

بِسْمِ اللَّهِ الرَّحْمَنِ الرَّحِيمِ

موضوع پروژه:

طراحی و توسعه‌ی یک فروشگاه اینترنتی با استفاده از چارچوب Django

نام و نام خانوادگی: علیرضا غلام زاده

شماره دانشجویی: ۴۰۰۷۰۴۷۰

رشته تحصیلی: مهندسی کامپیوتر (نرم افزار)

مقطع تحصیلی: کارشناسی

استاد راهنما: سرکار خانم سهرابی

نام دانشگاه: دانشگاه شیراز - دانشکده آموزش های الکترونیکی

فهرست مطالب

صفحه	عنوان
۱	فصل اول: مقدمه
۲	فصل دوم: معرفی با فریم‌ورک Django
۴	فصل سوم: تحلیل و نیازمندی‌های سیستم
۷	فصل چهارم: طراحی و پیاده‌سازی سیستم
۱۱	فصل پنجم: بررسی عملکرد و نتایج پیاده‌سازی
۱۳	فصل ششم: نتیجه‌گیری و پیشنهادات

فصل اول: مقدمه

۱-۱. مقدمه

با گسترش روزافزون اینترنت و پیشرفت فناوری‌های مبتنی بر وب، استفاده از سامانه‌های آنلاین در زمینه‌های مختلف به‌ویژه تجارت الکترونیک افزایش چشم‌گیری داشته است. امروزه فروشگاه‌های اینترنتی به یکی از مهم‌ترین ابزارها برای عرضه کالا و خدمات تبدیل شده‌اند و نقش مؤثری در تسهیل فرآیند خرید و فروش ایفا می‌کنند. این نوع فروشگاه‌ها امکان دسترسی آسان کاربران به محصولات، مقایسه قیمت‌ها و ثبت سفارش به‌صورت غیرحضوری را فراهم می‌سازند.

۱-۲. بیان مسئله

طراحی و پیاده‌سازی یک فروشگاه اینترنتی نیازمند استفاده از ابزارها و چارچوب‌هایی است که علاوه بر سادگی، از امنیت و قابلیت توسعه مناسبی برخوردار باشند. در این میان، فریم‌ورک Django به‌عنوان یکی از فریم‌ورک‌های قدرتمند زبان برنامه‌نویسی Python، امکانات متعددی را برای توسعه سریع و امن برنامه‌های تحت وب فراهم می‌کند. هدف این پروژه، آشنایی عملی با فرآیند طراحی و توسعه یک وب‌سایت فروشگاهی و استفاده از قابلیت‌های فریم‌ورک Django در پیاده‌سازی یک سیستم واقعی است.

۱-۳. هدف از انجام پروژه

هدف اصلی از انجام این پروژه، طراحی و توسعه‌ی یک فروشگاه اینترنتی با استفاده از چارچوب Django می‌باشد. در این پروژه تلاش شده است تا یک فروشگاه آنلاین با امکانات پایه و ضروری از جمله نمایش محصولات، دسته‌بندی کالاها، ثبت‌نام و ورود کاربران، سبد خرید، ثبت سفارش و مدیریت سفارش‌ها پیاده‌سازی شود. همچنین این پروژه با هدف افزایش مهارت عملی در کار با فریم‌ورک Django و آشنایی با ساختار پروژه‌های تحت وب انجام شده است.

۱-۴. معرفی کلی پروژه

پروژه حاضر یک وب‌سایت فروشگاهی با موضوع فروش عطر و ادکلن می‌باشد که با استفاده از زبان

برنامه‌نویسی Python و فریم‌ورک Django پیاده‌سازی شده است. این سامانه دارای بخش‌های مختلفی از جمله صفحه محصولات، صفحه جزئیات محصول، گالری تصاویر، سبد خرید، ثبت سفارش، پروفایل کاربری و پنل مدیریت می‌باشد. همچنین مدیر سیستم می‌تواند از طریق پنل ادمین Django محصولات و سفارش‌ها را مدیریت نماید. این پروژه به صورت محلی اجرا شده و به عنوان یک پروژه دانشجویی طراحی و توسعه داده شده است.

فصل دوم: معرفی فریم‌ورک Django

۲-۱. معرفی فریم‌ورک Django

Django یک فریم‌ورک متن‌باز و سطح بالا برای توسعه‌ی برنامه‌های تحت وب است که با استفاده از زبان برنامه‌نویسی Python پیاده‌سازی شده است. این فریم‌ورک با هدف افزایش سرعت توسعه، کاهش پیچیدگی پیاده‌سازی و ارتقای امنیت طراحی شده و بسیاری از نیازهای رایج در پروژه‌های وب را به صورت پیش‌فرض پوشش می‌دهد.

Django مبتنی بر اصل Don't Repeat Yourself (DRY) است که موجب کاهش تکرار کد و افزایش خوانایی آن می‌شود. این فریم‌ورک به‌ویژه برای توسعه وبسایت‌های داده‌محور، مانند فروشگاه‌های اینترنتی، سیستم‌های مدیریت محتوا و سامانه‌های تحت وب سازمانی بسیار مناسب است.

