



**دانشگاه بجنورد**  
**گروه مهندسی کامپیوتر**

**پایان نامه دوره کارشناسی مهندسی کامپیوتر**

**پایان نامه**

**پیاده سازی سایت فروشگاه لباس**

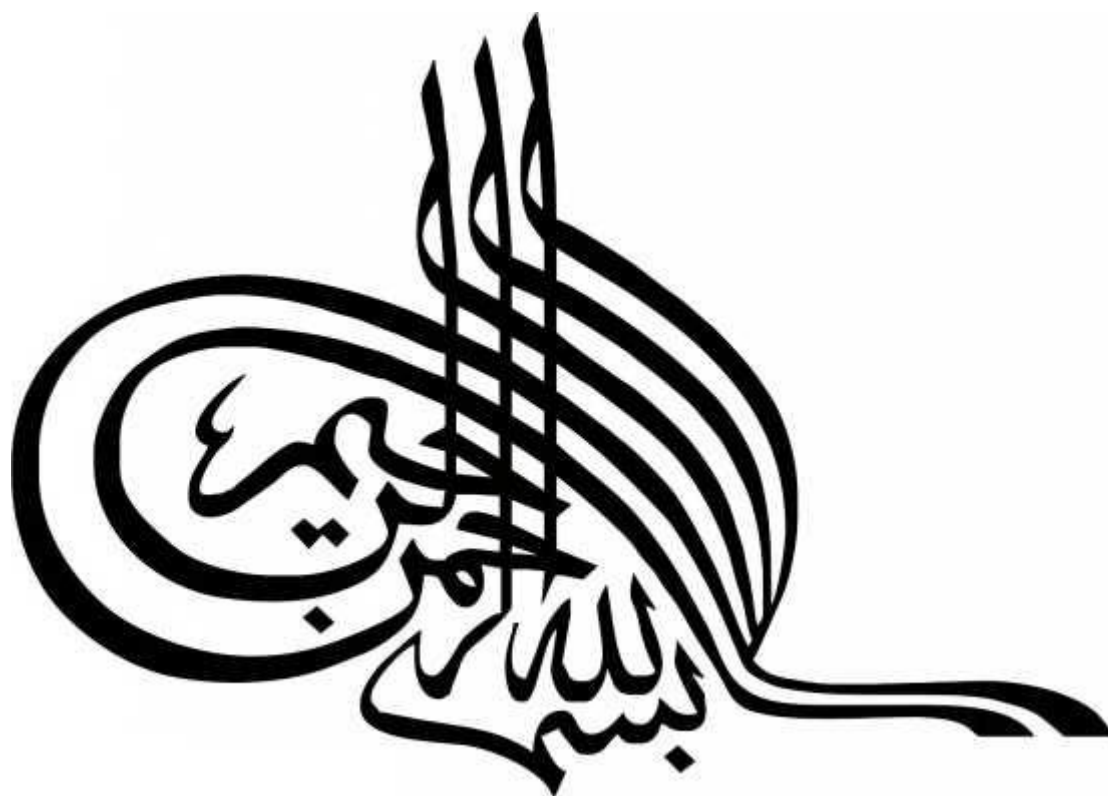
**دانشجو**

**محمد جعفری**

**استاد**

**دکتر وحید کیانی**

**تیر ۱۴۰۱**



## فهرست

۴.....	مقدمه.....
۵.....	وب سایت چیست ؟.....
۵.....	آدرس وب سایت یا URL چیست ؟.....
۶.....	مرورگر وب چیست ؟.....
۶.....	زبان ها و ابزار های استفاده شده .....
۱۰.....	وب سایت استاتیک .....
۱۱.....	وب سایت داینامیک.....
۱۱.....	نیازمندی های عملیاتی.....
۱۳.....	نمودار مورد کاربرد .....
۱۳.....	توضیح موارد کاربرد .....
۱۴.....	مراحل طراحی وب سایت .....
۱۴.....	چالش ها.....
۱۴.....	۱-responsive کردن بخش پیشنهاد ها.....
۱۶.....	۲-کشیده شدن تصاویر محصولات صفحه فیلتر(ناشی از پویا بودن ارتفاع بخش فیلتر ها).....
۱۶.....	۳-اعمال فیلتر ها روی محصولات.....
۱۷.....	توضیحات فایل های پروژه .....
۱۷.....	۱-Backend.....
۲۳.....	۲-Frontend.....
۵۳.....	راه اندازی اولیه .....

## مقدمه

با رشد تکنولوژی و حل بسیاری از مشکلات به شکلی راحت تر توسط کامپیوترها و شبکه‌های کامپیوتری، اینترنت محبوبیت بالایی بین مردم پیدا کرده است. افراد خیلی از کارهای خود را مانند خرید و فروش کالاها به صورت آنلاین انجام می‌دهند.

بازاریابی اینترنتی به فروشندگان کمک می‌کند کسب‌وکار خود را رشد دهند و به خریداران نیز کمک می‌کند تا خریدی مطمئن تر، به صرفه تر و سریع تر داشته باشند. مردم از اینترنت برای پیدا کردن اطلاعات گوناگونی استفاده می‌کنند. آن‌ها دائماً در حال جستجو هستند تا اطلاعات مربوط به کسب و کارها، محصولات و خدمات مدنظر خود را پیدا کنند.

فروشندگان نیز تمایل دارند برای پیشرفت کار خود به اطلاعاتی از مشتریان از جمله جایی که آن‌ها زندگی می‌کنند، محل کار آن‌ها و محل مورد نظر آن‌ها برای خرید دست پیدا کنند تا بتوانند سود بیشتری به دست آورند. همه‌ی این موارد و حتی اطلاعاتی بیشتر از این موارد در این محیط جمع شده‌اند: اینترنت. بازاریابی اینترنتی به صورت آنلاین برای کسب‌وکار افراد بسیار ارزشمند است زیرا به آن‌ها کمک می‌کند تا با مشتریانی احتمالی ارتباط برقرار کنند که به کسب‌وکارشان علاقمند هستند.

در تبلیغات سنتی، ارتباط به صورت یک طرفه بوده است. فروشندگان یک تبلیغ تلویزیونی و یا تبلیغ چاپی را در یک روزنامه قرار می‌دهند و صبر می‌کنند تا مخاطبین آن‌ها تبلیغاتشان را ببینند و به کسب‌وکار آن‌ها مراجعه کنند.

بنابراین، چه اتفاقی می‌افتد اگر مخاطبین آن‌ها سوالات یا ابهاماتی در خصوص این کسب و کار و نحوه‌ی تامین نیاز خود داشته باشند؟ چه می‌شود اگر این افراد بخواهند درباره کسب‌وکار، محصول و یا سرویس شما اطلاعات بیشتری کسب کنند؟ زمانی که یک مکالمه دوطرفه نباشد، این کار دشوار و هزینه‌بر است. اهمیت بازاریابی اینترنتی در توانایی ایجاد ارتباط دو طرفه نهفته است.

در بازاریابی اینترنتی فروشندگان به مشتریان راهی برای تماس با کسب‌وکار خود می‌دهند؛ هر زمان که یک پست در رسانه‌ای اجتماعی ایجاد می‌کنند، در حقیقت این فرصت را برای مردم ایجاد می‌کنند تا با کسب‌وکار آن‌ها ارتباط برقرار کنند. از این جهت می‌توان بازاریابی اینترنتی را روشی مطلوب هم برای خریدار و هم برای فروشنده دانست که منافع آن‌ها را بسیار بهتر از روش‌های سنتی تامین می‌کند. از این رو در این پروژه به طراحی وب سایت فروشگاه اینترنتی که یکی از عناوین اصلی بازاریابی اینترنتی محسوب می‌شود پرداخته شده است.

## وب سایت چیست ؟

یک وب سایت مجموعه‌ای از صفحات وب است و صفحات وب فایل‌هایی هستند که با استفاده از HTML ( زبان نشان‌گذاری گسترش یافته ) نوشته شده‌اند که در کامپیوترهایی به نام سرور ذخیره می‌شوند. هر سرور می‌تواند میزبان یک یا چندین وب سایت باشد و برای اینکه وب سایت‌ها از هم متمایز شوند یک شماره خاص برای هر وب سایت در نظر گرفته می‌شود که به این شماره IP می‌گویند.

## آدرس وب سایت یا URL چیست ؟

URL در واقع مخفف کلمات Universal Resource Locator است ، اگر به نوار آدرس مرورگر خود نگاه کنید، می‌بینید که URL نمایش داده می‌شود. این URL شامل تمام اطلاعات لازم برای یافتن صفحه مورد نظر در وب سایت است. URL تکنولوژی آدرس دهی در شبکه است. یک URL شامل موارد زیر می‌باشد:



**پروتکل:** پروتکل مجموعه‌ای از قوانین برای قالب‌بندی و پردازش داده‌ها است.

**زیر دامنه ( یا subdomain ):** بخشی از دامنه اصلی است که برای سازماندهی و هدایت به بخش‌های مختلف وب سایت‌ها ایجاد می‌شوند .

**دامنه:** اگر IP وب سایتی را در نوار مرورگر وارد کنید؛ وارد وب سایت مربوط به آن خواهید شد. اما از آنجایی که بخاطر آوردن IP ها و همچنین کار کردن با آن‌ها سخت است؛ بنابراین مفهومی به نام دامنه (Domain) ابداع شد.

**پسوندها:** پسوند است که نام دامنه را در یک آدرس وب دنبال می‌کند انواع مختلفی دارد که پرکاربردترین آن " .com " است. این پسوندها بر اساس نوع سایت تعریف میشوند، برای مثال اگر سایت ساخته شده مربوط به خدمات یک سازمان است بهتر است پسوند سایت را org (organization) انتخاب کنیم.

## مرورگر وب چیست ؟

به نرم‌افزاری گفته می‌شود که برای دریافت، نمایش و مرور اطلاعات از وب جهان‌گستر مورد استفاده قرار می‌گیرد. یک منبع اطلاعات با URL شناخته می‌شود، این منبع می‌تواند یک صفحه وب، تصویر، ویدئو و یا هر نوع اطلاعاتی دیگری باشد. هر چند مرورگرها برای دسترسی به وب جهان‌گستر ساخته شده‌اند، اما از آن‌ها می‌توان برای دسترسی به اطلاعات سرورهای وب در شبکه‌های خصوصی و یا پرونده‌ها در فایل سیستم‌ها هم استفاده کرد.

## زبان ها و ابزار های استفاده شده



### : HTML

HTML یک زبان نشانه گذاری و اسکلت اصلی تمامی سایت هایی است که در سراسر دنیا به وجود می آمده است. HTML یک زبان برنامه نویسی نیست و تنها برای به نمایش درآوردن محتوا در ساختار های از پیش تعیین شده ساخته شده است. HTML هیچ ظاهری ندارد و با استفاده از CSS ظاهر و شکل می گیرد.



### : CSS

CSS یک زبان طراحی صفحات وب برای ایجاد و ساخت مشخصات ظاهری اسناد و اطلاعات وب سایت می باشد. در کدنویسی با CSS می‌توان موارد ظاهری سایت مثل رنگ، فونت، تصاویر پس زمینه و ... بصورت دلخواه تغییر ایجاد کرد.

هدف از تولید این زبان جداسازی HTML از اطلاعات ظاهری مانند صفحه بندی، رنگ و سایز و نوع فونت می باشد. این جداسازی موجب افزایش سرعت در دسترسی به وب سایت، قابلیت طراحی چندین صفحه با یک فرمت یکسان و جلوگیری از پیچیدگی و انجام کارهای تکراری در طراحی وب سایت می گردد.



