



## **Web Based Online Shopping System**

A Report for the Evaluation 3 of Project 2

*Submitted By*

**Deeksha Kumari**

**(1613101876 / 17SCSE101933)**

*in partial fulfillment for the award of the degree*

*of*

**Bachelor in Technology**

**IN**

**Computer Science and Engineering**

**School of Computing Science and Engineering**

**Under the Supervision of**

**Dr. S. Srinivasan**

**Professor (SCSE)**

**APRIL/MAY-2020**



**School of Computing Science and Engineering**

**BONAFIDE CERTIFICATE**

Certified that this project report “ WEB BASED ONLINE SHOPPING SYSTEM ” is the bonafide work of “ DEEKSHA KUMARI (1613101876) ” who carried out the project work under my supervision.

**Signature of Head**

Dr. Munish Sabharwal

**Professor & Dean**

**School of Computing Science &**

**Engineering**

**Signature of Supervisor**

Dr. S. Srinivasan

**Professor**

**School of Computing Science &**

**Engineering**

# TABLE OF CONTENTS

# PAGE NO.

1.	Abstract.....	1
2.	Introduction.....	3
2.1	Project Objective.....	5
2.2	Project Overview.....	5
2.3	Project Scope.....	6
2.4	Study of the Systems.....	6
3.	Existing System.....	16
4.	Proposed System.....	17
5.	Implementation.....	18
5.1	System Design.....	18
5.2	Input and Output Design.....	19
5.3	Database Design.....	20
5.4	System Tools.....	21
5.5.1	Front End.....	21
5.5.2	Back End.....	22
5.5	Database Tables.....	23
5.6	E R Diagram.....	27
5.7	Dataflow Diagram.....	30
6.	Screenshot.....	34
7.	Conclusion.....	44
	References	

- **Abstract:**

This project is a web based online shopping system. The project objective is to deliver the online shopping application into android platform. This project is an attempt to provide the advantages of online shopping to customers. It helps buying the products in the shop anywhere through internet by using an android device.

Since the application is available in the Smartphone it is easily accessible and always available. The Server process the customers and the items are shipped to the address submitted by them. The application was designed into one module that is for the customers who wish to buy the articles. The end user of this product is a departmental store where the application is hosted on the web and the administrator maintains the database.

The application which is deployed at the customer database, the details of the items are brought forward from the database for the customer view based on the selection through the menu and the database of all the products are updated at the end of each transaction. Data entry into the application can be done through various screens designed for various levels of users. Once the authorized personnel feed the relevant data into the system, several reports could be generated as per the security.

An online shopping system that permits a customer to submit online orders for items and/or services from a store that serves both walk-in customers and online customers. The 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, many shopping systems provide a virtual shopping cart for holding items selected for purchase.

- **Introduction:**

This project is a web based shopping system for an existing shop. The project objective is to deliver the online shopping application into android platform.

Online shopping is the process whereby consumers directly online, without an intermediary service, over the Internet. It is a form of electronic commerce. This project is an attempt to provide the advantages of online shopping to customers. It helps buying the products anywhere through internet by using an android device. An online shopping system that permits a customer to submit online orders for items or services from a store that serves both walk-in customers and online customers. The 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, many shopping systems provide a virtual shopping cart for holding items selected for purchase.

This project is a web based online shopping system. The project objective is to deliver the online shopping application into android platform. This project is an attempt to provide the advantages of online shopping to customers. It helps buying the products in the shop anywhere through internet by using an android device.

Since the application is available in the Smartphone it is easily accessible and always available.

The Server process the customers and the items are shipped to the address submitted by them.

The application was designed into one module that is for the customers who wish to buy the articles. The end user of this product is a departmental store where the application is hosted on the web and the administrator maintains the database.

- **Project objective:**

The objective of the project is to make an application in android platform to purchase items. In order to build such an application complete web support need to be provided. A complete and efficient web application which can provide the online shopping experience is the basic objective of the project. The web application can be implemented in the form of an android application with web view.

- **Project overview:**

The central concept of the application is to allow the customer to shop virtually using the Internet and allow customers to buy the items and articles. The information pertaining to the products are stores on an RDBMS at the server side (store).

The Server process the customers and the items are shipped to the address submitted by them. The application was designed into a module that is for the customers who wish to buy the articles. The end user of this product is a departmental store where the application is hosted on the web and the administrator maintains the database. The application which is deployed at the customer database, the details of the items are brought forward from the database for the customer view based on the selection through the menu and the database of all the products are updated at the end of each transaction. Data entry into the application can be done through various screens designed for various levels of users. Once the authorized personnel feed the relevant data into the system, several reports could be generated as per the security.



- **PROJECT SCOPE:**

The system recommends a facility to accept the orders 24\*7 and a home delivery system which can make customers happy. If shops are providing an online portal where their customers can enjoy easy shopping from anywhere, the shops won't be losing any more customers to the trending online shops such as flipcart or eBay. Since the application is available in the Smartphone it is easily accessible and always available.

- **STUDY OF THE SYSTEM**

- Modules:

The system after careful analysis has been identified to be presented with the following modules and roles. The modules involved are:

- Administrator
- Moderators
- Users

– Administrator:

The administrator is the super user of this application. Only admin have access into this admin page. Admin may be the owner of the shop. The administrator has all the information about all the users and about all products.

This module is divided into different sub-modules.

1. Manage Moderators
2. Manage Products
3. Manage Users
4. Manage Orders

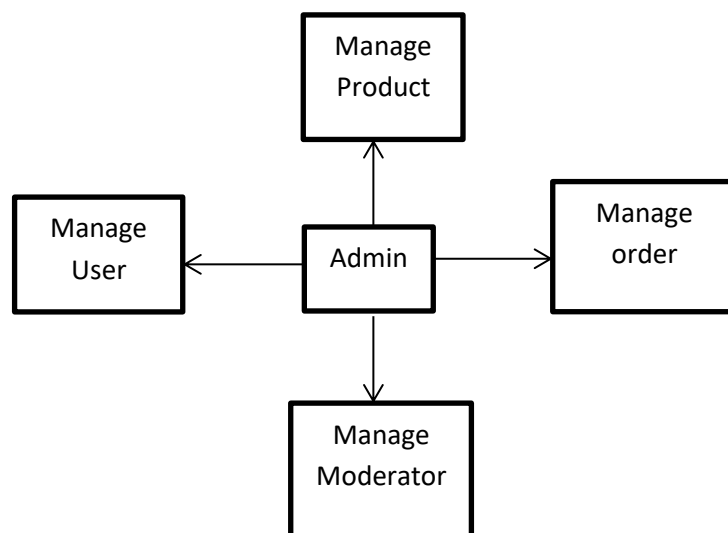


Fig 1.1: Admin module

– Manage moderator:

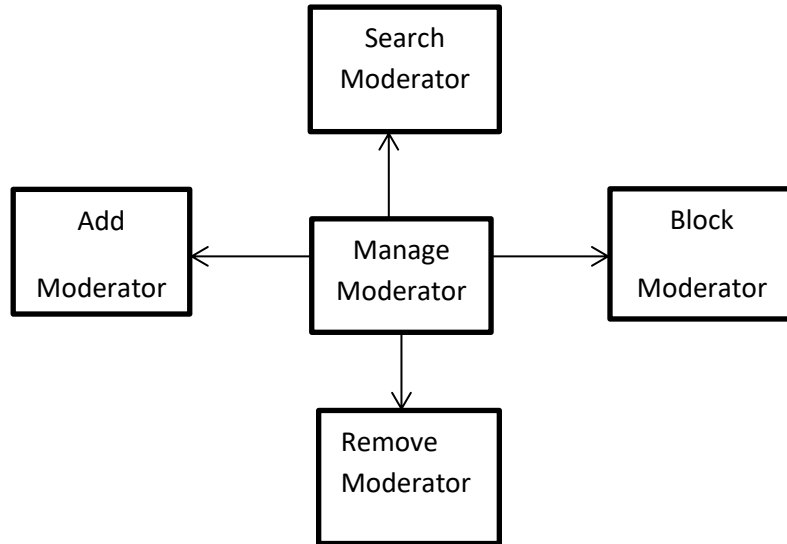


Fig 1.2: Manage Moderator

- Add Moderator :

Only admin is having the privilege to add a moderator. A moderator can be considered as staffs who manage the orders or owner of a group of products.

- Block moderator :

Admin can restrict a moderator from managing the orders by blocking them. Admin can unblock a blocked user if needed.

- Remove Moderator :

Admin has privilege to delete a moderator who was added.

- Search moderator:

All existing moderators can be viewed by the administrator as a list. If there is number of moderators and admin need to find one of them, the admin can search for a moderator by name.