از مهم‌ترین ویژگی‌های Django می‌توان به سیستم مسیریابی (URL Routing)، سیستم قالب (Template Engine)، پنل مدیریت پیش‌فرض و ORM قدرتمند اشاره کرد که همگی در این پروژه مورد استفاده قرار گرفته‌اند.

۲-۲. معماری MVT در Django

فریم‌ورک Django بر پایه‌ی معماری Model-View-Template (MVT) طراحی شده است که نوعی پیاده‌سازی از الگوی MVC محسوب می‌شود. در این معماری، هر بخش وظیفه‌ی مشخصی را بر عهده دارد:

Model: مسئول تعریف ساختار داده‌ها، ارتباط با پایگاه داده و پیاده‌سازی منطق مربوط به داده‌ها است. در این پروژه، مدل‌ها برای تعریف موجودیت‌هایی مانند محصولات، سفارش‌ها و کاربران استفاده شده‌اند. View: منطق اصلی برنامه در این بخش قرار دارد. View درخواست‌های ارسال شده از سمت کاربر را دریافت کرده، پردازش لازم را انجام می‌دهد و پاسخ مناسب را آماده می‌کند. View‌ها نقش اصلی در کنترل جریان داده بین Model و Template را ایفا می‌کنند.

Template: وظیفه‌ی نمایش داده‌ها به کاربر را بر عهده دارد. این بخش شامل فایل‌های HTML است که با استفاده از Template Engine جنگو، داده‌های دریافتی از View را به صورت پویا نمایش می‌دهند. استفاده از معماری MVT باعث جداسازی مسئولیت‌ها، افزایش خوانایی کد و سهولت در اشکال‌زدایی می‌شود، که در پروژه حاضر نیز این مزایا به خوبی قابل مشاهده است.

۲_۳. مزایای استفاده از Django

فریم‌ورک Django دارای ویژگی‌ها و مزایای متعددی است که آن را به گزینه‌ای مناسب برای پیاده‌سازی پروژه‌های فروشگاهی تبدیل می‌کند. از جمله مهم‌ترین این مزایا می‌توان به موارد زیر اشاره کرد:

- سرعت بالای توسعه به دلیل وجود ابزارها و ماژول‌های آماده
 - امنیت پیش فرض بالا شامل محافظت در برابر حملات رایج مانند SQL Injection، Cross-Site Scripting (XSS) و Cross-Site Request Forgery (CSRF)
 - ORM قدرتمند برای تعامل با پایگاه داده بدون نیاز به نوشتن مستقیم دستورات SQL
 - پنل مدیریت (Admin) داخلی برای مدیریت سریع و ساده‌ی داده‌ها
 - قابلیت توسعه پذیری و مقیاس پذیری برای پروژه‌های کوچک تا بزرگ
 - پشتیبانی گسترده و مستندات کامل که یادگیری و توسعه پروژه را تسهیل می‌کند
- این ویژگی‌ها باعث شده‌اند Django در میان فریم‌ورک‌های وب Python محبوبیت بالایی داشته باشد.

۴_۲. کاربرد Django در پروژه حاضر

در این پروژه، از فریم‌ورک Django به عنوان هسته‌ی اصلی سیستم فروشگاهی استفاده شده است. ساختار پروژه به گونه‌ای طراحی شده که هر بخش اصلی سایت در قالب یک اپلیکیشن مجزا پیاده‌سازی شده که این امر موجب افزایش نظم و ماژولار بودن پروژه شده است.

در این پروژه از قابلیت‌های مختلف Django استفاده شده است، از جمله:

- سیستم احراز هویت پیش فرض برای مدیریت ثبت نام و ورود کاربران
 - ORM برای مدیریت داده‌های مربوط به محصولات، سفارش‌ها و کاربران
 - سیستم قالب Django برای پیاده‌سازی رابط کاربری پویا
 - پنل مدیریت جهت مدیریت محصولات، سفارش‌ها و کاربران توسط مدیر سایت
- استفاده از این امکانات باعث شده فرآیند توسعه ساده‌تر شده و پروژه از نظر ساختاری، قابل توسعه و نگهداری باشد.

