



**WYŻSZA SZKOŁA BANKOWA**  
Szczecin | 10 lat razem

# Praca zaliczeniowa

## Studia podyplomowe - Tester oprogramowania

Raport z testów serwisu  
<https://sanibelsupply.eu/>

**Promotorzy:**

XXX  
XXX

**Autorzy projektu:**

*Agnieszka Bortnowska*  
XXX

## Spis treści

Spis treści .....	2
1. Wstęp .....	4
1.1 Przedmiot testów .....	4
1.2 Cel testów .....	5
1.3 Podejście do testowania .....	5
1.3.1 Ograniczenia .....	5
1.3.2 Funkcjonalności nietestowane .....	5
2. Zarządzanie projektem .....	6
2.1 Zasoby ludzkie .....	6
2.2 Potrzeby środowiskowe .....	6
2.2.1 Zasoby sprzętowe .....	6
2.2.2 Ranking Gemius .....	6
2.2.3 Komponenty testowe .....	7
2.3 Harmonogram prac .....	7
3. Opis testów .....	8
3.1 Plan testów .....	8
3.1.1 Priorytet testowanego modułu .....	8
3.1.2 Warunki wejścia oraz wyjścia podczas procesu testowego .....	9
3.2 Zakres testów .....	9
3.3 Typy testów w projekcie .....	10
3.3.1 Charakterystyka przypadku testowego .....	11
3.4 Definiowanie ważności defektów .....	13
3.5 Priorytet defektów .....	14
3.6 Bugtracker .....	14
3.6.1 Raporty defektów .....	15
3.7 Implementacja i wykonanie testów .....	16
3.7.1 Automatyzacja testów .....	17
3.7.1.1 PyCharm, Selenium Webdriver .....	17
3.7.1.2 Katalon Recorder .....	19
3.7.1.3 Katalon Studio .....	21
3.7.2 Podsumowanie testów .....	21
4. Wyniki testów .....	22
4.1 Kwestie niejasne .....	24

4.2 Raport błędów .....	25
5. Podsumowanie .....	27
5.1 Realizacja założeń.....	27
5.2 Rekomendacje ulepszeń.....	27
5.3 Ocena przedmiotu testów .....	28
6. Spis tabel i rycin.....	29
7. Załączniki.....	29

