

به نام خدا



دانشگاه صنعتی امیرکبیر
(پلی تکنیک تهران)

دانشکده مهندسی کامپیوتر

مهندسی نرم افزار ۱

"فاز ۵ پروژه"

موضوع پروژه:

پیاده سازی

استاد درس: دکتر کلباسی

مهلت تحویل: یکشنبه ۲۳ دی ۱۴۰۳

نیم سال اول ۱۴۰۳

پیاده سازی خدمت (میکروسرویس)

در این پروژه، سامانه‌ای برای پردازش و بهبود متن‌های فارسی طراحی می‌شود که شامل چندین خدمت است. هر خدمت عملکرد خاصی دارد و در کنار هم یک ابزار جامع برای مدیریت و ویرایش متن فارسی فراهم می‌کنند. هر گروه وظیفه طراحی و پیاده‌سازی یک میکروسرویس مشخص را دارد که باید به سرویس اصلی متصل شود. سرویس اصلی توسط تیم مدرس ساخته شده و شما فقط میکروسرویس‌های خود را توسعه داده.

تعریف میکروسرویس‌ها

میکروسرویس‌ها مجموعه‌ای از سرویس‌های کوچک، مستقل و خودمختار هستند که هر یک وظیفه‌ای مشخص و مرتبط را بر عهده دارند. این معماری به توسعه‌دهندگان اجازه می‌دهد تا بخش‌های مختلف یک نرم‌افزار را به صورت جداگانه طراحی، پیاده‌سازی و مدیریت کنند.

ارتباط میان میکروسرویس‌ها معمولاً از طریق رابط برنامه‌نویسی کاربردی (API) برقرار می‌شود. هر تیم باید میکروسرویس خود را توسعه دهد به گونه‌ای که بتواند به سرویس کلی متصل شده و بخشی از سیستم جامع را تشکیل دهد. این اتصال از طریق API ها یا پروتکل‌های ارتباطی استاندارد انجام می‌شود.

برای یادگیری نحوه ارتباط میکروسرویس‌ها در Django، می‌توانید از مقاله یا منبع زیر استفاده کنید:

[Building Microservices with Django](#)

در پیاده‌سازی می‌توانید از هر زبان برنامه‌نویسی استفاده کنید. فقط توجه داشته باشید در صورت استفاده از زبان یا framework دیگری (غیر از django) نیاز دارید ارتباط بین میکروسرویس خارجی خود، با سرور django که در اختیار شما قرار می‌گیرد برقرار شود که برای جابه‌جایی داده پیشنهاد می‌شود از grpc یا message broker (مثل rabbitmq) استفاده کنید. (البته که روش‌های دیگری هم موجود هست که به انتخاب خود می‌توانید استفاده کنید).

به نکات زیر توجه بفرمایید:

- برای استفاده از rabbitmq فایلی در پروژه وجود دارد که می‌توانید از آن استفاده کنید.
- grpc در واقع یک framework متن‌باز برای پیاده‌سازی remote procedure call یا همان rpc هست که یکی از راه‌های برقراری ارتباط میان ۲ process در دو ماشین remote هست که در حال حاضر از محبوب‌ترین راه‌ها برای برقراری ارتباط در معماری microservice میان سرویس‌ها هست. شما می‌توانید برای برقراری ارتباط سرویس خود با core django پروژه از grpc استفاده کنید.

موارد مربوط به استفاده از پایگاه داده و سرور جنگو:

در سرور جنگو داده شده یک دیتابیس ابری با مشخصات برای شما قرار داده شده است. اطلاعات مورد نیاز برای وصل شدن به این دیتابیس در قسمت README پروژه در اختیار شما قرار می‌گیرد. شما بر حسب نیاز خود و نوع میکروسرویسی که دارید می‌توانید table های خود را در این دیتابیس برای ثبت اطلاعات کاربران و ورود آن‌ها و همچنین table خاص میکروسرویس خود برای ذخیره متن یا دریافت متن ورودی و یا ثبت تاریخچه‌ها بر حسب نیاز استفاده کنید. علاوه بر این می‌توانید از دیتابیس local خود جنگو نیز که برای هر گروه جداگانه است در بخش model و migration استفاده کنید.

url های مورد نیاز برای ثبت نام و یا ورود کاربران در پوشه Registration قرار گرفته و می‌توانید در بخش url.py برنامه خودتان، مسیرهای مورد نیاز خود را ایجاد کنید.

