Yazılım Kalite Güvence Temelleri Projesi

Hazırlayanlar;

Atiye Bengü YANPAR

Burak YALÇIN

Furkan MİNARECİLER

**UI TEST SONUÇLARI**

| **Test Adı** | **Açıklama** | **Gerekli Veri** |
| --- | --- | --- |
| **1. Login başarılı** | **Geçerli kullanıcıyla giriş** | **furkanmnr / 12345** |

metin, ekran görüntüsü, yazı tipi içeren bir resim

Yapay zeka tarafından oluşturulmuş içerik yanlış olabilir.

from selenium import webdriver

from selenium.webdriver.common.by import By

import time

driver = webdriver.Chrome()

driver.get("https://parabank.parasoft.com/parabank/index.htm")

# Giriş bilgilerini doldur

driver.find\_element(By.NAME, "username").send\_keys("furkanmnr")

driver.find\_element(By.NAME, "password").send\_keys("12345")

driver.find\_element(By.XPATH, "//input[@value='Log In']").click()

# Sayfanın yüklenmesini bekle

time.sleep(3)

# Sayfa içeriğini yazdır

print(driver.page\_source

# Giriş başarılı mı kontrol et

if "Accounts Overview" in driver.page\_source:

print("Giriş başarılı, test geçti.")

else:

print("Giriş başarısız, test kaldi.")

driver.quit()

| **Test Adı** | **Açıklama** | **Gerekli Veri** |
| --- | --- | --- |
| **2. Login hatalı** | **Hatalı şifre ile giriş** | **john / 1234** |

metin, ekran görüntüsü, yazı tipi içeren bir resim

Yapay zeka tarafından oluşturulmuş içerik yanlış olabilir.

from selenium import webdriver

from selenium.webdriver.common.by import By

import time

driver = webdriver.Chrome()

driver.get("https://parabank.parasoft.com/parabank/index.htm")

# Giriş bilgilerini doldur

driver.find\_element(By.NAME, "username").send\_keys("john")

driver.find\_element(By.NAME, "password").send\_keys("1234")

driver.find\_element(By.XPATH, "//input[@value='Log In']").click()

# Sayfanın yüklenmesini bekle

time.sleep(3)

# Sayfa içeriğini yazdır

print(driver.page\_source)

# Giriş başarılı mı kontrol et

if "Accounts Overview" in driver.page\_source:

print("Giriş başarılı, test geçti.")

else:

print("Giriş başarısız, test kaldi.")

driver.quit()

| **Test Adı** | **Açıklama** | **Gerekli Veri** |
| --- | --- | --- |
| **3. Form boşken login** | **Giriş alanları boşken giriş denemesi** | **-** |

metin, ekran görüntüsü, yazı tipi içeren bir resim

Yapay zeka tarafından oluşturulmuş içerik yanlış olabilir.

from selenium import webdriver

from selenium.webdriver.common.by import By

import time

# Tarayıcıyı başlat

driver = webdriver.Chrome()

driver.get("https://parabank.parasoft.com/parabank/index.htm")

# Kullanıcı adı ve şifreyi boş bırak, doğrudan login tıkla

driver.find\_element(By.NAME, "username").send\_keys("")

driver.find\_element(By.NAME, "password").send\_keys("")

driver.find\_element(By.XPATH, "//input[@value='Log In']").click()

# Bekleme (sayfa yönlenme süresi için)

time.sleep(2)

# Hata mesajı var mı kontrol et (beklenen mesaj)

if "Please enter a username and password." in driver.page\_source:

print("Boş giriş test PASSED. Hata mesajı gösterildi.")

elif "The username and password could not be verified." in driver.page\_source:

print("Boş giriş test PASSED. Sistemde doğrulama reddedildi.")

else:

print("Boş giriş testi FAILED. Beklenen mesaj yok.")

driver.quit()

| **Test Adı** | **Açıklama** | **Gerekli Veri** |
| --- | --- | --- |
| **4. Kayıt formu doldurma** | **Register sayfasında tüm alanları doldur ve kaydet** | **-** |

metin, ekran görüntüsü, yazı tipi içeren bir resim

Yapay zeka tarafından oluşturulmuş içerik yanlış olabilir.

