**پایان نامه**

**(گزارش کار پروژه)**

بسم الله الرحمن الرحیم

**چکیده**

**کلمات کلیدی :**

پروژه پایان ترم کارشناسی نرم افزار

برنامه نویسی

وبسایت

وبسایت فروشگاهی

پی اچ پی

فهرست مطالب

عنوان صفحه

[فصل اول 4](#_Toc112006620)

[**مقدمه** 5](#_Toc112006621)

[**تاریخچه** 6](#_Toc112006622)

[**مبانی نظری** 7](#_Toc112006623)

[**انگیزه** 8](#_Toc112006624)

[**فصل دوم** 9](#_Toc112006625)

[**توضیح ابزارهای مورد نیاز این پروژه** 10](#_Toc112006626)

[**فصل سوم** 11](#_Toc112006627)

[**دیتابیس** 12](#_Toc112006628)

[**جدول Product** 13](#_Toc112006629)

[**جدول Users** 14](#_Toc112006630)

[**جدول Categories** 15](#_Toc112006631)

[**جدول Orders** 16](#_Toc112006632)

[**جدول order\_items** 17](#_Toc112006633)

[**جدول payments** 18](#_Toc112006634)

[**طراحی ظاهری سایت** 19](#_Toc112006635)

[**BOOTSTRAP چیست ؟** 20](#_Toc112006636)

[**BACKEND** 27](#_Toc112006637)

[**Backend چیست ؟** 27](#_Toc112006638)

[**زبان برنامه نویسی پی اچ پی** 28](#_Toc112006639)

[شی گرایی 29](#_Toc112006640)

[**معماری MVC** 30](#_Toc112006641)

[**فصل چهارم** 31](#_Toc112006642)

[**نتیجه گیری** 32](#_Toc112006643)

فهرست شکل ها

عنوان صفحه

[1. جدول محصولات 13](#_Toc111995700)

[2. جدول کاربران 14](#_Toc111995701)

[3. جدول دسته بندی ها 15](#_Toc111995702)

[4 . جدول سفارش ها 16](#_Toc111995703)

[5 . جدول اتیم های هر سفارش 17](#_Toc111995704)

[6 . جدول پرداخت ها 18](#_Toc111995705)

[7 . صفحه اصلی فروشگاه 21](#_Toc111995706)

[8 . صفحه تکی محصول 22](#_Toc111995707)

[9 . سبد خرید و پرداخت 23](#_Toc111995708)

[10 . پنل ادمین 24](#_Toc111995709)

[11 . پنل کاربر 25](#_Toc111995710)

# فصل اول

**کلیات**

## **مقدمه**

این پروژه نرم افزاری برای بخشی از فعالیت های علمی و پژوهشی مقطع کارشناسی نرم افزار ایجاد شده.

در این پروژه سعی ما بر این بوده تا با علمی که در این مقطع تحصیلی فرا گرفته ایم بتوانیم خروجی مفید و هم سو با نیاز های بازار کار داشته باشیم.

موضوع پروژه یک گالری انلاین هست که هنرمندان میتوانند اثار خود را به صورت دیجیتالی در این وبسایت به فروش برسانند.

تکنولوژی های استفاده شده در ساخت این وبسایت شامل زبان برنامه نویسی پی اچ پی و دیتابیس مای‌اس‌کیوال می باشد.

## **تاریخچه**

زمانی که Tim Berners-Lee در سال 1989 اینترنت را خلق کرد احتمالا فکرش را نمی کرد که دنیای اینترنت از آن زمان تا این حد تغییر کند.

در آن دوران طراحی وب به معنای واقعی تاریک بود و تنها یک صفحه نمایش، با قابلیت اضافه کردن چندین کد وجود داشت و طراحی توسط نمادها و دکمه‌ی Tab انجام می شد.

از اولین وبسایتی که در اوایل دهه 1990 ساخته شد، طراحان مشغول انجام آزمایش ها و به کار گیری راه های مختلف برای تغییر در ظاهر و کارآمدی وبسایت ها بوده اند. در ابتدا، اولین وبسایت هایی که ساخته شد کاملا بر پایه متن بودند و کمتر وبسایتی وجود داشت که در آن عکسی وجود داشته باشد و اساسا طرح بندی ها و تگ ها نظیر هدینگ (heading) ها و پاراگراف ها مفهوم چندانی نداشتند.