## : SASS

Sass به صورت خلاصه باید بگویم Sass یک پیش پردازنده برای زبان CSS است. با استفاده از Sass می توان متغیر ها را تعریف کرد و حتی از مفاهیمی همچون وراثت در CSS استفاده نمود. این کار باعث می شود کار خود را در طراحی سریعتر و آسان تر پیش ببریم.

کد های نوشته شده با این ابزار در مرورگر ها قابل درک و پردازش نمیباشد و دلیل این عدم درک هم این است که مرورگر ها تنها فایل های CSS را برای ظاهر صفحات میپذیرند.

درنتیجه ما به یک کامپایلر نیاز داریم تا کد های نوشته شده در Sass را به CSS تبدیل کند. نرم افزار های زیادی برای این کار وجود دارند که بعضی از آنها عبارتند از :

- Hammer
- CodeKit
- Live sass compiler



## : JAVASCRIPT

زبان جاوا اسکریپت سطح بالا، داینامیک، شی گرا است. با استفاده از این زبان می توانید صفحات وب مرتب و واکنش گرا ایجاد کرد. جاوا اسکریپت می تواند صفحات استاتیک وب سایت را به صفحاتی زیبا و داینامیک تبدیل کند. یکی دیگر از کاربرد های جاوا اسکریپت را می توان به طراحی اپلیکیشن های موبایل اشاره کرد که از فریم ورک های محبوب این حوزه میتوان react و angular را نام برد.



## : jQuery

jQuery یک کتابخانه سریع ، کوچک و دارای ویژگی های غنی از جاوا اسکریپت است. چند مورد از این ویژگی ها میتوان به Manipulation، مدیریت رویداد ها، انیمیشن و Ajax اشاره کرد. این کتابخانه با کمک از API آسان که در بسیاری از مرورگرها کار می کند ، استفاده از آن را بسیار ساده تر می کند. با ترکیبی از تطبیق پذیری و قابلیت گسترش ، jQuery شیوه کدنویسی جاوا اسکریپت میلیون ها نفر از برنامه نویسان را تغییر داده است.



## : BOOTSTRAP

بوت استرپ یک فریم ورک طراحی وب سایت است. این فریم ورک پس از مدتی در دسترس عموم قرار گرفت و به مجموعه ای مناسب از ابزارها برای توسعه رابط های کاربری مرورگر تبدیل شد. از بوت استرپ برای ایجاد و طراحی وب سایت های واکنش گرا استفاده می شود. قالب های Bootstrap نه تنها توسط توسعه دهندگان وب سایت بلکه توسط شرکت ها نیز استفاده می شود.



## : PHP

زبان PHP یک زبان شی گرا که به منظور مدیریت سرور توسعه یافته و شباهت زیادی به زبان C دارد. PHP یک زبان سمت سرور است یعنی کد های PHP ابتدا توسط سرور پردازش می شود و سپس خروجی به صورت کدهای HTML و جاوا اسکریپت برای مرورگر کاربر ارسال می شود. حدود ۲۴۰ میلیون وب سایت برای مدیریت سرور از زبان PHP بهره گرفته اند و در بیش از ۲ میلیون وب سرور مانند XAMPP مورد استفاده قرار گرفته است. از سایت های معروف دنیا که به زبان PHP کد نویسی شده اند می توان به ویکیپدیا و فیسبوک اشاره کرد. این زبان در نسخه ۴/۳ به بعد تغییرات زیادی کرد، یکی از این تغییرات اضافه کردن قابلیت واسط خط فرمان بود که این قابلیت می تواند برای ایجاد نرم افزارهایی با واسط گرافیکی کاربر و یا نرم افزار های غیر وب مورد استفاده قرار بگیرد.





## : XAMPP

XAMPP ، یک پکیج نرم‌افزاری است که خود حاوی چند نرم افزار دیگر است:

وب سرور Apache

پایگاه داده MySQL

مفسر زبان‌های PHP و Perl

علت ایجاد این پکیج نرم‌افزاری سهولت بخشیدن به نصب نرم افزار های نام برده به صورت یکپارچه است. XAMPP برای شبیه سازی localhost بر روی سیستم عامل مورد استفاده قرار می گیرد. از XAMPP برای نصب کردن، تست کردن، برنامه نویسی و اجرای نرم افزارهای وب استفاده می شود. به طور خلاصه میتوان گفت که این نرم افزار یک شبیه ساز وب سرور است.



## : MySQL

MySQL ، یک سیستم مدیریت پایگاه داده منبع باز به حساب می‌آید که برای مدیریت سیستم‌های پایگاه داده رابطه‌ای ایجاد شده است که توسط اوراکل (Oracle) پشتیبانی می‌شود. علاوه بر این، نرم افزار MySQL می‌تواند بر روی پلتفرم‌های مختلف یونیکس، لینوکس، ویندوز و بسیاری از سیستم عامل های دیگر اجرا شود. از ویژگی‌های مهم MySQL، قابلیت اطمینان، مقیاس پذیری و سریع بودن آن است.

## وب سایت استاتیک

وب سایت استاتیک (ایستا) از نخستین وب سایت هایی هستند که در شبکه جهانی اینترنت راه اندازی شدند. این وب سایت ها در برابر با وب سایت دینامیک بسیار کمتر مورد ویرایش قرار می گیرند. به عبارت دیگر وب سایت هایی هستند که در طول سال تنها چند بار مورد ویرایش قرار می گیرند.

برای ساخت این نوع وب سایت ها، اطلاعات و تمامی محتواهایی که باید در صفحات سایت نمایش داده شوند در ابتدا جمع آوری می شوند.

ساخت این وب سایت ها از نظر هزینه و مدت زمان ساخت به صرفه تر از وب سایت ها دینامیک است. اما برای اینکه کوچکترین تغییراتی در سایت اعمال کنید باید حتماً از طراح سایت خود آن وب سایت کمک بگیرید زیرا فقط آن ها از نحوه ساخت و معیار های آن خبر دارند که خود این موضوع باعث می شود هزینه های اضافه ای برای حفظ وب سایتتان پرداخت کنید.

امروزه دیگر افراد و شرکت ها به دنبال این نوع وب سایت نیستند (به جز ادارات که معمولاً جهت صرفه جویی در هزینه همچنان از این نوع وب سایت بهره می برند). چون صاحبان سایت ها بیشتر دوست دارند امکان ایجاد تغییرات، ویرایش و آپدیت نامحدود محتواهای موجود در سایت خود را داشته باشند.

### معایب وب سایت های استاتیک

- وابستگی همیشگی به برنامه نویس
- عدم بروزرسانی ماهانه یا چند ماهه
- از دست دادن رتبه بازدید گوگل
- تعداد صفحات محدود
- فاقد پنل مدیریتی

### مزایای وب سایت های استاتیک

- هزینه اولیه تولید پایین
- امنیت بالا
- سرعت لود بالا (زیرا صفحه های این سایت احتیاجی به درخواست سرور ندارند).
- سرعت تولید بالا

## وب سایت داینامیک

این نوع وبسایت ها چون به طور مداوم در حال تغییر و به روزرسانی هستند داینامیک یا پویا نام گرفته اند. هزینه تولید اولیه وب سایت داینامیک تقریباً بالاست اما ویرایش این سایت به خاطر داشتن پنل مدیریتی محتوا بسیار راحت تر است. طراحی کردن سایت داینامیک زمان بر است اما بعد از به پایان رسیدن ساخت وب سایت، شما مراجعات بسیار کمی جهت تغییر در وب سایت به برنامه نویس دارید. صفحات وب سایت استاتیک محتوا ثابتی دارند اما صفحات سایت داینامیک با توجه به زمان اجرای آنها تغییر می کنند.

به راحتی نمی توان محتوای صفحات وب سایت استاتیک را تغییر داد چون برای اینکار باید یک صفحه جدید ساخت و آن را دائلود کرد اما در سایت های داینامیک این کار در یک برنامه سرور انجام می شود.

## نیازمندی های عملیاتی

### ۱- موبایل ریسپانسیو:

- در عصر امروز، بیشترین تعداد کاربران برای وب سایت از طریق دستگاه های تلفن های هوشمند وارد می شوند و در گذشت زمان این تعداد رو به افزایش است.

### ۲- اشتراک گذاری در شبکه های اجتماعی:

- حضور در پلتفرم های رسانه های اجتماعی برای تجارت الکترونیک بسیار مهم است. ایجاد امکان اشتراک گذاری محتوای وب توسط کاربر باعث می شود وب سایت در پلتفرم رسانه های اجتماعی، آگاهی از برند فروشگاه را افزایش می دهد.

### ۳- چت زنده (live chat):

- چت آنلاین نکته ای است که باید در میان الزامات مدیریت ضروری به آن توجه کرد. مهم نیست که محصول شما چقدر ساده است، بسیار مطلوب است که یک چت آنلاین در سایت داشته باشید که در آن مشتریان بتوانند بلافاصله پاسخ سوالات خود را دریافت کنند.

### ۴- امکان فیلتر محصولات:

- کاربر بتواند با استفاده از گزینه های فیلتر و دسته بندی محصولات قرار داده شده در سایت محصولات را فیلتر کند تا به محصول مد نظر خود برسد.

- سیستم منطقی داخلی فیلترها به وبسایت این امکان را می دهد که بازه انتخاب را به حداقل برساند و هنگامی که مشتریان به دنبال محصول مورد نظرشان هستند، سردرگم و یا خسته نشوند و در نتیجه در دراز مدت، مشتری ها خریدهای بیشتری را برای این وبسایت به ارمغان می آورد.

#### ۵- ورود و خروج از حساب کاربری:

- کاربر یا طراح نظر سنجی باید بتواند وارد حساب کاربری خود شود تا با توجه به سطح دسترسی خود از سیستم استفاده نمایند همچنین از حساب خود در صورت نیاز خارج شود.

#### ۶- ایجاد حساب کاربری:

- کاربر در صورت نداشتن حساب قبلی می تواند حساب جدیدی بسازد و پس از تایید وارد حساب خود شود .

#### ۷- سبد خرید:

- کاربر این امکان را داشته باشد که محصولات خود را به سبد خرید اضافه کند تا بتواند تمامی محصولات مورد نظر خود را به صورت یکجا سفارش دهد.

#### ۸- پیگیری محصول خریداری شده:

- کاربر در صورت نیاز میتواند با وارد کردن کد پیگیری محصول در سایت، مشاهده کند که محصول سفارش داده شده ولی در چه مرحله ای از تحویل است.