توجه داشته باشید چون از قبل table هایی در دیتابیس ابری وجود دارد، برای عدم ابهام و تداخل، هر جدولی که ایجاد می‌کنید را با پیشوند G2_GroupNumber ایجاد کنید.

در سرور جنگو برای هر تیم یک اپ تعریف شده است (بر اساس شماره هر گروه)، شما باید پیاده‌سازی‌ها و اقدامات خود را در فولدر مربوط به گروه خود انجام دهید.

رابط کاربری:

هر تیم باید یک رابط کاربری برای میکروسرویس خود طراحی کند. این رابط کاربری باید به صورت مستقیم از میکروسرویس تیم استفاده کند و عملکردهای اصلی میکروسرویس را برای کاربر قابل دسترس و قابل استفاده کند.

تیم‌ها باید تلاش کنند رابط کاربری ساده، کارآمد و متناسب با نیازهای پروژه طراحی کنند.

تست میکروسرویس:

لطفاً توجه داشته باشید که در پروژه خود، باید به همراهی تدریس‌یار گروه خود اقدام به انجام تست‌های دستی کنید. این تست‌ها برای بررسی کارکرد درست نرم‌افزار و اطمینان از صحت عملکرد شما ضروری هستند و به شناسایی مشکلات احتمالی کمک می‌کنند.

در صورتی که از تست‌های خودکار برای پروژه خود استفاده کنید، شامل نمره امتیازی خواهد شد (جنگو به طور پیش فرض از فریم‌ورک تست `unittest` پایتون استفاده می‌کند، پیشنهاد میشود از آن استفاده کرده تا علاوه بر افزایش کیفیت کد، نمره بهتری کسب کنید).

ساختار پروژه

در شاخه اصلی ریپازیتوری برای هر تیم یک فولدر ایجاد شده است. لطفاً تمام موارد مربوط به پروژه خود را فقط و فقط در فولدر مربوط به تیم خود انجام دهید. برای مثال در صورتی که شما گروه ۱ هستید، تغییرات خود را در فولدر `group1` ایجاد کنید.

اگر قسمت‌های دیگر و ساختار کلی نیاز به تغییر داشت لطفاً ابتدا برای هماهنگی اقدام کنید.

برای هر تیم یک `Application` در نظر گرفته شده است که به نام آن گروه نام‌گذاری شده است. همچنین یک `url` در نظر گرفته شده است به فایل `urls.py` در `Application` خود تا بتوانید برای پروژه خود در فولدر مربوط به گروه خود `url` تعریف کنید.

در صورتی که نمی‌خواهید با django کار کنید می‌توانید میکروسرویس خود را با هر زبانی که می‌خواهید توسعه دهید و به پروژه django با روش‌های گفته‌شده متصل کنید.

برای پیاده‌سازی میکروسرویس‌های مرتبط با سرویس پردازش زبان، پروژه اصلی را دانلود کرده و تنها در پوشه مربوط به پروژه خود تغییرات ایجاد کنید. تمامی درخواست‌هایی که به سرویس اصلی Django می‌رسند، باید به یک Sub-URL خاص هدایت شوند. این URL مربوط به میکروسرویس شما خواهد بود و شما می‌توانید API های مختلف مرتبط با پردازش لازم را در آن تعریف کنید.

برای مثال، در پروژه شما، برای میکروسرویس «تکمیل خودکار کلمه» می‌تواند شامل مسیرهای زیر باشد:

/autocomplete

/autocomplete/suggestions

یا برای میکروسرویس «تصحیح املاي کلمات» مسیرهای زیر قابل تعریف هستند:

/spellcheck

/spellcheck/checkword

شما می‌توانید با اضافه کردن مسیرهای مختلف، تمامی درخواست‌ها را به میکروسرویس خود هدایت کرده و سرویس خود را مدیریت کنید.

فرایند مهندسی نرم افزار

تیم‌ها باید حداقل سه اسپرینت برای پیشبرد پروژه خود در نظر بگیرند. با این حال، در صورت نیاز و تشخیص تیم، امکان افزایش تعداد اسپرینت‌ها وجود دارد.