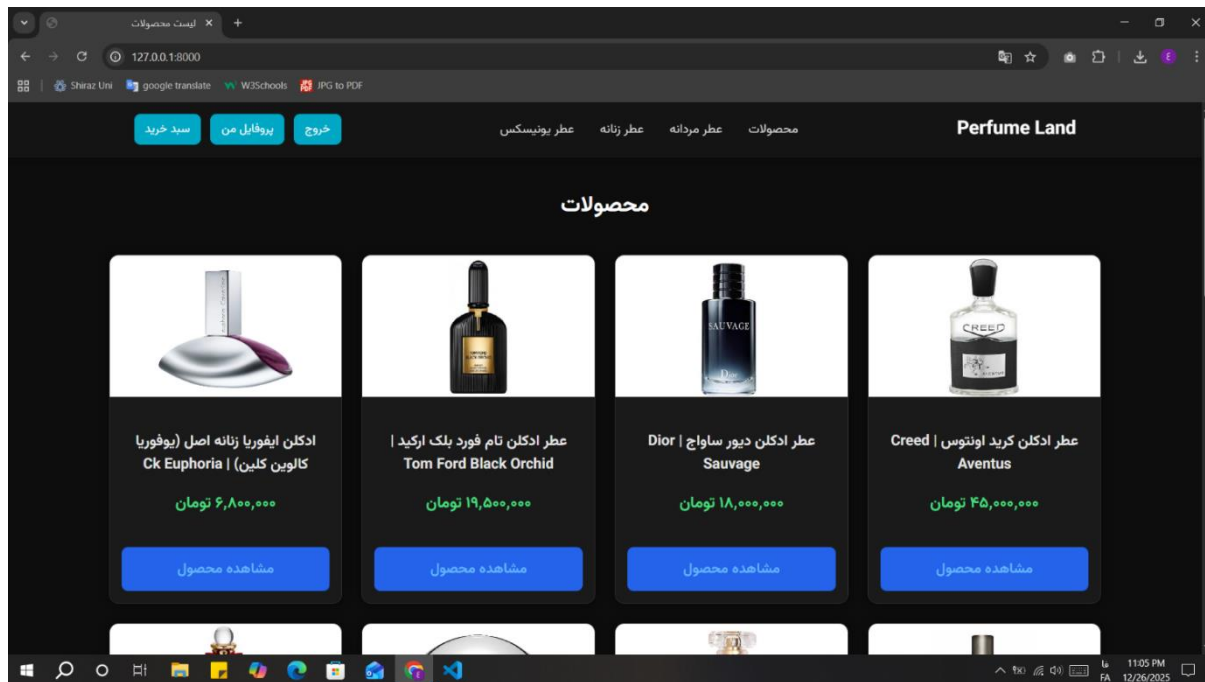
فصل سوم: تحلیل و نیازمندی‌های سیستم

۳_۱. تحلیل کلی سیستم

در این پروژه، یک فروشگاه اینترنتی با موضوع فروش عطر و ادکلن طراحی و پیاده‌سازی شده است. هدف اصلی سیستم، فراهم‌سازی بستری آنلاین جهت معرفی و عرضه محصولات، مدیریت کاربران، ثبت سفارش و پیگیری فرآیند خرید می‌باشد.

سیستم به گونه‌ای طراحی شده است که کاربران بتوانند پس از ثبت نام و ورود به حساب کاربری، محصولات موجود را مشاهده کرده، جزئیات هر محصول را بررسی نموده و محصولات مورد نظر خود را به سبد خرید اضافه کنند. در نهایت، امکان ثبت سفارش و مشاهده وضعیت سفارش‌ها برای کاربران فراهم شده است.

این فروشگاه اینترنتی نمونه‌ای از یک سیستم وب محور مبتنی بر پایگاه داده بوده و تمرکز اصلی آن بر پیاده‌سازی قابلیت‌های پایه یک فروشگاه آنلاین در قالب یک پروژه آموزشی می‌باشد.



تصویر ۱: نمایی از صفحه محصولات سایت

۳_۲. کاربران سیستم

در سیستم طراحی شده، دو نوع کاربر اصلی تعریف شده است:

۱. کاربر عادی (مشتري):

کاربران عادی می‌توانند در سامانه ثبت‌نام کرده و وارد حساب کاربری خود شوند. این کاربران امکان مشاهده محصولات، بررسی جزئیات هر محصول، افزودن محصولات به سبد خرید، ثبت سفارش و مشاهده سفارش‌های ثبت‌شده خود را دارند. همچنین امکان مدیریت و ویرایش اطلاعات پروفایل کاربری برای آن‌ها فراهم شده است.

۲. مدیر سیستم (ادمین):

مدیر سیستم از طریق پنل مدیریت Django قادر است اطلاعات مربوط به محصولات، دسته‌بندی‌ها و سفارش‌های ثبت‌شده را مدیریت نماید. این پنل امکان افزودن، ویرایش و حذف اطلاعات را بدون نیاز به پیاده‌سازی رابط مدیریتی جداگانه فراهم می‌کند.

۳_۳. نیازمندی‌های عملکردی سیستم

نیازمندی‌های عملکردی سیستم، قابلیت‌هایی هستند که سیستم باید به صورت مستقیم در اختیار

کاربران قرار دهد. مهم‌ترین نیازمندی‌های عملکردی این پروژه عبارت‌اند از:

- امکان ثبت‌نام، ورود و خروج کاربران
- نمایش لیست محصولات فروشگاه
- نمایش جزئیات هر محصول شامل نام، قیمت و توضیحات
- دسته‌بندی محصولات (مردانه، زنانه، یونیسکس)
- افزودن و حذف محصولات از سبد خرید
- ثبت سفارش و ذخیره اطلاعات سفارش در پایگاه داده
- مشاهده لیست سفارش‌های ثبت‌شده توسط کاربر
- مشاهده جزئیات هر سفارش
- مدیریت اطلاعات پروفایل کاربری
- مدیریت محصولات و سفارش‌ها از طریق پنل ادمین