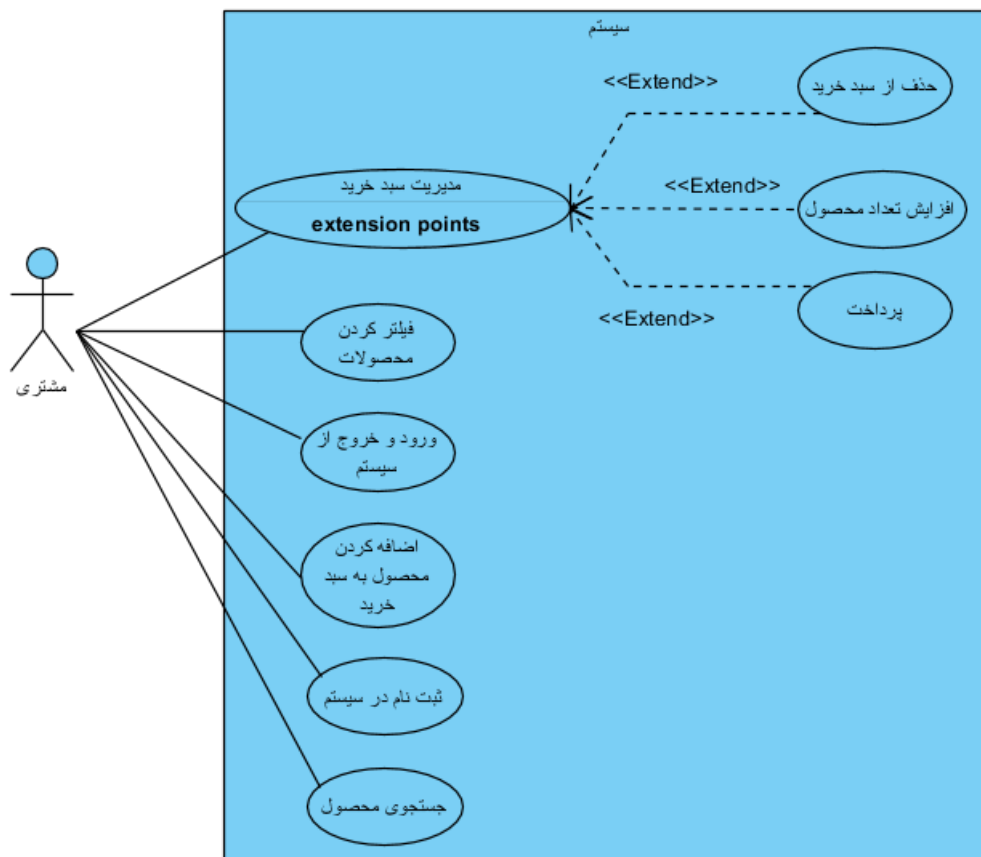
#### ۹- پرداخت:

- کاربر بعد از پیدا کردن محصول مورد نظر و اضافه کردن آن به سبد خرید با زدن روی دکمه پرداخت به صفحه پرداخت بانکی معتبر ارجاع داده شود و پرداخت هزینه محاسبه شده را انجام دهد.

#### ۱۰- دریافت کد پیگیری:

- کاربر بعد از پرداخت هزینه محاسبه شده کدی جهت پیگیری محصول خریداری شده دریافت می کند.

## نمودار مورد کاربرد



## توضیح موارد کاربرد

**ورود و خروج از سیستم :** کاربر بتواند با وارد کردن ایمیل و رمز عبور خود و کلیک روی گزینه ورود وارد حساب کاربری خود شود و در صورت نیاز با کلیک روی دکمه خروج از حساب کاربری از حساب خود خارج شود.

**ثبت نام در سیستم :** کاربر بتواند با کلیک روی دکمه ثبت نام وارد صفحه ثبت اطلاعات ثبت نام شده و با وارد کردن مشخصات (نام ، نام خانوادگی و ...) حساب کاربری مربوط به خود را ایجاد کند.

**اضافه کردن محصول به سبد خرید :** کاربر بتواند با کلیک روی گزینه "اضافه کردن به سبد خرید" در صفحه مربوط به محصول، محصول مورد نظر را به صفحه سبد خرید اضافه کند.

**جستجوی محصول :** کاربر بتواند با کلیک روی باکس جستجوی و وارد کردن محصول مدنظر، آن را پیدا کند.

**فیلتر کردن محصول :** کاربر بتواند با انتخاب پارامتر های مربوط به محصول مدنظر، محصولات فیلتر شده را مشاهده کند.

**مدیریت سبد خرید :** کاربر بتواند محصولات موجود در سبد خرید را حذف و یا تعداد آن ها را بیشتر کند و بعد از محاسبه شدن هزینه و کلیک روی گزینه پرداخت به روند پرداخت هزینه محصول بپردازد.

## مراحل طراحی وب سایت

در ابتدا وب سایت های مربوط به فروشگاه های اینترنتی را بررسی میکنیم تا به قالب کلی و اجزاء مختلف وب سایت پی ببریم. سپس یک قالب فرضی در نظر میگیریم و شروع به ساخت وب سایت میکنیم. نخست برای ایجاد اجزاء هر یک از صفحات فایل html ایجاد میکنیم و کد های مربوط به هریک از اجزاء را در آن مینویسیم. حال برای ایجاد یک ساختار منظم و زیبا از CSS بهره میگیریم و برای هر یک از اجزاء فایل html ، style تعریف میکنیم. علاوه بر این میتوانیم از فریم ورک sass استفاده کنیم که به ما امکاناتی مثل تعریف حلقه و ... میدهد و به ما کمک می کند تا style اجزاء html را آسان تر و بهتر تعیین کنیم.

برای تنظیم عملکرد اجزاء html از JavaScript استفاده می کنیم و در برخی از قسمت های وب سایت که نیاز به ذخیره سازی و پردازش داده ها در سمت سرور دارند باید از ajax استفاده میکنیم تا داده ها را از سمت مرورگر کاربر به سمت سرور ارسال کنیم تا بتوانیم از php برای مدیریت این اطلاعات استفاده کنیم مانند : ورود و خروج از حساب کاربری و نمایش محصولات به مشتری از داخل پایگاه داده جهت انجام اینکار باید از زبان sql استفاده کنیم تا بتوانیم جداول مربوط به محصولات و حساب های کاربری را ایجاد کنیم

## چالش ها

### ۱- responsive کردن بخش پیشنهاد ها

در بخش پیشنهاد ها که شامل ۷ محصول می شود (در اندازه صفحه دسکتاپ) لازم است که در هنگام کوچک تر شدن عرض صفحه چیدمان و تعداد محصولات موجود در این بخش تغییر پیدا کند که با استفاده از CSS grid و media tag این کار انجام می شود، به این صورت که در ابتدا به تمامی محصولات پیشنهادی ساختار و ظاهر یکسان می دهیم.

```

<section class="mt-5 w-100 ps-3 pe-3" id="offer-container">
  <section id="offer-header" ...>
    <section id="offer-main" dir="rtl">
      <div class="offers-box p-1" id="offer-1" ...>
      <div class="offers-box p-1" id="offer-2" ...>
      <div class="offers-box p-1" id="offer-3" ...>
      <div class="offers-box p-1" id="offer-4" ...>
      <div class="offers-box p-1" id="offer-5" ...>
      <div class="offers-box p-1" id="offer-6" ...>
      <div class="offers-box p-1" id="offer-7" ...>
    </section>
  </section>
</section>

```

سپس با استفاده از media tag و grid در اندازه‌های متفاوت عرض صفحه چینش محصولات پیشنهادی را تغییر می‌دهیم.

در سایز موبایل برای id="offer-main" شش ستون و شش ردیف تعریف شده که هر یک از محصولات سه ستون و یک ردیف را به خود اختصاص داده‌اند، با توجه به تعداد ستون‌های المنت پدر هر ردیف به دو محصول اختصاص یافته.

```

#offer-main {
  display: grid;
  grid-column: 1 / 7;
  grid-row: 2 / 7;
  grid-template: 40px repeat(6, 1fr) / repeat(6, 1fr);
}

```

Repeat(6, 1fr) : این متد شش ردیف با اندازه‌های یکسان برای المنتی که آن را فراخوانی کرده ایجاد می‌کند.

```

#offer-1 {
  display: grid;
  grid-column: 1 / 4;
  grid-row: 2 / 4;
}

```

از ستون اول تا ابتدای ستون چهار و همچنین از ابتدای ردیف دوم تا ابتدای ردیف چهارم به پیشنهاد اول اختصاص می‌یابد.

در سائز تبلت این ساختار به صورت شش ستون و هشت ردیف است که با اختصاص دادن این ساختار به محصولات پیشنهادی، شش محصول به صورت دو ردیف و سه ستون نمایش داده می‌شوند.

## ۲- کشیده شدن تصاویر محصولات صفحه فیلتر (ناشی از پویا بودن ارتفاع بخش فیلتر ها)

در صفحه فیلتر ها در صورت استفاده از ساختار مرسوم تقسیم صفحه به سه بخش header, main, footer محصولات و فیلتر ها توسط المان main احاطه می شدند که این امر باعث می شد با تغییر سائز بخش فیلتر ها (ناشی از کلیک کاربر روی هر سربرگ فیلتر) بخش محصولات هم به نسبت فیلتر ها تغییر اندازه پیدا کند. این رخداد ناشی از این است که در CSS grid ابعاد المان ها به صورت نسبی داده می شوند و لذا با تغییر ارتفاع بخش فیلتر ها، ارتفاع والد (main) و بخش محصولات تغییر پیدا می کند. راه حلی که برای رفع این مشکل در نظر گرفته شد، ایجاد دو والد مجزا (main و main2) برای دو بخش فیلتر ها و محصولات بود.

## ۳- اعمال فیلتر ها روی محصولات

با انتخاب هر یک از فیلتر ها ورودی های انتخاب شده با استفاده از ajax سمت سرور ارسال می شوند و در آنجا با استفاده از PHP فیلتر های مورد نظر بر روی پایگاه داده اعمال می شوند. در نهایت، محصولات فیلتر شده برای مرورگر کاربر ارسال می شوند.



## توضیحات فایل های پروژه

### Backend - ۱

#### ۱-۱-۱ Php

```
session_start();
$servername = "localhost";
$username = "star";
$password = "shopmestar";
$dbname = "star";
$pdo = new
PDO("mysql:host=$servername;dbname=$dbname;charset=utf8mb4",
$username, $password);
$pdo->setAttribute(PDO::ATTR_ERRMODE, PDO::ERRMODE_EXCEPTION);
```

در ابتدا جهت اتصال به پایگاه داده درخواستی به وب سرور ارسال می کنیم که درون این درخواست نام کاربر، رمز عبور و نام پایگاه داده را قرار می دهیم تا در صورت درست بودن و وجود چنین کاربری امکان دسترسی به جداول پایگاه داده به ما داده شود. این کار به این علت انجام می شود که کاربر عادی نتواند به اطلاعات درون پایگاه داده دسترسی پیدا کند.

```
if (isset($_POST['sub'])) {
    if($_POST['sub']== "ثبت نام"){
        $_name = $_POST['name'];
        $_lname = $_POST['lname'];
        $_email = $_POST['email'];
        $_pass1 = $_POST['pass1'];
        $_pass2 = $_POST['pass2'];
        $_mob = $_POST['mob'];
        $_address = $_POST['address1'] . " - " . $_POST['address2'];
        $_sql = "INSERT INTO users ( `fname`,`lname`,`email`,`pass`,`mob`,`address`,`credit`,`state` ) VALUES(?,?,?,?,?,?,?)";
        $_pdo->prepare($_sql)->execute([$_name, $_lname, $_email, $_pass1, $_mob,
        $_address, "0", "1"]);
        setlogin($_email);
        echo '1';
    }
}
```

در صورت فراخوانی این متد از سوی کلاینت (\$\_POST['sub'] مساوی ثبت نام) این متد مقادیر نام، فامیل و ... را دریافت و هر یک را به متغیر مربوط به خود اختصاص می دهد سپس درون پایگاه داده در جدول users اطلاعات کاربر را ثبت می کند(insert into users).

```
if ($_POST['sub'] == "ورود") {
    $_email = $_POST['email'];
    $_pass = $_POST['pass'];
```

```

$sql = "SELECT * from users where `email`=:email and `pass`=:pass ";
$stmt = $pdo->prepare($sql);
$stmt->bindParam(':email', $email, PDO::PARAM_STR);
$stmt->bindParam(':pass', $pass, PDO::PARAM_STR);
$stmt->execute();
$row = $stmt->fetch(PDO::FETCH_ASSOC);
if (!$row) {
    echo '0';
} else {
    setlogin($email);
    echo '1';
}
}
}

