

Test plan

1. INTRODUCTION (wprowadzenie)

1.1 Purpose (cel)

Opracowanie i przedstawienie określonego zakresu procesu testowego aplikacji e-commerce testującego dodawanie przedmiotów do koszyka, edycję koszyka oraz finalny zakup .

1.2 Project overview (opis projektu)

Aplikacja ma dać możliwość użytkownikowi dodanie różnych przedmiotów do koszyka, możliwość korygowania zawartości koszyka ,otrzymywanie bonifikatów, oraz przejście do etapu finalizującego zakup

1.3 Audience (odbiorcy)

-zespół developerski składający się z :

- 3 FE (Front dev)
- 2 BE (Backend dev)
- 1 PO (Product Owner)
- 1 SM (Scrum Master)
- 1 PM (Project Manager)

-zespół testerski składający się z :

- 3 MT (Manual Tester - 2 Junior + 1 Mid)
- 1 AT (Junior Automation Tester)
- 1 TM (Test Manager)

2. TEST STRATEGY (strategia testów)

2.1 Tests goal (cel testu)

Poprawność działania aplikacji , identyfikacja błędów , retesty , testy regresji , sprawdzenie szybkości działania oraz zapewnienie poprawności działania

2.2 Test Assumptions (założenia)

Aplikacja jest gotowa do testów i działa w środowisku testowym .Zespół QA zna zasadę działania aplikacji . Dostępne są narzędzia do wykonania testów – licencja Browserstack . Dokumentacja jest zrozumiała dla całego zespołu .

2.3 Levels and types of testing (poziomy i typy testów)

2.3.1 Unit tests (testy jednostkowe)

Purpose: sprawdzenie poprawności modułu , klasy ,funkcji programu

Scope: moduł , klasa , funkcja ,metoda programu

Testers: zespół developerski : 3 FE (Front dev) , 2 BE (Beckend dev)

Method: testy automatyczne z poziomu kodu

Timing: przez cały czas tworzenia oprogramowania

2.3.2 Integration and system tests (testy integracyjne i systemowe)

Purpose: zminimalizowanie przedstawiania się defektów na wyższy poziom

Scope: interfejs , integracja systemów

Testers: zespół QA : 3 MT (Manual Tester - 2 Junior + 1 Mid) , 1 AT (Junior Automation Tester)

Method: testy manualne , automatyczne

Timing: po integracji dwóch systemów

2.3.3 Acceptance tests

Purpose: zgodność aplikacji z założeniami początkowymi

Scope: aplikacja

Testers: PO (Product Owner) , PM (Project Manager) oraz przedstawiciel firmy zlecającej projekt

Method: test manualny

Timing: prze wdrożeniem na produkcję

2.3.4 Exploratory tests

Purpose: zapoznanie się z aplikacją ,poprawność działania aplikacji

Scope: dodawanie kursów do koszyka , edycja zawartości koszyka- zmiana ilości zamówień- dodawanie i odejmowanie kursów , finalizacja zakupu

Testers: Manual tester i Automation Tester

Method: test manualny na urządzeniach stacjonarnych i mobilnych

Timing: przed wdrożeniem na produkcję oraz po

2.3.5 Functional tests (testy funkcyjne)

Purpose: sprawdzenie funkcjonalności aplikacji

Scope: aplikacja działa poprawnie

Testers: zespół QA

Method: testy manualne

Timing: gdy dana funkcja jest ukończona

2.3.6 Non-functional tests (testy нефunkcjonalne)

Purpose: szybkość dostępu do bazy danych , wydajność obciążeniowa serwera , bezpieczeństwo

Scope: określone нефunkcjonalne właściwości aplikacji

Testers: zespół QA : 3 MT (Manual Tester - 2 Junior + 1 Mid) , 1 AT (Junior Automation Tester)

Method: testy przeważnie automatyczne oraz manualne

Timing: jak najwcześniej

2.3.7 Automated Regression Testing (automatyczne testy regresji)

Purpose: sprawdzenie poprawności działania modułu po wprowadzeniu zmian w kodzie

Scope: moduł , system

Testers: zespół QA

Method: testy automatyczne

Timing: jak najwcześniej

2.4 Test deliverables (wyniki testów)

- test plan
- przypadki testowe
- raporty defektów
- skrypt testów automatycznych
- raport końcowy