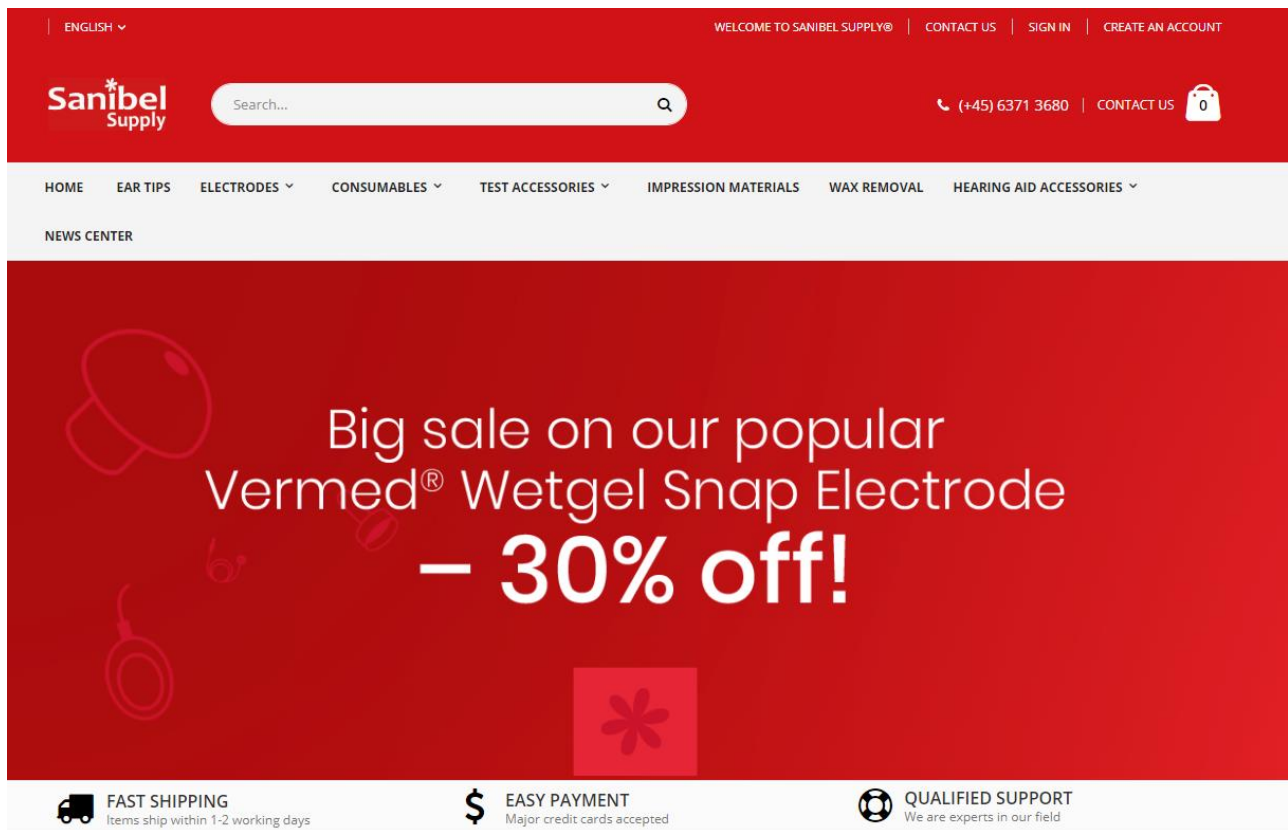
# 1. Wstęp

## 1.1 Przedmiot testów

Przedmiotem testów naszego projektu jest strona <https://sanibelsupply.eu/>. Firma ta specjalizuje się w zaopatrzeniu w sprzęt audiologiczny. Strona ma charakter sklepu internetowego, podzielona jest na 9 zakładek, w skład których wchodzi:

- Home
- Ear tips
- Electrodes
- Consumables
- Test accesories
- Impression material
- Wax removal
- Hearing aid accesories
- News center

W serwisie można dokonać rejestracji i założyć konto użytkownika. Po zalogowaniu się, możliwy jest zakup sprzętu audiologicznego, edycja danych, a także dalsze zarządzanie kontem.



Ryc. 1. Strona główna serwisu <https://sanibelsupply.eu/>.

Źródło: <https://sanibelsupply.eu/>

## 1.2 Cel testów

Celem projektu jest sprawdzenie poprawnego działania wybranych funkcjonalności serwisu, a także ocena postrzegania strony przez potencjalnych jej użytkowników.

## 1.3 Podejście do testowania

Projekt przedstawia proces testowy powierzony przez właścicieli serwisu <https://sanibelsupply.eu/> do zewnętrznego zespołu testerskiego. Powyższe zadanie zostało zlecone z powodu spadających przychodów firmy oraz małego zainteresowania sklepem przez potencjalnych klientów. Na podstawie analizy wyników z ruchu na stronie, właściciel firmy zauważył, że klienci podczas składania zamówień nie korzystali z opcji zakładania konta – co może sugerować problemy w tym obszarze strony. Dlatego też zleceniodawca oczekiwał pokrycia testami elementów serwisu takich jak: rejestracja, logowanie, panel użytkownika, koszyk, funkcjonalności strony, wishlista, checkout oraz strona główna „HOME”. Powyższe funkcjonalności miały zostać przetestowane pod kątem eksploracji przez przeciętnego użytkownika. Zlecenie nie narzucało określonej techniki testowania, testy odbywały się w środowisku produkcyjnym. Zespół testerski posiadał dowolność w wyborze sposobu wykonania testów, zarówno manualnych jak i automatycznych.

Ze zleceniodawcą została ustalona hierarchia, priorytet oraz zakres wykonywanych testów. Klient oczekiwał, że wszystkie przypadki testowe przedstawione przez zespół testerski będą stawnowiły 100% pokrycia testów dla modułów o wysokim i średnim priorytecie. Wedle założeń proces testerski miał trwać nie dłużej niż 5 miesięcy, a na zakończenie pracy miało nastąpić przesłanie wyników testów, raportu defektów oraz ewentualnych propozycji ulepszeń. W związku z brakiem dokumentacji technicznej, testy były przeprowadzane ad hoc. Testerzy opierali się głównie na doświadczeniach własnych oraz testach czarnoskrzynkowych.

### 1.3.1 Ograniczenia

Do głównych ograniczeń projektu możemy zaliczyć brak dokumentacji technicznej, specyfikacji serwisu oraz środowisko produkcyjne. Klient nie wymagał przetestowania modułu płatności – wykonane testy w tym module były prowadzone do kroków wyboru sposobu płatności, nie dalej. Z tego powodu historia zamówień użytkownika nie mogła zostać poddana testom. Kolejny wyjątek stanowiły obszary strony, gdzie wymagana jest interakcja klientów ze sklepem – dodawanie komentarzy, ocen itp..

### 1.3.2 Funkcjonalności nietestowane

Zleceniodawca zastrzegł sobie, aby testom nie zostały poddane sekcje strony takie jak:

- dodawanie komentarzy
- ocena produktów
- moduł płatności – od momentu podania danych wrażliwych
- historia zamówień

## 2. Zarządzanie projektem

### 2.1 Zasoby ludzkie

Zespół składa się z dwóch osób: Agnieszki Bortnowskiej – lider testów, tester oraz XXX – tester.

