

**Shanto-Mariam University of Creative Technology**

**Project Report**

On

**“One-Stop Online Shop”**

Submitted in partial

Fulfillment of the requirement

For the degree of

**Bachelor of Science**

In

**Computer Science and Engineering**

Submitted to: Submitted By:

Zahid Hasan Razoana Sultana

Assistant Professor Id#141071013

Department of CSE Salwa Solayman

Id#141071009

Batch#7th

Date:

Accepted by the University

Date Head of the University

The undersigned have examined the Project entitled “One-Stop Online Shop: flairbd.com”.

Presented by Razoana Sultana and Salwa Solayman, a candidate for the degree of Bachelor of Science, Computer Science and Engineering and hereby certify that it is worthy of acceptance.

Date: Adviser’s Name:

Date: Committee Member Name:

Date: Committee Member Name:

**Abstract**

Our One-Stop online shop permits a customer to submit online orders for items from our store. This online shopping system presents an online display of an order cut off time and an associated delivery window for items selected by the customer. The system accepts the customer's submission of a purchase order for the item in response to a time of submission being before the order cut off time. The online shopping system does not settle with a credit supplier of the customer until the item selected by the customer is picked from inventory but before it is delivered. Therefore, the customer can go online and make changes to the order. In addition, available service windows are presented to the customer as a function of customer selected order and service types and further, the order picking is assigned in accordance with a picker's preference. When ordering goods, our shopping systems provide a virtual shopping cart for holding items selected for purchase. Successive items selected for purchase are placed into the virtual shopping cart until a customer completes their shopping trip. Virtual shopping carts may be examined at any time. Once the customer decides to submit a purchase order, the customer may print the contents of the virtual shopping basket in order to obtain a hard copy record of the transaction.

**Acknowledgement**

I have taken efforts in this project. However, it would not have been possible without the kind support and help of many individuals and organizations. I would like to extend my sincere thanks to all of them.

I am highly indebted to Zahid Hasan Sir for his guidance and constant supervision as well as for providing necessary information regarding the project & also for his support in completing the project.

I would like to express my gratitude towards my teachers, my project committee members, who showed immense patience and understanding throughout the project and provided suggestions.

Finally, I would like to dedicate this project to my parents and my friends, for their love, encouragement and help throughout the project.

**Preface**

Learning comes from doing. To learn something one has to go through practical conditions. Recognizing this fact, the university has made it essential for Computer Science and Engineering(CSE) student to undergo the project. By these project we learn about functioning of a project and face the actual environment of a project. This project also helps us to implement the theoretical knowledge into practical life.

To develop this project, an effort was made to understand One-Stop Online Shop and to finish the project work assign to us. This report lays special emphasis on the operational work, tasks and project carried out during the period of project making.

Table of Contents

[Introduction 9](#_Toc526937403)

[What is E-Commerce Business? 10](#_Toc526937404)

[Types of E-Commerce 10](#_Toc526937405)

[**There are 6 basic types of e-commerce:** 10](#_Toc526937406)

[Component of the Project 11](#_Toc526937407)

[How to login 11](#_Toc526937408)

[How to be a member of this Application 11](#_Toc526937409)

[How to view details 11](#_Toc526937410)

[How to give order 11](#_Toc526937411)

[How to pay money 11](#_Toc526937412)

[System Used 11](#_Toc526937413)

[Hardware System 11](#_Toc526937414)

[Windows 12](#_Toc526937415)

[Web server 12](#_Toc526937416)

[Database Server 12](#_Toc526937417)

[phpMyAdmin 12](#_Toc526937418)

[XAMPP information 13](#_Toc526937419)

[Module 13](#_Toc526937420)

[Login Module 13](#_Toc526937421)

[Admin Module 13](#_Toc526937422)

[Shopping Module 13](#_Toc526937423)

[Administration 13](#_Toc526937424)

[INTRODUCTION TO TOOLS 13](#_Toc526937425)

[FRONT-END&BACK-END 13](#_Toc526937426)

[Front End 13](#_Toc526937427)

[Backend 14](#_Toc526937428)

[WHY PHP? 14](#_Toc526937429)

[PHP 14](#_Toc526937430)

[Why MYSQLi 15](#_Toc526937431)

[MYSQLi 16](#_Toc526937432)

[SYSTEM DEVELPOMENT LIFE CYCLE (SDLC) 16](#_Toc526937433)

[Agile Model 16](#_Toc526937434)

[Steges of agile model: 18](#_Toc526937435)

[Stage 1. Requirement gathering 19](#_Toc526937436)

[Stage 2. Planning and requirement analysis 19](#_Toc526937437)

[Stage 3. Designing project architecture 19](#_Toc526937438)

[Stage 4. Development and programming 19](#_Toc526937439)

[Stage 5. Testing 19](#_Toc526937440)

[Stage 6. Deployment 19](#_Toc526937441)

[When to use Agile model: 20](#_Toc526937442)

[Advantages of Agile model: 20](#_Toc526937443)

[PROBLEM ANALYSIS 21](#_Toc526937444)

[Applications 21](#_Toc526937445)

[Challenges 21](#_Toc526937446)

[REQUIREMENT ANALYSIS 21](#_Toc526937447)

[Goal of Projects 21](#_Toc526937448)

[A. Administrator 22](#_Toc526937449)

[B. Database 22](#_Toc526937450)

[C. Clients 22](#_Toc526937451)

[D. Security Constraints 22](#_Toc526937452)

[DATA FLOW DIAGRAM 22](#_Toc526937453)

[*DATA FLOW DIAGRAM OF* ONE-STOP ONLINE SHOP 22](#_Toc526937454)

# Introduction

E-commerce is fast gaining ground as an accepted and used business paradigm. More and more business houses are implementing web sites providing functionality for performing commercial transactions over the web. It is reasonable to say that the process of shopping on the web is becoming common place.

The objective of this project is to develop a general purpose e-commerce store where any product (such as books, CDs, computers, mobile phones, electronic items, and home appliances) can be bought from the comfort of home through the Internet. However, for implementation purposes, this paper will deal with an online multiple store.

An online store is a virtual store on the Internet where customers can browse the catalog and select products of interest. The selected items may be collected in a shopping cart. At checkout time, the items in the shopping cart will be presented as an order. At that time, more information will be needed to complete the transaction. Usually, the customer will be asked to fill or select a billing address, a shipping address, a shipping option, and payment information such as credit card number. An e-mail notification is sent to the customer as soon as the order is placed.

# What is E-Commerce Business?

**E-commerce** also known as e-Business, or electronic business, is simply the sale and purchase of services and goods over an electronic medium, like the Internet. It also involves electronically transferring data and funds between two or more parties. Simply put, it is online shopping as we commonly know it. E-commerce businesses may also employ some or all of the followings:

* [Online shopping](https://en.wikipedia.org/wiki/Online_shopping) for [retail sales direct](https://en.wikipedia.org/wiki/Direct_selling) to consumers via [Web sites](https://en.wikipedia.org/wiki/Web_sites) and [mobile apps](https://en.wikipedia.org/wiki/Mobile_apps), and [conversational commerce](https://en.wikipedia.org/wiki/Conversational_commerce) via [live chat](https://en.wikipedia.org/wiki/Live_chat), [chatbots](https://en.wikipedia.org/wiki/Chatbot" \o "Chatbot) , and [voice assistants](https://en.wikipedia.org/wiki/Voice_assistants).
* Providing or participating in [online marketplaces](https://en.wikipedia.org/wiki/Online_marketplace), which process third-party business-to-consumer or consumer-to-consumer sales
* [Business-to-business](https://en.wikipedia.org/wiki/Business-to-business) buying and selling;
* Gathering and using demographic data through web contacts and social media
* Business-to-business (B2B) [electronic data interchange](https://en.wikipedia.org/wiki/Electronic_data_interchange)
* Marketing to prospective and established customers by e-mail or fax (for example, with [newsletters](https://en.wikipedia.org/wiki/Newsletter))
* Engaging in [pre-tail](https://en.wikipedia.org/wiki/Pretail) for launching new products and services
* Online financial exchanges for currency exchanges or trading purposes.

## Types of E-Commerce

**There are 6 basic types of e-commerce:**

1. Business-to-Business (B2B)
2. Business-to-Consumer (B2C)
3. Consumer-to-Consumer (C2C)
4. Consumer-to-Business (C2B).
5. Business-to-Administration (B2A)
6. Consumer-to-Administration (C2A)

* Business-to-Business (B2B): Electronic transactions of goods and services between companies.
* Business-to-Consumer (B2C): Electronic transactions of goods and services between companies and consumers.
* Consumer-to-Consumer (C2C): Electronic transactions of goods and services between consumers, mostly through a third party.
* Consumer-to-Business (C2B): Electronic transactions of goods and services where individuals offer products or services to companies.
* Business-to-Administration (B2A): Electronic transactions of goods and services between companies and public administrations.
* Consumer-to-Administration (C2A): Electronic transactions of goods and services between individuals and public administrations.

# Component of the Project

## How to login

In this module , the user will enter his username and password to view and buy latest products. There will be 2 types of users (Administrator and Customer).

## How to be a member of this Application

In this site, the candidate can join this application, if he is not a member yet by pressing sign up link. User should provide some details that are asked to join.

## How to view details

Any products can be searched by selecting the name or image which the user wish to view or buy.

## How to give order

First the user has to go to our application . When the user visit our page they can see our product on our page. There he will select the products he wants to buy and then he will click on Add to cart. There he can purchase that products.

## How to pay money

The user have to pay cash on delivery.

# System Used

## Hardware System

|  |  |
| --- | --- |
| **Number** | **Description** |
| 1 | Intel Core i5 |
| 2 | 20 MB RAM |

## Windows

Windows 7 home premium

## Web server

|  |  |
| --- | --- |
| **Number** | **Description** |
| 1 | Apache/2.4.29 (Win32) OpenSSL/1.1.0g PHP/7.2.3 |
| 2 | Database client version: libmysql - mysqlnd 5.0.12-dev - 20150407 - $Id: 38fea24f2847fa7519001be390c98ae0acafe387 $ |
| 3 | PHP extension: mysqli[Documentation](http://localhost/phpmyadmin/url.php?url=https://secure.php.net/manual/en/book.mysqli.php) curl[Documentation](http://localhost/phpmyadmin/url.php?url=https://secure.php.net/manual/en/book.curl.php) mbstring[Documentation](http://localhost/phpmyadmin/url.php?url=https://secure.php.net/manual/en/book.mbstring.php) |
| 4 | PHP version: 7.2.3 |

## Database Server

|  |  |
| --- | --- |
| **Number** | **Description** |
| 1 | Server: 127.0.0.1 via TCP/IP |
| 2 | Server type: MariaDB |
| 3 | Server version: 10.1.31-MariaDB - mariadb.org binary distribution |
| 4 | Protocol version: 10 |
| 5 | User: root@localhost |
| 6 | Server charset: UTF-8 Unicode (utf8) |

## phpMyAdmin

Version information : 4.7.9

## XAMPP information

Version 3.2.2

# Module

## Login Module

In this module user can signup into the application and take offer in the next transection.

## Admin Module

Admin can add , modify and delete the latest verities of goods.

## Shopping Module

The customer can view and buy latest verities of goods.

## Administration

Can add , modify and delete the goods details.

# INTRODUCTION TO TOOLS

## FRONT-END&BACK-END

**Front-End:** Web Pages using PHP, HTML.

**Back-End:** MYSQLi

## Front End

**HTML** –It is used to generate web page. HTML, an initialize of Hypertext

Markup Language, is the predominant markup language for web pages. It

provides a means to describe the structure of text-based information in a

document — by denoting certain text as headings, paragraphs, lists, and so

on.

**PHP-** PHP is a technology that lets you mix regular, static HTML with

dynamically-generated HTML. Many Web pages that are built by CGI

programs are mostly static, with the dynamic part limited to a few small

locations. But most CGI variations, including servlets, make you generate

the entire page via your program, even though most of it is always the same.