در ابتدای پیدایش اینترنت، اکثر وبسایت ها توسط بخش های IT ساخته و طراحی می شدند، اما امروز طراحی وبسایت به یک صنعت جداگانه و مجزا تبدیل شده است و طراحان وبسایت با کیفیت نقش بسیار زیادی در پیشرفت و توسعه ی یک کسب و کار دارند.

## **مبانی نظری**

برای قسمت بک اند سایت از زبان برنامه نویسی پی اچ پی استفاده کرده ایم که یکی از پر استفاده ترین زبان ها در قسمت بک اند بوده و همچنین اوپن سورس و رایگان است و هدف اصلی از ساخت این زبان نیز برای توسعه سمت سرور بوده است .

پایگاه داده مای‌اس‌کیوال هم به دلیل انطباق خوبی که با زبان برنامه نویسی پی اچ پی دارد و همچنین به دلیل اینکه یک پایگاه داده رابطه ای است و ما را از مزایای خوب این دسته از پایگاه داده ها بهره مند میسازد استفاده کرده ایم.

برای طراحی قسمت ظاهری این سایت نیز مانند همه سایت های دنیا از زبان نشانه گذاری html و همچنین برای استایل دهی نیز از CSS استفاده کرده ایم تا قالبی ریسپانسیو داشته باشیم

## **انگیزه**

این پروژه با هدف فروش محصولات دیجیتال مانند تصاویر و غیره ایجاد شده است.

با استفاده از این وبسایت هنرمند قادر هست محصولات خود را به صورت اینترنتی به فروش برساند و بعد از خرید توسط کاربر محصولات به ایمیل کاربر ارسال شده و همچنین در پنل کاربر قابل دانلود کردن هستند.

# **فصل دوم**

**کار های مرتبط**

## **توضیح ابزارهای مورد نیاز این پروژه**

برای ساخت این وبسایت از زیان برنامه نویسی پی اچ پی و پایگاه داده مای‌اس‌کیوال استفاده شده است.

برای قسمت بک اند سایت از زبان برنامه نویسی پی اچ پی استفاده کرده ایم که یکی از پر استفاده ترین زبان ها در قسمت بک اند بوده و همچنین اوپن سورس و رایگان است و هدف اصلی از ساخت این زبان نیز برای توسعه سمت سرور بوده است .

پایگاه داده مای‌اس‌کیوال هم به دلیل انطباق خوبی که با زبان برنامه نویسی پی اچ پی دارد و همچنین به دلیل اینکه یک پایگاه داده رابطه ای است و ما را از مزایای خوب این دسته از پایگاه داده ها بهره مند میسازد استفاده کرده ایم.

برای طراحی قسمت ظاهری این سایت نیز مانند همه سایت های دنیا از زبان نشانه گذاری HTML و همچنین برای استایل دهی نیز از CSS استفاده کرده ایم تا قالبی ریسپانسیو داشته باشیم

# **فصل سوم**

**بیان جزئیات روش و کاری**

## **دیتابیس**

برای شروع طراحی این پروژه اولین قسمتی رو که قراره سراغش بریم دیتابیس هست

دیتابیس محل ذخیره سازی داده ها مورد نیاز ما است تا بتوانیم تغیرات دلخواه خود را روی ان به وجود بیاوریم

پایگاه داده انتخابی ما مای‌اس‌کیوال است که یک پایگاه داده رابطه ای است و یکی دیگر از دلایل انتخابی این پایگاه داده عملکرد و انطباق خوبی که با زبان برنامه نویسی سمت سرور ما یعنی پی اچ پی دارد.

ما با توجه به نیاز هایی که برای ذخیره سازی یا کار با دیتا داریم میایم شروع میکنیم به طراحی دیتابیس

به طور مثال ما نیاز به جدولی داریم که بتوانیم محصولات خود را در ان ذخیره کنیم یا حذف کنیم یا ...

