scripting language for WebPages. Scripts written with java script can be embedded into your HTML pages. With java script you have many possibilities for enhancing your HTML page with interesting elements. For example you are able to respond to user-initiated events quite easily. Some effects that are now possible with java script were some time ago only possible with CGI. So you can create really sophisticated pages with the helps of java script on the Internet.

How can Java Script scripts run?

The first browser to support java script was the Netscape Navigator 2.0 of course the higher versions do have java script as well. You might know that java does not run on all Netscape Navigators 2.0 (or higher versions) versions. But this is not true for java script -although there are some problems with the different versions.

The Mac version for example seems to have many bugs. In the near future there are going to be some other browsers, which support java script. The Microsoft Internet explorer 3.0 is going to support java script. JavaScript enabled browsers are going to spread soon - it is worth learning this new technique now. You might realize that is really easy to write Java Script scripts. We have to know is some basic techniques and some work-around for problems you might encounter. Of course we need a basic. Understanding HTML before reading this tutorial you can find many really good online resources about HTML. Best you make an online search about

‘html’ at yahoo if you want to get informed about HTML. Now I want to show some small scripts so you can learn how they are implemented into HTML-documents and to show which possibilities you have with the new scripting language. The following is a very small script, which will only print a text into an HTML document.

```
<html>
<head>
My first JavaScript
</head>
<body><br>
This is a normal HTML document
<br>
<script language="JavaScript">
Document.write ("this is a java script")
</script><b r>
Backing HTML again
</body>
</html>
```

If you are using a java script enabled-browser at the moment then you will have the possibility to see this script working. If your browser doesn't support Java Script then this output might be some kind of strange...

This is a normal HTML document

This is java script!

Back in HTML again.

Functions

Functions are bet declared between the <Head> tag of HTML page. Functions are called by user-initiated events. Seems reasonable to keep the functions between the <Head> tags. They are loaded first before a user can do anything that might call a function. Scripts can be placed between inside comment fields to ensure that older browser do not display the script itself.

```
<html>
<head>
<script language="JavaScript">
function pushbutton (){
alert ("Hello!");
}
</script>
</head>
<body>
<form>
<input type="button" name="Button1" value="push me" onclick="pushbutton ()">
</form>
</body>
</html>
```

If we want to test this one immediately and you are using a Java Script enabled browser then please go ahead and push the button.

This script will create a button and when you press it a window will pop up saying "hello!". In fact we have a lot of possibilities just by adding functions to our scripts. The common browsers transmit the form information by either method: here's the complete tag including the GET transmission method attribute for the previous form

Example

```
<Form method =GET action=http://www.mycompany.com/ upfdade.php>
.....
</form>
```

Input elements

Use the `<input>` tag to define any one of a number of common form elements including text fields multiple choice lists click able images and submission buttons. There are many attributers for this tag only that types and name attributes are required for each element, each type of input element uses only a subset of the followed attributes. Additional `<input>` attributes may be required based upon which type of the form element you specify.

Submit button:

The submit button (`<input type=submit>`) does what its name implies, settings in motion the form's submission to the server from the browser. We many have more than submit buttons will be added to the parameter list the browser sends along to the server.

Example

```
< Input type ="submit">
```

```
<Input type="submit" value="submit" name="name">
```

Reset button:

The reset button if firm `<input>` button is nearly self- explanatory; it lets the user reset erase or set to some default value all elements in the form. By default the browser displays a reset button worth the label "reset". We can change that by specifying a value attribute with tour own button label

DATABASE MODELS

Single Tier:

In a single tier the server and client are the same in the sense that a client program that needs information (client) and the source of this type of architecture is also possible in java, in case flat files are used to store the data. However this is useful only in case of small applications. The advantage with this is the simplicity and portability of the application developed.

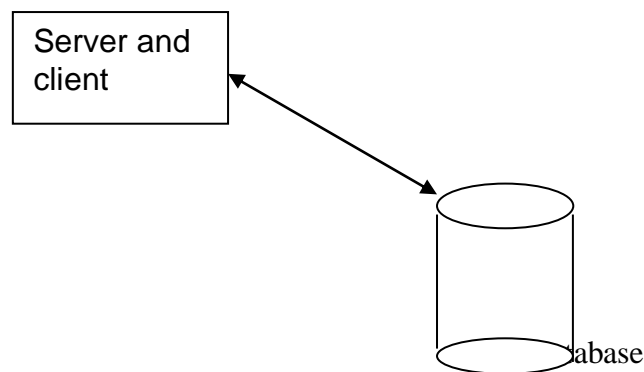


Figure13Single Tier

Two Tier (client-server):

In two tier architecture the database resides in one machine and client in different machine they are connected through the network. In this type of architecture a database management takes control of the database and provides access to clients in a network. This software bundle is also called as the server. Software in different machines, requesting for information are called as the clients