– Manage products :

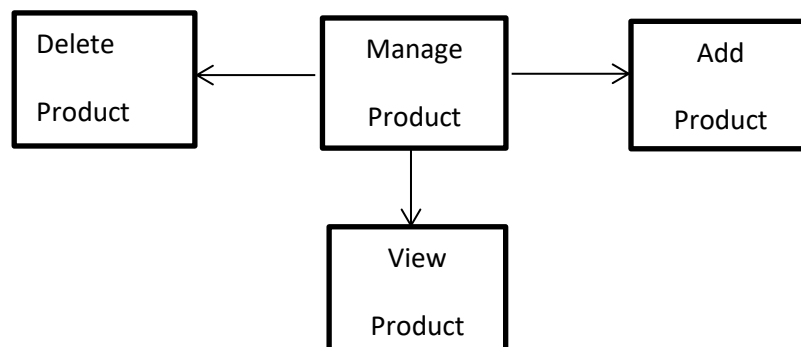


Fig 1.3: Manage Products

- Add Products

The shopping cart project contains different kind of products. The products can be classified into different categories by name. Admin can add new products into the existing system with all its details including an image.

- Delete Products

Administrator can delete the products based on the stock of that particular product.

- Search products

Admin will have a list view of all the existing products. He can also search for a particular product by name.

– MANAGE USER :

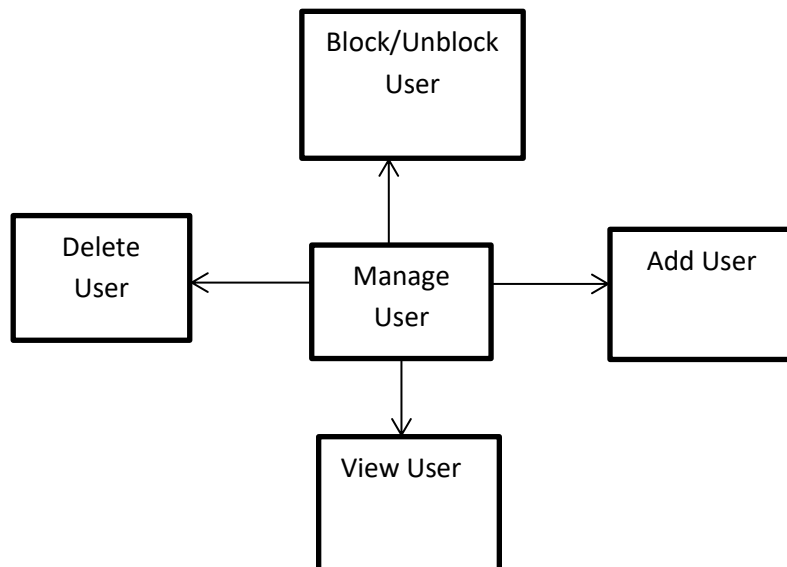


Fig 1.4: Manage User

- View Users

The admin will have a list view of all the users registered in the system. Admin can view all the details of each user in the list except password.

- Add Users

Admin has privileges to add a user directly by providing the details.

- Delete & Block/Unblock Users

Administrator has a right to delete or block or unblock a user. The default status of a new user registered is set as blocked. The admin must accept the new user by unblocking him.

– MANAGE ORDERS :

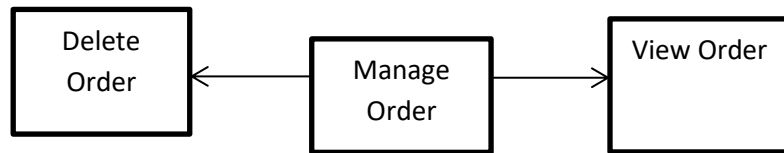


Fig 1.5: Manage Orders

- View Order :

Administrator can view the Orders which are generated by the users. He can verify the details of the purchase.

- Delete order :

Admin can delete order from the orders list when the product is taken for delivery.

– MODERATORS:

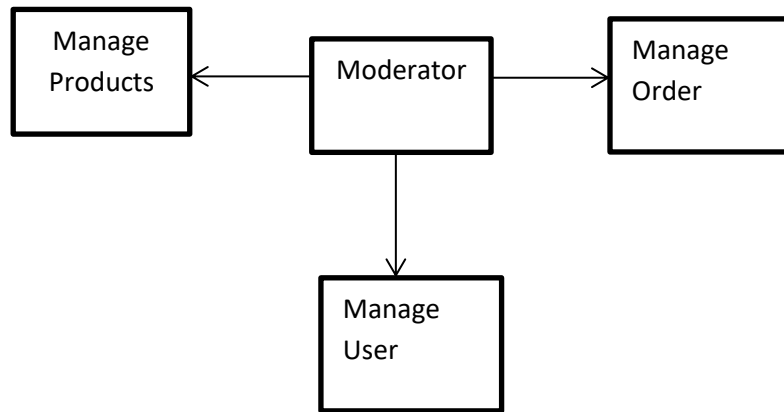


Fig 1.6: Moderator Module

A moderator is considered as a staff who can manage orders for the time being. As a future update moderator may give facility to add and manage his own products. Moderators can reduce the work load of admin. Now moderator has all the privilege an admin having except managing other moderators. He can add products and users. He can also check the orders and edit his profile.

- Manage products
- Manage users
- Manage orders



– USERS:

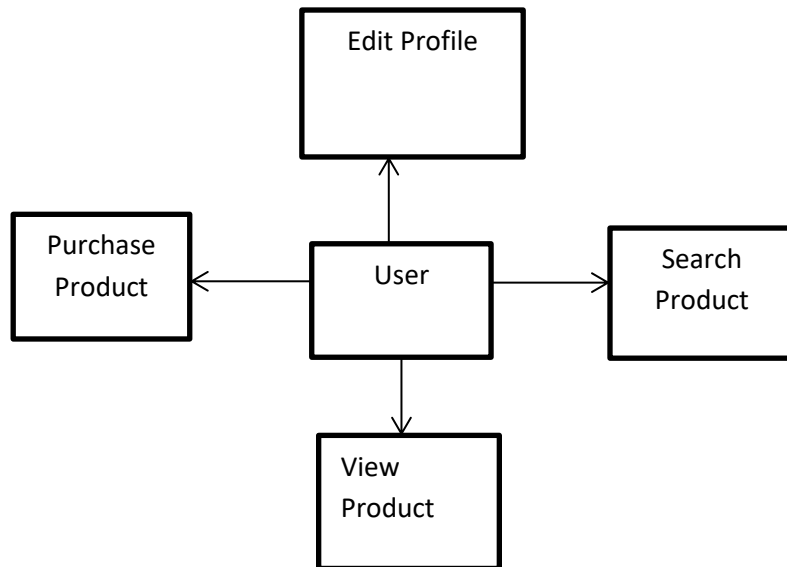


Fig 1.7: User Module

– Registration :

A new user will have to register in the system by providing essential details in order to view the products in the system. The admin must accept a new user by unblocking him.

– Login

A user must login with his user name and password to the system after registration.

- View Products

User can view the list of products based on their names after successful login. A detailed description of a particular product with product name, products details, product image and price can be viewed by users.

- Search Product

Users can search for a particular product in the list by name.

- Add to cart:

The user can add the desired product into his cart by clicking add to cart option on the product.

He can view his cart by clicking on the cart button. All products added by cart can be viewed in the cart. User can remove an item from the cart by clicking remove.

- Submit Cart:

After confirming the items in the cart the user can submit the cart by providing a delivery address. On successful submitting the cart will become empty.

- History

In the history the user will have a view of pending orders.

- Edit Profile

The user can view and edit the profile.

## – **EXISTING SYSTEM**

The current system for shopping is to visit the shop manually and from the available product choose the item customer want and buying the item by payment of the price of the item.

1. It is less user-friendly.
2. User must go to shop and select products.
3. It is difficult to identify the required product.
4. Description of the product limited.
5. It is a time consuming process
6. Not in reach of distant users.

## – **PROPOSED SYSTEM**

In the proposed system customer need not go to the shop for buying the products. He can order the product he wishes to buy through the application in his Smartphone.

The shop owner will be admin of the system. Shop owner can appoint moderators who will help owner in managing the customers and product orders. The system also recommends a home delivery system for the purchased products.

- **IMPLEMENTATION**
  - **SYSTEM DESIGN**

System design is the solution for the creation of a new system. This phase focuses on the detailed implementation of the feasible system. It emphasizes translating design specifications to performance specifications. System design has two phases of development.

- Logical design
- Physical design

During the logical design phase, the analyst describes inputs (sources), outputs (destinations), databases (data stores) and procedures (data flows) all in a format that meets the user requirements. The analyst also specifies the needs of the user at a level that virtually determines the information flow in and out of the system and the data resources. Here, the logical design is done through data flow diagrams and database design. The physical design is followed by physical design or coding. Physical design produces the working system by defining the design specifications, which specify exactly what the candidate system must do. The programmers write the necessary programs that accept input from the user, perform necessary processing on accepted data and produce the required report on a hard copy or display it on the screen.

