پروپوزال

مهندسی

نرم افزار 2

گروه F

محمد دولتشاه

ایرج احمدی

علی اتقایی

احمد نورزایی

پویا محصوری

**مقدمه**

* خلاصه

همواره انتخاب کردن یا به دست آوردن یک ایده ی خوب برای طراحی یک مشکل اساسی برای شرکت ها یا موسسات بوده وهست .

اگر امکانی برای این شرکتها فراهم میشد که به جای سفارش دادن لوگو ( یا طرح سایت یا تبلیغات …) به یک فرد به صورت انحصاری  , بتوانند از میان چند ایده بهترین را انتخاب کنند این مشکل برطرف میشد.

در واقع هدف پروژه ما استفاده از طراحان فری لنسر برای طراحی لوگو , سایت , اپلیکیشن , تبلیغات , لباس , بسته بندی , هنر , مجله و کتاب به صورت جمع سپاری  برای خدمت رسانی به نیازمندان این خدمات است.

این کار به صورت طراحی و پیاده سازی یک سایت امکان پذیر است به این صورت که طراحان freelancer  ( کارمزدی )میتوانند در سایت ثبت نام کنند و مهارت تخصص خود را بیان کنند , سپس کارفرمایانی که به طرحی برای موسسه خود نیاز دارند پروژه مورد نیاز خود را در سایت تعریف میکنند تا پیمانکاران سایت که همان طراحان باشند برای طرح مورد نظر یک طرح اولیه ثبت کنند سپس کارفرما میتواند از بین چندین طرح موجود بهترین ( یا چندین طرح در صورت بزرگ بودن پروژه ) را برای ادامه کار انتخاب کند .

بعد از انجام کامل پروژه مبلغ تعیین شده برای پروژه را برای طراح ( یا طراحان ) آزاد کند.

در واقع عمل طراحی به صورت جمع سپاری انجام خواهد شد.

* دامنه

دامنه ی افراد مرتبط با این سیستم محدود به قشر خاصی از جامعه نمی شود و تمام افرادی که برای پیشبرد کار و پروژه ی خود نیاز به یک طرح گرافیکی دارند و همچنین تمام افرادی که توانایی خلق ایده ها ی زیبا با توجه به نیازهای کارفرمایان و پیاده سازی آن در قالب یک طرح گرافیکی را دارند, با این سیستم در ارتباط خواهند بود.

به طور مثال: گرافیست ها, شرکت های تازه تاسیس که نیاز به داشتن یک لوگو دارند, شرکت ها و نهاد هایی که نیاز به طراحی وب سایت دارند, تولیدی های البسه که نیاز به ایجاد طرح برای تولیدات خود هستند و ...

* ابعاد محصول

زمان:

طبق بررسی ها وصحبت های انجام شده با اعضای گروه پیاده سازی و تیم تحقیقاتی در

نهایت تیم به این نتیجه رسید که کلیه عملیات پیاده سازی پروژه ظرف مدت 3 ماه انجام

شود که این 3 ماه شامل موارد زیر است :

1) ماه اول شامل طرح ریزی برای فاز اول پروژه شامل طراحی کامل پروژه من جمله

طراحی مدل ها و کلاس های اجرایی پروژه , طراحی دیتابیس و معماری اصلی پروژه ,

تعیین روند کلی اجرایی اجزای پروژه , مشخص کردن عملکرد تمامی فانکشن ها و

جزئیات کلاس ها , تعیین تمامی ویژگی ها و مشخصات اکشن های کاربرها , تعیین روند

اجرایی مراحل انجام پروژه توسط کارفرما یا پیمانکار , تعیین تواناییهای کاربرهای پروژه

به صورت دقیق و با تمام جزئیات

2) ماه دوم شامل پیاده سازی سطح پایین موارد مطرح شده در معماری است .

مثل پیاده سازی دیتابیس و مدل ها وکلاس های entity , پیاده سازی فایلهای اصلی سرور

شامل فایلهای اجرایی کنترلرها ,پیاده سازی فایلهای مربوط به backend , پیاده سازی

فایلهای اجرایی سرور , پیاده سازی تمامی کلاسها وفرم های لازم , مشخص کردن تمام

تنظیمات برنامه شامل تنظیمات سرور و شبکه , پیاده سازی فرم ها و ظاهر گرافیکی ساده

برای آنها

3) ماه سوم پیاده سازی شامل پیاده سازی جزییات متکی بر ظاهر پسند بودن پروژه است

که شامل پیاده سازی دقیق تر جزییات اکشن ها برای کاربرها , پیاده سازی امکانات فرعی

برای سیستم , پیاده سازی جزییات و کاربر پسند کردن پروژه , بررسی دوباره و تست

فانکشن ها و اکشن های پروژه ,

بعد از عملیات پیاده سازی پروژه مدت 1 ماه به عملیات تست پروژه اختصاص داده میشود

که شامل تست کردن تمامی امکانات پروژه شامل تست منطقی تمام فایلهای پروژه , تست

اکشن ها و عملکرد های اجرایی کاربران , بازبینی دوباره اشکالهای ریز پروژه یا همان

debug کردن پروژه , بررسی اشکال های نحوی پروژه به همراه رفع مفاهیم گنگ و

مبهم , ست کردن exception handeling های ست نشده در مرحله های قبل ,

سپردن پروژه به کاربران برای تست و نظر دهی دوباره

بعد از کامل شدن مراحل ذکر شده میتوان پروژه را به صورت ورژیون بتا پروژه به بازار

معرفی کرد که خود شامل تهیه امکانات سخت افزاری و شبکه ای برای پروژه و ارائه

تبلیغات برای پروژه میشود .

این مرحله زمان زیادی لازم دارد تا محصول بتواند جای خود را در بازار هدف ثابت کند

که زمان در نظر گرفته برای آن 3 ماه میباشد که تبلیغات و بهینه سازی برای موتورهای

جست و جو هم شامل آن میباشد

در این مرحله عملیات تست در اسکیل بزرگتر انجام میشود و کابران بازار هدف با ارتباط

برقرار کردن با نرم افزار میتوانند اشکالات نرم افزار را مشخص کرده یا با نظر گرفتن

از آنها میتوان نرم افزار را ارتقاع داد

بعد از ورژن بتا نوبت به پیاده سازی ورژن آلفا میرسد.

این قسمت از پیاده سازی مدت 1 ماه طول کشیده خواهد شد . که با استفاده از اشکالات

مشخص شده در ورژن بتا و نظرات کاربران این پروسه انجام خواهد پذیرفت

در پایان این مرحله محصول نهایی و بدون نقص به بازار وارد خواهد شد

بعد از بیرون آمدن ورژن نهایی مرحله پشتیبانی نرم افزار شروع میشود

این مرحله شامل در ارتباط بودن با کاربران نرم افزار از طریق راه های ارتباطی شامل

چت یا ایمیل تماس تلفنی میباشد.

این مرحله نیاز مند یک تیم پشتیبانی مستقل است که عملیات کمک و راهنمایی یا اطلاع

رسانی به کاربران را بر عهده میگیرند.

زمان اجرای این مرحله بستگی به زمان اجرایی خود نرم افزار دارد که ممکن است تا

چندیدن سال هم طول بکشد در واقع تا موقعی که نرم افزار وجود دارد تیم پشتیبان هم باید

وجود داشته باشد

تعداد افراد :

تعداد افرادی که برای قسمت پیاده سازی پروژه با پروژه درگیر هستند شامل 4 نفر از

اعضای اصلی گروه است که این 4 عضو وظیفه پیاده سازی اصلی فایلهای زیرین و سرور

ساید را به عهده دارند همچنین پیاده سازی سطحی و معقول رابط گرافیکی که برای یک

کابر پیشرفته قابل درک باشد.

بعد از پیاده سازی اصلی دو نفر طراح به گروه اضافه خواهند شد که در مجموع 6 نفر کار

طراحی و پیاده سازی ظاهر حرفه ای و مورد قبول نرم افزار را به عهده میگیرند.

این مرحله شامل بهینه کردن کاربر پسند بودن نرم افزار است.