### 2.2 Potrzeby środowiskowe

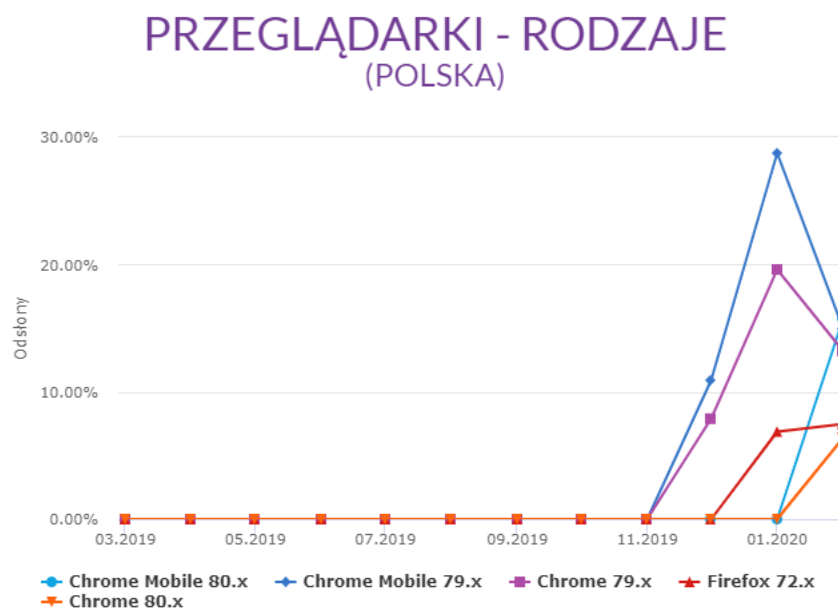
#### 2.2.1 Zasoby sprzętowe

Zespół testerski jest wyposażony w dwa komputery:

- Lenovo – system Windows 10 Home w wersji 1909, 64 -bitowy system operacyjny, procesor x64 oraz przeglądarka Google Chrome Wersja 79.0.3945.130 (Oficjalna wersja) (64-bitowa)
- Samsung – system Windows 10 Home w wersji 1909, 64 -bitowy system operacyjny, procesor x64 oraz przeglądarka Google Chrome Wersja 79.0.3945.130 (Oficjalna wersja) (64-bitowa)

#### 2.2.2 Ranking Gemius

Według rankingu Gemius najpopularniejszą przeglądarką w Polsce w miesiącu lutym 2020 jest przeglądarka Chrome w wersji 79.x (Ryc. 2), dlatego ta wersja została wykorzystana w niniejszym projekcie testowym.



**Ryc. 2. Najpopularniejsze przeglądarki w Polsce.**

Źródło: <http://ranking.gemius.com/pl/ranking/browsers/>

### 2.2.3 Komponenty testowe

Oprogramowanie użyte w procesie testowym:

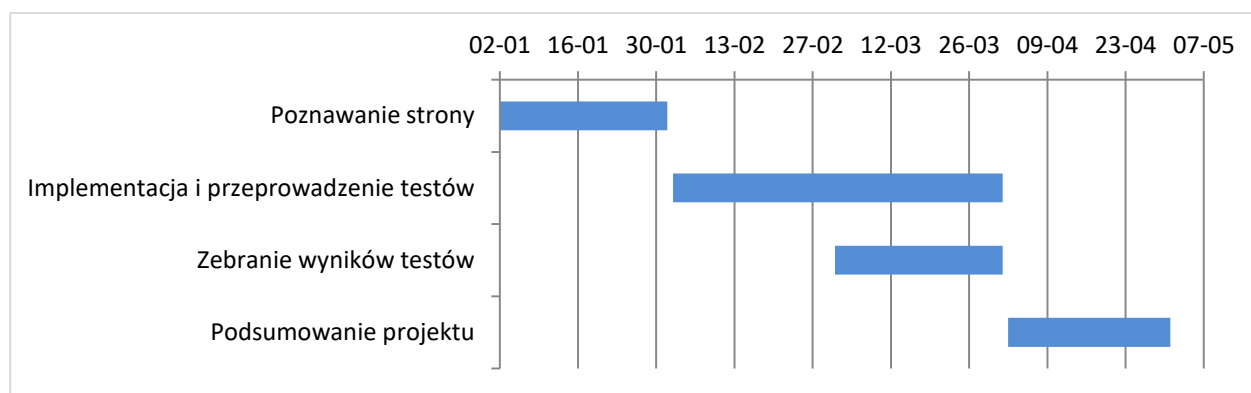
- Tworzenie i zarządzanie testami - Aplikacja Trello

Automatyzacja testów:

- Katalon Studio
- Katalon Recorder
- PyCharm
- Selenium Webdriver

### 2.3 Harmonogram prac

Harmonogram przeprowadzonych testów obejmował 5 miesięcy pracy. Szczegóły podziału pracy przedstawia poniższy wykres Gantta.



**Ryc. 3. Harmonogram prac**

Źródło: Opracowanie własne

## 3. Opis testów

### 3.1 Plan testów

Prace testerskie odbywały się na przełomie lutego oraz marca 2020 roku. Szczegółowy plan testów przedstawia tabela 1.

