Aşağıdaki kavramları örneklerle açıklayın. (10 PUAN)

Unit test

Kodun izole bir parçasının beklenen şekilde çalışıp çalışmadığını tespit eden test türüdür. Örnek olarak uygulamaya kayıt olmaya olanak sağlayan formu, üye olma işlevi üzerinden test edebiliriz.

Integration test

Yazılımdaki birçok bileşenin etkileşimde bulunduğu test türüdür. Bir veri tabanı ile etkileşim söz konusu ise Unit testin aksine bu işlem gerçekten yapılır, bu yüzden maliyetlidir. Örnek olarak bir uygulamaya kayıt olmanın yanı sıra kayıt işleminden sonra verinin veri tabanında doğru saklanıp saklanmadığını test edebiliriz.

Aşağıdaki özellikleri kredinbizde uygulamasına ekleyin. (40 PUAN)

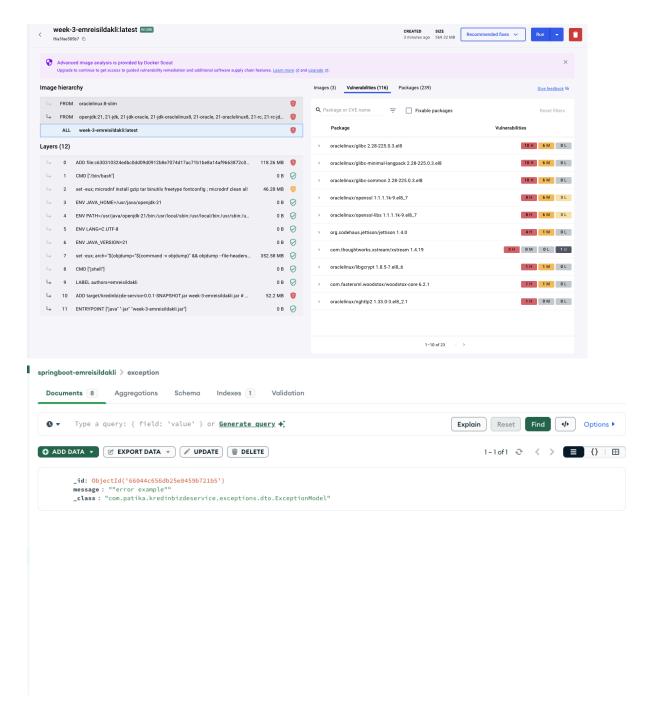
- Kredinbizde uygulaması için Garanti bankası için client oluşturun.
- Kullanıcı Garanti bankası için kredi başvurusu yapabilmelidir.
- Kullanıcı yapmış olduğu bütün başvuruları görüntüleyebilmelidir.
- NotificationService için EMAIL,SMS,MOBILE\_NOTIFICATION işlemlerini strateji pattern kullanarak implemantasyonu gerçekleştirin.

Garanti service repo: <a href="https://github.com/emreisildakli/GarantiService">https://github.com/emreisildakli/GarantiService</a>
Notification service repo:

https://github.com/emreisildakli/week-3-emreisildakli-notificationService

Docker ile Kafka kurulumu gerçekleştirin ve exception alındığı durumlarda Kafka ile bütün loglar toplanmalı ve MongoDB üzerinde kaydedilmelidir. (20 PUAN)





Birinci ödevde yaptığınız müşteri, fatura, sipariş ve ürün ile oluşturduğunuz projenizi ayrı servisler olacak şekilde bölün ve aşağıdaki geliştirmeleri ekleyin. (30 PUAN)

## https://github.com/emreisildakli/OnlineShop

https://github.com/emreisildakli/customer-service

- GW ekleyin. => https://github.com/emreisildakli/OnlineShopGW
- Discovery server ekleyin. => https://github.com/emreisildakli/OnlineShop-discovery
- Fatura bilgilerinin asenkron oluşmasını sağlayan yapıyı kurun.
- GlobalExceptionHandler ekleyin.