from selenium import webdriver

from selenium.webdriver.common.by import By

import time

import random

driver = webdriver.Chrome()

driver.get("https://parabank.parasoft.com/parabank/register.htm")

time.sleep(1)

# Form alanlarını doldur

driver.find\_element(By.ID, "customer.firstName").send\_keys("Bengü")

driver.find\_element(By.ID, "customer.lastName").send\_keys("Yanpar")

driver.find\_element(By.ID, "customer.address.street").send\_keys("Deneme Mah. 12. Sk.")

driver.find\_element(By.ID, "customer.address.city").send\_keys("Ankara")

driver.find\_element(By.ID, "customer.address.state").send\_keys("Çankaya")

driver.find\_element(By.ID, "customer.address.zipCode").send\_keys("06420")

driver.find\_element(By.ID, "customer.phoneNumber").send\_keys("05321234567")

driver.find\_element(By.ID, "customer.ssn").send\_keys("123-45-6789")

# Rastgele kullanıcı adı

username = "bengu" + str(random.randint(1000, 9999))

driver.find\_element(By.ID, "customer.username").send\_keys(username)

driver.find\_element(By.ID, "customer.password").send\_keys("test123")

driver.find\_element(By.ID, "repeatedPassword").send\_keys("test123")

# Kayıt butonuna tıkla

driver.find\_element(By.XPATH, "//input[@value='Register']").click()

time.sleep(2)

# Başarı mesajını kontrol et

if "Your account was created successfully" in driver.page\_source:

print(f"Kayıt başarılı. Kullanıcı adı: {username}")

else:

print("Kayıt başarısız. Mesaj bulunamadı.")

driver.quit()

| **Test Adı** | **Açıklama** | **Gerekli Veri** |
| --- | --- | --- |
| **5. Para transferi** | **Giriş yaptıktan sonra "Transfer Funds" formunu doldur** | **-** |

metin, ekran görüntüsü, yazı tipi, siyah içeren bir resim

Yapay zeka tarafından oluşturulmuş içerik yanlış olabilir.

from selenium import webdriver

from selenium.webdriver.common.by import By

import time

# Giriş bilgileri (daha önce register.py ile oluşturulan kullanıcı olmalı)

username = "furkanmnr"

password = "12345"

# Tarayıcıyı başlat

driver = webdriver.Chrome()

driver.get("https://parabank.parasoft.com/parabank/index.htm")

# Giriş işlemi

driver.find\_element(By.NAME, "username").send\_keys(username)

driver.find\_element(By.NAME, "password").send\_keys(password)

driver.find\_element(By.XPATH, "//input[@value='Log In']").click()

time.sleep(2)

# Transfer Funds sayfasına git

driver.find\_element(By.LINK\_TEXT, "Transfer Funds").click()

time.sleep(1)

driver.find\_element(By.ID, "amount").send\_keys("500")

from\_account = driver.find\_element(By.ID, "fromAccountId")

to\_account = driver.find\_element(By.ID, "toAccountId")

from\_account.click()

to\_account.click()

# Transfer butonuna tıkla

driver.find\_element(By.XPATH, "//input[@value='Transfer']").click()

time.sleep(2)

# Başarı mesajı kontrolü yapıyoruz

if "Transfer Complete!" in driver.page\_source:

print("Transfer başarılı şekilde gerçekleşti.")

else:

print("Transfer başarısız veya mesaj bulunamadı.")

driver.quit()

| **Test Adı** | **Açıklama** | **Gerekli Veri** |
| --- | --- | --- |
| 6. Yeni hesap açma | “Open New Account” sayfası dropdown seçimi + submit | - |

metin, ekran görüntüsü, yazı tipi, siyah içeren bir resim

Yapay zeka tarafından oluşturulmuş içerik yanlış olabilir.

from selenium import webdriver

from selenium.webdriver.common.by import By

from selenium.webdriver.support.ui import Select

import time

# Giriş bilgileri

username = "furkanmnr"

password = "12345"

# Tarayıcıyı başlat

driver = webdriver.Chrome()

driver.get("https://parabank.parasoft.com/parabank/index.htm")