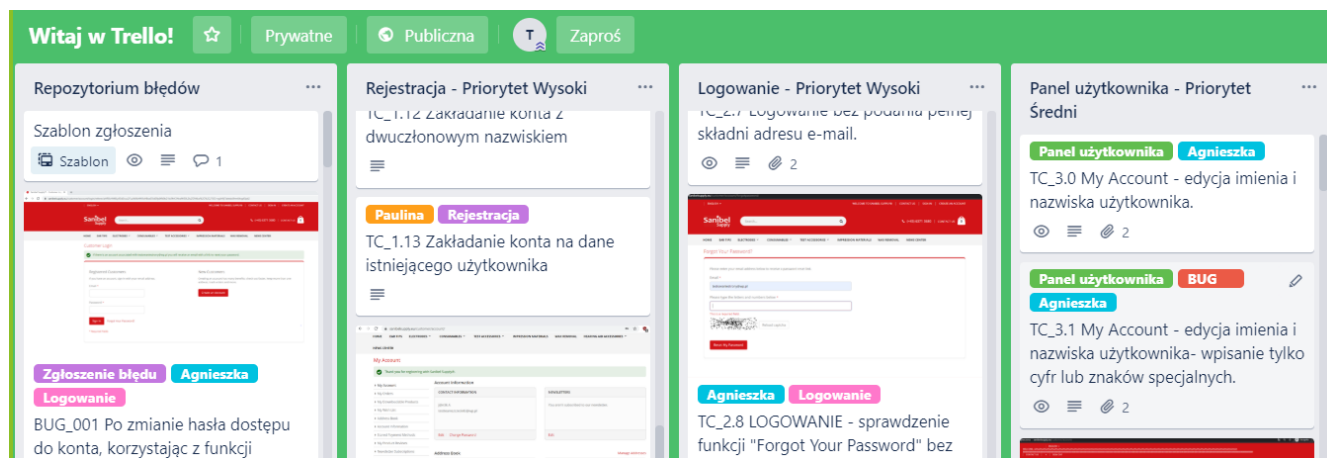
**Tabela 1. Plan testów.**

Moduł	Termin testów
Rejestracja	1 – 9.02.2020
Logowanie	23 – 29.02.2020
Panel użytkownika	9 – 15.02.2020
Koszyk	16 – 22.02.2020
Pozostałe funkcjonalności strony oraz zakładka home, wishlist, checkout	1 – 14.03.2020

Źródło: Opracowanie własne

#### 3.1.1 Priorytet testowanego modułu

Priorytet testowanego modułu określa kolejność wykonania testów w serwisie. Hierarchia ta została ustalona wraz ze zleceniodawcą na początku współpracy. Przypisanie priorytetu odbywało się w aplikacji Trello poprzez określenie go w nagłówku tablicy danego modułu, co przedstawia rycina 4. Charakterystyka każdego priorytetu została szczegółowo opisana w tabeli 2.



**Ryc. 4. Priorytet modułu.**

Źródło: Trello.com



**Tabela 2. Priorytet modułu.**

Priorytet	Kryterium
Wysoki	Moduł powinien zostać przetestowany w pierwszej kolejności.
Średni	Testy powinny nastąpić w drugiej kolejności.
Niski	Moduł nie wymaga szybkiego przetestowania.

Źródło: Opracowanie własne

### 3.1.2 Warunki wejścia oraz wyjścia podczas procesu testowego

Warunki wejścia do procesu testowego są definiowane poprzez:

- termin podpisania umowy zlecenia z Klientem
- plan testów oraz priorytet testowanego modułu
- harmonogram prac

Umowa z Klientem na przeprowadzenie projektu została podpisana 31.12.2019r i obejmowała rozpoczęcie prac testerskich od 1.01.2020r. Zgodnie z umową zleceniobiorca posiadał 5 miesięcy na przeprowadzenie swoich prac i zdanie projektu. Harmonogram prac został dokładnie przedstawiony w podrozdziale 2.4 niniejszej pracy.

Określone na wstępie procesu testowego warunki wyjścia oznaczały zakończenie prac. Na te warunki składały się:

- ramy czasowe projektu – od 1.01 do 31.05.2020
- budżet wynikający z umowy ze Zleceniodawcą
- 100% pokrycie testów dla modułów o wysokim i średnim priorytecie

### 3.2 Zakres testów

Testy obejmowały 8 modułów. Dokładny zakres testów prezentuje tabela 3.

**Tabela 3. Zakres testów.**

Moduł	ID przypadku testowego
Rejestracja	TC_1.0 – TC_1.18
Logowanie	TC_2.0 – TC_2.10
Panel użytkownika	TC_3.0 – TC_3.30
Koszyk	TC_4.0 – TC_4.5
Funkcjonalności strony	TC_5.0 – TC_5.7
Wishlist	TC_6.0 – TC_6.12
Checkout	TC_7.0 – TC_7.6
Strona główna „HOME”	TC_8.0 – TC_8.12

Źródło: Opracowanie własne

W modułach rejestracji nowego użytkownika oraz logowania skupiliśmy się głównie na testach negatywnych. Testy opierały się na zakładaniu nowego konta z niepoprawnymi danymi, moduł logowania został przetestowany pod kątem poprawnego oraz negatywnego zalogowania się do systemu itp.

Panel użytkownika to sprawdzenie poprawnego funkcjonowania zakładek dostępnych w module „my account” po założeniu konta i zalogowaniu się do serwisu.

Testy w module koszyk opierały się na dodawaniu i usuwaniu z niego produktów oraz akcji z tym związanych.

