

## 1. INTRODUCTION

### 1.1 Purpose/Cel

Opracowanie planu testów i przetestowanie aplikacji ecommerce

### 1.2 Project overview/ Zarys Projektu

Aplikacja ecommerce umożliwiająca dodawanie produktów do koszyka, edytowanie jego zawartości oraz przechodzenie do etapu finalizacji zamówienia

### 1.3 Audience/ Zespoły

- zespół developerski (Polska, Czechy, Niemcy

zespół FE ( frontend dev)

zespół BE (backend dev)

- zespół testerski ( 3 lokalizacje)
- Product Owner
- Product Manager

## 2. TEST STRATEGY

### 2.1 Tests goal/ Cel testów

- sprawdzenie zgodności aplikacji z dokumentacją
- wykrycie błędów aplikacji
- retesting
- 

### 2.2 Test Assumptions/ Założenia testów

- środowisko testowe jest gotowe
- aplikacja działa
- dostępne są urządzenia mobilne ( android, iOS)

### 2.3 Levels and types of testing/ Poziomy i typy testów

#### 2.3.1 Unit tests/ Testy jednostkowe

**Purpose:** Cel

- sprawdzenie działania poszczególnych kodów

**Scope:** Rezultat

- pojedyncze kody

**Testers:** Testerzy

- zespół BE ( backend dev)

**Method:** Metoda

- testy automatyczne

**Timing:** termin/czas

- podczas tworzenia oprogramowania

### **2.3.2 Integration and system tests/Testy integracyjne**

**Purpose:** Cel

- sprawdzenie poprawności działania aplikacji

**Scope:** Rezultat

- przetestowanie poszczególnych funkcjonalności ( dodawanie produktów do koszyka, edytowanie koszyka, zmiany w koszyku, finalizacja zamówienia)
- sprawdzenie frontend'u z założeniami projektu, czytelność aplikacji, kolorystyka, położenie poszczególnych "przycisków", wyświetlanie produktów
- wykrycie defektów
- retesting

**Testers:** Testrzy

zespół testerski

**Method:** Matoda

- testy manualne

**Timing:** Termin/czas

- wykonywane po sprawdzeniu poszczególnych kodów i przygotowaniu środowiska testowego

### **2.3.3 Acceptance tests/ Testy akceptacyjne**

**Purpose:** cel

- akceptacja działania całej aplikacji

**Scope:** Rezultat

- potwierdzenie działania aplikacji
- potwierdzenie Frontend'u aplikacji zgodnego z założeniami dokumentacji projektu

**Testers:** Testerzy

- Project Manager
- Product Owner

**Method:** Metoda

- testy manualne

**Timing:** Termin/czas

- po wykonaniu testów integracyjnych i wykonaniu retestów

### **2.3.4 Exploratory tests/ Testy eksploracyjne**

**Purpose:** Cel

- sprawdzenie działania całej aplikacji

**Scope:** Rezultat

- cała aplikacja ( również urządzenia mobilne )

**Testers:** Testerzy

- zespół testerski
- Project Manager
- Product Owner
- Klient

**Method:** Metoda

- testy manualne/ automatyczne ( logowanie, dodawania produktów
- testy wydajnościowe

**Timing:** Termin/czas

- aplikacja już działa, wykonane testy integracyjne, wykonane retesty

### **2.3.5 Functional tests/ testy funkcjonalne**

**Purpose:** Cel

- testowanie funkcjonalności aplikacji

**Scope:** Rezultat

- cała aplikacja

**Testers:** Testerzy

- zespół testerski

**Method:** Metoda

- testy manualne
- testy automatyczne

**Timing:** Termin/czas

- aplikacja już działa, wykonane testy integracyjne, wykonane retesty

### **2.3.6 Non-functional tests/ Testy нефункционалне**

**Purpose: Cel**

- sprawdzenie aplikacji pod względem wydajnościowy
- sprawdzenia aplikacji pod względem wyglądu

**Scope: Rezultat**

- testy wydajnościowe aplikacji

**Testers:**Testerzy

- zespół testerski

**Method:**Metoda

- narzędzia testerskie/testy automatyczne

**Timing:** termin/czas

- aplikacja już działa, wykonane testy integracyjne, wykonane retesty

### **2.3.7 Automated Regression Testing/ Automatyczne testy regresji**

**Purpose: Cel**

- usprawnienie testów manualnych regresji

**Scope: Rezultat**

- poszczególne funkcje aplikacji

**Testers:** Testerzy

- zespół testerski

### **Method: Metoda**

- automatyczne

### **Timing: Termin/czas**

- aplikacja już działa, wykonane testy integracyjne, wykonane retesty

## **2.4 Test deliverables/ Wyniki testów**

- Plan testów
- raporty testów
- raporty defektów
- raport retestów
- testy automatyczne - kody