و به طور کلی عملیات CRUD را روی اون انجام دهیم

عملیات CRUD چیست :

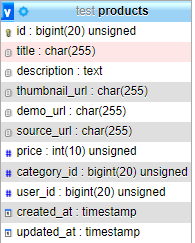
1. Create ساختن
2. Read خواندن
3. Update بروزرسانی
4. Delete حذف

به این چهار عمل اصلی کار با دیتا عملیات CRUD می گویند.

خب با توجه به عنوان پروژه ما که یک فروشگاه محصولات دیجیتال هست و یا به اختصار نام دیگر ان یک گالری انلاین است شروع به طراحی دیتابیس میکنیم

### **جدول Product**

یکی از نیاز هایی که این فروشگاه ما دارد این است یک جدول برای ذخیره سازی محصولات میخواهد



1. جدول محصولات

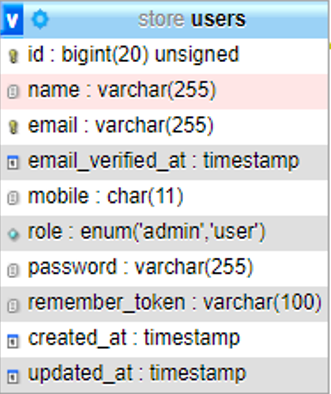
نمای کلی از جدول محصولات رو در بالا مشاهده میکنید .

که در ادامه شروع میکنیم به تشریح تک تک ستون ها

1. id : این ستون کلید اصلی جدول ماست و برای هر محصول یکتاست
2. Title : این ستون از جدول برای دخیره سازی عنوان هر محصول به کار میرود
3. Description : این قسمت برای ذخیره توضیحات مورد نیاز برای هر محصول به کار می رود
4. Thumbnail : این ستون برای ذخیره مسیر تصویر کوچک معرفی محصول به کار میرود
5. Demo : این ستون برای ذخیره سازی عکس دمو محصول به کار میرود
6. Source : این ستون برای ذخیره سازی سورس دانلودی محصول به کار میرود
7. Price : این ستون برای ذخیره سازی قیمت محصول به کار میرود
8. Category : این ستون برای ذخیره سازی دسته بندی محصول به کار میرود
9. User\_id : این ستون برای ذخیره سازی ایدی کاربر صاحب محصول به کار می رود

### **جدول Users**

از این جدول برای ذخیره سازی کاربران سایت استفاده میشود.



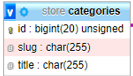
2. جدول کاربران

تشریح ستون های جدول :

1. Id : این ستون کلید اصلی جدول ماست و برای هر کاربر یکتاست
2. Name : در این ستون نام کاربر ذخیره میشود.
3. Email : در این ستون ایمیل کاربر ذخیره میشود.
4. Email\_verified\_at: در این ستون زمان تایید شدن ایمیل کاربر ثبت میشود و اگر مقدار ان خالی باشد به معنای تایید نشده بودن ایمیل کاربر است
5. mobile : در این ستون موبایل کاربر ذخیره میشود.
6. Role : در این ستون نقش کاربر که ادمین است یا کاربر معمولی ذخیره میشود.
7. Password : در این ستون پسورد کاربر به صورت رمز نگاری شده ذخیره میشود.
8. remember\_token : محل ذخیره سازی remember\_token
9. Created\_at : در این ستون زمان ایجاد کاربر ذخیره میشود.

### **جدول Categories**

از این جدول برای ذخیره کردن دسته بندی محصولات استفاده میشود.



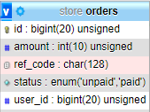
3. جدول دسته بندی ها

تشریح ستون های جدول :

1. Id : این ستون کلید اصلی جدول ماست و برای هر دسته بندی یکتاست
2. Slug : در این ستون اسلاگ هر دسته بندی ذخیره میشود.
3. Title : در این ستون عنوان هر دسته بندی ذخیره میشود.

### **جدول Orders**

از این جدول برای ذخیره سازی سفارش ها استفاده می شود.