بعد از بیرون آمدن ورژن نهایی نرم افزار پروژه به یک تیم پشتیبانی نیاز دارد که تعداد

افراد این تیم به بزرگ شدن بازار نرم افزار بستگی دارد .

افرادی که برای پشتیبانی در نظر گرفته شده است برای هر هزار کاربر و یا هر 200

کاربر فعال نرم افزار 50 ساعت نفر در هفته میباشد .

تکنولوژی :

تکنولوژی به کار رفته در نرم افزار استفاده از یک زبان برنامه نویسی مفسر و فریم ورک وب تحت آن زبان میباشد.

زبان مورد نظرPython3 و فریم ورک Django1.5 میباشد.

جنگو (به انگلیسی: Django) یک چارچوب نرم‌افزاری تحت وب آزاد و متن‌باز است که به زبان پایتون نوشته شده است و از معماری مدل-نما-کنترل‌گر (به انگلیسی: Model–view–controller) پیروی می‌کند. جنگو اولین بار برای مدیریت صفحات خبری وبسایت شرکت Lawrence Journal-World توسعه داده شد و سپس در سال ۲۰۰۵ تحت اجازه‌نامه بی‌اس‌دی منتشر شد

هدف اصلی جنگو ساخت آسان سایت‌های پیچیده و وابسته به دیتابیس است و بر پایهٔ قابلیت استفادهٔ مجدد و قابل اتصال بودن اجزای مختلف، توسعهٔ سریع و اصل خودت تکرار نکن (DRY) طراحی شده است. جنگو سراسر از پایتون استفاده می‌کند، حتی برای تنظیمات، فایل‌ها و مدل‌های اطلاعات.

برای سطح دیتابیس از تکنولوژی دیتابیس های رابطه ای استفاده میشود که زبان مورد

استفاده ما زبان MySql است.

برای سطح کاربری از تکنولوژی HTML 5 و CSS3 و jQuery 11 و

Bootstrap 3 استفاده میشود

* مسائل حقوقی

همانطور که گفته شد نرم افزار ما تحت لیسانس جنگو منتشر میشود که خود تحت اجازه‌نامه

بی‌اس‌دی منتشر شده است.

علاوه بر مساله تمام حقوق کپی رایت نرم افزار متعلق به اعضای گروه پیاده ساز میباشد.

**ویژگی های اصلی**

* موارد فنی

امکانات اضافه ای که قرار است در پروژه قرار داده شود شامل موارد زیر است:

* امکان چت کردن کارفرما با پیمانکاران پروژه به صورت مجزا
* امکان به اشتراک گذاری فایل بین کارفرما و پیمانکاران
* امکان ادیت پروفایل کاربری
* امکان آپلود رزومه پیمانکاران
* امکان رتبه دادن به کارفرما و پیمانکاران
* امکان شکایت کردن  از طرفین پروژه در صورت ناراضی بودن از کار
* امکان تقسیم کار و مدیریت زمان و هزینه  توسط کارفرما
* امکان مشاهده روند اجرایی پروژه به صورت آماری
* فرستادن ایمیل پروژه های جدید به صورت هفتگی به پیمانکاران سایت بر اساس مهارت های وارد شده
* امکان شکایت کردن  از طرفین پروژه در صورت ناراضی بودن از کار
* امکان امتیاز بندی برای کافرمایان و پیمانکاران در صورت دعوت کردن افراد دیگر یا به اشتراک گذاری سایت
* اضافه کردن مهارت ها یا رشته ی تحصیلی و فعالیت های کاری
* زمانبندی

با بررسی های انجام شده توسط اعضای گروه، ما برنامه داریم که در مدت قریب به 1 ماه فاز اول پروژه را مطابق با معماری ها و متد های معرفی شده انجام دهیم و بعضی از امکانات گفته شده را هرچند در سطح کم پیاده سازی نماییم.

در این فاز مسئله کاملا با پلان و طرح نرم افزاری روبروی ما کلید می خورد و شروع می شود و نقاط قوت و ضعف آن معلوم می شود و در واقع زیرساخت ها فراهم می گردد و ابهامات برطرف می شود.

در ماه بعد فاز بعدی پروژه ی ما طبق برنامه طراحی می گردد.در این فاز امکانات جدی تر و پیچیده تر گفته شده را پیاده سازی می کنیم به نحوی که تقریبا کار های اصلی شرح داده شده به طور کامل و به درستی اجرا شوند و مشکلات و معایب و نقاط قوت نهایی معین میگردد.

در واقع نسخه ی بتای این پروژه در این فاز ارائه می شود تا با نقاط ضعف و قوت آشنا شویم و در صدد اصلاح آن برآییم.

در ماه سوم کل پروژه باید تکمیل گردد.بدین صورت که با بررسی نقطه نظرات و انتقادات و پیشنهادات به اصلاح جای جای مورد بحث در پروژه بپردازیم.

در این فاز پروژه ی نهایی به عنوان محصول نرم افزاری تحویل داده می شود و وب سایت به طور کاملا دقیق همانطور که انتظار می رود و شرح داده شده کار می کند و به انتقادات مشتریان و نیازهای استفاده کنندگان نسبت به نسخه ی قبلی به درستی و به طور کامل پاسخ داده می دهد.

* **ابزارهای مورد استفاده**

1. استفاده از یک ویرایشگر کد برای نوشتن سورس کد پروژه و اصلاح آن (Eclipse)
2. فریم ورک جنگو (Django Framework)
3. استفاده از پایگاه داده (MySQL or PostgreSQL)

**سامانه مدیریت محتوا**

سامانه مدیریت محتوا به انگلیسی: Content Management System و به اختصار CMS مجموعه‌ای از رویه‌ها برای مدیریت روند کار در یک محیط مشارکتی است. این رویه‌ها می‌توانند دستی یا رایانه‌ای باشند. این رویه‌ها طراحی می‌شوند که:

* افراد زیادی بتوانند مشارکت کنند و داده‌های مختلفی را به اشتراک بگذارند
* داده‌ها دارای کنترل دسترسی براساس نقش کاربران باشند. نقش کاربران تعیین می‌کند که چه کاربری چه چیزی را می‌تواند ویرایش کند یا ببیند
* ذخیره‌سازی و بازیابی داده‌ها را آسان کند
* ورودی‌های مشابه تکراری را کاهش دهد
* گزارش‌نویسی را بهبود بخشد
* ارتباط بین کاربران را بهبود بخشد

در یک CMS ، داده می‌تواند هر چیزی باشد، مانند اسناد، فیلم، عکس، شماره تلفن، داده‌های علمی و غیره.

CMS ها بیشتر برای ذخیره، کنترل، بازبینی، توسعه معنایی و انتشار اسناد به‌کار می‌رود.

Django نوعی CMS است.

**Django چیست ؟**

[](http://www.mspsoft.com/wp-content/uploads/2011/08/django-logo-positive1.png)

**جنگو** به انگلیسی Django یک چارچوب نرم‌افزاری تحت وب [آزاد و متن‌باز](http://fa.wikipedia.org/wiki/%D8%A2%D8%B2%D8%A7%D8%AF_%D9%88_%D9%85%D8%AA%D9%86%E2%80%8C%D8%A8%D8%A7%D8%B2) است که به زبان پایتون نوشته شده است و از معماری (مدل-نما-کنترل‌گر) به انگلیسی Model–view–controller پیروی می‌کند.

**جنگو** اولین بار برای مدیریت صفحات خبری وبسایت شرکت Lawrence Journal-World توسعه داده شد و سپس در سال ۲۰۰۵ تحت اجازه‌نامه بی‌اس‌دی منتشر شد. نام جنگو ازجنگو راینهارت، نوازنده گیتار جاز گرفته شده است.

هدف اصلی جنگو ساخت آسان سایت‌های پیچیده و وابسته به دیتابیس است و بر پایه قابلیت استفاده مجدد و **قابل اتصال بودن** اجزای مختلف، توسعه سریع و اصل خودت تکرار نکن (DRY) طراحی شده است. جنگو سراسر از پایتون استفاده می‌کند، حتی برای تنظیمات، فایل‌ها و مدل‌های اطلاعات ، یکی از نکات قوت این سیستم توانایی آن در ساخت سایت ها و برنامه های کاربردی وب با ساختار پایگاه داده پیچیده می باشد .