۳-۴. نیازمندی‌های غیرعملکردی سیستم

نیازمندی‌های غیرعملکردی به ویژگی‌های کیفی سیستم اشاره دارند که مستقیماً به عملکرد خاصی مربوط نیستند اما نقش مهمی در کیفیت کلی سیستم ایفا می‌کنند. مهم‌ترین نیازمندی‌های غیرعملکردی این پروژه شامل موارد زیر است:

- رابط کاربری ساده و قابل فهم برای کاربران
- امنیت مناسب در بخش احراز هویت و ورود کاربران
- سرعت مناسب در بارگذاری صفحات و نمایش اطلاعات
- ساختار منظم و قابل توسعه کدها
- قابلیت توسعه و افزودن امکانات جدید در آینده

۳_۵. محدودیت‌ها و فرضیات پروژه

پروژه حاضر به‌عنوان یک پروژه دانشجویی طراحی و پیاده‌سازی شده است و با هدف آموزش مفاهیم توسعه وب و آشنایی عملی با فریم‌ورک Django انجام شده است. از این‌رو، برخی قابلیت‌ها از جمله اتصال به درگاه پرداخت آنلاین واقعی و سیستم ارسال پیامک یا ایمیل در نسخه فعلی پیاده‌سازی نشده‌اند.

در مرحله نهایی، به‌منظور امکان ارزیابی خروجی سیستم، پروژه بر روی یک سرور رایگان (Render) مستقر شده است. این محیط، با وجود فراهم‌کردن امکان دسترسی آنلاین به سامانه، دارای محدودیت‌هایی از جمله منابع سخت‌افزاری محدود، زمان پاسخ‌دهی متغیر و عدم پشتیبانی از امکانات پیشرفته امنیتی و مقیاس‌پذیری می‌باشد.

با این حال، ساختار پروژه به‌گونه‌ای طراحی شده است که در صورت نیاز، قابلیت استقرار بر روی سرورهای حرفه‌ای و توسعه به یک سیستم عملیاتی در مقیاس بزرگ‌تر را دارا می‌باشد.

فصل چهارم: طراحی و پیاده‌سازی سیستم

۴_۱. ساختار کلی پروژه

پروژه حاضر با استفاده از فریم‌ورک Django و بر اساس ساختار استاندارد این فریم‌ورک پیاده‌سازی شده است. در طراحی پروژه تلاش شده تا اصول مهندسی نرم‌افزار از جمله تفکیک مسئولیت‌ها، خوانایی کد و قابلیت توسعه رعایت شود.

به همین منظور، بخش‌های مختلف سیستم در قالب اپلیکیشن‌های مستقل Django طراحی شده‌اند و هر اپلیکیشن وظیفه مشخصی را بر عهده دارد. همچنین فایل‌های مربوط به قالب‌ها (Templates)، فایل‌های ایستا (Static Files) و فایل‌های رسانه‌ای (Media) به‌صورت مجزا سازمان‌دهی شده‌اند که این موضوع باعث نظم بیشتر پروژه و سهولت در نگهداری آن می‌شود.

۴_۲. اپلیکیشن های پروژه

پروژه شامل چند اپلیکیشن اصلی است که هر یک بخشی از منطق سیستم فروشگاه را پیاده سازی می کنند. این اپلیکیشن ها عبارت اند از:

accounts: این اپلیکیشن مسئول مدیریت کاربران سیستم است و قابلیت هایی مانند ثبت نام، ورود، خروج کاربران و ویرایش اطلاعات پروفایل را فراهم می کند. در این بخش از سیستم احراز هویت پیش فرض Django استفاده شده است که امنیت و پایداری بالایی را فراهم می کند.

The image shows a screenshot of the 'Perfume Land' website. At the top, there is a navigation bar with the site name 'Perfume Land' on the right and several menu items in Persian: 'محصولات' (Products), 'عطر مردانه' (Men's Perfume), 'عطر زنانه' (Women's Perfume), and 'عطر بوتیکس' (Boutique Perfume). Below the navigation bar, there are two main columns. The left column is for registration ('ثبت نام') and contains fields for 'نام کاربری' (Username), 'ایمیل (اختیاری)' (Email - optional), 'رمز عبور' (Password), and 'تکرار رمز عبور' (Repeat password), along with a blue 'ثبت نام' (Register) button and a link 'حساب دارید؟ ورود' (Have an account? Login). The right column is for login ('ورود') and contains fields for 'نام کاربری' (Username) and 'رمز عبور' (Password), a blue 'ورود' (Login) button, and a link 'حساب ندارید؟ ثبت نام' (Don't have an account? Register).

تصویر ۲: نمایی از صفحه ورود / ثبت نام

products: اپلیکیشن products برای مدیریت محصولات فروشگاه طراحی شده است. این بخش شامل تعریف مدل محصولات، دسته بندی ها، نمایش جزئیات محصول و مدیریت تصاویر مربوط به هر محصول می باشد.

orders: اپلیکیشن orders مسئول مدیریت فرآیند سبد خرید و ثبت سفارش است. در این بخش، عملیات افزودن محصول به سبد خرید، ثبت سفارش نهایی و مشاهده وضعیت سفارش ها پیاده سازی شده است. shop: این اپلیکیشن مدیریت صفحات اصلی فروشگاه و نمایش کلی محصولات را بر عهده دارد و به عنوان نقطه اتصال بین بخش های مختلف سیستم عمل می کند.

