SYNOPSIS OF ONLINE SHOPPING SYSTEM

by

Syed Mohd Adnan

Abstract

Online Shopping play a great importance in the modern business environment. It is a web-based application intended for existing local retailers. This project is an attempt to provide the advantages of online shopping to customers of a real shop. It helps buy the products in the shop anywhere through the internet by using any device. Thus, the customer will get the service of online shopping and home delivery from his favorite shop. This system can be implemented in any shop in the locality or multinational branded shops having retail outlet chains. 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 e-commerce platform such as Flipkart or amazon.

Solitude and safety risk emerge regularly as a reason for being cautious about internet shopping. Shopping convenience, information seeking, social contact, and diversity affects the consumer attitude towards online shopping. The impossibility of product testing, problems with complaints, product return and missus of personal data are the main doubts regarding on-line shopping

Contents

Introduction

Traditionally, customers are used to buying the products at the real, in other words, factual shops or supermarkets. It needs the customers to show up in the shops in person, and walk around different shopping shelves, and it also needs the owners of shops to stock, exhibit, and transfer the products required by customers. It takes labor, time, and space to process these operations. Online shopping is the process whereby consumers directly buy goods or services from a seller in real-time, 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 of a real shop. It helps buy the products in the shop anywhere on the internet by using any device. Thus, the customer will get the service of online shopping and home delivery from his favorite shop.

Need of the application

Problem

The spread of the Covid-19 pandemic has caused a lot of changes in our lifestyle, people fearing to get outside their homes, transportation almost shut down and social distancing becoming all the more important. Big to small-scale businesses that relied on the traditional incur a lot of consequences due to the lockdown issues. Some tend to be more towards using social media platforms like Facebook and WhatsApp to sell their product. However, the social media platforms have been beneficial for marketing purposes alone but leave the whole task of the customer and massive order management via direct messaging (DM), which takes a lot of time to respond to all customers. In addition, everyone tends to use social media, posing a great challenge to differentiate between scammers (fraudsters) and legit sellers.

Furthermore, this project is an attempt to provide the advantages of online shopping to customers of a real shop, as more and more Local shops are losing to big e-commerce platforms, with only choice to sell their products on those platforms and give them their profit cuts.

Solution

Shopping has long been considered a recreational activity by many. Shopping online is no exception. The goal of this application is to develop a web-based interface for online retailers. The system would be easy to use and hence make the shopping experience pleasant for the users.

Objective

The goal of this application is:

- To develop an easy-to-use web-based interface where users can browse for products.
- To develop an easy-to-use web-based interface where users can search for products.
- To develop an easy-to-use web-based interface where users can view a complete description of the products.
- To develop an easy-to-use web-based interface where users can add the product to the cart and order the products.

Project Scope

This system can be implemented in any shop in the locality or multinational branded shops having retail outlet chains. The system recommends a facility to accept the orders 24*7 and a home delivery system that 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 Flipkart
or Amazon. Since the application is available on the internet it is easily accessible and always available on any device.

Project Category

OOP (Object Oriented Programming)

- The project uses the OOP paradigm which is supported by Python.
- Django is a web framework written in Python that uses OOP as a programming paradigm for Class-Based Views.

The Three OOP Principals:

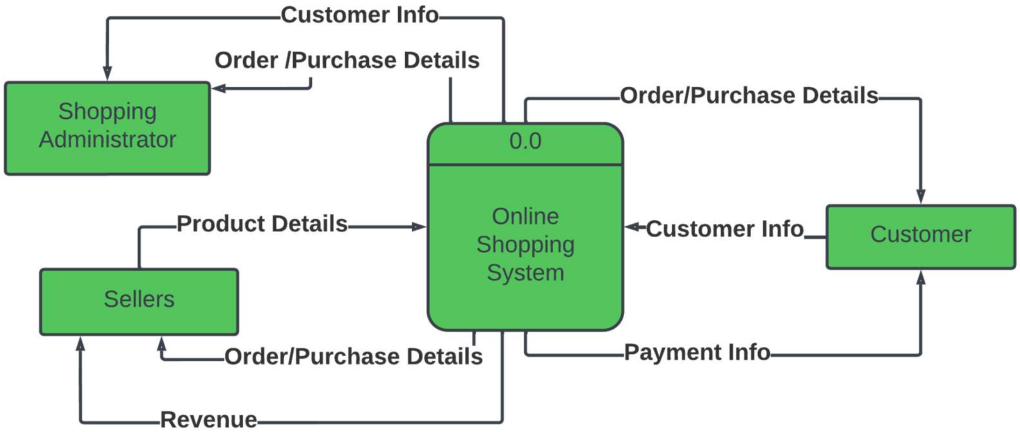
1. **Encapsulation:** Encapsulation is a mechanism that binds together code and the data it manipulates and keeps both safe outside interference and misuse.
2. **Inheritance:** It is the process by which one object acquires the properties of another object. This is important because it supports the concepts of hierarchical classification.
3. **Polymorphism:** Polymorphism is a feature that allows one interface to be used for the general class of actions.