## Backend

MySQLi:

The **MySQLi** Extension (MySQL Improved) is a relational database driver used in the PHP scripting language to provide an interface with MySQL databases. **My SQL** is a relational database management system (RDBMS) which has

more than 6 million installations. MySQL stands for "My Structured Query

Language". The program runs as a server providing multi-user access to a

number of databases.

XAMPP:

**XAMPP**- Apache is a web container, or application server

developed at the Apache Software Foundation (ASF).It adds tools for

configuration and management but can also be configured by editing

configuration files that are normally XML-formatted. Apache includes its

own internal HTTP server.

# WHY PHP?

**PHP** is a widely used, general-purpose scripting language that was originally

designed for web development, to produce dynamic web pages. It can be

embedded into HTML and generally runs on a web server, which needs to be

configured to process PHP code and create web page content from it. It can be

deployed on most web servers and on almost every operating system and platform

free of charge. PHP is installed on over 20 million websites and 1 million web

servers.

PHP was originally created by **Rasmus Lerdorf** in 1994 and has been in

continuous development ever since. The main implementation of PHP is now

produced by **The PHP Group** and serves as the *de facto* standard for PHP as there

is no formal specification. PHP is free software released under the PHP License,

which is incompatible with the GNU General Public License (GPL) because of

restrictions on the use of the term *PHP*.

PHP has evolved to include a command line interface capability and can also be

used in standalone graphical applications.

## PHP

**PHP: Hypertext Preprocessor** (or simply **PHP**) is a [server-side scripting](https://en.wikipedia.org/wiki/Server-side_scripting) language designed for [Web development](https://en.wikipedia.org/wiki/Web_development), but also used as a [general-purpose programming language](https://en.wikipedia.org/wiki/General-purpose_programming_language). It was originally created by [Rasmus Lerdorf](https://en.wikipedia.org/wiki/Rasmus_Lerdorf" \o "Rasmus Lerdorf) in 1994, the PHP [reference implementation](https://en.wikipedia.org/wiki/Reference_implementation) is now produced by The PHP Group. PHP originally stood for *Personal Home Page*, but it now stands for the [recursive acronym](https://en.wikipedia.org/wiki/Recursive_acronym) *PHP: Hypertext Preprocessor*.

PHP code may be embedded into [HTML](https://en.wikipedia.org/wiki/HTML) code, or it can be used in combination with various [web template systems](https://en.wikipedia.org/wiki/Web_template_system), web content management systems, and [web frameworks](https://en.wikipedia.org/wiki/Web_framework). PHP code is usually processed by a PHP [interpreter](https://en.wikipedia.org/wiki/Interpreter_(computing)) implemented as a [module](https://en.wikipedia.org/wiki/Plugin_(computing)) in the web server or as a [Common Gateway Interface](https://en.wikipedia.org/wiki/Common_Gateway_Interface) (CGI) executable. The web server combines the results of the interpreted and executed PHP code, which may be any type of data, including images, with the generated web page. PHP code may also be executed with a [command-line interface](https://en.wikipedia.org/wiki/Command-line_interface) (CLI) and can be used to implement [standalone](https://en.wikipedia.org/wiki/Computer_software) [graphical applications](https://en.wikipedia.org/wiki/Graphical_user_interface).

The standard PHP interpreter, powered by the [Zend Engine](https://en.wikipedia.org/wiki/Zend_Engine" \o "Zend Engine), is [free software](https://en.wikipedia.org/wiki/Free_software) released under the [PHP License](https://en.wikipedia.org/wiki/PHP_License). PHP has been widely ported and can be deployed on most web servers on almost every [operating system](https://en.wikipedia.org/wiki/Operating_system) and [platform](https://en.wikipedia.org/wiki/Computing_platform), free of charge.

The PHP language evolved without a written [formal specification](https://en.wikipedia.org/wiki/Formal_specification) or standard until 2014, with the original implementation acting as the [*de facto*](https://en.wikipedia.org/wiki/De_facto) standard which other implementations aimed to follow. Since 2014 work has gone on to create a formal PHP specification.

During the 2010s there have been increased efforts towards standardization and code sharing in PHP applications by projects such as [PHP-FIG](http://www.php-fig.org/) in the form of [PSR initiatives](http://www.php-fig.org/psr/) as well as the [Composer dependency manager](https://en.wikipedia.org/wiki/Composer_(software)) and associated [Packagist repository](https://packagist.org/). PHP hosts a diverse array of web frameworks requiring framework-specific knowledge, with [Laravel](https://en.wikipedia.org/wiki/Laravel" \o "Laravel) recently emerging as a popular option by incorporating ideas made popular from other competing non-PHP web frameworks, like [Ruby on Rails](https://en.wikipedia.org/wiki/Ruby_on_Rails).

# Why MYSQLi

[PHP](http://www.php.net/) supports MySQL using a PHP extension. Thousands or millions projects have been written worldwide using PHP and MySQL. PHP team announced plans to deprecate [MySQL extension](http://php.net/manual/en/book.mysql.php) in [mid 2011](http://news.php.net/php.internals/53799" \t "_blank). Old MySQL extension [officially deprecated](https://wiki.php.net/rfc/mysql_deprecation) since PHP 5.5.0 in late 2012 and it will be removed in the future. The alternatives since PHP 5 and later are [MySQLi](http://php.net/manual/en/book.mysqli.php" \t "_blank) ("i" stands from "improved") and [PDO](http://php.net/manual/en/book.pdo.php) (PHP Data Objects).

Everyone can understand that is not so easy to immediately migrate old projects. However, the old extension **must not be used anymore in new development.**

Old extension didn't support Prepared Statements. Both MySQLi and PDO are object oriented and do support Prepared Statements (also support Transactions, Stored Procedures and more). Prepared Statements are very important for web application security, as they protect from SQL injection. Using Prepared Statements you do not have to escape strings before insert them in Database. Moreover, PDO offers support for many databases (not only MySQL).

So, the question is "Which Should I Use? MySQLi or PDO?". A short answer would be "whatever you like". Personally, I prefer MySQLi. I could select PDO if "multiple database support" was included in project requirements (however, in this case [php ADODB](https://www.pontikis.net/blog/how-to-write-code-for-any-database-with-php-adodb) could also be an alternative)

## MYSQLi

The **MySQLi Extension** ([MySQL](https://en.wikipedia.org/wiki/MySQL) Improved) is a [relational database](https://en.wikipedia.org/wiki/Relational_database) driver used in the [PHP](https://en.wikipedia.org/wiki/PHP) [scripting language](https://en.wikipedia.org/wiki/Scripting_language) to provide an interface with [MySQL](https://en.wikipedia.org/wiki/MySQL) [databases](https://en.wikipedia.org/wiki/Database).

There are three main API options when considering connecting to a MySQL database server:

* PHP's MySQL Extension
* PHP's MySQLi Extension
* PHP Data Objects (PDO)

The PHP code consists of a core, with optional extensions to the core functionality. PHP's MySQL-related extensions, such as the MySQLi extension, and the MySQL extension, are implemented using the PHP extension framework. An extension typically exposes an API to the PHP developer, to allow its facilities to be used programmatically. However, some extensions which use the PHP extension framework do not expose an API to the PHP developer.

The PDO MySQL driver extension, for example, does not expose an API to the PHP developer, but provides an interface to the PDO layer above it.

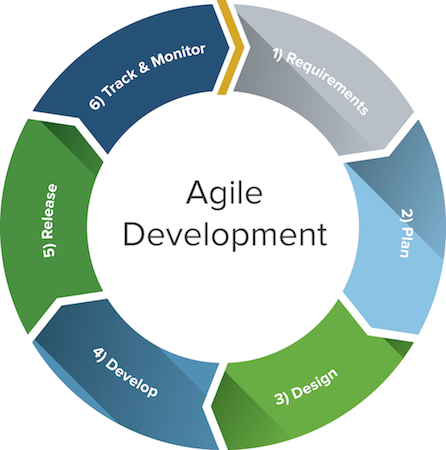
MySQLi is an improved version of the older PHP MySQL driver, offering various benefits.[[1]](https://en.wikipedia.org/wiki/MySQLi#cite_note-overview-1)

The authors of the PHP scripting language recommend using MySQLi when dealing with MySQL server versions 4.1.3 and newer (takes advantage of new functionality).

# SYSTEM DEVELPOMENT LIFE CYCLE (SDLC)

The software development life cycle (SDLC) is a framework defining tasks performed at each step in the software development process. SDLC is a structure followed by a development team within the software organization. It consists of a detailed plan describing how to develop, maintain and replace specific software. The life cycle defines a methodology for improving the quality of software and the overall development process.

## Agile Model



Agile model believes that every project needs to be handled differently and the existing methods need to be tailored to best suit the project requirements. In Agile, the tasks are divided to time boxes (small time frames) to deliver specific features for a release.

Iterative approach is taken and working software build is delivered after each iteration. Each build is incremental in terms of features; the final build holds all the features required by the customer.

Here is a graphical illustration of the Agile Model −



The Agile thought process had started early in the software development and started becoming popular with time due to its flexibility and adaptability.

# Steges of agile model:

Agile SDLC model is a combination of iterative and incremental process models with focus on process adaptability and customer satisfaction by rapid delivery of working software product. Agile Methods break the product into small incremental builds. These builds are provided in iterations. Each iteration typically lasts from about one to three weeks. Every iteration involves cross functional teams working simultaneously on various areas like −

* Requirement gathering
* Requirement Analysis
* Design
* Coding
* Unit testing
* Acceptance testing

## Stage 1. Requirement gathering

It is the first stage of our software development cycle model. In this stage we gather the requirement by observing many website and by asking the stakeholders. The goal of the project is to collect all the possible requirement for the product.

## Stage 2. Planning and requirement analysis

Each software development life cycle model developed with the analysis, in which the stakeholders of the process discuss the requirements for the final product. The goal of this stage is the detailed definition of the system requirements. We make sure that all the process participants have clearly understood the tasks and how every requirement is going to be implemented. The discussion involves the QA specialists who can interfere the process with additions even during the development stage if it is necessary.

## Stage 3. Designing project architecture

At the third phase of the software development life cycle, we design the architecture. All the different technical questions that may appear on this stage are discussed by all the stakeholders, including the customer. Also, here we define the technologies used in the project, team load, limitations, time frames, and budget. The most appropriate project decisions are made according to the defined requirements.

## Stage 4. Development and programming

After the requirements approved, the process goes to the next stage – actual development. Here we start the source code writing while keeping in mind previously defined requirements. We adjust the software environment, develop the user interface of the program and the logics for its interaction with the server.  
The programming by itself assumes four stages

Algorithm development

Source code writing

Compilation

Testing and debugging

## Stage 5. Testing

The testing phase includes the debugging process. All the code flaws missed during the development are detected here, documented, and passed back to the developers to fix. The testing process repeats until all the critical issues are removed and software workflow is stable.

Whereas we are still updating the project ,some of the part of our project may faulty here.

## Stage 6. Deployment

When the program is finalized and has no critical issues – it is time to launch it for the end users. After the new program version release, the tech support team joins. This department provides user feedback; consult and support users during the time of exploitation. Moreover, the update of selected components is included in this phase, to make sure, that the software is up-to-date and is invulnerable to a security breach.

## ****When to use Agile model:****

* When new changes are needed to be implemented. The freedom agile gives to change is very important. New changes can be implemented at very little cost because of the frequency of new increments that are produced.
* To implement a new feature the developers need to lose only the work of a few days, or even only hours, to roll back and implement it.
* Unlike the [waterfall model](http://tryqa.com/what-is-waterfall-model-advantages-disadvantages-and-when-to-use-it/) in agile model very limited [planning](http://tryqa.com/what-is-the-purpose-and-importance-of-test-plans/) is required to get started with the project. Agile assumes that the end users’ needs are ever changing in a dynamic business and IT world. Changes can be discussed and features can be newly effected or removed based on feedback. This effectively gives the customer the finished system they want or need.
* Both system developers and stakeholders alike, find they also get more freedom of time and options than if the software was developed in a more rigid sequential way. Having options gives them the ability to leave important decisions until more or better data or even entire hosting programs are available; meaning the project can continue to move forward without fear of reaching a sudden standstill.