# Giriş işlemi

driver.find\_element(By.NAME, "username").send\_keys(username)

driver.find\_element(By.NAME, "password").send\_keys(password)

driver.find\_element(By.XPATH, "//input[@value='Log In']").click()

time.sleep(2)

# "Open New Account" sayfasına git

driver.find\_element(By.LINK\_TEXT, "Open New Account").click()

time.sleep(1)

#SAVINGS veya CHECKING

account\_type\_dropdown = Select(driver.find\_element(By.ID, "type"))

account\_type\_dropdown.select\_by\_visible\_text("SAVINGS")

existing\_account\_dropdown = Select(driver.find\_element(By.ID, "fromAccountId"))

existing\_account\_dropdown.select\_by\_index(0)

# Submit (Aç butonuna tıkla)

driver.find\_element(By.XPATH, "//input[@value='Open New Account']").click()

time.sleep(2)

# Başarı mesajı kontrolü

if "Account Opened!" in driver.page\_source:

print("Yeni hesap başarıyla açıldı.")

else:

print("Hesap açma başarısız veya mesaj görüntülenemedi.")

driver.quit()

| **Test Adı** | **Açıklama** | **Gerekli Veri** |
| --- | --- | --- |
| **7. Logout testi** | **Giriş yaptıktan sonra logout butonu çalışıyor mu** | **-** |

metin, ekran görüntüsü, yazı tipi, siyah içeren bir resim

Yapay zeka tarafından oluşturulmuş içerik yanlış olabilir.

from selenium import webdriver

from selenium.webdriver.common.by import By

import time

# Giriş bilgileri

username = "furkanmnr"

password = "12345"

# Tarayıcıyı başlat

driver = webdriver.Chrome()

driver.get("https://parabank.parasoft.com/parabank/index.htm")

# Giriş işlemi

driver.find\_element(By.NAME, "username").send\_keys(username)

driver.find\_element(By.NAME, "password").send\_keys(password)

driver.find\_element(By.XPATH, "//input[@value='Log In']").click()

time.sleep(2)

# Logout bağlantısına tıkla

driver.find\_element(By.LINK\_TEXT, "Log Out").click()

time.sleep(2)

# Ana sayfaya dönüldü mü kontrol et

if "Customer Login" in driver.page\_source:

print("Çıkış işlemi başarıyla gerçekleşti.")

else:

print("Çıkış işlemi başarısız.")

driver.quit()

**1. Proje Özeti**

Bu proje kapsamında, test ortamı olarak parabank adlı web tabanlı bankacılık uygulaması seçilmiştir. Proje, yazılım kalite güvencesi süreçlerini simüle etmeye yönelik olarak, uygulamanın hem kullanıcı arayüzü (UI) hem de arka planda çalışan API servisleri üzerinde otomasyon testlerinin gerçekleştirilmesini hedeflemektedir. Selenium ile temel kullanıcı akışları test edilirken; Postman aracı ve HTTP istekleri aracılığıyla API’nin farklı endpoint’leri test edilmiştir. Bu kapsamda, kullanıcı giriş işlemleri, hesap oluşturma, para transferi, hesap özeti görüntüleme gibi temel bankacılık senaryoları uygulanarak sistemin işlevselliği, hata yönetimi ve kullanıcı deneyimi test edilmiştir.

**📄 2. Uygulama Analizi ve Test Kapsamı**

**2.1. Seçilen Uygulama**

Parabank, sanal bankacılık işlemlerinin test edilmesi için geliştirilmiş, web tabanlı ve kullanıcı etkileşimlerine açık bir demo uygulamadır. Uygulama, müşteri girişi, hesap işlemleri, para transferi ve yeni hesap açma gibi temel finansal işlemleri simüle etmektedir.