 این فریم ورک سطح بالا به زبان پایتون برای وب می باشد که امکان طراحی و ایجاد بسیار سریع و آسان برنامه های تحت وب را فراهم می کند .

**پروانه بی‌اس‌دی** به انگلیسی BSD License مخفف پروانه نرم‌افزار توزیع برکلی است. از جمله پراستفاده‌ترین مجوزهای نرم‌افزارهای آزاد است.این مجوز اولین بار برای توزیع برکلی به طور خلاصه نسخه‌ای از یونیکس است استفاده شده‌ است.

**جنگو یک فریم ورک سطح بالا با زبان پایتون می باشد**

جنگو باعث سهولت ایجاد وب سایت هایی داینامیک می شود . بسیاری از موارد برنامه نویسی را بصورت خودکار فراهم کرده و در اختیار برنامه نویس قرار می دهد . همچنین روش هایی میانبر و واسط برای اجرای اعمال مختلف را دارا می باشد . پس برنامه نویس نیازی به نوشتن کد هایی اضافی و مشخص را ندارد .

یک وب سایت دایانامیک و پویا وبسایتی هست که شامل کد های ساده Html نبوده بلکه بصورت پویا و توسط نرم افزار

(web application) تگ های اچ تی ام ال و در حقیقت صفحات سایت از روی کد ها و دستورات نوشته شده توسط برنامه نویس ایجاد می گردد . برای مثال ممکن است بر اساس در خواست کاربر اطلاعاتی بصورت داینامیک از پایگاه داده دریافت و نمایش داده شود .

**نکات کلیدی**

جنگو مجموعه ای از کتابخانه های نوشته شده در زبان برنامه نویسی پایتون است، برای گسترش سایت مورد نظر با جنگو باید کد پایتون زد که از کتابخانه های jango استفاده می کند.

در یک کلام : جنگو = نوشتن برنامه به زبان پایتون + کار با کتابخانه های جنگو

در هر زمان دو نسخه از جنگو در دسترس است :

1. انتشار آخرین نسخه رسمی (تست شده و ثابت از جنگو)
2. نسخه .trunk (حاوی آخرین تغییرات و خصوصیات)

بسته به کار مورد نظر هر یک از نسخه ها انتخاب می شود اما ما در این پروژه از آخرین نسخه رسمی استفاده می کنیم.

در سیستم عامل ویندوز فایل Django2.0.zip را از حالت فشرده خارج کرده و در محیط cmd دایرکتوری که با نام ... Django - 2.0 شروع می شود می نویسیم :

Python setup.py install

فایل های جنگو درون دایرکتوری site – packages نصب می شوند . یک دیکشنری جایی که در آن جا کتابخانه های third – party را جستجو کند معمولا در جایی مثل آدرس زیر قرار دارد :

./usr/lib/python2.7/site – packages

**ویژگی های یک فریم ورک خوب عبارتند از** :

**مکانیسمی برای نگاشت URL درخواست شده به کدهایی که وظیفه پاسخ دادن به آن درخواست را دارند**

به عبارت دیگر امکان تعیین اینکه کدام کد یا تابع مسئول جواب دادن به آدرس درخواست شده می باشد . برای مثال می توان از فریم ورک خواست تا برای آدرسهایی شبیه به /users/joe/ کدی را اجرا کند که اطلاعات کاربران را نمایش می دهد .

**تسهیل نمایش ، اعتبار سنجی و نمایش مجدد فرم های html**

فرم های اچ تی ام ال یکی از مهترین روش ها برای دریافت اطلاعات از یک کاربر وب می باشد . پس یک فریم ورک خوب باید دارای روش هایی مناسب برای نمایش و کار با این فرم ها باشد . (برای مثال امکان مشخص کردن خطاهای رخ داده)

**تبدیل اطلاعات وارد شده توسط کاربر به ساختاری مناسب برای مدیریت و استفاده**

برای مثال تبدیل اطلاعات وارد شده در یک فرم اچ تی ام ال به ساختاری متناسب با زبان مورد استفاده

**امکان جداسازی محتوا از ظاهر نمایشی**

در اکثر زبانهای موجود برنامه نویسی کد های اچ تی ام ال و در حقیقت ظاهر یا template سایت با کد ها و محتوای سایت آمیخته می گردد که باعث سختی و ایجاد مشکلاتی هنگام تغییرات بعدی و نگهداری می گردد . با استفاده از این روش ظاهر نمایشی سایت بصورت جداگانه در فایلی خاص ذخیره می گردد . اکنون با اعمال تغییر در هر کدام نیازی به ویرایش دیگری نیست و این دو مجودیتی مستقل می باشد .

**امکان درونی سازی آسان با لایه اطلاعات ذخیره شده (storage)**

در حالت معمول هر برنامه نوشته شده با جنگو دارای ۳ لایه مهم می باشد بخش نمایشی (template)، محتوا یا کد های کنترلی و بخش ذخیره دائمی اطلاعات(برای مثال پایگاه داده) . البته بسته به نوع برنامه و خواست برنامه نویس می توان این لایه ها را ادغام کرده یا نادیده گرفت .

**امکان استفاده موثر از سطح بالایی از تجرید و انتزاع**

جنگو در موارد زیادی با استفاده از مفهوم انتزاع سهولت بسیاری را فراهم کرده است . برای مثال برای کار با تکنولوژی چون ftp یا http با یک مفهوم انتزاعی و سطح بالا روبرو خواهید بود که با استفاده از روابط و توابع متعدد برنامه نویسی را بسیار آسان و قدرتمند می کند . البته امکان نفوذ و استفاده از سطوح پایین تر نیز فراهم است .

**حذف موارد و مشکلات معمول در برنامه نویسی وب**

حذف و رفع موارد و مشکلات معمول در برنامه نویسی سمت وب . برای مثال حذف پسوند فایل از آدرس های وب (.aspx) یا (.php)  
فریم ورک django تمامی موارد فوق را دارا می باشد . این فریم ورک با استفاده از زبان زیبا ، قدرتمند و سطح بالای پایتون نوشته شده است . برای ایجاد وب سایت برنامه نویس کدهای پایتونی می نویسد که امکان استفاده از ویژگی های پایتون و کتابخانه های این زبان را دارد .

**تاثیر پایتون بر این فریم ورک**

کل این فریم ورک با استفاده از زبان پایتون پیاده سازی شده است . پس بسیاری از ویژگی های خود را از زبان پایتون به ارث برده است . با استفاده از این فریم ورک امکان ایجاد وب سایت هایی حرفه ای و پیچیده در کمترین زمان ممکنه و همچنین با در نظر گرفتن مورادی چون امنیت و سرعت و … فراهم می باشد . در ادامه برخی از ویژگی های قابل تاکید زبان پایتون که تاثیر مهمی بر این فریم ورک دارند مطرح می گردد .

* پایتون زبانی تفسیری بوده و برای اجرا نیازی به کامپایل ندارد . در برنامه نویسی وب با استفاده از این ویژگی بعد از تغییر کد یا ایجاد آن نتایج کار بلافاصله قابل مشاهده می باشد .
* انواع داده در پایتون داینامیک می باشد . پس شما نگرانی از تعریف نوع متغییر ندارید .
* ساینتکس زبان پایتون کوتاه و در عین حال واضح و قابل فهم می باشد . این بدین معنی هست که برای انجام کار های مشابه کدی بسیار کمتر لازم می باشد . برای مثال معمولا هر خط پایتون معادل ۱۰ خط در جاوا (Java) می باشد !!
* پایتون روشهایی قدرتمند برای meta-programming در اختیار قرار می دهد . این ویژگی امکان ویرایش با اضافه کردن رفتار ها و توابع را با اشیا در زمان اجرای برنامه فراهم می کند .

جدایی از ویژگی های فوق جنگو خود روشهایی برای افزایش سرعت و سهولت برنامه نویسی همزمان با رعایت کامل موارد امنیتی و کارایی را دارا می باشد . که در ادامه با برخی از این ویژگی ها بیشتر آشنا خواهیم شد.