استفاده از این ساختار ماژولار باعث شده تا توسعه و گسترش سیستم در آینده به سادگی امکان پذیر باشد.

۴_۳. طراحی پایگاه داده و مدل‌ها

برای طراحی پایگاه داده، از مدل‌های Django و قابلیت ORM این فریم‌ورک استفاده شده است. ساختار داده‌ها در فایل `models.py` هر اپلیکیشن تعریف شده و شامل اطلاعات مربوط به کاربران، محصولات، سفارش‌ها و آیتم‌های سبد خرید می‌باشد.

ارتباط بین مدل‌ها با استفاده از کلیدهای خارجی (Foreign Key) و سایر امکانات ORM پیاده‌سازی شده است. استفاده از ORM باعث شده تا بدون نیاز به نوشتن مستقیم دستورات SQL، عملیات پایگاه داده به‌صورت ایمن و ساده انجام شود و همچنین احتمال بروز خطاهای رایج پایگاه داده کاهش یابد.

۴_۴. پیاده‌سازی View ها و منطق برنامه

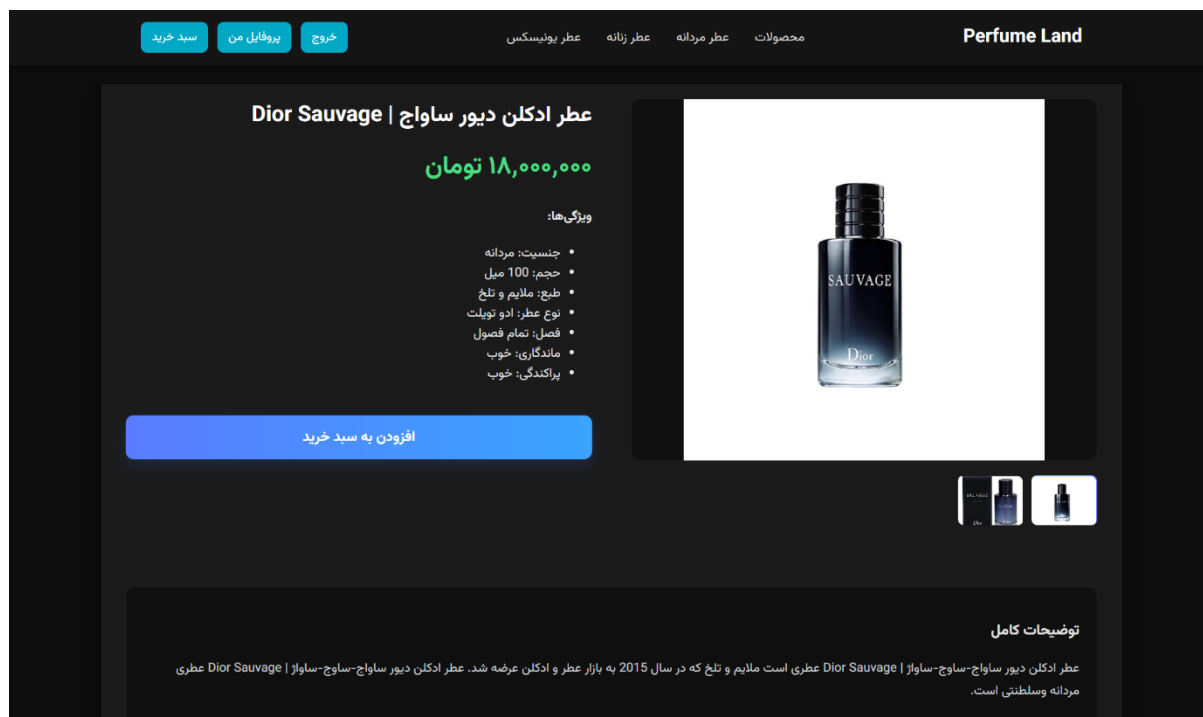
منطق اصلی برنامه در فایل‌های `views.py` هر اپلیکیشن پیاده‌سازی شده است. View ها وظیفه دریافت درخواست‌های کاربران، پردازش داده‌ها و ارسال پاسخ مناسب به قالب‌ها را بر عهده دارند.

در این پروژه، View ها برای نمایش لیست محصولات، جزئیات هر محصول، مدیریت سبد خرید، ثبت سفارش و نمایش اطلاعات کاربر استفاده شده‌اند. این بخش نقش اصلی را در ارتباط بین مدل‌ها و قالب‌ها ایفا می‌کند و هسته‌ی عملکردی سیستم محسوب می‌شود.

۴_۵. طراحی قالب‌ها (Templates)

برای طراحی رابط کاربری سیستم از سیستم قالب‌بندی Django استفاده شده است. صفحات مختلف سایت از جمله صفحه محصولات، جزئیات محصول، سبد خرید، پروفایل کاربری و صفحه سفارش‌ها با استفاده از قالب‌های HTML طراحی شده‌اند.