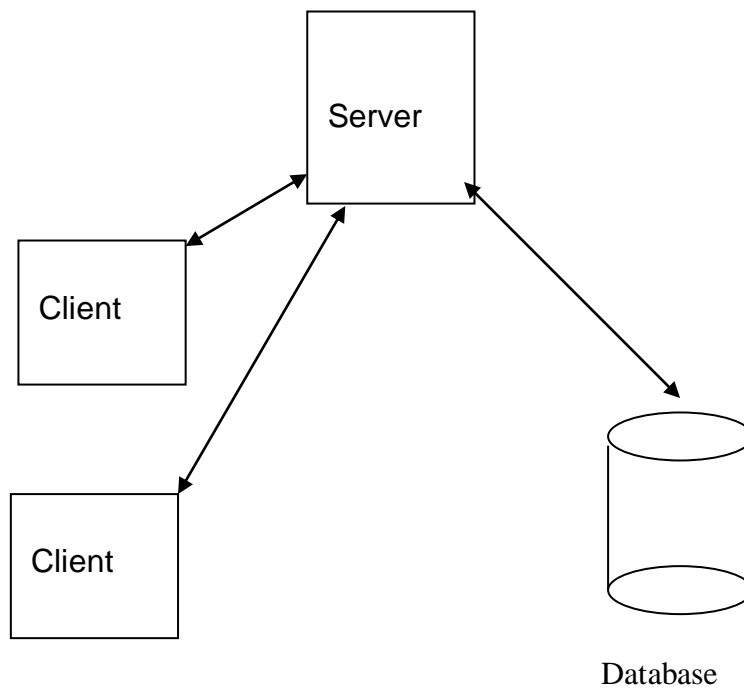


Figure 14 Two-Tier

Three-tier:

In the three-tier architecture, any number servers can access the database that resides on server. Which in turn serve clients in a network. For example, you want to access the database using java applets, the applet running in some other machine, can send request only to the server from which it is down loaded. For this reason we will need to have a intermediate server which will accept the requests from applets and them to the actual database server. This intermediate server acts as a two-way communication channel also. This is the information or data from the database is passed on to the applet that is requesting it. This can be extended to make n tiers of servers, each server carrying to specific type of request from clients, however in practice only 3 tiers architecture is popular.

FEASIBILITY STUDY:

Feasibility study is conducted once the problem is clearly understood. Feasibility study is a high level capsule version of the entire system analysis and design process. The objective is to determine quickly at a minimum expense how to solve a problem. The purpose of feasibility is not to solve the problem but to determine if the problem is worth solving.

The system has been tested for feasibility in the following points.

1. Technical Feasibility
2. Economical Feasibility
3. Operational Feasibility.

1. Technical Feasibility

The project entitles "Courier Service System" is technically feasibility because of the below mentioned feature. The project was developed in Java which Graphical User Interface.

It provides the high level of reliability, availability and compatibility. All these make Java an appropriate language for this project. Thus the existing software Java is a powerful language.

2. Economical Feasibility

The computerized system will help in automate the selection leading the profits and details of the organization. With this software, the machine and manpower utilization are expected to go up by 80-90% approximately. The costs incurred of not creating the system are set to be great, because precious time can be wanted by manually.

3. Operational Feasibility

In this project, the management will know the details of each project where he may be presented and the data will be maintained as decentralized and if any inquires for that particular contract can be known as per their requirements and necessities.

Implementation:

Implementation is the stage where the theoretical design is turned into a working system. The most crucial stage in achieving a new successful system and in giving confidence on the new system for the users that it will work efficiently and effectively.

The system can be implemented only after thorough testing is done and if it is found to work according to the specification.

It involves careful planning, investigation of the current system and its constraints on implementation, design of methods to achieve the change over and an evaluation of change over methods a part from planning. Two major tasks of preparing the implementation are education and training of the users and testing of the system.

The more complex the system being implemented, the more involved will be the systems analysis and design effort required just for implementation.

The implementation phase comprises of several activities. The required hardware and software acquisition is carried out. The system may require some software to be developed. For this, programs are written and tested. The user then changes over to his new fully tested system and the old system is discontinued.

INTRODUCTION TO HTML4.0

What is the World Wide Web?

The World Wide Web is a network of information resources. The Web relies on three mechanisms to make these resources readily available to the widest possible audience.

A uniform naming scheme for locating resources on the Web (e.g. URLs)

Protocols, for access to named resources over the Web (e.g. HTTP)

- Hypertext, for easy navigation among resources (e.g.HTML)
- The ties between the three mechanisms are apparent throughout this specification

What is HTML?

To publish information for global distribution, one needs a universally understood language, a kind of publishing mother tongue that all computers may potentially understand. The publishing language used by the World Wide Web is HTML (from Hyper Text Markup Language). HTML gives authors the means to

- Publish online documents with headings, text, tables, lists, photos, etc.
- Retrieve online information via hypertext links, at the click of a button
- Design forms for conducting transactions with remote services, for use in searching for information, making reservations, ordering products etc.
- Include spread - sheets, video clips, sound clips, and other applications directly in their documents.

