طراحی سامانه‌های میکروسرویس

تمرین چهارم

استادان درس:
علی ابریشمی
سیدحسین میرحسینی

نیم‌سال بهار ۱۴۰۵

دانشکده مهندسی کامپیوتر

پیش از انجام تمرین حتما نکات زیر را مطالعه فرمایید:

۱. پیش از اقدام به انجام تمرین، حتماً حداقل یک بار اسلایدهای درس را مرور نمایید.

۲. استفاده از هوش مصنوعی به منظور تحقیق و جستجو مانعی ندارد. اما سپردن وظیفه نوشتن پاسخ‌های نهایی به آن مجاز نیست و در صورت تشخیص نمره سوال یا سوالات مربوطه برای شما صفر در نظر گرفته خواهد شد.

۳. هم‌فکری بین دانشجویان موردی ندارد و تشویق می‌شود، اما اشتراک‌گذاری پاسخ‌ها طبق آیین‌نامه‌های اداره آموزش دانشگاه ممنوع بوده و در صورت تشخیص با هر دو طرف خاطی مطابق با قوانین و شیوه‌نامه‌های آموزش برخورد خواهد شد.

۴. تمرین سه بخش نظری و پژوهشی و عملی دارد که بخش نظری ۳۵ درصد، بخش پژوهشی ۲۵ درصد، و بخش عملی ۴۰ درصد نمره را داراست. همچنین این تمرین ۱نمره از کل نمره درس را تشکیل می‌دهد. به منظور افزایش دقت نمره‌دهی نمره کل این تمرین از ۱۰۰ لحاظ خواهد شد.

۵. پیشنهاد می‌شود از منابع و کتب خارجی و همچنین مقالات علمی و پروژه‌های متن‌باز [a] برای پاسخگویی هرچه دقیقتر استفاده فرمایید. البته لازم به ذکر است که قید کردن منابع در انتهای پاسخ‌نامه از شما انتظار می‌رود.

۶. سیاست جریمه تاخیر و تاخیر مجاز مطابق آنچه که در ابتدای نیم‌سال تصویب شد عملی می‌شود.

۷. آنچه که نهایتاً در سامانه بارگذاری می‌کنید، باید یک فایل فشرده با فرمت `zip`. باشد که ساختار و قالب نام‌گذاری آن به صورت زیر است:

file_format.txt

```
MS-HW4-[STDID].zip
|
|--MS-HW4-[STDID].pdf
|--HW4-Practical/
|        |
|        |-- *
```

که `STDID` شماره دانشجویی شماست. فایل PDF پاسخ سوالات نظری[b] به همراه گزارش بخش عملی[c] را داراست و فایل‌های بخش عملی در پوشه `HW4_Practical` قرار می‌گیرند.

۸. در انجام تمرینات عملی درس به طور معمول محدودیتی در انتخاب زبان و ابزارها ندارید مگر اینکه صراحتاً در ابتدای تمرین عملی قید شود. با این حال استفاده از زبان پایتون به منظور سادگی و تسریع انجام تمرین توصیه می‌شود. همچنین در گزارش خود باید هرکدام از پکیج‌ها و کتاب‌خانه‌ها و فریم‌ورک‌ها و ... که استفاده کردید را مختصراً توضیح داده و علت وجود آن را توجیه نمایید.

موفق باشید.

---

[a] Open-source Projects
[b] Theoretical
[c] Practical

بخش نظری

۱ تعریف مفاهیم (۶ نمره)

هرکدام از موارد زیر را با ادبیات خود و به زبان ساده توضیح دهید.

۱. Domain Coupling

۲. Data Decomposition

۳. Event-Driven Communication Pattern

۲ مشکل عملکرد (۹ نمره)

در یک پروژه فروشگاه آنلاین، تیم توسعه از ابتدای کار سیستمشان را به تعداد زیادی میکروسرویس مستقل با دیتابیس و Deployment جداگانه تقسیم می‌کند، در حالی که دامنه مسئله هنوز به‌خوبی شناخته نشده و نیاز به مقیاس‌پذیری واقعی وجود ندارد. پس از مدتی توسعه با مشکلاتی مانند پیچیدگی ارتباطات بین سرویس‌ها، وابستگی شدید بین APIها، افزایش باگ‌های مرزی، دشواری تست‌های End-to-End و کاهش سرعت تحویل فیچرها مواجه می‌شود. با توجه به این سناریو، مشکل معماری اصلی را شناسایی کرده و توضیح دهید چرا این تصمیم از نظر طراحی سیستم نادرست بوده و چه رویکردی جایگزین مناسبی می‌توانست باشد.