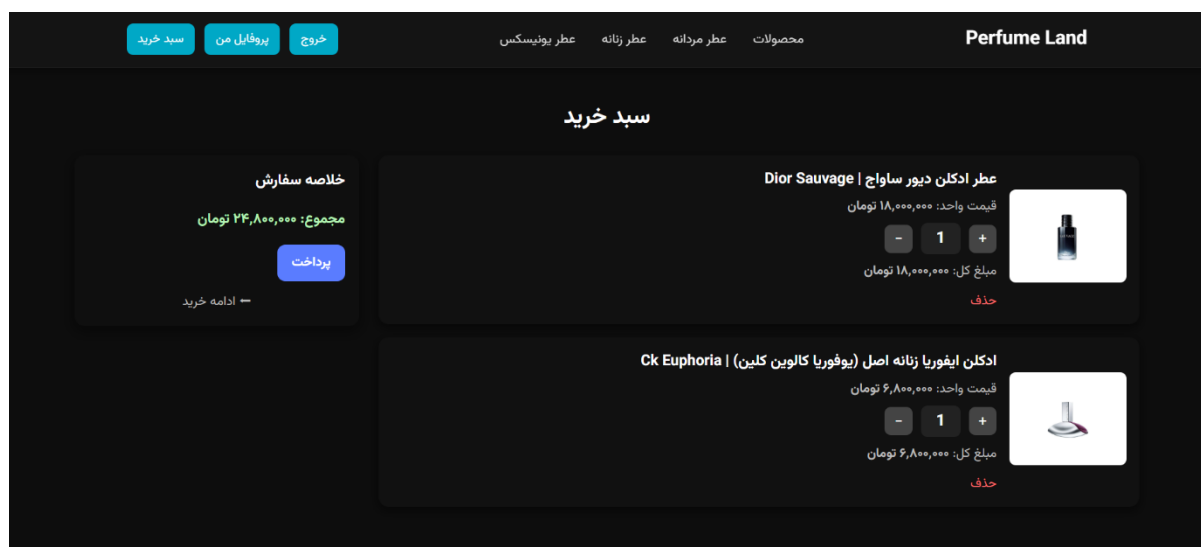
به‌منظور جلوگیری از تکرار کد، ساختار قالب‌ها به‌صورت ماژولار طراحی شده و بخش‌هایی مانند هدر، فوتر و منوی اصلی به‌صورت مشترک در قالب‌های مختلف مورد استفاده قرار گرفته‌اند. این موضوع باعث افزایش خوانایی و سهولت در اعمال تغییرات ظاهری سایت شده است.



تصویر ۳: نمایی از صفحه جزئیات محصول

۴_۶. سبد خرید و ثبت سفارش

سیستم سبد خرید به گونه‌ای طراحی شده است که کاربران بتوانند محصولات مورد نظر خود را انتخاب و به سبد خرید اضافه نمایند. پس از بررسی نهایی، کاربر قادر است سفارش خود را ثبت کند. اطلاعات مربوط به سفارش‌ها در پایگاه داده ذخیره شده و کاربران می‌توانند از طریق بخش «سفارش‌های من» وضعیت سفارش‌های ثبت‌شده‌ی خود را مشاهده کنند. این فرآیند با استفاده از Viewها و مدل‌های مربوطه پیاده‌سازی شده است.



تصویر ۴: نمایی از صفحه سبد خرید

۴-۷. پنل مدیریت سیستم

یکی از قابلیت‌های مهم فریم‌ورک Django، پنل مدیریت (Admin Panel) آن است. در این پروژه از پنل مدیریت Django برای مدیریت محصولات، دسته‌بندی‌ها و سفارش‌ها استفاده شده است. این پنل امکان افزودن، ویرایش و حذف اطلاعات را بدون نیاز به طراحی رابط کاربری جداگانه فراهم می‌کند و نقش مهمی در افزایش سرعت توسعه پروژه داشته است.

