

Plan testów

Paweł Goj

26.05.2021

Spis treści

1. Wprowadzenie	3
1.1. Cel dokumentu	3
1.2. Oczekiwania.....	3
1.3. Załączniki	3
2. Zakres i Założenia	3
2.1. Przedmiot testów	3
2.2. Zakres	3
2.3. Zakres testów	4
2.4. Strategia i techniki testowania	5
2.5. Strategia testów regresji	5
2.6. Metryki testów	6
3. Środowisko testowe	6
3.1. Założenia.....	6
3.2. Zasoby sprzętowe.....	6
1.2. Wymagane oprogramowanie.....	6
4. Narzędzia do przeprowadzenia i monitorowania testów oraz kanały komunikacji.....	7
4.1. Kanały komunikacji.....	7
4.2. Narzędzia do przeprowadzania i monitorowania testów:	7
5. Rezultaty testów.....	7
6. Kryteria zakończenia testów	7
7. Zasoby ludzkie	7
8. Harmonogram	8

1. Wprowadzenie

1.1. Cel dokumentu

Celem dokumentu jest nakreślenie planu testów sprawdzających poprawność implementacji aplikacji e-commerce (sklep internetowy), znajdującego się pod adresem: <http://skleptest.pl/>.

1.2. Oczekiwania

Faza testowania ma za zadanie sprawdzenie czy aplikacja e-commerce spełnia funkcjonalne oczekiwania użytkownika. Oczekiwanym rezultatem jest zmniejszenie ilości defektów obecnych w aplikacji wprowadzanej na środowisko produkcyjne w cotygodniowych cyklach. Przeprowadzane będą zarówno testy funkcjonalne systemowe na środowisku testowym i testy akceptacyjne już na środowisku produkcyjnym. Ze względu na brak specyfikacji technicznej, wymagania określone są na podstawie historyjek użytkownika i przypadków użycia. Funkcjonalne obszary aplikacji testowane w oparciu o historyjki użytkownika i przypadki użycia zostały nakreślone we wcześniej przygotowanej mindmapie (zob. Załączniki).

1.3. Załączniki

1. Mindmapa prezentująca obszary do przetestowania – 27.03.2021, autor: Paweł Goj

2. Zakres i Założenia

2.1. Przedmiot testów

Aplikacja aplikacji e-commerce (sklep internetowy), znajdującego się pod adresem: <http://skleptest.pl/>.

2.2. Zakres

Planowane jest przetestowanie następujących obszarów funkcjonalnych aplikacji:

1. Zgodność z najnowszymi wersjami przeglądarek: Google Chrome, Mozilla Firefox, Safari na systemach zarówno Windows jak i MacOS
2. Działanie Newslettera aplikacji
3. Funkcjonalność wyszukiwarki na stronie
4. Funkcjonalności związane z obszarem „Blog” aplikacji webowej.
5. Strony kart produktów
6. Funkcjonalności związane z koszykiem użytkownika

7. Proces rejestracji i logowania
8. Funkcjonalności konta użytkownika
9. Kara płatności
10. Formularze kontaktowe
11. Podstrona aplikacji „About Us”
12. Lokalizacja aplikacji.

Pełna lista obszarów wchodząca w skład aplikacji znajduje się we wcześniej przygotowanej mindmapie (zob. załączniki)

2.3. Zakres testów

1. Testy funkcjonalne związane z Newsletterem:
 - Zapisywanie do Newslettera
 - Rezygnacja z Newslettera
2. Testy funkcjonalne bloga:
 - Funkcjonalności związane z sekcją komentarzy.
 - Poprawność działania kalendarza na blogu.
3. Testy wyszukiwarki sklepu:
 - Testy poprawności wyszukiwania produktów.
4. Testy kart produktów:
 - Poprawność wyświetlania zdjęć produktów
 - Poprawność opisu produktu
 - Poprawność ceny
5. Testowanie koszyka:
 - Dodawanie produktu do koszyka
 - Usuwanie produktu z koszyka
 - Zmian ilości danego produktu w koszyku
 - Naliczanie sumy ceny produktów w koszyku
6. Testowanie funkcjonalności rejestracji:
 - Testy walidacji danych wpisywanych podczas rejestracji
 - Testy funkcjonalności wysyłania maili do osoby zarejestrowanej
7. Testy systemu logowania:
 - Testy systemu odzyskiwania hasła

- Testy logowania i wylogowywani osoby zarejestrowanej
 - Testowanie procesu kupowania produktu przez zarejestrowanego użytkownika
8. Testy procesu zakupu produktu przez niezarejestrowanego użytkownika
 9. Testy formularza kontaktowego:
 - Testy możliwości wysłania formularza kontaktowego
 - Walidacja danych w formularzu kontaktowym
 10. Testy zgodności z przeglądarkami:
 - Google Chrome, Mozilla Firefox, Safari na systemach zarówno Windows jak i MacOS
 11. Testy lokalizacji