A brief history of HTML:

HTML was originally developed by Tim Berners-Lee while at CERN, and popularized by the Mosaic browser developed at NCSA. During the course of the 1990s it has blossomed with the explosive growth of the Web during this time. HTML has been extended in a number of ways. The Web depends on Web page authors and vendors sharing the same conventions for HTML. This has motivated joint work on specifications for HTML.

HTML 2.0 (November 1995) was developed under the aegis of the Internet Engineering Task Force (IETF) to codify common practice in late 1994. HTML (1993) and ([HTML.30]) (1995) proposed much richer versions of HTML, despite never receiving consensus in standards discussions, these drafts led to the adoption of a range new features. The efforts of the World Wide Web Consortium's HTML working group to codify common in 1996 resulted in HTML 3.2 (January 1997). Most people agree that HTML documents should work well across different browsers and platforms. Achieving interoperability lowers costs to content providers since they must develop only one version of a document. If the effort is not made, there is much greater risk that the Web will devolve into a proprietary world of incompatible formats, ultimately reducing the Web's commercial potential for all participants.

SOFTWARE METHODOLOGY

The software methodology followed in this project includes the object-oriented methodology and the application system development methodologies. The description of these methodologies is given below.

Application System Development – A Life cycle Approach:

Although there are a growing number of applications (such as decision support systems) that should be developed using an experimental process strategy such as prototyping, a significant amount of new development work continue to involve major operational applications of broad scope. The application systems are large highly structured. User task comprehension and developer task proficiency is usually high. These factors suggest a linear or iterative assurance strategy. The most common method for this stage class of problems is a system development life cycle modal in which each stage of development is well defined and has straightforward requirements for deliverables, feedback and sign off. The system development life cycle is described in detail since it continues to be an appropriate methodology for a significant part of new development work.

The basic idea of the system development life cycle is that there is a well-defined process by which an application is conceived and developed and implemented. The life cycle gives structure to a creative process. In order to manage and control the development effort, it is necessary to know what should have been done, what has been done, and what has yet to be accomplished. The phrases in the system development life cycle provide a basis for management and control because they define segments of the flow of work, which can be identified for managerial purposes and specifies the documents or other deliverables to be produced in each phase.

The phases in the life cycle for information system development are described differently by different writers, but the differences are primarily in the amount of necessity and manner of categorization. There is a general agreement on the flow of development steps and the necessity for control procedures at each stage.

The information system development cycle for an application consists of three major stages.

- Definition.
- Development.
- Installation and operation.

The first stage of the process, which defines the information requirements for a feasible cost effective system. The requirements are then translated into a physical system of forms, procedures, programs etc., by the system design, computer programming and procedure development. The resulting system is test and put into operation. No system is perfect so there is always a need for maintenance changes. To complete the cycle, there should be a post audit of the system to evaluate how well it performs and how well it meets the cost and performance specifications. The stages of definition, development and installation and operation can therefore be divided into smaller steps or phrases as follows.

Definition

Proposed definition : preparation of request for proposed applications.

Feasibility assessment : evaluation of feasibility and cost benefit of proposed system.

Information requirement analysis : determination of information needed.

Design

Conceptual design : User-oriented design of application development.

Physical system design : Detailed design of flows and processes in applications processing system and preparation of program specification

Development : coding and testing of computer programs.

Procedure development : design of procedures and preparation of user instructions.

Installation and operation

Conversion : final system test and conversion.

Operation and maintenance : Month to month operation and maintenance

Post audit : Evaluation of development process,application system and results of use at the completion of the each phase, formal approval sign-off is required from the users as well as from the manager of the project development.

HTML CODE:

```
<html>
<head>
<h4 class="">sign in</h4>
<p class="">Hello, Welcome to your account.</p>
<form class="register-form outer-top-xs" method="post">
<div class="form-group">
<label class="info-title" for="exampleInputEmail1">Email Address
<span>*</span></label>
<input type="email" name="email" class="form-control unicast-form-control text-
input" id="exampleInputEmail1" >
</div>
<div class="form-group">
<label class="info-title" for="exampleInputPassword1">Password
<span>*</span></label>
<input type="password" name="password" class="form-control unicast-form-control
text-input" id="exampleInputPassword1" >
</div>
<div class="radio outer-xs">
<a href="forgot-password.php" class="forgot-password pull-right">Forgot your
Password?</a>
</div>
<button type="submit" class="btn-upper btn btn-primary checkout-page-button"
name="login">Login</button>
</form></div> </body></html>
```

TESTING

Testing is a process of executing a program with the intent of finding an error. Testing is a crucial element of software quality assurance and presents ultimate review of specification, design and coding. System Testing is an important phase. Testing represents an interesting anomaly for the software. Thus a series of testing are performed for the proposed system before the system is ready for user acceptance testing. A good test case is one that has a high probability of finding an as undiscovered error. A successful test is one that uncovers an as undiscovered error.

