

GET-ITEM

(E-commerce)

A mini project report submitted
for partial fulfillment for the award of the degree

BACHELOR OF TECHNOLOGY

in

COMPUTER SCIENCE & ENGINEERING

SUBMITTED BY:

- S. Sai Mohan (Team Lead) - 319129510043
- S. Uma Devi - 319129510041
- P.Lokesh - 319129510034
- P.Aruna Suma Sree - 319129510036



Under the Esteemed Guidance of

Mr. M. Hari Naga Raju

Asst. Prof. Department of CSE , WISTM.

**DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING WELFARE
INSTITUTE OF SCIENCE TECHNOLOGY AND MANAGEMENT**

(Approved AICTE ,DELHI&affiliated to Andhra University) Pinagadi

Pendurthi(M),Visakhapatnam-531173

WELFARE INSTITUTE OF SCIENCE TECHNOLOGY AND MANAGEMENT

**Approved AICTE ,DELHI&Affiliated to Andhra
University(V), Pendurthi(M),Visakhapatnam-531173**



CERTIFICATE

**This is to certify that the mini project report entitled
"GET-ITEM(e-commerce) "**

SUBMITTED BY:

- S.Sai Mohan (Team Lead) - 319129510043
- S. Uma Devi - 319129510041
- P. Lokesh - 319129510034
- P.Aruna Suma Sree - 319129510036

For partial fulfillments for the award of the degree Bachelor of Technology in computer science and Engineering is a bonafide project work carried out by him .This work is not submitted any University for the award of any degree.

INTERNAL GUIDE

Mr.M.Hari Naga Raju
Assistant Professor,
Department of CSE ,
WISTM , ENGG.COLLEGE

HEAD OF THE DEPARTMENT

Mrs.K.V.LAKSHMI,M.TECH
Assistant Professor,
Department of CSE ,
WISTM , ENGG.COLLEGE

EXTERNAL EXAMINER

ACKNOWLEDGEMENT

The satisfaction that accompanies the successful completion of any task would be incomplete without the mention of people who made it possible and whose constant guidance and encouragement crown all the effort with success. I like to thank **Prof. M. James Stephen**, Principal, of Welfare Technology and Management , for this kind cooperation that gives me boundless pleasure to avail this opportunity to express my deep sense of gratitude and while hearted thanks to Head of Department **Mrs.K.V.Lakshmi, Assistant Professor**, department of computer science and Engineering and my project guide **Mr.M. Hari Naga Raju, Assistant Professor**, Department of Computer Science and Engineering, and all the faculty of computer science department for their valuable suggestions and co-operation during the development of this project. I am really grateful to my parents and all my friends for their constructive suggestions to do this project in effective manner

SUBMITTED BY:

- S.Sai Mohan (Team Lead) - 319129510043
- S.Uma Devi - 319129510041
- P.Lokesh - 319129510033
- P.Aruna Suma Sree - 319129510037

DECLARATION

I here declare that the project entitled “online library management system” has been done under the guidance of **Mr. M. Hari Naga Raju Asst. Prof. Department of CSE** and is dissertation of my own work except where specifically ask to the contrary and is submitted to the department of computer science and Engineering, wellfare Insistute of science technology and management for the partial fulfilment of the requirement for the award of B.Tech degree.

SUBMITTED BY:

- S.Sai Mohan (Team Lead) - 319129510043
- S.Uma Devi - 319129510041
- P. Lokesh - 319129510034
- P.Aruna Suma Sree - 319129510036

INTRODUCTION

LITERATURE:

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 of their desire from the store. The information pertaining to the products are stores on 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 two modules first the customers who wish to buy the items. 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. This is an e-platform but there is very less number of secured online shopping applications.

SCOPE OF PROJECT:

Purchasing and selling products and services over the internet without the need of going physically to the market is what online shopping all about. Online shopping is just like a retail store shopping that we do by going to the market, but it is done through the internet. Online shopping has made shopping painless and added more fun. Online stores offer product description, pictures, comparisons, price and much more. Few examples of these are Amazon.com, ebay.com, framt.com and the benefits of online shopping is that by having direct access to consumer, the online stores can offer products that cater to the needs of consumer ,cookies can be used for tracking the customer selection over the internet or what is of their interest when they visit the site again . Online shopping makes use of digital technology for managing the flow of information, products, and payment between consumer, site owners and suppliers. Online shopping can be either B2B (business to business) or B2C (business to consumer)

Shopping cart is one of the important facility provided in online shopping, this lets customer to browse different goods and services and once they select an item to purchase they can place the item in shopping cart, and continue browsing till the final selection. Customers can even remove the items from shopping cart that were selected earlier before they place the final order. It reminds us of shopping basket that we carry in departmental store

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. And there is existing some of the online applications but there in less number of applications

Disadvantages:

There is some disadvantages in existing systems like.

- ☐ Unsatisfied product quality
- ☐ Money transaction issues
- ☐ Refund issues etc..

PROPOSED SYSTEM:

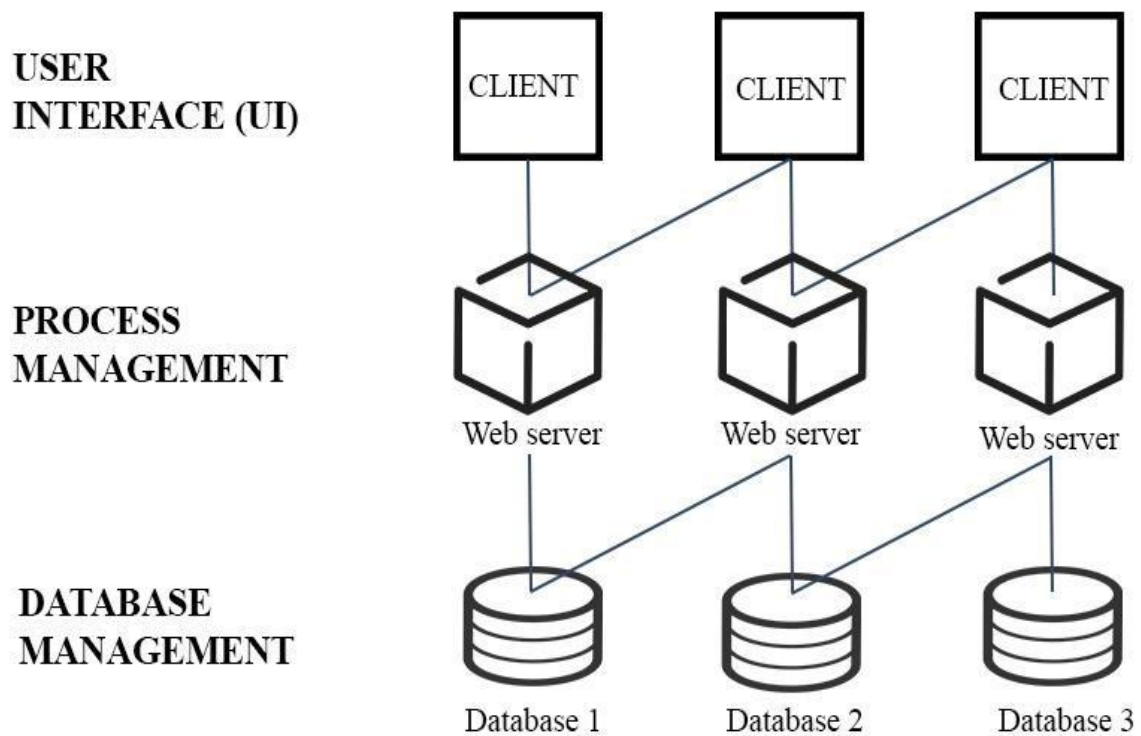
The development of this GET ITEM contains the following activities, which try to develop on-line application by keeping the entire process in the view of database integration approach.

Advantages:

- Secure registration and profile management facilities for Customers.
- Browsing through the e-Malls to see the items that are there in each category of products like Apparel, Kitchen accessories, Bath accessories, Food items etc.
- Creating a Shopping cart so that customer can Shoppe 'n' no. of items and checkout finally with the entire shopping cart
- Customers should be able to mail the Shop about the items they would like to see in the Shop
- Secured mechanism for checking out from the Shop (Credit card verification mechanism)
Updates to customers about the Recent Items in the Shop.

ARCHITECTURE:

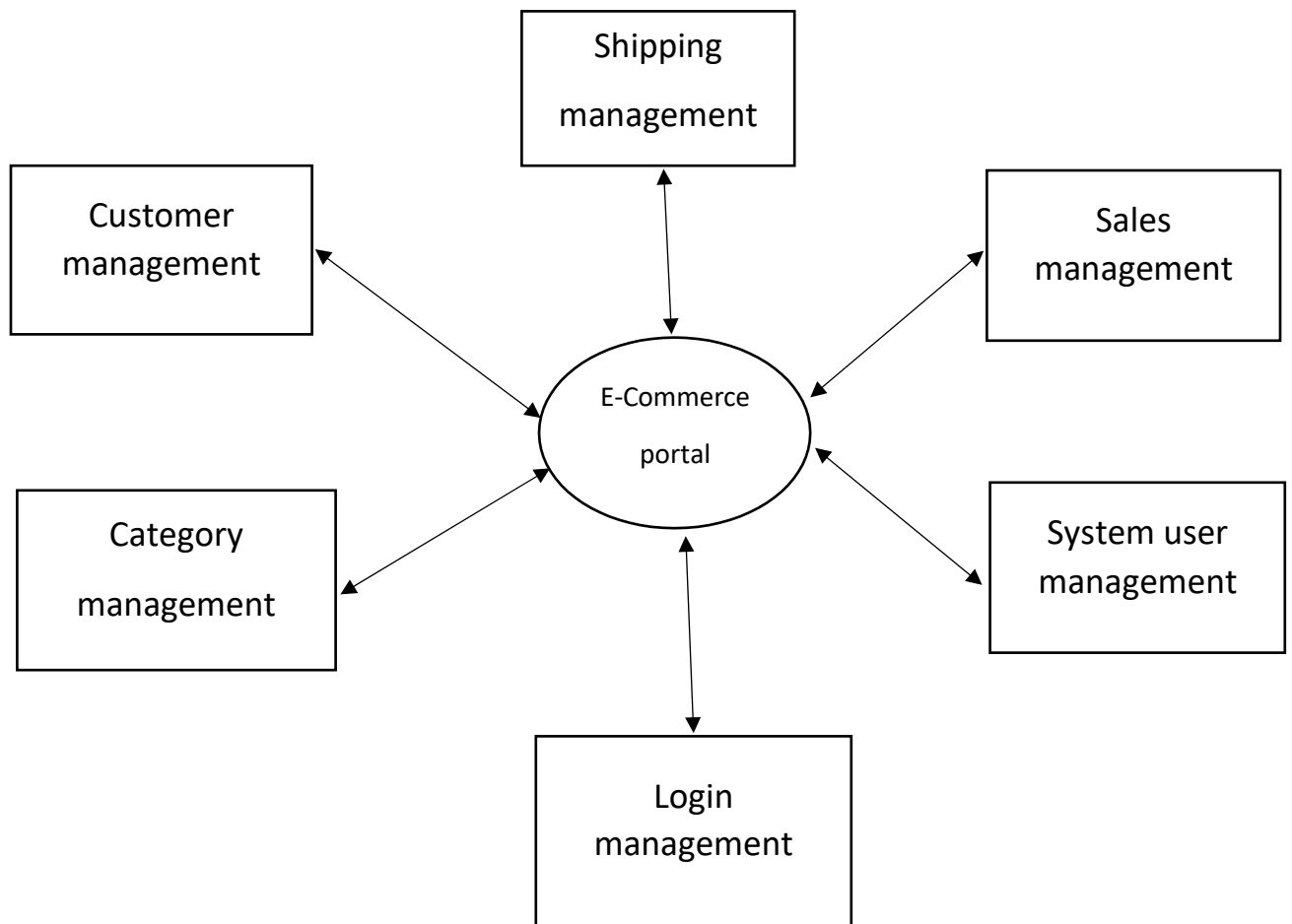
THREE-TIER ECOMMERCE ARCHITECTURE



This is the basic architecture of e commerce service. In three-tier architecture, the user interface and the business application logic, also known as business rules and data storage and access, are developed and maintained as independent modules.

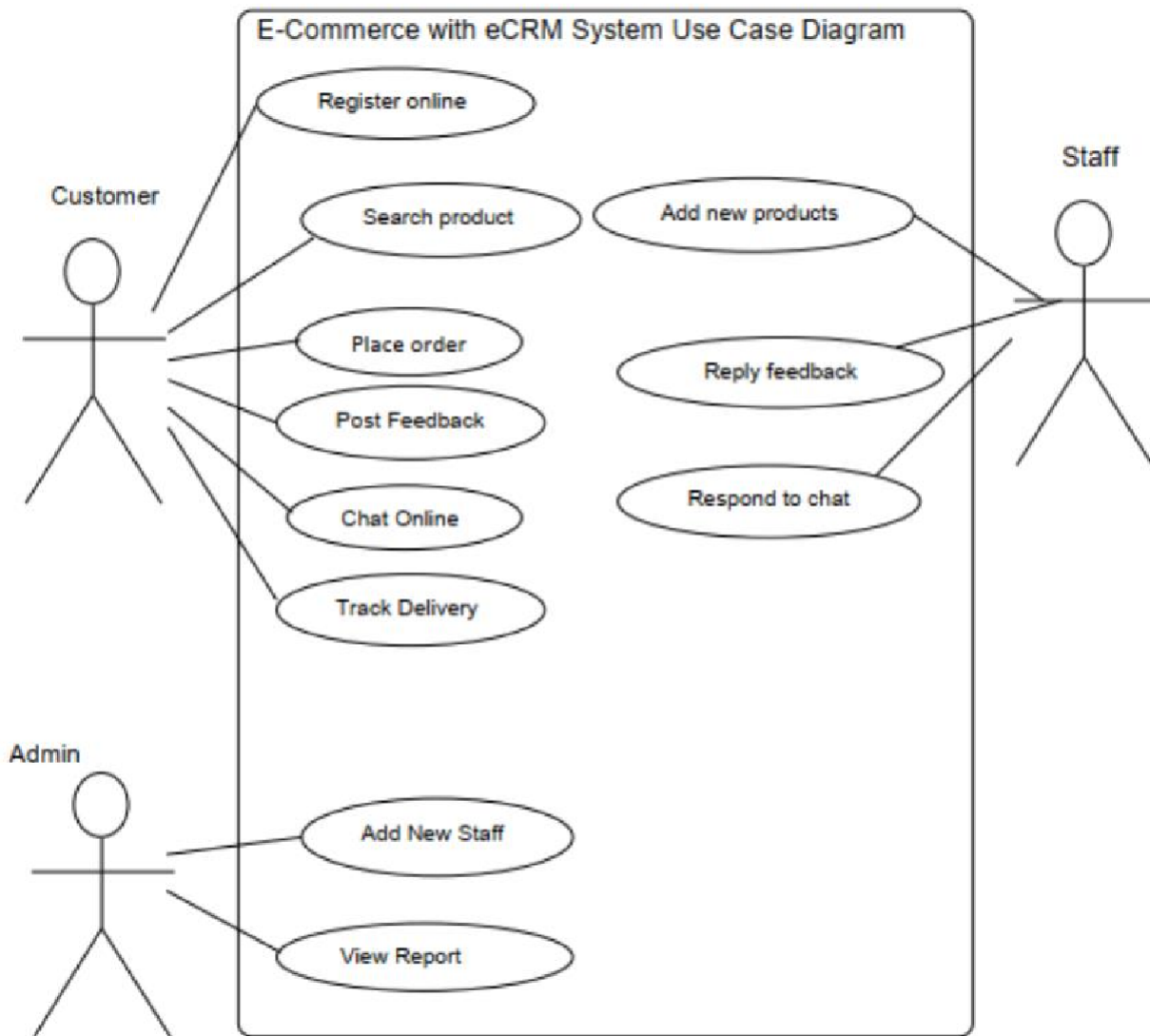
UML DIAGRAMS

DATAFLOW DIARAM:



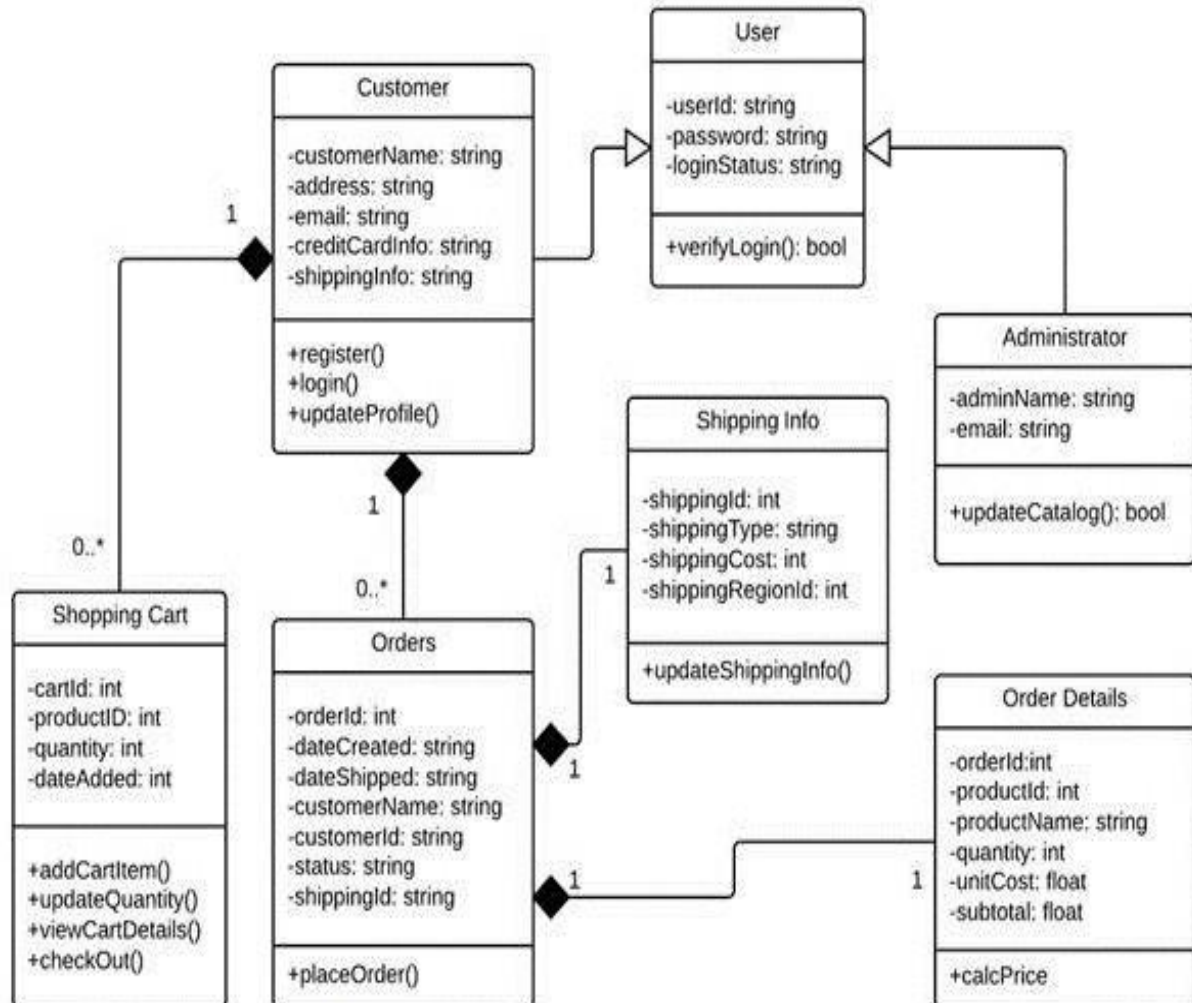
The context level data flow diagram (dfd) is describe the whole system. The (o) level data flow diagram describe the all user module who operate the system. Below data flow diagram of online shopping site shows the two users can operate the system Admin and Member user.

USE CASE DIARAM:



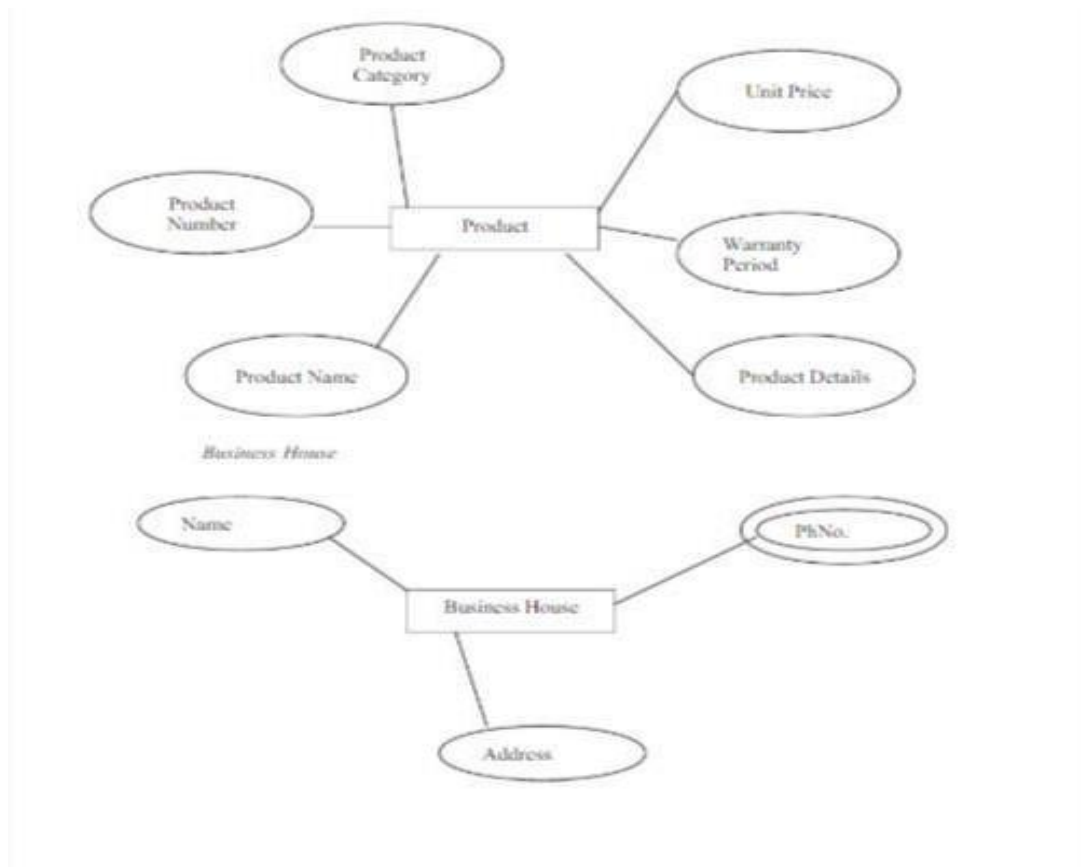
Use Case Diagram for E-Commerce Website uses ellipses that represents the processes and are arranged orderly. The processes in E-Commerce Website are the Customer information Management, Sales and Stocks Monitoring, Product Information Management, and Payment and Delivery Management.

CLASS DIAGRAM:



The Class Diagram for E-commerce Website is a form of structural (UML) diagram that depicts the website's structure. This is designed by displaying the e-commerce website's classes, attributes, methods, and relationships between classes.

PRODUCT ENTITY:



The E–Commerce Website ER Diagram shows how it caters the data used in *e– commerce* management activities w/ the use of database design (ERD). It reveals the relationships between website’s entities in the database. This displays the logical structure of databases and done by identifying **entities**, their **properties**, and the **interactions** between them.

SOFTWARE & HARDWARE REQUIREMENT

Software Requirements:

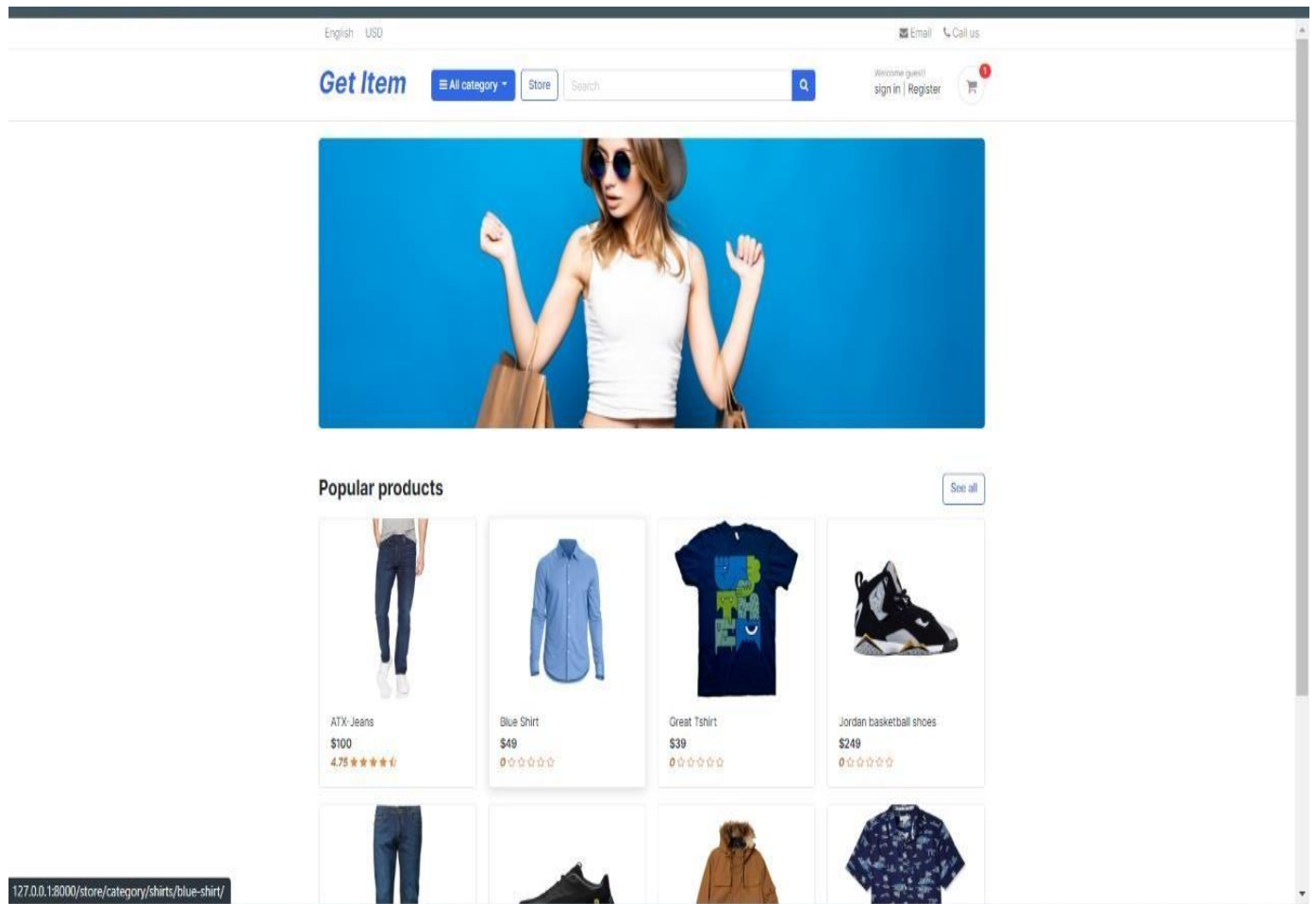
Operating System	:	Windows XP/7 or Linux
Front End	:	HTML, CSS, JS
Back End	:	PYTHON, DJANGO , MYSQL
Browser	:	Internet Explorer 8.0

Hardware Requirements:

Processor	:	Intel Pentium IV
Hard Disk	:	1.44 MB FDD
RAM	:	512MB or more
Processor Speed	:	1.7 GHZ
CD Drive	:	52-X CD RO

SCREENS

Home page:



Login page:

EnglishUSD

EmailCall us

Get Item

All categoryStore

Search

Q

Welcome guest!
sign in | Register

1

Sign in

Email Address

Password

Forgot password?

Login

Don't have account? Sign up

Register page:

EnglishUSD

EmailCall us

Get Item

All categoryStore

Search

Q

Welcome guest!
sign in | Register

1

Sign up

First name

Last name

First Name

Last Name

Phone Number

Email

Enter Phone Number

Enter Email

Create password

Repeat password

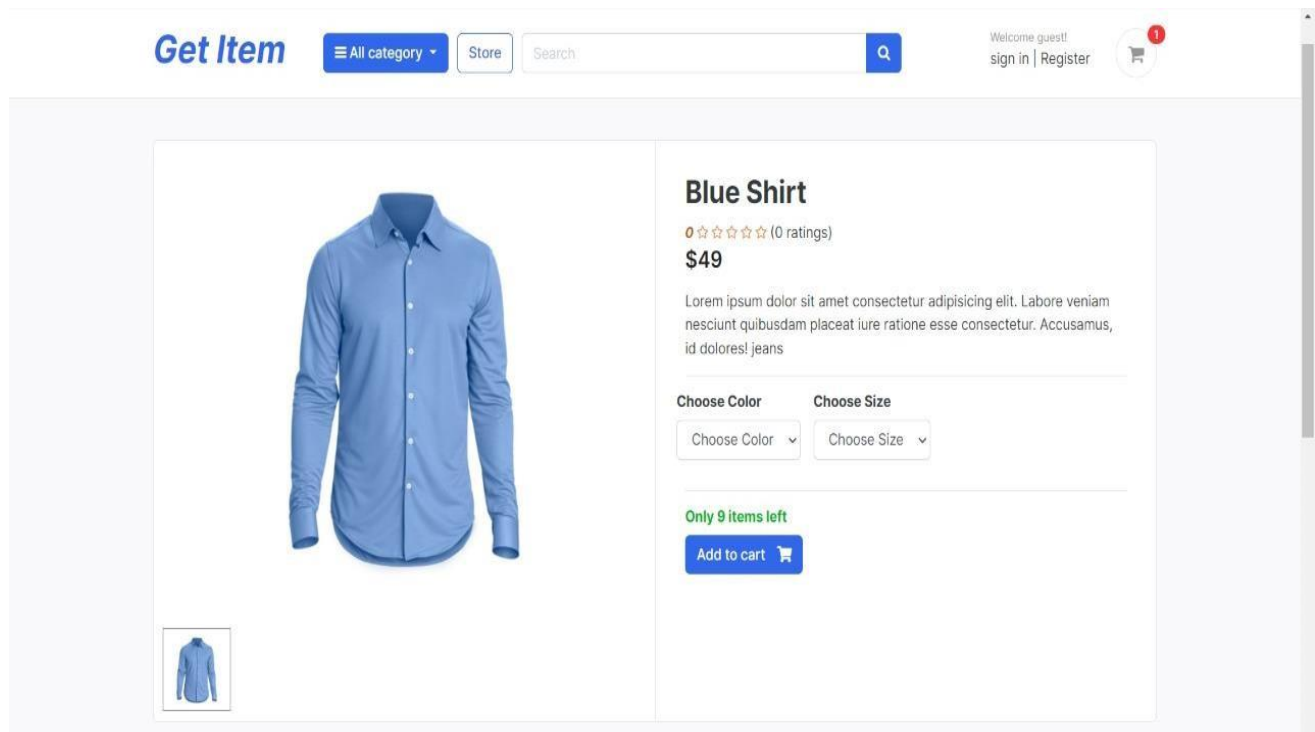
Enter Password

Confirm Password

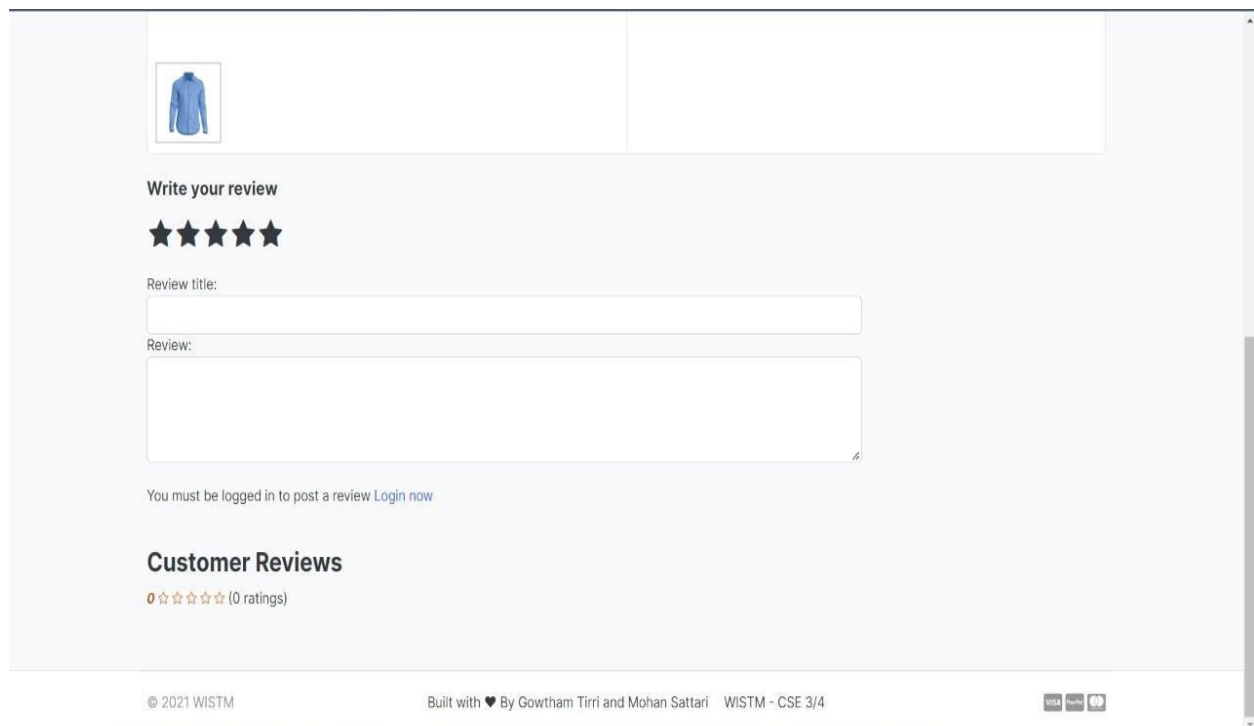
Register

Have an account? Log In

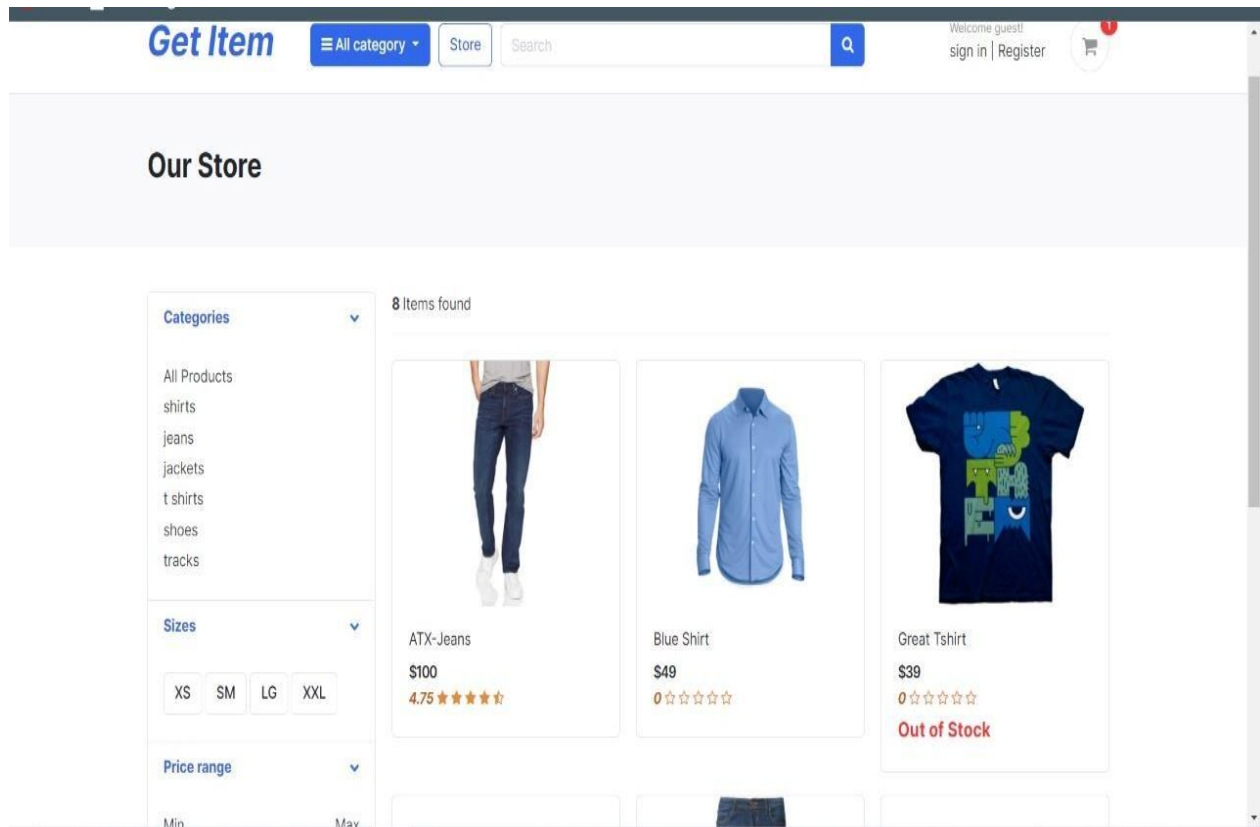
Single product page:



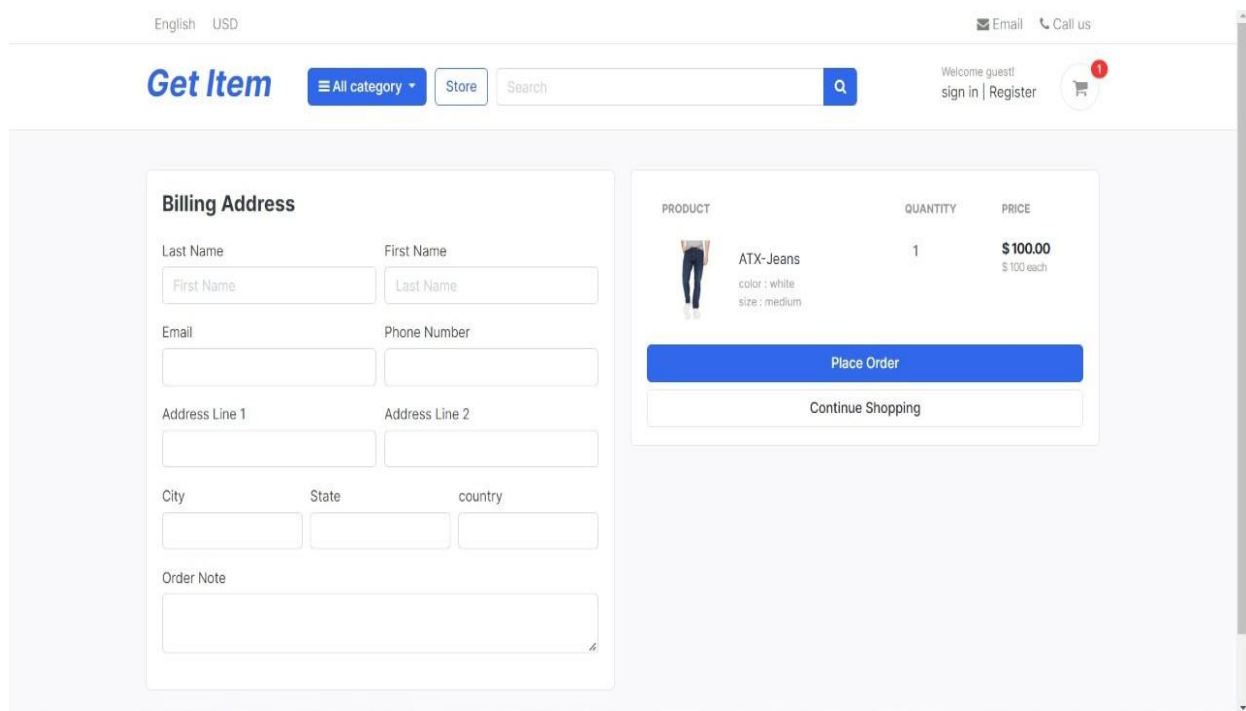
Comment section page:



All products page:



Billing address page:



HOME PAGE CODE :

```
{% extends 'base.html' %}

{% load static %}

{% block content %}

<!-- ===== SECTION MAIN ===== -->

<section class="section-intro padding-y-sm">

<div class="container">

<div class="intro-banner-wrap">

</div>

</div> <!-- container // -->

</section>

<!-- ===== SECTION MAIN END// ===== -->

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

<section class="section-name padding-y-sm">

<div class="container">

<header class="section-heading">

    <a href="{% url 'store' %}" class="btn btn-outline-primary float-right">See all</a>

    <h3 class="section-title">Popular products</h3>

</header><!-- sect-heading -->

<div class="row">

    {% for product in products %}
```

```

<div class="col-md-3">

    <div class="card card-product-grid">

        <a href="{{product.get_url}}" class="img-wrap">  </a>

        <figcaption class="info-wrap">

            <a href="{{product.get_url}}"
class="title">{{product.product_name}}</a>

            <div class="price mt-1">${{product.price}}</div> <!-- price-wrap.// -->

            <div class="rating-stars">

                <span>

                    <i class="average">{{product.averageRating}}</i>

                    <i class="fa fa-star{% if product.averageRating < 0.5 %}-o{% elif
product.averageRating >= 0.5 and product.averageRating < 1 %}-half-o{% endif %}" aria-hidden="true"
></i>

                    <i class="fa fa-star{% if product.averageRating < 1.5 %}-o{% elif
product.averageRating >= 1.5 and product.averageRating < 2 %}-half-o{% endif %}" aria-hidden="true"
></i>

                    <i class="fa fa-star{% if product.averageRating < 2.5 %}-o{% elif
product.averageRating >= 2.5 and product.averageRating < 3 %}-half-o{% endif %}" aria-hidden="true"
></i>

                    <i class="fa fa-star{% if product.averageRating < 3.5 %}-o{% elif
product.averageRating >= 3.5 and product.averageRating < 4 %}-half-o{% endif %}" aria-hidden="true"
></i>

                    <i class="fa fa-star{% if product.averageRating < 4.5 %}-o{% elif
product.averageRating >= 4.5 and product.averageRating < 5 %}-half-o{% endif %}" aria-hidden="true"
></i>

                </span>

            </div>

        </figcaption>

    </div>

</div> <!-- col.// -->

{% endfor %}

</div> <!-- row.// -->

</div><!-- container // -->

```

</section>

<!-- ===== SECTION END// ===== -->

{% endblock content %}

Base page source code:

{% load static %}

<!DOCTYPE HTML>

<html lang="en">

<head>

<meta charset="utf-8">

<meta http-equiv="pragma" content="no-cache" />

<meta http-equiv="cache-control" content="max-age=604800" />

<meta name="viewport" content="width=device-width, initial-scale=1, shrink-to-fit=no">

<title>GreatKart | One of the Biggest Online Shopping Platform</title>

<link href="{% static 'images/favicon.ico' %}" rel="shortcut icon" type="image/x-icon">

<!-- jQuery -->

<script src="{% static 'js/jquery-2.0.0.min.js' %}" type="text/javascript"></script>

<!-- Bootstrap4 files-->

<script src="{% static 'js/bootstrap.bundle.min.js' %}" type="text/javascript"></script>

```
<link href="{% static 'css/bootstrap.css' %}" rel="stylesheet" type="text/css"/>
```

```
<!-- Font awesome 5 -->
```

```
<link href="{% static 'fonts/fontawesome/css/all.min.css' %}" type="text/css" rel="stylesheet">
```

```
<link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/font-awesome/4.7.0/css/font-awesome.min.css">
```

```
<link href="{% static 'css/custom.css' %}" type="text/css" rel="stylesheet">
```

```
<!-- custom style -->
```

```
<link href="{% static 'css/ui.css' %}" rel="stylesheet" type="text/css"/>
```

```
<link href="{% static 'css/responsive.css' %}" rel="stylesheet" media="only screen and (max-width: 1200px)" />
```

```
<!-- custom javascript -->
```

```
<script src="{% static 'js/script.js' %}" type="text/javascript"></script>
```

```
<script type="text/javascript">
```

```
/// some script
```

```
// jquery ready start
```

```
$(document).ready(function() {
```

```
    // jQuery code
```

```
});
```

```
// jquery end
```

```
</script>
```

```
<!-- Include the PayPal JavaScript SDK -->
```

```
<script src="https://www.paypal.com/sdk/js?client-id=AXmSv24F8yww_XBQ0eJzwDHE2ycakfNONEBPEs7KiiDOL6zbMWctmDsQ_7w1XKtuPZgK9qL-eKsRBG7y&currency=USD"></script>
```

</head>

<body>

<header class="section-header">

<nav class="navbar p-md-0 navbar-expand-sm navbar-light border-bottom">

<div class="container">

<button class="navbar-toggler" type="button" data-toggle="collapse" data-target="#navbarTop4" aria-controls="navbarNav" aria-expanded="false" aria-label="Toggle navigation">

</button>

<div class="collapse navbar-collapse" id="navbarTop4">

<ul class="navbar-nav mr-auto">

<li class="nav-item dropdown">

 English

<li class="nav-item dropdown">

 USD

<ul class="navbar-nav">

 <i class="fa fa-envelope"></i> Email

 <i class="fa fa-phone"></i> Call us

 <!-- list-inline // -->

</div> <!-- navbar-collapse .// -->

</div> <!-- container // -->

</nav>

<section class="header-main border-bottom">

<div class="container">

```

<div class="row align-items-center">
    <div class="col-lg-2 col-md-3 col-6">
        <a href="/" class="brand-wrap">
            
        </a> <!-- brand-wrap.// -->
    </div>
    <div class="col-lg col-sm col-md col-6 flex-grow-0">
        <div class="category-wrap dropdown d-inline-block float-right">
            <button type="button" class="btn btn-primary dropdown-toggle" data-
toggle="dropdown">
                <i class="fa fa-bars"></i> All category
            </button>
            <div class="dropdown-menu">
                <a class="dropdown-item" href="{% url 'store' %}">All Products</a>
                {% for category in links %}
                <a class="dropdown-item"
href="{{category.get_link}}">{{category.category_name}}</a>
                {% endfor %}
            </div>
        </div> <!-- category-wrap.// -->
    </div> <!-- col.// -->
    <a href="{% url 'store' %}" class="btn btn-outline-primary">Store</a>
    <div class="col-lg col-md-6 col-sm-12 col">
        <form action="{% url 'search' %}" method="GET" class="search">
            <div class="input-group w-100">
                <input type="text" class="form-control" style="width:60%;"
placeholder="Search" name="query">
            <div class="input-group-append">
                <button class="btn btn-primary" type="submit">
                <i class="fa fa-search"></i>
            </div>
        </form>
    </div>

```

```

        </button>

    </div>

</div>

</form> <!-- search-wrap .end// -->

</div> <!-- col.// -->

<div class="col-lg-3 col-sm-6 col-8 order-2 order-lg-3">

    <div class="d-flex justify-content-end mb-3 mb-lg-0">

        <div class="widget-header">

            {% if user.is_authenticated %}

            <small class="title text-muted">Welcome

{{user.first_name}}</small>

            <div>

                <a href="{% url 'dashboard' %}">Dashboard</a>

<span class="dark-transp"> | </span>

                <a href="{% url 'logout' %}">logout</a> <span

class="dark-transp"> </span>

            </div>

            {% else %}

            <small class="title text-muted">Welcome

guest!</small>

            <div>

                <a href="{% url 'login' %}">sign in</a> <span

class="dark-transp"> | </span>

                <a href="{% url 'register' %}">Register</a>

            </div>

            {% endif %}

        </div>

        <a href="{% url 'cart' %}" class="widget-header pl-3 ml-3">

            <div class="icon icon-sm rounded-circle border"><i

class="fa fa-shopping-cart"></i></div>

```

```

                                <span class="badge badge-pill badge-danger
notify">{{cart_count}}</span>

                                </a>

                                </div> <!-- widgets-wrap.// -->

                                </div> <!-- col.// -->

</div> <!-- row.// -->

                                </div> <!-- container.// -->

</section> <!-- header-main .// -->


</header> <!-- section-header.// -->


{% block content %}

    <!-- Content extends here -->

{% endblock content %}


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

<footer class="section-footer border-top">

    <div class="container">

        <section class="footer-bottom border-top row">

            <div class="col-md-2">

                <p class="text-muted"> &copy; 2021 WISTM</p>

            </div>

            <div class="col-md-8 text-md-center">

                <span class="px-2">Built with ❤ By Gowtham Tirri and Mohan
Sattari</span>

                <span class="px-2">WISTM - CSE 3/4</span>

            </div>

            <div class="col-md-2 text-md-right text-muted">

```



```

        <i class="fab fa-lg fa-cc-visa"></i>

        <i class="fab fa-lg fa-cc-paypal"></i>

        <i class="fab fa-lg fa-cc-mastercard"></i>

    </div>

</section>

</div><!-- //container -->

</footer>

<!-- ===== FOOTER END // ===== -->

<script src="https://ajax.googleapis.com/ajax/libs/jquery/3.5.1/jquery.min.js"></script>

<!-- Product image gallery -->

<script>
    $(document).ready(function(){
        $('.thumb a').click(function(e){
            e.preventDefault();
            $('.mainImage img').attr('src', $(this).attr("href"));
        })
    })
</script>

</body>

</html>