RDBMS (Relational Database Management System)

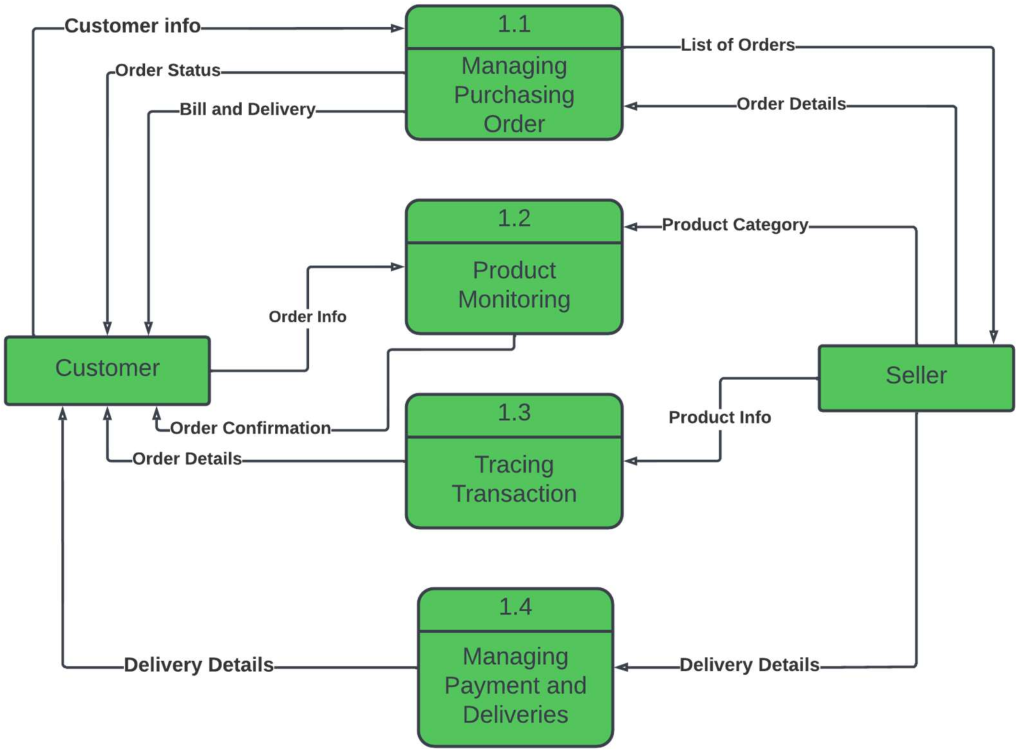
Project uses RDBMS which is supported by SQLite which is file-based RDBMS that does not require any installation or setup. This, in turn, means that the application does not run under a separate server process that needs to be started, stopped, or configured. This serverless architecture enables the database to be cross-platform compatible.