4 . جدول سفارش ها

تشریح ستون های جدول :

1. Id : این ستون کلید اصلی جدول ماست و برای هر سفارش یکتاست
2. Amount : مجموع هزینه سبد خرید
3. Ref\_code : اینم یک کد یکتا است که ما خودمون برای هر تراکنش درست میکنم
4. Status : وضعیت هر سفارش است که یا به صورت پرداخت شده است یا پرداخت نشده
5. User\_id : ایدی کاربری است که سفارش رو ایجاد کرده

### **جدول order\_items**

از این جدول برای ذخیره سازی ایتم های هر سفارش استفاده می شود.



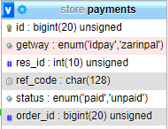
5 . جدول اتیم های هر سفارش

تشریح ستون های جدول :

1. Id : این ستون کلید اصلی جدول ماست و برای هر محصول سفارش یکتاست
2. Price : قیمت هر محصول به صورت جدا
3. Order\_id : بیان میکنه که این ایتم مربوط به کدوم سفارش است.
4. Product\_id : بیان میکنه که این ایتم مربوط به مدوم محصول است.

### **جدول payments**

از این جدول برای ذخیره سازی پرداخت ها استفاده می شود.



6 . جدول پرداخت ها

تشریح ستون های جدول :

1. Id : این ستون کلید اصلی جدول ماست و برای هر پرداخت یکتاست
2. Gateway : درگاه پرداختی رو که کاربر استفاده کرده برای پرداخت رو در این ستون ذخیره میکنیم
3. Res\_id : وقتی پرداختی با موفقیت همراه بود براش یک کد پیگیری تولید میکنیم و در این ستون ذخیره میکنیم
4. Ref\_code : اینم یک کد یکتا است که ما خودمون برای هر تراکنش درست میکنم
5. Status : وضعیت هر پرداخت که به صورت پرداخت شده و نشده ذخیره میگردد.
6. Order\_id : کلید خارجی سفارش در ایجا ذخیره میشود تا بفهمیم هر پرداخت مربوط به کدام سفارش بوده.

در این سرفصل ما به تشریح جداول پروژه پرداختیم و دلیل ایجاد هر جدول و هر ستون رو توضیح دادیم در ادامه به بخش طراحی ظاهری سایت یا frontend میرسیم.

## **طراحی ظاهری سایت**

به عنوان یک طراح تجربه کاربری ، قبل از اینکه شروع به طراحی کنیم ، همیشه باید حواسمان به این باشه که دقیقا باید چه مساله ای را حل کنم .

باید بفهمم مشکلات و نیاز ها و اهداف چیه و من باید برای چه چیزی بجنگم.

"اهداف" برای من به عنوان یک طراح همه چیزه و همیشه باید بدونم که "چرا" و برای چه "هدفی" میخوام طراحی کنم.

شناسایی دقیق "اهداف" به ما کمک میکنه تا استراتژی و مسیر درستی رو برای پروژه تدوین کنم و از هدر رفت زمان و پول و سرمایه بیزینس و همینطور کاربران جلوگیری کنم.

ما باید جوری رابط کاربری پروژه رو پیاده ساز کنیم تا کاربر به واضح ترین شکل بتواند به اون چیزی که نیاز داره برسه و در راه رسیدن به هدفش به واسطه پیچیدگی های احتمالی در طراحی از کاری که میخواست بکنه باز نمونه

همچنین میتوانیم از تجربه بقیه وبسایت هایی که همانند ما در این زمینه فعالیت میکردن استفاده کنیم

چون معمولا کاربر ها با طراحی های مشابه ای که از ان ها تجربه خوبی داشتن استقبال میکنن

یک نمونه موفق و معروف وبسایت دیجی کالا است که ما تصمیم گرفتیم تا حد زیادی طراحی وبسایتمون رو مثل اون انجام بدیم

برای طراحی این پروژه تصمیم داریم که از فریم ورک طراحی BOOTSTRAP استفاده کنیم.

در ادامه توضیح مختصری راجب به این فریم ورک ارائه میدهیم.

### **BOOTSTRAP چیست ؟**

بوت استرپ یک فریم ورک طراحی وب سایت است که در ابتدا توسط توسعه دهندگان توییتر برای استفاده داخلی ایجاد شد. bootstrap پس از مدتی در دسترس عموم قرار گرفت و به مجموعه‌ای مناسب از ابزارها برای توسعه رابط‌های کاربری با هر میزان پیچیدگی تبدیل شد.