- **INPUT AND OUTPUT DESIGN**
- **INPUT DESIGN:**

Input design is the link that ties the information system into the world of its users. The input design involves determining the inputs, validating the data, minimizing the data entry and provides a multi-user facility. Inaccurate inputs are the most common cause of errors in data processing. Errors entered by the data entry operators can be controlled by input design. The user-originated inputs are converted to a computer based format in the input design. Input data are collected and organized into groups of similar data. Once identified, the appropriate input media are selected for processing. All the input data are validated and if any data violates any conditions, the user is warned by a message. If the data satisfies all the conditions, it is transferred to the appropriate tables in the database. In this project the student details are to be entered at the time of registration. A page is designed for this purpose which is user friendly and easy to use. The design is done such that users get appropriate messages when exceptions occur.

– **OUTPUT DESIGN:**

Computer output is the most important and direct source of information to the user. Output design is a very important phase since the output needs to be in an efficient manner. Efficient and intelligible output design improves the system relationship with the user and helps in decision making. Allowing the user to view the sample screen is important because the user is the ultimate judge of the quality of output. The output module of this system is the selected notifications.

• **DATABASE**

– **DATABASE DESIGN:**

Databases are the storehouses of data used in the software systems. The data is stored in tables inside the database. Several tables are created for the manipulation of the data for the system.

Two essential settings for a database are :

Primary Key- The field that is unique for all the record occurrences.

Foreign Key- The field used to set relation between tables.

Normalization is a technique to avoid redundancy in the tables.

- **SYSTEM TOOLS**

The various system tools that have been used in developing both the front end and the back end of the project.

- **FRONT END:**

JSP, HTML, CSS, JAVA SCRIPT, ANDROID are utilized to implement the frontend.

- **Java Server Pages (JSP)**

Different pages in the applications are designed using jsp. A Java Server Pages component is a type of Java servlet that is designed to fulfill the role of a user interface for a Java web application. Web developers write JSPs as text files that combine HTML or XHTML code, XML elements, and embedded JSP actions and commands. Using JSP, one can collect input from users through web page.

- **HTML (Hyper Text Markup Language)**

HTML is a syntax used to format a text document on the web.

- **CSS (Cascading Style Sheets)**

CSS is a style sheet language used for describing the look and formatting of a document written in a markup language.



- **Java Script**

JS is a dynamic computer programming language. It is most commonly used as part of web browsers, whose implementations allow client-side scripts to interact with the user, control the browser, communicate asynchronously, and alter the document content that is displayed.

Java Script is used to create pop-up windows displaying different alerts in the system like “User registered successfully”, “Product added to cart” etc.

- **Android**

The application is delivered to customer through an android application. So android platform is used to develop the user application.

- **BACK END**

The back end is implemented using MySQL which is used to design the databases.

- **MySQL**

MySQL is the world's second most widely used open-source relational database management system (RDBMS). The SQL phrase stands for Structured Query Language. An application software called Navicat was used to design the tables in MySQL.

- **Database Tables:**

- **Account:**

DB Browser for SQLite - C:\Users\dell\Desktop\onlineshopping\db.sqlite3

File Edit View Tools Help

New Database Open Database Write Changes Revert Changes Open Project Save Project Attach Database Close Database

Database Structure Browse Data Edit Pragma Execute SQL

Table: accounts\_profile

id	phone	address	user_id
1	9872134511	B-23, Sec-12, ...	2
2	9822134511	E-43, Alpha-1, ...	3
3	9855431277	A-23, Beta-1, ...	4
4	9822114455	A-23, Sec-12, ...	5
5	9821342122	A-10, Sec-10, N...	6
6	9811231145	Greater Noida	7
7	7875856758	Alfa 1	8

Go to: 1

- **User:**

DB Browser for SQLite - C:\Users\dell\Desktop\onlineshopping\db.sqlite3

File Edit View Tools Help

New Database Open Database Write Changes Revert Changes Open Project Save Project Attach Database Close Database

Database Structure Browse Data Edit Pragma Execute SQL

Table: auth\_user

id	password	last_login	is_superuser	username	first_name	email	is_staff	is_active	date_joined	last_name
1	pbkdf2_sha25...	2020-04-30 0...	1	admin			1	1	2019-05-29 0...	
2	pbkdf2_sha25...	2019-06-05 1...	0	manoj.pande...	Manoj	manoj.pande...	0	1	2019-05-29 0...	Pandey
3	pbkdf2_sha25...	2019-05-31 0...	0	pankaj.mishra...	Pankaj	pankaj.mishra...	0	1	2019-05-29 0...	Mishra
4	pbkdf2_sha25...	2019-06-02 0...	0	test.user@gm...	Test	test.user@gm...	0	1	2019-06-02 0...	User
5	pbkdf2_sha25...	2019-06-03 0...	0	mahima.gupt...	Mahima	mahima.gupt...	0	1	2019-06-03 0...	Gupta
6	pbkdf2_sha25...	2019-07-14 0...	0	test@gmail.com	test	test@gmail.com	0	1	2019-07-12 0...	user
7	pbkdf2_sha25...	2020-02-13 0...	0	vijay.mehra@...	Vijay	vijay.mehra@...	0	1	2020-02-12 1...	Mehra
8	pbkdf2_sha25...	NULL	0	deekshakuma...	Deeksha	deekshakuma...	0	1	2020-04-30 0...	Kumari

Go to: 1

## Admin log:

DB Browser for SQLite - C:\Users\delf\Desktop\onlineshopping\db.sqlite3

File Edit View Tools Help

New Database Open Database Write Changes Revert Changes Open Project Save Project Attach Database Close Database

Database Structure Browse Data Edit Pragmas Execute SQL

Table: django\_admin\_log

	id	action_time	object_id	object_repr	change_message	content_type_id	user_id	action_flag
	Filter	Filter	Filter	Filter	Filter	Filter	Filter	Filter
1	1	2019-05-29 0...	1	Angel Fashion...	["added": {}]	7	1	1
2	2	2019-05-29 0...	2	Brand Clothin...	["added": {}]	7	1	1
3	3	2019-05-29 0...	3	Fashion Shirt ...	["added": {}]	7	1	1
4	4	2019-05-29 0...	4	Fashion Trend...	["added": {}]	7	1	1
5	5	2019-05-29 0...	5	Hot Men Desi...	["added": {}]	7	1	1
6	6	2019-05-29 0...	6	HP Notebook ...	["added": {}]	7	1	1
7	7	2019-05-29 0...	7	Latest Fashio...	["added": {}]	7	1	1
8	8	2019-05-29 0...	8	Lenovo V310 l...	["added": {}]	7	1	1
9	9	2019-05-29 0...	9	Men Black Jea...	["added": {}]	7	1	1
10	10	2019-05-29 0...	10	Men Formal S...	["added": {}]	7	1	1
11	11	2019-05-29 0...	11	Samsung 32 I...	["added": {}]	7	1	1
12	12	2019-05-29 0...	12	Sony 32 Inche...	["added": {}]	7	1	1
13	13	2019-05-29 0...	13	VU 32 Inches ...	["added": {}]	7	1	1
14	14	2019-05-29 0...	14	Zip Black Text...	["added": {}]	7	1	1

References auth\_user(id)  
Hold Ctrl+Shift and click to jump there

1 - 14 of 14

Go to: 1

UTF-8

## Content:

DB Browser for SQLite - C:\Users\dell\Desktop\onlineshopping\db.sqlite3

File Edit View Tools Help

New Database Open Database Write Changes Revert Changes Open Project Save Project Attach Database Close Database

Database Structure Browse Data Edit Pragma Execute SQL

Table: django\_content\_type

	id	app_label	model
	Filter	Filter	Filter
1	1	admin	logentry
2	2	auth	permission
3	3	auth	group
4	4	auth	user
5	5	contenttypes	contenttype
6	6	sessions	session
7	7	products	product
8	8	products	cart
9	9	accounts	profile
10	10	products	order
11	11	products	orderitem
12	12	products	payment

1 - 12 of 12

Go to: 1

UTF-8

## Product cart:

DB Browser for SQLite - C:\Users\dell\Desktop\onlineshopping\db.sqlite3

File Edit View Tools Help

New Database Open Database Write Changes Revert Changes Open Project Save Project Attach Database Close Database

Database Structure Browse Data Edit Pragma Execute SQL

Table: products\_cart

	id	quantity	total	user_id	item_name	price	image
	Filter	Filter	Filter	Filter	Filter	Filter	Filter
1	34	1	425.0	2	Brand Clothin...	425.0	/media/produ...
2	35	1	540.0	2	Fashion Shirt ...	540.0	/media/produ...
3	36	1	450.0	6	Angel Fashion...	450.0	/media/produ...
4	39	1	425.0	6	Brand Clothin...	425.0	/media/produ...
5	47	1	450.0	1	Angel Fashion...	450.0	/media/produ...
6	48	1	29500.0	1	Lenovo V310 I...	29500.0	/media/produ...
7	49	1	575.0	1	Fashion Trend...	575.0	/media/produ...