**2.2. Test Edilen Sayfalar ve Fonksiyonlar**

| **Sayfa** | **Test Edilen İşlev** |
| --- | --- |
| **Login Sayfası** | Kullanıcı adı ve şifre ile giriş, hatalı giriş kontrolü |
| **Register Sayfası** | Yeni kullanıcı oluşturma, tüm alanların doldurulması |
| **Transfer Funds** | Para transferi yapma, formun doğruluğu |
| **Open New Account** | Yeni hesap açma, hesap türü seçimi ve açılış işlemi |
| **Logout** | Oturum kapatma işleminin doğrulanması |
| **Accounts Overview** | Giriş sonrası hesap listesinin görüntülenmesi |

**2.3. UI Test Kapsamı (Selenium)**

UI testleri Python ve Selenium WebDriver kullanılarak gerçekleştirilmiştir. Aşağıdaki kullanıcı senaryoları otomasyon yoluyla test edilmiştir:

* Geçerli kullanıcı ile başarılı giriş
* Hatalı kullanıcı bilgisi ile başarısız giriş
* Boş form alanları ile giriş denemesi
* Kayıt formunun tam ve eksiksiz doldurulması
* Transfer Funds sayfasında para gönderme
* Open New Account sayfasında yeni hesap açma
* Giriş sonrası oturumun kapatılması (logout)
* Sayfa yönlendirme ve form doğrulama kontrolleri

**2.4. API Test Kapsamı (Postman)**

API testleri Postman kullanılarak HTTP istekleri ile gerçekleştirilmiştir. Parabank'ın RESTful servisleri doğrudan kullanıcıya açık olmamakla birlikte, bazı temel endpoint’ler aşağıdaki senaryolarda test edilmiştir:

* GET /services/bank/customers/12212/accounts → Kullanıcının hesaplarının listelenmesi
* GET /services/bank/customers/12212 → Müşteri bilgisi sorgulama
* POST /services/bank/createAccount → Yeni hesap oluşturma isteği (varsa)
* GET /services/bank/accounts/12345/transactions → Hesap hareketlerini listeleme

Parabank API Test Raporu

Bu rapor, Parabank API'sine yönelik gerçekleştirilen testleri ve her bir API çağrısının neden test edilebilir ya da edilemez olduğunu açıklamaktadır. Postman kullanılarak yapılan testler, oturum gerektirmeyen ve oturum gerektiren istekler olarak ayrılmıştır.

Parabank’te sabit olarak çalışan müşteri ID’leri:

| **Müşteri Adı** | **Kullanıcı Adı** | **Şifre** | **ID** | |
| --- | --- | --- | --- | --- |
| John Smith | john | demo | | 12212 |
| Demo Test | demo | demo | 12323 | |
| Mary Smith | mary | demo | 12434 | |

# **API Ana Adresi:**

https://parabank.parasoft.com/parabank/services/bank/

# **Örnek Endpoint:**

GET https://parabank.parasoft.com/parabank/services/bank/customers/12212

Bu, ID’si 12212 olan müşterinin bilgilerini verir.

# Örnek Script

Test senaryolarını koştururken dönen yanıtları handle edip testin başarılı ya da başarısız kontrolünü sağlayan yapılar.

metin, ekran görüntüsü, yazı tipi, beyaz içeren bir resim

Yapay zeka tarafından oluşturulmuş içerik yanlış olabilir.

# TC01 - GET Valid Customer Info

- Endpoint: GET /customers/12212  
- Açıklama: Geçerli bir müşteri ID ile yapılan istek.  
- Beklenen Sonuç: 200 OK ve müşteri bilgisi içeren XML.  
- Sonuç: ✅ Test edilebilir, oturum gerekmez.

metin, ekran görüntüsü, yazı tipi içeren bir resim

Yapay zeka tarafından oluşturulmuş içerik yanlış olabilir.

# TC02 - GET Customer Accounts (Oturum Gerekebilir)

- Endpoint: GET /customers/12212/accounts  
- Açıklama: Belirli bir müşterinin hesaplarını getirir.  
- Beklenen Sonuç: 200 OK veya 401 Unauthorized (oturum yoksa).  
- Sonuç: ⚠️ Test edilebilir, ancak oturum gerektirebilir. Test sonucu değişkenlik gösterebilir.

metin, ekran görüntüsü, yazı tipi içeren bir resim

Yapay zeka tarafından oluşturulmuş içerik yanlış olabilir.

# TC03 - GET Account Info

- Endpoint: GET /accounts/12455  
- Açıklama: Hesap detaylarını getirir.  
- Beklenen Sonuç: 200 OK  
- Sonuç: ✅ Test edilebilir, oturum gerekmez.

metin, yazı tipi, sayı, numara, ekran görüntüsü içeren bir resim

Yapay zeka tarafından oluşturulmuş içerik yanlış olabilir.

