



Test Plan Execution Report (on specific build)

Test Project: Testowy projekt

Test Plan: Test Plan - Wyszukiwarka Kayak.com

Build: Regresja Ver. 0.0.9

Printed by TestLink on 03/07/2023

2012 © TestLink Community

Table Of Contents

1.1.Wyszukiwarka Kayak.com

TP-107: Weryfikacja walidacji

TP-108: Weryfikacja logiki wyszukiwarki

Test Plan: Test Plan - Wyszukiwarka Kayak.com

1. Wstęp

W przygotowanym dokumencie zostały zebrane informacje na temat działań testowych. Wyszczególnione zostały wszystkie komponenty oprogramowania, które zostaną poddane weryfikacji, oraz typy testów, jakie zostaną przeprowadzone. Głównym celem działań testowych jest dostarczenie interesariuszom informacji o jakości testowanego produktu.

2. Zakres Testów

Testy obejmują wszystkie funkcjonalności i wymagania wyszukiwarki kayak.com, w tym wyszukiwanie lotów, hoteli, wynajmu samochodów i pakietów podróży.

Realizowane typy testów:

- jednostkowe
- wydajnościowe
- funkcjonalne
- kompatybilności
- bezpieczeństwa

Typy testów, które nie zostaną przeprowadzone:

- testy automatyczne- ze względu na brak kwalifikacji w zespole
- testy integracji

3. Przedmioty testów

Przedmiotem poddawanych testom jest wyszukiwarka ze strony głównej kayak.com z uwzględnieniem całej logiki filtrowania, po odpowiednich polach.

4. Kryteria zaliczenia/niezaliczenia testów

Testy zostaną uznane za zaliczone, jeśli wszystkie wymagania/funkcjonalności zostaną przetestowane i spełnione zgodnie z wymaganiami. Testy zostaną uznane za niezaliczone, jeśli wystąpią poważne błędy lub nie spełnią wymagań/funkcjonalności.

5. Kryteria wejścia / wyjścia

- **kryteria wejścia:**
 - zakończona jest faza implementacji wyszukiwarki
 - działające i skonfigurowane środowisko testowe
 - dostęp do działającej i skonfigurowanej maszyny wirtualnej
- **kryteria wyjścia:**
 - wszystkie przypadki testowe zostały zakończone pomyślnie
 - komponent spełnia wszystkie ustalone założenia z ustalonej dokumentacji

6. Lista wymagań/funkcjonalności do wykonania

Załączenie wszystkich dokumentacji, user story, scenariuszy itp.

7. Środowisko testowe

- testowy serwer (development)
- Mac OS Ventura 13.0.1
- przeglądarki biorące udział w testach Google Chrome, Mozilla Firefox, Safari, Opera i Microsoft Edge.

8. Harmonogram testów

Przeprowadzenie testów funkcjonalnych:

- Przetestuj wyszukiwanie lotów z różnymi kombinacjami dat i miast.
- Przetestuj wyświetlanie wyników wyszukiwania wraz z cenami lotów, czasami i ilością przesiadek.
- Przetestuj wybieranie ofert, rezerwowanie biletów i płatności online.
- Przetestuj funkcje dodatkowe, takie jak rezerwacja hoteli, wynajem samochodów i ubezpieczenia.

Przeprowadzenie testów wydajnościowych:

- Weryfikacja czasu ładowania strony głównej.
- Weryfikacja średniego czasu odpowiedzi.
- Weryfikacja maksymalnej ilości requestów, przy jakiej przeglądarka działa stabilnie.
- Weryfikacja czasu wyszukiwania i wyświetlania wyników.

Przeprowadzenie testów bezpieczeństwa:

- Weryfikacja, czy strona jest bezpieczna i czy wyświetlany jest certyfikat SSL.
- Weryfikacja, czy strona nie ma podatności na ataki XSS, SQL Injection i inne ataki.

Przeprowadzenie testów jednostkowe:

- Testowanie funkcji filtrowania wyników: sprawdzenie, czy funkcje filtrowania wyników działają poprawnie dla różnych kryteriów, takich jak liczba przesiadek, cena i czas lotu.
- Testowanie funkcji sortowania wyników: sprawdzenie, czy funkcje sortowania wyników działają poprawnie dla różnych kryteriów, takich jak cena, czas lotu i liczba przesiadek.
- Testowanie funkcji generowania zapytań do API: sprawdzenie, czy funkcje generowania zapytań do API lotniczego działają poprawnie dla różnych parametrów wyszukiwania, takich jak miejsce wylotu, cel podróży, daty i liczba pasażerów.
- Testowanie funkcji przetwarzania odpowiedzi API: sprawdzenie, czy funkcje przetwarzania odpowiedzi API lotniczego działają poprawnie dla różnych formatów odpowiedzi, takich jak JSON i XML, oraz czy przetwarzają odpowiednie dane, takie jak kody lotów, daty, godziny i ceny.

