

g191210018 Mert Can Yılmaz - 2A

Yazılım Testi - Ödev 2

Proje yapısı:

Projede Program.java dosyası içinde Chrome driver ile sistemde yüklü chrome'a mı bağlanılacağına, yada standalone docker chrome sürümüne mi bağlanılacağına karar verilir. Uygulama ayağa kaldırılırken "local" parametresi verilmişse yüklü Chrome driver ile bilgisayarda yüklü Chrome sürümüne bağlanılmaya çalışılır. Bu method ile çalıştırılan işlemler anlık olarak kullanıcı tarafından chrome arayüzünden gözlemlenebilir. 2. bağlantı seçeneği olan "remote" parametresi ise docker image'i üzerindeki chromeda işlevleri gerçekleştirir ve bu yapılan işlemlerin grafik arayüz üzerinde gösterimi olmamasına karşı testler aynı şekilde sonuç verir ve sürüm farklılıklarından kaynaklanan problemlerin oluşmamasını sağlar.

Program.java içinde oluşturulan driver SeleniumTestRunnerCMD class'ı oluşturulurken parametre olarak verilir, bu class içinde ise çalıştırılacak tüm testlerin birer instance'ı oluşturulur.

Uygulama ilk başlatıldığında gösterilen menü bu class ile yazdırılır ve kullanıcıdan input bekler. Test edilen uygulamanın sayfalarına göre testler 5 grup altında toplanmıştır. Bunlar

g1: Inventory sayfasında uygulanan testler

g2: Login ekranında yapılan testler.

g3: Sepet ekranında çalıştırılan testler

g4: Sipariş tamamlama ekranı 1. kısımda çalıştırılan testler(Sepetteki ürünleri görüntüleme)

g5: Sipariş tamamlama ekranı 2. kısımda çalıştırılan testler(Kullanıcı bilgilerini girme)

Bu testler <https://www.saucedemo.com/> adresi üzerinde yapılmıştır. Bu web sitesinin seçilmesinin sebebi üzerinde test çalıştırmak için dummy bir web sitesi olması ve ticari bir web sitesi olmadığı için ödevin kontrol tarihine kadar arayüz üzerinde bir değişiklik olması ihtimalinin daha az olması sebebi ile seçilmiştir.

Uygulama içindeki test çeşitliliğine birkaç örnek verecek olursak:

Login error selenium testlerinde inputlara veri girişleri yapıp hatalar kontrol edilmiş,
Inventory selenium testlerinde liste elementlerini sıralaması, stilleri, responsive davranış,
sepete ekleme işlevi, bu işlevler sonrası stil değişimleri,
Sepet selenium testlerinde, seçilenlerin doğru gösterilip gösterilmediği sayfa işlevleri,
Checkout ilk adım testlerinde form işlevleri, hataları,
Checkout ikinci adım testlerinde veri doğruluğu ve sayfa işlevleri test edilmiştir.

Xpath çağrıları inventory selenium testlerinde kullanılmıştır. Ayrıca animasyonlu drawer'ın açılmasında da WebDriverWait kullanılarak elementin oluşması beklenmiştir. Web sitesinin işlevleri genel olarak farklı açılardan test edilmiştir .

Uygulamada cmd üzerinde çalıştırılan her test hataların önüne geçmek adına uygulama state'ini sıfırlamaktadır.

Projenin Dockerize Edilmesi:

Proje dockerize lib klasörü içinde bulunan tüm kütüphaneler derlenirken relative pathler kullanılıyor. Manifest.txt dosyası da jar oluşturulurken kullanılıyor. Bu oluşturulan docker image'i docker compose dosyası için app olarak bulunuyor. Ayrıca dockerize edilmiş ortam içinde chrome bağımlılığını da selenium/standalone-chrome imagi ile sağlamış oluyoruz. Bu sayede selenium ve chrome arasında oluşacak versiyon uyumsuzluğu problemlerinin önüne geçmiş oluyoruz.

Tüm bu işlemleri tetiklemek için "docker compose stop && docker-compose run app" komutunu çalıştırmamız yeterli. Bu komut dockerfile üzerindeki işlemleri çalıştırıp imagi hazırlayacak. Sonrasında selenium/standalone-chrome imagini ayağa kaldırıp healthcheck servisine belli aralıkla istek atacak. Ve bu servisin kalktığını anladığı anda test uygulamasını ayağa kaldıracak. docker compose stop ise sonraki ayağa kaldırmada chrome servisin yeniden başlatılabilmesi için durdurulmasını sağlıyor. Yoksa bağlantı problemi oluşuyor. Uygulamanın gerekli yüklemeler yapıldıktan sonra ayağa kalkması i7 3630qm işlemcili bir arc linux makinada ortalama 10 saniye sürüyor. Uygulama açıldığında böyle bir menü bizi karşılıyor:

Menu:

menu: show commands

exit: exits from app

g1: Inventory Page Test

g2: Login Page Errors Tests

g3: Cart Page Selenium Tests

g4: Checkout Step One Tests

g5: Checkout Step Two Tests

Execute command (For Available commands type help):

Kullanıcıdan bir girdi bekliyor. Grup testleri, g1,g2,g3,g4 ve g5 yazılarak çalıştırılabilir.

Örn:

Execute command (For Available commands type help):g1

First Element Of List Test: Success :Sauce Labs Backpack=Sauce Labs Backpack

7

Sauce Labs Backpack

Sauce Labs Backpack

Sauce Labs Bike Light

Sauce Labs Bolt T-Shirt

Sauce Labs Fleece Jacket

Sauce Labs Onesie

Test.allTheThings() T-Shirt (Red)

A to Z order Test: Success :[Test.allTheThings() T-Shirt (Red), Sauce Labs Onesie, Sauce Labs Fleece Jacket, Sauce Labs Bolt T-Shirt, Sauce Labs Bike Light, Sauce Labs Backpack, Sauce Labs Backpack]=[Test.allTheThings() T-Shirt (Red), Sauce Labs Onesie, Sauce Labs Fleece Jacket, Sauce Labs Bolt T-Shirt, Sauce Labs Bike Light, Sauce Labs Backpack, Sauce Labs Backpack]

7

Test.allTheThings() T-Shirt (Red)

Test.allTheThings() T-Shirt (Red)

...

Uygulama içinde karşılaşıp sürekli üretilmemiş ve nadiren oluşan birkaç hata durumu var.
Uygulama ayağa kalktıktan sonra uzun bir süre beklendiğinde uygulama bir kere hata verdi