1 - 7 of 7

Go to: 1

UTF-8

## – Product Order:

DB Browser for SQLite - C:\Users\dell\Desktop\onlineshopping\db.sqlite3

File Edit View Tools Help

New Database Open Database Write Changes Revert Changes Open Project Save Project Attach Database Close Database

Database Structure Browse Data Edit Pragmas Execute SQL

Table: products\_order

	id	grand_total	date	user_id	address
Filter	Filter	Filter	Filter	Filter	Filter
1	2	1240.0	2019-05-30 1...	2	B-23, Sec-12, ...
2	3	1090.0	2019-05-30 1...	2	B-23, Sec-12, ...
3	4	965.0	2019-05-30 1...	3	E-43, Alpha-1, ...
4	5	1140.0	2019-05-31 0...	2	B-23, Sec-12, ...
5	6	1000.0	2019-05-31 0...	3	E-43, Alpha-1, ...
6	7	1250.0	2019-06-01 0...	2	B-23, Sec-12, ...
7	8	965.0	2019-06-02 0...	4	A-23, Beta-1, ...
8	9	575.0	2019-06-02 0...	4	A-23, Beta-1, ...
9	10	1090.0	2019-06-03 0...	5	A-23, Sec-12, ...
10	11	2805.0	2020-02-13 0...	7	Greater Noida
11	12	2980.0	2020-02-13 0...	7	Greater Noida
12	13	2290.0	2020-02-13 0...	7	Greater Noida
13	14	2515.0	2020-02-13 0...	7	Greater Noida

1 - 13 of 13

Go to: 1

UTF-8

## – Product Payment:

DB Browser for SQLite - C:\Users\dell\Desktop\onlineshopping\db.sqlite3

File Edit View Tools Help

New Database Open Database Write Changes Revert Changes Open Project Save Project Attach Database Close Database

Database Structure Browse Data Edit Pragmas Execute SQL

Table: products\_payment

	id	date	mode	order_id	amount
Filter	Filter	Filter	Filter	Filter	Filter
1	1	2020-02-13 0...	Cash On Deliv...	11	2805.0
2	2	2020-02-13 0...	Cash On Deliv...	12	2980.0
3	3	2020-02-13 0...	Cash On Deliv...	13	2290.0
4	4	2020-02-13 0...	Cash On Deliv...	14	2515.0

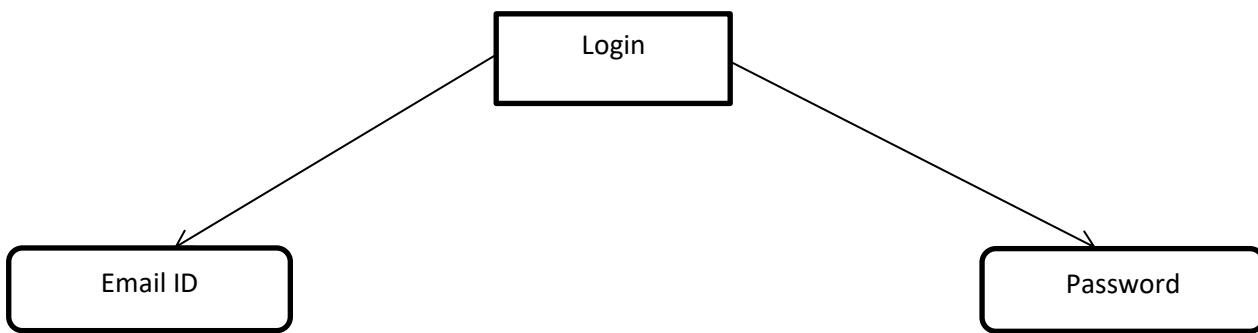
1 - 4 of 4

Go to: 1

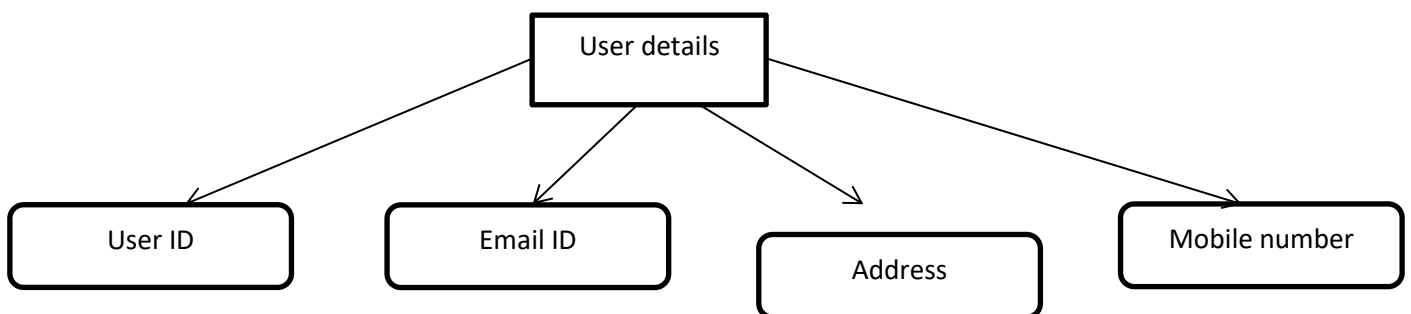
UTF-8

- **E R Diagram:** An ER diagram shows the relationship among entity sets. An entity set is a group of similar entities and these entities can have attributes. In terms of DBMS, an entity is a table or attribute of a table in database, so by showing relationship among tables and their attributes, ER diagram shows the complete logical structure of a database.