Testing Objectives:

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

Testing Principles:

- All tests should be traceable to end user requirements
- Tests should be planned long before testing begins
- Testing should begin on a small scale and progress towards testing in large
- Exhaustive testing is not possible
- To be most effective testing should be conducted by an independent third party

The primary objective for test case design is to derive a set of tests that has the highest likelihood for uncovering defects in software. To accomplish this objective two different categories of test case design techniques are used. They are

- White box testing.
- Black box testing.

White-box testing:

White box testing focuses on the program control structure. Test cases are derived to ensure that all statements in the program have been executed at least once during testing and that all logical conditions have been executed.

Block-box testing:

Black box testing is designed to validate functional requirements without regard to the internal workings of a program. Black box testing mainly focuses on the information domain of the software, deriving test cases by partitioning input and output in a manner that provides thorough test coverage. Incorrect and missing functions, interface errors, errors in data structures, error in functional logic are the errors falling in this category.

Testing strategies:

A strategy for software testing must accommodate low-level tests that are necessary to verify that all small source code segment has been correctly implemented as well as high-level tests that validate major system functions against customer requirements.

Testing fundamentals:

Testing is a process of executing program with the intent of finding error. A good test case is one that has high probability of finding an undiscovered error. If testing is conducted successfully it uncovers the errors in the software. Testing cannot show the absence of defects, it can only show that software defects present.

Testing Information flow:

Information flow for testing flows the pattern. Two class of input provided to test the process. The software configuration includes a software requirements specification, a design specification and source code.

Test configuration includes test plan and test cases and test tools. Tests are conducted and all the results are evaluated. That is test results are compared with expected results. When erroneous data are uncovered, an error is implied and debugging commences.

Unit Testing:

Unit testing is essential for the verification of the code produced during the coding phase and hence the goal is to test the internal logic of the modules. Using the detailed design description as a guide, important paths are tested to uncover errors within the boundary of the modules. These tests were carried out during the programming stage itself. All units of ViennaSQL were successfully tested.

Integration Testing :

Integration testing focuses on unit tested modules and build the program structure that is dictated by the design phase.

System Testing:

System testing tests the integration of each module in the system. It also tests to find discrepancies between the system and its original objective, current specification and system documentation. The primary concern is the compatibility of individual modules. Entire system is working properly or not will be tested here, and specified path ODBC connection will correct or not, and giving output or not are tested here these verifications and validations are done by giving input values to the system and by comparing with expected output. Top-down testing implementing here.

Acceptance Testing:

This testing is done to verify the readiness of the system for the implementation. Acceptance testing begins when the system is complete. Its purpose is to provide the end user with the confidence that the system is ready for use. It involves planning and execution of functional tests, performance tests and stress tests in order to demonstrate that the implemented system satisfies its requirements.

Tools to special importance during acceptance testing include:

Test coverage Analyzer : records the control paths followed for each test case.

Timing Analyzer :also called a profiler, reports the time spent in various regions of the code are areas to concentrate on to improve system performance.

Coding standards :static analyzers and standard checkers are used to inspect code for deviations from standards and guidelines.

Test Cases:

Test cases are derived to ensure that all statements in the program have been executed at least once during testing and that all logical conditions have been executed.

Using White-Box testing methods, the software engineer can drive test cases that

- Guarantee that logical decisions on their true and false sides.
- Exercise all logical decisions on their true and false sides.
- Execute all loops at their boundaries and with in their operational bounds.
- Exercise internal data structure to assure their validity.
- The test case specification for system testing has to be submitted for review before system testing commences.

Screenshots

Home page

[My Account](#)
[Wishlist](#)
[My Cart](#)
[Checkout](#)
[Login](#)

Track Order

Shopping Portal

Search here...

CART - RS. 00.00

HOME
BOOKS
ELECTRONICS
FURNITURE
FASHION

CATEGORIES

BOOKS

ELECTRONICS

FURNITURE

FASHION

Electronic Appliances

MONEY BACK

30 DAY MONEY BACK GUARANTEE.

FREE SHIPPING

FREE SHIP-ON ORDER OVER RS. 600.00

SPECIAL SALE

ALL ITEMS-SALE UP TO 20% OFF

FEATURED PRODUCTS

Micromax 81cm (32) HD Ready LED TV (32T6175MHQ, 2 x HDMI, 2 x USB)

★★★★☆

Rs. 139900 ~~Rs. 0~~

ADD TO CART

Apple iPhone 6 (Silver, 16 GB)

★★★★☆

Rs. 36990 ~~Rs. 0~~

ADD TO CART

Redmi Note 4 (Gold, 32 GB) (With 3 GB RAM)

★★★★☆

Rs. 10999 ~~Rs. 0~~

ADD TO CART

Lenovo K6 Power (Silver, 32 GB)

★★★★☆

Rs. 9999 ~~Rs. 0~~

ADD TO CART

SMART PHONES

LAPTOPS

Apple iPhone 6 (Silver, 16 GB)

★★★★☆

Rs. 36990 ~~Rs. 0~~

ADD TO CART

Redmi Note 4 (Gold, 32 GB) (With 3 GB RAM)

★★★★☆

Rs. 10999 ~~Rs. 0~~

ADD TO CART

Acer ES 15 Pentium Quad Core

★★★★☆

Rs. 19990 ~~Rs. 0~~

ADD TO CART

Micromax Canvas Laptab II (WIFI) Atom 4th Gen

★★★★☆

Rs. 10999 ~~Rs. 0~~

ADD TO CART

FASHION

Asian Casuals (White, White)

★★★★☆

Rs. 379

ADD TO CART

Adidas MESSI 16.3 TF Football turf Shoes (Blue)

★★★★☆

Rs. 4129

ADD TO CART

OUR BRANDS

Shopping Portal

Nam libero tempore, cum soluta nobis est esse eligendi optio cumque cum soluta nobis est esse eligendi optio cumque.



