الف) در این بخش ترجیح آن است که یک backend مجزا برای aggregate کردن پاسخ micro-service های مختلف در نظر گرفته شود. چیزی شبیه به پروژه­ای که با نام apigateway در github قرار داده شده است. ابتدا لیست کالاهای قابل نمایش از review-service دریافت شده و پس از آن قیمت محصولات از سرویس enquiry استعلام میشود. بعد از آن تجمیع نتایج سرویسها انجام شده و در نهایت لیست نهایی برای کاربر نمایش داده خواهد شد.

تنها ورودی برای انجام این کار id یا correlation\_id کاربری خواهد بود که قرار است اطلاعات برای آن نمایش داده شود که آن هم یا از طریق session ای که در آن لاگین کرده و یا از طریق توکنی که در Header درخواست ارسال نموده قابل احصا خواهد بود. خروجی هم لیستی از محصولات به همراه قیمت و میانگین امتیاز اخذ شده و سه کامنت آخر خواهد بود.

همچنین در صورت رخداد time-out ما به ازای هر یک از سرویسهایی که نتایج آنها میبایست تجمیع شوند، می­توان تصمیم گرفت که نتیجه نهایی بصورت ناقص ارائه شوند و یا اینکه اصلا هیچ نتیجه ای بازگشت داده نشود. این بستگی به میزان اهمیت اطلاعات سرویسی است که time-out می­شود.

ب) خروجی این بخش نیز در پروژه­ای با نام review-service در github قرار داده شده است.

دیتابیس h2 بصورت فایلی در نظر گرفته شده و تعدادی رکورد تستی نیز در آن وارد شده است. شیوه اتصال در محیط intellij نیز در فایلی با نام h2-conn-in-intellij.jpg در مسیر src/main/resources قرار داده شده است.

خروجی postman نیز در شاخه اصلی پروژه review-service قرار داده شده است.

نقاط بهبود:

در هر دو پروژه بهتر آن است که خروجی ها بصورت Pageable در نظر گرفته شوند که صرفا همان بخش از دیتا که قرار است نمایش داده شود استخراج شود.

در پروژه های micro-service ای بهتر است یک پروژه مشترکی نیز تعریف شود که data type های مشترک در آن قرار گیرند که مجبور به تعریف چند باره آنها نباشیم.

در نهایت هر micro-service ای که end-piont هایی را به دنیای بیرون expose میکند میبایست توسط auth-server محافظت شود و دسترسی به سرویسهای آن تنها با توکن معتبر امکان پذیر باشد.

در نگاشت داده ها­ می­توان از پروژه هایی همچون map-struct یا dozer-mapper یا مشابه آنها نیز بهره گرفت.

قابل نمایش بودن یا نبودن در اینجا در سطح محصول در نظر گرفته شده اما میتوان آن را ما به ازای گروه های کاربری نیز تعریف نمود که به نظر منطقی تر است.

مکانیسم آنکه کامنت و نظر به طور کلی برای عموم باز باشد یا نه در اینجا بصورت مقادیری در application.properties لحاظ شده که در نهایت در سرویس comment به هنگام ثبت نظر و امتیاز میبایست کنترل شود. لازم به ذکر است که این مقادیر را در جدولی جداگانه نیز میتوان ذخیره نمود که امکان تغیر آن از طریق کنسول مدیریتی راحت تر باشد.