۱. مدیریت تسک‌ها و زمان‌بندی اسپرینت‌ها:

- تعریف تسک‌ها و زمان‌بندی اسپرینت‌ها بر عهده تیم‌ها خواهد بود.
- تیم‌ها باید خودشان برنامه‌ریزی کرده و وظایف را به‌صورت گروهی یا انفرادی تخصیص دهند.

۲. ثبت تسک‌ها در جیرا:

- تمام وظایف و فعالیت‌ها باید در ابزار **Jira** ثبت شود و به هر فرد assign شود تا وضعیت پیشرفت پروژه به‌صورت شفاف قابل مشاهده باشد.
 - هر تسک باید توسط یک نفر انجام شود، و در صورتی که تسکی حجم زیادی دارد بهتر است شکسته شود به تسک‌های کوچک‌تر و هر نفر هر یک از تسک‌های کوچک‌تر را انجام دهد.
 - اعضای تیم باید وضعیت تسک‌ها را به‌روزرسانی کنند تا پایان اسپرینت مشخص باشد.
- در پایان هر اسپرینت، جلسه‌ای با تدریس‌یار مربوطه برگزار شود تا روند پیشرفت کار ارزیابی شود و سوالات یا مشکلات احتمالی بررسی شوند.

رعایت کدنویسی تمیز

- در تمامی مراحل از نام گذاری های واضح و با معنا استفاده کنید.
- توابع در حد امکان باید کوچک باشند و تنها یک وظیفه را انجام دهند.
- از نام گذاری های طولانی نترسید؛ یک نام مناسب که مفهوم خود را توضیح دهد بهتر از نام های کوتاه و مبهم است.
- کدهای خود را به صورت تدریجی و به مرور زمان کامیت کنید. پیام های کامیت نیز باید با معنا و واضح باشند.
- سعی کنید در تیم خود قراردادهایی برای کدنویسی تعریف کنید تا خروجی نهایی یکدست و منسجم باشد.
- برای کدهای خود مستندسازی کنید تا در صورت نیاز به مراجعه در آینده، فهم آن آسان تر باشد.
- مشابه فایل requirements.txt که در پروژه استفاده می شود و با دستور زیر نیازمندی ها را نصب می کند:

`pip install -r requirements.txt`

*برای میکروسرویس خود نیز یک فایل requirements تعریف کرده و تمام نیازمندی های لازم برای اجرای آن را در آن قرار دهید.

- برای Application خود در فولدر مربوط به گروه خود یک فایل README ایجاد کرده و توضیحات لازم را در آن قرار دهید.

ارسال کد در گیت هاب

آدرس ریپازیتوری:

[لینک گیت هاب](#)

برای توضیحات کامل، به فایل README ریپازیتوری مراجعه کنید.

برای ارسال کد خود، باید مراحل زیر را انجام دهید:

۱. ریپازیتوری را Fork کنید.
۲. ریپازیتوری Fork شده را Clone کنید.
۳. تغییرات خود را در یک Branch جدید انجام دهید و سپس آن را Commit & Push کنید.
۴. درخواست Pull Request ارسال کنید.

- اگر در مورد استفاده از گیت هاب سوال، پیشنهاد یا مشکلی دارید، می توانید از طریق تلگرام یا ایمیل با راه های ارتباطی زیر زیر در ارتباط باشید:

@k3rn3lpanic

matin.ghiasvand1381@gmail.com

- اگر در رابطه با سرور جنگو سوال، پیشنهاد یا مشکلی داشتید می توانید از طریق تلگرام یا ایمیل با راه های ارتباطی زیر زیر در ارتباط باشید:

@Hartage8186

mohammad.sh8186@gmail.com

آنچه باید تحویل دهید

یک فایل zip. که حاوی پروژه نهایی شما است (آخرین نسخه از کدتان که کامیت کردید)

نکات تحویل

- دقت شود برای ایجاد یک تجربه کاربری ساده و آسان، تمامی این خدمات باید با رابط کاربری کاربرپسند همراه باشند تا کاربران بتوانند به راحتی از هر خدمت بهره ببرند.
- دقت کنید که تمام مراحل این فاز پروژه به صورت گروهی دنبال شود.
- فایل zip خواسته شده را سرگروه در سایت courses آپلود کند.
- فرمت نام فایل zip به صورت P5_GroupName.zip باشد که در آن GroupName نام گروه است.