**ایجاد وب سایتی با طرح و ساختار درست و تمیز**

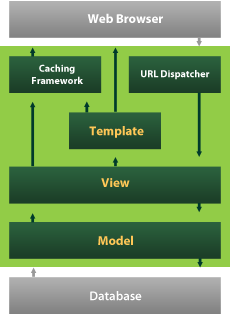
جنگو با استفاده از قوانین از پیش تعیین شده و روش هایی مختلف باعث ایجاد برنامه هایی با سرح و ساختار درست و تمیز می گردد . بنابراین نگهداری و توسعه برنامه های نوشته شده با این فریم ورک بسیار آسان تر خواهد بود .

جنگو از ایجاد برنامه هایی با آمیختگی و روابط نادرست جلوگیری می کند . با استفاده از این ویژگی، برنامه به بخش هایی مستقل و مجرا تقسیم می شود که هر یک بدون نیاز به اعمال تغییرات در بقیه قسمت ها قابل تغییر بوده با استفاده از روابط (API) واضح و بهینه با یکدیگر در ارتباط هستند . برای مثال سیستم ظاهر نمایشی (templates) هیچ گونه اطلاعاتی از سیستم پایگاه داده و ذخیره اطلاعات یا لایه مربوطه به پاسخگویی به درخواست های کاربران یا نهان سازی اطلاعات برای استفاده های بعدی (cache) ندارد . هر کدام از این بخش ها بصورت جداگانه بوده و در صورت نیاز با استفاده از رابط هایی با یکدیگر تبادل اطلاعات کرده یا درخواستی را ارسال می کنند .

جنگو از ساختار MVC یا Model-View-Controller تبعیت می کند . پس کد های مربوط به کار با داده ها (model) و بخش کنترلی (controller) یا bussiness logic و بخش مربوط به رابط کاربر (view) از هم جدا می باشد . برای روشن شدن هرچه بیشتر موضوع این ویژگی با استفاده از یک مثال شرح داده می شود .

**آمیختگی ظاهر نمایشی با کد برنامه**

در تکنولوژی MVC که جنگو نیز از آن بهره می برد ظاهر نمایشی (یعنی تگ های اچ تی ام ال) یا template سایت در فایلی جدا ذخیره می گردد . بخش کنترلی نیز به عنوان ماژول پایتون ایجاد و ذخیره می گردد . اکنون برنامه نویس با بخش کنترلی و طراح با بخش html سروکار خواهد داشت .

[](http://www.mspsoft.com/wp-content/uploads/2011/08/Django_overview21.png)

**آمیختگی کد مربوط به پایگاه داده با کدهای کنترلی**

علاوه بر مشکل فوق این مورد باعث افزونگی نیز می گردد . اگر برای مثال شما نام پایگاه داده را تغییر دهید باید دستور sql را تغییر دهید و ممکن است این تغییر در بخش های مختلفی صورت گیرد . با استفاده از MVC یک لایه واحد و انتزاعی از محل ذخیره سازی اطلاعات ایجاد می گردد که مسئول انجام کلیه درخواست های مربوط به دسترسی به داده ها می باشد . در حالتی که شما از پایگاه داده استفاده کنید نیازی به نوشتن دستورات sql نبوده بلکه با استفاده از دستورات پایتون با اطلاعات و داده ها سروکار دارید . پس برای مثال اگر نام یک فیلد تغییر کند کافیست در یک جا نام آن را تغییر دهید .

**کد نوشته شده با آدرس اینترنتی (URL) گره خورده است**

برای مثال اگر این فایل php در مسیری چون foo/index.php/ باشد این فایل برای تمامی درخواست های این مسیر اجرا خواهد شد . حال اگر بخواهیم همین فایل درخواست های مسیر /bar/ را هم اجرا کند به مشکل بر می خوریم ! در فریم ورک جنگو مسیر اینترنتی غیر وابسته به مسیر فایل روی هارد دیسک سرور و کد مربوطه می باشد . به عبارت دیگر با روشی خاص هر مسیر را به کدی دلخواه در مسیر دلخواه نگاشت می کنیم . اکنون درخواست های مربوطه به آن مسیر خاص به کد مربوطه ارجاع می گردد .

**نوشتن پارامتر های مربوط به اتصال پایگاه داده و نوع پایگاه داده در فایل اصلی برنامه**

نوشتن اطلاعات مربوط به اتصال پایگاه داده در داخل کد اصلی برنامه صحیح نمی باشد . چون این اطلاعات تنظیمات برنامه می باشد نه برنامه نویسی منطقی برنامه . جنگو یک محل واحد برای ذخیره اطلاعات و تنظیمات برنامه دارد . پس براحتی می توان تغییراتی چون تغییر نوع پایگاه داده را انجام داد .

موراد ذکر شده در این کد PHP تنها مثالی برای بررسی موارد و مشکلات معمول بود . بطور حتم با زبان PHP و البته با کمی زحمت بیشتر و نوشتن بسیاری از بخش ها بصورت دستی می توان موارد فوق را پیاده سازی کرد .

**بخش های مختلف Django**

**هسته مرکزی و اصلی**

هسته اصلی و مرکزی جنگو از سه بخش اصلی تشکیل شده است .

همچنین بخش مرکزی جنگو شامل موارد دیگری می باشد که عبرتند از :

* یک وب سرور مستقل و کوچک که برای تست برنامه هنگام توسعه آن استفاده می گردد .
* یک سیستم برای معتبر سازی و serialization فرم های اچ تی ام ال .
* یک فریم ورک نهان سازی اطلاعات برای استفاده مجدد یا کش (cache) که روش های مختلف کش را در اختیار قرار می دهد .
* حمایت از ابزار های میانی (middleware) که امکان اجرای توابع و دستورات مورد نظر را در بخش های مختلف پردازش یک درخواست فراهم می کند .
* یک توزیع کننده (dispatcher) درونی که به بهش ای مختلف یک برنامه وب امکان دریافت سیگنا لها و رویداد های مختلف را می دهد .
* سیستم بین الملل سازی (internationalization) که حتی اماکن ترجمه بخش های مختلف جنگو به زبان های مختلف را فراهم می کند .
* یک سیستم برای تسلسل و سری سازی (serialization) که امکان کار با انواع داده های مبتنی بر XML و JSON و در حالت کلی آژاکس (AJAX) را فراهم می کند .
* سیستمی برای توسعه قابلیت های موتور قالب (template)

**سایر بخش های جنگو**

بسته جنگو همچنین شامل ابزار ها و برنامه های جانبی مختلفی می باشد که در داخل بسته contrib قرار دارند . برخی آز این ابزار ها عبارتند از :

* یک سیستم تصدیق و شناسایی کاربر (authentication) قابل توسعه
* یک رابط مدیریتی داینامیک و پویا
* ابزار هایی برای ایجاد RSS و Atom
* سیستم نظر دهی (commenting) قابل انعطاف و توسعه
* ابزار هایی برای تولید Google Sitemaps
* ابزار های امنیتی برای جلوگیری از حملات cross-site request forgery
* کتابخانه های قالب (tamplate) که امکان استفاده از زبان های نشانه گذاری سبک و کوچک چون Textile و Markdown را فراهم می آورد .

**سرور های قابل استفاده**

Django با استفاده از ماژول mod\_python بخوبی بر روی وب سرور آپاچی (apache) اجرای می گردد . همچنین بر روی تمامی سروی هایی که از WSGI حمایت کنند (اکثر سرور های موجود) قابل اجرا می باشد . همچنین جنگو توانایی راه اندازی سرور FastCGI را دارا می باشد که بدین تریتب می توان از جنگو در داخل هر وب سروری که از FastCGI پشتیبانی می کند استفاده کرد .

سرورهای زیر بطور رسمی پشتیبانی می شود و با استفاده از جنگو قابل اسفتاده می باشد .

* PostgreSQL
* MySQL
* SQLite
* Oracle

ابزار و بخش های لازم برای پشتیبانی از سرور SQL مایکروسافت نیز در حال توسعه می باشد . )در مراحل آزمایشی قرار دارد (

**راه اندازی پایگاه داده :**

جنگو از چهار پایگاه داده زیر پشتیبانی می کند :

PostgreSQL

SQLite 3

MySQL

Oracle

تمامی این پایگاه داده ها با هسته جنگو به خوبی کار می کنند. تنها در مورد پشتیبانی از GIS جنگو پایگاه داده PostgreSQL پر قدرت از بقیه کار می کند.