2.4. Strategia i techniki testowania

Wykonywane będą testy funkcjonalne systemowe na środowisku testowym i testy akceptacyjne na środowisku produkcyjnym. Testy będą wykonywane w cotygodniowych sprintach zgodnie z metodyką scrum. Strategia prowadzenia testów będzie kombinacją strategii: analitycznej, minimalizującą regresję i reaktywnej. Priorytet testów będzie ustalany na podstawie poziomu ryzyka. Największy nakład pracy będzie kierowany w stronę najbardziej newralgicznych funkcjonalności związanych z procesem zakupu w aplikacji e-commerce. W cotygodniowych sprintach testów wykorzystywane będą już istniejące testalia w tym istniejące już przypadki testowe z wcześniejszych sprintów. Testy będą modyfikowane zgodnie ze strategią reaktywną na podstawie wiedzy uzyskanej ze wcześniejszych testów. W większości wykonywane będą testy manualne czarnoskrzynkowe. Testy będą wykonywane w oparciu o przypadki użycia na podstawie historyjek użytkownika. Podczas testów akceptacyjnych ok 20 % wykonywanych testów będą to testy eksploracyjne.

2.5. Strategia testów regresji

Testy regresji będą wykonywane dla przypadków testowych związanych z najważniejszymi funkcjonalnościami aplikacji takimi jak koszyk czy proces rejestracji, logowania. Będzie również określany wpływ wprowadzanych zmian na daną funkcjonalność tak by nie wykonywać nadmierowej ilości testów regresji. Wpływ zmian na funkcjonalność będzie określany na podstawie doświadczenia zespołu deweloperów i testerskiego oraz zdobytym doświadczeniu z poprzednich sprintów. Testy regresji kluczowych funkcjonalności będą wykonywane w każdym sprincie. Testy regresji pozostałych funkcjonalności tylko wtedy, gdy wpływ zmian w aplikacji na tą funkcjonalność będzie istotny.

2.6. Metryki testów

W celu monitorowania postów testów i stanu jakościowego produktu monitorowane będą następujące metryki:

1. Poziom wykonania przypadków testowych powstałych na podstawie historyjek użytkownika i dla testów regresji.
2. Informacje o gęstości defektów, liczbie wykrytych defektów i usuniętych defektach.
3. Rezultaty testów potwierdzających.
4. Pokrycie testowe historyjek użytkownika
5. Procent wykonania zaplanowanych prac

3. Środowisko testowe

3.1. Założenia

1. Testy funkcjonalne systemowe wykonywane są na specjalnie przygotowanym środowisku testowym.
2. Testy akceptacyjne są wykonywane na środowisku produkcyjnym, aplikacja znajduje się na serwerze jest ogólnie dostępna w Internecie pod adresem: <http://skleptest.pl/>
3. Testy wykonywane są na najnowszych przeglądarkach Google Chrome, Mozilla Firefox, Safari na systemach zarówno Windows jak i MacOS.

3.2. Zasoby sprzętowe

Do wykonania testów zgodnie z harmonogramem wymagane są następujące zasoby sprzętowe:

1. 6 x Komputer PC z kartą sieciową i połączeniem do Internetu

1.2. Wymagane oprogramowanie

1. MacOS Sierra 10.12 lub nowszy
2. Windows 7 lub nowszy
3. Najnowsza wersja Google chrome
4. Najnowsza wersja Mozilla Firefox
5. Najnowsza wersja Safari

4. Narzędzia do przeprowadzenia i monitorowania testów oraz kanały komunikacji

4.1. Kanały komunikacji

1. Defekty do teamu developerskiego w tygodniowych sprintach zgłaszane będą za pomocą oprogramowania Jira firmy Atlassian. Tam też będą przydzielane zadania poszczególnym testerom.
2. Bezpośrednia komunikacja wewnątrz temu testerskiego będzie prowadzona mailowo lub przez komunikator Slack.
3. Zespół Testerski może komunikować się z developerami mailowo.
4. Co tydzień organizowane będą spotkania w celu określenia trudności i postępów prac.

4.2. Narzędzia do przeprowadzania i monitorowania testów:

1. Test case'y będą pisane i przechowywane oprogramowaniu TestRail
2. Oprogramowanie Jira będzie służyć do śledzenia defektów i monitorowania postępu testów.

5. Rezultaty testów

1. Raporty o defektach po każdym sprincie.
2. Dokumentacja dotycząca poszczególnych przypadków testowych.
3. Po zakończeniu testów sumaryczny raport o testach.

6. Kryteria zakończenia testów

Testy kończą się wraz z zakończeniem projektu po 3 miesiącach od jego rozpoczęcia.

7. Zasoby ludzkie

1. Kierownik testów x 1
 - Planowanie i logistyka
 - Zdobywanie zasobów
 - Składanie raportów product menagerowi
2. Tester x 6
 - Implementacja testów

- Wykonywanie testów
- Składanie raportów kierownikowi

8. Harmonogram

1. Testy funkcjonalne systemowa na środowisku testowym – 3 dni w czasie 5 dniowego sprintu.
2. Testy akceptacyjne na środowisku produkcyjnym – 2 dni w czasie 5 dniowego sprintu.
3. Pisanie całościowego raportu z testów 2 dni po zakończeniu projektu.

Projekt trwa 3 miesiące (60 dni, 12 sprintów)