```

در صورت فراخوانی این متد از سوی کلاینت (\$\_POST['sub'] مساوی ورود) پس از دریافت مقادیر ایمیل و رمز عبور از سوی کاربر، تمامی کاربر های موجود در جدول users انتخاب شده و درون متغیر sql ریخته می شود سپس با ایمیل و رمز عبور تمامی کاربر ها مقایسه می شود. در صورت وجود چنین کاربری خروجی عدد یک برگردانده می شود و همچنین متد setlogin جهت ثبت و حفظ ورود کاربر فراخوانی می شود.

```

if (isset($_GET['islogin'])) {
    if (@$_SESSION['HTTP_USER_AGENT'] !=
md5($_SERVER['HTTP_USER_AGENT']) || @$_SESSION['HTTP_USER'] !=
md5(@$_SESSION['HTTP_USER_STAMP']) || $_COOKIE["login"] !=
md5("myhash")) {
        echo '0';
    } else {
        echo $_SESSION['HTTP_USER_NAME'];
    }
}
}

```

در قطعه کد بالا بررسی می شود که آیا کاربر وارد حساب کاربری خود شده یا خیر در صورتی که cookie ها در متد setlogin ثبت نشده باشند به یعنی کاربر وارد حساب خود نشده و در نتیجه عدد صفر بازگردانده خواهد شد و در غیر این صورت عدد یک به عنوان خروجی چاپ می شود.

```

if (isset($_GET['filter'])) {
    $sqlfilt = "";
    if (@$_GET['ferioli'] == 0) {
        $sqlfilt.= " and `brand` not like 'فریولی(ferioli)' ";
    }
    if (@$_GET['massimi'] == 0) {
        $sqlfilt.= " and `brand` not like 'ماسیمو دوتی(massimi dutti)' ";
    }
    if (@$_GET['fred'] == 0) {

```

```

$sqlfilt.= " and `brand` not like 'فرد(fred)' ";
}
if (@$_GET['black'] == 0) {
    $sqlfilt.= " and `color` not like 'مشکی' ";
}

```

در صورت set شدن filterها مورد به مورد ورودی های radio و check box را بررسی می کند، در صورتی که گزینه ها تیک خورده باشند عدد یک و در غیر این صورت عدد صفر را به خود اختصاص داده اند و کافیسست بررسی کنیم که این عدد برای هر فیلتر صفر است یا یک در صورت یک بودن از کلمه like که یکی از دستورات مربوط به زبان mysql است استفاده می کنیم به این معنی که این مقدار باشد و در صورت صفر بود دستور not like را استفاده می کنیم که تمامی این ها به صورت یک رشته در آمده و به متغیر sqlfilt اختصاص میابد و در هر شرط به انتهای آن رشته ای دیگر اضافه می شود.

```

$sql = "SELECT * from products where `id`>0 $sqlfilt";
$stmt = $pdo->prepare($sql);
$stmt->execute();
$row = $stmt->fetchAll(PDO::FETCH_ASSOC);
$json = json_encode($row);
echo $json;

```

حال به پایگاه داده ، جدول محصولات مراجعه می کنیم و تمامی محصولاتی که در sqlfilt قرار داده شده بود و همچنین id آن ها یک است را واکنشی می کنیم و با استفاده از ajax به سمت مرورگر کاربر ارسال می کنیم.

تابع setLogin() :

```

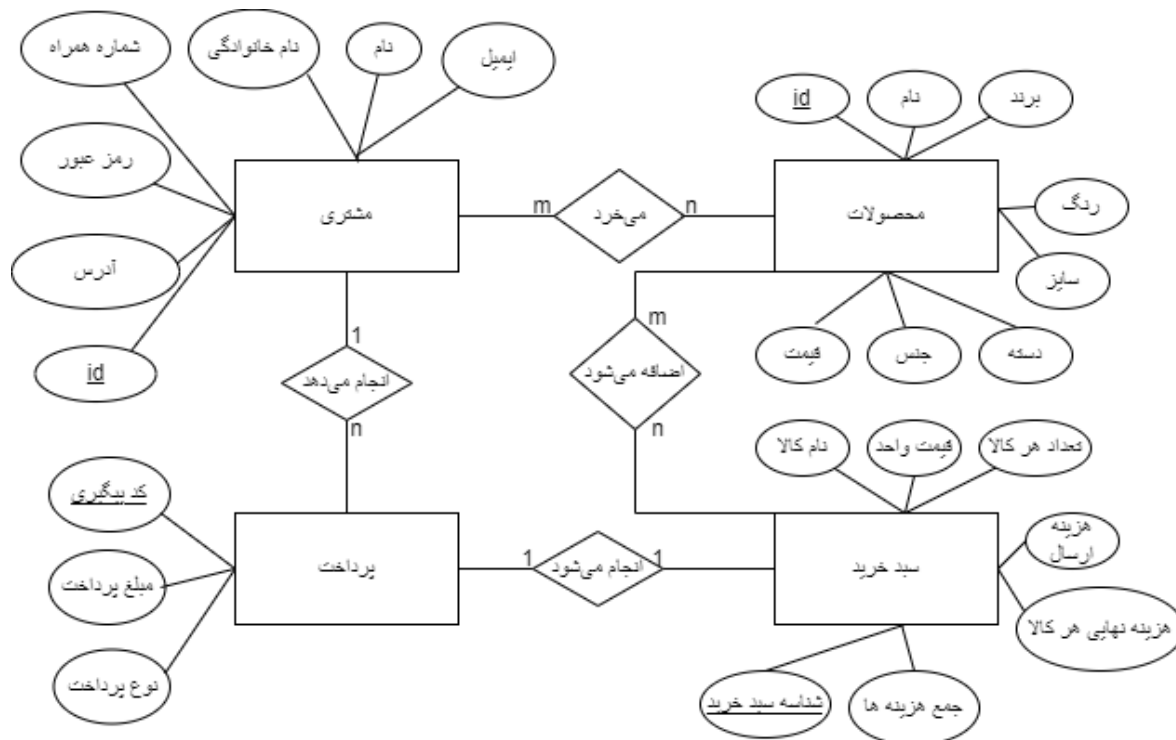
function setlogin($username){
    setcookie("login", md5("myhash"), time() + 655500);
    $_SESSION['HTTP_USER_AGENT'] = md5($_SERVER['HTTP_USER_AGENT']);
    $_SESSION['HTTP_USER_STAMP'] = time();
    $_SESSION['HTTP_USER'] = md5($_SESSION['HTTP_USER_STAMP']);
    $_SESSION['HTTP_USER_NAME'] = $username;
}

```

Cookie با نام login برای یک کاراکتر دلخواه set می کند که برای ۶۵۵۵۰۰ ثانیه فعال باشد ، حال user agent کاربر را به صورت رمز شده جایگزین مقدار قبلی می کند سپس زمان ورود را درون user stamp ریخته و رمز شده این مقدار را هم درون متغیر user قرار می دهیم. در صورت فعال بودن این اطلاعات کاربر login است در غیر این صورت کاربر login نیست. این رمز گذاری مانع از حمله hijacking می شود.

## ۲-۱ - Sql

نمودار ER پایگاه داده



Query های این فایل (shop\_moh.sql) برای ایجاد ساختار جداول مربوط به sql است با اجرا شدن این کد ها در محیط sql دو جدول ایجاد می شود (جدول کاربران و محصولات)

```
CREATE TABLE IF NOT EXISTS `products` (
  `id` int(11) NOT NULL AUTO_INCREMENT,
  `name` varchar(100) NOT NULL,
  `brand` varchar(100) NOT NULL,
  `color` varchar(100) NOT NULL,
  `size` varchar(250) NOT NULL,
  `material` varchar(100) NOT NULL,
  `price` int(11) unsigned NOT NULL,
  `cat` varchar(100) NOT NULL,
  PRIMARY KEY (`id`)
```

```
) ENGINE=MyISAM DEFAULT CHARSET=utf8 AUTO_INCREMENT=6 ;
```

در صورت عدم وجود جدول products آن را ایجاد میکند. در این جدول ستون‌های ورودی اطلاعات شامل (کد محصول) id، (نام محصول) name، (برند محصول) brand و ... می‌شوند.

کد محصول به عنوان کلید اصلی در این جدول انتخاب می‌شود به این معنی که ورودی‌های این فیلد در جدول پایگاه داده باید منحصر به فرد باشند.

فیلدی با ویژگی PRIMARY KEY ویژگی UNIQUE را نیز شامل می‌شود. و مقادیر NULL را نمی‌پذیرد.

یک جدول در پایگاه داده ی SQL، فقط می‌تواند یک فیلد PRIMARY KEY داشته باشد.

Auto-increment : به ازای وارد شدن هر ردیف جدید در جدول، عددی غیر تکراری و خاص را برای آن می‌سازد. اکثر اوقات این قابلیت را برای ستونی قرار می‌دهیم که primary key ما را داشته باشد تا برای هر ردیف جدید یک primary key خاص تولید شود.

```
INSERT INTO `products` (`id`,`name`,`brand`,`color`,`size`,`material`,`price`,`cat`) VALUES
(1,'مردانه پیراهن','feriolti','سفید','-L--M--S-','نخی',671500,'مردانه پیراهن'),
(2,'نخی','massimi dutti','سفید','-XL--XXL--M-','دوتی ماسیمو','دوتی ماسیمو سفید پیراهن',534000,'مردانه پیراهن'),
(3,'پیراهن','استر پلی','-XL--XXL--L--M-','مشکی','fred','فرد','فرد مشکی پیراهن',398000,'مردانه پیراهن'),
(4,'نخی','-XL--XXL--M--L-','مشکی','feriolti','فریولی','فریولی مشکی پیراهن',671500,'مردانه پیراهن'),
(5,'پیراهن','-XL--XXL--M-','مشکی','feriolti','فریولی','فریولی مشکی پیراهن',416500,'مردانه پیراهن');
```

حال با اطلاعات از پیش آماده شده محصولات، جدول محصولات را پر میکنیم.

```
CREATE TABLE IF NOT EXISTS `users` (
  `id` int(11) unsigned NOT NULL AUTO_INCREMENT,
  `fname` varchar(60) NOT NULL,
  `lname` varchar(70) NOT NULL,
  `email` varchar(70) NOT NULL,
```

```
` pass` varchar(100) NOT NULL,  
` mob` varchar(11) NOT NULL,  
` address` varchar(150) NOT NULL,  
` credit` int(11) NOT NULL,  
` state` int(11) NOT NULL,  
  
PRIMARY KEY (`id`)  
(ENGINE=MyISAM DEFAULT CHARSET=utf8 AUTO_INCREMENT=1;
```

در صورت وجود نداشتن جدول users این جدول ایجاد می‌شود ستون هایی که برای آن باید ساخت id، fname و ...

Not null یعنی هیچ یک از ستون ها ورودیشان نباید خالی باشد

Varchar() یک رشته دریافت میکند با تعداد مشخص حداکثر مقداری که داخل پرانتز برای این متد قرار داده شود

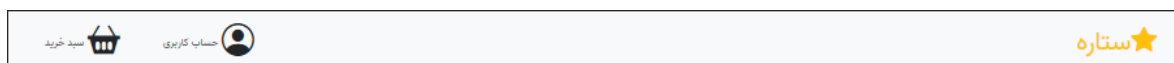
Id به عنوان کلید اصلی برای این جدول در نظر گرفته شده

## ۲- Frontend

### HTML&CSS

#### : Header

- مارک فروشگاه
- logo حساب کاربری برای ورود و ثبت نام در سایت
- logo سبد خرید برای هدایت به صفحه سبد خرید



```
<header>
<nav class="navbar navbar-light fixed-top p-3 pt-2 pb-0">
  <div class="container-fluid justify-content-center">
    <div class="full-width m-0">
      <div class="row">
        <div id="item-bar" class="col-3 p-0">
          <div id="brand" class="d-flex justify-content-center">
            <div id="hamburger-menu" class=" justify-content-end align-items-center p-0">
              <div class="offcanvas offcanvas-start">
```

کلاس navbar از کلاس های bootstrap است که یک ردیف به اندازه عرض صفحه نمایش اشغال می کند و با استفاده از کلاس navbar-light به این ردیف یک رنگ پس زمینه خاکستری روشن می دهیم.

