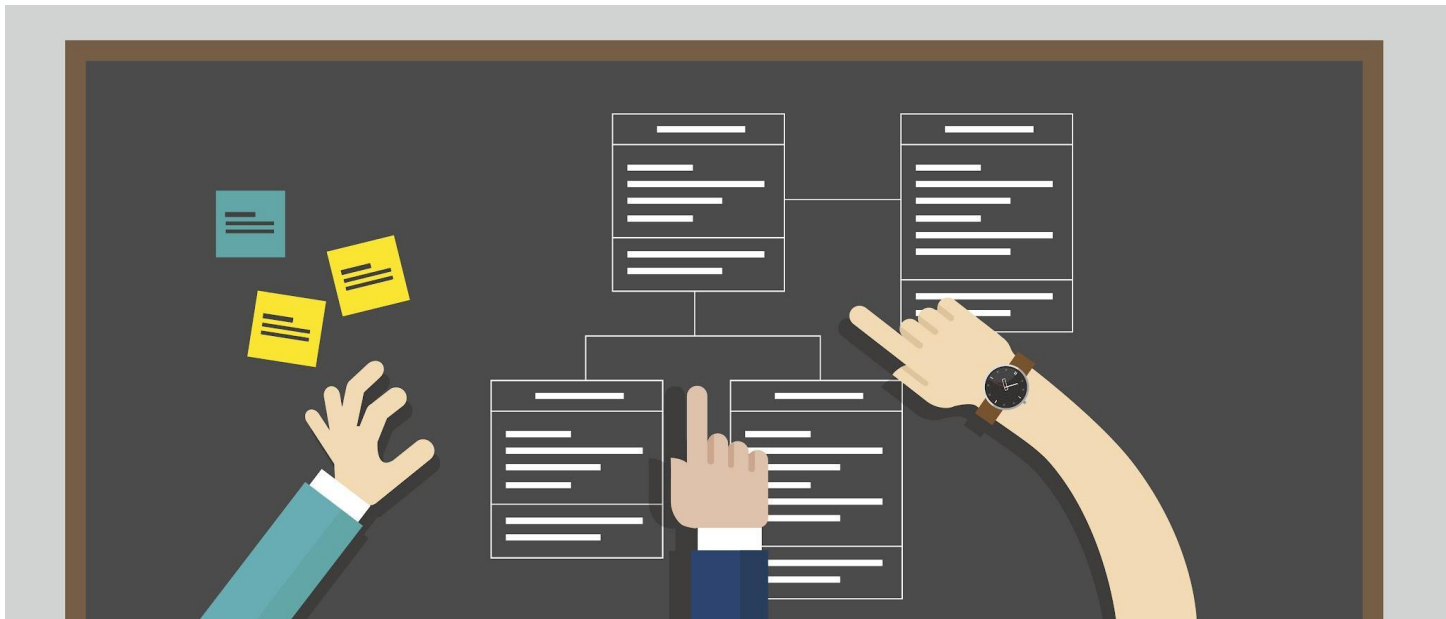


به نام خدا

تحلیل و طراحی سیستم ها

فاز ۵ پروژه

اعضای گروه : امیرحسین پرهیزگار - علی پورزنگنه - محمد فاتح



- تصمیم های طراحی

- createPrimaryRequest

- شی primaryRequeset را چه کسی بسازد؟

- بنا بر اصل Low Coupling ترجیح دادیم آن را کنترل بسازد.

- چرا که هم primaryRequeset باید کاربر را ببیند. هم کاربر بتواند به

- primaryRequeset های خودش دسترسی داشته باشد.

- لیست primaryRequeset ها را آیا در UserModel ذخیره کنیم؟

- بنا بر اصل expertDoer باید آن را به کلاس Workflow دهیم

- همچنین در غیر این صورت بین UserModel و primaryRequeset

- وابستگی دو طرفه بوجود می آید.

- AddHealthPackage

- با اشیاء مختلفی سرو کار داریم. مثلا اشیاء مربوط به خود بسته درمانی،

- مرحله های بسته درمانی و... که هرکدامشان پیچیدگی کار خودشان را

- دارند.

- بنا بر اصل controller بهتر است یک کلاس جدا برای مدیریت بسته های

- درمانی بوجود آوریم. نام این کلاس را PackageController میگذاریم.

- چه کسی Package را بسازد؟ مدل کلینیک یا کنترلر؟ بنا بر اصل creator ما

- کنترلر را انتخاب کردیم. چرا که کنترلر هم اطلاعات اولیه را دارد هم مدام از

- package استفاده میکند.

- UserModel

- در ابتدا برای یوزر یک کلاس کنترل داشتیم که عملیات ها را انجام میداد. و یک

- کلاس userModel برای صرفا ذخیره سازی داده های کاربر. متوجه شدیم

- userModel عملا کار خاصی انجام نمیدهد. و صرفا یک بار اضافه بر روی کنترلر

- است. اما با توجه به اصل high cohesion تصمیم گرفتیم آن را نگه داریم. چرا که

- ذخیره سازی در دیتابیس پیچیدگی های خود را دارد. در صورتی که بخواهیم شمای

- داده کاربر را تغییر دهیم دیگر نیازی به اعمال تغییر در کلاس های دیگر نیست.

○ جدا سازی PackageCatalog و ResidenceCatalog

همانطور که از نامشان پیداست یکی از کلاس ها مربوط به نمایش و نگه داری بسته های درمانی و دیگری مربوط به اقامت گاه هاست. در اول این دو در یک کلاس بودند. اما بنا بر اصل high cohesion تصمیم بر جدا سازی به دو کلاس مجزا شد.