```

Alert page source code:

```

{% if messages %}

<div id="message" class="container">

    {% for message in messages %}

        <div {% if message.tags %} class="alert alert-{{ message.tags }}" {% endif %} role="alert">

            <button type="button" class="close" data-dismiss="alert"><span aria-
hidden="true">&times;</span></button>

```

```
        {{ message }}
    </div>

{% endfor %}

</div>

{% endif %}
```

Cart page source code:

```
{% extends 'base.html' %}

{% load static %}

{% block content %}

<section class="section-content padding-y bg">

    <div class="container">

        <!-- ===== COMPONENT 1 ===== -->

        <div class="row">

            <aside class="col-lg-9">

                <div class="card">

                    {% if cart_items %}

                        <table class="table table-borderless table-shopping-cart">

                            <thead class="text-muted">

                                <tr class="small text-uppercase">

                                    <th scope="col">Product</th>

                                    <th scope="col" width="120">Quantity</th>

                                    <th scope="col" width="120">Price</th>

                                    <th scope="col" class="text-right" width="200"> </th>

                                </tr>
```

```

</thead>

<tbody>

{% for cart_item in cart_items %}

<tr>

<td>

<figure class="itemside align-items-center">

<div class="aside"></div>

<figcaption class="info">

<a href="{{cart_item.product.get_url}}" class="title text-
dark">{{cart_item.product.product_name}}</a>

{% if cart_item.variation %}

{% for item in cart_item.variation.all %}

<p class="text-muted small">{{item.variation_category}} : {{item.variation_value}} <br>
</p>

{% endfor %}

{% endif %}

</figcaption>

</figure>

</td>

<td>

<!-- col.// -->

<div class="col">

<div class="input-group input-spinner">

<div class="input-group-prepend">

<a href="{% url 'remove_cart' cart_item.product.id cart_item.id %}" class="btn btn-
light" type="button" id="button-plus"><i class="fa fa-minus"></i> </a>

</div>

<input type="text" class="form-control" value="{{cart_item.quantity}}">

<div class="input-group-append">

<form action="{% url 'add_cart' cart_item.product.id %}" method="POST" >

```

```

        {% csrf_token %}

        {% for i in cart_item.variation.all %}

            <input type="hidden" name="{{i.variation_category}}"
value="{{i.variation_value}}" >

            {% endfor %}

            <button href="" class="btn btn-light" type="submit" id="button-minus"> <i class="fa
fa-plus"></i> </>

        </form>

    </div>

</div> <!-- input-group.// -->

</div> <!-- col.// -->

</td>

<td>

    <div class="price-wrap">

        <var class="price">$ {{cart_item.sub_total}}.00</var>

        <small class="text-muted"> $ {{cart_item.product.price}} each </small>

    </div> <!-- price-wrap .// -->

</td>

<td class="text-right">

    <a href="{% url 'remove_cart_item' cart_item.product.id cart_item.id %}" class="btn btn-danger">
<i class="fa fa-trash"> </i> Remove</a>

</td>

</tr>

{% endfor %}

{% else %}

<div class="col p-5 text-center" >

    <h4>Your Cart is empty</h4>

    <a href="{% url 'store' %}" class="btn btn-primary">Shop now</a>

</div>

```

```
{% endif %}
```

```
</tbody>
```

```
</table>
```

```
</div> <!-- card.// -->
```

```
</aside> <!-- col.// -->
```

```
<aside class="col-lg-3">
```

```
<div class="card">
```

```
<div class="card-body">
```

```
<dl class="dlist-align">
```

```
<dt>Total price:</dt>
```

```
<dd class="text-right">${{cart_total}}</dd>
```

```
</dl>
```

```
<dl class="dlist-align">
```

```
<dt>Tax:</dt>
```

```
<dd class="text-right"> ${{tax}}</dd>
```

```
</dl>
```

```
<dl class="dlist-align">
```

```
<dt>Grand total:</dt>
```

```
<dd class="text-right text-dark b"><strong>${{grand_total}}</strong></dd>
```

```
</dl>
```

```
<hr>
```

```
<p class="text-center mb-3">
```

```

```

```
</p>
```

```
<a href="{% url 'checkout' %}" class="btn btn-primary btn-block"> Checkout </a>
```

```
<a href="{% url 'store' %}" class="btn btn-light btn-block">Continue Shopping</a>
```

```
</div> <!-- card-body.// -->
```

```
</div> <!-- card.// -->
```

```
</aside> <!-- col.// -->
```

```
</div> <!-- row.// -->
```

```
<!-- ===== COMPONENT 1 END .//  
===== -->
```

```
</div> <!-- container .// -->
```

```
</section>
```

```
{% endblock content %}
```

Checkout page source code:

```
{% extends 'base.html' %}
```

```
{% load static %}
```

```
{% block content %}
```

```
<section class="section-content padding-y bg">
```

```
<div class="container">
```

```
<!-- ===== COMPONENT 1 ===== -->
```

```
<div class="row">
```

```
<aside class="col-lg-6">
```

```
<div class="card">
```

```
<div class="card-body">

    {% include 'alerts.html' %}

    <h4 class="card-title mb-4">Billing Address</h4>
```

```
<!-- ===== checkout form ===== -->
```

```
<form action="{% url 'place_order' %}" method="POST">
```

```
    {% csrf_token %}
```

```
        <div class="form-row">
```

```
            <div class="col form-group">
```

```
                <label >Last Name</label>
```

```
                <input type="text" placeholder="First Name" name="first_name" class="form-control"
required>
```

```
            </div>
```

```
            <div class="col form-group">
```

```
                <label >First Name</label>
```

```
                <input type="text" placeholder="Last Name" name="last_name" class="form-control"
required>
```

```
            </div>
```

```
        </div>
```

```
        <div class="form-row">
```

```
            <div class="col form-group">
```

```
                <label >Email</label>
```

```
                <input type="text" name="email" class="form-control" required>
```

```
            </div>
```

```
            <div class="col form-group">
```

```
                <label >Phone Number</label>
```

```
                <input type="text" name="phone" class="form-control" required>
```

```
            </div>
```

```
        </div>
```

```
        <div class="form-row">
```

```
            <div class="col form-group">
```

```
                <label >Address Line 1</label>
```

```

        <input type="text" name="address_line_1" class="form-control" required>
    </div>

    <div class="col form-group">
        <label >Address Line 2</label>

        <input type="text" name="address_line_2" class="form-control" required>
    </div>
</div>

<div class="form-row">
    <div class="col form-group">
        <label >City</label>

        <input type="text" name="city" class="form-control" required>
    </div>

    <div class="col form-group">
        <label >State</label>

        <input type="text" name="state" class="form-control" required>
    </div>

    <div class="col form-group">
        <label >country</label>

        <input type="country" name="country" class="form-control" required>
    </div>
</div>

<div class="form-row">
    <div class="col form-group">
        <label >Order Note</label>

        <textarea type="text" name="order_note" class="form-control"></textarea>
    </div>
</div>

</div>

</div> <!-- card.// -->

```



```

</aside> <!-- col.// -->

<aside class="col-lg-6">

    <div class="card">
        <div class="card-body">
            <table class="table table-borderless table-shopping-cart">
                <thead class="text-muted">
                    <tr class="small text-uppercase">
                        <th scope="col">Product</th>
                        <th scope="col" width="120">Quantity</th>
                        <th scope="col" width="120">Price</th>
                    </tr>
                </thead>
                <tbody>

                    {% for cart_item in cart_items %}

                    <tr>
                        <td>
                            <figure class="itemside align-items-center">
                                <div class="aside"></div>
                                <figcaption class="info">
                                    <a href="{{cart_item.product.get_url}}" class="title text-
dark">{{cart_item.product.product_name}}</a>
                                    {% if cart_item.variation %}
                                    {% for item in cart_item.variation.all %}
                                    <p class="text-muted small">{{item.variation_category}} :
{{item.variation_value}} <br> </p>
                                    {% endfor %}
                                    {% endif %}
                                </figcaption>
                            </figure>

```

```

        </td>

        <td>

            <!-- col.// -->

            <div class="col">

                {{cart_item.quantity}}

            </div> <!-- col.// -->

        </td>

        <td>

            <div class="price-wrap">

                <var class="price">$ {{cart_item.sub_total}}.00</var>

                <small class="text-muted"> $ {{cart_item.product.price}} each </small>

            </div> <!-- price-wrap .// -->

        </td>

    </tr>

    {% endfor %}

</tbody>

</table>

<button type="submit" name="submit" class="btn btn-primary btn-block"> Place Order
</button>

<a href="{% url 'store' %}" class="btn btn-light btn-block">Continue Shopping</a>

</div> <!-- card-body.// -->

</div> <!-- card.// -->

</aside> <!-- col.// -->

</form>

</section>

{% endblock content %}

```

Product – details page source code:

```
{% extends 'base.html' %}

{% load static %}

{% block content %}

<section class="section-content padding-y bg">
<div class="container">

<!-- ===== COMPONENT 1 ===== -->

<div class="card">
    <div class="row no-gutters">
        <aside class="col-md-6">
            <article class="gallery-wrap">
                <div class="img-big-wrap mainImage">
                    <center></center>
                </div> <!-- img-big-wrap.// -->
            </article> <!-- gallery-wrap .end// -->
            <ul class="thumb">
                <li>
                    <a href="{{product.image.url}}" target="mainImage"></a>
                    {% for item in product_gallery %}
                        <a href="{{item.image.url}}" target="mainImage"></a>
                    {% endfor %}
                </li>
            </ul>
        </aside>
        <main class="col-md-6 border-left">

<form action="{% url 'add_cart' product.id %}" method="POST">
```

```
{% csrf_token %}

<article class="content-body">

<h2 class="title">{{product.product_name}}</h2>

<div class="rating-stars">
    <span>
        <i class="average">{{product.averageRating}}</i>
        <i class="fa fa-star{% if product.averageRating < 0.5 %}-o{% elif product.averageRating
        >= 0.5 and product.averageRating < 1 %}-half-o{% endif %}" aria-hidden="true" ></i>
        <i class="fa fa-star{% if product.averageRating < 1.5 %}-o{% elif product.averageRating
        >= 1.5 and product.averageRating < 2 %}-half-o{% endif %}" aria-hidden="true" ></i>
        <i class="fa fa-star{% if product.averageRating < 2.5 %}-o{% elif product.averageRating
        >= 2.5 and product.averageRating < 3 %}-half-o{% endif %}" aria-hidden="true" ></i>
        <i class="fa fa-star{% if product.averageRating < 3.5 %}-o{% elif product.averageRating
        >= 3.5 and product.averageRating < 4 %}-half-o{% endif %}" aria-hidden="true" ></i>
        <i class="fa fa-star{% if product.averageRating < 4.5 %}-o{% elif product.averageRating
        >= 4.5 and product.averageRating < 5 %}-half-o{% endif %}" aria-hidden="true" ></i>
    <span> {{ product.countReview}} ratings</span>
</span>
</div>

<div class="mb-3">
    <var class="price h4">${{product.price}}</var>
</div>

<p>{{product.description}}</p>

<hr>

<div class="row">
    <div class="item-option-select">
        <h6>Choose Color</h6>
        <select name="color" class="form-control" required>
```

```

        <option value="" disabled selected>Choose Color</option>

        {% for i in product.variation_set.colors %}

            <option value="{{i.variation_value |
lower}}">{{i.variation_value | capfirst}}</option>

            {% endfor %}

        </select>

    </div>

    <div class="item-option-select">

        <h6>Choose Size</h6>

        <select name="size" class="form-control" required>

            <option value="" disabled selected>Choose Size</option>

            {% for i in product.variation_set.sizes %}

                <option value="{{i.variation_value | lower}}">{{i.variation_value
| capfirst}}</option>

                {% endfor %}

            </select>

        </div>

</div> <!-- row.// -->

<hr>

{% if product.stock < 10 and product.stock != 0 %}

    <h6 class="text-success">Only {{product.stock}} items left</h6>

{% endif %}

{% if product.stock <= 0 %}

    <h3 class="text-danger">Out of Stock</h3>

{% else %}

    <button type="submit" class="btn btn-primary "> <span class="text">Add to
cart</span> <i class="fas fa-shopping-cart"></i> </button>

    {% endif %}

</article> <!-- product-info-aside.// -->

</form>

```

```

        </main> <!-- col.// -->

    </div> <!-- row.// -->

</div> <!-- card.// -->

<!-- ===== COMPONENT 1 END .// ===== -
->

<br>

<div class="row">

    <div class="col-md-9">

        {% include 'alerts.html' %}

        <form action="{% url 'submit_review' product.id %}" method="POST">

            {% csrf_token %}

            <h5>Write your review</h5>

            <!-- Rating stars -->

            <div class="rate">

                <input type="radio" id="rating10" name="rating" value="5" required><label
for="rating10" title="5"></label>

                <input type="radio" id="rating9" name="rating" value="4.5" required><label
for="rating9" title="4.5" class="half"></label>

                <input type="radio" id="rating8" name="rating" value="4" required><label
for="rating8" title="4"></label>

                <input type="radio" id="rating7" name="rating" value="3.5" required><label
for="rating7" title="3.5" class="half"></label>

                <input type="radio" id="rating6" name="rating" value="3" required><label
for="rating6" title="3"></label>

                <input type="radio" id="rating5" name="rating" value="2.5" required><label
for="rating5" title="2.5" class="half"></label>

                <input type="radio" id="rating4" name="rating" value="2" required><label
for="rating4" title="2"></label>

                <input type="radio" id="rating3" name="rating" value="1.5" required><label
for="rating3" title="1.5" class="half"></label>

```

☐

☐

</div>

Review title:

<input type="text" class="form-control" name="subject">

Review:

<textarea name="review" rows="4" class="form-control"></textarea>

{% if request.user.is_authenticated %}

{% if ordered_product %}

{% else %}

<p>You must purchase this product to post a review.</p>

{% endif %}

{% else %}

<p>You must be logged in to post a review Login
now</p>

{% endif %}

</form>

<header class="section-heading">

<h3>Customer Reviews </h3>

<div class="rating-stars">

<i class="average">{{product.averageRating}}</i>

<i class="fa fa-star{% if product.averageRating < 0.5 %}-o{% elif
product.averageRating >= 0.5 and product.averageRating < 1 %}-half-o{% endif %}" aria-hidden="true"
></i>

```
        <i class="fa fa-star{% if product.averageRating < 1.5 %}-o{% elif
product.averageRating >= 1.5 and product.averageRating < 2 %}-half-o{% endif %}" aria-hidden="true"
></i>
```

```
        <i class="fa fa-star{% if product.averageRating < 2.5 %}-o{% elif
product.averageRating >= 2.5 and product.averageRating < 3 %}-half-o{% endif %}" aria-hidden="true"
></i>
```

```
        <i class="fa fa-star{% if product.averageRating < 3.5 %}-o{% elif
product.averageRating >= 3.5 and product.averageRating < 4 %}-half-o{% endif %}" aria-hidden="true"
></i>
```

```
        <i class="fa fa-star{% if product.averageRating < 4.5 %}-o{% elif
product.averageRating >= 4.5 and product.averageRating < 5 %}-half-o{% endif %}" aria-hidden="true"
></i>
```

```
        <span> {{{ product.countReview}} ratings}</span>
```

```
    </span>
```

```
  </div>
```

```
</header>
```

```
{% for review in reviews %}
```

```
<article class="box mb-3">
```

```
  <div class="icontext w-100">
```

```
    <div class="text">
```

```
      <span class="date text-muted float-md-right">{{review.updated_at}}
```

```
</span>
```

```
      <h6 class="mb-1">{{review.user.full_name}} </h6>
```

```
      <div class="rating-stars">
```

```
        <i class="fa fa-star{% if review.rating == 0.5 %}-half-o{% elif
review.rating < 1 %}-o{% endif %}" aria-hidden="true" ></i>
```

```
        <i class="fa fa-star{% if review.rating == 1.5 %}-half-o{% elif
review.rating < 2 %}-o{% endif %}" aria-hidden="true" ></i>
```

```
        <i class="fa fa-star{% if review.rating == 2.5 %}-half-o{% elif
review.rating < 3 %}-o{% endif %}" aria-hidden="true" ></i>
```

```
        <i class="fa fa-star{% if review.rating == 3.5 %}-half-o{% elif
review.rating < 4 %}-o{% endif %}" aria-hidden="true" ></i>
```

```
        <i class="fa fa-star{% if review.rating == 4.5 %}-half-o{% elif
review.rating < 5 %}-o{% endif %}" aria-hidden="true" ></i>
```

```
      </div>
```



```

        </div>
    </div> <!-- icontext.// -->

    <div class="mt-3">
        <p>{{review.subject}}</p>
        <p>{{review.review}}</p>
    </div>
</article>
{% endfor %}
```

```

    </div> <!-- col.// -->
</div> <!-- row.// -->
```

```

</div> <!-- container .// -->
</section>
{% endblock %}
```