کلاس fixed-top ، position المنت مد نظر را مقدار fixed قرار می دهد و فاصله این المنت را از بالا، چپ و راست صفحه مقدار صفر در نظر میگیرد.

P یکی دیگر از کلاس های bootstrap است، این کلاس باعث ایجاد padding برای المنتی که آن را فراخوانی کرده می شود. (pt مخفف padding-top است)

Container-fluid المنت را به عنوان یک container تعریف میکند که باعث ایجاد padding در سمت راست و چپ المنت می شود.

Justify-content-center المنت را با فاصله یکسان از سمت راست و چپ قرار میدهد تا المنت مورد نظر در مرکز المنت پدر خود قرار بگیرد.

Full-width به المنتی که آن را فراخوانی کرده عرض کامل صفحه را اختصاص میدهد.

m-0 به این معنی است که margin المنت مورد نظر صفر در نظر گرفته شود.

المنت brand که درون آن icon های ورود به حساب کاربری و سبد خرید قرار داده شده است کلاس justify content center را دارا است که باعث می شود این المنت در مرکز header ما قرار بگیرد (در اندازه موبایل این style اعمال می شود در اندازه تبلت این کلاس با کلاس justify content end عوض می شود)

المنت hamburger menu که درون آن navbar ای قرار دارد که شامل دسته بندی محصولات و باکس جستجو می باشد دارای کلاس justify content end است که باعث می شود این المنت فاصله اش از سمت راست المنت پدر خود صفر شود.

## : Nav

دکمه dropDown برای انتخاب  
Search box برای جستجوی لباس



```
<div class="container-fluid pe-3 ps-3">

<form class="d-flex" lang="fa" dir="rtl">

<button class="btn btn-outline-success white-font" type="submit">

جستجو
</button>
<input
class="form-control ms-2"
type="search">
```



```
placeholder="کنید جستجو"  
aria-label="Search"  
</>
```

در المنت nav فرم ورودی برای جستجوی کاربر ایجاد شده که شامل یک دکمه برای اعمال جستجوی کاربر می‌شود.

Placeholder یک ویژگی برای تگ input است که در این قطعه کد وظیفه دارد در پس زمینه input، جمله "جستجو کنید" را نمایش دهد.

کلاس btn-outline-success اطراف دکمه جستجو را سبز پر رنگ می‌کند.

```
<div class="dropdown">  
  <button class="btn btn-secondary dropdown-toggle white-font"  
    type="button" id="dropdownMenuButton2"  
    data-bs-toggle="dropdown"  
    aria-expanded="false" >  
    بیچگانه  
  </button>
```

به المنت پدر کلاس dropdown داده شده که باعث می‌شود در صورت کلیک روی دکمه‌های با id=dropdownMenuButton2، منو تعیین شده نمایش داده شود. Btn-secondary رنگ دکمه را خاکستری می‌کند.

```
<ul class="dropdown-menu dropdown-menu-light"  
  aria-labelledby="dropdownMenuButton2">  
  <div class="pbox">  
    <p class="ms-2 text-danger">دخترانه</p>  
  </div>  
  <li>  
    <a class="dropdown-item" href="/filter.html">بلوز و پیراهن</a>  
  </li>
```

همانطور که مشاهده می‌کنید aria-labelledby برای این المنت با id المنت دکمه یکسان است. Text-danger رنگ نشسته را قرمز می‌کند.

به tag a کلاس dropdown-item داده شده به این معنا که زیر مجموعه (فرزند) المنتی با کلاس dropdown menu است.

## Slider

- اسلایدر های تبلیغاتی با قابلیت عوض شدن خودکار.
- در این بخش عکس های تبلیغاتی مهم تر قرار داده می شوند.

```
<div class="d-flex full-width mt-5">
  <div class="full-width ps-3 pe-3" id="grid-slide-container">
    <div id="poster-box">
      <div id="poster-top" class="">
        
      </div>
      <div class="poster-bot">
        
      </div>
    </div>
  </div>
</div>
```

```
#grid-slide-container {
  display: grid;
  grid-template: repeat(6, 1fr) / repeat(6, 1fr);
  gap: .5rem 0;
  #carouselExampleIndicators {
    display: grid;
    grid-column: 1 / 7;

    grid-row: 1 / 3;
  }
  #poster-box {
    display: grid;
    grid-column: 1 / 7;

    grid-row: 3 / 7;
    gap: .5rem 0;
  }
}
```

برای `id = grid-slide-container` ، `display` مقدار `grid` گرفته است و به عنوان قالب شش ستون و شش ردیف ایجاد شده.

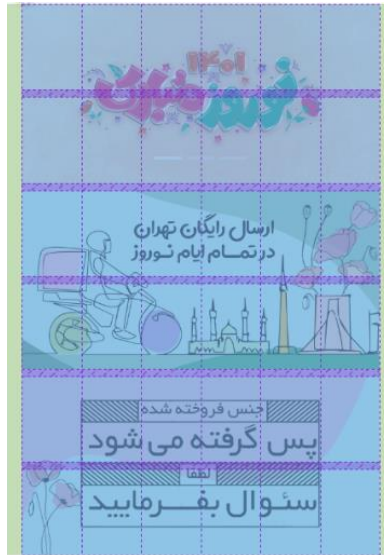
`Gap` فاصله بین ستون ها و ردیف ها است که مقدار اولیه آن مربوط به ردیف ها و مقدار دوم مربوط به فاصله بین ستون ها است.

`Rem` واحدی است که معادل `16px` (سایز فونت نرمال browser) است و عددی که قبل آن نوشته می شود در این عدد ضرب می شود. از این واحد در طراحی صفحات `responsive` استفاده می شود و

علت آن این است که این واحد نسبی است و به نسبت بزرگ تر شدن فونت پیش فرض صفحه تغییر می کند.

Em این واحد به جای اینکه از سایز فونت browser استفاده کند از سایز فونت والد خود استفاده می کند.

خروجی کد :



همانطور که در تصویر مشاهده می کنید شش ستون و شش ردیف به عنوان قالب برای المنت -grid slide-container ایجاد شده که بین هر ردیف 0.5rem (معادل 8px) ایجاد شده است. ps و pe : padding از سمت راست و چپ صفحه ایجاد کرده اند. هر چه رقم بعد از این حروف بیشتر باشد (حداکثر تا ۵) فاصله اعمال شده بیشتر می شود.

به هر پوستر دو ردیف اختصاص یافته که به ترتیب از ردیف یک تا ابتدای سه به المنت carouselExampleIndicators تعلق می گیرد و از ردیف ۳ تا ابتدای ردیف ۷ به المنت -poster box تعلق می گیرد.

### محصولات پیشنهادی

- نمایش محصولات تخفیف دار یا پر فروش در سه تفکیک زنانه مردانه و بچگانه

```
<section id="offer-main">
  <div id="offer-nav">
    <div id="item1">
      <a class="text-decoration-none active-1"
href="javascript:void(0)">مردانه</a>
    </div>
```

```

#offer-main {
  display: grid;
  grid-column: 1 / 7;
  grid-row: 2 / 7;
  grid-template: 40px repeat(6, 1fr) / repeat(6, 1fr);
  gap: .3rem .3rem;
  #offer-nav {
    text-align: center;
    display: grid;
    grid-column: 1 / 7;
    grid-row: 1 / 2;
    grid-template: 1fr / repeat(9, 1fr);
    @for $i from 1 through 3 {
      #item#{$i} {
        $i: $i + ($i - 1);
        display: grid;
        grid-column: #{$i} / #{$i+2};
        grid-row: 1 / 2;
      }
    }
  }
}

```



خروجی کد :

المنت offer-main یک قالب شش ستون و هفت ردیف ایجاد کرده است که ردیف اول نسبت به ردیف‌های دیگر طول کمتری دارد که به المنت offer-nav اختصاص یافته این المنت شامل نه ستون و یک ردیف است که هر یک از دکمه‌های مردانه، زنانه، بچه گانه سه ستون از این المنت را به خود اختصاص داده‌اند.

حلقه for نوشته شده برای همین دکمه‌ها هست که روش عملکرد این حلقه به این شکل است که یک شمارنده (i) از عدد یک تا سه را پیمایش می‌کند و این اندیس برای المنت‌های item با توجه به فرمول  $i: \$i + (\$i - 1);$  مقادیر ۱، ۳ و ۵ را تولید می‌کند. با این روش دیگر نیاز نیست تک تک این المنت‌ها را انتخاب و کدهای تکراری برای آن‌ها بنویسیم.

## Second poster

- نمایش پوسترهای تبلیغاتی ثابت بدون قابلیت عوض شدن خودکار
- در این بخش عکس‌های تبلیغاتی کم اهمیت تر از پوسترهای اول سایت قرار داده می‌شود.



```
<div class="mt-5 w-100 ps-3 pe-3" id="second-poster-container" dir="rtl">
  <div id="seconde-poster-top" class="second-poster">
    
  </div>
  <div id="seconde-poster-bottom" class="second-poster">
    
  </div>
</div>
```

```
#second-poster-container{
  display: grid;
  grid-template: repeat(2, 1fr) / 1fr;
  gap: .5rem 0;
  #seconde-poster-top {
    display: grid;
```

```

grid-column: 1 / 2;
grid-row: 1 / 2;
}
#seconde-poster-bottom {
display: grid;
grid-column: 1 / 2;
grid-row: 2 / 3;
}
}

```

در قطعه کد بالا dir مقدار rtl گرفته است که این بدان معناست که ترتیب چینش ستون ها در grid از راست به چپ در نظر گرفته می شود.

المنت second-poster-container دو ستون هم اندازه و یکی ردیف به عنوان قالب در نظر گرفته است که هر یک از این ستون ها به یک poster اختصاص یافته.

در اندازه موبایل این دو poster زیر هم قرار می گیرند و چینش قالب المنت second-poster-container به شکل یک ستون و دو ردیف تغییر می یابد.

## Footer

- Logo صفحه درباره ما
- Logo و آدرس محل فروشگاه
- Logo و شماره پشتیبانی
- توضیح کوتاهی درباره خدمات فروشگاه به مشتریان
- ساعت کار
- Logo شبکه های اجتماعی



```

<footer class="mt-5 ps-3 pe-3 pt-3 white-font" dir="rtl">
  <section id="about-us">
  <section id="customer-and-worktime">
    <div id="worktime">
    <section id="social">

```

```

footer {
  grid-template: repeat(3, 1fr) / repeat(3, 1fr);
  background-color: #2c3e50;
  gap: .2rem .2rem;
  #about-us {
    grid-row: 1 / 2;
    grid-column: 1 / 4;
    grid-template: repeat(3, 1fr) / repeat(12, 1fr);
  }
  #customer-and-worktime {
    grid-row: 2 / 3;
    grid-column: 1 / 4;
  }
  #social {
    grid-row: 3 / 4;
    grid-column: 1 / 4;
    grid-template: 1fr / repeat(12, 1fr);
  }
}

```

سه ستون برای قالب المنت footer در نظر گرفته شده که به ترتیب از راست به چپ صفحه به المنت about-us ، customer-and-worktime و social اختصاص یافته.