OPENING TIME

| | |
|----------------|----------------|
| Monday-Friday: | 08.00 To 18.00 |
| Saturday: | 09.00 To 20.00 |
| Sunday: | 10.00 To 20.00 |

INFORMATION

| |
|---------------------|
| New Delhi, Delhi |
| (011) 000000000000 |
| (011) 000000000000 |
| info@phpgurukul.com |

User Registration and login



[Home](#) / [Authentication](#)

SIGN IN

Hello, Welcome to your account.

Email Address *

Password *

LOGIN

[Forgot your Password?](#)

CREATE A NEW ACCOUNT

Create your own Shopping account.

Full Name *

Email Address *

Contact No. *

Password *

Confirm Password. *

SIGN UP

Category

[My Account](#)
[Wishlist](#)
[My Cart](#)
[Checkout](#)
[Login](#)

Track Order

Shopping Portal

Search here...

CART - RS. 00.00

HOMEBOOKSELECTRONICSFURNITUREFASHION

SUB CATEGORIES

LED TELEVISION

TELEVISION

MOBILES

MOBILE ACCESSORIES

LAPTOPS

COMPUTERS

SHOP BY

Category

Books

Electronics

Furniture

Fashion

Category

ELECTRONICS

Micromax 81cm (32) HD Ready LED TV (32T6175MRD, 2 x HDMI, 2 x USB)

★★★★★

Rs. 139900

Add to cart

Apple iPhone 6 (Silver, 16 GB)

★★★★★

Rs. 56999

Add to cart

Redmi Note 4 (Gold, 32 GB) (With 3 GB RAM)

★★★★★

Rs. 10999

Add to cart

Lenovo K6 Power (Silver, 32 GB)

★★★★★

Rs. 9999

Add to cart

Lenovo Vibe K5 Note (Gold, 32 GB)

★★★★★

Rs. 11999

Add to cart

Micromax Canvas Mega 4G

★★★★★

Rs. 6999

Add to cart

SAMSUNG Galaxy On5

★★★★★

Rs. 9999

Add to cart

OPPO A57

★★★★★

Rs. 14999

Add to cart

Affix Back Cover for Mi Redmi Note 4

★★★★★

Rs. 699

Add to cart

Acer ES 15 Pentium Quad Core

★★★★★

Rs. 19990

Add to cart

Micromax Canvas Laptab II (WIFI) Atom 4th Gen

★★★★★

Rs. 10999

Add to cart

HP Core i5 5th Gen

★★★★★

Rs. 41990

Add to cart

Lenovo Ideapad 110 APU Quad Core A6 6th Gen

★★★★★

Rs. 22990

Add to cart

OUR BRANDS

AOC

BAJAJ

BlackBerry

CANON

COMPAQ

DAIKIN

Shopping Portal

Nam libero tempore, cum soluta nobis est ess eligendi optio cumque cum soluta nobis est ess eligendi optio cumque.



OPENING TIME

| | |
|----------------|----------------|
| Monday-Friday: | 08.00 To 18.00 |
| Saturday: | 09.00 To 20.00 |
| Sunday: | 10.00 To 20.00 |

INFORMATION

| |
|------------------------|
| New Delhi, Delhi: |
| (011) 000000000000 |
| (011) 000000000000 |
| info@phpggurunukul.com |

Sub Category

Shopping Portal

Search here...

CART - RS. 00.00

0

HOME

BOOKS

ELECTRONICS

FURNITURE

FASHION

SHOP BY

Category

Books



Electronics



Furniture



Fashion



SubCategory

LAPTOPS



Acer ES 15 Pentium Quad Core

★★★★☆

Rs. 19990 ~~Rs. 0~~

[Add to cart](#) [♥](#)



Micromax Canvas Laptab II (WIFI) Atom 4th Gen

★★★★☆

Rs. 10999 ~~Rs. 0~~

[Add to cart](#) [♥](#)



HP Core i5 5th Gen

★★★★☆

Rs. 41990 ~~Rs. 0~~

[Add to cart](#) [♥](#)



Lenovo Ideapad 110 APU Quad Core A6 6th Gen

★★★★☆

Rs. 22990 ~~Rs. 0~~

[Add to cart](#) [♥](#)

OUR BRANDS



AOC

BAJAJ

BlackBerry

Canon

COMPAQ

DAIKIN

Shopping Portal

Nam libero tempore, cum soluta nobis est esse eligendi optio cumque cum soluta nobis est esse eligendi optio cumque.