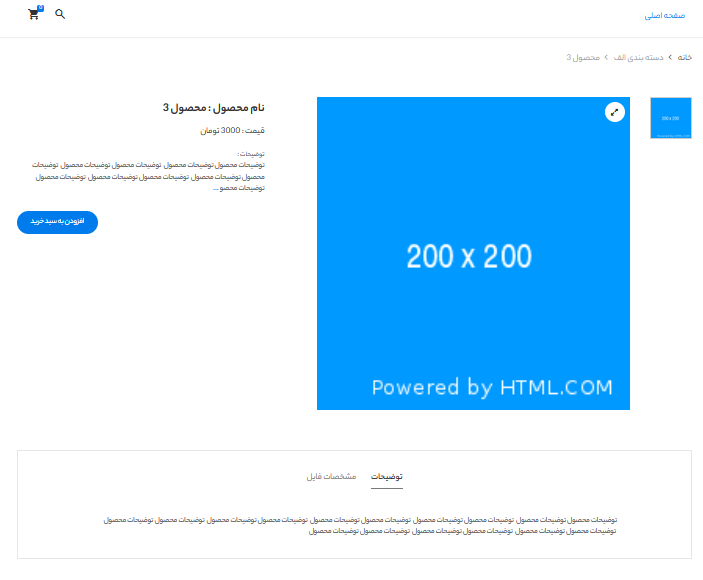
اکنون Bootstrap یک مجموعه ابزار HTML ، CSS و JS منبع باز و رایگان است. هنگامی که توسعه دهنده وب Bootstrap این فایل‌ها را به صفحه متصل می‌کند، تعداد زیادی کلاس و اجزای آماده برای طرح بندی در دسترس قرار می‌گیرد. از بوت استرپ برای ایجاد و طراحی وب سایت‌های ریسپانسیو در زمان کم استفاده می‌شود.

بوت استرپ توابع و دستورات مناسب را ایجاد کرده و در اختیار برنامه نویسان قرار می‌دهد. این کار باعث می‌شود نه تنها زمان طراحی و ایجاد وب سایت کاهش یابد بلکه نتیجه نهایی مطابق با استانداردهای جهانی باشد. به همین دلیل بوت استرپ از اهمیت بالایی در زمینه طراحی و تولید وب سایت برخوردار است.

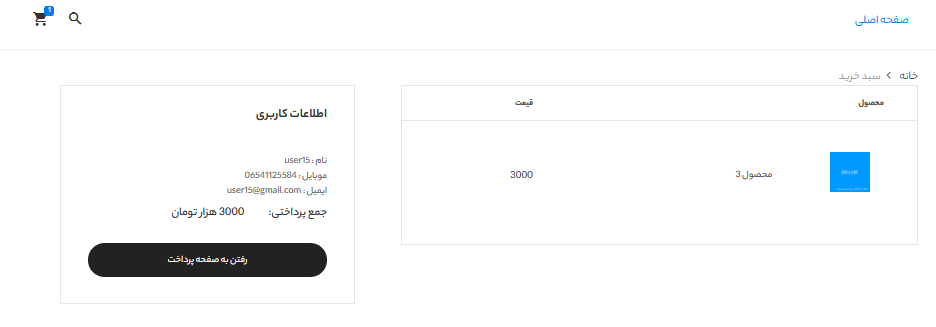
مهم‌ترین کاربرد بوت استرپ این است که با استفاده از آن، توسعه دهندگان وب می‌توانند بر کار توسعه تمرکز کنند، بدون اینکه نگران طراحی باشند و به سرعت یک وب سایت زیبا را طراحی و راه اندازی کنند.