Store page source code:

```
{% extends 'base.html' %}
```

```
{% block content %}
```

```
<!-- ===== SECTION PAGETOP ===== -->
```

```

<section class="section-pagetop bg">

  <div class="container">

    {% if 'search' in request.path %}

      <h2 class="title-page">Seach Results </h2>

    {% else %}

      <h2 class="title-page">Our Store</h2>

    {% endif %}

  </div> <!-- container // -->

</section>

<!-- ===== SECTION INTRO END// ===== -->

<!-- ===== SECTION CONTENT ===== -->

<section class="section-content padding-y">

  <div class="container">

    <div class="row">

      <aside class="col-md-3">

        <div class="card">

          <article class="filter-group">

            <header class="card-header">

              <a href="#" data-toggle="collapse" data-target="#collapse_1" aria-expanded="true" class="">

                <i class="icon-control fa fa-chevron-down"></i>

                <h6 class="title">Categories</h6>

              </a>

            </header>

            <div class="filter-content collapse show" id="collapse_1">

              <div class="card-body">

                <ul class="list-menu">

                  <li><a href="{% url 'store' %}">All Products </a></li>

```

```

    {% for category in links %}

    <li><a href="{{category.get_link}}">{{category.category_name}} </a></li>

    {% endfor %}

</ul>


</div> <!-- card-body.// -->

</div>

</article> <!-- filter-group .// -->

<article class="filter-group">

    <header class="card-header">

        <a href="#" data-toggle="collapse" data-target="#collapse_4" aria-expanded="true" class="">

            <i class="icon-control fa fa-chevron-down"></i>

            <h6 class="title">Sizes </h6>

        </a>

    </header>

    <div class="filter-content collapse show" id="collapse_4" >

        <div class="card-body">

            <label class="checkbox-btn">

                <input type="checkbox">

                <span class="btn btn-light"> XS </span>

            </label>


            <label class="checkbox-btn">

                <input type="checkbox">

                <span class="btn btn-light"> SM </span>

            </label>


            <label class="checkbox-btn">

                <input type="checkbox">

                <span class="btn btn-light"> LG </span>

```

</label>

<label class="checkbox-btn">

<input type="checkbox">

 XXL

</label>

</div><!-- card-body.// -->

</div>

</article> <!-- filter-group .// -->

<article class="filter-group">

<header class="card-header">

<i class="icon-control fa fa-chevron-down"></i>

<h6 class="title">Price range </h6>

</header>

<div class="filter-content collapse show" id="collapse_3" >

<div class="card-body">

<div class="form-row">

<div class="form-group col-md-6">

<label>Min</label>

<!-- <input class="form-control" placeholder="\$0" type="number"> -->

<select class="mr-2 form-control">

<option value="0">\$0</option>

<option value="50">\$50</option>

<option value="100">\$100</option>

<option value="150">\$150</option>

<option value="200">\$200</option>

<option value="500">\$500</option>

```

        <option value="1000">$1000</option>
    </select>
</div>
<div class="form-group text-right col-md-6">
    <label>Max</label>
    <select class="mr-2 form-control">
        <option value="50">$50</option>
        <option value="100">$100</option>
        <option value="150">$150</option>
        <option value="200">$200</option>
        <option value="500">$500</option>
        <option value="1000">$1000</option>
        <option value="2000">$2000+</option>
    </select>
</div>
</div> <!-- form-row.// -->
    <button class="btn btn-block btn-primary">Apply</button>
</div><!-- card-body.// -->
</div>
</article> <!-- filter-group.// -->

</div> <!-- card.// -->

</aside> <!-- col.// -->
<main class="col-md-9">

<header class="border-bottom mb-4 pb-3">
    <div class="form-inline">
        <span class="mr-md-auto"> <b>{{products_count}} </b> Items found </span>

    </div>

```

```
</header><!-- sect-heading -->
```

```
<div class="row">
```

```
{% if not products %}
```

```
  <div class="col text-center">
```

```
    
```

```
    <h4 class="text-secondary">Looks like you found nothing</h4>
```

```
  </div>
```

```
{% else %}
```

```
{% for product in products %}
```

```
<div class="col-md-4">
```

```
  <figure class="card card-product-grid">
```

```
    <div class="img-wrap">
```

```
      <a href="{{product.get_url}}"></a>
```

```
    </div>
```

```
    <figcaption class="info-wrap">
```

```
      <div class="fix-height">
```

```
        <a href="{{product.get_url}}" class="title">{{product.product_name}}</a>
```

```
        <div class="price-wrap mt-2">
```

```
          <span class="price">${{product.price}}</span>
```

```
        </div>
```

```
      <div class="rating-stars">
```

```
        <span>
```

```
          <i class="average">{{product.averageRating}}</i>
```

```
          <i class="fa fa-star{% if product.averageRating < 0.5 %}-o{% elif product.averageRating  
>= 0.5 and product.averageRating < 1 %}-half-o{% endif %}" aria-hidden="true" ></i>
```

```
          <i class="fa fa-star{% if product.averageRating < 1.5 %}-o{% elif product.averageRating  
>= 1.5 and product.averageRating < 2 %}-half-o{% endif %}" aria-hidden="true" ></i>
```

```
        <i class="fa fa-star{% if product.averageRating < 2.5 %}-o{% elif product.averageRating
>= 2.5 and product.averageRating < 3 %}-half-o{% endif %}" aria-hidden="true" ></i>
```

```
        <i class="fa fa-star{% if product.averageRating < 3.5 %}-o{% elif product.averageRating
>= 3.5 and product.averageRating < 4 %}-half-o{% endif %}" aria-hidden="true" ></i>
```

```
        <i class="fa fa-star{% if product.averageRating < 4.5 %}-o{% elif product.averageRating
>= 4.5 and product.averageRating < 5 %}-half-o{% endif %}" aria-hidden="true" ></i>
```

```
    </span>
```

```
  </div>
```

```
</div>
```

```
{% if product.stock <= 0 %}
```

```
    <h5 class="text-danger">Out of Stock</h5>
```

```
{% endif %}
```

```
</figcaption>
```

```
</figure>
```

```
</div> <!-- col.// -->
```

```
{% endfor %}
```

```
{% endif %}
```

```
</div> <!-- row end.// -->
```

```
<nav class="mt-4" aria-label="Page navigation sample">
```

```
{% if products.has_other_pages %}
```

```
  <ul class="pagination">
```

```
    {% if products.has_previous %}
```

```
      <li class="page-item"><a class="page-link"
href="?page={{products.previous_page_number}}">Previous</a></li>
```

```
    {% else %}
```

```
      <li class="page-item disabled"><a class="page-link" href="#">Previous</a></li>
```

```
    {% endif %}
```

```

{% for i in products.paginator.page_range %}

    {% if products.number == i %}

        <li class="page-item active"><a class="page-link" href="#">{{i}}</a></li>

    {% else %}

        <li class="page-item"><a class="page-link" href="?page={{i}}">{{i}}</a></li>

    {% endif %}

{% endfor %}

{% if products.has_next %}

    <li class="page-item"><a class="page-link"
href="?page={{products.next_page_number}}">Next</a></li>

    {% else %}

        <li class="page-item disabled"><a class="page-link" href="#">Next</a></li>

    {% endif %}

</ul>

{% endif %}

</nav>

</main> <!-- col.// -->

</div>

</div> <!-- container .// -->

</section>

<!-- ===== SECTION CONTENT END// ===== -->

{% endblock content %}

```

Order pages:

Order_complete page:

<div class="invoice-from">

```

        <ul class="list-unstyled text-right">
            <li><strong>Invoiced To</strong></li>
            <li>{{order.full_name}}</li>
            <li>{{order.full_address}}</li>
            <li>{{order.state}}, {{order.city}}</li>
            <li>{{order.country}}</li>
        </ul>
    </div>
</div>
<!-- col-lg-6 end here -->
<div class="col-lg-12">
    <!-- col-lg-12 start here -->
    <div class="invoice-details mt25">
        <div class="well">
            <ul class="list-unstyled mb0">
                <li><strong>Order</strong> #{{order.order_number}}</li>
                <li><strong>Transaction</strong> #{{transID}}</li>
                <li><strong>Order Date:</strong> {{order.created_at}}</li>
                <li><strong>Status:</strong> {{payment.status}} </li>
            </ul>
        </div>
    </div>
</div>

<div class="invoice-items">
    <div class="table-responsive" style="overflow: hidden; outline: none;" tabindex="0">
        <table class="table table-bordered">
            <thead>
                <tr>
                    <th class="per70 text-center">Description</th>
                    <th class="per5 text-center">Qty</th>
                    <th class="per25 text-center">Total</th>

```

```

        </tr>
    </thead>
    <tbody>
        {% for item in order_products %}
        <tr>
            <td>{{item.product.product_name}}
                <p class="text-muted small">
                    {% if item.variations.all %}
                    {% for i in item.variations.all %}
                        {{item.variation_category}}
                        {{i.variation_value}}
                    {% endfor %}
                    {% endif %}
                </p>
            </td>
            <td class="text-center">{{item.quantity}}</td>
            <td class="text-center">${{item.product.price}} USD</td>
        </tr>
        {% endfor %}
    </tbody>
    <tfoot>
        <tr>
            <th colspan="2" class="text-right">Sub Total:</th>
            <th class="text-center">${{sub_total}} USD</th>
        </tr>
        <tr>
            <th colspan="2" class="text-right">Tax:</th>
            <th class="text-center">${{order.tax}}USD</th>
        </tr>
        <tr>

```

```

        <th colspan="2" class="text-right">Grand Total:</th>

        <th class="text-center">${{order.order_total}}USD</th>

    </tr>

</tfoot>

</table>

</div>

</div>

<div class="invoice-footer mt25">

    <p class="text-center">Thank you for shopping with us!</p>

</div>

</div>

<!-- col-lg-12 end here -->

</div>

<!-- End .row -->

</div>

</div>

<!-- End .panel -->

</div>

<!-- col-lg-12 end here -->

</div>

</div>

{% endblock content %}

```

Order_details pages:

```

{% extends 'base.html' %}

{% load static %}

{% block content %}

```

```
<div class="container" style="margin: 0 auto;width: 50%;padding: 50px;background: #f1f1f1;margin-top: 50px;margin-bottom: 50px;">
```

```
<div class="row invoice row-printable">
```

```
<div class="col-md-12">
```

```
<!-- col-lg-12 start here -->
```

```
<div class="panel panel-default plain" id="dash_0">
```

```
<!-- Start .panel -->
```

```
<div class="panel-body p30">
```

```
<div class="row">
```

```
<!-- Start .row -->
```

```
<div class="col-lg-6">
```

```
<!-- col-lg-6 start here -->
```

```
<div class="invoice-logo"></div>
```

```
</div>
```

```
<!-- col-lg-6 end here -->
```

```
<div class="col-lg-6">
```

```
<!-- col-lg-6 start here -->
```

```
<div class="invoice-from">
```

```
<ul class="list-unstyled text-right">
```

```
<li><strong>Invoiced To</strong></li>
```

```
<li>{{order.full_name}}</li>
```

```
<li>{{order.full_address}}</li>
```

```
<li>{{order.state}}, {{order.city}}</li>
```

```
<li>{{order.country}}</li>
```

```
</ul>
```

```
</div>
```

```
</div>
```

```
<!-- col-lg-6 end here -->
```

```
<div class="col-lg-12">
```

```
<!-- col-lg-12 start here -->
```

```
<div class="invoice-details mt25">
```

```

<div class="well">
  <ul class="list-unstyled mb0">
    <li><strong>Order</strong> #{{order.order_number}}</li>
    <li><strong>Transaction</strong> #{{order.payment.payment_id}}</li>
    <li><strong>Order Date:</strong> {{order.created_at}}</li>
    <li><strong>Status:</strong> {{order.payment.status}}</li>
  </ul>
</div>
</div>

<div class="invoice-items">
  <div class="table-responsive" style="overflow: hidden; outline: none;" tabindex="0">
    <table class="table table-bordered">
      <thead>
        <tr>
          <th class="per70 text-center">Description</th>
          <th class="per5 text-center">Qty</th>
          <th class="per25 text-center">Total</th>
        </tr>
      </thead>
      <tbody>
        {% for item in order_products %}
          <tr>
            <td>{{item.product.product_name}}
              <p class="text-muted small">
                {% if item.variations.all %}
                {% for i in item.variations.all %}
                  {{item.variation_category}}
                  {{i.variation_value}}
                {% endfor %}
              {% endif %}
            </td>
            <td>{{item.quantity}}</td>
            <td>{{item.total}}</td>
          </tr>
        {% endfor %}
      </tbody>
    </table>
  </div>
</div>

```

```

        </p>
    </td>

    <td class="text-center">{{item.quantity}}</td>

    <td class="text-center">${{item.product.price}} USD</td>

</tr>

{% endfor %}

</tbody>
<tfoot>

<tr>

    <th colspan="2" class="text-right">Sub Total:</th>

    <th class="text-center">${{sub_total}} USD</th>

</tr>

<tr>

    <th colspan="2" class="text-right">Tax:</th>

    <th class="text-center">${{order.tax}}USD</th>

</tr>

<tr>

    <th colspan="2" class="text-right">Grand Total:</th>

    <th class="text-center">${{order.order_total}}USD</th>

</tr>
</tfoot>
</table>

</div>

</div>

<div class="invoice-footer mt25">

    <p class="text-center">Thank you for shopping with us!</p>

</div>

</div>

<!-- col-lg-12 end here -->

```

```
        </div>

        <!-- End .row -->

    </div>

</div>

<!-- End .panel -->

</div>

<!-- col-lg-12 end here -->

</div>

<div class="container text-center m-2">

    <a href="{% url 'my_orders' %}" class="btn btn-primary">Back to Orders</a>

</div>

</div>

{% endblock content %}
```

Order_recived_email page:

```
{% autoescape off %}
```

Hii {{user.full_name}}

Your Order has been received.

Order number : {{order.order_number}}

```
{% endautoescape %}
```


Payments page:

```
{% extends 'base.html' %}
```

```
{% load static %}
```

```
{% block content %}
```

```
<section class="section-content padding-y bg">
```

```
<div class="container">
```

```
<!-- ===== COMPONENT 1 ===== -->
```

```
<h5 class="text-center text-primary mb-2">Review your order and make payment</h5>
```

```
<div class="row">
```

```
<aside class="col-lg-8">
```

```
<div class="card">
```

```
<h5 class="card-header">Billing Address</h5>
```

```
<div class="card-body">
```

```
<p class="card-text">{{order.full_name}}</p>
```

```
<p class="card-text">{{order.full_address}}</p>
```

```
<p class="card-text">{{order.city}} {{order.state}}</p>
```

```
<p class="card-text">{{order.country}}</p>
```

```
{% if order.order_note %}
```

```
<p class="card-text"> Note : {{order.order_note}} </p>
```

```
{% endif %}
```

```
</div>
```

```
</div> <!-- card.// -->
```

```
<div class="card">
```

```
<h5 class="card-header">Payment Method</h5>
```

```
<div class="card-body">
```

```
<p class="card-text">Paypal</p>
```

```
</div>
```

```
</div> <!-- card.// -->
```

```

<div class="card">

  <h5 class="card-header">Review Products</h5>

  <div class="card-body">

    <table class="table table-borderless table-shopping-cart">

      <thead class="text-muted">

        <tr class="small text-uppercase">

          <th scope="col">Product</th>

          <th scope="col" width="120">Quantity</th>

          <th scope="col" width="120">Price</th>

        </tr>

      </thead>

      <tbody>

        {% for cart_item in cart_items %}

        <tr>

          <td>

            <figure class="itemside align-items-center">

              <div class="aside"></div>

              <figcaption class="info">

                <a href="{{cart_item.product.get_url}}" class="title text-
dark">{{cart_item.product.product_name}}</a>

                {% if cart_item.variation %}

                {% for item in cart_item.variation.all %}

                <p class="text-muted small">{{item.variation_category}} : {{item.variation_value}}

<br> </p>

                {% endfor %}

                {% endif %}

              </figcaption>

            </figure>

          </td>

          <td>

            <!-- col.// -->

```

```

        <div class="col">
            {{cart_item.quantity}}
        </div> <!-- col.// -->
    </td>
    <td>
        <div class="price-wrap">
            <var class="price">$ {{cart_item.sub_total}}.00</var>
            <small class="text-muted"> $ {{cart_item.product.price}} each </small>
        </div> <!-- price-wrap .// -->
    </td>

</tr>
{% endfor %}

</tbody>
</table>
</div>
</div> <!-- card.// -->

</aside> <!-- col.// -->
<aside class="col-lg-4">

    <div class="card">
        <div class="card-body">
            <dl class="dlist-align">
                <dt>Total price:</dt>
                <dd class="text-right">${{cart_total}}</dd>
            </dl>
            <dl class="dlist-align">
                <dt>Tax:</dt>