Przeprowadzenie testów kompatybilności:

- Przetestuj działanie na różnych przeglądarkach internetowych, takich jak Google Chrome, Mozilla Firefox, Safari, Opera i Microsoft Edge.
- Przetestuj działanie na różnych wersjach przeglądarek.
- Przetestuj, czy strona działa poprawnie na różnych urządzeniach, takich jak smartfony i tablety.

9. Raport z testów

- lista zrealizowanych przypadków testowych wraz ze statusami
- pomiary z testów wydajnościowych
- inne raporty z testów

10. Lista narzędzi wykorzystanych w testach

- Jeter
- TestLink
- Jira
- Browserstack

11. Zarządzanie incydentami, błędami

W procesie testowym każdy wykryty błąd powinien być odpowiednio zareportowany do systemu Jira.

Uwzględniając priorytet błędu, przypisaną do niego osobę (developera) i komponent, którego dotyczy problem.

Zgodnie z flow przyjętym przez organizację, problem powinien zostać naprawiony przez developera i trafić do retestów.

12. Role i odpowiedzialność

Jakub Miga - wykonywanie przypadków testowych

Jakub Miga - projektowanie przypadków testowych

Jakub Miga - weryfikacja user story

13. Role i odpowiedzialność

W test planie dla wyszukiwarki kayak.com mogą występować różne rodzaje ryzyk, takie jak:

1. Ryzyko wydajności - aplikacja może działać wolno lub zbyt długo reagować na użytkownika
2. Ryzyko niezgodności - aplikacja może nie działać poprawnie w różnych przeglądarkach internetowych
3. Ryzyko bezpieczeństwa - aplikacja może być podatna na ataki hakerskie lub naruszenia prywatności
4. Ryzyko niekompletnych funkcjonalności - aplikacja może nie działać zgodnie z oczekiwaniami użytkowników

Aby zarządzać ryzykiem w test planie, należy podjąć następujące kroki:

1. Identyfikacja ryzyka - zidentyfikuj wszystkie potencjalne ryzyka, które mogą wpłynąć na testy i aplikację.
2. Analiza ryzyka - określ prawdopodobieństwo wystąpienia i wpływ każdego ryzyka na testy i aplikację.
3. Planowanie ryzyka - opracuj plan zarządzania ryzykiem, który określi, jakie działania zostaną podjęte w przypadku wystąpienia każdego ryzyka.
4. Monitorowanie ryzyka - monitoruj i kontroluj ryzyko podczas testów i na bieżąco aktualizuj plan zarządzania ryzykiem.
5. Wdrożenie działań zarządzania ryzykiem - podczas testów, jeśli wystąpią nieprzewidziane sytuacje, zastosuj odpowiednie działania z planu zarządzania ryzykiem.

1.1.Test Suite : Wyszukiwarka Kayak.com

| Test Case TP-107: Weryfikacja walidacji [Version : 1] | | | | |
|---|--|--|------------------|-------------------|
| <u>Author:</u> | | tsumee | | |
| #: | Step actions: | Expected Results: | Execution notes: | Execution Status: |
| 1 | Przechodzimy na stronę kayak.com | Wyszukiwarka wczytała się pomyślnie | | Passed |
| 2 | Usuujemy wartość z pola "Z?" i wartość "Do?" i klikamy w przycisk szukaj (lupka na pomarańczowym tle po prawej stronie) | Pojawia się komunikat walidacyjny: "Podczas wyszukiwania wystąpił błąd. Podaj lotnisko wylotu. Podaj lotnisko docelowe." | | Passed |
| 3 | 1. Uzupełniamy wartość w polu "Z?" frazą "Kraków" 2. Klikamy w przycisk lupki na pomarańczowym tle (Szukaj) | Pojawia się komunikat: "Musisz wybrać lotnisko": 1. Wiesz, dokąd chcesz lecieć. Podaj lotnisko w polu wyszukiwania. Opcja do wyboru: "Szukaj" 2. Nie wiesz, dokąd chcesz lecieć. Sprawdź, dokąd i za ile możesz polecieć Opcja do wyboru: "Odkrywaj" | | Passed |
| 4 | 1. Uzupełniamy wartość w polu "Z?" frazą "Kraków" 2. Uzupełniamy wartość w polu "Do?" frazą "Gdańsk" 3. Klikamy w przycisk lupki na pomarańczowym tle (Szukaj) | 1. Pojawia się strona flights/KRK-GDN z dostępnym wynikiem z danego zakresu 2. Widoczna jest fraza "3 frazy kolejno: • Najtańszy "X" zł wraz z czasem lotu (h i min) X - cena lotu w złotychkach • Najlepszy lot "X" zł wraz z czasem lotu (h i min) X - cena lotu w złotychkach • Najmniej CO2(Nowość) "X" zł wraz z czasem lotu (h i min) X - cena lotu w złotychkach 3. Widoczna jest boczne menu z filtrami do danego wyszukanego rezultatu. | | Passed |
| 5 | 1. W zakładce filtru "Przesiadki" w lewym bocznym panelu klikamy kolejno: • 0 przesiadek • 1 przesiadka • 2 przesiadki • itd. | 1. Dla 0 przesiadek: • Pojawiają się oferty bez przesiadek 2. Dla 1 przesiadki: • Pojawiają się oferty z 1 przesiadką 3. Dla 2 przesiadek: | | Passed |