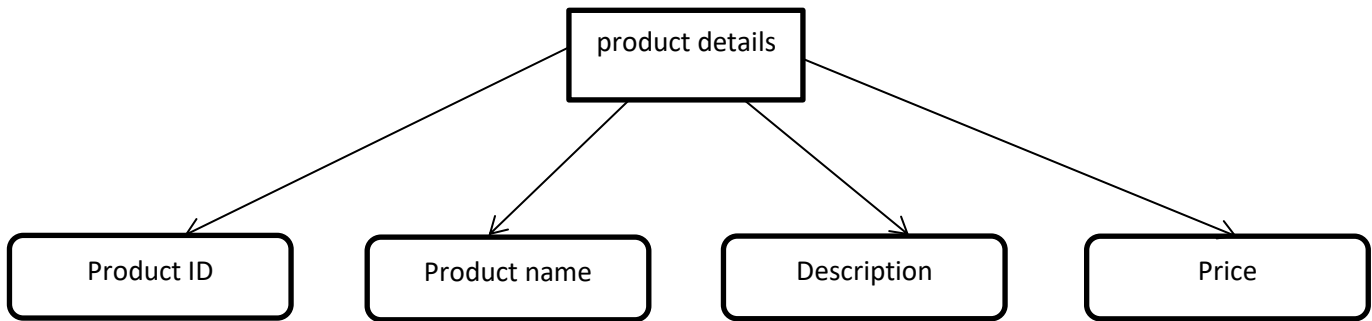
– **Login**



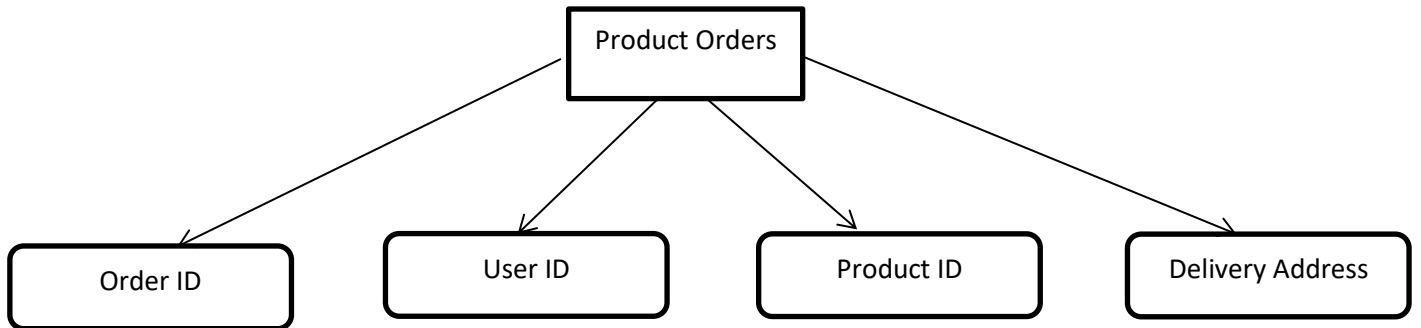
– **User Detail**



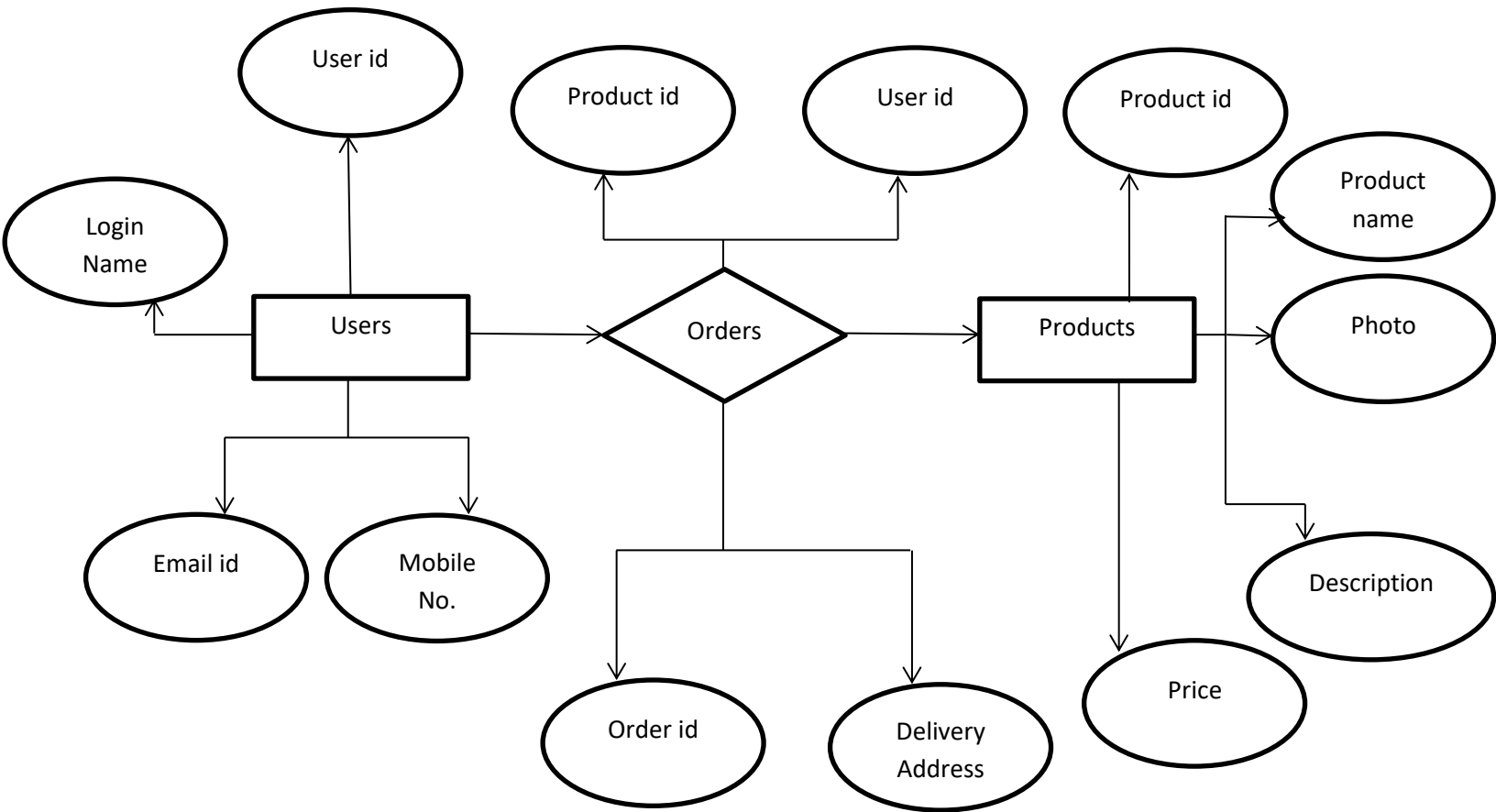
– **Product Details**



– **Product Orders**



– Complete Diagram





- **DATA FLOW DIAGRAM**

A Data Flow Diagram (DFD) is a structured analysis and design tool that can be used for flowcharting. A DFD is a network that describes the flow of data and the processes that change or transform the data throughout a system. This network is constructed by using a set of symbols that do not imply any physical implementation. It has the purpose of clarifying system requirements and identifying major transformations. So it is the starting point of the design phase that functionally decomposes the requirements specifications down to the lowest level of detail. DFD can be considered to an abstraction of the logic of an information-oriented or a process-oriented system flow-chart. For these reasons DFD's are often referred to as logical data flow diagrams.

- **EXTERNAL ENTITY**

An external entity is a source or destination of a data flow. Only those entities which originate or receive data are represented on a data flow diagram. The symbol used is a rectangular box.

- **PROCESS**

A process shows a transformation or manipulation of data flow within the system. The symbol used is an oval shape.

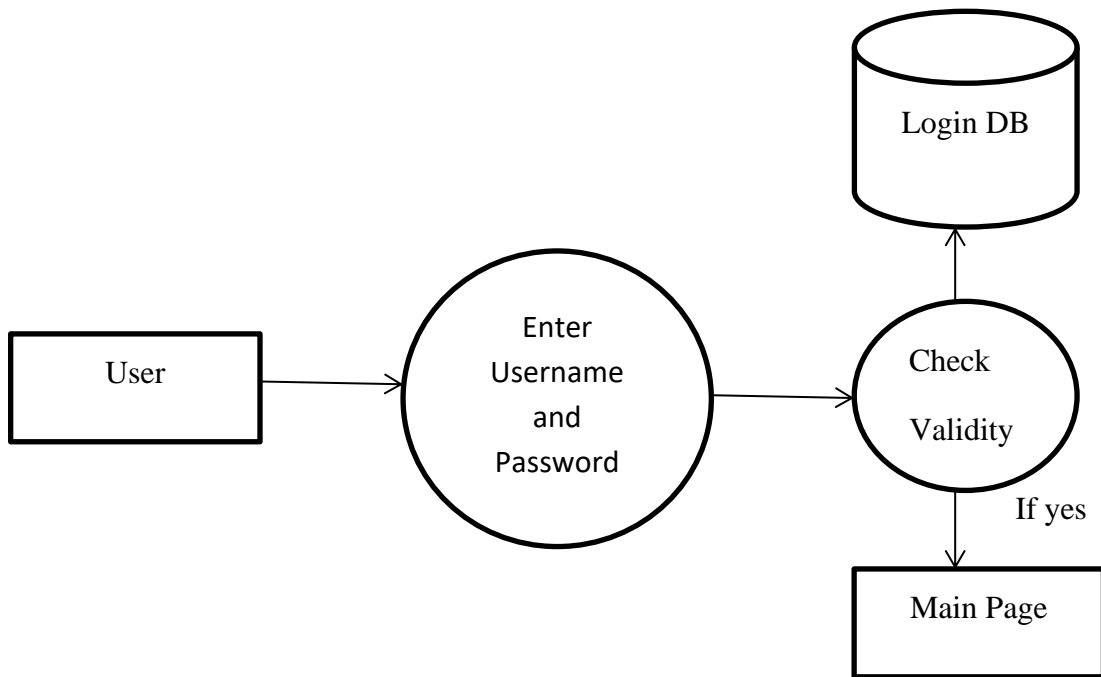
## – DATAFLOW

The data flow shows the flow of information from a source to its destination. Data flow is represented by a line, with arrowheads showing the direction of flow. Information always flows to or from a process and may be written, verbal or electronic. Each data flow may be referenced by the processes or data stores at its head and tail, or by a description of its contents.