این پایگاه داده از نظر سرعت و خصوصیت و هزینه و پایداری تعادل خوبی برقرار کرده است. در این پروژه از این پایگاه داده یا به احتمال قوی از پایگاه داده MySQL استفاده می کنیم.

**جنگو با PostgreSQL :**

1- پکیج Psycopg یا نسخه 2 را از سایت زیر نصب می کنیم، نسخه 2 دارای نصبی آسان تر فعالیت بیشتر در توسعه و جدیدتر است :

<http://www.djangoproject.com/r/python-pgsq/>

پایگاه داده برای نسخه ویندوز باینری های کامپایل شده در آدرس زیر موجودند :

<http://www.djangoproject.com/r/python-pgsql/windows/>

**جنگو با MySQL :**

با MySQL 4.0 به بالا کار می کند. نسخه 3.x subquery های تو در تو و تعدادی از جملات SQL را پشتیبانی نمی کند، همچنین باید پکیج MySQLdb را از سایت زیر دریافت کرد :

<http://www.djangoproject/r/python-mysql/>

**سرور جنگو :**

برای بخش سرور این پروژه از سرور خود جنگو به نام runserver استفاده می کنیم.

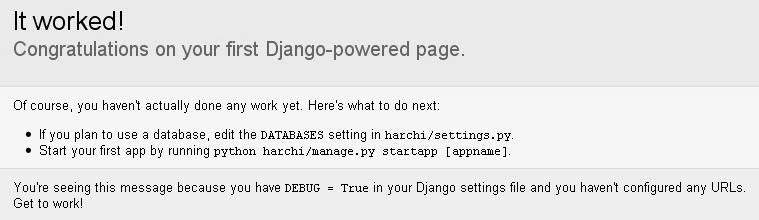
یک سرور سبک و داخلی است و سایت را با آن توسعه می دهیم. بدون راه اندازی و تنظیمات سرور (Apache) سرعت سایت را افزایش داده و اگر در کد تغییری ایجاد کنیم به طور اتومات آن تغییر در سرور به روز رسانی می شود.

برای اجرا کردن سرور به دایرکتوری پروژه رفته (cd mysite) :

python manage.py runserver

دستور بالا سرور را به صورت داخلی روی پورت 8000 راه اندازی می کند. تنها بر روی رایانه شخصی قابل دسترسی است.

اگر مسیر رو به رو را بنویسید در مرورگر عکس زیر را می بینید : http://127.0.0.1:8000



**تغییر هاست یا پورت :**

برای اشتراک گذاری سایت با دیگر اعضا IP آدرس 0.0.0.0 به سرور می گوید به هر رابط شبکه گوش کند.

python manage.py runserver 0.0.0.0:8000

**معماری مورد استفاده**

معماری نرم افزار چیست ؟

معماری نرم افزار ، ساختار یک نرم افزار را بصورت کلی و سطح بالا بیان میکند. یعنی در انتخاب معماری ، جزئیات ریز و نحوه ی پیاده سازی مورد بررسی قرار نمی گیرد.

هر معماری شامل یک سری اجزاء (Component) بوده که این اجزاء از طریق یک سری اتصالات (Connector) با هم در ارتباطند. هر کدام از اجزاء درون معماری می تواند یک وظیفه ی مشخص داشته باشد.

هر نرم افزاری که پیاده سازی شده ، از یک معماری خاص تبعیت کرده است. حتی اگر طراح آن متوجه معماری آن نباشد و یا جزو معماری های شناخته شده نباشد!

برخی از معماری های شناخته شده عبارتند از :

- معماری 3 لایه

- معماری Sense Compute Actuator

- معماری Client - Server

- معماری Peer To Peer

- معماری MVC

و ...

یک نرم افزار می تواند در آن واحد از چند معماری بصورت همزمان استفاده کند!

* محدودیت ها

محدودیت های طراحی:

سایت مورد پیاده سازی قابلیت دارا بودن امکانات و بخشهای زیادی را دارد, اما به علت اهمیت سرعت سایت و محدودیت منابع, تصمیم گرفته شد که تعدادی از مهمترین امکانات مورد نیاز پیاده سازی شود.

با توجه به اینکه اقشار مختلف مردم با سطوح متفاوتی از آشنایی با اینترنت و وب می توانند کاربر این سایت باشند, طراحی سایت باید به گونه ای باشد که درک بخشهای مختلف سایت و استفاده از آن برای همه ممکن و آسان باشد.

به علت ذخیره سازی اطلاعات کاربران, انجام تراکنشهای مالی در طول انجام پروژه ها در سایت و همچنین حفظ حق مالکیت پروژه های انجام شده, از نظر امنیتی نیز محدودیت هایی بوجود خواهد آمد که در طراحی سایت باید مدنظر قرار بگیرد.

با توجه به امکان انجام گفتگو بین کاربران سایت(از طریق چت روم) , برای حفظ شئون اسلامی و رعایت قوانین جمهوری اسلامی ایران, باید نظارت دقیقی بر این قسمت از سایت وجود داشته باشد.

همچنین با توجه به هدف سایت, کاربران محدود به افرادی خواهند بود که توانایی انجام کارهای گرافیکی داشته باشند و همچنین قسمت های مختلف سایت باید محدود باشند. برای مثال قسمت ارائه ی پروژه باید محدود به فیلد هایی باشد که بیانگر مشخصات یک پروژه ی گرافیکی باشد.

پیش فرض

اولین گام آماده کردن کامپیوتر برای کار، قبل از شروع پروژه است.  
قبل از این که وب سایت خودتان را بسازید، می بایست کامپیوتر خود را برای کار آماده نمایید که شامل برنامه هایی است که یا در سیستم شما موجود می باشد و یا این که می بایست آن ها را نصب نمایید.  
برنامه های زیادی در بازار موجود می باشد ولی در ابتدا از برنامه های رایگان و قابل دسترس استفاده می کنیم.  
وسایل و برنامه های اصلی که شما نیاز دارید:

1. سیستم عامل تحت ویندوز (7 یا 8)
2. نرم افزار نمایش فایل های فشرده شده (Winrar or 7Zip)
3. یک مرورگر (web browser) برای دیدن صفحات وب در آن (Mozilla Firefox or Google Chrome)
4. وسیلۀ ابتدایی برای ویرایش فایل های متنی که در واقع صفحه وب شما را شامل می شود (Notepad++ یا هر TextEditor دیگری )
5. اما همه عناصر که متن نیستند شما برای ویرایش عکس هم به برنامه هایی نیاز دارید که می توانید از برنامه های زیر استفاده کنید:  
   Adobe photoshop  
   Adobe fireworks  
   Picasa
6. نسخه نصب شده python 2.7

با توجه به این که قرار است پروژه سایت با استفاده از فریم ورک Django پیاده سازی شود (در قسمت ابزار های مورد استفاده به تفصیل در مورد فریم ورک می پردازیم) این فریم ورک با استفاده از زبان پایتون استوار شده است پس یکی از پیش فرض های سیستم ما زبان برنامه نویسی Python می باشد.

اولین گام در نصب فریم ورک اطمینان حاصل کردن از نصب پایتون است.

پایتون نسخه 3.0 نیز منتشر شده است اما جنگو هنوز از آن پشتیبانی نمی کند بنابراین نسخه مورد نظر ما برای این فریم ورک python 2.7 می باشد.

* متدولوژی

متدولوژی مورد استفاده برای این پروژه, روش آبشاری به همراه تغییرات جزئی به منظور تطبیق بهتر با مراحل ذاتی طراحی وب سایت خواهد بود.

دلایل استفاده:

1. این نوع سایت قبلاً چند بار پیاده سازی شده است.

2. نیازمندی ها مشخص هستند و تقریباً هیچ نیاز مبهمی وجود ندارد.

3. منابع کافی و تجارب قبلی به صورت رایگان موجود هستند.