Pozostałe funkcjonalności związane były ze sprawdzeniem poprawnego działania dostępnych akcji na stronie takich jak: filtrowanie, sortowanie, wyszukiwanie produktów, a także dodawaniu ich do wishlist czy listy porównawczej.

Proces realizacji zamówień – checkout został przetestowany pod kątem możliwości dodawania adresów z książki adresowej oraz realizacji zamówień jako zalogowany i niezalogowany użytkownik. Funkcjonalność ta została przetestowana w niepełnym zakresie w związku z późniejszą koniecznością podania danych wrażliwych, co było ustalone ze Zleceniodawcą.

### 3.3 Typy testów w projekcie

Do głównych typów testów w projekcie możemy zaliczyć:

- Funkcjonalne – weryfikacja funkcji, jakie ma pełnić system. Można tu wyróżnić użyte techniki projektowania testów:
  - techniki czarnoskrzynkowe – testowanie przejść między stanami, tablice decyzyjne, testowanie w oparciu o przypadki użycia
  - techniki oparte na doświadczeniu oraz testowanie eksploracyjne
- Niefunkcjonalne
  - testy użyteczności – efektem testów będzie między innymi zgłoszenie propozycji ulepszeń

### 3.3.1 Charakterystyka przypadku testowego

Szczegółową charakterystykę przypadku testowego przedstawia tabela 4.

**Tabela 4. Charakterystyka przypadku testowego.**

Element przypadku testowego	Opis
Etykieta – Autor, Moduł oraz ewentualnie BUG	Skrótowa informacja nt. autora, testowanego modułu oraz ewentualnego znalezionej błędu
ID	Unikalna nazwa przypadku testowego
Tytuł	Skrótowy opis danego przypadku testowego
Środowisko	Dane dotyczące przeglądarki oraz początkowej strony testu
Warunek wstępny	Uwarunkowania środowiska, jakie muszą być spełnione przed przeprowadzeniem testu
Dane testowe	Dane potrzebne do przeprowadzenia testu (np. login, hasło)
Kroki reprodukcji	Opis poszczególnych kroków do wykonania podczas testu
Warunki końcowe	Efekt wykonania testu
Wykonanie	Określenie sposobu wykonania testu – manualne bądź automatyczne
Autor	Osoba wykonująca test

Źródło: Opracowanie własne

Opcjonalnie do przypadku testowego załączany jest zrzut ekranu.

Tworzone przypadki testowe charakteryzowały się niskim poziomem, dane testowe były dokładnie określone. Przykładowy przypadek testowy prezentuje rycina 5.

TC\_2.1 Logowanie z nieprawidłowym hasłem.

na liście Logowanie

ETykiETY

Agnieszka

Logowanie

+

Opis

Edytuj

ID TC\_2.1

Tytuł: Logowanie z nieprawidłowym hasłem.

Środowisko:

<https://www.sanibelsupply.eu/customer/account/login/referer/aHR0cHM6Ly93d3cuc2FuaWJlbHN1cHBseS5ldS8%2C/>

Przeglądarka Google Chrome Wersja 79.0.3945.130 (Oficjalna wersja) (64-bitowa)

Warunek wstępny: Niezalogowany użytkownik, posiadający aktywne konto w sklepie internetowym "[www.sanibelsupply.eu](http://www.sanibelsupply.eu)"

Dane testowe:

e-mail: [testowaniestrony@wp.pl](mailto:testowaniestrony@wp.pl)

hasło: abc12

Kroki reprodukcji:

1. Przejść do strony:

<https://www.sanibelsupply.eu/customer/account/login/referer/aHR0cHM6Ly93d3cuc2FuaWJlbHN1cHBseS5ldS9jdXN0b21lci9hY2NvdW50L2ZvcmdvdHBhc3N3b3JkLw%2C%2C/>

2. Wprowadzić adres e-mail w polu "Email"

3. Wprowadzić hasło w polu "Password"

4. Kliknąć przycisk "Sign In"

Warunki końcowe: Próba logowania z nieprawidłowym hasłem zakończyła się komunikatem "The account sign-in was incorrect or your account is disabled temporarily. Please wait and try again later". Użytkownik nie został zalogowany.

Wykonanie: Manualne. Załączony screenshot.

Autor: Agnieszka Bortnowska

DODAJ DO KARTY

Członkowie

Etykiety

Lista zadań

Termin

Załącznik

DODATKI

Butler Tips (15)

Uzyskaj więcej dodat...

DZIAŁANIA

Przenieś

Kopiuuj

Utwórz szablon

Obserwuj

Zarchiwizuj

Udostępnij

Załączniki

TC\_2.1.PNG

**Ryc. 5. Przykładowy przypadek testowy.**

Źródło: Trello.com

### 3.4 Definiowanie ważności defektów

Ważność zgłoszonego defektu mówi nam o poziomie dotkliwości systemu. Została stworzona czteropoziomowa skala oceny ważności defektu. Wyróżniłyśmy następujące poziomy krytyczności błędu:

- krytyczny
- wysoki
- średni
- niski

Charakterystykę każdego z nich przedstawia tabela 5.