OPENING TIME

Monday-Friday 08.00 To 18.00

Saturday 09.00 To 20.00

Sunday 10.00 To 20.00

INFORMATION

New Delhi, Delhi

(011) 000000000000
(011) 000000000000

info@phpgurukul.com

Product Details

Welcome - Anuj Kumar

My Account

Wishlist

My Cart

Checkout

Logout

Track Order

Shopping Portal

Search here...

CART - RS. 00.00

HOMEBOOKSELECTRONICSFURNITUREFASHION

Home / Electronics / Mobiles / Apple iPhone 6 (Silver, 16 GB)

CATEGORY

Books

Electronics

Furniture

Fashion

HOT DEALS

Lenovo Vibe K5 Note (Gold, 32 GB)

★★★★☆

Rs. 11999.00

Add to cart

Apple iPhone 6 (Silver, 16 GB)

★★★★☆

(0 Reviews)

AVAILABILITY : In Stock

PRODUCT BRAND : Apple INC

SHIPPING CHARGE : Free

Rs. 36990

QTY : 1

ADD TO CART

SHARE : f t in s e

DESCRIPTION

REVIEW

1 GB RAM | 16 GB ROM |
4.7 inch Retina HD Display
8MP Primary Camera | 1.2MP Front
Li-Ion Battery
A8 Chip with 64-bit Architecture and M8 Motion Co-processor Processor

REALTED PRODUCTS

Apple iPhone 6 (Silver, 16 GB)

★★★★☆

Rs.36990

Redmi Note 4 (Gold, 32 GB) (With 3 GB RAM)

★★★★☆

Rs.10999

Lenovo K6 Power (Silver, 32 GB)

★★★★☆

Rs.9999

Lenovo Vibe K5 Note (Gold, 32 GB)

★★★★☆

Rs.11999

OUR BRANDS

AOC

BAJAJ

BlackBerry

Canon

COMPAQ

DAIKIN

Shopping Portal

Nam libero tempore, cum soluta nobis est ses eligendi optio cumque cum soluta nobis est ses eligendi optio cumque.



OPENING TIME

| | |
|----------------|----------------|
| Monday-Friday: | 08.00 To 18.00 |
| Saturday: | 09.00 To 20.00 |
| Sunday: | 10.00 To 20.00 |

INFORMATION

- New Delhi, Delhi
- (011) 000000000000
- (011) 000000000000
- info@phpgurukul.com

Forgot Password

Shopping Portal

Search here...



CART - RS. 00.00



HOME

BOOKS

ELECTRONICS

FURNITURE

FASHION

[Home](#) / [Forgot Password](#)

FORGOT PASSWORD

Email Address *

Contact no *

Password *

Confirm Password: *

CHANGE

OUR BRANDS



AOC

BAJAJ

BlackBerry

Canon

COMPAQ

DAIKIN

Shopping Portal

Nam libero tempore, cum soluta nobis est ses eligendi optio cumque cum soluta nobis est ses eligendi optio cumque.



OPENING TIME

| | |
|----------------|----------------|
| Monday-Friday: | 08.00 To 18.00 |
| Saturday: | 09.00 To 20.00 |
| Sunday: | 10.00 To 20.00 |

INFORMATION

- New Delhi, Delhi
- (011) 000000000000
(011) 000000000000
- info@phpgurukul.com

Track Order Without Login

[My Account](#) [Wishlist](#) [My Cart](#) [Checkout](#) [Login](#) [Track Order](#)

Shopping Portal

Search here...

CART - RS. 00.00

HOMEBOOKSELECTRONICSFURNITUREFASHION

Home / [Track your orders](#)

TRACK YOUR ORDER







Please enter your Order ID in the box below and press Enter. This was given to you on your receipt and in the confirmation email you should have received.

Order ID

Registered Email

TRACK

OUR BRANDS



1

Shopping Portal

Nam libero tempore, cum soluta nobis est ses eligendi optio cumque cum soluta nobis est ses eligendi optio cumque.



OPENING TIME

| | |
|----------------|----------------|
| Monday-Friday: | 08.00 To 18.00 |
| Saturday: | 09.00 To 20.00 |
| Sunday: | 10.00 To 20.00 |

INFORMATION

-  New Delhi, Delhi
-  (011) 000000000000
(011) 000000000000
-  info@phpgurukul.com

User Profile

Shopping Portal

Search here...



CART - RS. 00.00



HOME

BOOKS

ELECTRONICS

FURNITURE

FASHION

Home / Checkout

1 MY PROFILE

Personal info

Name*

Anuj Kumar

Email Address *

anuj.lpu1@gmail.com

Contact No. *

9009857868

UPDATE

2 CHANGE PASSWORD

YOUR CHECKOUT PROGRESS

My Account

Shipping / Billing Address

Order History

Payment Pending Order

OUR BRANDS



AOC

BAJAJ

BlackBerry

Canon

COMPAQ

DAIKIN

Shopping Portal

Nam libero tempore, cum soluta nobis est ses eligendi optio cumque cum soluta nobis est ses eligendi optio cumque.