4.تکنولوژی های مورد نیاز برای ساخت این سایت مشخص هستند.

1. **استخراج و تحلیل نیازمندی ها**

اولین گام شناخت نیازمندی های کاربران است. شناسایی کاربران نهایی سیستم, گامی بسیار مهم به شمار می آید.

در این فاز هدف کلی سایت و نحوه ی تعامل کاربران با آن مورد بررسی قرار می گیرد. در این مرحله شناخت صحیح کاربران سیستم از اهمیت بالایی برخوردار است.

با استفاده از روش های موجود و رسم دیاگرام های مختلف و تحلیل دقیق نیازها, باید نیازمندی ها ی کاربران استخراج و نحوه ی تعامل کاربران با یکدیگر مشخص شود.

این مرحله از اهمیت بالایی برخوردار است و باید زمان مناسبی برای آن اختصاص داده شود. به طوری که نه وقت زیادی صرف آن شود و نه به خاطر عجله و تخصیص زمان کم, برخی از نیازها به درستی شناسایی نشوند.

1. **تعیین مشخصات سایت**

در این مرحله با استفاده از نتایج تحلیل نیازمندی ها که در بخش قبل بدست آمد, روی طرح کلی سایت و قسمتهای مختلف آن تصمیم گیری می شود.

در واقع در این مرحله با استفاده از گزارش بدست آمده در مرحله ی تحلیل نیازمندی ها و همچنین در نظر گرفتن زمان و هزینه و در مجموع منابع در دسترس, یک طرح کلی از سایت ایجاد می شود.

در این مرحله, یک پروپوزال کامل, درباره ی بخشهای مختلف سایت و نحوه ی برطرف کردن نیاز های کاربران توسط این بخش ها تهیه خواهد شد.

1. **طراحی**

در این مرحله, با توجه به قسمتهای مختلف سایت که در مرحله ی قبل تعیین شد, قالب و گرافیک صفحات سایت ایجاد می شود.

برای ایجاد قالب سایت موارد مختلفی از جمله نوع کاربران سایت, نوع فعالیت سایت و ... باید درنظر گرفته شود, تا طراحی بخش های مختلف سایت از جمله لوگو, منوها, هدر و همچنین رنگ بندی قسمتهای مختلف بدرستی و با توجه به هدف سایت انجام شود.

1. **افزودن محتوا**

در این مرحله محتوای مرتبط با سایت به قالب طراحی شده اضافه می گردد.رعایت نکات گرامری نکته ی مهمی برای درک قسمتهای مختلف سایت توسط کاربران می باشد.

نوع نگاشت محتوای سایت برای موتورهای جستجو بسیار حائز اهمیت است. موتورهای جستجو به کلمات، جمله بندی و بسیاری موارد دیگر از محتوا اهمیت ویژه ای می دهند.

در این مرحله دو نوع محتوا به سایت اضافه خواهد شد: محتوای متنی و محتوای چند رسانه ای با محتوای متنی به راحتی می توان اطلاعات اولیه مورد نیاز مشتریان و مخاطبان را مرتفع نمود.

در استفاده از محتوای چند رسانه ای می بایست ابتدا همه آنها را جهت بالا بردن سرعت نمایش در وب سایت بهینه کنیم.

1. **نوشتن کدها**

بعد از طراحی قالب سایت و افزودن محتوا, بخش اصلی ساخت سایت یعنی افزودن کدها آغاز می شود. در این مرحله تمام کدهای سایت از جمله کدهای html, css, javascript, python و همچنین طراحی پایگاه داده و کوئری های آن انجام خواهد شد.

نوشتن کدها باید طوری صورت پذیرد که منطبق با ساختار گرافیکی سایت باشد و آنرا دچار تغییر نکند.

البته به خاطر گستردگی این بخش, می توان آنرا به چند بخش کوچکتر تقسیم بندی نمود. برای مثال:

نوشتن کدهای سمت کاربر

طراحی پایگاه داده

نوشتن کدهای کنترلی برای ارتباط کاربر و پایگاه داده

...

1. **تست سایت**

در این مرحله وب سایت طراحی شده از ابعاد مختلف و با استفاده از ابزارهای مناسب تست خواهد شد و کارهای لازم برای افزایش کارایی سیستم انجام خواهد شد.

برای مثال با توجه به سرعت اینترنت در ایران, سرعت لود قسمتهای گرافیکی سایت, از اهمیت بالایی برخوردار خواهد بود.

همچنین وب سایت از نظر امنیت مثلاً امنیت دسترسی به پایگاه داده باید مورد آزمایش قرار بگیرد.

همچنین در این مرحله سعی خواهد شد مواردی که اضافی هستند و وجود آنها تاثیر چندانی در بهبود عملکرد سایت ندارد حذف شود.

همچنین کمبود ها و نقص های سایت شناسایی خواهند شد و تا حد امکان برطرف خواهند شد.

ابزارهای مختلفی برای تست سایت از جمله ابزارهای تست آنلاین برای این منظور موجود می باشد.

برخی از تستهایی که در این مرحله قابل انجام هستند عبارتند از :

,Integration testing, Stress testing, Scalability testing, load testing resolution testing, cross-browser compatibility testing

1. **SEO و ارتقا، سایت**

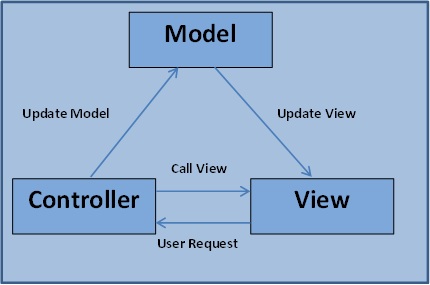
در این مرحله با استفاده از روش های افزایش رتبه, از جمله استفاده ی مناسب از تگ های متا, استفاده از شبکه های اجتماعی و سایر روشهای مناسب, سعی در معرفی هرچه بهتر سایت به کاربران اینترنت خواهد شد.

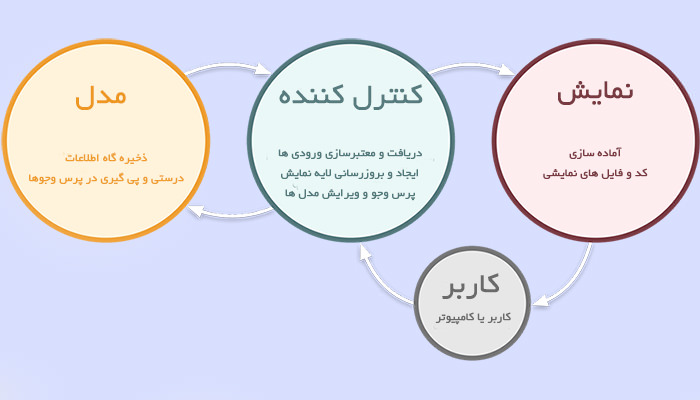
1. **آپدیت و پشتیبانی سایت**

یک وب سایت همواره نیاز به بروزرسانی و تغییر دارد تا با شرایط و نیازهای کاربران وفق پیدا کند. بعد از ساخت سایت و شروع به استفاده از آن, نیاز داریم تا همواره کارایی سایت را بهبود دهیم و باگ های احتمالی را کشف و برطرف کنیم.

* معماری مورد استفاده

**بررسی معماری نرم افزاری MVC –**



****

نیاز روز افزون به کامپیوتر و مکانیزه کردن و سپردن تقریبی تمامی امور به دست ماشین امری انکار ناشدنی است. در این بین تولید کنندگان نرم افزار نیز تلاش می کنند تا نرم افزاری تولید کنند که بتواند اکثر نیاز های متقاضیان را به بهترین نحو ممکن تامین کند؛ در همین راستا در تلاش هستند که روند تولید نرم افزار را به سمتی بکشانند که ساختار استاندارد و تائید شده ای داشته باشد.

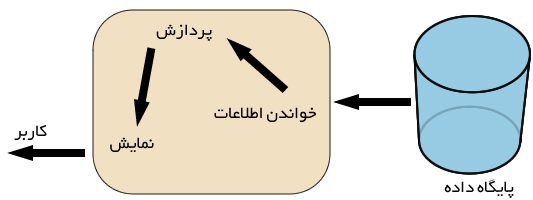
شاید اینطور بتوان گفت که دوران کد نویسی به پایان رسیده و همه چیز به سمت زیر ساخت ها و بنیان نهادن چارچوب های استاندارد و پیروی از آن ها در امر تولیدِ بهتر نرم افزار در حرکت است.