۳ Synchronous vs. Asynchronous (۱۰ نمره)

در یک سیستم مبتنی بر معماری میکروسرویس، سرویس ثبت سفارش برای تکمیل فرآیند خرید باید با سرویس‌های پرداخت و ارسال ایمیل ارتباط برقرار کند. تیم معماری بین استفاده از ارتباط Synchronous و ارتباط Asynchronous (مانند Message Broker و Event-Driven) مردد است. با توجه به معیارهایی مانند Coupling، Latency، Fault Tolerance و Scalability توضیح دهید تفاوت این دو رویکرد چیست و در چه شرایطی هر کدام انتخاب مناسب‌تری خواهند بود.

۴ طراحی سامانه میکروسرویس (۱۰ نمره)

یک پلتفرم آنلاین آموزش مهارت راه‌اندازی شده است که کاربران می‌توانند در آن ثبت‌نام کنند، دوره‌ها را مشاهده و خریداری کنند، ویدئوها را تماشا کنند، تمرین ارسال کنند و گواهی پایان دوره دریافت کنند. مدرس‌ها می‌توانند دوره جدید ایجاد کنند، محتوا بارگذاری کنند و وضعیت پیشرفت دانشجویان را ببینند. همچنین سیستم دارای بخش پرداخت، امتیازدهی، اعلان‌ها و پیشنهاد دوره‌های مرتبط است. انتظار می‌رود در زمان برگزاری کمپین‌ها یا انتشار دوره‌های محبوب، تعداد کاربران به‌صورت ناگهانی افزایش یابد. تیم توسعه تصمیم دارد این سامانه را با معماری Microservices طراحی کند.

با در نظر گرفتن مفاهیمی مانند Bounded Context، مالکیت داده، نحوه ارتباط Synchronous و Asynchronous بین سرویس‌ها، مدیریت تراکنش‌های توزیع‌شده و الزامات مقیاس‌پذیری، معماری پیشنهادی خود را طراحی کنید. سرویس‌های اصلی را مشخص کرده، مسئولیت هرکدام را توضیح دهید و نحوه تعامل آن‌ها را تشریح نمایید.

بخش پژوهشی

ابزارهای مهم

درباره هرکدام از ابزارهای زیر تحقیق کنید و کاربرد هر یک را در سامانه‌های میکروسرویس با ذکر یک مثال واقعی توضیح دهید:

(آ) nginx

(ب) Prometheus

(پ) RabbitMQ

(ت) Celery

(ث) ELK

(ج) Redis

(چ) Kubernetes

(ح) Kafka

(خ) Jaeger

نهایتاً هم یک دیاگرام سطح بالای معماری برای یک سامانه دلخواه ترسیم کنید که همه این موارد در آن استفاده شده.

بخش عملی

۴ برنامه مدیریت کارها

در این تمرین، باید یک API مبتنی بر GraphQL طراحی و پیاده‌سازی نمایید. موضوع پروژه «سیستم مدیریت کارهای روزانه (Mini Task Manager)» است که در آن کاربران می‌توانند کارهای جدید ایجاد کنند، اطلاعات آن‌ها را ویرایش یا حذف نمایند و وضعیت هر کار را میان حالت‌های TODO، DOING و DONE تغییر دهند. هر کار می‌تواند شامل عنوان، توضیحات، تاریخ سررسید و مجموعه‌ای از برچسب‌ها باشد. هدف از این تمرین آشنایی عملی با طراحی Schema، تعریف انواع داده و مدیریت ارتباط بین موجودیت‌ها در GraphQL است. در پیاده‌سازی این برنامه لازم است Schema مناسبی برای موجودیت‌های Task و Tag طراحی شود و قابلیت‌هایی مانند دریافت لیست کارها، فیلتر بر اساس وضعیت، دریافت یک کار مشخص بر اساس شناسه، ایجاد، ویرایش و حذف کار و همچنین تغییر وضعیت آن فراهم گردد. استفاده از فناوری‌هایی مانند Node.js به همراه Apollo Server یا Python با استفاده از کتابخانه‌هایی نظیر Strawberry یا Graphene مجاز است. ذخیره‌سازی داده‌ها می‌تواند با استفاده از یک پایگاه داده سبک مانند SQLite یا حتی ساختار درون‌حافظه‌ای انجام شود.