```

```

        <dd class="text-right"> ${{tax}}</dd>

    </dl>

    <dl class="dlist-align">

        <dt>Grand total:</dt>

        <dd class="text-right text-dark b"><strong>${{grand_total}}</strong></dd>

    </dl>

    <hr>

    <p class="text-center mb-3">

    </p>

    <!-- -->

    <div id="paypal-button-container">

        <!-- Paypal button -->

    </div>

</div> <!-- card-body.// -->

</div> <!-- card.// -->

</aside> <!-- col.// -->


</div> <!-- row.// -->

<!-- ===== COMPONENT 1 END .//
===== -->


</div> <!-- container .// -->

</section>


<script>

function getCookie(name) {

    let cookieValue = null;

    if (document.cookie && document.cookie !== "") {

```

```

const cookies = document.cookie.split(';');
for (let i = 0; i < cookies.length; i++) {
    const cookie = cookies[i].trim();
    // Does this cookie string begin with the name we want?
    if (cookie.substring(0, name.length + 1) === (name + '=')) {
        cookieValue = decodeURIComponent(cookie.substring(name.length + 1));
        break;
    }
}
}
return cookieValue;
}

```

```

const csrftoken = getCookie('csrftoken');
var amount = "{{grand_total}}"
var orderID = "{{order.order_number}}"
let url = "{% url 'payments' %}"
let payment_method = 'PayPal'
let redirect_url = '{% url "order_complete" %}'

```

```

// Render the PayPal button into #paypal-button-container

```

```

paypal.Buttons({

```

```

    // Set up the transaction

```

```

    createOrder: function(data, actions) {

```

```

        return actions.order.create({

```

```

            purchase_units: [{

```

```

                amount: {

```

```

                    value: amount

```

```

                }

```

```

            }]

```

```

    });
},

// Finalize the transaction
onApprove: function(data, actions) {
    return actions.order.capture().then(function(details) {
        // Show a success message to the buyer
        alert('Transaction completed by ' + details.payer.name.given_name + '!');

        sendData();

        function sendData(){
            fetch(url, {
                method:'POST',
                headers:{
                    'Content-type':'application/json',
                    "X-CSRFToken":csrftoken,
                },
                body:JSON.stringify({
                    orderID : orderID,
                    transID : details.id,
                    payment_method:payment_method,
                    status : details.status,
                }),
            }).then(response => response.json())
                .then((data => {
                    window.location.href =
                    redirect_url+'?order_number='+data.order_number+'&payment_id='+data.transID
                }))
        } // send data func close

    });
} // on approve close

```

```

    }).render('#paypal-button-container');

</script>

{% endblock content %}

```

Includes pages:

Dashboard_sidebar page:

```

<aside class="col-md-3">

    <!-- SIDEBAR -->

    <ul class="list-group">

        <a class="list-group-item {% if '/accounts/' == request.path %}active{% endif %}" href="{% url
'dashboard' %}">Dashboard </a>

        <a class="list-group-item {% if '/my_orders/' in request.path %}active{% endif %}" href="{% url
'my_orders' %}"> My Orders </a>

        <a class="list-group-item {% if '/edit_profile/' in request.path %}active{% endif %}" href="{% url
'edit_profile' %}"> Edit profile </a>

        <a class="list-group-item {% if '/change_password/' in request.path %}active{% endif %}" href="{%
url 'change_password' %}">Change Password </a>

    </ul>

    <br>

    <a class="btn btn-light btn-block" href="{% url 'logout' %}"> <i class="fa fa-power-off"></i> <span
class="text">Log out</span> </a>

    <!-- SIDEBAR .//END -->

</aside>

```

Account pages:

Change password page:

```

{% extends 'base.html' %}

{% block content %}

```

```

<section class="section-content padding-y bg">

  <div class="container">

    <div class="row">

      {% include 'includes/dashboard_sidebar.html' %}

      <main class="col-md-9">

        <article class="card">

          <header class="card-header" style="background: rebeccapurple; color:white">

            <strong class="d-inline-block mr-3">Change your Password</strong>

          </header>

          <div class="card-body">

            {% include 'alerts.html' %}

            <div class="row">

              <div class="col-md-12">

                <form action="{% url 'change_password' %}" method="POST">

                  {% csrf_token %}

                  <div class="form-group">

                    <label for="">Current Password</label>

                    <input type="password" name="current_password" placeholder="Current Password"
class="form-control" id="">

                  </div>

                  <div class="form-group">

                    <label for="">Create New Password</label>

                    <input type="password" name="new_password" placeholder="Create New Password"
class="form-control" id="">

                  </div>

                  <div class="form-group">

                    <label for="">Confirm New Password</label>

                    <input type="password" name="confirm_password" placeholder="Confirm New
Password" class="form-control" id="">

                  </div>

                </div>

              </div>

            </div>

          </div>

        </article>

      </main>

    </div>

  </div>

</section>

```



```

        <input type="submit" value="Save" class="btn btn-primary">
    </form>

</div>

</div> <!-- row.// -->

</div> <!-- card-body .// -->

</article> <!-- order-group.// -->

</main>

</div> <!-- row.// -->

</div>

</section>

{% endblock content %}

```

Dashboard page :

```

{% extends 'base.html' %}

{% block content %}

<section class="section-content padding-y bg">
    {% include 'alerts.html' %}
    <div class="container">
        <div class="row">
            {% include 'includes/dashboard_sidebar.html' %}
            <main class="col-md-9">
                <article class="card">
                    <header class="card-header" style="background: rebeccapurple; color:white">
                        <strong class="d-inline-block mr-3">{{request.user.full_name | capfirst }}</strong>
                    </header>
                    <div class="card-body">
                        <div class="row">

```

```

<div class="col-md-6">

  <div class="col card rounded" style="background:rgb(250, 211, 255);">

    <div class="card-body " style="text-align: center; ">

      <h5 class="card-title">Total Orders</h5>

      <h4>{{order_count}}</h4>

      <a class="fa fa-eye" href="{% url 'my_orders' %}">View Orders</a>

    </div>

  </div>

</div>

<div class="col-md-6">

  <div class="col card rounded" style="background:rgb(211, 222, 255);">

    <div class="card-body" style="text-align: center;">

      <p class="mb-0">{{user.email}}</p>

      <p>{{user.phone_number}}</p>

    </div>

  </div>

</div>

</div> <!-- row.// -->

</div> <!-- card-body.// -->

</article> <!-- order-group.// -->

</main>

</div> <!-- row.// -->

</div>

</section>

{% endblock content %}

```

Edit profile page:

```
{% extends 'base.html' %}
```

{% block content %}

```
<section class="section-content padding-y bg">
```

```
{% include 'alerts.html' %}
```

<div class="container">

```
<div class="row">
```

```
{% include 'includes/dashboard_sidebar.html' %}
```

```
<main class="col-md-9">
```

<article class="card">

```
<header class="card-header" style="background: rebeccapurple; color:white">
```

[Edit your profile](#)

</header>

<div class="card-body">

<div >

```

```

</div>

```
<div class="row">
```

<div class="col-md-12">

```
<form action="{% url 'edit_profile' %}" method="POST" enctype="multipart/form-data">
```

```
{% csrf_token %}
```

<div class="form-row">

<div class="form-group col-md-6">

```
<label for="">First Name</label>
```

```
{{user_form.first_name}}
```

</div>

<div class="form-group col-md-6">

<label for="">First Name</label>

{{user_form.last_name}}

</div>

</div>

<div class="form-row">

<div class="form-group col-md-6">

<label for="">Phone Number</label>

{{user_form.phone_number}}

</div>

<div class="form-group col-md-6">

<label for="">Profile Pic</label>

{{profile_form.profile_pic}}

</div>

</div>

<div class="form-group ">

<label for="">Address Line 1</label>

{{profile_form.address_line1}}

</div>

<div class="form-group ">

<label for="">Address Line 2</label>

{{profile_form.address_line2}}

</div>

<div class="form-row">

<div class="form-group col-md-4">

<label for="">City</label>

{{profile_form.city}}

</div>

<div class="form-group col-md-4">

```

        <label for="">State</label>

        {{profile_form.state}}

    </div>

    <div class="form-group col-md-4">

        <label for="">Country</label>

        {{profile_form.country}}

    </div>

</div>

    <input type="submit" value="Save" class="btn btn-primary">

</form>

</div>

</div> <!-- row.// -->

</div> <!-- card-body .// -->

</article> <!-- order-group.// -->

</main>

</div> <!-- row.// -->

</div>

</section>

{% endblock content %}

Forgot pass page:

{% extends 'base.html' %}

{% load static %}

{% block content %}

<section class="section-conten padding-y" style="min-height:84vh">

    <!-- ===== COMPONENT LOGIN =====>

-->

    <div class="card mx-auto" style="max-width: 500px; margin-top:100px;">

```

```

<div class="card-body">
    {% include 'alerts.html' %}
    <h4 class="card-title mb-4">Forgot Password</h4>
    <form action="{% url 'forgotPassword' %}" method="POST">
        {% csrf_token %}
        <div class="form-group">
            <input class="form-control" name="email" placeholder="Email Address" >
        </div> <!-- form-group -->

        <div class="form-group">
            <a href="{% url 'login' %}">Got password? login</a>
        </div> <!-- form-group -->

        <div class="form-group">
            <button type="submit" class="btn btn-primary btn-block"> submit </button>
        </div> <!-- form-group -->
    </form>
</div> <!-- card-body -->
</div> <!-- card -->

<p class="text-center mt-4">Don't have account? <a href="{% url 'register' %}"#>Sign up</a></p>
<br><br>

<!-- ===== COMPONENT LOGIN END.//
===== -->

</section>

{% endblock content %}

```

Login page:

```
{% extends 'base.html' %}
```

```
{% load static %}
```

```
{% block content %}
```

```
<section class="section-content padding-y" style="min-height:84vh">
```

```
<!-- ===== COMPONENT LOGIN ===== -->
```

```
{% if request.GET.command %}
```

```
<div class="card mx-auto alert alert-info text-center" role="alert" style="max-width: 800px; margin-top:100px;">
```

```
<p>Registration successful, Please verify your email, We have sent an verification email to {{request.GET.email}}</p>
```

```
<p>Already verified? <a href="{% url 'login' %}">Login</a></p>
```

```
</div>
```

```
{% else %}
```

```
<div class="card mx-auto" style="max-width: 380px; margin-top:100px;">
```

```
<div class="card-body">
```

```
{% include 'alerts.html' %}
```

```
<h4 class="card-title mb-4">Sign in</h4>
```

```
<form action="{% url 'login' %}" method="POST">
```

```
{% csrf_token %}
```

```
<div class="form-group">
```

```
<input class="form-control" name="email" placeholder="Email Address" >
```

```
</div> <!-- form-group -->
```

```
<div class="form-group">
```

```
<input type="password" class="form-control" name="password" placeholder="Password" >
```

```
</div> <!-- form-group -->
```

```

    <div class="form-group">
        <a href="{% url 'forgotPassword' %}" class="float-right">Forgot password?</a>

    </div> <!-- form-group form-check .// -->

    <div class="form-group">
        <button type="submit" class="btn btn-primary btn-block"> Login </button>
    </div> <!-- form-group// -->
</form>
</div> <!-- card-body.// -->
</div> <!-- card .// -->

<p class="text-center mt-4">Don't have account? <a href="{% url 'register' %}#">Sign up</a></p>
<br><br>

<!-- ===== COMPONENT LOGIN END.//
===== -->

{% endif %}

</section>

{% endblock content %}

```

My order page:

```
{% extends 'base.html' %}
```

```
{% block content %}
```

```
<section class="section-conten padding-y bg">
```

```
    {% include 'alerts.html' %}
```

```
    <div class="container">
```

```
        <div class="row">
```



```

{% include 'includes/dashboard_sidebar.html' %}

<main class="col-md-9">

  <article class="card">

    <header class="card-header" style="background: rebeccapurple; color:white">

      <strong class="d-inline-block mr-3">Your Order history</strong>

    </header>

    <div class="card-body">

      <div class="row">

        <div class="col-md-12">

          <table class="table table-hover">

            <thead>

              <tr>

                <th scope="col">Order #</th>

                <th scope="col">Billig Name</th>

                <th scope="col">Phone Number</th>

                <th scope="col">Order Total</th>

                <th scope="col">Ordered Date</th>

              </tr>

            </thead>

            <tbody>

              {% for order in orders %}

                <tr>

                  <th scope="row"><a href="{% url 'order_detail' order.order_number
%}">{{order.order_number}}</a></th>

                  <td>{{order.full_name}}</td>

                  <td>{{order.phone}}</td>

                  <td>${{order.order_total}}</td>

                  <td>{{order.created_at}}</td>

                </tr>

              {% endfor %}

            </tbody>

```

```

        </table>

        </div>

        </div> <!-- row.// -->

        </div> <!-- card-body .// -->

        </article> <!-- order-group.// -->

    </main>

</div> <!-- row.// -->

</div>

</section>

{% endblock content %}

```

Register page:

```
{% extends 'base.html' %}

{% load static %}
```

```
{% block content %}
```

```
<!-- ===== COMPONENT REGISTER
===== -->

<div class="card mx-auto" style="max-width:520px; margin-top:40px;">

  <article class="card-body">

    {% include 'alerts.html' %}

    <header class="mb-4"><h4 class="card-title">Sign up</h4></header>

    <form id="registration-form" method="POST" action="{% url 'register' %}">

      {% csrf_token %}

      <div class="form-row">

        <div class="col form-group">
```

```
<label>First name</label>
    {{form.first_name}}
</div> <!-- form-group end.// -->
<div class="col form-group">
    <label>Last name</label>
    {{form.last_name}}
</div> <!-- form-group end.// -->
</div> <!-- form-row end.// -->
```

```
<div class="form-row">
    <div class="form-group col-md-6">
        <label>Phone Number</label>
        {{form.phone_number}}
    </div> <!-- form-group end.// -->
    <div class="form-group col-md-6">
        <label>Email</label>
        {{form.email}}
    </div> <!-- form-group end.// -->
</div> <!-- form-row.// -->
```

```
<div class="form-row">
    <div class="form-group col-md-6">
        <label>Create password</label>
        {{form.password}}
    </div> <!-- form-group end.// -->
    <div class="form-group col-md-6">
        <label>Repeat password</label>
        {{form.confirm_password}}
    </div> <!-- form-group end.// -->
</div>
```

```
<div class="form-group">
    <button type="submit" class="btn btn-primary btn-block"> Register </button>
```

```

        </div> <!-- form-group// -->

        {{form.email.errors}}

        {{form.non_field_errors}}

    </form>

</article><!-- card-body.// -->

</div> <!-- card .// -->

<p class="text-center mt-4">Have an account? <a href="{% url 'login' %}">Log In</a></p>

<br><br>

<!-- ===== COMPONENT REGISTER END.//
===== -->

```

{% endblock content %}

Reset pass page :

{% autoescape off %}

Hii {{user.first_name}}

To reset your password click on the link below

http://{{domain}}{% url 'resetpassword_validation' uidb64=uid token=token %}

if you think it's not you, ignore this Email.

{% endautoescape %}

Reset pass page:

```
{% extends 'base.html' %}
```

```
{% load static %}
```

```
{% block content %}
```

```
<section class="section-content padding-y" style="min-height:84vh">
```

```
<div class="card mx-auto" style="max-width: 380px; margin-top:100px;">
```

```
<div class="card-body">
```

```
    {% include 'alerts.html' %}
```

```
<h4 class="card-title mb-4">Reset Password</h4>
```

```
<form action="{% url 'resetPassword' %}" method="POST">
```

```
    {% csrf_token %}
```

```
<div class="form-group">
```

```
    <input class="form-control" name="password" placeholder="Create password" >
```

```
</div> <!-- form-group -->
```

```
<div class="form-group">
```

```
    <input type="password" class="form-control" name="confirm_password"
placeholder="Confirm password" >
```

```
</div> <!-- form-group -->
```

```
<div class="form-group">
```

```
    <button type="submit" class="btn btn-primary btn-block"> Reset </button>
```

```
</div> <!-- form-group -->
```

```
</form>
```

```
</div> <!-- card-body -->
```

</div> <!-- card .// -->

<p class="text-center mt-4">Don't have account? Sign up</p>

</section>

{% endblock content %}

Verification email page:

{% autoescape off %}

Hii {{user.first_name}}

Please click on the link below to activate your account

http://{{domain}}{% url 'activate' uidb64=uid token=token %}

if you think it's not you, ignore this Email.