سوالی که اکنون پیش می آید این است که خصوصیات یک نرم افزار خوب چیست؟

* قابل حمل بودن
* قابل استفاده مجدد بودن
* قابل تغییر بودن
* بهینه بودن از لحاط حافظه و زمان

فرض کنید نرم افزاری برای شرکتی نوشتید که یک بخش از آن مقدار سود و زیان شرکت را در سال های مختلف بر اساس ارقام بیان می‌کند. حال مدیر شرکت پس از مدتی ازشما می خواهد برنامه را طوری تغییر دهید که همین اطلاعات را به گونه های مختلف دیگری مانند نمودارهای مختلف (میله ای، دایره ای و …) در اختیار داشته باشد و یا حتی بخواهد آنها را به فرمت خاصی و در فایل های خاصی ذخیره کند؛ در این مواقع چطور مشکل را حل می کنید؟

همانطور که در تصویر1 نیز نشان داده شده است تمامی اعمال، از دریافت داده ها که مهمترین بخش است و همچنین پردازش آن‌ها تا نمایش آن‌ها به کاربر همگی در یک فرم طراحی و پیاده سازی شده اند و دقیقا مشکل همین جا نمایان می‌شود. ارتباط مستقیم با منبع داده بر قرار کردن، جدا از اینکه مشکلات امنیتی دارد که صحبت در مورد آن خارج از این مبحث است، باعث می‌شود که دست برنامه نویس را برای تغییرات آتی دربرنامه ببندد. چون داده ها درون خود فرم از منبع داده و به صورت مستقیم خوانده می‌شوند پس دسترسی به داده های خوانده شده دیگر وجود ندارد یا حداقل متحمل سربار زیادی می باشد.

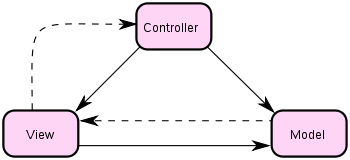


**تصویر 1: خواندن و نمایش اطلاعات از پایگاه داده**

یکی از راه کار هایی که امروزه بیشتر شاهد استفاده از آن هستیم، تولید نرم افزار بر اساس ساختار های لایه ای می باشد. بدین صورت که کل نرم افزار به تعداد لایه هایی تقسیم می‌شود، هر لایه وظیفه خاص خود را دارد و لایه ها از نتایج لایه های دیگر استفاده می کنند و از نظر ساختار جدا از یکدیگر هستند.

تعداد این لایه ها بسته به نرم افزار و طراحی می تواند 2، 3، 4 یا 5 لایه یا حتی بیشتر باشد. اما استاندارد آن که بیشتر از بقیه هم استفاده می‌شود 3 لایه است و به روشی که بر اساس این تئوری پیاده سازی می‌شود اصطلاحاTire Programming 3 گفته می‌شود.

MVC مخفف سه کلمه Model، View و Controller است. در واقع هدف از معماری نرم افزاری MVC ایجاد معماری چند لایه ای جهت جداسازی قسمت های مختلف برنامه است. به طور دقیق‌تر MVC باعث جدا شدن بخش های منطقی برنامه که شامل اطلاعات، سطح دسترسی ها، چک کردن صحت داده ها از لایه نمایش (presentation) یا در واقع همان لایه ای که مستقیما با کاربر نهایی (end-User) در ارتباط است، می‌شود. (تصویر 2)



**تصویر 2: معماری سه لایه MVC**

پس بر اساس توضیحات فوق میتوان هر یک از بخش های معماری MVC یعنی Model ، View و Controller را معرفی کرد.

**معرفی مدل، نما و کنترلگر**

معماری MVC به نوعی از معماری در مبحث برنامه نویسی گفته می شود که تاکید بسیار به لایه بندی سیستم دارد. در این جلسه و در ادامع بررسی معماری نرم افزاری MVC می خواهیم سه بخش اصلی در معماری نرم افزاری MVC را بررسی مختصری داشته باشیم.

**مدل**

بار اصلی معماری MVC بر عهده این بخش است. مدل میتواند با داده ها در ارتباط باشد. الزاما منظور از داده حتما ارتباط با پایگاه های داده همچون MYSQLو MSSQL نیست، حتی منبع داده ها در بخش Modelمیتواند یک آرایه از اعداد و یا هر چیز دیگری باشد. Model وظیفه چک کردن داده ها جهت صحت درستی آنها را هم بر عهده دارد.

**نما**

نما وظیفه‌ی بر قراری ارتباط با کاربر نهایی و گرفتن داده از کاربر و نمایش داده های آماده به کاربر از طریق برقراری ارتباط با دو بخش دیگر یعنی Model و Controller را دارد. در واقع نکته مهمی که در بخش View باید آن را مد نظر داشت این است که این لایه مسئول کنترل صحت داده های وارد شده از طریق کاربر و همچنین مسئول صحت داده های نشان داده شده به کاربر نیست و در واقع این بخش با داده های خام کار می‌کند.

به عنوان یک مثال ساده، بسیاری از برنامه نویسان برای زمانی که در فرم Login برنامه، کاربر کلمه عبور خود را وارد می‌کند، در همان فرم Login اقدام به چک کردن پسورد مبنی بر صحت آن و دیگر موارد مورد نظرشان می کنند؛ که این عمل در معماری MVC قابل قبول نیست! در واقع برای حل این مسئله در معماری MVC، در فرم Login، هنگامی که کاربر کلمه عبور را وارد کرد و دکمه Login را زد، کلمه عبور داده شده بدون هیچ گونه اعمالی مانندرمزگذاری **(encrypt)** کردن و . . . به بخش های دیگر فرستاده می شود و فقط یک نتیجه ساده مبنی بر این که کاربر اجازه ورود دارد یا خیر را از بخش های دیگر دریافت می‌کند که بر اساس آن اجازه ورود کاربر به برنامه داده می شود.

**کنترلگر**

در این لایه درخواست های کاربر پردازش می‌شود و به قسمت های مربوطه هدایت می‌شود. این لایه، واسطه‌ی ارتباط بین لایه‌ی نما و مدل است، بدین صورت که درخواست های کاربر از طریق این لایه به مدل ارسال می‌شود.

**روند اجرای برنامه در MVC**

در معماری MVC روند کلی برنامه (تصویر زیر) به این شکل است که کاربر تقاضای خود را از طریق واسط های برنامه نویسی نظیر Formها، Linkها و … از بخش View درخواست می‌کند. بخشView درخواست ها را به بخش Controller فرستاده و این بخش با برقراری ارتباط با بخش Model در خواست های کاربر را پردازش کرده و پس از پایان پردازش زمانی که خروجی درخواست داده شده آماده گردید، بخش Controller بخش View را آگاه می سازد تا خود را بر اساس تغییرات جدید به روز سازد.



* انواع روش های تست نرم افزار

1. تست عملکرد (WHITE BOX TESTING):

تست نرم افزار از نظر درستی عملکرد , در این تست کامپوننتها و فرآیندهای مورد تست قرار می گیرند .

این استراتژی با تست کدهای نرم افزاری و ساختمان داخلی نرم افزار سر و کار دارد .

در این روش کدهای باید به گونه ای اجرا و بررسی شود که مطمئن شویم سطر به سطر کد های برنامه حداقل یکبار اجرا شده است .

در کد برنامه باید تمام عبارتهای شرطی(if else ,switch case) را تک تک مورد بررسی قرار دارد بدین صورت که در یک عبارت if else هم قسمت if و هم قسمت else هر کدام به صورت مجزا یکبار اجرا شود، حلقه ها در نرم افزار نقش اساسی دارند چون می توانند با اشتباه جزئی مقدار زیادی از منابع را مصرف کرد و برای مثال شرط خروج از حلقه به اشتباه هیچ وقت true نشود برای نمونه حلقه را برای ورودی بزرگتر تست کنید .