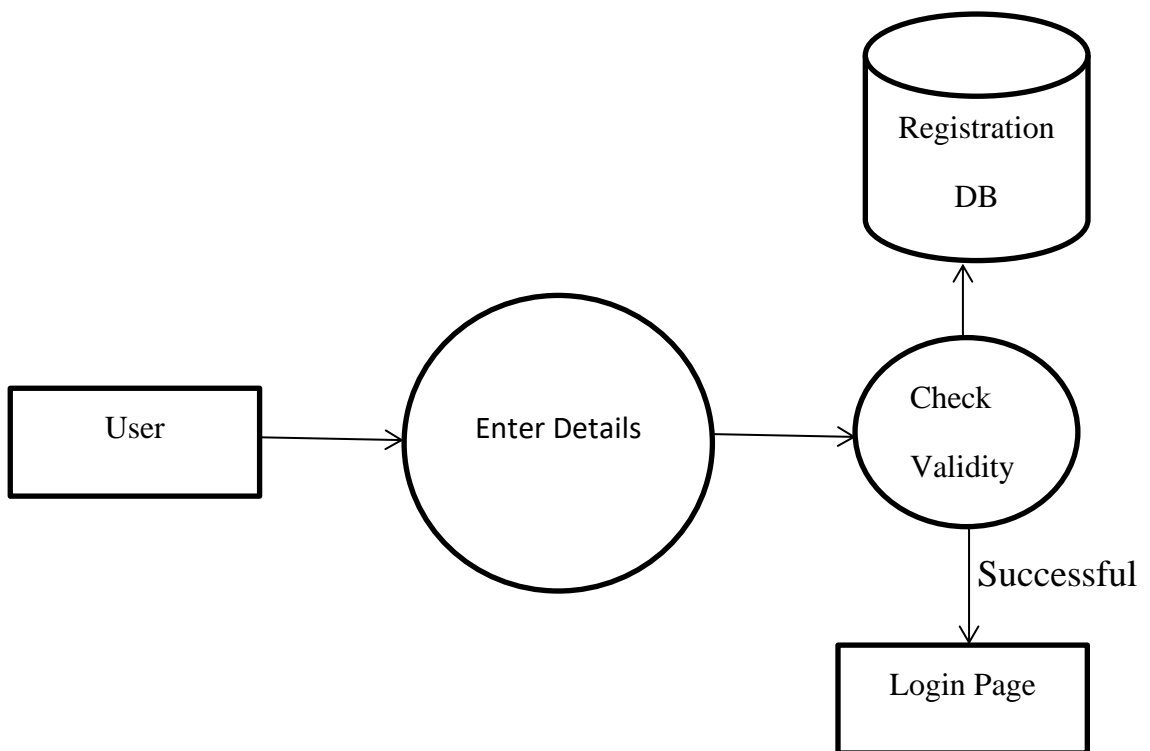
## – DATA STORE

A data store is a holding place for information within the system: It is represented by an open ended narrow rectangle. Data stores may be long-term files such as sales ledgers, or may be short-term accumulations: for example batches of documents that are waiting to be processed. Each data store should be given a reference followed by an arbitrary number.