{% endautoescape %}

Python codes :

Account folder :

Admin.py page:

```
from django.contrib import admin
from .models import Account,UserProfile
from django.contrib.auth.admin import UserAdmin
from django.utils.html import format_html
```

Register your models here.

```
class AccountAdmin(UserAdmin):
    list_display = ('email', 'first_name', 'last_name', 'username', 'date_joined', 'last_login', 'is_active')
    list_display_links = ('email', 'first_name', 'username')
    ordering = ('-date_joined',)
    readonly_fields = ('date_joined', 'last_login')

    filter_horizontal = ()
    list_filter = ()
    fieldsets = ()
```

```
class UserProfileAdmin(admin.ModelAdmin):

    def thumbnail(self, object):

        return format_html(''.format(object.profile_pic.url))

    list_display = ['thumbnail','user', 'city','state', 'country']


admin.site.register(Account, AccountAdmin)
admin.site.register(UserProfile, UserProfileAdmin)
```

Apps.py page:

```
from django.apps import AppConfig
```

```
class AccountsConfig(AppConfig):

    default_auto_field = 'django.db.models.BigAutoField'
    name = 'accounts'
```

forms.py page:

```
from django.contrib.auth.models import User
from django.db import models
from django.forms import fields
from .models import Account, UserProfile
from django import forms
```



```

class RegistrationForm(forms.ModelForm):

    password = forms.CharField(widget=forms.PasswordInput(attrs={
        'placeholder':'Enter password',
        'class':'form-control'
    }))

    confirm_password = forms.CharField(widget=forms.PasswordInput(attrs={
        'placeholder':'Confirm password',
        'class':'form-control'
    }))


class Meta:

    model = Account

    fields = ['first_name','last_name','email','password','phone_number']


def __init__(self, *args, **kwargs):

    super(RegistrationForm, self).__init__(*args, **kwargs)

    self.fields['first_name'].widget.attrs['placeholder'] = 'First Name'
    self.fields['last_name'].widget.attrs['placeholder'] = 'Last Name'
    self.fields['email'].widget.attrs['placeholder'] = 'Enter Email'
    self.fields['phone_number'].widget.attrs['placeholder'] = 'Enter Phone Number'
    self.fields['password'].widget.attrs['placeholder'] = 'Enter Password'
    self.fields['confirm_password'].widget.attrs['placeholder'] = 'Confirm Password'


    for field in self.fields:

        self.fields[field].widget.attrs['class'] = 'form-control'

        # self.fields[field].widget.attrs['placeholder'] = field


# Confirm password in the form itself
def clean(self):

```

```
cleaned_data = super(RegistrationForm, self).clean()
password = cleaned_data.get('password')
conf_password = cleaned_data.get('confirm_password')
```

```
if password != conf_password:
    raise forms.ValidationError(
        'Password Does Not Match'
    )
```

```
class UserForm(forms.ModelForm):
```

```
    class Meta :
```

```
        model = Account
```

```
        fields = ['first_name', 'last_name', 'phone_number']
```

```
    def __init__(self, *args, **kwargs):
```

```
        super(UserForm, self).__init__(*args, **kwargs)
```

```
        for field in self.fields:
```

```
            self.fields[field].widget.attrs['class'] = 'form-control'
```

```
class UserProfileForm(forms.ModelForm):
```

```
    profile_pic = forms.ImageField(required=False, error_messages={'Invalid': ('image files only')},
        widget=forms.FileInput)
```

```
    class Meta :
```

```
        model = UserProfile
```

```
        fields = ['address_line1', 'address_line2', 'city', 'state', 'country', 'profile_pic']
```

```
    def __init__(self, *args, **kwargs):
```

```
super(UserProfileForm,self)._init__(*args,**kwargs)

for field in self.fields:

    self.fields[field].widget.attrs['class']='form-control'
```

method.py page:

```
from django.db import models

from django.contrib.auth.models import AbstractBaseUser, BaseUserManager
```

Create your models here.

```
class MyAccountManager(BaseUserManager):

    def create_user(self, first_name, last_name, username, email, password=None):

        if not email:

            raise ValueError('User must have an email address')

        if not username:

            raise ValueError('User must have an username')

        user = self.model(

            email = self.normalize_email(email),

            username = username,

            first_name = first_name,

            last_name = last_name,

        )

        user.set_password(password)

        user.save(using=self._db)

        return user
```

```

def create_superuser(self, first_name, last_name, email, username, password):
    user = self.create_user(
        email = self.normalize_email(email),
        username = username,
        password = password,
        first_name = first_name,
        last_name = last_name,
    )
    user.is_admin = True
    user.is_active = True
    user.is_staff = True
    user.is_superuser = True
    user.save(using=self._db)
    return user

```

```

class Account(AbstractBaseUser):
    first_name = models.CharField(max_length=50)
    last_name = models.CharField(max_length=50)
    username = models.CharField(max_length=50, unique=True)
    email = models.EmailField(max_length=100, unique=True)
    phone_number = models.CharField(max_length=50)

    # required

    date_joined = models.DateTimeField(auto_now_add=True)
    last_login = models.DateTimeField(auto_now_add=True)
    is_admin = models.BooleanField(default=False)
    is_staff = models.BooleanField(default=False)
    is_active = models.BooleanField(default=False)
    is_superuser = models.BooleanField(default=False)

```

```
USERNAME_FIELD = 'email'
```

```
REQUIRED_FIELDS = ['username', 'first_name', 'last_name']
```

```
objects = MyAccountManager()
```

```
def full_name(self):
```

```
    return f'{self.first_name} {self.last_name}'
```

```
def _str_(self):
```

```
    return self.email
```

```
def has_perm(self, perm, obj=None):
```

```
    return self.is_admin
```

```
def has_module_perms(self, add_label):
```

```
    return True
```

```
class UserProfile(models.Model):
```

```
    user = models.OneToOneField(Account, on_delete=models.CASCADE)
```

```
    address_line1 = models.CharField(max_length=100, blank=True)
```

```
    address_line2 = models.CharField(max_length=100, blank=True)
```

```
    city = models.CharField(max_length=20, blank=True)
```

```
    state = models.CharField(max_length=20, blank=True)
```

```
    country = models.CharField(max_length=20, blank=True)
```

```
    profile_pic = models.ImageField(upload_to='user_profile/', blank=True)
```

```
def _str_(self):
```

```
    return self.user.first_name
```

```
def full_address(self):  
    return f'{self.address_line1} {self.address_line2}'
```

test.py :

```
from django.test import TestCase
```

```
# Create your tests here.
```

url.py:

```
from django.urls import path
```

```
from . import views
```

```
urlpatterns = [
```

```
    # Authentication
```

```
    path('register', views.register, name='register'),
```

```
    path('login', views.login, name='login'),
```

```
    path('logout', views.logout, name='logout'),
```

```
    # Account validations
```

```
    path('activate/<uidb64>/<token>/', views.activate, name='activate'),
```

```
    path('forgotPassword/', views.forgotPassword, name='forgotPassword'),
```

```
    path('resetpassword_validation/<uidb64>/<token>/', views.resetpassword_validation,  
        name='resetpassword_validation'),
```

```
    path('resetPassword/', views.resetPassword, name='resetPassword'),
```

```
    # Dashboard
```

```
    path('', views.dashboard, name='dashboard'),
```

```
    path('my_orders/', views.my_orders, name='my_orders'),
```

```
    path('edit_profile/', views.edit_profile, name='edit_profile'),
```

```
    path('change_password/', views.change_password, name='change_password'),
    path('order_detail/<int:order_id>/', views.order_detail, name='order_detail'),
]
```

Views.py page:

```
from django.forms import utils
from cart.models import Cart
from accounts.models import Account
from django.shortcuts import get_object_or_404, redirect, render, HttpResponseRedirect
from .forms import RegistrationForm
from .models import Account, UserProfile
from django.contrib import messages, auth
from django.contrib.auth.decorators import login_required

# email verification
from django.contrib.sites.shortcuts import get_current_site
from django.template.loader import render_to_string
from django.utils.http import urlsafe_base64_decode, urlsafe_base64_encode
from django.contrib.auth.tokens import default_token_generator
from django.core.mail import EmailMessage
from django.utils.encoding import force_bytes

from cart.models import Cart, CartItem
from cart.views import _cart_id
from orders.models import Order, OrderProduct

# forms to edit profile
from .forms import UserProfileForm, UserForm

# change password
# from django.contrib import ch
```

```

def register(request):
    if request.method == 'POST':
        form = RegistrationForm(request.POST)
        if form.is_valid():
            first_name = form.cleaned_data['first_name']
            phone_number = form.cleaned_data['phone_number']
            last_name = form.cleaned_data['last_name']
            email = form.cleaned_data['email']
            password = form.cleaned_data['password']
            # confirm_password = form.cleaned_data['confirm_password']
            username = email.split('@')[0]

            # if password == confirm_password:
            user = Account.objects.create_user(first_name=first_name, last_name=last_name, email=email,
                username=username,password=password)

            user.phone_number = phone_number
            user.save()

            # user activation
            current_site = get_current_site(request)
            email_subject = 'Account Activation'
            message = render_to_string('accounts/verification_email.html',{
                'user':user,
                'domain':current_site,
                'uid': urlsafe_base64_encode(force_bytes(user.pk)),
                'token': default_token_generator.make_token(user),
            })
            to_email = email
            send_mail = EmailMessage(email_subject,message, to=[to_email])

```



```

        send_mail.send()

        # messages.success(request, 'Registration successful, We have sent an verification email')
        return redirect('/accounts/login?command=verification&email='+email)
    else:
        form = RegistrationForm()

    context = {'form':form}
    return render(request, 'accounts/register.html', context)


def activate(request, uidb64, token):
    try:
        uid = urlsafe_base64_decode(uidb64).decode()
        user = Account._default_manager.get(pk=uid)
    except(ValueError, Account.DoesNotExist, TypeError, OverflowError):
        user = None

    if user is not None and default_token_generator.check_token(user, token):
        user.is_active = True
        user.save()
        messages.success(request, "Your account has been activated")
        return redirect('login')
    else:
        messages.error(request, 'invalid acitvation link')
        return redirect('register')

```

```

def login(request):

```

```

if request.method == 'POST':

    email = request.POST['email']

    password = request.POST['password']

    user = auth.authenticate(request, email=email, password=password)
    if user:

        try:

            cart = Cart.objects.get(cart_id=_cart_id(request))
        except Cart.DoesNotExist:

            cart = None

    is_cart_item_exist = CartItem.objects.filter(cart=cart).exists()
    if is_cart_item_exist:

        cart_items = CartItem.objects.filter(cart = cart)

        # Get product variation by cart_id
        product_variation = []

        for item in cart_items:

            variation = item.variation.all()

            product_variation.append(list(variation))

        # get cart item of user and store existing variations list
        cart_items = CartItem.objects.filter(user=user)

        ex_var_list = []

        id = []

        for item in cart_items:

            existing_variation = item.variation.all()

            ex_var_list.append(list(existing_variation))

            id.append(item.id)

        for pr in product_variation:

```

```

if pr in ex_var_list:
    index = ex_var_list.index(pr)
    item_id = id[index]

    item = CartItem.objects.get(id=item_id)
    print('item :', item)

    item.quantity +=1
    item.user = user
    print('item.user :',user)
    item.save()
else:
    cart_items = CartItem.objects.all().filter(cart=cart)
    for item in cart_items:
        item.user = user
        item.save()

else:
    messages.error(request, 'Invalid credentials')
    return redirect('login')

auth.login(request, user)
messages.success(request,'Now You logged in')
return redirect('dashboard')

return render(request, 'accounts/login.html')

```

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

```
def logout(request):
```

```
    auth.logout(request)
```

```
messages.success(request, 'You was logged out')  
return redirect('login')
```

```
def forgotPassword(request):  
    if request.method == 'POST':  
        email = request.POST['email']  
  
        if Account.objects.filter(email=email).exists():  
            user = Account.objects.get(email=email)  
  
            # Reset password email  
            current_site = get_current_site(request)  
            email_subject = 'Account Activation'  
            message = render_to_string('accounts/reset_password_email.html',{  
                'user':user,  
                'domain':current_site,  
                'uid': urlsafe_base64_encode(force_bytes(user.pk)),  
                'token': default_token_generator.make_token(user),  
            })  
            to_email = email  
            send_mail = EmailMessage(email_subject,message, to=[to_email])  
            send_mail.send()  
  
            messages.success(request, 'Reset password link has been sent to your email')  
            return redirect('login')  
        else:  
            messages.error(request, 'Email does not exist')  
            return redirect('forgotPassword')  
    return render(request, 'accounts/forgot_password.html')
```

```

def resetpassword_validation(request, uidb64, token):
    try:
        uid = urlsafe_base64_decode(uidb64).decode()
        print(uid)
        user = Account._default_manager.get(pk=uid)

    except(ValueError, Account.DoesNotExist, TypeError, OverflowError):
        user = None

    if user is not None and default_token_generator.check_token(user, token):
        request.session['uid'] = uid
        messages.success(request, 'Reset your password')
        return redirect('resetPassword')
    else:
        messages.error(request, 'This link has been expired.')
        return redirect('login')

def resetPassword(request):
    if request.method == 'POST':
        password = request.POST['password']
        confirm_password = request.POST['confirm_password']

        if password == confirm_password:
            uid = request.session.get('uid')
            user = Account.objects.get(pk=uid)
            user.set_password(password)
            user.save()
            messages.success(request, 'Password reset was succesful')

```

```
        return redirect('login')
    else:
        messages.error(request, 'Password in not matched')
        return redirect('resetPassword')
    else:
        return render(request, 'accounts/reset_password.html')
```

```
@login_required(login_url='login')
def dashboard(request):
    orders = Order.objects.filter(user_id=request.user.id,is_ordered=True).order_by('-created_at')
    order_count = orders.count()
    userprofile = UserProfile.objects.get(user_id=request.user.id)

    context = {
        'order_count' : order_count,
        'userprofile':userprofile
    }
    return render(request, 'accounts/dashboard.html',context)
```

```
@login_required(login_url='login')
def my_orders(request):
    orders = Order.objects.filter(user=request.user,is_ordered=True).order_by('-created_at')
    context={
        'orders':orders
    }
    return render(request, 'accounts/my_orders.html',context)
```

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

```

def edit_profile(request):
    userprofile = get_object_or_404(UserProfile, user=request.user)

    if request.method == 'POST':
        user_form = UserForm(request.POST, instance=request.user)
        profile_form = UserProfileForm(request.POST, request.FILES, instance=userprofile)

        if user_form.is_valid() and profile_form.is_valid():
            user_form.save()
            profile_form.save()
        else:
            user_form = UserForm(instance=request.user)
            profile_form = UserProfileForm(instance=userprofile)

    context = {
        'user_form': user_form,
        'profile_form': profile_form,
        'userprofile': userprofile
    }
    return render(request, 'accounts/edit_profile.html', context)

```

```

@login_required(login_url='login')

```

```

def change_password(request):
    if request.method == 'POST':
        current_password = request.POST['current_password']
        new_password = request.POST['confirm_password']
        confirm_password = request.POST['confirm_password']

        user = Account.objects.get(username_exact=request.user.username)

        if new_password == confirm_password:

```

```
    success = user.check_password(current_password)
    if success:
        user.set_password(new_password)
        user.save()
        messages.success(request, 'Your password has been changed')
        return redirect('change_password')
    else:
        messages.error(request, 'Invalid Current Password')
        return redirect('change_password')
else:
    messages.error(request, 'Password does not matched')
    return redirect('change_password')
context = {}
return render(request, 'accounts/change_password.html', context)
```

```
@login_required(login_url='login')
def order_detail(request, order_id):
    order_products = OrderProduct.objects.filter(order_order_number = order_id)
    order = Order.objects.get(order_number=order_id)
    sub_total = 0

    for i in order_products:
        sub_total += i.product.price * i.quantity
    context = {
        'order_products':order_products,
        'order':order,
        'sub_total':sub_total
    }
    return render(request, 'orders/order_detail.html', context)
```


cart app pages:

admin.py page:

```
from django.contrib import admin
from .models import Cart, CartItem
# Register your models here.

class CartItemAdmin(admin.ModelAdmin):
    list_display = ('product', 'cart','user','quantity', 'sub_total', 'is_active')

class CartAdmin(admin.ModelAdmin):
    list_display = ('cart_id', 'date_added')

admin.site.register(Cart, CartAdmin)
admin.site.register(CartItem, CartItemAdmin)
```

app.py page:

```
from django.apps import AppConfig

class CartConfig(AppConfig):
    default_auto_field = 'django.db.models.BigAutoField'
    name = 'cart'
```

context-processor page:

```
from .models import Cart, CartItem
from .views import _cart_id
```

```

def cart_counter(request):
    cart_count = 0

    try:
        if request.user.is_authenticated:
            cart_items = CartItem.objects.all().filter(user=request.user)
        else:
            cart_items = CartItem.objects.all().filter(cart_cart_id=_cart_id(request))

        for cart_item in cart_items:
            cart_count += cart_item.quantity

    except Cart.DoesNotExist:
        cart_count = 0

    return dict(cart_count=cart_count )