| | | | | |
|--|--|---|--|--------|
| | | <ul style="list-style-type: none">Pojawiają się oferty z 2 przesiadką 4. ltd. | | |
| 6 | 1. W lewym panelu zaznaczamy w "Godziny lotów" godzinę 14:00 z Krakowa | 1. Zablokowane są wszystkie wcześniejsze godziny niż data zaznaczona na pasku | | Passed |
| 7 | 1. Klikamy w przycisk lupki na pomarańczowym tle w górnej części wyszukiwarki ("Szukaj") | <ul style="list-style-type: none">Strona zostanie przeładowanaWybrane filtry i wartości z głównego panelu wyszukiwania (Panel górny) zostają załadowaneWidoczne są rezultaty wyszukiwania | | Passed |
| <u>Execution type:</u> | | Manual | | |
| <u>Estimated exec. duration (min):</u> | | | | |
| <u>Priority:</u> | | Medium | | |
| <u>Requirements</u> | | None | | |
| <u>Keywords:</u> | | None | | |
| Execution Details | | | | |
| Build | | Regresja Ver. 0.0.9 | | |
| Tester | | tsumee | | |
| <u>Execution Result:</u> | | Passed | | |
| <u>Execution Mode:</u> | | Manual | | |
| <u>Execution duration (min):</u> | | 25.00 | | |

Test Case TP-108: Weryfikacja logiki wyszukiwarki [Version : 1]

Author: tsumee

Summary:

Warunki konieczne:

- Odpowiednia konfiguracja

| #: | Step actions: | Expected Results: | Execution notes: | Execution Status: |
|----|---|---|------------------|-------------------|
| 1 | Przechodzimy na stronę Kayak.com | Wyszukiwarka wczytała się pomyślnie | | Passed |
| 2 | Klikamy w pole "Z?" | Pojawiają się popularne miejsca w pobliżu | | Passed |
| 3 | Klikamy w pole "Do?" | Pojawiają się popularne miejsca turystyczne | | Passed |
| 4 | Wpisujemy w pole "Z?" frazę "Kraków" | Na pierwszej pozycji w podpowiedzi pojawia się lokalizacja "Kraków" | | Failed |
| 5 | Klikamy w pole datowe | Pojawia się Date Picker z dostępnym przedziałem | | Passed |
| 6 | Wybieramy dostępny zakres z Date picker'a | Zakres został poprawnie wprowadzony do pola datowego | | Passed |
| 7 | Klikamy w pole "Ilość osób" | <ul style="list-style-type: none">Pojawił się popup z możliwością wyboru:<ol style="list-style-type: none">Ilości Dorosłych +18Ilości Studentów +18Ilości Nastolatk 12-17 | | Passed |

| | | | | |
|--|--|---|--|--------|
| | | 4. Ilości Dzieci 2-11 5. Ilości Dzieci na siedzeniu <2 6. Niemowlęta na kolanach <2 | | |
| 8 | Dodajemy wartość "Ilość Studentów +18" | Pojawił się komunikat: "Wymagane potwierdzenie statusu studenta" | | Passed |
| <u>Execution type:</u> | Manual | | | |
| <u>Estimated exec. duration (min):</u> | | | | |
| <u>Priority:</u> | Medium | | | |
| <u>Requirements</u> | None | | | |
| <u>Keywords:</u> | None | | | |
| Execution Details | | | | |
| Build | Regresja Ver. 0.0.9 | | | |
| Tester | tsumee | | | |
| <u>Execution Result:</u> | Failed | | | |
| <u>Execution Mode:</u> | Manual | | | |
| <u>Execution duration (min):</u> | 30.00 | | | |

Execution time metrics

Time used for executing 2 test cases (min):55