جهت ایجاد فاصله میان ستون ها و ردیف ها از gap استفاده شده.

هر یک از المنت های ذکر شده خود دارای قالب جداگانه ای هستند تا محتوای آن ها چینش درستی داشته باشند.

```

#whatsapp {
  display: grid;
  grid-column: 9 / 12!important;
  grid-row: 2 / 3!important;
  font-size: 2rem!important;
  justify-items: center;
  animation: float 4s infinite;
  animation-delay: 2.5s;
}

```

```

}
@keyframes float {
  0%{transform: translateY(0);}
  50%{transform: translateY(-15px);}
  100%{transform: translateY(0);}
}

```

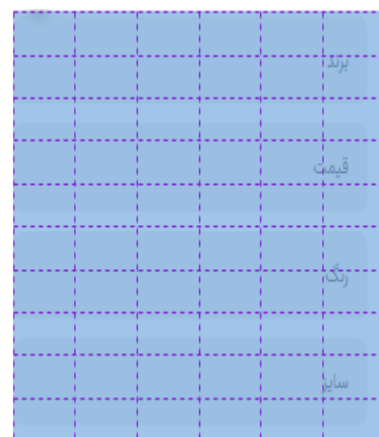
به المنت های whatsapp، telegram و instagram انیمیشنی داده شده با نام float که در آن از متد transform استفاده شده.

یکی از قابلیت های این متد translateY است که با توجه به زمان (4s) در نظر گرفته شده برای اجرای این انیمیشن و همچنین تقسیم این زمان به درصد های برگزیده، انیمیشن به ترتیب در ابتدا مقدار صفر دارد سپس پانزده پیکسل به سمت بالا حرکت کرده و دوباره در انتهای زمان به موقعیت ابتدای خود باز می گردد.

Infinite یکی از صفات animation است که به معنای تکرار بی انتهای انیمیشن است.

### فیلتر ها

- با استفاده از گزینه های قیمت ، برند ، رنگ و سایز میتوانیم محصولات مورد نظر را فیلتر کنیم و به محصولی که می خواهیم دست پیدا کنیم (در صورت موجود بودن)



```
<section id="main" dir="rtl" class="ps-3 pe-3">
```

```
<button class="btn btn-dark" id="advance-search">
```



```
<nav id="filter">
  <button class="accordion">برند</button>
  <div class="panel" id="panel-one">
<button class="accordion">قیمت</button>
  <div class="panel" id="panel-two">
<button class="accordion">رنگ</button>
  <div class="panel" id="panel-three">
<button class="accordion">سایز</button>
  <div class="panel" id="panel-four">
</nav>
</section>
```

```
#main {
  display: grid;
  grid-template: repeat(10, 1fr) / repeat(6, 1fr)!important;
  row-gap: 0 !important;
  width: 30%;
  margin-top: 0px !important;
  float: right;
  #filter {
    grid-row: 1 / 11 !important;
    grid-column: 1 / 7 !important;
    padding-right: 16px !important;
    .active {
      background-color: #eee;
    }
    .panel {
      padding: 0 18px;
      display: none;
```

```
background-color: white;
overflow: hidden;
transition: display 3s;
a {
  color: black;
  padding: .35em;
  display: block;
  text-decoration: none;
```

المنت main قالب شش ستون و ۱۰ ردیف دارد، المنت filter تمامی ستون ها و ردیف ها را به خود اختصاص داده است.

کلاس active برای دکمه های مردانه، زنانه و بچه گانه در بخش پیشنهادات قرار داده شده، در صورتی که روی این دکمه کلیک شود این کلاس به آن ها اضافه می شود. این کلاس رنگ پس زمینه دکمه ها را تغییر می دهد.

کلاس panel که محتویات فیلتر مربوط به هر دکمه فیلتر در آن قرار دارد به طور پیش فرض از دید کاربر مخفی شده است و نمایش داده نمی شود، تنها زمانی که کاربر روی دکمه فیلتر مورد نظر کلیک می کند panel مربوطه مقدار display = block می گیرد و این ظاهر شدن به مدت 3 ثانیه ( transition: display 3s) زمان می برد.

Text-decoration: none این ویژگی مانع از رسم خط زیر لینک های تگ a و همچنین آبی شدن این المنت می شود.

```
<section id="main2">
  <div id="product-box">
    <nav id="topics">
      <div id="products" class="pe-1 ps-1">
        <section id="offer-main" dir="rtl" class="w-100 h-100">
          <div class="offers-box p-1" id="offer-1">
            <div class="offers-box p-1" id="offer-2">
              <div class="offers-box p-1" id="offer-3">
                <div class="offers-box p-1" id="offer-4">
                  <div class="offers-box p-1" id="offer-5">
```

```
<div class="offers-box p-1" id="offer-6">
```

```
#main2{
  grid-template: repeat(6, 1fr) / repeat(6, 1fr);
  width: 68%;
  #product-box {
    grid-row: 1 / 7;
    grid-column: 1 / 7;
    grid-template: 50px repeat(6, 1fr) / repeat(4, 1fr);
    #topics {
      grid-row: 1 / 2;
      grid-column: 1 / 5;
      grid-template: 1fr / repeat(3, 1fr);
    }
    #products {
      grid-row: 2 / 8;
      grid-column: 1 / 5;
      grid-template: repeat(6, 1fr) / repeat(4, 1fr);
      #offer-main {
        grid-column: 1 / 7;
        grid-row: 1 / 7;
        #offer-1 {
          grid-column: 1 / 4;
          grid-row: 1 / 3;
        }
      }
    }
  }
}
```

برای المنت `main2` قالب شش ستون و شش ردیف ایجاد شده و عرض این المنت شش و هشت درصد عرض صفحه می باشد.

تمامی ستون ها و ردیف ها به المنت `product-box` تخصیص می یابد و خود این المنت شامل دو فرزند می شود که فرزند اول آن المنت `topics` است و فرزند دوم آن المنت `products` می باشد.





المنت `topics` سه گزینه ارزان ترین، پرفروش ترین، بیشترین تخفیف را در خود جای داده.

المنت `products` دارای شش ردیف و چهار ستون می باشد (در عرض صفحه 768px) که برای شش محصول در نظر گرفته شده.

## سبد خرید

### ۱- محصولات اضافه شده به سبد خرید

- در این بخش میتوان عکس محصول ، قیمت ، تعداد ، هزینه ارسال و هزینه کل را مشاهده کرد.

نام کالا	قیمت واحد	تعداد	قیمت نهایی
 زیرپوش نخی	20000	<input type="text" value="1"/>	20000
 زیرپوش نخی	20000	<input type="text" value="1"/>	20000
 زیرپوش نخی	40000	<input type="text" value="1"/>	40000
 زیرپوش نخی	30000	<input type="text" value="1"/>	30000
ارسال رایگان برای سفارش بالای 200 هزار تومان			
هزینه ارسال 10000 تومان		مجموع 120000 تومان	
ادامه خرید		پرداخت	

```

<div class="container mb-4">
  <div class="row">
    <div class="col-12">
      <div class="table-responsive">
        <table class="table table-striped" dir="rtl" style="text-align: center">
          <thead>
            <tr>
              <th class="col"></th>
              <th class="col">نام کالا</th>
              <th class="col">قیمت واحد</th>
              <th class="col">تعداد</th>
              <th class="col">قیمت نهایی</th>
            </tr>
          </thead>
          <tbody>
            <tr>
              <td>
              <td><p>زیرپوش نخی</p>
              <td class="single_cost">20000</td>

```

```

<td>
  <button class="btn btn-secondary"></button>
  <input class="form-control number_of_product" type="text"
    value="1"/>
  <button class="btn btn-secondary"> </button>
</td>
<td class="final_cost">0<span> تومان</span></td>
<td class="text-right remove">
  <button class="btn btn-sm btn-danger"></button>
</td>

```

جهت طراحی لیست کالاها در صفحه سبد خرید از تگ **table** استفاده شده که این تگ برای ایجاد جدول ساخته شده است. تگ هایی که درون تگ **table** دارای معنا هستند شامل **th**, **thead**, **tr** و ... می باشند. **thead**: این تگ **header** (سر ستون ها) مربوط به جدول را اشغال می کند. شامل یک ردیف و چندین ستون می باشد.

**tbody**: مجموعه ای از تگ های **tr** را در خود جای می دهد و بدنه جدول را تشکیل می دهد.

**tr**: این تگ یک ردیف از تگ پدر خود را به خود اختصاص می دهد.

**th**: به جای استفاده از تگ **td** از این المنت برای تعیین ستون در **header** استفاده می شود علت اینکار بهره گیری از **style** از پیش تعیین شده در این تگ است.

**td**: یک ستون از تگ پدر خود را به خود اختصاص می دهد.

به المنت **th** کلاس **col** داده شده است به یکی از صفات مورد استفاده در روش **flex** است. این صفت بدان معنا است که عرض المنت پدر به تعداد فرزندان تقسیم شده و هر تمامی فرزندان عرض یکسانی را به خود اختصاص می دهند، در صورتی که این صفت به **col-2** تغییر پیدا کند به این معنی است که این فرزند خاص نسبت به فرزندان دیگر دو برابر عرض آنها را به خود اختصاص می دهند.

عدد **col-md** این کلاس جهت **responsive** کردن المنت ها استفاده می شود، این کلاس به این معنی است که در عرض صفحه **720px** به بعد المنتی که این کلاس را دارد به تعداد عدد نوشته شده ستون هایی را از المنت پدر خود اشغال می کند.

تگ **i**: این تگ برای استفاده از **icon** ها است. علت استفاده از این تگ به جای قرار دادن عکس این است که **icon** های استفاده شده حالت نوشته دارند و حجم کمی را به خود اختصاص می دهند، این در

حالی است که عکس ها معمولا برای وضوح بیشتر نیاز به کیفیت بالاتری دارن که این امر خود باعث بالا رفتن حجم عکس می شود و زمانی که در یک سایت تنها از عکس ها به جای icon ها استفاده کنیم زمان load سایت بسیار نسبت به حالتی که از تگ i استفاده می کنیم بیشتر می شود.

کلاس table-striped یکی از کلاس های آماده bootstrap است که رنگ پس زمینه ردیف ها را به صورت یکی در میان از هم متمایز می کند.

### فرم ثبت نام

- این فرم شامل درگاه هایی برای دریافت اطلاعات ثبت نام است.
- این اطلاعات شامل نام ، نام خانوادگی ، ایمیل ، رمز عبور ، شماره همراه و آدرس است.

#### فرم ثبت نام

نام خانوادگی	نام
<input type="text"/>	<input type="text"/>
ایمیل	<input type="text"/>
رمز عبور	<input type="text"/>
تکرار رمز عبور	<input type="text"/>
شماره همراه	<input type="text"/>
آدرس 1	<input type="text"/>
آدرس 2	<input type="text"/>
<input type="button" value="ثبت نام"/>	

```
<div class="panel panel-primary">
  <div class="panel-heading text-center mb-4"><h4> فرم ثبت نام</h4></div>
  <div class="panel-body" id="resultreg">
    <form id="signup_form" onsubmit="return false" dir="rtl">
      <div class="row">
        <div class="col-md-6">
          <label for="f_name">نام</label>
          <input type="text" id="f_name" name="f_name"
            class="form-control" required>
        </div>
      </div>
    </form>
  </div>
</div>
```

```

<div class="col-md-6">
    <label for="l_name">نام خانوادگی</label>
    <input type="text" id="l_name" name="l_name"
class="form-control" required>
</div>
</div>

```

از تگ form جهت ایجاد فرمی برای دریافت ورودی از کاربر استفاده می‌شود. یک فرم می‌تواند شامل فیلدهای متنی، checkbox، radio button و موارد دیگر باشد. فرم‌ها برای ارسال داده‌های کاربر به یک URL مشخص استفاده می‌شوند.