## Advantages of Agile model:

* Customer satisfaction by rapid, continuous delivery of useful software.
* People and interactions are emphasized rather than process and tools. Customers, developers and testers constantly interact with each other.
* Working software is delivered frequently (weeks rather than months).
* Face-to-face conversation is the best form of communication.
* Close, daily cooperation between business people and developers.
* Continuous attention to technical excellence and good design.
* Regular adaptation to changing circumstances.
* Even late changes in requirements are welcomed

# PROBLEM ANALYSIS

## Applications

The main applications of the On Line Placement System is the ability of the

website to properly show enroll the artists and manage information about them.

The administrator has the ability to change ,modify, view and delete the various

details regarding the users and arts. The users have the ability to log in and post

their queries and download arts.

## Challenges

The challenges mainly lie in detecting attacks like viruses, hacking and also in the

implementation of firewall. A virus can enter the system and can disrupt the

working of the website. Hacking can be done by some people who want to access

some restricted sections of the website (e.g. administrator’s area) and to modify or

taper some aspects of the website.

Scanning attacks may yield:

(i) The method used by viruses to enter the system.

(ii) The types of database allowed through a firewall.

(iii) The paths or ways used by hackers to enter the system

(iv) The loopholes remaining in the system (or website) which are used by

attackers.

(v) The server from where the viruses or hackers are gaining access to the system.

(vi) The types of viruses able to affect the website.

And with the implementation of firewall and other security mechanisms that are

designed for it, the On Line Placement System Website safe and secure.

# REQUIREMENT ANALYSIS

## Goal of Projects

The goal of our thesis is to develop a website that can be used as an enrollment

website with the features of interaction and problem solving. The whole project

will be based on PHP with MYSQL as the database with certain security

constraints added to it.

Our aim is also to implement the Administrator part in to the project so that the

server or administrator himself can view, add, delete and modify.

## A. Administrator

He has to see whether the website is working properly and whether the details

available in the system are relevant and correct. He can view, add, modify, delete

details.

## B. Database

The database keeps all the records of all the users i.e. name ,course, phone no. ,dob,

city, country ,etc. For creating such records it takes the help of tables which is

created in the MYSQL. The tables can have infinite entries of all the registered

users as well as administrators.

## C. Clients

Our aim will also to provide efficient way by which client can enter to see his

profile, ask questions to his teacher and download softwares.

## D. Security Constraints

There need to be certain constraints which have to be implemented on the database

as well as on the administrator in order to work properly the whole system, such as

declaring the primary key, or such constraints in order to keep the database work

properly.

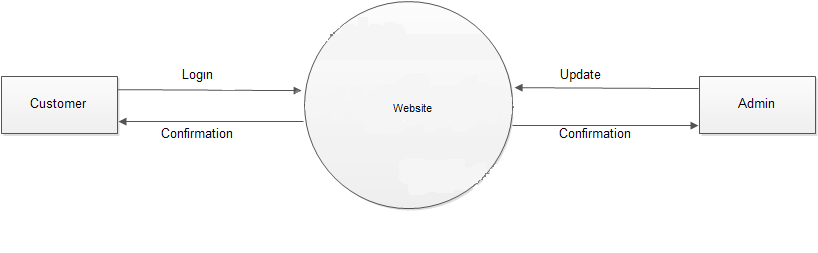
# DATA FLOW DIAGRAM

# *DATA FLOW DIAGRAM OF* ONE-STOP ONLINE SHOP

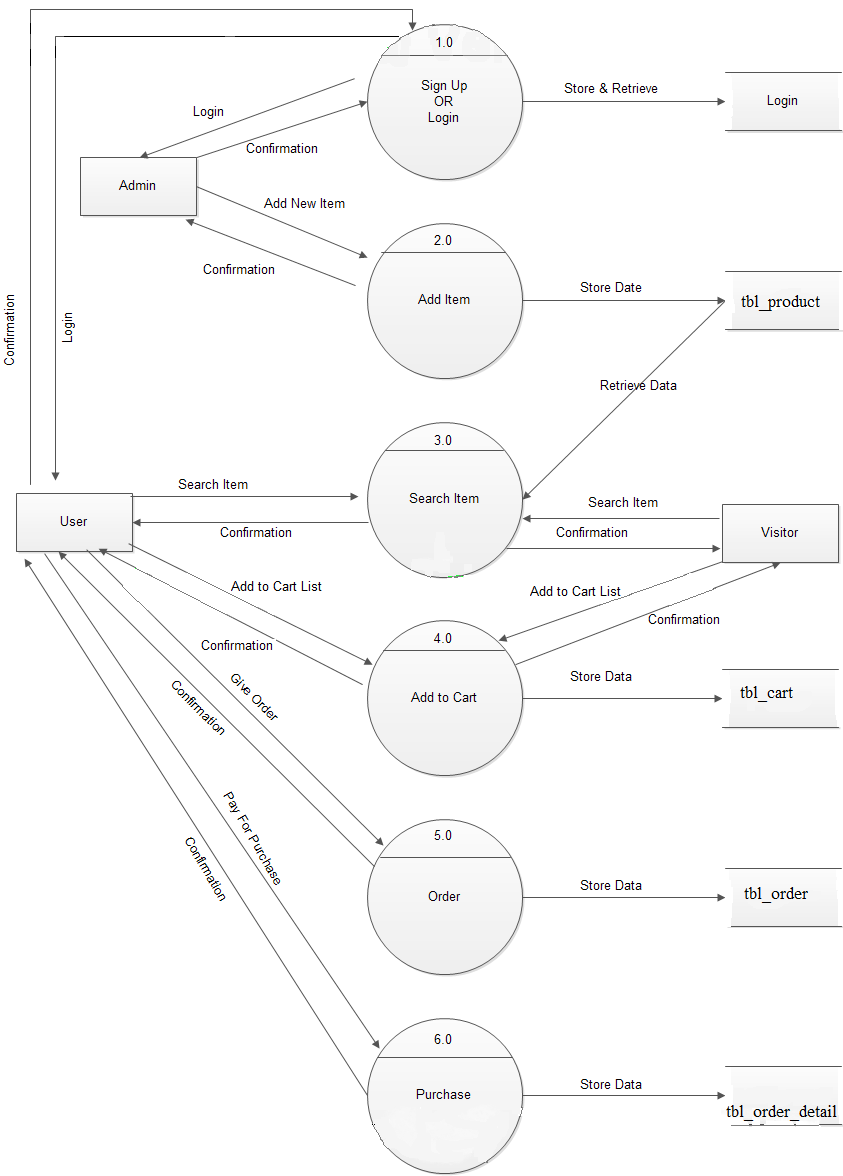
A picture is worth a thousand words. A Data Flow Diagram (DFD) is traditional visual representation of the information flows within a system. A neat and clear DFD can depict a good amount of the system requirements graphically. It can be manual, automated, or combination of both.

It shows how information enters and leaves the system, what changes the information and where information is stored. The purpose of a DFD is to show the scope and boundaries of a system as a whole. It may be used as a communications tool between a systems analyst and any person who plays a part in the system that acts as the starting point for redesigning a system.

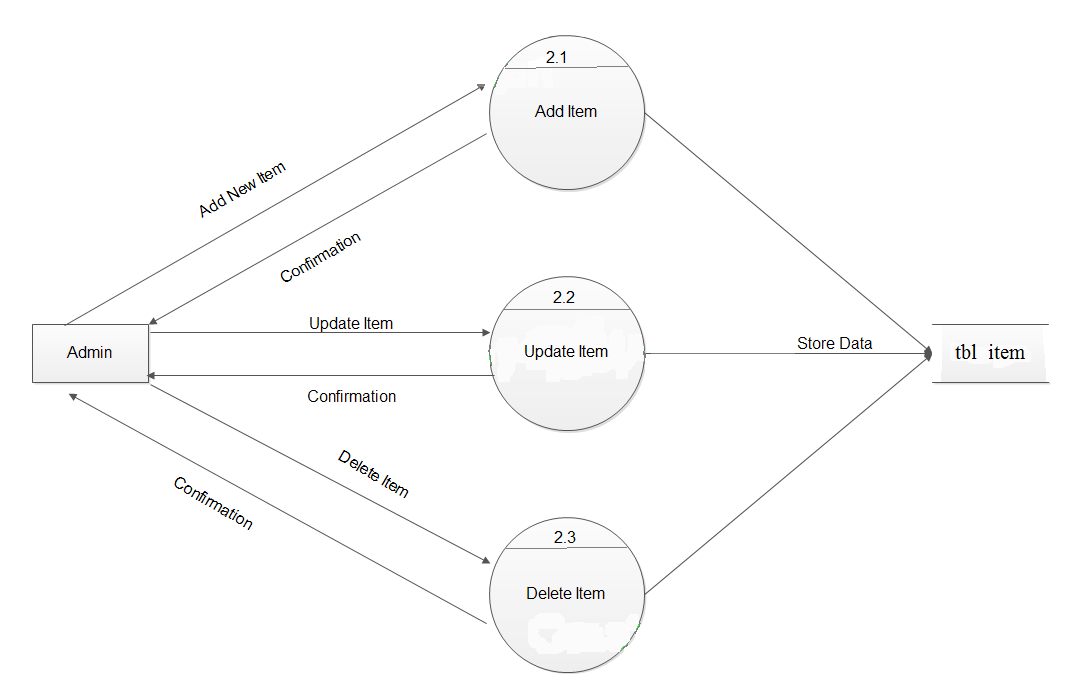
It is usually beginning with a context diagram as the level 0 of DFD diagram, a simple representation of the whole system. To elaborate further from that, we drill down to a level 1 diagram with lower level functions decomposed from the major functions of the system. This could continue to evolve to become a level 2 diagram when further analysis is required. Progression to level 3, 4 and so on is possible but anything beyond level 3 is not very common. Please bear in mind that the level of details for decomposing particular function really depending on the complexity that function.