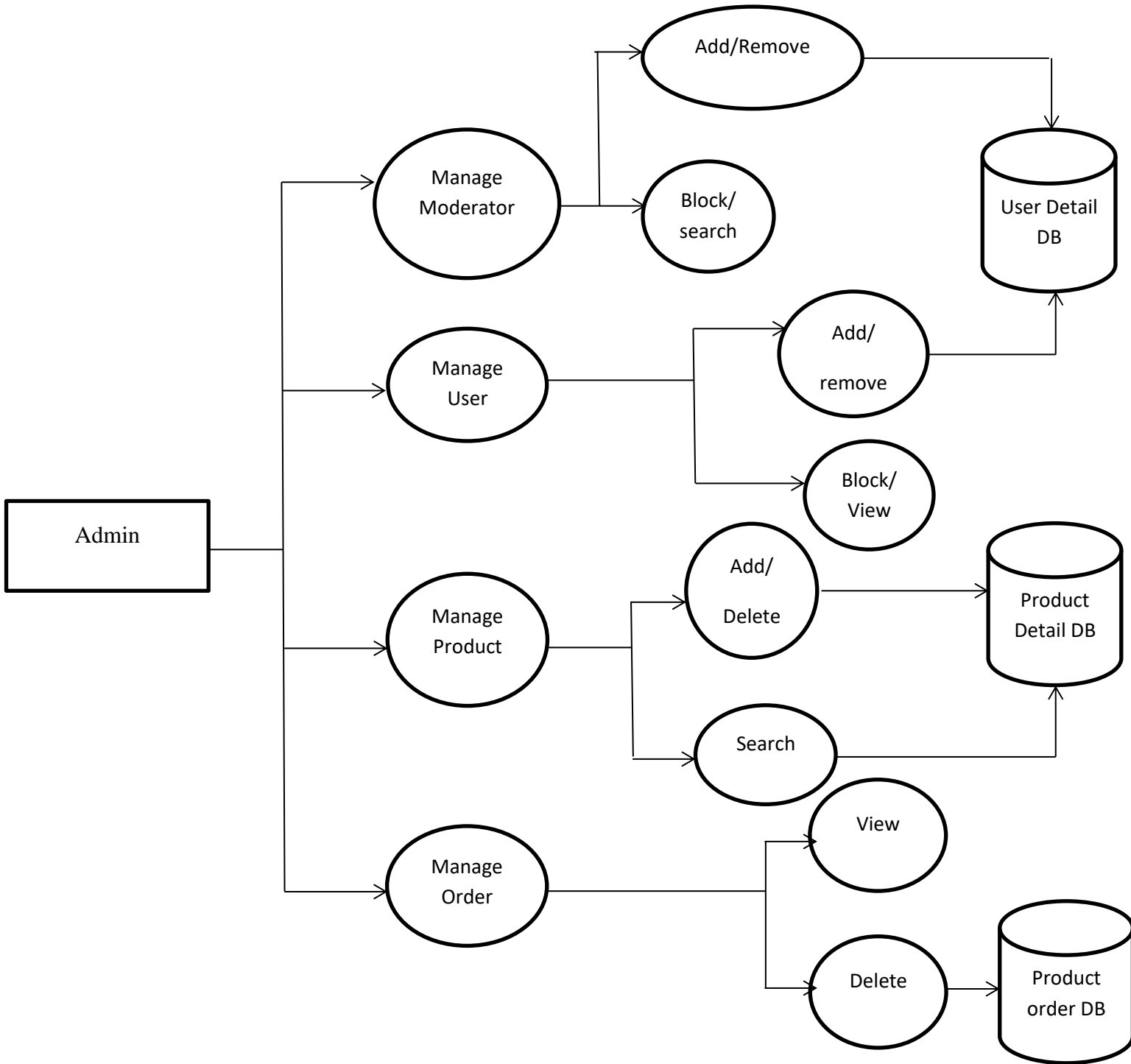
- **LOGIN DFD**



- **Registration DFD**

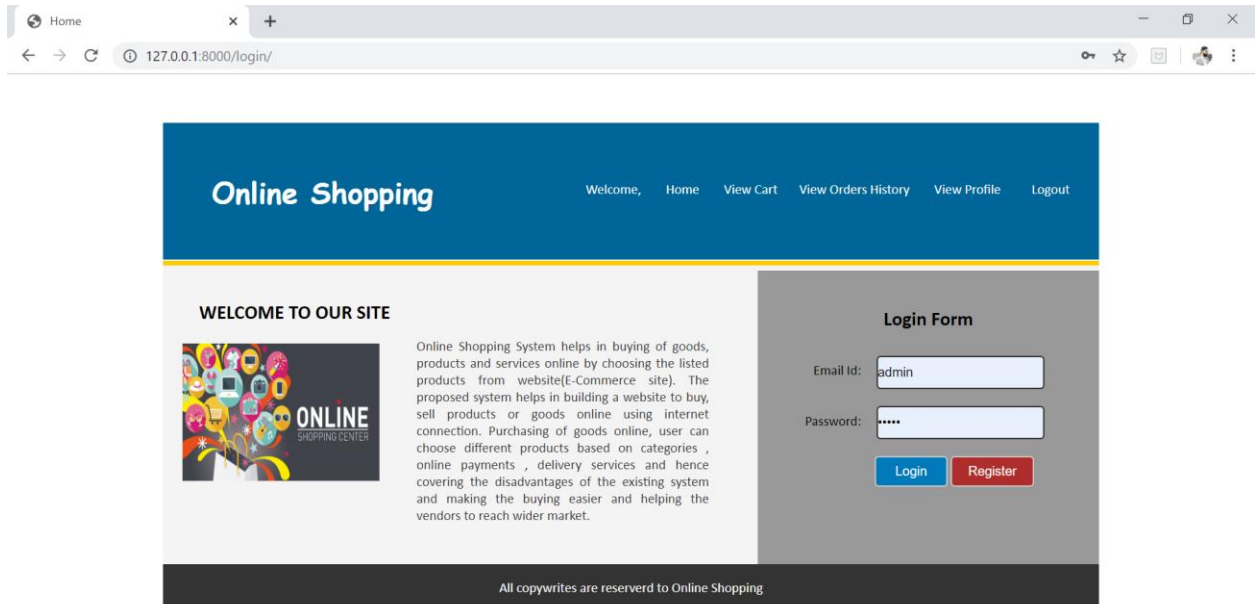


- Admin DFD

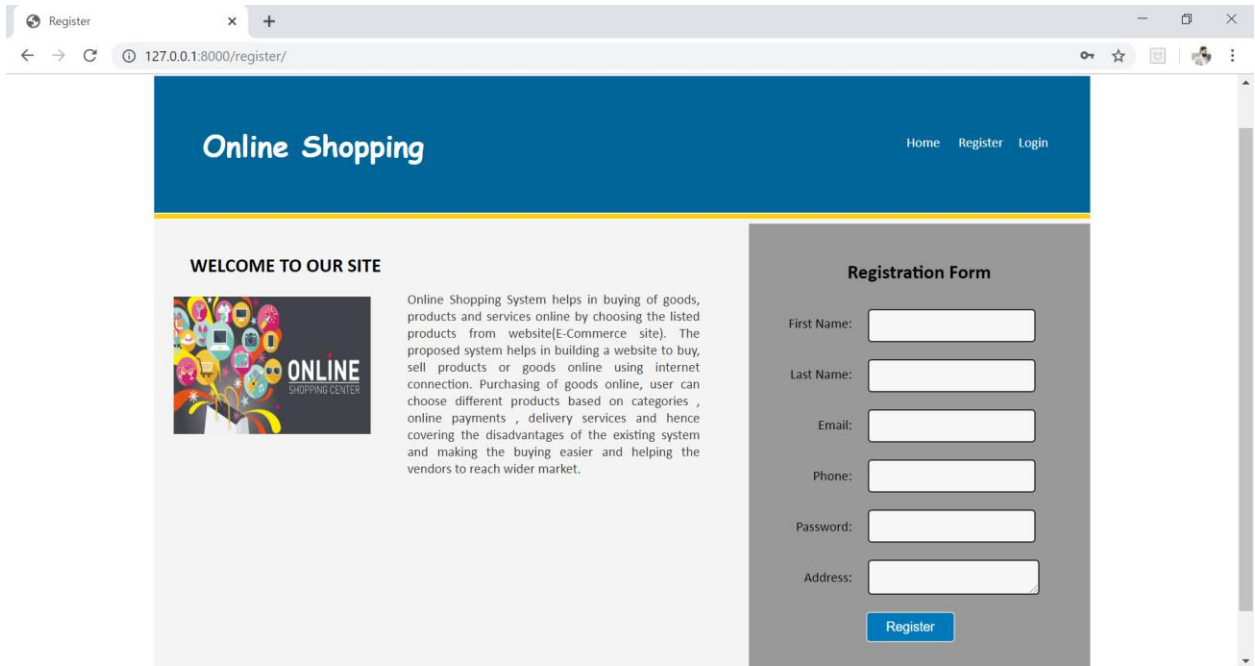


- **Screenshots:**

## **Login-**




## – Registration-



The screenshot shows a web browser window with the title "Register" and the URL "127.0.0.1:8000/register/". The page has a blue header with the text "Online Shopping" and navigation links for "Home", "Register", and "Login". The main content area is divided into two sections: a welcome message and a registration form.

**WELCOME TO OUR SITE**



Online Shopping System helps in buying of goods, products and services online by choosing the listed products from website(E-Commerce site). The proposed system helps in building a website to buy, sell products or goods online using internet connection. Purchasing of goods online, user can choose different products based on categories , online payments , delivery services and hence covering the disadvantages of the existing system and making the buying easier and helping the vendors to reach wider market.

**Registration Form**

First Name:

Last Name:

Email:

Phone:

Password:

Address:




## Home

The screenshot shows the Home page of an online shopping website. The browser address bar displays "127.0.0.1:8000/home/". The page features a blue header with the text "Online Shopping" and navigation links for "Home", "Register", and "Login". On the left side, there is a vertical menu with categories: "Men Wear", "Women Wear", and "Electronics". The main content area displays four product cards, each with a product image, name, price, and an "Add to Cart" button. The products are:

- Angel Fashion Girls Dress: Price: Rs. 450.0/-
- Brand Clothing Men Dress Pants Men: Price: Rs. 425.0/-
- Fashion Shirt Collar Slimming Color Block Shirt: Price: Rs. 540.0/-
- Fashion Trendy Collar Shirt: Price: Rs. 575.0/-

## Product

The screenshot shows the "My Cart" page of the online shopping website. The browser address bar displays "127.0.0.1:8000/viewcart/". The page features a blue header with the text "Online Shopping" and navigation links for "Welcome", "Home", "View Cart", "View Orders History", "View Profile", and "Logout". The main content area displays a table titled "My Cart" with the following items:

Item Name	Price	Quantity	Total	
 Angel Fashion Girls Dress	450.0	1	450.0	<a href="#">Remove from Cart</a>
 Lenovo V310 laptop	29500.0	1	29500.0	<a href="#">Remove from Cart</a>
 Fashion Trendy Collar Shirt	575.0	1	575.0	<a href="#">Remove from Cart</a>

Net Amount: **Rs. 32365.0** /-

Address:

[Order Now](#)

# – Order



## Online Shopping

Welcome, [Home](#) [View Cart](#) [View Orders History](#) [View Profile](#) [Logout](#)

### Payment Details

Amount to Pay:

Payment Mode:

[Pay Now](#)

All copywrites are reserved to Online Shopping



- **Sample Code:**

```
views.py - C:\Users\dell\Desktop\onlineshopping\accounts\views.py (3.7.2)
File Edit Format Run Options Window Help
from django.shortcuts import render, redirect
from .models import Profile
from django.contrib.auth.models import User
from django.contrib.auth.hashers import make_password
from django.contrib.auth import authenticate, login, logout
from django.contrib.auth.decorators import login_required

# Create your views here.
def register_view(request):
    if request.method == 'POST':
        user = User(
            first_name = request.POST['firstname'],
            last_name = request.POST['lastname'],
            email = request.POST['email'],
            username = request.POST['email'],
            password = make_password(request.POST['password']),
        )
        user.save()
        profile = Profile(
            user = user,
            phone = request.POST['phone'],
            address = request.POST['address']
        )
        profile.save()
        return redirect('login')
    else:
        return render(request, 'Register.html')

def login_view(request):
    if request.method == 'POST':
        user = authenticate(
            username = request.POST['email'],
            password = request.POST['password']
        )
        if user is not None:
            login(request, user)
            if 'next' in request.POST:
                return redirect(request.POST['next'])
            return redirect('home')
        return render(request, 'Login.html')

@login_required(login_url='login')
```

Ln: 1 Col: 45

```
views.py - C:\Users\dell\Desktop\onlineshopping\accounts\views.py (3.7.2)
File Edit Format Run Options Window Help
        if 'next' in request.POST:
            return redirect(request.POST['next'])
        return redirect('home')
    return render(request, 'Login.html')

@login_required(login_url='login')
def profile_view(request):
    return render(request, 'ViewProfile.html', {'user': request.user})

@login_required(login_url='login')
def edit_profile_view(request):
    if request.method == 'POST':
        user = User.objects.get(id=request.user.id)
        user.first_name = request.POST['firstname']
        user.last_name = request.POST['lastname']
        user.email = request.POST['email']
        user.username = request.POST['email']
        user.save()
        profile = Profile.objects.get(user_id = request.user.id)
        profile.phone = request.POST['phone']
        profile.address = request.POST['address']
        profile.save()
        return redirect('viewprofile')
    else:
        return render(request, 'EditProfile.html', {'user': request.user})

@login_required(login_url='login')
def change_password_view(request):
    if request.method == 'POST':
        user = User.objects.get(id=request.user.id)
        user.password = make_password(request.POST['newpassword'])
        user.save()
        return redirect('home')
    else:
        return render(request, 'ChangePassword.html',
            {'user': User.objects.get(id=request.user.id)})

@login_required(login_url='login')
def logout_view(request):
    logout(request)
    return redirect('login')
```