فصل پنجم: بررسی عملکرد و نتایج پیاده‌سازی

۵-۱. ارزیابی عملکرد سیستم

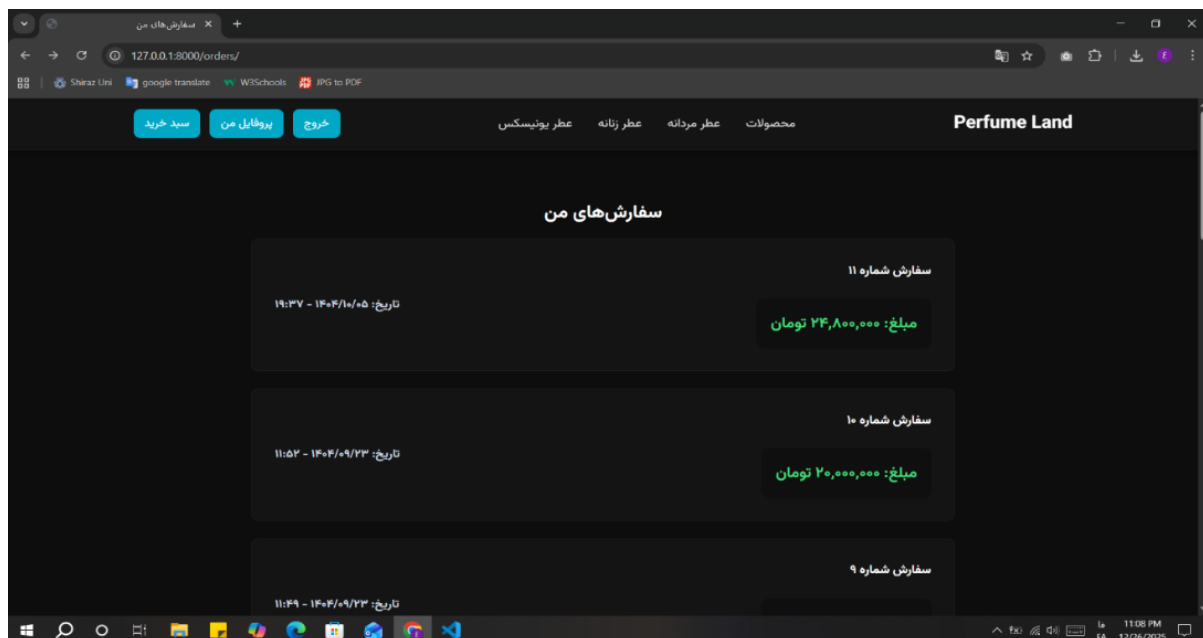
پس از تکمیل مراحل طراحی و پیاده‌سازی فروشگاه اینترنتی، عملکرد کلی سیستم مورد ارزیابی قرار گرفت. بررسی‌ها نشان می‌دهد که سامانه پیاده‌سازی شده قادر است سناریوهای اصلی یک فروشگاه اینترنتی را به درستی پشتیبانی نماید. سیستم از نظر ساختار نرم‌افزاری پایدار بوده و تعامل بین بخش‌های مختلف آن، از جمله لایه نمایش، منطق برنامه و پایگاه داده، به صورت هماهنگ انجام می‌شود. استفاده از فریم‌ورک Django و معماری MVT باعث شده است که پردازش درخواست‌ها با سرعت مناسب انجام شده و توسعه‌پذیری سیستم حفظ گردد.

۵-۲. بررسی فرآیند خرید و ثبت سفارش

فرآیند خرید در این سیستم به صورت مرحله‌ای و مطابق با استانداردهای رایج فروشگاه‌های اینترنتی طراحی شده است. کاربر پس از مشاهده لیست محصولات و جزئیات هر محصول، می‌تواند محصول مورد نظر را به سبد خرید اضافه نماید.

امکان تغییر تعداد اقلام، حذف محصول از سبد خرید و مشاهده خلاصه سفارش پیش از ثبت نهایی فراهم شده است. پس از ثبت سفارش، اطلاعات مربوطه در پایگاه داده ذخیره شده و کاربر قادر است از طریق بخش «سفارش‌های من» وضعیت سفارش‌های خود را مشاهده نماید.

این فرآیند با استفاده از Viewها و مدل‌های Django پیاده‌سازی شده و صحت عملکرد آن در تست‌های انجام‌شده تأیید گردیده است.

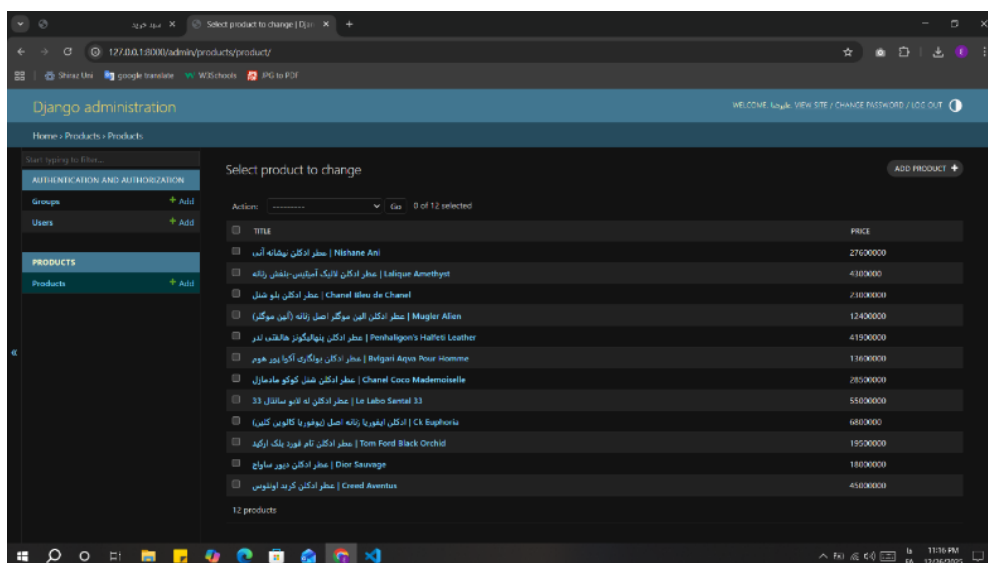


تصویر ۵: نمایی از صفحه سفارش‌های من

۵_۳. بررسی پنل مدیریت

یکی از نقاط قوت اصلی فریم‌ورک Django، پنل مدیریت قدرتمند آن می‌باشد که در این پروژه به‌طور کامل مورد استفاده قرار گرفته است. پنل مدیریت امکان انجام عملیات CRUD (افزودن، ویرایش، حذف و مشاهده) بر روی داده‌هایی مانند محصولات، دسته‌بندی‌ها و سفارش‌ها را برای مدیر سیستم فراهم می‌کند.