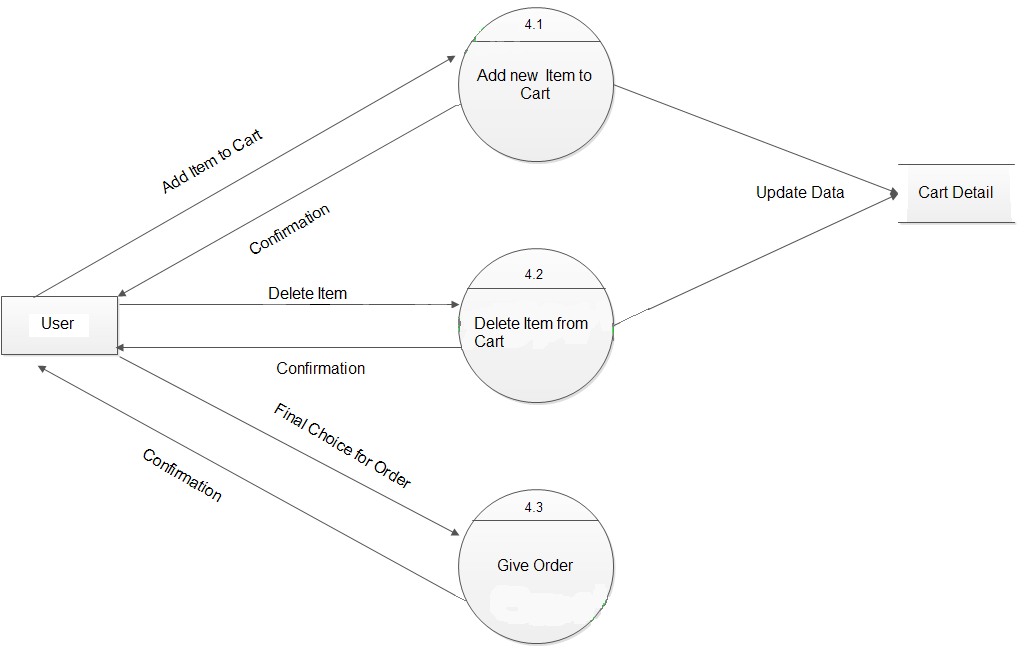
Level-0



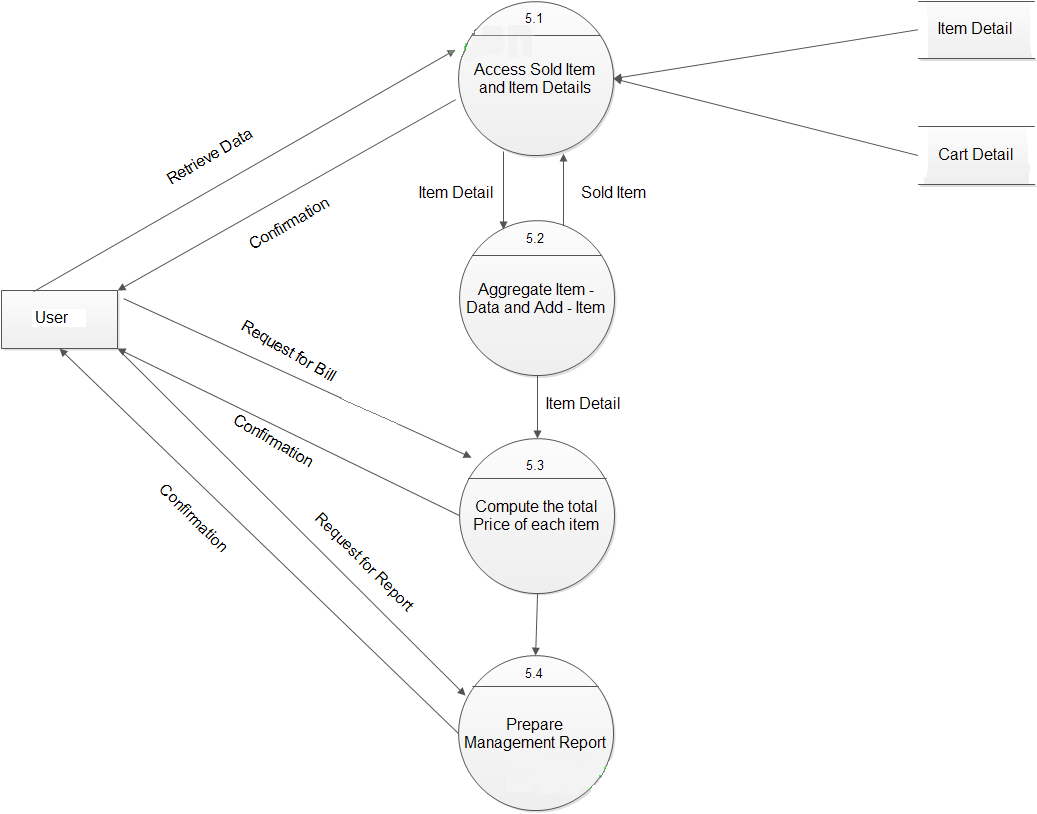
Level-1

****

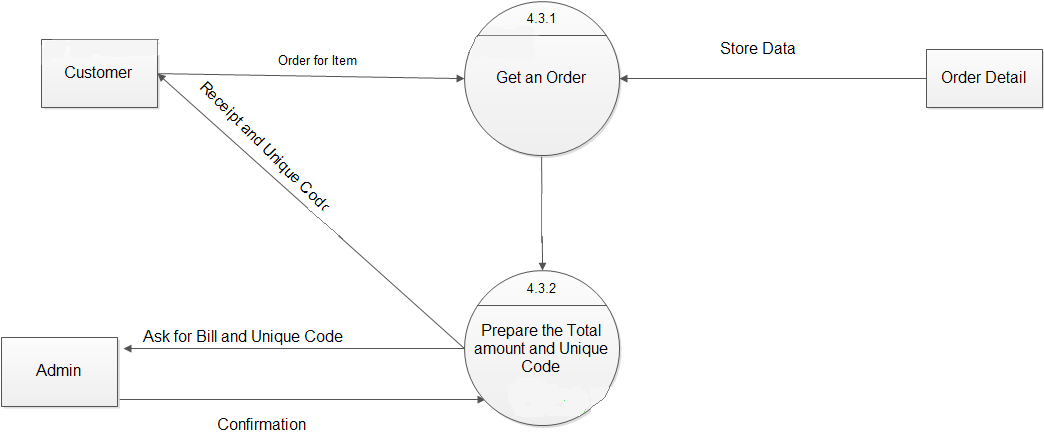
Level-2 for 2.0



Level-2 for 4.0



Level-2 for 5.0



Level-3 for 4.3

# ER Diagram

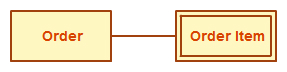
An entity relationship diagram (ERD) shows the relationships of entity sets stored in a database. An entity in this context is an object, a component of data. An entity set is a collection of similar entities. These entities can have attributes that define its properties.

By defining the entities, their attributes, and showing the relationships between them, an ER diagram illustrates the logical structure of databases.

ER diagrams are used to sketch out the design of a database.

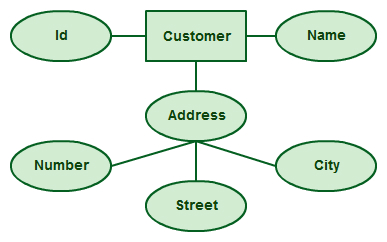
## Entity

Entities are represented by means of rectangles. Rectangles are named with the entity set they represent.



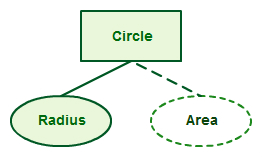
## Attributes

Attributes are the properties of entities. Attributes are represented by means of ellipses. Every ellipse represents one attribute and is directly connected to its entity (rectangle).



If the attributes are **composite**, they are further divided in a tree like structure. Every node is then connected to its attribute. That is, composite attributes are represented by ellipses that are connected with an ellipse.

**Multivalued** attributes are depicted by double ellipse.



**Derived** attributes are depicted by dashed ellipse.

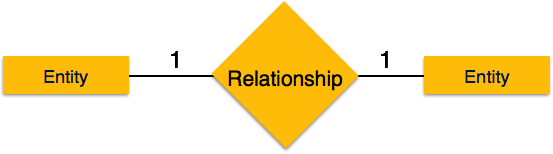
## Relationship

Relationships are represented by diamond-shaped box. Name of the relationship is written inside the diamond-box. All the entities (rectangles) participating in a relationship, are connected to it by a line.

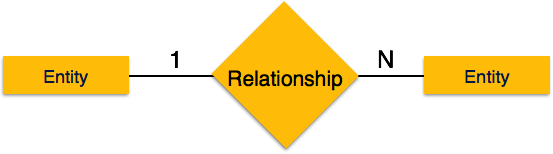
## Binary Relationship and Cardinality

A relationship where two entities are participating is called a **binary relationship**. Cardinality is the number of instance of an entity from a relation that can be associated with the relation.

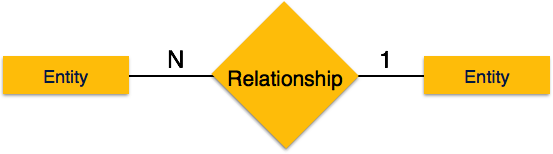
* **One-to-one** − When only one instance of an entity is associated with the relationship, it is marked as '1:1'. The following image reflects that only one instance of each entity should be associated with the relationship. It depicts one-to-one relationship.



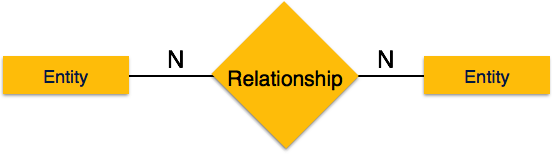
* **One-to-many** − When more than one instance of an entity is associated with a relationship, it is marked as '1:N'. The following image reflects that only one instance of entity on the left and more than one instance of an entity on the right can be associated with the relationship. It depicts one-to-many relationship.

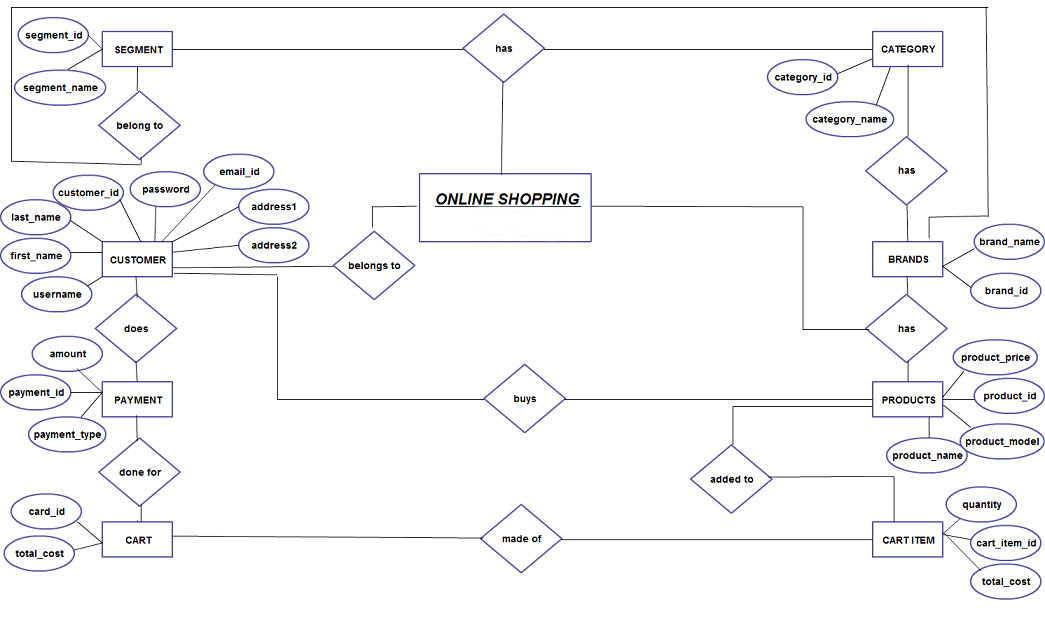


* **Many-to-one** − When more than one instance of entity is associated with the relationship, it is marked as 'N:1'. The following image reflects that more than one instance of an entity on the left and only one instance of an entity on the right can be associated with the relationship. It depicts many-to-one relationship.



* **Many-to-many** − The following image reflects that more than one instance of an entity on the left and more than one instance of an entity on the right can be associated with the relationship. It depicts many-to-many relationship.



****

# Primary Key and Foreign Key

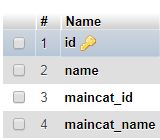
## Primary key

In the [relational model](https://en.wikipedia.org/wiki/Relational_model) of [databases](https://en.wikipedia.org/wiki/Database), a **primary key** is a *specific choice* of a *minimal* set of attributes ([columns](https://en.wikipedia.org/wiki/Column_(database))) that uniquely specify a tuple ([row](https://en.wikipedia.org/wiki/Row_(database))) in a [relation](https://en.wikipedia.org/wiki/Relation_(database)) ([table](https://en.wikipedia.org/wiki/Table_(database))). Informally, a primary key is "which attributes identify a record", and in simple cases are simply a single attribute: a unique id. More formally, a primary key is a choice of [candidate key](https://en.wikipedia.org/wiki/Candidate_key) (a minimal [super key](https://en.wikipedia.org/wiki/Superkey)); any other candidate key is an **alternate key**.

A primary key may consist of real-world observables, in which case it is called a [*natural key*](https://en.wikipedia.org/wiki/Natural_key), while an attribute created to function as a key is called a [*surrogate key*](https://en.wikipedia.org/wiki/Surrogate_key). For example, for a database of people (of a given nationality), time and location of birth is a natural key, while [national identification numbers](https://en.wikipedia.org/wiki/National_identification_number) are surrogate keys.

