Clothes Online Store with Payment TRD

* **Purpose**

The purpose of the project is to build a web application for an online store. Users can buy merchandise from the store so we also need to set up a payment module to handle transactions.

* **Owner:** Ankit Kumar Malik

Technical Changes

* **Overview**

This simple E-commerce site is built using Python Django which has features for buying items from the store, add to the cart, change or update the items in cart and checkout.

* **Use Case**

Admin Features

* **Dashboard –** For the admin dashboard, you will be able to all the basic access in the whole system. Such as summary of products, orders, and the likes.
* **Manage Products** – The admin has access to the product management information system. He can add, update and delete the product.
* **Manage Orders** – As the main functions of the admin, the admin can accept or reject the order from the customers on a case to case basis.
* **Manage Customer**– The admin can manage the customer’s account. Admin can add, update and Block customer in the system.
* **Login and Logout** – By default one of the security features of this system is the secure login and logout system.

Technical Requirements

* Python
* Django
* HTML
* CSS
* SQLite Studio
* VS Code studio
* Pages that a user can visit:
* **Login** **Page**– Customer enter their website credentials on this page to gain access to all of the website’s features.
* **Home Page**– When customer visit the website, this is the system’s default page. This page shows the products for sale in the store, or by entering a keyword in the search box above the products.
* **Customers Profile** – In the frontend, the customer can register and update their profile. In short, this system has a customer management system feature.
* **Product View Page** – The page on which the product’s specific information is shown, as well as the page on which the customer adds the product to his or her cart.
* **Cart List Page**– The page that lists the items that customer have chosen. This is the page where the customer can complete the order checkout process.
* **My Order Page** – The page that lists the customer’s orders.
* **Payments** – This Online Shopping Project in Django has a payment method that uses Payments.
* **FAQs** – Frequently Asked Question by the user make the developer and the site to add the most asked product. And monitor which products are having more issue.

* APIs
* Authentication with google: The internet is an extremely dangerous place for an online store without any form of security. So authorising with google account will API give genuine users authorization and protection from malicious web users.
* Shipping APIs: Having a robust **shipping and delivery API** helps reduce errors and complications associated with manual sorting. eCommerce retailers can use this API to coordinate the customer’s experience from the moment they complete a purchase to the moment it is delivered at their doorstep.
* Payment gateway: This API provide the payment gateway so the user can make their online payment. A payment API makes it easy for both developers and businesses to manage transactions. It also provides a simplified interface for users to present their card details.