1. تست استرس (STRESS TESTING) :

در این تست برنامه در مقابل بار سنگینی مانند مقادیر عددی پیچیده , مقادیر زیاد ورودی و مقادیر زیاد پرس و جو امتحان میشود.

فشار همزمان چند ایستگاه مختلف برای تست حافظه، از دیگر نمونه های این نوع تست می باشد.هدف از این تست طراحی محیطی است که مخرب تر از محیطی که برنامه در دنیای واقعی و در شرایط نرمال با آن روبرو می شد .

1. تست اکتشافی :

در این روش فرض می شود شخصی که هیچ گونه دانشی در خصوص نرم افزار ندارد در حال کار با سیستم است و به روشهای غیر معمول نرم افزار را مورد تست قرار می دهد.

این روش یک روش جالب و قوی برای تست نرم افزار است.

1. تست تطبیق پذیری با محیط :

در این تست، نرم افزار در سیستم عاملهای مختلف یا مرورگرهای مختلف و کالاً زیر ساختهای مختلف تست می شود ومیزان تطبیق پذیری آن با سایر محیط ها مشخص می شود .

1. تست امنیت :

این نوع تست می تواند شامل تست محرمانه بودن،جامعیت،تصدیق هویت،مجوز کاربری

می باشد .

1. تست واحد (Unit Test):

این تست معمولا توسط خود برنامه نویسان انجام می گیرد و تک تک ماژولها را در حین پیاده سازی تست می کنند.

1. آزمون تحمل خرابی :

این تست می تواند شامل موارد زیر باشد :

اختلال در محیط سخت افزاری(قطع ناگهانی برق – خرابی دیسکهای دستگاه سرویس دهنده یا ایستگاه کاری – قطع اتصالات شبکه داخلی)

روش های مختلفی وجود دارد که بتوان یک نرم افزار را مورد ارزیابی عملکرد قرار داد به عنوان مثال کار پیوسته با سیستم بدون خارج شدن از نرم افزار ( حداقل 72 ساعت)، استفاده فعال همزمان از نرم افزار توسط چند کاربر، بررسی کارکرد نرم افزار با استفاده از حداکثر گنجایش بانک اطلاعاتی به این منظور جدول باید حاوی حداکثر تعداد پیش بینی شده رکورد باشد(مانند تست سیستم در مواجه با بیش از چند میلیون رکورد، بازخورد سیستم در هنگام قطع و اختلال در خطوط انتقال داده یا آماده نبودن تجهیزات جانبی.

در پایان تست،هر آزمونگر ضمن تولید مستند گزارش انجام آزمون (Test Result) نتیجه آزمون رادر یک TR درج می نماید. از دیگر نکات مهم تست نحوه برخورد سیستم ها در هنکام بروز خطا و کنترل محیط اجرایی در جهت حفظ کارایی کل سیستم می باشد.

* نحوه ی تست نرم افزار

بر اساس صحبت ها و هماهنگی های انجام شده اعضای گروه به این نتیجه رسیدند که از بین

روشهای موجود روش تست واحد (unit) را انتخاب بنمایند.

در نتیجه اجزای نرم افزاری پروژه در عین حال برنامه نویسی تست خواهند شد.

* سایت استفاده شده تقسیم وظایف

سایت در نظر گرفته شده برای کار گروهی سایت Trello.com است که تمامی اعضای

گروه در آن عضو شده اند و تقسیم وظایف در آن انجام خواهد گرفت.

آدرس Board اختصاصی گروه در لینک زیر است :

<https://trello.com/b/SdnfzsML/se2>

* سایت مدیریت کد نویسی

برای مدیریت کدهای برنامه نویسی و کد نویسی گروهی از سایت Github.com استفاده

شده است.

که آدرس آن در زیر ذکر شده است:

<https://github.com/KinGoverm/SE2>

* افراد و نقش ها

در این پروژه افرادی که نقش دارند شامل سه دسته هستند. مدیر ، کارفرما و پیمانکار که هرکدام نقش ها و امکانات خاص خود را دارند.

کارفرما با شناسه و رمز خود وارد می شود و سفارش و قیمت و مدت زمان کار مورد نظرش را در سیستم وارد می کند.و در سیستم ثبت می شود و به بقیه استفاده کنندگان توسط صفحه ی عمومی ابلاغ می شود.

پیمانکار با شناسه و رمز خود وارد سیستم می شود و از بین سفارش های ثبت شده کاری را برای خودش انتخاب می کند با احتساب شرایط مالی و زمانی و ... . بدین ترتیب طرحی را در سیستم بارگذاری می کند.

کارفرما با بررسی طرح های اولیه ثبت شده آنهایی را که می خواهد با مبلغ پیشنهادی انتخاب می کند و در صورت بروز اختلافات در مورد هزینه ها و پرداخت پیش پرداخت و پرداخت کامل و مواردی اینچنین هریک از استفاده کنندگان با انتخاب گزینه ی شکایت مدیر سایت را مطلع می سازد و مدیر با بررسی شکایت ها موضوع را بررسی خواهد کرد.در این سیستم نحوه ی تعامل کارفرما و مشتری به منظور پرهیز از ایجاد اختلافات و اختلالات بدین گونه است که کارفرما پس از بررسی طرح های اولیه به آن طرح هایی که مورد پسندش واقع شده مبلغی پیشنهادی عنوان می کند و اگر پیمانکار نیز قبول کرد کارفرما موظف به پرداخت درصدی از آن مبلغ به سایت می سپارد و پیمانکار از جهت پشتوانه ی مالی نگرانی نخواهد داشت و چون اطلاعات و تراکنش ها و قرارداد ها در نزد سایت محفوظ است امکان پیگیری های احیانا قانونی نیز فراهم است.

پس از آنکه پیمانکار طرح نهایی را پیاده سازی نمود و خواستار بارگذاری آن در سایت شد کارفرما باید همه ی مبلغ قرارداد را بپردازد و طرح یا همان سفارش کار را دریافت کند.در آخر در هنگام تصفیه ی حساب سهم پیمانکار و کارفرما و سایت به درستی و طبق قرارداد و قوانین تقسیم می گردد.

در همه ی این مراحل برای مدیر سایت امکان بررسی ها و تنظیمات و حل و فصل کردن اشتباهات فراهم است.

در حین کار کردن با سیستم به طریقی که گفته شده امکان گفتگو نیز فراهم است و کارفرما و پیمانکار می توانند با هم به توافق بیشتری برسند و تبادل اطلاعات به منظور پیشرفت کار کنند.

* پشتیبانی

پشتیبانی نرم افزاری سیستم تحت وب ما شامل برطرف کردن اختلالات مالی در استفاده از سیستم و در صورت وقوع چنین رخدادی با بررسی آن مورد را برطرف می سازیم.بررسی شکایات مطرح شده توسط استفاده کنندگان توسط مدیریت سایت صورت می گیرد.در صورت بروز اختلالات در سیستم اتصال به بانک اطلاعاتی و موارد اینچنین مدیریت سایت موظف به پیگیری مورد می شود و حق استفاده کننده را می ستاند.

در حین استفاده از سیستم در صورت بروز اشکال در سیستم یا سوال در مورد قوانین و اطلاعات هر کدام از استفاده کنندگان اعم از کارفرما یا پیمانکار می توانند با استفاده از سیستم گفتگوی آنلاین موجود در سایت با بخش پشتیبانی در ارتباط باشند و پشتیبان با ارائه ی پاسخ های مفهوم و مربوط به این وظیفه عمل می نماید.

در صورت بروز اشکال در سیستم سرور وب سایت بخش پشتیبانی و مدیریت سایت وظیفه ی برطرف کردن اختلال را دارد و در اسرع وقت این کار را با به کارگیری بهترین راه حل های کاربردی انجام می دهد.

بخش دیگر پشتیبانی این وب سایت ، درگاه پرداخت اینترنتی بانک معتبر جمهوری اسلامی ایران می باشد که با ارائه ی خدمات خود تبادلات مالی را تسهیل می کنند.

در صورت بروز اشکال در تراکنش بانکی بخش پشتیبانی با بانک مربوط تماس گرفته و مسئله را پیگیری می کند.