

TATWiN – PROJEKT
TESTOWANIE STRONY: AMAZON.COM
AUTOR: KRZYSZTOF PROCHOŃ, U-14849

Spis treści:

1. Wstęp
2. Opis plików w projekcie
3. Opis przypadków testowych

1. Wstęp.

Projekt został zrealizowany zgodnie z przekazanymi wytycznymi. Na jego potrzeby zostało utworzone jedno konto. Cały projekt został przeprowadzony przy pomocy RobotFrameworka w pythonie.

2. Opis plików w projekcie.

Plik MyRunner.py zawiera wyzwalacz testów oraz jest w nim utworzone tworzenie logów z wykonanych testów.

Plik Tests.robot zawiera kolejność wykonywania test case-ów wraz z niezbędnymi im argumentami.

Plik MyKeywords.robot zawiera w sobie utworzone PageObjecty do obiektów na stronie, na których wykonywane będą testy oraz określa poszczególne test case'y.

Pliki Logs.html oraz Report.html zawierają kolejno logi oraz raport z wykonanych testów.

3. Opis przypadków testowych.

Test nr 1 – rejestracja.

Na samym początku testów uruchamiana jest przeglądarka (w tym wypadku chrome z parametrami startowymi "--start_maximized". Przeglądarka uruchamiana jest na stronie amazon.com. Następnie skrypt testowy przechodzi na stronę logowania, a następnie przechodzi na podstronę odpowiedzialną za rejestrację nowego konta.

- a. Użytkownik nie podał nazwy użytkownika - w tym teście sprawdzamy zachowanie strony w momencie, gdy użytkownik nie poda nazwy użytkownika (pola email, oraz hasło i jego powtórzenie uzupełniamy kolejno następującymi wartościami [aaaa@aaa.pl 123456 123456]. Po naciśnięciu przycisku "Create your Amazon account" sprawdzamy czy wyświetlił się komunikat o błędzie (nie podaniu nazwy użytkownika), jeśli się wyświetlił test przechodzi pozytywnie.
- b. Użytkownik podał nieprawidłowy format adresu email – sprawdzamy zachowanie w momencie błędnego adresu email. Dla poprawnego wykonania testów odświeżamy stronę. Wszystkie pola rejestracji wypełniamy kolejno wartościami [janusz87356586 aaauh.aa.pl 123456 123456]. W tym przypadku również sprawdzamy czy

wyświetlony został komunikat o błędzie - jeśli tak, test wykonany pozytywnie.

- c. Użytkownik podał zbyt krótkie hasło - zachowanie strony dla zbyt krótkiego hasła. Odświeżamy stronę. Podajemy wartości [janusz3532652 aaa@aaa.pl 123 123]. Jeśli wyświetlony został komunikat o błędzie - test przechodzi jako pozytywny.
- d. Hasła podane przez użytkownika nie są takie same – zachowanie strony dla różnych haseł. Odświeżamy stronę. Podajemy wartości [janusz3532652 aaa@aaa.pl 123456 654321]. Jeśli wyświetlono błąd - test przechodzi pozytywnie.

Test nr 2 – logowanie.

W przeglądarce zostaje otwarta nowa karta ze stroną amazon.com. Przechodzimy do strony logowania.

- a. Mail nie został podany - jeśli został wyświetlony błąd - test przechodzi pozytywnie.
- b. Konto z podanym mailem znajduje się w bazie – najpierw sprawdzamy zachowanie strony w przypadku podania błędnego adresu email [tatwintest20202pk@gazeta.pl]. Jeśli wyświetlił się błąd test jest uznawany jako pozytywny, strona jest odświeżana i próbujemy zalogować się na istniejące konto [tatwintest2020pk@gazeta.pl]. Jeśli nie został wyświetlony żaden błąd i przeszliśmy do strony z hasłem test przechodzi jako pozytywny.

- c. Podane hasło musi być prawidłowe - na początku podajemy błędne hasło [gshgfsh], jeśli wyświetlił się komunikat błędu, wszystko jest w porządku i podajemy poprawne hasło [123456]. Klikamy przycisk logowania. Jeśli zostaliśmy przeniesieni na stronę główną test przechodzi pozytywnie.

Test nr 3 – wyszukiwanie produktów.

Otwieramy kolejną kartę ze strona amazon.com. W polu wyszukiwarki wpisujemy interesujący nas produkt, w tym przypadku [headphones].

- a. Wyszukujemy produkt i sprawdzamy czy na stronie został wyświetlony komunikat o ilości wyświetlonych przedmiotów, jeśli taki się znajduje sprawdzamy czy zawiera liczbę 16 - jeśli tak, test przechodzi jako pozytywny.
- b. Sortowanie produktów według ceny – od najniższej do najwyższej. Klikamy dropdown i wybieramy opcję [Price: low to high], jeśli nic się nie wydarzy test przechodzi pozytywnie.
- c. Sortowanie produktów według ceny – od najwyższej do najniższej. Ponownie wyszukiujemy [headphones], klikamy dropdown i wybieramy [Price: high to low]. Jeśli nic się nie wydarzy test przechodzi pozytywnie.

Uwagi:

Czasami możemy otrzymać błąd informujący o braku klikanego elementu odpowiedzialnego za sortowanie produktów.

Test nr 4 – koszyk.

W tym teście sprawdzamy dodawanie produktów do koszyka oraz samą interakcję w koszyku. Otwierana jest nowa karta przeglądarki i po kolei przechodzone jest do stron kolejnych produktów, aby na końcu trafić do koszyka.

- a. Możliwość dodania 10 różnych produktów do koszyka:
 - i. Produkt nr 1 - przejście do strony produktu i dodanie do koszyka.
 - ii. Produkt nr 2 - przejście do strony produktu i dodanie do koszyka.
 - iii. Produkt nr 3 - przejście do strony produktu i dodanie do koszyka.
 - iv. Produkt nr 4 - przejście do strony produktu i dodanie do koszyka.
 - v. Produkt nr 5 - przejście do strony produktu i dodanie do koszyka.
 - vi. Produkt nr 6 - przejście do strony produktu i dodanie do koszyka.
 - vii. Produkt nr 7 - przejście do strony produktu i dodanie do koszyka.

- viii. Produkt nr 8 - przejście do strony produktu i dodanie do koszyka.
- ix. Produkt nr 9 - przejście do strony produktu i dodanie do koszyka.
- x. Produkt nr 10 - przejście do strony produktu i dodanie do koszyka.
- b. Przejście do koszyka.
- c. Zakomentowanie, że w koszyku znajduje się 10 przedmiotów.
- d. Zmiana ilości danego produktu – wybranie dropdown-list i zmiana ilości z 1 na [2]. Następnie zakomentowanie zmiany ilości produktów w koszyku.
- e. Usuwanie przedmiotu z koszyka - kliknięcie [delete] przy wybranym produkcie. Zakomentowanie, że produkt został usunięty z koszyka. Zamknięcie przeglądarki.

Uwagi:

Zmiana ilości produktów w koszyku jak i samo usunięcie zależne jest od ich kolejności. Każdy dropdown z ilością oraz sam przycisk [delete] posiada atrybuty do klasy konkretnego produktu jeśli występuje on w konkretnym miejscu. Możliwe, że w innej kolejności by nie zadziałały.