المنت با کلاس row یک ردیف از پدر خود را اشغال می‌کند و در صورتی که مانند بخش ابتدای فرم ثبت نام قصد داشته باشیم دو ورودی در یک ردیف قرار گیرند کفایت دو ورودی خود را درون یک المنت row قرار دهیم و همچنین در صورتی که قصد داریم یک ردیف کامل را مانند ایمیل به یک المنت اختصاص دهیم کفایت این ورودی را درون یک المنت row قرار دهیم.

تگ label جهت ایجاد یک تیترا برای ورودی‌ها ساخته شده است که جهت اتصال آن به ورودی مربوطه به خود کفایت صفت for را برابر name ورودی مربوطه کنیم.

### مشخصات محصول

این بخش شامل قیمت، جزئیات، دکمه اضافه کردن به سبد خرید، کد، رنگ و سایز محصول است. در صورتی که سایز یا رنگ محصول انتخاب محصول انتخاب نشود دکمه اضافه کردن به سبد خرید فعال نمی‌شود.

248,000

372,000

سایز :

یک سایز را انتخاب کنید ▼

رنگ :

یک رنگ را انتخاب کنید ▼

راهنمای انتخاب سایز ①

کد محصول : A323324

جزئیات :

- برند: 1991
- مناسب فصل: بهار و تابستان
- جنس: چودون
- مدل یقه: شومیز
- قد جلوی تیشرت از پشت آن کوتاه‌تر است.
- ملاحظات نگهداری: شستشو در دمای 30 درجه انجام شود
- سایز L:
- عرض شانه: 46 سانتی متر
- قد از سرشانه: 75 سانتی متر
- دور سینه: 104 سانتی متر
- طول آستین: 27 سانتی متر

افزافه کردن به سبد خرید

تیشرت مردانه ورزشی

```

<div id="product" class="p-0 pe-3 ps-3 pt-4" dir="rtl">
  <div id="product-slider" class="carousel carousel-dark slide"
    data-bs-ride="carousel">
    <div id="product-box" class="mt-4" dir="rtl">
      <div id="product-detail" class="pt-5" dir="rtl">
        <div id="add-to-basket">

```

```

#product {
  grid-template: 1fr .4fr .5fr .1fr / 1fr;
  #product-slider {
    grid-area: 1 / 1 / 2 / 2;
  }
  #product-box {
    grid-area: 2 / 1 / 3 / 2;
    grid-template: 1fr 25px 30px 60px 60px 20px 20px / repeat(4, 1fr);
    max-width: 700px;
  }
  #product-detail {
    grid-area: 3 / 1 / 4 / 2;
    grid-template: 30px 1fr / 1fr;
  }
  #add-to-basket {
    grid-area: 4 / 1 / 5 / 2;
  }

```

المنت product شامل سه فرزند است، product-slider، product-box، product-detail، add-to-basket. این المنت در سایز موبایل قالب چهار ردیف و یک ستون را داراست.

Grid-area شامل چهار عدد است که به ترتیب از چپ به راست شروع ردیف، شروع ستون، پایان ردیف و پایان ستون را در خود جای داده و short hand دو ویژگی grid-column و grid-row است.

max-width این ویژگی باعث می شود المنت مورد نظر عرض بیشتری از حد تعیین شده پیدا نکند. در قطعه کد بالا این کار به این علت انجام شده تا در اندازه های بزرگ تر از موبایل عکس مربوط به محصول بیش از اندازه بزرگ نشود.



## Javascript : login.js

- در صورت دریافت عدد ۰ از سمت سرور خطای ثبت نام برای کاربر ظاهر می شود.
- در صورت دریافت عدد ۱ از سمت سرور پیام خوش آمدید برای کاربر ظاهر می شود.
- در صورت login بودن کاربر نام کاربر را در کنار جمله خوش آمدید قرار داده و به کاربر نمایش میدهد.

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

یکی از متدهای jquery است که باعث می شوند تا زمانی که صفحه html به طور کامل load نشده کدهای javascript مربوط به آن صفحه اجرا نشوند.

```
$("#login").submit(function (event) {  
    var formData = {  
        email: $("#email").val(),  
        pass: $("#password").val(),  
        sub: $("#loginsubmit").val(),  
    };  
});
```

در قطعه کد بالا یک add event listener برای المنت login تعریف شده که در صورت فشردن دکمه ورود به حساب کاربری اجرا می شود و مقادیر وارد شده در فرم ورود به حساب کاربری به عنوان value های دیکشنری formdata قرار داده می شوند.

```
$.ajax({  
    type: "POST",  
    url: "process.php",  
    data: formData,  
    dataType: "json",  
    encode: true,  
}).done(function (data) {  
    console.log(data);  
    if (data == "1") {  
        $("#login").html($("#email").val() + " خوش آمدید.");  
    } else {  
        $("#login").html("خطا در ورود");  
    }  
});
```

```
});
```

```
event.preventDefault();
```

در قطعه کد بالا از ajax استفاده شده تا بتوان اطلاعات را به سمت سرور ارسال و دریافت کرد. نوع ارسال اطلاعات به صورت post تعریف شده که از نظر امنیتی از نوع get بهتر است به این دلیل که نوع get، در بخش addressBar مرورگر کاربر، اطلاعات ارسالی به سمت سرور را نمایش می‌دهد و کاربر میتواند از محتویات دیکشنری (نام کلید ها)، ورودی و خروجی ها کد ajax با خبر شود و با کمی تغییر در نام ها باعث اختلال در سامانه ما شود. از این نوع در ارسال از کلاینت به سرور بهتر است استفاده نشود اما در زمان دریافت از سمت سرور مشکل از نظر امنیت ایجاد نمی‌کند. اطلاعات درون دیکشنری به سمت سرور ارسال می‌شود و در صورت موفقیت در ارسال وارد بخش done می‌شود. حال در صورتی که اطلاعات ورودی درست باشند از سمت سرور توسط کد های موجود در process.php عدد ۱ به سمت کلاینت باز گردانده می‌شود که در این حالت نام ایمیل فرد چاپ شده و پیام خوش آمدید برای کاربر ظاهر می‌شود و در صورت عدم صحت اطلاعات ورودی (یعنی کاربری با این ایمیل در سامانه ثبت نام نکرده است) پیام خطا در ورود برای کاربر نمایش داده می‌شود. سپس مقادیر اختصاص یافته به متغیر ها توسط event.preventDefault() به حالت اولیه خود باز می‌گردند.

```
$ajax({
  type: "GET",
  url: "process.php?islogin=1",
}).done(function (data) {
  console.log(data);
  if (data == "0") {
    // waiting to login...
  } else {
    $("#login").html(data + " خوش آمدید ");
  }
});
```

در صورتی که متد islogin به کاربر مقدار یک را برگرداند به این معنا است که کاربر قبلاً وارد حساب خود شده و نیاز به ورود مجدد در هر صفحه به حساب کاربری نمی‌باشد.

: script.js

۱- تابع getWidth() :

```
function getWidth (){  
    var maxwidth = Math.max(  
        document.body.scrollWidth,  
        document.documentElement.scrollWidth,  
        document.body.offsetWidth,  
        document.documentElement.offsetWidth,  
        document.documentElement.clientWidth);  
    if (maxwidth >= 576) {  
        $("#brand").removeClass("justify-content-center");  
        $("#brand").addClass("justify-content-end");  
        $("#item-bar").removeClass("col");  
        $("#item-bar").addClass("col-3");  
    }  
    if (maxwidth < 576) {  
        $("#brand").removeClass("justify-content-end");  
        $("#brand").addClass("justify-content-center");  
        $("#item-bar").removeClass("col-3");  
        $("#item-bar").addClass("col");  
    }  
    const navHeight = $("header .navbar").height() + 16;  
    $("header").css({ height: navHeight });  
}  
window.addEventListener("resize", getWidth);
```

عرض صفحه را از مرورگر دریافت کرده و از آن برای تعیین موقعیت های برند و سبد خرید و حساب کاربری در header استفاده می کند. این متد زمانی فراخوانی می شود که صفحه مرورگر کاربر تغییر اندازه پیدا کند.

```

$("#item1 > a").click(function () {
    $("#item1 > a").addClass("active-1");
    $("#item2 > a, #item3 > a").removeClass("active-1");
});
$("#item2 > a").click(function () {
    $("#item2 > a").addClass("active-1");
    $("#item1 > a, #item3 > a").removeClass("active-1");
});
$("#item3 > a").click(function () {
    $("#item3 > a").addClass("active-1");
    $("#item1 > a, #item2 > a").removeClass("active-1");
});

```

در صورت کلیک کاربر روی دکمه‌های item1,item2,item3 این متدها فراخوانی می‌شوند. به دکمه مورد نظر (مردانه، زنانه و بچه‌گانه) کلاس active-1 را در صورت عدم وجود اضافه می‌کنند. در صورتی که از قبل این کلاس به المنت مورد نظر ما اضافه شده باشد، این کلاس از آن المنت پاک می‌شود که اینکار باعث می‌شود تنها یکی از این المنت ها کلاس active-1 را داشته باشند.

۲- تابع scrollFunction() :

```

$(window).scroll(function () {
    scrollFunction();
});

```

در صورتی که پیمایش در صفحه رخ دهد این متد اجرا می‌شود و متد scrollFunction را درون خود فراخوانی می‌کند.

```

Var header = document.getElementsByTagName("header");
var sticky = header.offsetTop;
var sticky_box = document.getElementById("sticky_box");
function scrollFunction() {
    if (window.pageYOffset > sticky) {
        header.addClass("sticky");
    }
}

```

المنت header و sticky\_box را به متغیرهایی با همان نام‌ها اختصاص می‌دهیم. فاصله المنت header را از بالای صفحه برابر صفر قرار می‌دهیم.

```
$(sticky_box).click(function(){scrollBack()});
```

در صورت کلیک روی دکمه sticky\_box این متد فراخوانی می‌شود و متد scrollBack را فراخوانی می‌کند.

```
if (document.documentElement.scrollTop > 500) {  
    sticky_box.style.display = "block;"  
} else {  
    sticky_box.style.display = "none;"  
}  
function scrollBack() {  
    document.documentElement.scrollTop = 0;  
}  
}
```

در صورتی که پنجره مربوط به مرورگر کاربر به اندازه 500px از ابتدای سایت به سمت پایین پیمایش پیدا کند این متد فراخوانی می‌شود و المنت sticky\_box را نمایش می‌دهد.  
در صورتی که کمتر از 500px باشد هم display این المنت را برابر none قرار می‌دهد.  
متد scrollBack پنجره مربوط به مرورگر کاربر را به ابتدای صفحه سایت هدایت می‌کند.  
۳- نمایش منوهای فیلتر :

```
var acc = document.getElementsByClassName("accordion");  
var i;  
for (i = 0; i < acc.length; i++) {  
    acc[i].addEventListener("click", function () {  
        var panelList = ["panel-one", "panel-two", "panel-three", "panel-four"];  
        this.classList.toggle("active");  
        var panel = this.nextElementSibling;  
        $(`#${panel.id}`).css("display", "block");  
        panelList = panelList.filter((e) => e !== panel.id);  
        for (let index = 0; index < panelList.length; index++) {  
            $(`#${panelList[index]}`).css("display", "none");  
        }  
    });  
}
```

```
});  
}
```

در صورت کلیک روی تیتريهای فیلترهای مثل رنگ ، مارک و ... ، پنل مربوط به هر بخش زیر آن به نمایش در می آید. این عمل در قطعه کد بالا به این صورت انجام می شود که لیستی از المنت هایی که کلاس accordion دارند را به متغیر acc اختصاص می دهیم، سپس حلقه ای به اندازه تعداد این المنت ها ایجاد می کنیم و برای تمامی این المنت ها addEventListener قرار می دهیم تا در صورت کلیک روی این المنت ها panel مربوط به خودشان نمایش داده شوند. جهت بسته شدن panel مربوط به المنت های دیگر بعد از باز شدن panel مربوط به المنت فیلتری که روی آن کلیک شده درون متغیر panelList، id تمامی panel ها را قرار می دهیم و با this.nextElementSibling ، panel المنت فیلتری که روی آن کلیک شده را درون متغیر panel می ریزیم حال id آن panel ای که المنت فیلتر آن انتخاب شده را از درون لیست پاک می کنیم و در حلقه دوم display تمامی panel های دیگر را none می کنیم.