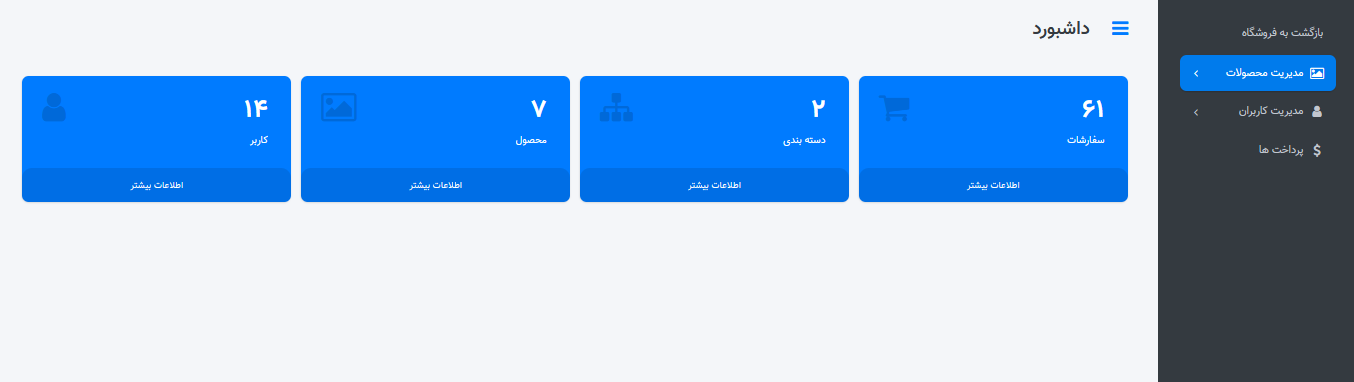
7 . صفحه اصلی فروشگاه



8 . صفحه تکی محصول



9 . سبد خرید و پرداخت



10 . پنل ادمین



11 . پنل کاربر

در این سرفصل به معرفی طراحی ظاهری وبسایت پرداختیم و قسمت هایی مختلف سایت رو در قالب عکس نمایش دادیم در ادامه به قسمت کد نویسی و backend سایت میرسیم

## **BACKEND**

### **Backend چیست ؟**

بک اند، به بخشی از یک وب سایت یا نرم افزار می‌گویند که برای کاربران قابل مشاهده نیست. به عبارتی دیگر هسته و مغز یک سایت است که وظیفه کنترل منطق آن را بر عهده دارد. سایت‌های دینامیک به برنامه نویس بک اند نیاز دارند تا منطق سایت را به وسیله زبان‌های برنامه نویسی پیاده‌سازی کنند. کاربران به کدهای نوشته‌شده در بک اند دسترسی ندارند و نمی‌ توانند آن‌ها را مشاهده کنند.

این بخش از سایت مانند قسمتی از کوه یخ است که در زیر سطح آب قرار گرفته است. سمت سرور با بخش سمت کاربر ارتباط مستقیم دارد و به اجزایی که در رابط کاربری طراحی شده‌اند جان می‌بخشد. برنامه نویس بک اند باید اطلاعات را متناسب با اهداف مختلف از پایگاه‌داده دریافت کند و در صورت نیاز پس از پردازش به کاربر نمایش دهد. بنابراین Backend از دو بخش منطق سایت و پایگاه داده تشکیل شده است.

مثالی از زبان های برنامه نویسی سمت Backend :

* PHP
* C#
* GOLANG
* JAVA
* PYTHON

### **زبان برنامه نویسی پی اچ پی**

زبان برنامه نویسی انتخابی ما برای انجام این پروژه زبان پی اچ پی است.

در ادامه درباره با این زبان آشنا می شویم :

یک زبان تفسیرشونده است که اساس و بنیان آن طراحی و ایجاد صفحات پویای وب بوده است. به عبارتی دیگر، کسانی که وظیفه کدنویسی سایت را بر عهده دارند با استفاده از زبان پی اچ پی قادر خواهند بود وب اپلیکیشن‌هایی طراحی کنند که به صورت تعاملی می‌باشند. منظور از تعاملی بودن وجود قابلیت هایی مانند سطوح دسترسی مختلف برای کاربران، امکان ثبت‌نام در سایت، پنل کاربری ، ذخیره اطلاعات کاربران و بسیاری از امکانات دیگر است که امروزه در تمامی سایت‌های امروزی مشاهده می‌کنیم.