## **2.5 Test effort estimation/ Estymacja zadań testowych**

<b>QA activity/ aktywność- zadanie</b>	<b>Test effort (MD) - czas trwania</b>
1. Test Plan - przygotowanie	2MD
2. Analiza dokumentacji	2MD
3. Przygotowanie przypadków testowych	3MD
4. Wykonanie przypadków testowych	4MD
5. Raport defektów	2MD
6. Retesty	3MD
7. Regresja	3MD
8. Przygotowanie raportu końcowego	2MD

[L]  
[SEP]

1. EXECUTION STRATEGY Strategia wykonania

## **3.1 Entry and exit criteria Kryteria początkowe i końcowe**

### **Entry criteria: Kryteria początkowe**

- przygotowane środowisko testowe
- aplikacja jest przygotowana do testów
- dostępne są urządzenia mobilne do testowania

- przygotowane dostępy do aplikacji browserstack

#### Exit criteria: Kryteria końcowe

- wykonano wszystkie etapy i rodzaje testów
- wykonano wszystkie przypadki testowe
- usunięto defekty
- wykonano testy regresji
- przygotowano raport końcowy

### **3.2 Test cycles/ Cykl testów**

1. przygotowanie przypadków testowych
2. wykonanie przypadków testowych
3. stworzenie raportu defektu/założenie buga
4. wykonanie retestu
5. przygotowanie raportu z testów

### **3.3 Validation and defect management/ sprawdzanie i zarządzanie defektem**

1. wykrycie defektu
2. założenie buga
3. przekazanie Buga do zespołu BE/FE
4. po usunięciu defektu wykonanie retestów
5. brak defektu -zamknięcie buga
6. defekt nadal występuje lub pojawił się inny - założenie buga powrót pkt 2

### **3.4. Test Metrics/ Miary testów**

<b>Metric</b>	<b>Formuła</b>
ilość wszystkich przypadków testowych	
ilość poprawnych przypadków testowych	$\frac{\text{ilość przypadków poprawnych}}{\text{ilość wszystkich przypadków}}$
ilość defektów	$\frac{\text{ilość defektów}}{\text{ilości wszystkich przypadków testowych}}$

## **4. TEST MANAGEMENT PROCESS Zarządzanie procesem testowym**

### **4.1 Test management tool narzędzia testowe**

- przypadki testowe Jira
- defekty zakładane w Jira jako bugi

### **4.2 Test design process Projektowanie procesu**

- przypadki testowe tworzone będą na podstawie dokumentacji i utworzone zostaną w testrailu
- wykryte defekty będą raportowane w postaci założenie buga w Jira
- “bug” będzie przekazywany do odpowiedniego zespołu BE lub FE

#### 4.3 Test execution process wykonanie procesu testowego

- ?
- wykryte defekty będą raportowane w postaci założenie buga w Jira

#### 4.4 Test risks and mitigation factors

Risk	Probability	Impact	Mitigation plan
	średnie	średni	
	wysokie		
		wysoki	
1. Błędy w dokumentacji			
2. Problemy z komunikacją pomiędzy zespołami testerskim - 3 lokalizacje			rozpoczęcie testowania od analizy dokumentacji
3. rozproszona struktura zespołów QA, BE,FE- brak wymiany informacji, różnice geograficzne, kulturowe	wysokie		poprawne formułowanie zadań, postępowanie zgodne z procedurami, duża aktywność na spotkaniach, szczegółowe przygotowanie planu testów oraz zadań testowych
4. małe doświadczenie w zespole testerskim 2 juniorów		wysoki	aktywność na spotkaniach, SM - prawidłowa koordynacja prac zespołów, dyscyplina pracy, przygotowanie harmonogramu pracy i przystosowanie go do różnych zespołów (różne godziny pracy, różne systemy pracy)
5. zbyt mało testów automatycznych - przygotowanie testów automatycznych - 1 osoba	średnie		
		średni	koordynacja prac przez TM, popranie napisane zadania testowe
6. duża ilość retestów i testów regresji z uwagi na duże zależności pomiędzy systemami i rozproszone zespoły	średnie		zminimalizowanie testów automatycznych do koniecznego minimum
		średni	
	wysokie		

średni

## **4.5 Test responsibility**

### **4.5.1 QA team**

- przygotowanie planu testów
- szczegółowa analiza dokumentacji
- przygotowanie przypadków testowy
- przeprowadzenie testów
- założenie bugów
- retesty, testy regresji
- przygotowanie raportu końcowego

### **4.5.2 Development team**

- napisanie i przeprowadzenie unit testów
- debugowanie

## **5. TEST ENVIRONMENT**

- środowisko testowe- środowisko produkcyjne
- Windows 10,11; macOS
- przeglądarki: Chrome, Firefox, Mozilla, Safari, Opera
- urządzenia mobilne: Android iOS



## 6. TESTING TOOLS

Area	Tool
test plan,	Jira confluence
przypadki testowe wykonanie	testrail
raport z wykonania przypadków testowych	postman
tworzenie bugów	Jira
testy automatyczne	Selenium IDE, DEVtools, InteliJ
przygotowanie raportu końcowego	pakiet Office