ANALYSIS

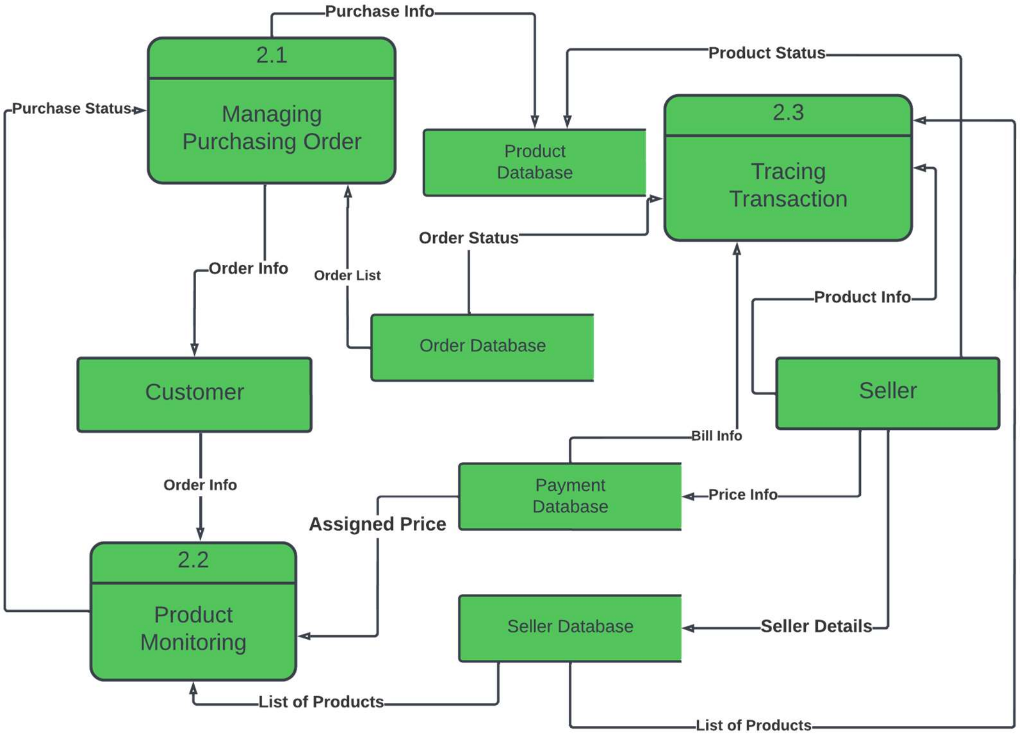
Level 0 DFD:



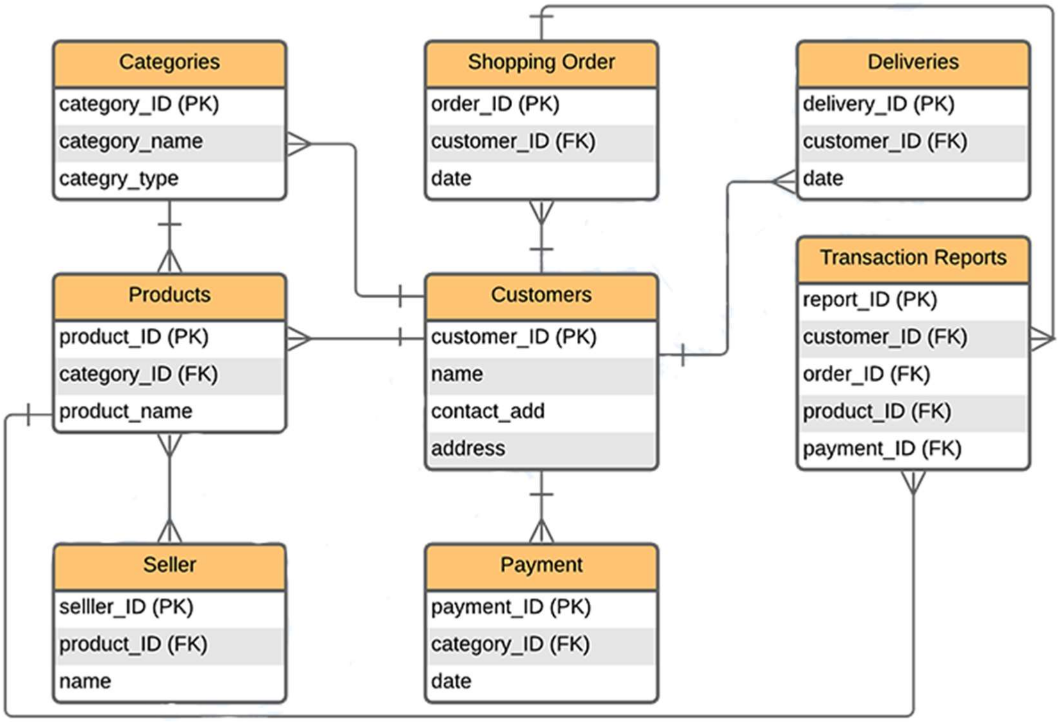
Level 1 DFD:


Customer info
1.1
Managing Purchasing Order
List of Orders
Order Status
Order Details
Bill and Delivery
1.2
Product Category
Product Monitoring
Order Info
Customer
Seller
1.3
Product Info
Order Confirmation
Order Details
Tracing Transaction
1.4
Managing Payment and Deliveries
Delivery Details
Delivery Details