## Foreign key

In the context of relational databases, a foreign key is a field (or collection of fields) in one table that uniquely identifies a row of another table or the same table. In simpler words, the foreign key is defined in a second table, but it refers to the primary key or a unique key in the first table.

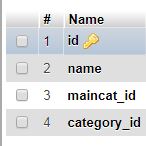
 category Table

product Table

 Primary key

Foreign key

sub\_category Table

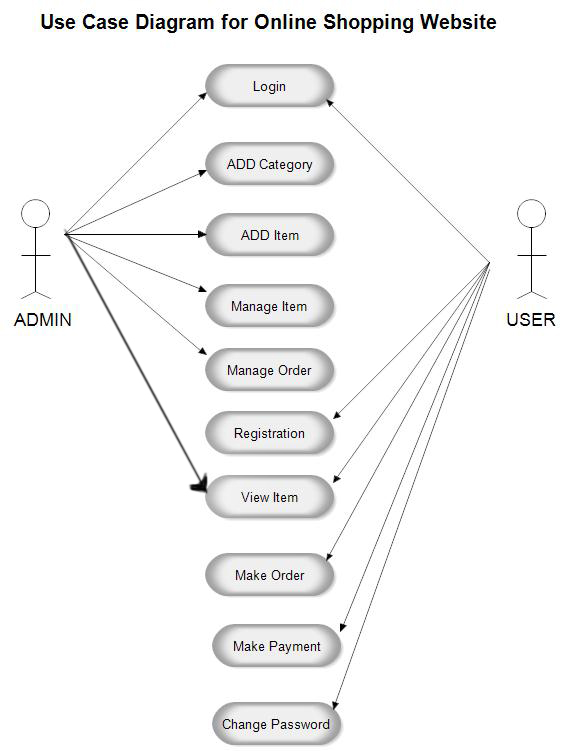


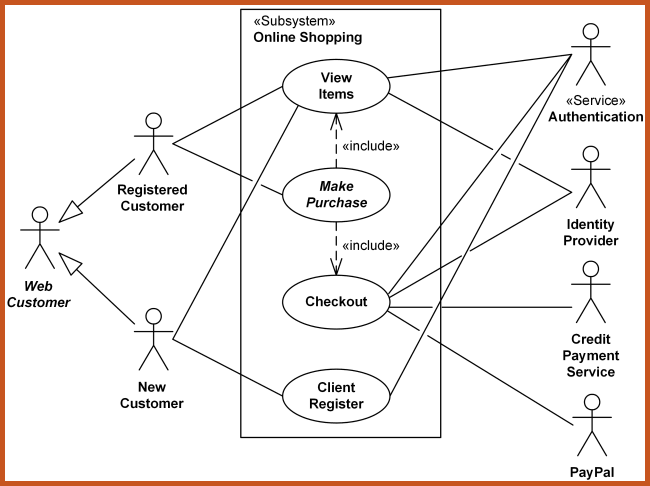
# UML Diagram

The Unified Modeling Language (UML) was created to forge a common, semantically and syntactically rich visual modeling language for the architecture, design, and implementation of complex software systems both structurally and behaviorally. UML has applications beyond software development, such as process flow in manufacturing.

It is analogous to the blueprints used in other fields, and consists of different types of diagrams. In the aggregate, UML diagrams describe the boundary, structure, and the behavior of the system and the objects within it.

UML is not a programming language but there are tools that can be used to generate code in various languages using UML diagrams. UML has a direct relation with object-oriented analysis and design.

****

****

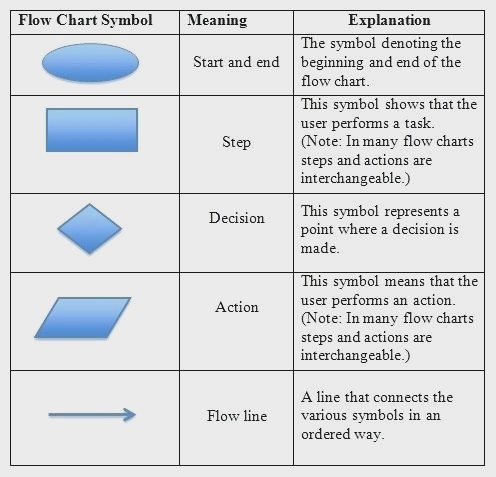
# Project Design

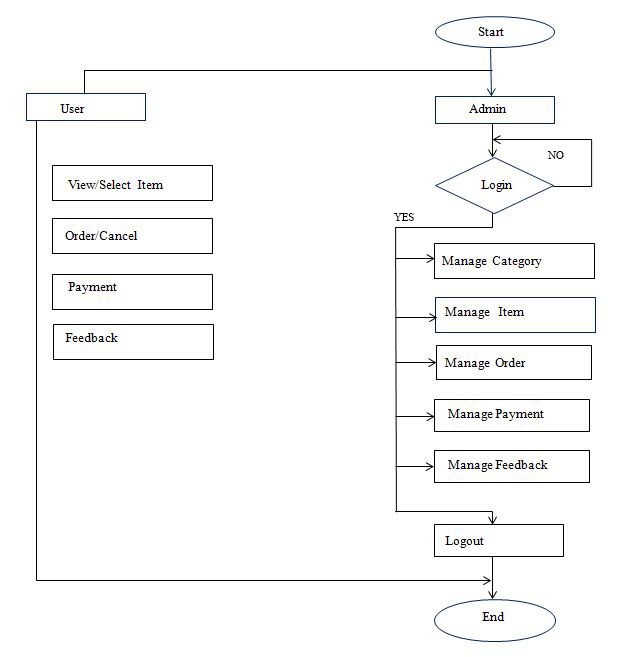
In order to design a web site, the relational database must be designed first. Conceptual design can be divided into two parts: The data model and the process model. The data model focuses on what data should be stored in the database while the process model deals with how the data is processed. To put this in the context of the relational database, the data model is used to design the relational tables. The process model is used to design the queries that will access and perform operations on those tables.

## Flow Chart:

A flowchart is a diagram that depicts a process, system or computer algorithm. They are widely used in multiple fields to document, study, plan, improve and communicate often complex processes in clear, easy-to-understand diagrams. Flowcharts, sometimes spelled as flow charts, use rectangles, ovals, diamonds and potentially numerous other shapes to define the type of step, along with connecting arrows to define flow and sequence. They can range from simple, hand-drawn charts to comprehensive computer-drawn diagrams depicting multiple steps and routes. If we consider all the various forms of flowcharts, they are one of the most common diagrams on the planet, used by both technical and non-technical people in numerous fields.

## Flow chart Symbols





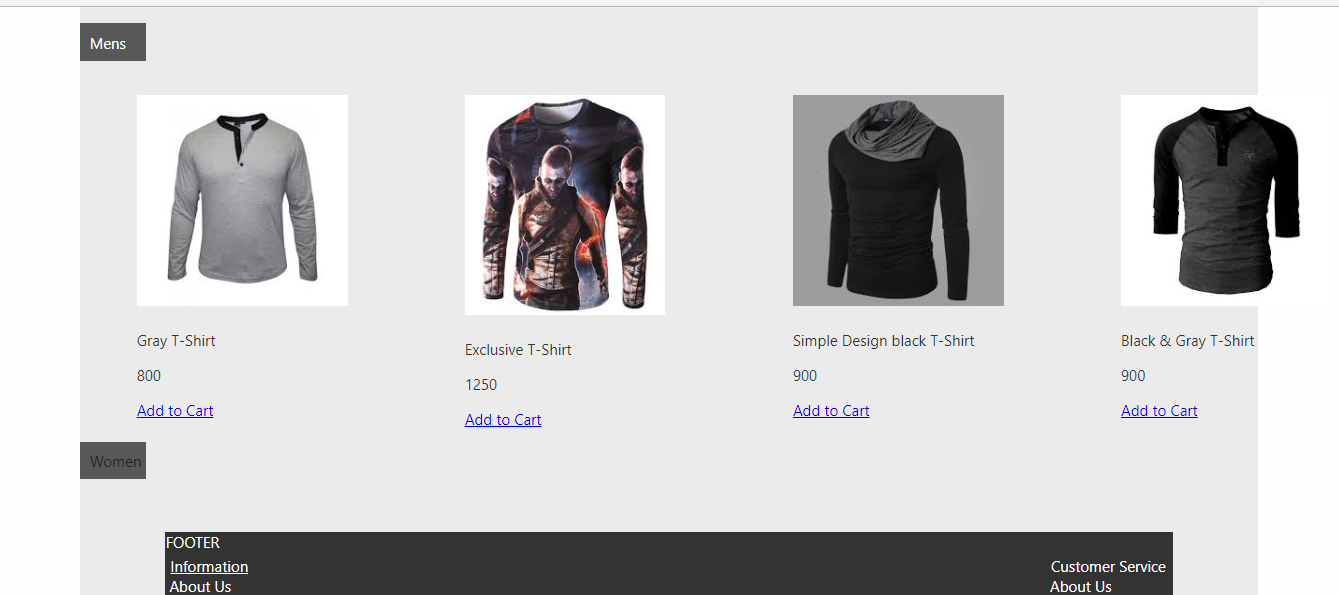
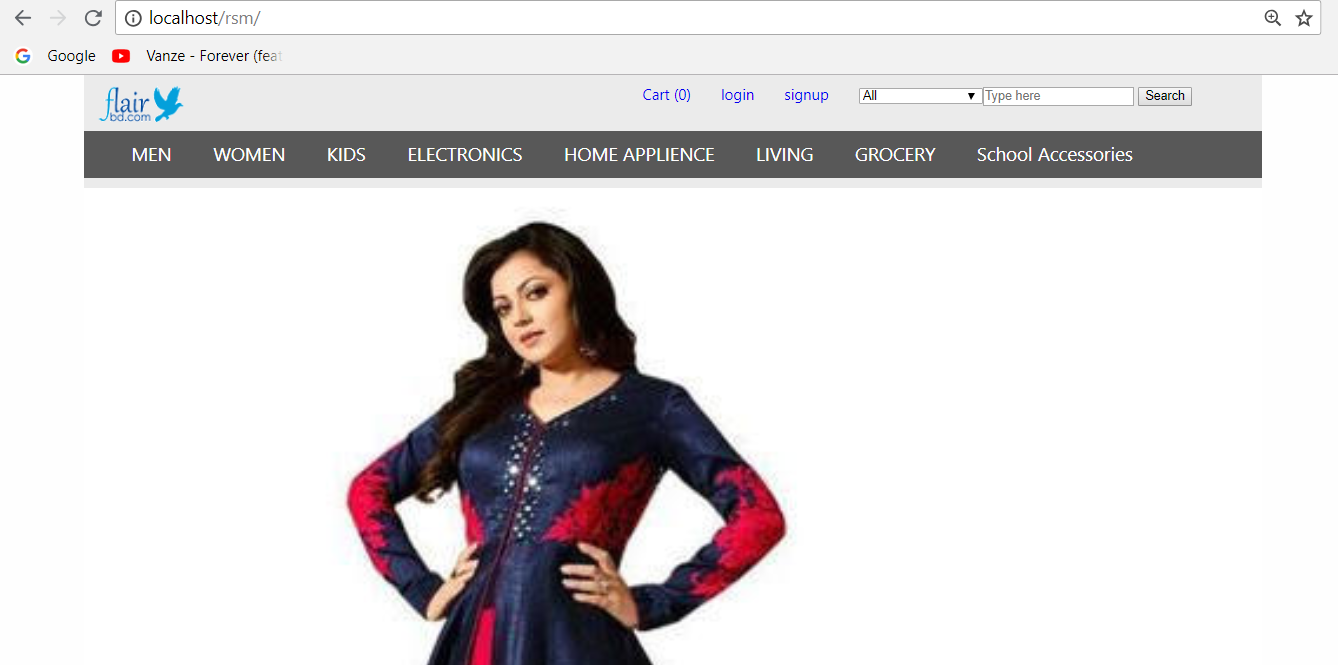
## Front End and Back End:

Front-end and back-end are terms used to characterize program interfaces and services relative to the initial user of these interfaces and services. (The "user" may be a human being or a program.) A "front-end" application is one that application users interact with directly. A "back-end" application or program serves indirectly in support of the front-end services, usually by being closer to the required resource or having the capability to communicate with the required resource. The back-end application may interact directly with the front-end or, perhaps more typically, is a program called from an intermediate program that mediates front-end and back-end activities.