از نقاط قوت این زبان می‌توان به وجود فریمورک‌های بسیار قدرتمند و محبوبی هم چون لاراول و سیمفونی، دارا بودن بیش از 80% از سهم بازار وب ، متن باز و رایگان بودن ، جامعه بزرگ و پشتیبانی خوب اشاره کرد. از طرفی پی اچ پی توانسته است خودش را به عنوان یکی از محبوب‌ترین زبان‌های برنامه نویسی در حوزه طراحی وب معرفی کند.

### **شی گرایی**

از آنجایی که ما سایت رو با زبان پی اچ پی و به صورت شی گرا توسعه داده ایم اکنون توضیح مختصری راجب این پارادایم برنامه نویسی میدهیم

شی گرایی چیست؟

خیلی از شما ممکنه اسم شی گرایی رو شنیده باشید و براتون سوال باشه که معنیش چیه. این عبارت در برنامه نویسی استفاده میشه ولی مفهومش کلیه و ما همه جا بدون اطلاع از اون استفاده می کنیم.

تعریف کلی شی گرایی

شی گرایی: تعریف یک سری ویژگی یا رفتار و نسبت دادن اون ویژگی و رفتار ها به چیز های مختلف.

مثال:

ما تو ذهنمون یه تعریف از ماشین (خودرو) داریم: خودرو چیزیه که رنگ، سرعت، قدرت، شرکت سازنده داره و قابلیت حرکت، ترمز کردن، تعویض دنده رو داره.

وقتی می گیم 《پراید یک خودرو است.》تعریف ما از خودرو در ذهنمون ظاهر میشه و می فهمیم پراید چیه و چه ویژگی یا رفتار هایی داره. این یعنی شی گرایی. همون طور که اوایل مقاله گفتم، ما اینجا یه سری ویژگی و رفتار برای خودرو تعریف کردیم و اون رو به پراید نسبت دادیم. به همین سادگی

### **معماری MVC**

از آنجایی که ما برای طراحی سایت از معماری MVC استفاده کرده ایم اکنون مقداری راجب این معماری توضیح میدهیم.

MVC چیست؟

خیلی از جاها MVC رو یک الگوی طراحی مینامند(Design Pattern) ولی در حقیقت، MVC یک معماری نرم افزار است، نه یک الگوی طراحی، چرا؟ اگر بخوام در کوتاه ترین حالت، و در عین حال ملموس ترین حالت ممکن توضیح بدم، معماری نرم افزار ساختار و استراکچر اصلی پروژه نرم افزاری مارو مشخص میکند، اما دیزاین پترن ها تنها راه حل هایی هستند، برای مشکلات رایج در برنامه نویسی شی گرا.

این معماری نرم افزار که مخفف Model-View-Controller هست، همونطور که میبینید از قانون

Separation of concern پی روی میکند، یعنی چی؟ یعنی اینکه نگرانی های ما (همون Concern ها) که در

اینجا منظورش همین سه بخش نرم افزاری هست، رو از هم جدا میکنه، چه مزیت هایی داره این کار؟!

زمانی که شما این سه بخش نرم افزاری رو از هم جدا میکنید، ساختار پروژتون قابل درک تر میشه (برای یک شخص سومی که قراره با کد های شما کار کنه)، زمانی که به مشکل برخوردید خیلی راحت میتونید مشکل رو رفع کنید، چون بخش های مختلف از هم جدا هستند و شما فقط به سراغ همان بخشی میروید که مشکل دارد، هزینه نگهداری کد ها کاهش میابد، سرعت توسعه افزایش میابد، تایم توسعه کاهش میابد، سرعت دیباگینگ افزایش میابد، خطاهایی که به دلیل تمیز نبودن کد و یا در هم بودن کد ها به وجود آمدند، کاهش میابند و . . .

# **فصل چهارم**

**نتیجه گیری**

## **نتیجه گیری**

امروزه ما با کمک وبسایت ها توانسته ایم بسیاری از امور زندگی خود را بهبود ببخشیم

همانطور که در این گزارش کار خوانده ایم ما با استفاده از زبان برنامه نویسی پی اچ پی توانستیم وبسایتی را خلق

کنیم که بتواند به جامعه هنرمندان از جمله عکاس ها در فروش آثار هنری خود کمک کند.

# پیوست ها

ندارد.

# فهرست منابع

ندارد.

پایان