**Tabela 5. Ważność defektów.**

Ważność defektu	Kryteria
Krytyczny	Do defektu krytycznego zaliczamy awarię systemu z ryzykiem utraty danych; system nie może zostać wydany.
Wysoki	Niezgodny z wymaganiami lub tylko częściowo zaimplementowany; obiekt testowy może być użyty z pewnymi restrykcjami.
Średni	Mniejsze defekty; obiekt testowy może być używany bez restrykcji.
Niski	Niekrytyczne defekty to te z nich, które wprowadzają pewne problemy w obsłudze, niezdarzające się zbyt często.

Źródło: Opracowanie własne

### 3.5 Priorytet defektów

Priorytet zgłoszonego defektu ustalany był na podstawie określonych kryteriów szybkości naprawy danego błędu. Dokładny podział priorytetów przedstawia tabela 6.

**Tabela 6. Priorytet defektów.**

Priorytet defektu	Kryteria
Wysoki	Defekt powinien zostać naprawiony w pierwszej kolejności.
Średni	Naprawa błędu powinna nastąpić w drugiej kolejności.
Niski	Błąd nie wymaga szybkiej naprawy.

Źródło: Opracowanie własne

### 3.6 Bugtracker

Do tworzenia i zarządzania testów została użyta aplikacja Trello. Każdy testowany moduł posiadał własną tablicę z przypisanymi przypadkami testowymi, które zawierały etykietę, pliki ze skryptami testów oraz zrzuty ekranu. Zostały utworzone następujące tablice:

- Repozytorium błędów
- Logowanie
- Rejestracja
- Panel użytkownika
- Koszyk
- Funkcjonalności strony
- Wishlist
- Checkout
- Strona główna „HOME”
- Propozycje ulepszeń

### 3.6.1 Raporty defektów

Raportowanie błędów odbywało się w serwisie Trello poprzez przypisanie etykiety „BUG” w przypadku testowym, który wykazywał defekt. W programie zostało również utworzone repozytorium błędów w postaci tablicy pt. „Repozytorium błędów”. Większość zgłoszeń zawiera zrzut ekranu błędu oraz takie informacje jak:

**Tabela 7. Charakterystyka zgłoszenia błędu.**

Element zgłoszenia	Opis
Tytuł zgłoszenia	Skrócony opis zgłoszenia błędu
ID zgłoszenia	Unikalna nazwa zgłoszenia
ID przypadku	Unikalna nazwa przypadku testowego, którego dotyczy zgłoszenie
Priorytet	Określa szybkość naprawy danego błędu
Poziom krytyczności błędu	Poziom dotkliwości systemu
Środowisko	Dane dotyczące przeglądarki oraz strony testowanego serwisu
Zgłaszający	Osoba, która wykryła defekt
Data	Czas zgłoszenia błędu
Warunki wstępne	Uwarunkowania środowiska, jakie muszą być spełnione do wystąpienia błędu
Kroki postępowania	Opis poszczególnych kroków do wywołania błędu
Oczekiwany rezultat	Oczekiwana odpowiedź przedmiotu testów
Efekt	Rzeczywista odpowiedź przedmiotu testów

Źródło: Opracowanie własne

Szablon zgłoszenia

na liście [Repozytorium błędów](#)

Opis

Edytuj

Tytuł zgłoszenia:

ID zgłoszenia:

ID przypadku:

Priorytet:

Ważność defektu:

Środowisko:

Zgłaszający:

Data:

Warunki wstępne:

Kroki postępowania:

Oczekiwany rezultat:

Efekt:

Aktywność

Pokaż Szczegóły

T

Napisz komentarz...

T

testteamszczecin 31 maj o 18:33

Priorytet: wysoki/średni/niski

Ważność defektu: krytyczny/wysoki/średni/niski

**Ryc. 6. Szablon zgłoszenia defektów**

Źródło: Trello.com

### 3.7 Implementacja i wykonanie testów

Testerzy po wstępnym zapoznaniu się z serwisem <https://sanibelsupply.eu/> podjęli przygotowanie środowiska testowego. Wykonanie testów odbywało się zgodnie z założonym harmonogramem, nad którym czuwał lider testów. Zadania zostały rozdysponowane przez lidera testów pomiędzy testerami – każdy tester otrzymał poszczególne moduły, za które był odpowiedzialny w procesie testowym.

Agnieszka Bortnowska testowała:

- moduł logowania
- panel użytkownika
- wishlist
- checkout
- strona główna „home”

XXX testowała:

- moduł rejestracji
- koszyk
- funkcjonalności strony



### 3.7.1 Automatyzacja testów

W poniższym podrozdziale zostały przedstawione programy wykorzystane do napisania i przeprowadzenia testów automatycznych. Testy automatyczne obejmowały następujące przypadki testowe:

Nazwa programu	ID przypadku testowego
PyCharm, Selenium Webdriver	TC_2.0, TC_2.3, TC_2.9, TC_3.0, TC_3.1, TC_3.18, TC_3.19, TC_3.21, TC_6.1
Katalon Recorder	TC_2.5, TC_2.7, TC_3.2, TC_3.14, TC_3.17, TC_3.26
Katalon Studio	TC_4.0 – 4.5, TC_5.0 – 5.7, TC_8.1, TC_8.2, TC_8.7 – 8.12

Źródło: Opracowanie własne

Całkowita liczba testów automatycznych wyniosła  $n=37$ , co stanowi 33,94% wykonanych łącznie testów. Powyższe przypadki testowe w większości wykazywały pewną powtarzalność, dlatego zostały zautomatyzowane, by w przyszłości oszczędzić czas przy przeprowadzaniu testów regresyjnych i innych testów związanych ze zmianami. Znaczna część testów posiada weryfikację „assert” czy „verify”. W testach gdzie nie było możliwości dodania wyżej wymienionych poleceń, weryfikacja polegała na sprawdzeniu zgodności zrzutu ekranu wykonanego testu ze specyfikacją przypadku testowego.