# USER INTERFACE DESIGN (Front End):

For this project, we managed to show some of the features and functionalities of both front end and back end. First let us look at the overview of the front end and the features of the website. Some of the features are:

1. Cart
2. Login
3. Signup
4. View all category bar
5. Search
6. Drop down menu of all Categories
7. Add to cart



7

6

5

4

3

2

1

## Cart:

One of important parts of online trading is a shopping cart for websites, which allows customers to browse through your webstore, select and add products into their carts with a view to purchase. Thus purchasing goes in the convenient way. The main target is involving more and more customers, you can do it by giving comfortable shopping (without problems) in your virtual store. Actually, payment can proceed online or in another way, it depends on your preference. Such software allows shoppers to gradually collect products to buy in their carts next they could check the list of items and purchase. This solution calculates the total value of the order with shipping and packaging the goods, also associated taxes will be taken into an account. There is no need to be the I.T. man to be able to add, edit and do other things in your online shop, everything you can do with ease. As far as payment gateway, there is a huge number, which you can use. But if you haven’t chosen a payment system for virtual store, there are many supported systems in our solution, like Visa, MasterCard, eWallet, etc., that are available all over the world. What is more, your customers can choose their language for easy reference.

## Login:

This login is only for the user who have register to the website.

## Register:

This feature allows a user to create an account on to the website, where, the user enters all the detailed information such as email id, shipping address so the user doesn’t have to enter their information every time the visit the website.

## View all Category Bar:

This is a short cut way to search any category.

## Search Bar:

The user can search for any item by just typing the desired item the user is looking for.

## Drop Down menu of Categories:

A drop down menu is horizontal list of options that each contain a vertical menu. When you roll over or click one of the primary options in a drop down menu, a list of choices will "drop down" below the main menu.

## Add to Cart:

Add to Cart is a way to create a temporary list of items by adding them to your cart, which will keep track of the items until you leave our website. You can export items in your cart by saving the list to a file or sending it to an email address. You can also place the items on hold or add them to your wish list.

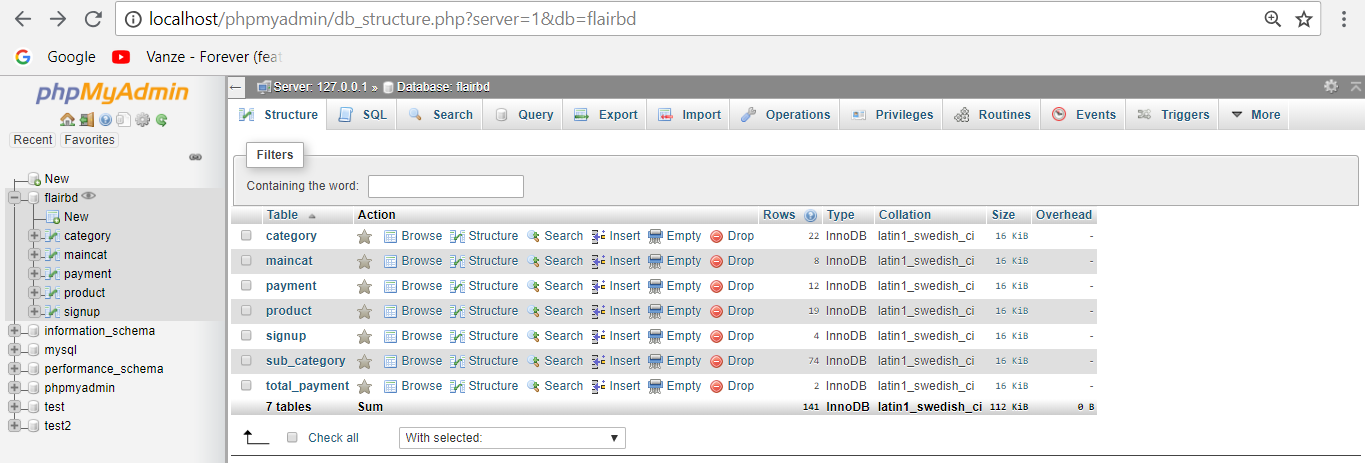
# Database (Back End)

A database is a collection of [information](https://searchsqlserver.techtarget.com/definition/information) that is organized so that it can be easily accessed, managed and updated.

Data is organized into rows, columns and tables, and it is indexed to make it easier to find relevant information. Data gets updated, expanded and deleted as new information is added. Databases process workloads to create and update themselves, querying the data they contain and running applications against it.

## DATABASE CONNECTIVITY:

For this project we have created a database from localhost/phpmyadmin named “flairbd”.



The database contains different tables and each contains different information. Each table maybe connected on one another. When we click on the maincat, it will show the list of main categories we have selected for the website.

## Main-Category/ Sub-Category /Product list:

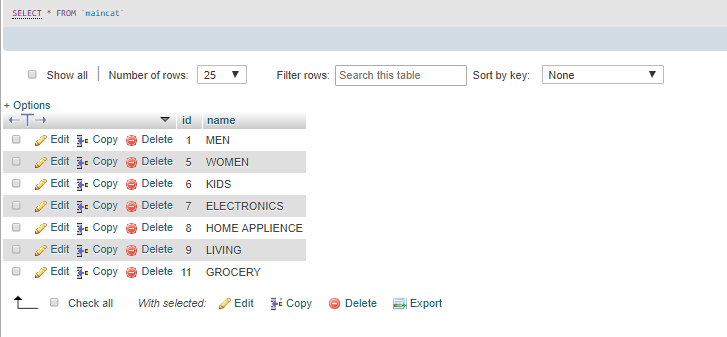
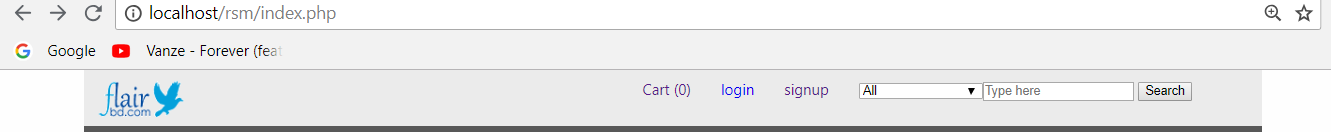


Figure: Main-Category on Database

This image shows the list of main categories we have selected for the website. The image below shows how it looks on the website.



Each main categories have their own sub-category. In the sub-category section it will show all kinds of products of each main category. Like:

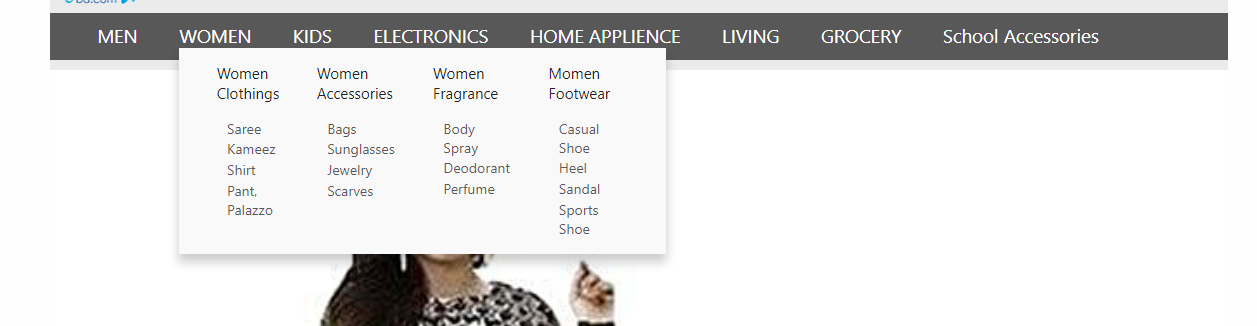


Figure: Sub-Category on Website

And in database the sub-categories are show as below:

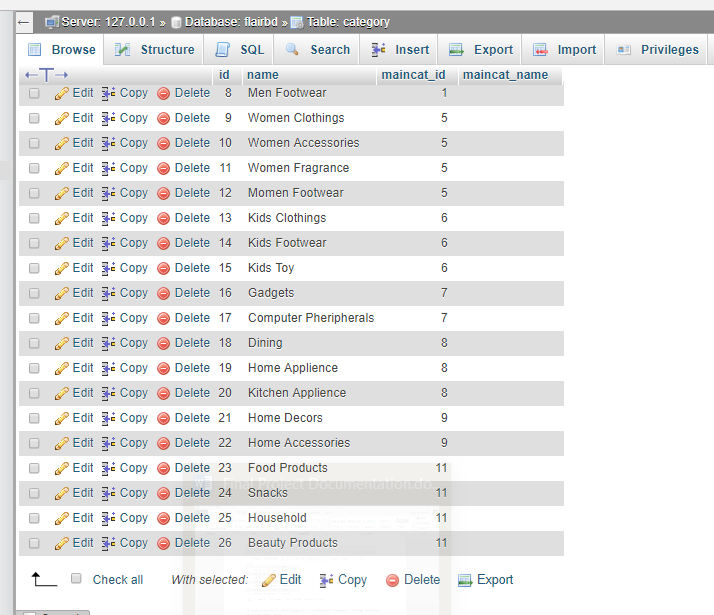


Figure: Sub-Category on Database

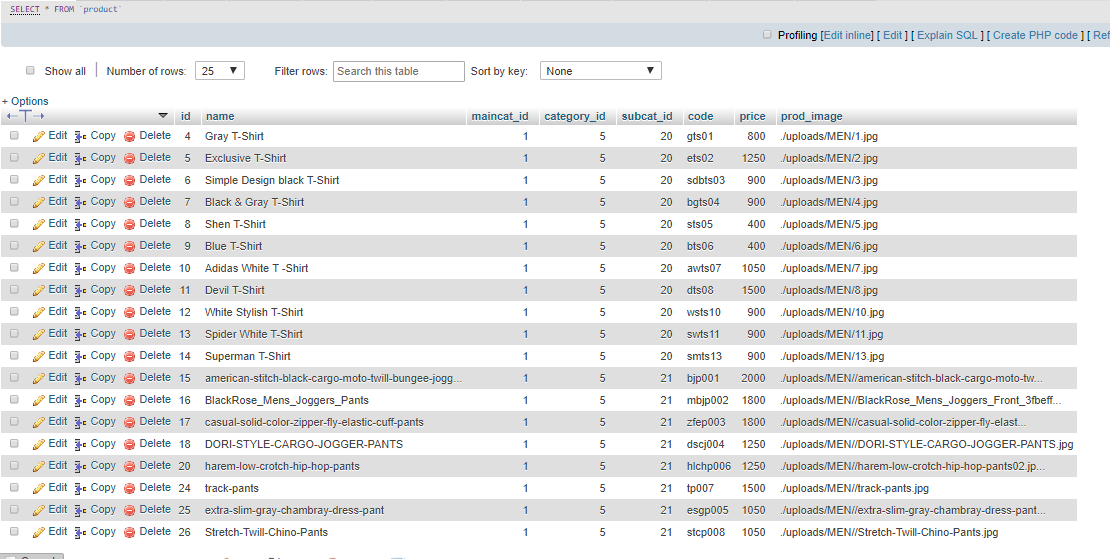
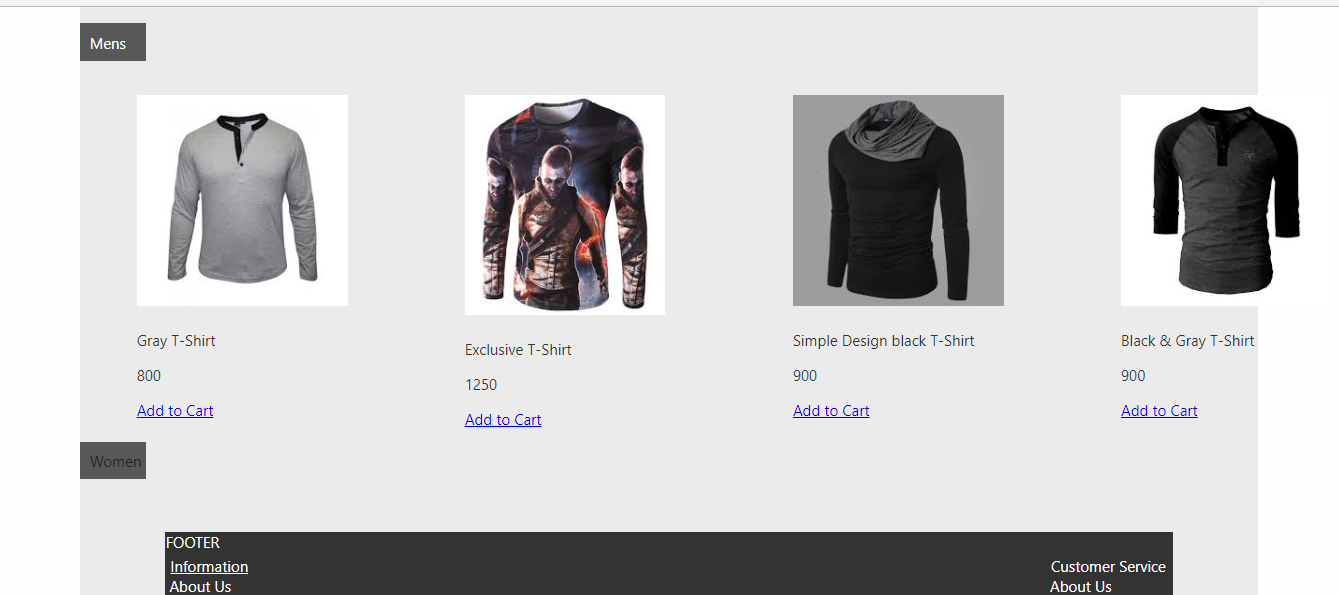
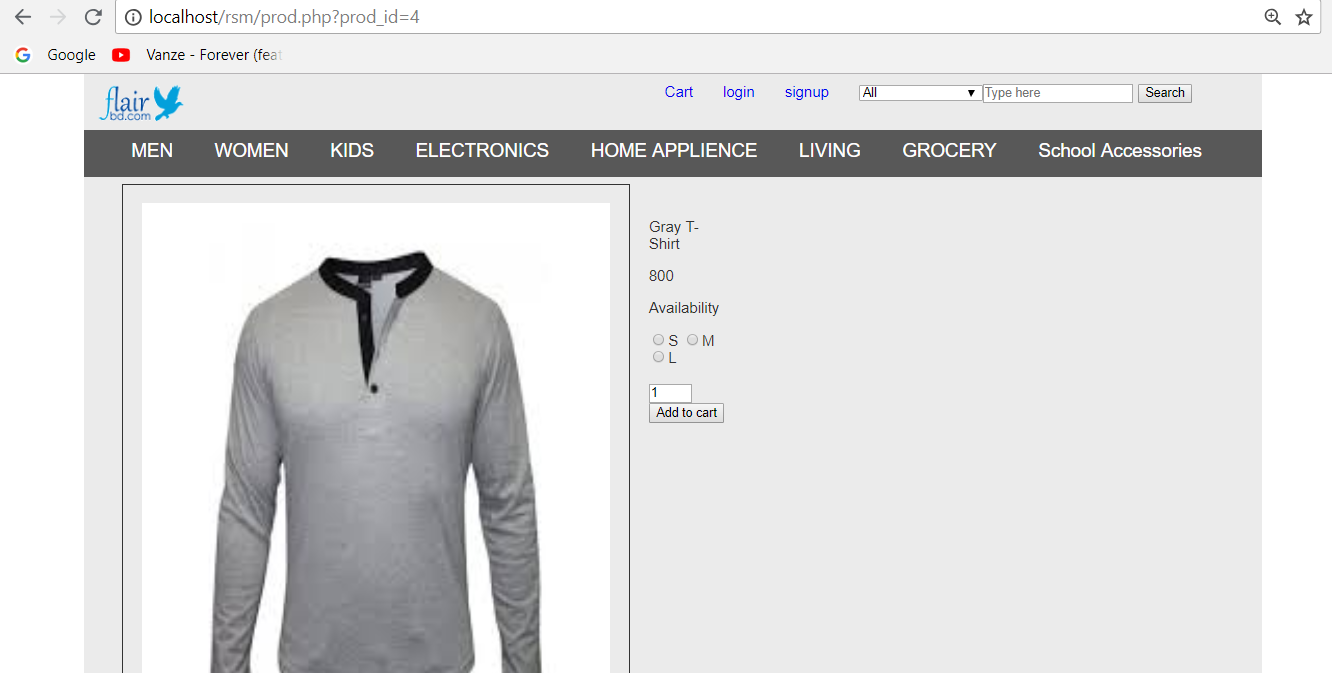
This table shows every product under each sub-category. 

Figure: Products of sub-category on database

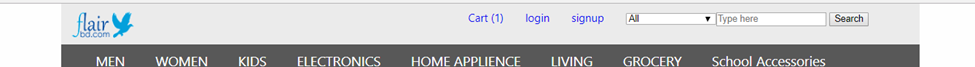
## Add to cart:

In this add to cart section, any user registered or guest user can add product to cart. 

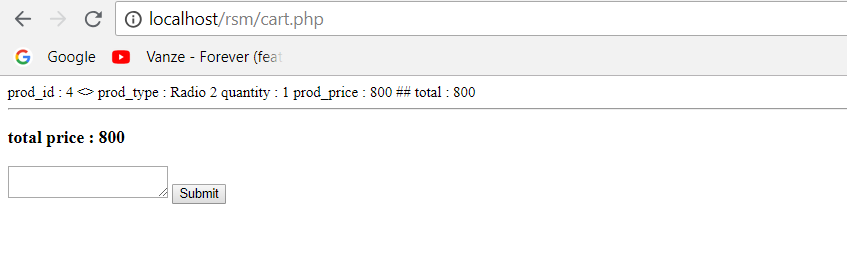
As the user clicks on a specific product it will direct the selected product page. The user can select his/her own specific choice, for example size, then the user has to select add to cart if he/she is ready to buy.



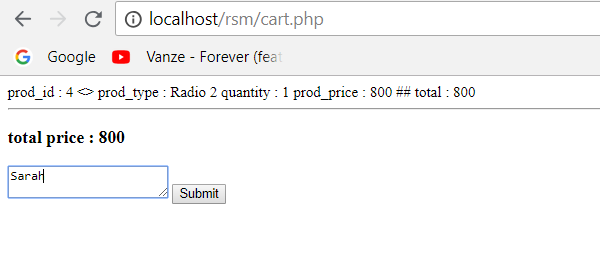
As the user clicks on Add to Cart, the product is added to cart.



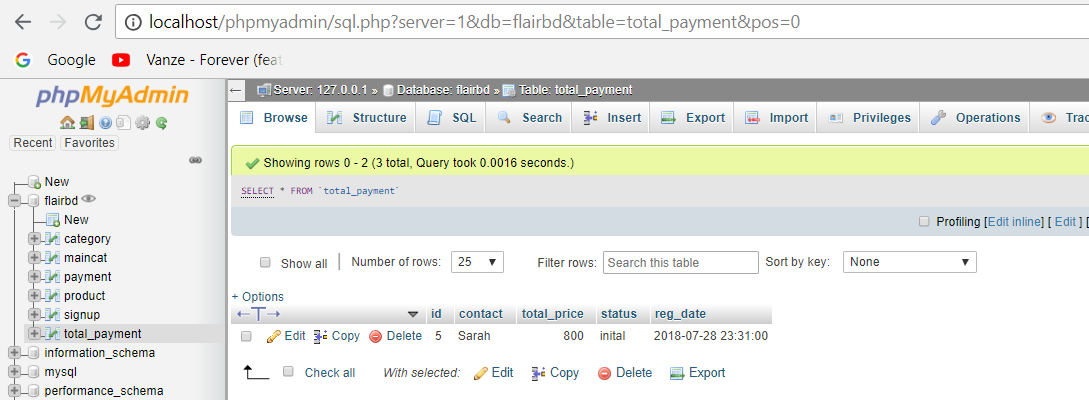
When the user clicks on the cart option, then the user is asked for some details like: Name, address etc.



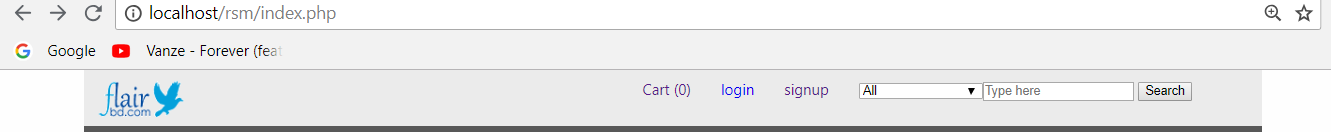
When the user type the details and clicks submit, the users request is sent.



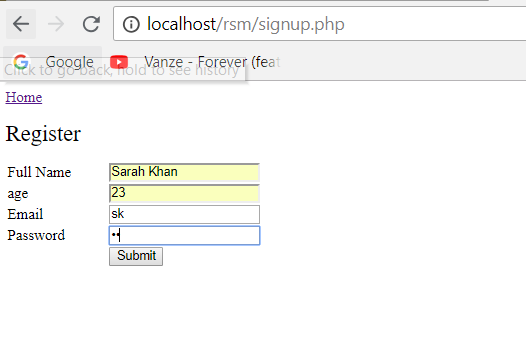
As the request is sent the admin can see all the request of product of different users.



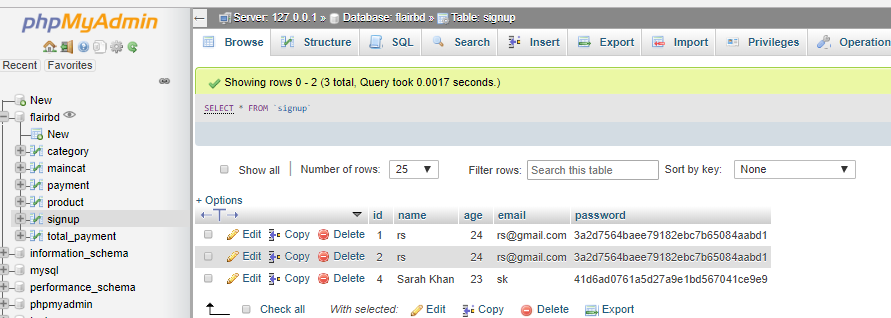
## Sign up as User:



When a user wants to sign up in the website, the user has to fill in some details, such as below:



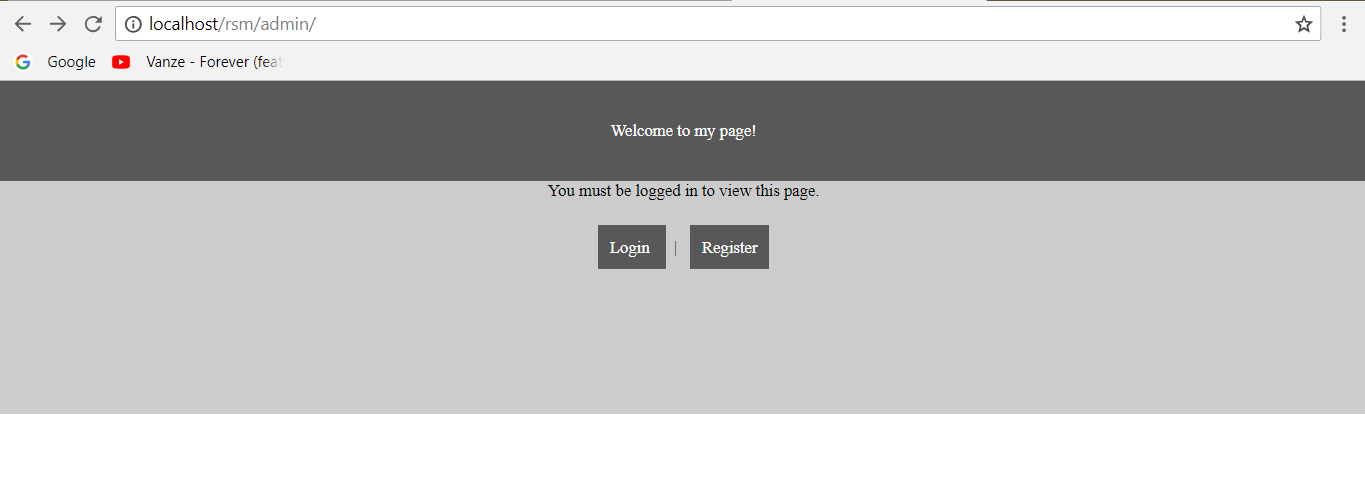
When the user successfully signs up, his information are updated into the database.