```

model.py pages:

```

from accounts.models import Account
from django.db import models
from store.models import Product, Variation

# Create your models here.

class Cart(models.Model):

    cart_id = models.CharField(max_length=50, blank=True)
    date_added = models.DateTimeField(auto_now_add=True)

    def __str__(self):
        return self.cart_id

```

```

class CartItem(models.Model):
    user = models.ForeignKey(Account, on_delete=models.CASCADE , null=True)
    product = models.ForeignKey(Product, on_delete=models.CASCADE)
    variation = models.ManyToManyField(Variation, blank=True)
    cart = models.ForeignKey(Cart, on_delete=models.CASCADE, null=True)
    quantity = models.IntegerField()
    is_active = models.BooleanField(default=True)

    def sub_total(self):
        return (self.product.price * self.quantity)

    def __str__(self):
        return str(self.product)

```

views.py page:

```

from django.core.exceptions import ObjectDoesNotExist
from store.models import Product, Variation
from django.shortcuts import get_object_or_404, redirect, render
from store.models import Product
from .models import CartItem, Cart
from django.contrib.auth.decorators import login_required
# Create your views here.

def _cart_id(request):
    cart = request.session.session_key
    if not cart:
        cart = request.session.create()

```

```
return cart
```

```
# Add items to cart with variations | if same variation & same product increase quantity else create new  
cart item
```

```
def add_cart(request, product_id):
```

```
    current_user = request.user
```

```
    # get the product which has to be added to cart
```

```
    product = Product.objects.get(id=product_id)
```

```
#####
```

```
##### if the user IS authenticated #####
```

```
#####
```

```
if current_user.is_authenticated:
```

```
    product_variation = []
```

```
    if request.method == 'POST':
```

```
        for item in request.POST:
```

```
            key = item
```

```
            value = request.POST[key]
```

```
            try:
```

```
                variation = Variation.objects.get(product=product, variation_category_iexact=key,  
variation_value_iexact=value)
```

```
                product_variation.append(variation)
```

```
            except:
```

```
                pass
```

```
    # check if the item is already in cart
```

```
    is_cart_item_exists = CartItem.objects.filter(user=current_user, product=product).exists()
```

```
    if is_cart_item_exists:
```

```
        cart_item = CartItem.objects.filter(product=product,user=current_user)
```

```
    # list of existing variations
```

```

ex_var_list = []

id = []

for item in cart_item:

    existing_variation = item.variation.all()

    ex_var_list.append(list(existing_variation))

    id.append(item.id)


# if new item variations is in existing cart item variations
if product_variation in ex_var_list:

    # ----- increase item quantity -----

    index = ex_var_list.index(product_variation)

    item_id = id[index]

    item = CartItem.objects.get(product=product, id=item_id)

    item.quantity += 1

    item.save()

else:

    item = CartItem.objects.create(product=product, quantity=1, user=current_user)

    if len(product_variation) > 0:

        item.variation.add(*product_variation)

        item.save()


# if item does not exist in cart then create new cart item
else:

    cart_item = CartItem.objects.create(

        product=product, user=current_user, quantity =1

    )

    cart_item.variation.clear()

    if len(product_variation) > 0:

        for item in product_variation:

            cart_item.variation.add(item)

    cart_item.save()

```

```
return redirect('cart')
```

```
#####
```

```
##### if the user NOT authenticated #####
```

```
#####
```

```
else:
```

```
# get all variations from user and get those form db and store them in a list
```

```
product_variation = []
```

```
if request.method == 'POST':
```

```
    for item in request.POST:
```

```
        key = item
```

```
        value = request.POST[key]
```

```
        try:
```

```
            variation = Variation.objects.get(product=product, variation_category_iexact=key,  
            variation_value_iexact=value)
```

```
            product_variation.append(variation)
```

```
        except:
```

```
            pass
```

```
# get user cart if not exist create one | cart_id = user session id
```

```
try:
```

```
    cart = Cart.objects.get(cart_id = _cart_id(request))
```

```
except Cart.DoesNotExist:
```

```
    cart = Cart.objects.create(
```

```
        cart_id = _cart_id(request),
```

```
    )
```

```
    cart.save()
```

```

# check if the item is already in cart
is_cart_item_exists = CartItem.objects.filter(cart=cart, product=product)

if is_cart_item_exists:
    cart_item = CartItem.objects.filter(product=product, cart=cart)

    # list of existing variations
    ex_var_list = []
    id = []
    for item in cart_item:
        existing_variation = item.variation.all()
        ex_var_list.append(list(existing_variation))
        id.append(item.id)

    # if new item variations is in existing cart item variations
    if product_variation in ex_var_list:
        # ----- increase item quantity -----
        index = ex_var_list.index(product_variation)
        item_id = id[index]
        item = CartItem.objects.get(product=product, id=item_id)
        item.quantity += 1
        item.save()
    else:
        item = CartItem.objects.create(product=product, quantity=1, cart=cart)
        if len(product_variation) > 0:
            item.variation.add(*product_variation)
            item.save()

# if item does not exist in cart then create new cart item
else:
    cart_item = CartItem.objects.create(
        product=product, cart=cart, quantity=1

```

```
)  
  
cart_item.variation.clear()  
  
if len(product_variation) > 0:  
    for item in product_variation:  
        cart_item.variation.add(item)  
  
cart_item.save()
```

```
return redirect('cart')
```

Delete entire cart item irrespective of quantity

```
def remove_cart(request, product_id, cart_item_id):
```

```
    product = get_object_or_404(Product, id = product_id)
```

```
    if request.user.is_authenticated:
```

```
        cart_item = CartItem.objects.get(user=request.user, product=product, id=cart_item_id)
```

```
    else:
```

```
        cart = Cart.objects.get(cart_id=_cart_id(request))
```

```
        cart_item = CartItem.objects.get(cart=cart, product=product, id=cart_item_id)
```

```
    if cart_item.quantity > 1:
```

```
        cart_item.quantity -= 1
```

```
        cart_item.save()
```

```
    else:
```

```
        cart_item.delete()
```

```
    return redirect('cart')
```

remove items from the cart | if multiple items decrease quantity

```
def remove_cart_item(request, product_id, cart_item_id):
```



```
product = get_object_or_404(Product, id = product_id)
if request.user.is_authenticated:
    cart_item = CartItem.objects.get(user=request.user, product=product, id=cart_item_id)
else:
    cart = Cart.objects.get(cart_id=_cart_id(request))
    cart_item = CartItem.objects.get(cart=cart, product=product, id=cart_item_id)

cart_item.delete()
return redirect('cart')
```

Reder cart page

```
def cart(request, cart_total=0, quantity=0, cart_item=None):
    try:
        if request.user.is_authenticated:
            cart_items = CartItem.objects.all().filter(user=request.user, is_active=True)
        else:
            cart = Cart.objects.get(cart_id=_cart_id(request))
            cart_items = CartItem.objects.filter(cart=cart, is_active=True)

        for cart_item in cart_items:
            cart_total += (cart_item.product.price * cart_item.quantity)
            quantity += cart_item.quantity

        tax = (2 * cart_total)/100
        grand_total = tax + cart_total

        context = {
            'cart_total':cart_total,
            'quantity':quantity,
```

```
        'cart_items':cart_items,
        'tax':tax,
        "grand_total":grand_total
    }
except ObjectDoesNotExist:
    pass
    context = {}
return render(request, 'cart.html',context)
```

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

```
def checkout(request, cart_total=0, quantity=0, cart_item=None):
```

```
    try:
```

```
        if request.user.is_authenticated:
```

```
            cart_items = CartItem.objects.all().filter(user=request.user, is_active=True)
```

```
        else:
```

```
            cart = Cart.objects.get(cart_id=_cart_id(request))
```

```
            cart_items = CartItem.objects.filter(cart=cart, is_active=True)
```

```
    for cart_item in cart_items:
```

```
        cart_total += (cart_item.product.price * cart_item.quantity)
```

```
        quantity += cart_item.quantity
```

```
    tax = (2 * cart_total)/100
```

```
    grand_total = tax + cart_total
```

```
    context = {
```

```
        'cart_total':cart_total,
```

```
        'quantity':quantity,
```

```
        'cart_items':cart_items,
```

```
        'tax':tax,
```

```
    "grand_total":grand_total  
}
```

```
except ObjectDoesNotExist:
```

```
    pass
```

```
    context = {}
```

```
return render(request, 'checkout.html', context)
```

Get ietm app pages:

Asgi.py page:

```
"""
```

ASGI config for tirriCart project.

It exposes the ASGI callable as a module-level variable named ``application``.

For more information on this file, see

<https://docs.djangoproject.com/en/3.2/howto/deployment/asgi/>

```
"""
```

```
import os
```

```
from django.core.asgi import get_asgi_application
```

```
os.environ.setdefault('DJANGO_SETTINGS_MODULE', 'tirriCart.settings')
```

```
application = get_asgi_application()
```

settings.py pages:

```
"""
```

Django settings for tirriCart project.

Generated by 'django-admin startproject' using Django 3.2.4.

For more information on this file, see

<https://docs.djangoproject.com/en/3.2/topics/settings/>

For the full list of settings and their values, see

<https://docs.djangoproject.com/en/3.2/ref/settings/>

"""

```
from pathlib import Path
```

```
import os
```

```
from decouple import config
```

```
# Build paths inside the project like this: BASE_DIR / 'subdir'.
```

```
BASE_DIR = Path(_file_).resolve().parent.parent
```

```
# Quick-start development settings - unsuitable for production
```

```
# See https://docs.djangoproject.com/en/3.2/howto/deployment/checklist/
```

```
# SECURITY WARNING: keep the secret key used in production secret!
```

```
SECRET_KEY = config('SECRET_KEY')
```

```
# SECURITY WARNING: don't run with debug turned on in production!
```

```
DEBUG = True
```

```
ALLOWED_HOSTS = ['127.0.0.1', 'http://127.0.0.1:8000/', 'ecommerce-env.eba-rcnp2i7q.us-west-2.elasticbeanstalk.com']
```

```
# Application definition
```

```
INSTALLED_APPS = [
```

```
    'django.contrib.admin',
```

```
    'django.contrib.auth',
```

```
    'django.contrib.contenttypes',
```

```
    'django.contrib.sessions',
```

```
    'django.contrib.messages',
```

```
'django.contrib.staticfiles',  
'category',  
'accounts',  
'store',  
'cart',  
'orders',  
)
```

```
MIDDLEWARE = [  
    'django.middleware.security.SecurityMiddleware',  
    'django.contrib.sessions.middleware.SessionMiddleware',  
    'django.middleware.common.CommonMiddleware',  
    'django.middleware.csrf.CsrfViewMiddleware',  
    'django.contrib.auth.middleware.AuthenticationMiddleware',  
    'django.contrib.messages.middleware.MessageMiddleware',  
    'django.middleware.clickjacking.XFrameOptionsMiddleware',  
)
```

```
ROOT_URLCONF = 'tirriCart.urls'
```

```
TEMPLATES = [  
    {  
        'BACKEND': 'django.template.backends.django.DjangoTemplates',  
        'DIRS': ['templates'],  
        'APP_DIRS': True,  
        'OPTIONS': {  
            'context_processors': [  
                'django.template.context_processors.debug',  
                'django.template.context_processors.request',  
                'django.contrib.auth.context_processors.auth',  
                'django.contrib.messages.context_processors.messages',
```

```
        'category.context_processors.menu_links',
        'cart.context_processors.cart_counter',
    ],
},
},
]
```

```
WSGI_APPLICATION = 'tirriCart.wsgi.application'
```

```
AUTH_USER_MODEL = 'accounts.Account'
```

```
# Database
```

```
# https://docs.djangoproject.com/en/3.2/ref/settings/#databases
```

```
if 'RDS_DB_NAME' in os.environ:
```

```
    DATABASES = {
        'default': {
            'ENGINE': 'django.db.backends.postgresql',
            'NAME': os.environ['RDS_DB_NAME'],
            'USER': os.environ['RDS_USERNAME'],
            'PASSWORD': os.environ['RDS_PASSWORD'],
            'NAME': os.environ['RDS_HOSTNAME'],
            'PORT': os.environ['RDS_PORT'],
        }
    }
```

```
else:
```

```
    DATABASES = {
        'default': {
            'ENGINE': 'django.db.backends.sqlite3',
            'NAME': BASE_DIR / 'db.sqlite3',
```

```
}  
}
```

Password validation

<https://docs.djangoproject.com/en/3.2/ref/settings/#auth-password-validators>

```
AUTH_PASSWORD_VALIDATORS = [  
    {  
        'NAME': 'django.contrib.auth.password_validation.UserAttributeSimilarityValidator',  
    },  
    {  
        'NAME': 'django.contrib.auth.password_validation.MinimumLengthValidator',  
    },  
    {  
        'NAME': 'django.contrib.auth.password_validation.CommonPasswordValidator',  
    },  
    {  
        'NAME': 'django.contrib.auth.password_validation.NumericPasswordValidator',  
    },  
]
```

Internationalization

<https://docs.djangoproject.com/en/3.2/topics/i18n/>

```
LANGUAGE_CODE = 'en-us'
```

```
TIME_ZONE = 'Asia/Kolkata'
```

```
USE_I18N = True
```



```
USE_L10N = True
```

```
USE_TZ = True
```

```
# Static files (CSS, JavaScript, Images)
```

```
# https://docs.djangoproject.com/en/3.2/howto/static-files/
```

```
STATIC_URL = '/static/'
```

```
STATICFILES_DIRS = [
```

```
    os.path.join(BASE_DIR, 'static'),
```

```
]
```

```
MEDIA_URL = '/media/'
```

```
MEDIA_ROOT = BASE_DIR/'media'
```

```
from django.contrib.messages import constants as messages
```

```
MESSAGE_TAGS = {
```

```
    messages.ERROR: 'danger',
```

```
}
```

```
# Default primary key field type
```

```
# https://docs.djangoproject.com/en/3.2/ref/settings/#default-auto-field
```

```
DEFAULT_AUTO_FIELD = 'django.db.models.BigAutoField'
```

```
# SMTP CONFIGURATION
```

```
EMAIL_HOST = config('EMAIL_HOST')
```

```
EMAIL_PORT = config('EMAIL_PORT', cast=int)
EMAIL_HOST_USER = config('EMAIL_HOST_USER')
EMAIL_HOST_PASSWORD = config('EMAIL_HOST_PASSWORD')
EMAIL_USE_TLS = config('EMAIL_USE_TLS', cast=bool)
```

Urls.py page:

```
"""tirriCart URL Configuration
```

The `urlpatterns` list routes URLs to views. For more information please see:

<https://docs.djangoproject.com/en/3.2/topics/http/urls/>

Examples:

Function views

1. Add an import: from my_app import views
2. Add a URL to urlpatterns: path("", views.home, name='home')

Class-based views

1. Add an import: from other_app.views import Home
2. Add a URL to urlpatterns: path("", Home.as_view(), name='home')

Including another URLconf

1. Import the include() function: from django.urls import include, path
2. Add a URL to urlpatterns: path('blog/', include('blog.urls'))

```
"""
```

```
from django.conf import settings
```

```
from django.conf.urls.static import static
```

```
from django.contrib import admin
```

```
from django.urls import path,include
```

```
from . import views
```

```
urlpatterns = [  
    path('secure-login/', admin.site.urls),  
    path("", views.home, name='home'),  
    # define path when have multiples apps  
    path('store/', include('store.urls')),  
    path('cart/', include('cart.urls')),  
    path('accounts/', include('accounts.urls')),  
    path('orders/', include('orders.urls')),  
  
] + static(settings.MEDIA_URL, document_root=settings.MEDIA_ROOT)
```

Views.py pages:

```
from django.shortcuts import render  
from store.models import Product  
from store.models import ReviewRating
```

```
def home(request):  
    products = Product.objects.all().filter(is_available=True).order_by('date_created')  
    # Get reviews
```

```
context = {'products':products}

return render(request, 'home.html', context)
```

wsgi.py page:

```
"""
```

WSGI config for tirriCart project.

It exposes the WSGI callable as a module-level variable named ``application``.

For more information on this file, see

<https://docs.djangoproject.com/en/3.2/howto/deployment/wsgi/>

```
"""
```

```
import os
```

```
from django.core.wsgi import get_wsgi_application
```

```
os.environ.setdefault('DJANGO_SETTINGS_MODULE', 'tirriCart.settings')
```

```
application = get_wsgi_application()
```