استفاده از این پنل موجب کاهش حجم کدنویسی، افزایش سرعت توسعه و سهولت در مدیریت سیستم شده است. همچنین سطح دسترسی کاربران مدیریتی از طریق سیستم احراز هویت Django کنترل می‌شود که این موضوع به افزایش امنیت سیستم کمک می‌کند.



تصویر ۶: نمایی از پنل مدیریت سیستم

۴_۵. بررسی رابط کاربری و تجربه کاربر

رابط کاربری سیستم با هدف سادگی، خوانایی و سهولت استفاده طراحی شده است. ساختار صفحات به گونه‌ای است که کاربران بدون نیاز به راهنمایی خاصی بتوانند به بخش‌های مختلف سایت دسترسی داشته باشند.

استفاده از قالب‌های مشترک برای اجزایی مانند هدر و فوتر باعث یکپارچگی بصری در صفحات مختلف شده است. همچنین رابط کاربری به صورت واکنش‌گرا (Responsive) طراحی شده و سایت در دستگاه‌های مختلف از جمله موبایل، تبلت و دسکتاپ به درستی نمایش داده می‌شود.

فصل ششم: نتیجه‌گیری و پیشنهادات

۱_۶. نتیجه‌گیری

در این پروژه، یک فروشگاه اینترنتی با استفاده از فریم‌ورک Django طراحی و پیاده‌سازی گردید. هدف اصلی پروژه، آشنایی عملی با مفاهیم توسعه وب، معماری نرم‌افزار و استفاده از قابلیت‌های یک فریم‌ورک قدرتمند سمت سرور در قالب یک پروژه کاربردی بود.

نتایج حاصل از اجرای پروژه نشان می‌دهد که سیستم پیاده‌سازی شده قادر است نیازهای اصلی یک فروشگاه اینترنتی را پوشش دهد. امکاناتی نظیر مدیریت کاربران، نمایش محصولات، سبد خرید، ثبت سفارش و مشاهده سفارش‌ها به صورت صحیح و پایدار در سیستم پیاده‌سازی شده‌اند.

استفاده از فریم‌ورک Django و معماری MVT باعث جداسازی مناسب منطق برنامه، لایه نمایش و داده‌ها شده و این موضوع به افزایش خوانایی کدها و سهولت در توسعه و نگهداری سیستم کمک کرده است. همچنین بهره‌گیری از ORM، سیستم احراز هویت پیش‌فرض و پنل مدیریت Django موجب کاهش پیچیدگی پیاده‌سازی و افزایش سرعت توسعه پروژه شده است.

در مرحله نهایی، پروژه علاوه بر اجرای محلی، بر روی یک سرور رایگان مبتنی بر پلتفرم Render مستقر گردید. این اقدام امکان ارزیابی خروجی نهایی سیستم در محیط واقعی را فراهم کرده و نشان داد که پروژه قابلیت اجرا و استفاده در بستر وب را نیز دارا می‌باشد.

۲_۶. محدودیت‌ها و پیشنهادات برای توسعه آینده

با توجه به دانشجویی بودن پروژه، برخی قابلیت‌ها در نسخه فعلی سیستم پیاده‌سازی نشده‌اند. با این حال، ساختار فعلی پروژه به گونه‌ای طراحی شده است که امکان توسعه و افزودن قابلیت‌های جدید در آینده وجود دارد. از جمله پیشنهادات برای توسعه‌های آتی می‌توان به موارد زیر اشاره کرد:

- اتصال سیستم به درگاه‌های پرداخت آنلاین
 - افزودن ارسال اعلان از طریق پیامک یا ایمیل پس از ثبت سفارش
 - امکان ثبت نظرات و امتیازدهی کاربران به محصولات
 - پیاده‌سازی فیلتر و مرتب‌سازی پیشرفته محصولات
 - توسعه پنل مدیریت اختصاصی با امکانات گزارش‌گیری
 - بهبود رابط کاربری و تجربه کاربری (UI/UX)
- افزودن این قابلیت‌ها می‌تواند سیستم را از یک پروژه آموزشی به یک فروشگاه اینترنتی کامل و قابل استفاده در مقیاس‌های بزرگ‌تر تبدیل نماید.
- در مجموع، این پروژه با هدف پیاده‌سازی یک فروشگاه اینترنتی مبتنی بر فریم‌ورک Django طراحی و اجرا شد. بخش‌های اصلی سیستم فروش آنلاین در آن پیاده‌سازی گردیده و ساختار پروژه به گونه‌ای است که امکان توسعه و تکمیل آن در مراحل بعدی وجود دارد.