# Admin Panel:

The admin panel is where new post, categories, tags, pages, links and custom post types are created. It is also where theme files are changed, widgets are added, plugins are activated or updated, and reading, writing, general settings are changed.

## Sign up as Admin:

When a user is to have an access to edit, add, delete any changes from the database, they must b registered. 

When the user clicks register, he is given a form to enter information to gain access to admin panal.

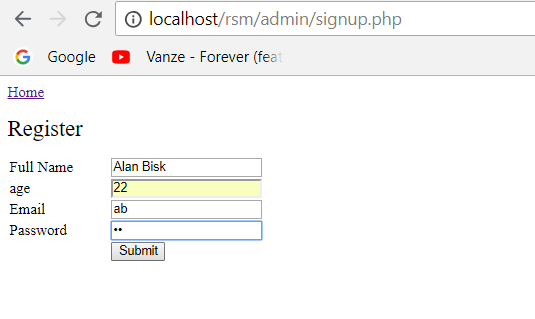


Figure: Signup as Admin User

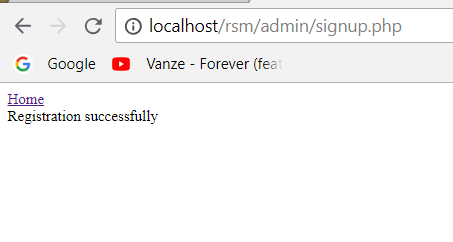
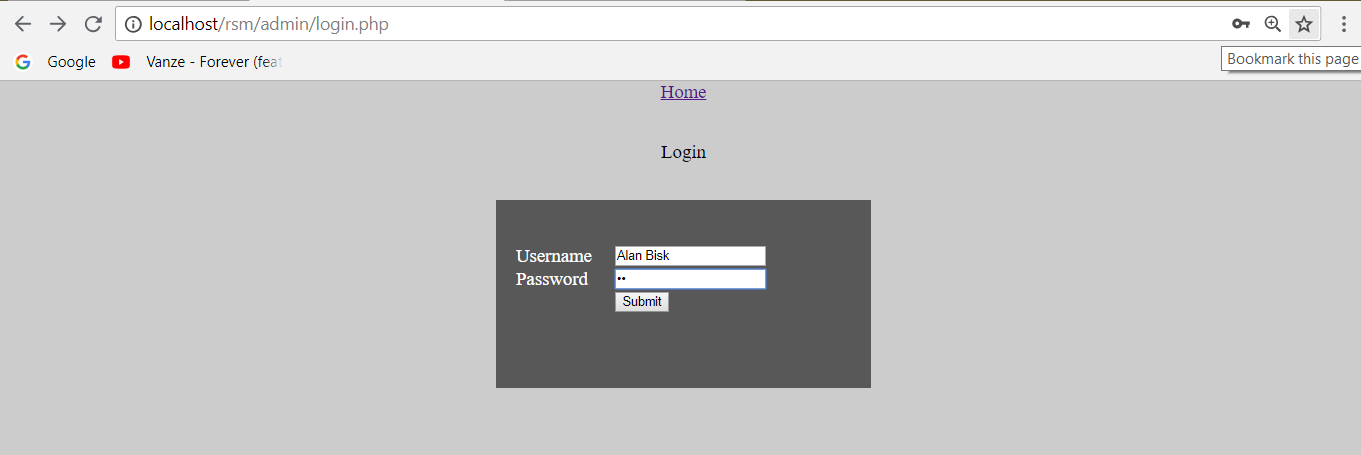
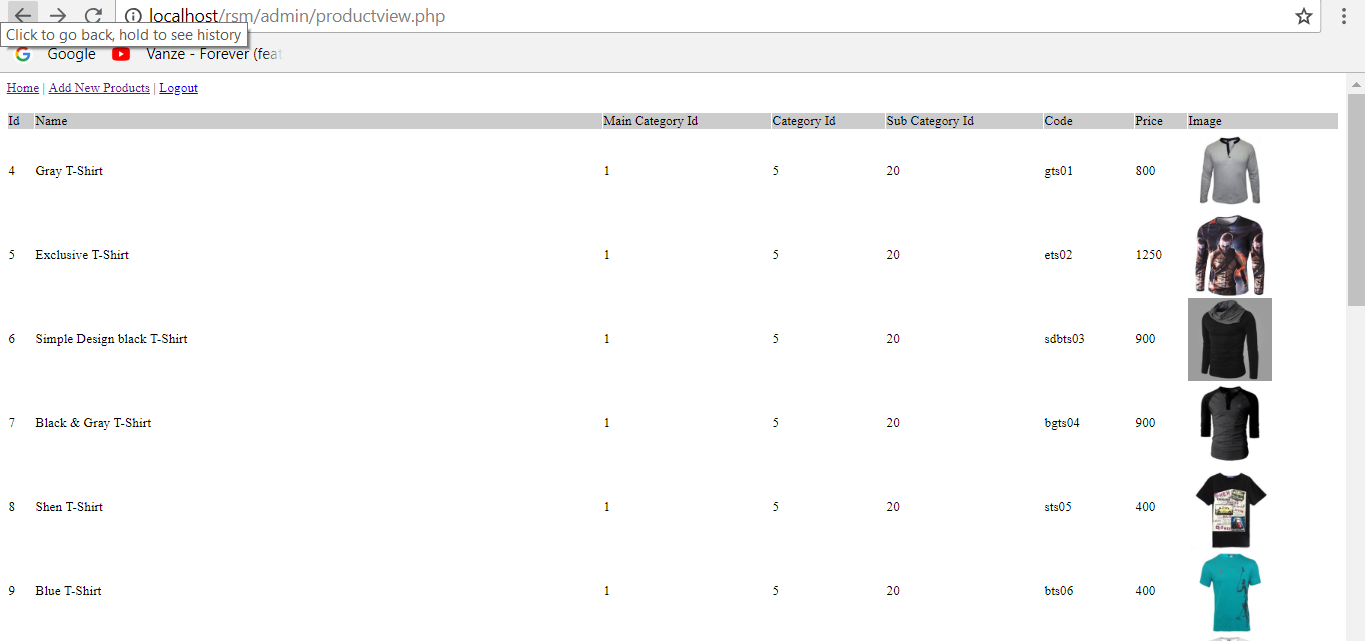


Figure: Message after Successful registration

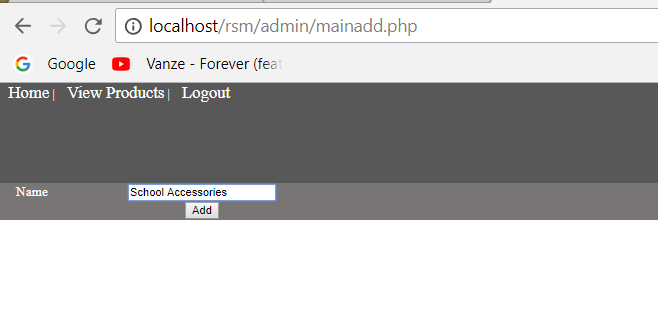


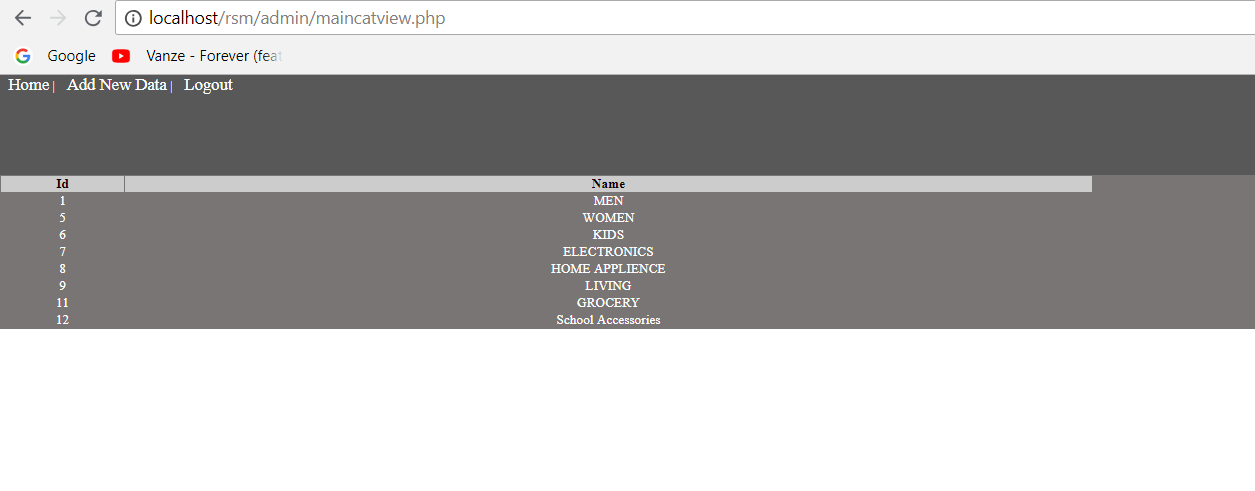
## Add/ Remove/ Change information from admin panel:

The admin user has the control to add, remove or change any kind of changes of the website from the admin panel. For example the admin user can change the price or remove the product from the list.



The admin user can also add main-categories, sub-categories to the web from the admin panel. Such as:





# Limitations and Future Development

There are some limitations for the current system to which solutions can be provided as a future development:

1. The system is not configured for multi-users at this time. The concept of transaction can be used to achieve this.

2. The Website is not accessible to everyone. It can be deployed on a web server so that everybody who is connected to the Internet can use it.

3. Credit Card validation is not done. Third party proprietary software can be used for validation check. As for other future developments, the following can be done:

* The Administrator of the web site can be given more functionalities, like looking at a specific customer’s profile, the books that have to be reordered, etc.
* Multiple Shopping carts can be allowed.

# Conclusion

The Internet has become a major resource in modern business, thus electronic shopping has gained significance not only from the entrepreneur’s but also from the customer’s point of view. For the entrepreneur, electronic shopping generates new business opportunities and for the customer, it makes comparative shopping possible. As per a survey, most consumers of online stores are impulsive and usually make a decision to stay on a site within the first few seconds. “Website design is like a shop interior. If the shop looks poor or like hundreds of other shops the customer is most likely to skip to the other site”. Hence we have designed the project to provide the user with easy navigation, retrieval of data and necessary feedback as much as possible.

A good shopping cart design must be accompanied with user-friendly shopping cart application logic. It should be convenient for the customer to view the contents of their cart and to be able to remove or add items to their cart. The shopping cart application described in this project provides a number of features that are designed to make the customer more comfortable.

# Appendix