2.5 Test effort estimation (estymacja zadań testowych)

QA activity	Test effort (MD)
1.Analiza dokumentacji	
2.Przygotowanie Test Planu	
3.Stworzenie przypadków testowych	
4.Wykonanie przypadków testowych	
5.Raport defektów	
6.Retesty	
7.Przygotowanie raportu końcowego	

1. EXECUTION STRATEGY (strategia wykonania)

3.1 Entry and exit criteria (kryteria początkowe i końcowe)

Entry criteria:

- dokumentacja jest kompletna
- dostępne są narzędzia do tworzenia , wykonywania i raportowania przypadków testowych
- gotowe środowisko testowe
-

Exit criteria:

- wykonano wszystkie założone przypadki testowe
- usunięto defekty
- brak błędów krytycznych i funkcjonalnych
- wykonano testy regresji
- utworzono raport końcowy

3.2 Test cycles (cykl testów)

3.3 Validation and defect management (sprawdzanie i zarządzanie defektem)

1. wykrycie defektu
2. zaraportowanie defektu
3. zwrócenie naprawionego defektu
4. retest defektu
5. po uzyskaniu wyniku pozytywnego zamknięcie defektu

3.4. Test Metrics (miary testów)

- procent wykonanych przypadków testowych

- procent pozytywnych przypadków testowych
- procent defektów w ogólnej ilości przypadków testowych

Metric	Formula

1. TEST MANAGEMENT PROCESS (zarządzanie procesem testowym)

4.1 Test management tool (narzędzia testowe)

- do zarządzania projektem Jira , Confluence
- do komunikacji w zespole i pomiędzy zespołami Slack , Microsoft Teams
- do komunikacji z serwerem Postman
- do tworzenia przypadków testowych i ich wykonywania Test Rail
- do zapisania kodów testów automatycznych GitHub
- do tworzenia testów automatycznych IntelliJ

4.2 Test design process (projektowanie procesu testowego)

- analiza wymagań
- planowanie testów
- tworzenie przypadków testowych
- wykonywanie przypadków testowych
- raportowanie przypadków testowych

4.3 Test execution process (proces wykonywania testów)

4.4 Test risks and mitigation factors

- mało doświadczony zespół testerski
- dodatkowe szkolenia , praca z mid/senior testerami
- problem z komunikacją
- używanie jednego języka
- wiele językowy zespół
- użycie innowacyjnego programu
- dokładne zapoznanie się z dokumentacją
- tworzenia oprogramowania
- tworzenia oprogramowania , zatrudnienie
- osoby znającej innowacyjny program

-

4.5 Test responsibility (odpowiedzialność)

4.5.1 QA team

3 MT (manual tester, 2 junior, 1 mid), 1 AT (junior automation tester), 1 TM (Test Manager)

Odpowiedzialny za opracowanie testów i przypadków testowych , wykonanie przypadków testowych , zgłaszanie błędów , wykonywanie retestów i testów regresji , prowadzenie dokumentacji

4.5.2 Development team

3 FE (frontend dev), 2 BE (backend dev), 1 PO (Product Owner), 1 SM (Scrum Master), 1 PM (Project Manager)

Odpowiedzialny za tworzenie dobrej jakości kodu , naprawa błędów , wykonywanie testów jednostkowych , zarządzanie projektem i zespołami

1. TEST ENVIRONMENT

- Windows 10, 11 32/64 bity , Android 10,11 ,iOS 12,13 , macOS
- przeglądarka Chrome , Firefox , Mozilla , Opera

1. TESTING TOOLS

Area	Tool
------	------

Narzędzie do zarządzania projektem	Jira
Narzędzia do zarządzania dokumentacją	Confluence
Narzędzie do testowania webservice	Postman
Narzędzie do tworzenia kodu	IntelliJ
Zarządzanie przypadkami testowymu	Test Rail