OPENING TIME

| | |
|----------------|----------------|
| Monday-Friday: | 08.00 To 18.00 |
| Saturday: | 09.00 To 20.00 |
| Sunday: | 10.00 To 20.00 |

INFORMATION

- New Delhi, Delhi
- (011) 000000000000
(011) 000000000000
- info@phpgurukul.com

Shipping-Billing Address Details

Shopping Portal

Search here...

CART - RS. 00.00

HOME

BOOKS

ELECTRONICS

FURNITURE

FASHION

[Home](#) / [Checkout](#)

1 BILLING ADDRESS

Billing Address*

New Delhi

Billing State*

New Delhi

Billing City *

Delhi

Billing Pincode *

110092

UPDATE

YOUR CHECKOUT PROGRESS

[My Account](#)

[Shipping / Billing Address](#)

[Order History](#)

[Payment Pending Order](#)

2 SHIPPING ADDRESS

OUR BRANDS

AOC

BAJAJ

BlackBerry

Canon

COMPAQ

DAIKIN

Shopping Portal

Nam libero tempore, cum soluta nobis est ses eligendi optio cumque cum soluta nobis est ses eligendi optio cumque.



OPENING TIME

| | |
|----------------|----------------|
| Monday-Friday: | 08.00 To 18.00 |
| Saturday: | 09.00 To 20.00 |
| Sunday: | 10.00 To 20.00 |

INFORMATION

- New Delhi, Delhi
- (011) 000000000000
(011) 000000000000
- info@phpgurukul.com

Order History

Track Order

Shopping Portal

Search here....