Ln: 1 Col: 45



```
69      <input type="hidden" name="csrfmiddlewaretoken" value="a8J6K1XqXyaPstaVZEEdHUARa4zb08rbDE2ZMKP1SIkK9KCGUytLuZtxs10BfISVd">
70      &nbsp;&nbsp;&nbsp;Quantity: &nbsp;&nbsp;&nbsp;
71      <select name="quantity" class="iform">
72          <option value="1">1</option>
73          <option value="2">2</option>
74          <option value="3">3</option>
75          <option value="4">4</option>
76          <option value="5">5</option>
77      </select>
78      <br><br>
79      <input name="itemname" type="hidden" value='Brand Clothing Men Dress Pants Men' />
80      <input name="price" type="hidden" value='425.0' />
81      <input name="image" type="hidden" value='/media/products_images/brand-clothing-men-dress-pants-mens.jpg' />
82      <input class="bluebutton" value="Add to Cart" type="submit" />
83      </form>
84      </div>
85      </div>
86
87      <div class="item">
88          
89          <div class="itemd">
90              <font size="+1.5">Fashion Shirt Collar Slimming Color Block Shirt</font><br><br>
91              Price: <font color="#AD1025">Rs. 540.0</font>
92          </div>
93          <div class="itemb">
94              <form action="/addtocart/" method="post">
95                  <input type="hidden" name="csrfmiddlewaretoken" value="a8J6K1XqXyaPstaVZEEdHUARa4zb08rbDE2ZMKP1SIkK9KCGUytLuZtxs10BfISVd">
96                  &nbsp;&nbsp;&nbsp;Quantity: &nbsp;&nbsp;&nbsp;
97                  <select name="quantity" class="iform">
98                      <option value="1">1</option>
99                      <option value="2">2</option>
100                     <option value="3">3</option>
101                     <option value="4">4</option>
102                     <option value="5">5</option>
103                 </select>
104                 <br><br>
105                 <input name="itemname" type="hidden" value='Fashion Shirt Collar Slimming Color Block Shirt' />
106                 <input name="price" type="hidden" value='540.0' />
107                 <input name="image" type="hidden" value='/media/products_images/Fashion-Shirt-Collar-Slimming-Color-Block-Shirt.jpg' />
108                 <input class="bluebutton" value="Add to Cart" type="submit" />
109             </form>
110         </div>
111     </div>
112
113     <div class="item">
114         
115         <div class="itemd">
116             <font size="+1.5">Fashion Trendy Collar Shirt</font><br><br>
117             Price: <font color="#AD1025">Rs. 575.0</font>
118         </div>
119         <div class="itemb">
120             <form action="/addtocart/" method="post">
121                 <input type="hidden" name="csrfmiddlewaretoken" value="a8J6K1XqXyaPstaVZEEdHUARa4zb08rbDE2ZMKP1SIkK9KCGUytLuZtxs10BfISVd">
122                 &nbsp;&nbsp;&nbsp;Quantity: &nbsp;&nbsp;&nbsp;
123                 <select name="quantity" class="iform">
124                     <option value="1">1</option>
125                     <option value="2">2</option>
126                     <option value="3">3</option>
127                     <option value="4">4</option>
128                     <option value="5">5</option>
129                 </select>
130                 <br><br>
131                 <input name="itemname" type="hidden" value='Fashion Trendy Collar Shirt' />
132                 <input name="price" type="hidden" value='575.0' />
133                 <input name="image" type="hidden" value='/media/products_images/Fashion-trendy-collar-shirt.jpg' />
134                 <input class="bluebutton" value="Add to Cart" type="submit" />
135             </form>
136         </div>
137     </div>
138
139     <div class="item">
140         
141         <div class="itemd">
142             <font size="+1.5">Hot Men Designer Biker Jeans</font><br><br>
```

```
103     </select>
104     <br><br>
105     <input name="itemname" type="hidden" value='Fashion Shirt Collar Slimming Color Block Shirt' />
106     <input name="price" type="hidden" value='540.0' />
107     <input name="image" type="hidden" value='/media/products_images/Fashion-Shirt-Collar-Slimming-Color-Block-Shirt.jpg' />
108     <input class="bluebutton" value="Add to Cart" type="submit" />
109 </form>
110 </div>
111
112 <div class="item">
113     
114     <div class="itemd">
115         <font size="+1.5">Fashion Trendy Collar Shirt</font><br><br>
116         Price: <font color="#AD1025">Rs. 575.0</font>
117     </div>
118     <div class="itemb">
119         <form action="/addtocart/" method="post">
120             <input type="hidden" name="csrfmiddlewaretoken" value="a8J6K1XqXyaPstaVZEEdHUARa4zb08rbDE2ZMKP1SIkK9KCGUytLuZtxs10BfISVd">
121             &nbsp;&nbsp;&nbsp;Quantity: &nbsp;&nbsp;&nbsp;
122             <select name="quantity" class="iform">
123                 <option value="1">1</option>
124                 <option value="2">2</option>
125                 <option value="3">3</option>
126                 <option value="4">4</option>
127                 <option value="5">5</option>
128             </select>
129             <br><br>
130             <input name="itemname" type="hidden" value='Fashion Trendy Collar Shirt' />
131             <input name="price" type="hidden" value='575.0' />
132             <input name="image" type="hidden" value='/media/products_images/Fashion-trendy-collar-shirt.jpg' />
133             <input class="bluebutton" value="Add to Cart" type="submit" />
134         </form>
135     </div>
136 </div>
137
138 <div class="item">
139     
140     <div class="itemd">
141         <font size="+1.5">Hot Men Designer Biker Jeans</font><br><br>
```



```
207       </select>
208       <br/><br/>
209       <input name="itemname" type="hidden" value='Latest Fashion Design Slim Fit Men' />
210       <input name="price" type="hidden" value='675.0' />
211       <input name="image" type="hidden" value="/media/products_images/latest-fashion-design-slim-fit-mens.jpg" />
212       <input class="bluebutton" value="Add to Cart" type="submit" />
213     </form>
214   </div>
215 </div>
216
217 <div class="item">
218   
219   <div class="itemd">
220     <font size="+1.5">Lenovo V310 Laptop</font><br/><br/>
221     Price: <font color="#A1025">Rs. 29500.0</font>
222   </div>
223   <div class="itemb">
224     <form action="/addtocart/" method="post">
225       <input type="hidden" name="csrfmiddlewaretoken" value="a8J6K1XqXyaPstaVZEedHUARa4zb08rbDE2ZMKP1S1kk9KcgUytLuZtxs10BfISVd">
226       &nbsp;&nbsp;&nbsp;Quantity: &nbsp;&nbsp;&nbsp;
227       <select name="quantity" class="iform">
228         <option value="1">1</option>
229         <option value="2">2</option>
230         <option value="3">3</option>
231         <option value="4">4</option>
232         <option value="5">5</option>
233       </select>
234       <br/><br/>
235       <input name="itemname" type="hidden" value='Lenovo V310 laptop' />
236       <input name="price" type="hidden" value='29500.0' />
237       <input name="image" type="hidden" value="/media/products_images/Lenovo-V310-laptop.png" />
238       <input class="bluebutton" value="Add to Cart" type="submit" />
239     </form>
240   </div>
241 </div>
242
243 <div class="item">
244   
245   <div class="itemd">
246     <font size="+1.5">Men Black Jeans</font><br/><br/>
```

```
244   
245   <div class="itemd">
246     <font size="+1.5">Men Black Jeans</font><br/><br/>
247     Price: <font color="#A1025">Rs. 580.0</font>
248   </div>
249   <div class="itemb">
250     <form action="/addtocart/" method="post">
251       <input type="hidden" name="csrfmiddlewaretoken" value="a8J6K1XqXyaPstaVZEedHUARa4zb08rbDE2ZMKP1S1kk9KcgUytLuZtxs10BfISVd">
252       &nbsp;&nbsp;&nbsp;Quantity: &nbsp;&nbsp;&nbsp;
253       <select name="quantity" class="iform">
254         <option value="1">1</option>
255         <option value="2">2</option>
256         <option value="3">3</option>
257         <option value="4">4</option>
258         <option value="5">5</option>
259       </select>
260       <br/><br/>
261       <input name="itemname" type="hidden" value='Men Black Jeans' />
262       <input name="price" type="hidden" value='580.0' />
263       <input name="image" type="hidden" value="/media/products_images/Mens-black-jeans.jpg" />
264       <input class="bluebutton" value="Add to Cart" type="submit" />
265     </form>
266   </div>
267 </div>
268
269 <div class="item">
270   
271   <div class="itemd">
272     <font size="+1.5">Men Formal Shirt</font><br/><br/>
273     Price: <font color="#A1025">Rs. 485.0</font>
274   </div>
275   <div class="itemb">
276     <form action="/addtocart/" method="post">
277       <input type="hidden" name="csrfmiddlewaretoken" value="a8J6K1XqXyaPstaVZEedHUARa4zb08rbDE2ZMKP1S1kk9KcgUytLuZtxs10BfISVd">
278       &nbsp;&nbsp;&nbsp;Quantity: &nbsp;&nbsp;&nbsp;
279       <select name="quantity" class="iform">
280         <option value="1">1</option>
281         <option value="2">2</option>
282         <option value="3">3</option>
```







## • CONCLUSION

The project entitled Online shopping system was completed successfully. The system has been developed with much care and free of errors and at the same time it is efficient and less time consuming. The purpose of this project was to develop a web application and an android application for purchasing items from a shop. This project helped us in gaining valuable information and practical knowledge on several topics like designing web pages using html & css, usage of responsive templates, designing of android applications, and management of database using mysql . The entire system is secured. Also the project helped us understanding about the development phases of a project and software development life cycle. We learned how to test different features of a project. This project has given us great satisfaction in having designed an application which can be implemented to any nearby shops or branded shops selling various kinds of products by simple modifications. There is a scope for further development in our project to a great extent. A number of features can be added to this system in future like providing moderator more control over products so that each moderator can maintain their own products. Another feature we wished to implement was providing classes for customers so that different offers can be given to each class. System may keep track of history of purchases of each customer and provide suggestions based on their history. These features could have implemented unless the time did not limited us.



## REFERENCES:

1. <https://readthedocs.org/projects/django-shop/downloads/pdf/latest/>
2. [https://platforma-kooperativa.org/media/uploads/article-documents/beginning\\_django\\_e-commerce.pdf](https://platforma-kooperativa.org/media/uploads/article-documents/beginning_django_e-commerce.pdf)
3. <https://snipcart.com/blog/django-ecommerce-tutorial-wagtail-cms>
4. <https://docs.djangoproject.com/en/3.0/>