

به نام خدا

علی حیدری

(۱) وب سرور یک سخت افزار یا نرم افزاری است که معمولاً درخواست های http را از کلاینت دریافت کرده و پاسخ آنها را در غالب یک فایل html برمیگرداند که میتواند حاوی عکس و موارد دیگر نیز باشد به شیوه دیگر وب سرور یک کامپیوتری است که وب سایت ها را ذخیره کرده و آنها را در دسترس کاربران قرار می دهد. نقش nginx علاوه بر ارائه محتواهایی مثل فایل های ... , css , html وظایف دیگری نیز دارد مثلاً اگر یک وبسایت بر روی چند سرور باشد درخواست ها را بین سرور ها پخش می کند و در نیجه سرعت و پایداری نیز افزایش می یابد یا می تواند رمز گشایی در خواست های https و رمز گذاری آن ها را انجام دهد و می تواند یک واسط بین کاربران و سرور اصلی باشد در این حالت تمام درخواست اول به nginx رفته سپس آنها را به سرور برنامه مناسب می فرستد .

(۲) ارتباط بین این دو از طریق http که از مدل request – response پیروی می کند . کاربر ادرس یک سایت در مرور گر خودش وارد کرده و مرورگر یک درخواست http آماده کرده (شامل هدر و بدنه و نوع پیام( get , post, ... ) و به ip وب سرور ارسال می کند - در مرحله بعد سرور باید پیام کلاینت را پردازش کند مثلاً اگر درخواست برای یک فایل استاتیک بود که همانموقع برای کلاینت ارسال می کند اما اگر نیاز به پردازش منطقی داشت آنرا برای برنامه بک اند ارسال می کند - پس از پردازش باید یک response برای کلاینت ارسال کند که خب شامل برخی کدها مثل ۲۰۰ و ۴۰۴ و همچنین محتوای فایل html می باشد - مرورگر جواب سرور را دریافت کرده و آنرا در صفحه نمایش می دهد .

(۳) http پروتکلی است که پایه و اساس پروتکل انتقال داده را تشکیل می دهد که شامل یکسری قوانین بین کلاینت و سرور می باشد تا پیام ها در آن ساختار برود وقتی که یک url را وارد می کنیم اول باید ip سروری که آن دامنه روی آن است را پیدا کنیم که این کار با استفاده از سرور dns رخ می دهد مرورگر یک درخواست به سرور dns می دهد تا ip متناظر آن را دریافت کند پس از دریافت ip مرورگر از طریق پورت مناسب یک ارتباط tcp با سرور ایجاد میکند وب سرور درخواست را دریافت می کند. اگر URL مربوط به برنامه جنگو باشد، وب سرور درخواست را از طریق یک واسط به نام WSGI به برنامه جنگو منتقل می کند. برنامه جنگو درخواست را پردازش می کند پس از انجام عملیات لازم، برنامه جنگو یک پاسخ HTTP را تولید می کند و آن را به وب سرور بازمی گرداند. وب سرور این پاسخ را از طریق شبکه به مرورگر ارسال می کند

(۴) اپلیکیشن جنگو در سمت سرور و در لایه اپلیکیشن قرار دارد

۵) کاربر ادرس وبسایت در مرورگرش وارد می کند مرورگر نام دامنه را به ادرس ip سرور تبدیل میکند مرورگر با استفاده از ip بدست آمده یک اتصال ایجاد کرده و درخواست http را به سرور ارسال می کند وب سرور پس از دریافت در صورت دریافت یک درخواست داینامیک آن را به اپلیکیشن جنگو ما ارسال می کند جنگو درخواست را از وب سرور دریافت می کند URL. درخواست را با الگوهای تعریف شده در urls.py تطبیق می دهد تا view مناسب را پیدا کند. view منطق برنامه را اجرا می کند. اگر نیاز به داده باشد از Model ها برای ارتباط با پایگاه داده استفاده می کند model ها نیز درخواست view را به کوئری sql تبدیل می کند و آنها را به پایگاه داده ارسال می کند .