CART - RS. 00.00

0



HOME







BOOKS

ELECTRONICS

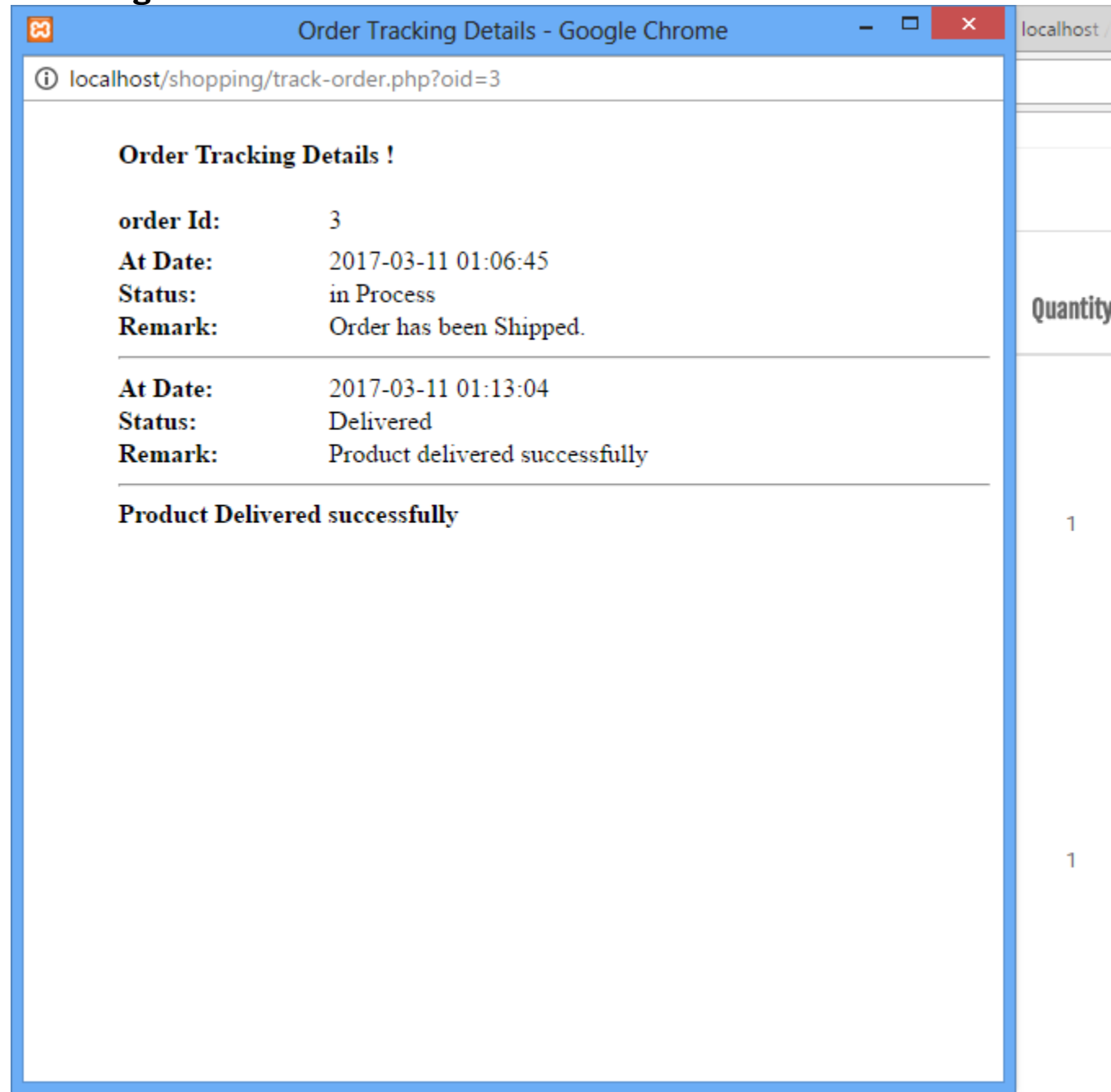
FURNITURE

FASHION

Home / Shopping Cart

| # | Image | Product Name | Quantity | Price Per unit | Grandtotal | Payment Method | Order Date | Action |
|---|---|---|----------|----------------|------------|---------------------|---------------------|-----------------------|
| 1 |  | REDMI NOTE 4 (GOLD, 32 GB) (WITH 3 GB RAM) | 1 | 10999 | 10999 | COD | 2017-03-08 01:02:57 | Track |
| 2 |  | LENOVO K6 POWER (SILVER, 32 GB) | 1 | 9999 | 9999 | Debit / Credit card | 2017-03-11 01:13:04 | Track |
| 3 |  | INDUSCRAFT SOLID WOOD KING BED WITH STORAGE | 1 | 32566 | 32566 | COD | 2017-03-08 21:44:17 | Track |
| 4 |  | REDMI NOTE 4 (GOLD, 32 GB) (WITH 3 GB RAM) | 1 | 10999 | 10999 | COD | 2017-03-09 00:51:38 | Track |
| 5 |  | LENOVO K6 POWER (SILVER, 32 GB) | 1 | 9999 | 9999 | COD | 2017-03-09 00:51:38 | Track |
| 6 |  | THEA STILTON AND THE TROPICAL TREASURE | 2 | 240 | 480 | COD | 2017-04-03 23:38:54 | Track |

Tracking Details



The screenshot shows a Google Chrome browser window titled "Order Tracking Details - Google Chrome". The address bar displays "localhost/shopping/track-order.php?oid=3". The main content area is titled "Order Tracking Details !" and contains two sections of tracking information, separated by horizontal lines.

Order Tracking Details !

| | |
|------------------|-------------------------|
| order Id: | 3 |
| At Date: | 2017-03-11 01:06:45 |
| Status: | in Process |
| Remark: | Order has been Shipped. |

| | |
|-----------------|--------------------------------|
| At Date: | 2017-03-11 01:13:04 |
| Status: | Delivered |
| Remark: | Product delivered successfully |

Product Delivered successfully

Wishlist

Shopping Portal

Search here...

CART - RS. 00.00

HOME

BOOKS

ELECTRONICS

FURNITURE

FASHION

Home / Wishlist

MY WISHLIST



Apple iPhone 6 (Silver, 16 GB)

★★★★★ (0 Reviews)

Rs. 36990.00 ~~\$900.00~~

ADD TO CART



OUR BRANDS



AOC

BAJAJ

BlackBerry

Canon

COMPAQ

DAIKIN

Shopping Portal

Nam libero tempore, cum soluta nobis est ses eligendi optio cumque cum soluta nobis est ses eligendi optio cumque.



OPENING TIME

| | |
|----------------|----------------|
| Monday-Friday: | 08.00 To 18.00 |
| Saturday: | 09.00 To 20.00 |
| Sunday: | 10.00 To 20.00 |

INFORMATION

New Delhi, Delhi

(011) 000000000000
(011) 000000000000

info@phpggurukul.com

FUTURE ENHANCEMENTS

This application avoids the manual work and the problems concern with it. It is an easy way to obtain the information regarding the various products information that are present in the Super markets.

Well I and my team members have worked hard in order to present an improved website better than the existing one's regarding the information about the various activities. Still ,we found out that the project can be done in a better way. Primarily, when we request information about a particular product it just shows the company, product id, product name and no. of quantities available. So, after getting the information we can get access to the product company website just by a click on the product name .

CONCLUSION

The package was designed in such a way that future modifications can be done easily. The following conclusions can be deduced from the development of the project.

- Automation of the entire system improves the efficiency
- It provides a friendly graphical user interface which proves to be better when compared to the existing system.
- It gives appropriate access to the authorized users depending on their permissions.
- It effectively overcomes the delay in communications.
- Updating of information becomes so easier.
- System security, data security and reliability are the striking features.
- The System has adequate scope for modification in future if it is necessary.

BIBLIOGRAPHY

The following books were referred during the analysis and execution phase of the project

PHP and MySQL Web Development

Book by Luke Welling

Head First PHP & MySQL

Book by Lynn Beighley and Michael Morrison

PHP & MySQL for Dummies

Book by Janet Valade

WEBSITES:

- www.google.com
- www.w3schools.com
- www.tutorialspoint.php
- <http://stackoverflow.com>