#### 3.7.1.1 PyCharm, Selenium Webdriver

Dla większej czytelności oraz ułatwienia tworzenia skryptów testowych napisanych w języku Python, a wykonanych w programie PyCharm wraz z zewnętrznym oprogramowaniem do sterowania przeglądarką – Selenium Webdriver, został zaprojektowany plik „Helper\_zalogowany.py”. W pliku tym są zawarte następujące akcje:

- otworenie strony logowania do serwisu sanibelsupply.eu
- akceptacja plików cookie
- zalogowanie do serwisu
- wylogowanie z serwisu
- zamknięcie okna przeglądarki

W kodzie stworzono dwie funkcje: logowania „def Login(driver)” oraz wylogowania „def Logout(driver)”. W celu potwierdzenia poprawnego wykonania powyższych akcji zostały umieszczone asercje sprawdzające adres url po zalogowaniu, a także po wylogowaniu. Aby skorzystać w kolejnym skrypcie z pliku „Helper\_zalogowany.py” należy użyć następującej komendy:

```
from Helper_zalogowany import Login
from Helper_zalogowany import Logout
```

Również plik „Helper...” musi znajdować się w tym samym folderze co plik, w którym będzie on wykorzystany.

```
from selenium.webdriver.common.keys import Keys
```

```

def Login(driver):

driver.get('https://www.sanibelsupply.eu/customer/account/login/referer/aHR0cHM6Ly93d3cuc2FuaWJ')

    cookie = driver.find_element_by_xpath('//*[@id="coiPage-1"]/div[2]/span[2]')
    cookie.click()

    email = driver.find_element_by_id('email')
    email.send_keys('testowaniestrony@wp.pl')

    haslo = driver.find_element_by_id('pass')
    haslo.send_keys('Tester123')

    signinButton = driver.find_element_by_id('send2')
    signinButton.send_keys(Keys.ENTER)

    urlafterlogin = driver.current_url
    assert urlafterlogin == 'https://www.sanibelsupply.eu/customer/account/'

def Logout(driver):
    signoutButton = driver.find_element_by_xpath('/html/body/div[2]/header/div[1]/div/ul/li[5]/')
    signoutButton.send_keys(Keys.ENTER)

    urlafterlogout = driver.current_url
    assert urlafterlogout == 'https://www.sanibelsupply.eu/customer/account/logoutSuccess/'

    driver.quit()

```

**Ryc. 7. Treść pliku „Helper\_zalogowany.py”**

Źródło: Opracowanie własne

Asercje zastosowane w skryptach testowych mają na celu sprawdzenie poprawnego działania danej funkcjonalności. Opierało się to na weryfikacji adresu url bądź czy komunikat wyświetlający się po zakończeniu konkretnej akcji odpowiada założonemu tekstowi.

W poniższym teście asercja sprawdzała czy po edycji imienia i nazwiska w zakładce „My Account” i kliknięciu przycisku „save”, użytkownik otrzyma komunikat mówiący o poprawnym zapisaniu danych.

```

from selenium import webdriver
from Helper_zalogowany import Login
from Helper_zalogowany import Logout

driver = webdriver.Chrome()
driver.maximize_window()

Login(driver)

actionSwitch = driver.find_element_by_xpath('/html/body/div[2]/header/div[1]/div/ul/li[3]/span/')
actionSwitch.click()

```

```

my_account = driver.find_element_by_xpath('/html/body/div[2]/header/div[1]/div/ul/li[3]/div/ul/
my_account.click()

editButton =
driver.find_element_by_xpath('//*[id="maincontent"]/div[3]/div[1]/div[2]/div[2]/div[1]/div[2]/
editButton.click()

first_name = driver.find_element_by_id('firstname')
first_name.click()
first_name.clear()
first_name.send_keys('Anna')

last_name = driver.find_element_by_id('lastname')
last_name.click()
last_name.clear()
last_name.send_keys('Kowalska')

saveButton = driver.find_element_by_xpath('//*[id="form-validate"]/div[2]/div[1]/button')
saveButton.click()

positiveMessage = driver.find_element_by_xpath('//*[id="maincontent"]/div[2]/div[2]/div/div/di
assert positiveMessage.text == 'You saved the account information.'

driver.get_screenshot_as_file('TC_3.0.png')

Logout(driver)

