

دورنمای کلی پروژه :

این پروژه، یک فروشگاه تحت وب برای یک فروشگاه واقعی می باشد. به بیانی واضح تر این پروژه، یک سایت فروشگاهی آنلاین می باشد که می توان اجناس را در آن به نمایش گذاشته یا خرید و فروش انجام داد.

امکانات اپلیکیشن :

کاربران قابلیت جستجو درباره ی جزئیات محصولات و ... را دارند.

کاربران قابلیت ثبت سفارش و پرداخت و جزئیات مربوط به آن را خواهند داشت.

مدیریت اطلاعات در مورد پرداخت، سفارش، shopping cart و

تکنولوژی های مورد استفاده :

در این اپلیکیشن از دیتابیس mySql بهره گیری می شود و به منظور اتصال به دیتابیس از ای پی آی (API) ، JDBC استفاده شده است.

در این پروژه از تکنولوژی های زیر بهره برده شده :

HTML (1

CSS (2

Java Script (3

Java (4

Apache Tomcat (5

چگونگی کارکرد برنامه :

در HomePage پروژه، لیستی از محصولات (کفش) وجود دارد و به منظور راحتی در انتخاب، سه دستی‌بندی با عناوین :

Safety Shoes , Sport Shoes , More Shoes ایجاد شده که هر کدام محصولی متنوع با اطلاعات مخصوص به خودشان را نمایش می‌دهند.

از امکانات دیگر قابل ذکر، می‌توان به :

1 (قابلیت جستجو محصولات براساس داده‌های اختصاصی خودشان

2) نمایش موجودی حساب

3) قابلیت Sign in & Sign up

4) آیکون Contact به منظور برقراری ارتباط مشتریان با فروشنده و

توضیحات فایل‌ها :

پوشه WebContent همانطور که از نامش مشخص است شامل فایل‌های css , js , jsp , html می‌باشد.

هر کدام از فایل‌های jsp در این پوشه با توجه به نامشان وظیفه‌ی دریافت و جمع‌آوری دیتاهایی که از سمت client وارد می‌گردد را برعهده دارند.

برای مثال ، `checkout.jsp` داده‌های کاربر به منظور فرآیند ثبت سفارش را دریافت و ذخیره می‌کند. سپس وظیفه راستی‌آزمایی‌های مورد نیاز در مورد دیتاها و ذخیره‌سازی در `db` و `Response` به `final.jsp` و اطلاع کاربر از پایان فرآیند، با `CheckoutController.java` خواهد بود.

و فایل‌های دیگر هم به همین منوال می‌باشند

پوشه `src` هم شامل پنج فایل هست که هر کدام یک وظیفه را بر عهده دارند.

فایل `CartController.java` همانطور که از نامش پیداست، وظیفه‌ی دریافت ، کنترل و تایید

`shopping cart` ها را بر عهده دارد. بدین شکل که ابتدا یک `extends` از `HttpServlet` ایجاد کرده،

سپس با دو پارامتر وارد شده‌ی `name` و `price` از محصول توسط کاربر ، به `welcome.jsp` ،

`Response` می‌دهیم. بعد از آن با تایید شدن ورودی، درخواست به `checkout.jsp` ارسال می‌گردد.

فایل `CheckoutController.java` به منظور بررسی صحت اطلاعات ورودی برای ثبت سفارش فعالیت

می‌کند به این شکل که، پارامترهای وارد شده از سمت کاربر از قبیل نام ، آدرس ، شهر و ... را چک می‌کند و اگر

هر کدام از آنها خالی باشند، کاربر اروری با محتوای `Please fill all the fields` دریافت خواهد کرد و اگر

اطلاعات کامل وارد شده باشد، آنگاه درایور پایگاه داده بارگذاری می‌گردد. سپس اجرای `Query` صورت گرفته و

با `add` کردن داده ها، `Order Successfully` نمایش و به `final.jsp` ، `Response` داده می‌شود.

فایل `ContactController.java` به منظور تایید و بررسی ورودی داده از سوی کاربر در منوی `Contact`

می‌باشد. عملاً همان مراحل رفته شده در `CheckoutController.java` چه در تایید و عدم تایید و چه در

مراحل ارتباط‌گیری با دیتابیس، صورت می‌گیرد و تفاوت‌ها، در نمایش پیام‌های ارور و `Response` ها به `jsp`

های مختلف می‌باشد.

فایل `LoginController.java` به منظور انجام رویه‌های لاگین کاربر می‌باشد. به طوری که با گرفتن `user` و `pass` از کاربر در فیلدها، درایور پایگاه داده را بارگذاری کرده و با اجرای `Query`، اگر صحت داده‌ها در `db` محرز شد، به `welcome.jsp`، `Response` می‌دهد در غیر این صورت : `! html error`

فایل `RegistrationController.java` نیز مسئول بررسی و تایید مراحل ثبت‌نامی است. بدین صورت که ابتدا پارامترهای وارد شده مانند نام، نام‌خ، یوزرنیم، ایمیل و... را چک می‌کند اگر موردی خالی باشد، ارور `please fill all the field` دریافت می‌گردد و اگر کامل بود که به فرآیند بارگذاری `db` و اجرای `Query` می‌پردازد و در اتمام کار پیام `successfully inserted` نمایش داده می‌گردد. و سپس به `employeedetails.jsp`، `Response` داده می‌شود.

امید است که کمی و کاستی پروژه را بر بیان ضعیف و فشردگی این برهه‌ی زمانی که درش قرار دارم و البته بزرگی جنابعالی ببخشید.

ایمان زراعتی

ورودی سال ۹۷

ترم ۱۰