۶) جنگو از یک معماری به نام MVT استفاده می کند که تغییر یافته از معماری شناخته شده MVC می باشد

### MVC (Model-View-Controller):

**Model :** مسئول مدیریت داده ها و منطق کسب و کار است. با پایگاه داده تعامل دارد.

**View:** مسئول نمایش داده ها به کاربر است

**Controller:** به عنوان واسطه بین Model و View عمل می کند. ورودی کاربر را از View دریافت کرده، آن را پردازش می کند در صورت نیاز با فراخوانی Model و سپس View مناسب را برای نمایش نتیجه انتخاب می کند.

### MVT (Model-View-Template)

در جنگو :جنگو خود وظیفه "Controller" انتخاب اینکه کدام کد برای کدام URL اجرا شود را بر عهده می گیرد .بنابراین، معماری آن کمی متفاوت است :

Model مدل دقیقاً مشابه MVC ، Model در جنگو لایه داده را تعریف می کند .هر مدل به یک جدول در پایگاه داده متصل است و فیلدهای آن ستون های جدول را نشان می دهند. این بخش مسئول تمام تعاملات با پایگاه داده است.

View برخلاف MVC ، View در جنگو مسئول منطق پردازش درخواست است .این بخش درخواست HTTP را دریافت می کند، (در صورت نیاز) با

Model برای دسترسی به داده‌ها تعامل می‌کند و در نهایت یک پاسخ HTTP را باز می‌گرداند View. در جنگو بیشتر شبیه به Controller در MVC عمل می‌کند.

Template این بخش مسئول لایه نمایش است.

Template ها فایل‌های HTML هستند که دارای متغیرها و تگ‌های منطقی ساده‌ای هستند View. داده‌ها را به Template ارسال می‌کند و موتور رندرینگ جنگو این قالب را با داده‌ها پر کرده و یک صفحه HTML نهایی برای نمایش به کاربر تولید می‌کند. این بخش شبیه به View در MVC است.

### URL Dispatcher:

علاوه بر این سه جزء، جنگو دارای یک URL Dispatcher است. این بخش بر اساس URL درخواست شده، تصمیم می‌گیرد که کدام

View باید آن را پردازش کند. این کار از طریق فایل urls.py انجام می‌شود.

۷) محیط مجازی، یک پوشه ایزوله برای یک پروژه پایتون است که کتابخانه‌ها و پکیج‌های آن پروژه را از سایر پروژه‌ها جدا می‌کند. هر پروژه می‌تواند نسخه‌های متفاوتی از یک کتابخانه را بدون اینکه روی پروژه‌های دیگر تأثیر بگذارد، داشته باشد. به شما اجازه می‌دهد تا لیستی دقیق از تمام بسته‌های مورد نیاز پروژه‌تان در فایل requirements.txt تهیه کنید تا دیگران بتوانند به راحتی پروژه شما را راه‌اندازی کنند.

۸) این دو ابزار برای ۲ هدف کاملاً متفاوت استفاده می‌شوند Django Templates برای ساخت وبسایت‌های کامل استفاده می‌شود. خروجی آن صفحات HTML است که مستقیماً در مرورگر نمایش داده می‌شود. زمانی از آن استفاده می‌کنید که فرانت‌اند و بک‌اند شما یکپارچه هستند. Django REST Framework برای ساخت API استفاده می‌شود. خروجی آن داده خام معمولاً با فرمت JSON است. زمانی از آن استفاده می‌کنید که بک‌اند شما باید به کلاینت‌های مختلف سرویس‌دهی کند.

۹) نصب ابزار محیط مجازی `pip install virtualenv` ایجاد یک محیط مجازی به نام `venv` :  
`python -m venv venv` فعالسازی محیط مجازی : `venv\Scripts\activate` و از دستور  
`django-admin startproject` برای ایجاد پروژه استفاده می کنیم

۱۰) یک پروژه جنگو مجموعه‌ای از اپلیکیشن‌ها است. هر اپ مسئول یک بخش مجزا از عملکرد وبسایت است برای ایجاد یک اپ : `python manage.py startapp myapp` برای اینکه پروژه جنگو از وجود اپ جدید مطلع شود، باید آن را در لیست `INSTALLED_APPS` در فایل  
`myproject/settings.py` ثبت کنیم اتصال URL های اپ به پروژه : برای اینکه URL های مربوط به `myapp` قابل دسترسی باشند، باید فایل `urls.py` پروژه اصلی را ویرایش کنید تا درخواست‌ها را به فایل `urls.py` اختصاصی اپ هدایت کند.