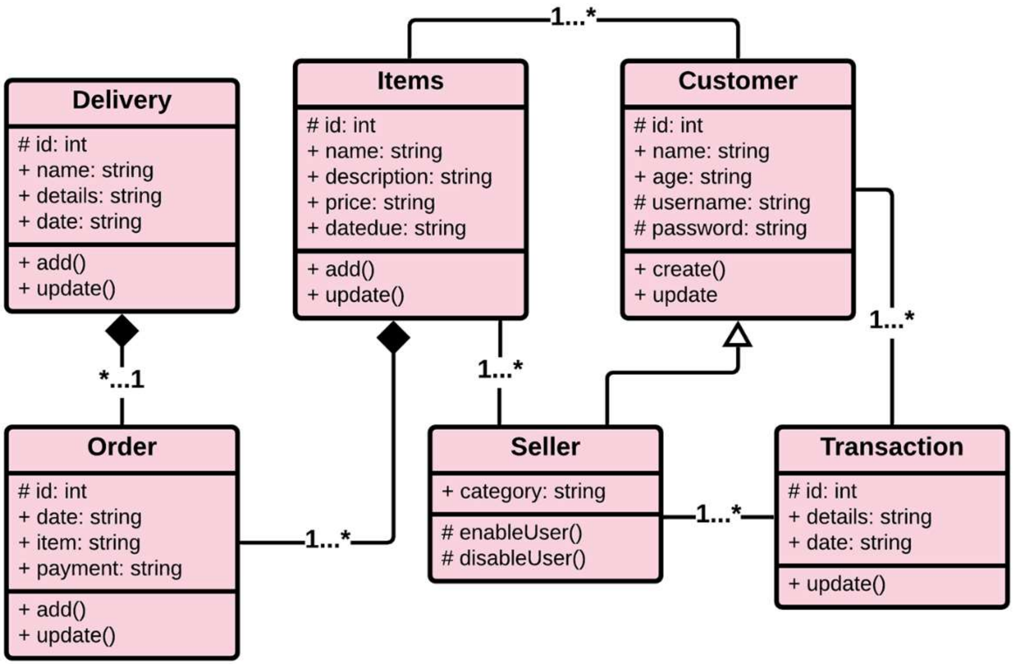
Level 2 DFD:


Purchase Info
2.1
Managing Purchasing Order
Product Status
Purchase Status
Product Database
2.3
Tracing Transaction
Order Status
Order Info
Order List
Product Info
Order Database
Customer
Seller
Bill Info
Order Info
Payment Database
Price Info
Assigned Price
2.2
Product Monitoring
Seller Database
Seller Details
List of Products
List of Products

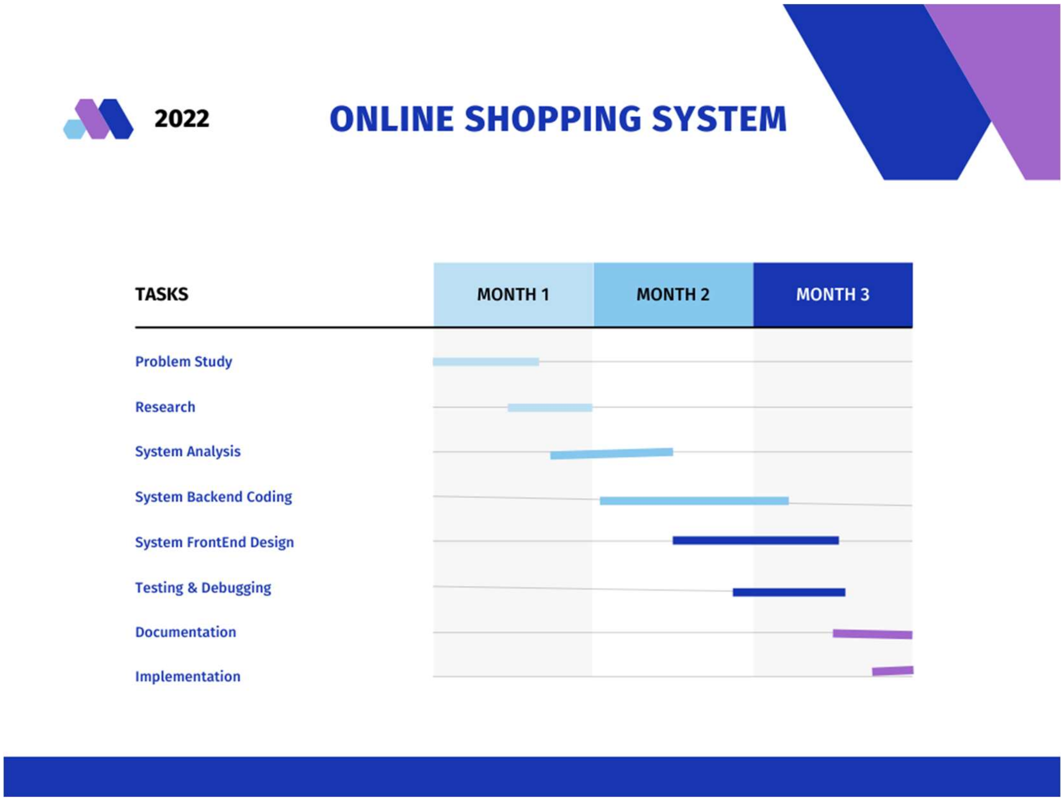
ENTITY-RELATIONSHIP DIAGRAM:



Class Diagram:



GANTT CHART:


2022

ONLINE SHOPPING SYSTEM

Database Design

Table Name: **Customer**

Field	**Description**	**Type**	**Length**
customer_id	Customer Id	int	11
name	Custormer Name	varchar	255
contact_no	Contact No	int	11
address	Address	varchar	255

Table Name: **Products**

Field	**Description**	**Type**	**Length**
product_id	Product Id	int	11
category_id	Category Id	int	11
product_name	Product Name	varchar	255

Table Name: **Categories**

Field	**Description**	**Type**	**Length**
category_id	Category Id	int	11
category_name	Category Name	varchar	255

Table Name: **Order**

Field	**Description**	**Type**	**Length**
order_id	Order Id	int	11
product_id	Product_id	int	11
date	Date	datetime	

Table Name: **Payment**

Field	**Description**	**Type**	**Length**
payment_id	Payment Id	int	11
order_id	Order Id	int	11
date	Date	datetime	

Table Name: **Deliveries**

Field	**Description**	**Type**	**Length**
delivery_id	Delivery Id	int	11
order_id	Order Id	int	11
payment_id	Payment Id	int	11
date	Date	datetime	

Structure

Modules

Module 1: **Administrator Module**

The Administrator module of the site will have the following responsibilities:

- An Administrator can add, delete, or modify the Store, Category, and Products.
- An administrator shall control all activities taking place on this web portal.
- An administrator shall be able to view customer orders.
- The admin shall be able to view the reports.

Module 2: **User Module**

- The User who is trying to access the Online Shopping must log in to the application by providing a login id and password. A log file is created on the server-side view, the details of the user who tried to access the system.
- The unregistered user must have to register for purchasing any product.
- Registration of User will have Name, Email-Id, & Password.
- Only a valid User can buy a product.

Module 3: **Search Module**

- Users can search on the site about products. This module is available to every valid/invalid user of this site.

Module 4: **Shopping Cart**

- Registered User is only able to send products to the cart before ordering them.

Module 5: **Products**

- Every Product will have a product ID, Name Model Number, Market Price Own price, Warranty, and Image.

Module 6: **Report Generation**

The system shall be able to generate various reports like:

- Order Slips for Customers.
- Daily Report of the orders placed by customers.
- Report about product availability.

Process Logic

Online Shopping System's Main Functionality is to manage and provide the best Products available on the internet to a visitor of the site. It helps the visitors to browse the available Products and search for them the whole Project goes as follows:

- The visitor visits the site to Search or Browse the product view its details and selects the product.
- If the visitor selects the Product, then he is authenticated for his validity. The control flows to the login page where only a valid user can log in.
- If a user is already registered then he gives his Email-id and password. Then this information is verified by the user data source. If it founds a match in the data source then the user's selection is moved to the cart.
- If a user is not registered then he must register himself first. A User can register himself just by giving essential details. These details are stored in the User database.
- On Ordering finalize product from the Cart The system asks for the user's bank details for Payments.