۴- تابع صفحه ورود به حساب کاربری :

```
$("#profile_icon").click(function () {  
    $("#login").css("display", "block");  
    $("#myModal").css("display", "block");  
});  
var modal = document.getElementById("myModal");  
window.onclick = function (event) {  
    if (event.target == modal) {  
        modal.style.display = "none";  
    }  
};
```

در صورت کلیک روی icon حساب کاربری ، منو ورود برای کاربر نمایش داده می شود و در صورت کلیک روی نواحی خارج از محدوده myModal این منو بسته می شود.

۵- تابع حذف ردیف در صفحه سبد خرید :

```
$(".remove").click(function () {  
    this.parentElement.remove();  
    var costList = document.getElementsByClassName("single_cost");  
    var number_of_product_list =
```

```

document.getElementsByClassName("number_of_product");
var final_cost_list = document.getElementsByClassName("final_cost");
// main_cost = sum of costs
var main_cost = 0;
for (let index = 0; index < costList.length; index++) {
    let cost = costList[index].innerHTML;
    let number_of_product = parseInt(number_of_product_list[index].value);
    final_cost_list[index].innerHTML = cost * number_of_product;
    main_cost += parseInt(final_cost_list[index].innerHTML);
}
if (main_cost >= 200000) {
    total_cost_element = document.getElementById("total_cost");
    total_cost_element.innerHTML = main_cost;
    $("#free_post").css("color", "#2ecc71");
} else {
    var post_cost = 10000;
    total_cost_element = document.getElementById("total_cost");
    total_cost_element.innerHTML = main_cost + post_cost;
    $("#free_post").css("color", "#e74c3c");
    setTimeout(function () {
        var post_cost_element = document.getElementById("post_cost");
        post_cost_element.innerHTML = `${post_cost}` + `<span>تومان</span>`;
    }, 1500);
}
});

```

این متد برای کلاس remove نوشته شده است و روی این کلاس add event listener قرار داده شده تا در صورت کلیک روی این کلاس متد فراخوانی شود. بعد از کلیک روی این کلاس ردیفی که مربوط به این محصول است از لیست محصولات سبد خرید حذف شده و هزینه آن از مجموع هزینه کل کم می شود. برای انجام اینکار المنت پدر را انتخاب و سپس با متد remove حذف می کنیم.

حال برای اینکه هزینه کل اشتباه نباشد بیاد یک بار دیگر مجموع هزینه کالا های باقی مانده را محاسبه کنیم که برای انجام اینکار مبالغ موجود تمامی محصولات موجود در سبد خرید را به صورت لیست درون متغیر `costList` می‌ریزیم همین کار را برای تعداد محصولات انجام می‌دهیم و آن‌ها را به متغیر `number_of_product_list` اختصاص می‌دهیم.

جهت محاسبه هزینه اصلی (بدون احتساب هزینه ارسال) درون حلقه هزینه هر یک از محصولات را در تعداد آن ضرب می‌کنیم و در نهایت با هم جمع می‌کنیم سپس در صورتی که هزینه اصلی از ۲۰۰ هزار تومان بیشتر باشد هزینه ارسال محاسبه نشده و هزینه نهایی ما همان هزینه اصلی می‌شود و در غیر این صورت هزینه ارسال با هزینه اصلی جمع شده و به عنوان هزینه نهایی به کاربر نمایش داده می‌شود.

۶- تابع دکمه های اضافه کردن تعداد محصول :

```
$(“button:has(i.bi-caret-down-fill)”).click(function(){
    let buttonParent = this.parentNode;
    let number_of_product = buttonParent.children[1];
    number_of_product_value = parseInt(number_of_product.value);
    if (number_of_product_value == 1) {
        alert("تعداد محصول کمتر از ۱ نمیتواند باشد");
    } else {
        number_of_product_value -= 1;
        number_of_product.value = number_of_product_value;
    } ... });
$(“button:has(i.bi-caret-up-fill)”).click(function(){...});
```

در صورت کلیک روی دکمه های فلش بالا و پایین تعداد محصولات در هر ردیف به ترتیب بیشتر و کمتر می‌شود ، در صورتی که تعداد کمتر از ۱ شود `alert` برای کاربر ظاهر می‌شود و در صورتی که جمع هزینه کل محصولات از ۲۰۰ هزار تومان بیشتر شود هزینه ارسال رایگان می‌شود.

**: filter.js**

در این بخش میتوانیم محصولاتی که روی آن ها فیلتر اعمال شده را مشاهده کنیم.

```
var objclone;
$(document).ready(function() {
    objclone = $("#offer-main").html()
    loadproducts()
```



```

$("input[type=checkbox]").click(function () {
    loadproducts()
});
$("input[type=radio]").click(function () {
    loadproducts()
});
});

```

متغیر global با نام objclone تعریف شده که درون آن html المنت offer-main قرار دارد. علت اینکار دریافت یک محتوای خام و پیش فرض است تا در صورتی که کاربر به صفحه filter بازگشت بتوانیم دوباره تمام محصولات اولیه را به نمایش بگذاریم. (نه کالا های فیلتر شده)

به صورت پیش فرض متد loadproducts فراخوانی می شود در صورت کلیک کاربر روی ورودی های از نوع radio و checkbox، باز هم این متد فراخوانی می شود و مقادیر فیلتر را بروزرسانی می کند.

```

function loadproducts() {
    $("#offer-main").html(objclone);
    $(".id^=offer-").each(function () {
        if ($(this).attr("id") != "offer-main") {
            $(this).hide();
        }
    });
}

```

در متد loadproducts، display محصولات موجود در المنت offer-main (فرزندان این المنت) را برابر none می گذاریم.

```

var qstring = "";
$("input[type=checkbox]").each(function () {
    if ($(this).attr("id")) {
        qstring +=
            "&" + $(this).attr("id") + "=" + ($(this).prop("checked") ? "1" : 0);
    }
});
$("input[type=radio]").each(function () {
    if ($(this).attr("id")) {

```

```

qstring +=
    "&" + $(this).attr("id") + "=" + ($(this).prop("checked") ? "1" : 0);
}
});

```

یک رشته با نام qstring تعریف شده که درون آن id المنت های ورودی فیلتر به همراه عددی که نشان دهنده فعال بودن (۱) یا عدم فعال بودن (۰) ورودی است، قرار دارد.

این مقادیر به صورت پشت سر هم به انتهای این رشته اضافه می شوند.

```

$.ajax({
    type: "GET",
    url: "process.php?filter=1" + qstring,
}).done(function (data) {
    const dataobj = JSON.parse(data);
    for (i = 0; i < dataobj.length; i++) {
        id = i + 1;
        var replaced = $("#offer-" + id)
            .html()
            .replace("زیرپوش نخی", dataobj[i].name);
        $("#offer-" + id).html(replaced);
        $("#offer-" + id).show();
    }
});

```

یکی از کاربردهای رایج json تبادل داده به/از یک وب سرور است که هنگام دریافت داده از یک وب سرور، داده ها همیشه یک رشته هستند و داده ها را با `parse.json()` تجزیه می کنید و داده ها به یک شی جاوااسکریپت تبدیل می شوند.

درون حلقه محصولات پیش فرض موجود در صفحه فیلتر جایگزین محصولات مورد نظر می شوند که اینکار با متد `replace` صورت می گیرد و درون متغیر `replaced` ریخته می شوند و در نهایت نمایش داده می شوند (`display = block`).

## : signup.js

- در صورتی که روی دکمه ثبت نام کلیک شود با استفاده از jquery مقادیر ورودی ها را درون متغیر ها ریخته و این متغیر ها را با استفاده از ajax به سرور ارسال می کند.
- در صورت خالی قرار دادن ورودی های فرم ثبت نام و کلیک روی دکمه ثبت نام برای کاربر پیام خطا در ثبت نام ظاهر می شود.

```
$("#signup_form").submit(function (event) { (
```

```
var formData = {  
    name: $("#f_name").val(),  
    lname: $("#l_name").val(),  
    email: $("#emailreg").val(),  
    pass1: $("#password1").val(),  
    pass2: $("#password2").val(),  
    mob: $("#mobile").val(),  
    sub: $("#submit").val(),  
    address1: $("#address1").val(),  
    address2: $("#address2").val(),  
};
```

در صورت کلیک روی دکمه ثبت نام یک دیکشنری ایجاد می شود به نام formData . مقادیر اطلاعات وارد شده توسط کاربر درون کلید های این دیکشنری ذخیره می شوند.

```
$.ajax({  
    type: "POST",  
    url: "process.php",  
    data: formData,  
    dataType: "json",  
    encode: true,  
}).done(function (data) {  
    console.log(data);
```

```
$("#signup_form").hide();  
if (data == "1") {  
    $("#resultreg").html("<center>ثبت نام با موفقیت انجام شد.</center>");  
} else {  
    $("#resultreg").html("<center>خطا در ثبت نام.</center>");  
}  
});  
event.preventDefault();
```

دیکشنری ساخته شده با نام `formData` را به سمت سرور ارسال می‌شود. در صورت دریافت عدد یک از سرور پیام "ثبت نام با موفقیت انجام شد" برای کاربر به نمایش در می‌یاید و در صورت عدم تکمیل فرم از سمت سرور عدد صفر بازگردانده می‌شود که در این حالت خطا در ثبت نام برای کاربر نمایش داده می‌شود.

## راه اندازی اولیه

- نصب xampp و راه اندازی سرویس mysql و apache
- استخراج کردن فایل star.rar در فولدر به آدرس C:\xampp\htdocs\
- باز کردن مرورگر و وارد کردن آدرس <http://localhost/phpmyadmin/>
- روی گزینه user accounts کلیک کرده و سپس در صفحه باز شده روی گزینه add user account کلیک می‌کنیم.
- در صفحه باز شده باید نام کاربری و رمز عبور را مطابق فایل process.php به ترتیب star و shopmestar قرار بدهیم سپس باید تیک گزینه Create database with same... را بزنید تا database ساخته شده با نام کاربر یکسان شود و اجازه دسترسی مورد نیاز به این کاربر داده شود حال روی گزینه go در پایین صفحه کلیک می‌کنیم.
- روی دیتابیس ایجاد شده به نام star کلیک کرده و سپس روی زبانه sql کلیک میکنیم و محتویات فایل shop\_moh.sql موجود در فایل پروژه را در این بخش pase کرده و روی گزینه go کلیک میکنیم تا جداول products و users در آن ایجاد شوند.
- وب سایت آماده ارائه خدمات است ، حال میتوانید با وارد کردن آدرس زیر وارد سایت شوید.
- آدرس : <http://localhost/star/main.html>