# TC04 - GET Account Transactions (Beklenen Hatalı Yanıt)

- Endpoint: GET /accounts/12455/transactions  
- Açıklama: Belirli bir hesaba ait işlemleri getirir.  
- Beklenen Sonuç: 200 OK veya 500 Internal Server Error  
- Sonuç: ✅ Test edilebilir ama veri olmayabilir, bu nedenle hata dönebilir.

metin, yazı tipi, ekran görüntüsü içeren bir resim

Yapay zeka tarafından oluşturulmuş içerik yanlış olabilir.

# TC05 - POST Transfer (Oturum Gerekir)

- Endpoint: POST /transfer  
- Body: fromAccountId=12455, toAccountId=12456, amount=100  
- Açıklama: Para transferi işlemi.  
- Beklenen Sonuç: 401 Unauthorized (çünkü oturum yok)  
- Sonuç: ❌ Doğrudan test edilemez, oturum gerektirir. 401 alınması beklenen bir sonuçtur.

# TC06 - POST Transfer Invalid Body

- Endpoint: POST /transfer  
- Body: Eksik fromAccountId alanı  
- Amaç: Eksik parametreyle gönderilen transfer isteğinin hata döndürdüğünü test etmek  
- Beklenen Sonuç: 400 Bad Request  
- Gerçekleşen: 400 dönerse test başarılıdır.metin, yazı tipi, sayı, numara, çizgi içeren bir resim

Yapay zeka tarafından oluşturulmuş içerik yanlış olabilir..

metin, yazı tipi, ekran görüntüsü, beyaz içeren bir resim

Yapay zeka tarafından oluşturulmuş içerik yanlış olabilir.

# TC07 - POST Create Account (Unsupported)

- Endpoint: POST /createAccount  
- Body: customerId, type, balance  
- Amaç: Desteklenmeyen bir servis çağrısının nasıl karşılandığını test etmek  
- Beklenen Sonuç: 404 Not Found, 403 Forbidden veya 500 Internal Error  
- Gerçekleşen: API bu işlemi desteklemiyorsa test yine de anlamlıdır.

metin, ekran görüntüsü, yazı tipi, sayı, numara içeren bir resim

Yapay zeka tarafından oluşturulmuş içerik yanlış olabilir.

metin, yazı tipi, beyaz, ekran görüntüsü içeren bir resim

Yapay zeka tarafından oluşturulmuş içerik yanlış olabilir.

# TC08 - PUT Update Account Info

- Endpoint: PUT /accounts/12455  
- Body: JSON veri ile hesap tipi güncelleme denemesi  
- Amaç: PUT işleminin sistemde engellenip engellenmediğini gözlemlemek  
- Beklenen Sonuç: 403, 405 veya 500 hata kodu  
- Gerçekleşen: PUT desteklenmediği için alınan hata testin başarılı olduğunu gösterir.

metin, yazı tipi, ekran görüntüsü içeren bir resim

Yapay zeka tarafından oluşturulmuş içerik yanlış olabilir.

# TC09 - DELETE Account (Desteklenmeyen)

- Endpoint: DELETE /accounts/12455  
- Amaç: Hesap silme gibi tehlikeli bir işlemin engellenip engellenmediğini test etmek  
- Beklenen Sonuç: 403, 405 veya 500 hata kodu  
- Gerçekleşen: DELETE işlemi başarısız olmalı; aksi takdirde güvenlik açığı olabilir.  
metin, yazı tipi, ekran görüntüsü içeren bir resim

Yapay zeka tarafından oluşturulmuş içerik yanlış olabilir.

## Genel Sonuç

Bu testlerde yalnızca oturum gerektirmeyen API çağrıları doğrudan test edilebilmektedir. Oturum kontrolü olan çağrılar için ise 'beklenen başarısızlık' testleri yazılmıştır. Bu yaklaşım, sistemin oturum güvenliğinin testlerde nasıl yansıtıldığını göstermek amacıyla kullanılmaktadır. Sonuç olarak 10 testten hepsi başarılı olmuştur.