Now for the process followed by the administrator:

- The administrator performs the controlling functions of the whole system.
- He adds, deletes, and modifies various Products and categories to the database.
- He manages the inventory and manages the product stock if the product is unavailable, he can remove the product so users don't buy that product.
- The administrator generates various reports on the Products made by the user. This helps higher authorities evaluate revenue earned by the firm.
- The administrator can view the orders made by the users, to confirm they generate order invoices and ship them to users.

Methodology

A methodology is a model, which project managers employ for the design, planning, implementation, and achievement of their project objectives. Effective project management is essential in absolutely any organization, regardless of the nature of the business and the scale of the organization. From choosing a project right through to the end, the project must be carefully and closely managed. Based on the nature of my project solution, it was essential to use an incremental **Software development life cycle** (SDLC). The project typically has several Phases and the level of control required over each phase is primarily defined by the nature of the project, its complexity of the same, and the industry to which the Project has to cater. An Incremental (SDLC) model consists of several dependent increments that are completed in a prescribed sequence. Each increment includes Models, Views, and Templates for the functions and features. integrates additional parts of the solution until the final increment, where the remaining parts of the solution are integrated.

Architecture

Django is based on MVT (Model-View-Template) architecture. MVT is a software design pattern for developing a web application.

MVT Structure has the following three parts:

1. **Model:** The model is going to act as the interface of your data. It is responsible for maintaining data. It is the logical data structure behind the entire application and is represented by a database (generally relational databases such as MySQL, and SQLite).
2. **View:** The View is the user interface ' that you see in your browser when you render a website. It is represented by HTML/CSS/JavaScript and Jinja files.
3. **Template:** A template consists of static parts of the desired HTML output as well as some special syntax describing how dynamic content will be inserted.

Testing Process

Testing is vital. Without properly testing your code, you will never know if the code works as, it should, now or in the future when the codebase changes. Countless hours can be lost fixing problems caused by changes to the codebase. What's worse, you may not even know that there are problems at all until your end-users complain about them, which is not how you want to find out about code breaks.

Having tests in place will help ensure that if a specific function breaks you will know about it. Tests also make debugging breaks in code much easier, which saves time and money.

Testing helps you structure good code, find bugs, and write documentation.

Types of tests:

Unit and integration are the two main types of tests:

1. Unit Tests are isolated tests that test one specific function.
2. Integration Tests, meanwhile, are larger tests that focus on user behavior and testing entire applications. Put another way, integration testing combines different pieces of code functionality to make sure they behave correctly.

Best Practices:

- If it can break, it should be tested. This includes models, views forms, templates, validators, and so forth.
- Each test should generally only test one function.
- Keep it simple. You do not want to write a test on top of other tests.
- Run test whenever code is pulled or pushed from the repository
- When upgrading to a newer version of Django:
    - upgrade locally,
    - run your tests
    - fix bugs
    - PUSH to the repository in a new branch and staging,
    - test again before merging to the main branch.

Structure:

Structure your tests to fit your Project. I tend to favor putting all tests for each app in the tests.py file and grouping tests by what I'm testing - e.g., models, views, forms, etc.

```
Django
└── Online Shopping System
    └── tests
        ├── __init__.py
        ├── test_forms.py
        ├── test_models.py
        └── test_views.py
```

Reports generation

The project report will tentatively contain the following topics in detail:

- Methodology
- Use of MVT architecture in Django:
    - Model
    - View
    - Template

- Python (Django) usage.
- Database Design and CRUD Operations in Code.
- Deployment
- Security
- Limitations
- Miscellaneous

Tools / Platform

Hardware Requirement Specification:

- Intel Core i3+ PC to run locally.
- 4gb+ RAM to run a decent modern web browser.
- Monitor to view the web application.
- Preferably Linux installation Windows will also work.
- Internet Connectivity to download required packages.

Software Requirement Specification:

- Python3
- pip3
- Virtual Environment Wrapper
- Django
- git - (to manage versions)
- Visual Studio Code
- PyCharm
- sqllite3
- HTML
- CSS
- Bootstrap3
- Any Modern Web Browser (Chrome, Edge, Brave, etc.)
- GitHub with account to back up the code.

Terminology Meaning:

DFD	Data Flow Diagram
DBD	Database Design
HTTP	Hyper Text Transfer Protocol
ERD	Entity Relationship Diagram
SQL	Structured Query Language
DBMS	Database Management System

Scope For Future Enhancement

The System is highly Flexible and Scalable well efficient to can deploy and run on a server anytime it can handle any further development it can be developed into Full-fledged E-Commerce.