```

**Ryc. 8. TC\_3.0 My Account - edycja imienia i nazwiska użytkownika.**

Źródło: Opracowanie własne

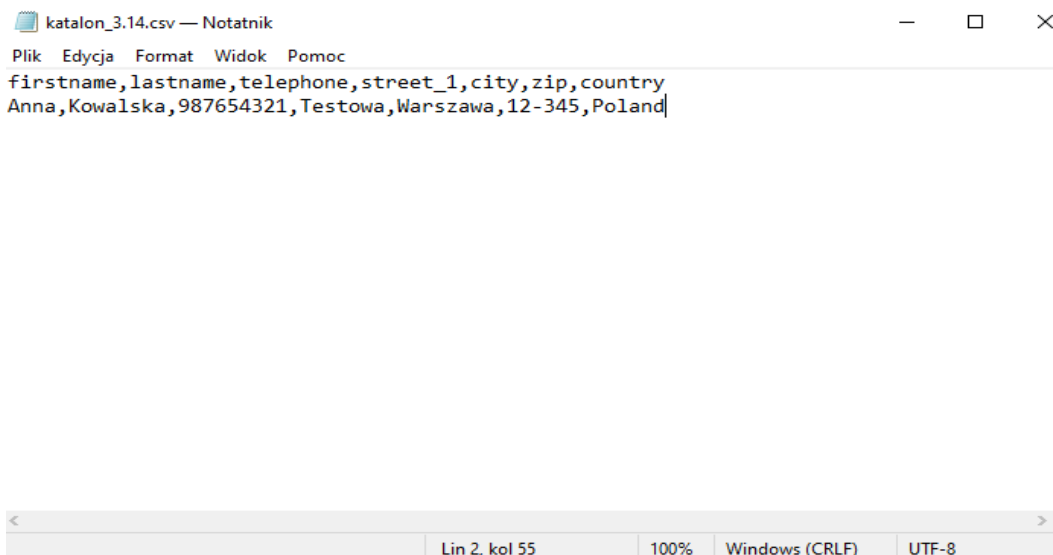
### 3.7.1.2 Katalon Recorder

Część testów wykonanych w programie Katalon Recorder, które opierały się na wypełnieniu formularza z danymi, korzystała z plików zewnętrznych zapisanych w formacie „.csv” co przedstawia rycina 9 oraz 10. Aby wczytać taki plik, należało przejść do zakładki „Data Driven”, następnie w tworzonym teście wpisać komendę „loadVars”, a w polu „Target” podać nazwę pliku. Na zakończenie testu wpisujemy komendę „endLoadVars” oraz nazwę pliku.

loadVars	TC_3.2.csv	
open	https://www.sanibelsupply.eu/customer/account/login/referer/aHR0cHM6Ly93d3cuc2FuaWJl bHN1cHBseS5idS8%2C/	
click	//*[@id="coiPage-1"]/div[2]/span[2]	
type	id=email	testowaniestrony@wp.pl
click	id=pass	
type	id=pass	Tester123
click	//button[@id='send2']/span	
click	//button[@type='button']	
click	link=My Account	
click	//main[@id='maincontent']/div[3]/div/div[2]/div[2]/div/div[2]/a/span	
click	id=firstname	
type	id=firstname	\${firstname}
click	id=lastname	
type	id=lastname	\${lastname}
click	//form[@id='form-validate']/div[2]/div/button/span	
verifyText	//*[@id='maincontent']/div[2]/div[2]/div/div	You saved the account information.
captureEntirePageScreenshot	TC_3.2.png	
click	xpath=(//a[contains(text(),'Sign Out')])[2]	
endLoadVars	TC_3.2.csv	
close		

**Ryc. 9. TC\_3.2 My Account - edycja imienia i nazwiska - sprawdzenie możliwości wpisania nieograniczonej ilości znaków.**

Źródło: Katalon Recorder



**Ryc. 10. Zewnętrzny plik „.csv”.**

Źródło: Opracowanie własne

### 3.7.1.3 Katalon Studio

Część testów została również przeprowadzona w oparciu o program Katalon Studio. Testy te polegały na wykorzystaniu zewnętrznego oprogramowania do sterowania przeglądarką i nagrania ich za pomocą Katalon Studio. W miejscu „Item” określamy komendę, „Object” określa nam akcję zaś w miejscu „Input” wartość jaka ma się znajdować w danym polu.

Item	Object	Input
→ 1 - Open Browser		""
→ 2 - Navigate To Url		"http://sanibelsupply.eu/"
→ 3 - Click	span_OK	
→ 4 - Click	span_Electrodes	
→ 5 - Click	span_OK	
→ 6 - Click	button_Add to Cart	
→ 7 - Click	button_Add to Cart	
→ 8 - Verify Element Text	div_You added Sanibel Infant EarCup Kit wo _77a8db	"You added Sanibel™ Infant EarCup
→ 9 - Close Browser		

**Ryc. 11. Przykładowy przypadek testowy z weryfikacją.**

Źródło: Katalon Studio

### 3.7.2 Podsumowanie testów

Jak wynika z analizy danych przedstawionych w tabeli 8, łączna liczba testów wyniosła n=109 największą grupę w projekcie stanowiły testy manualne n=72 (66,06%), testy automatyczne liczyły n=37 (33,94%).

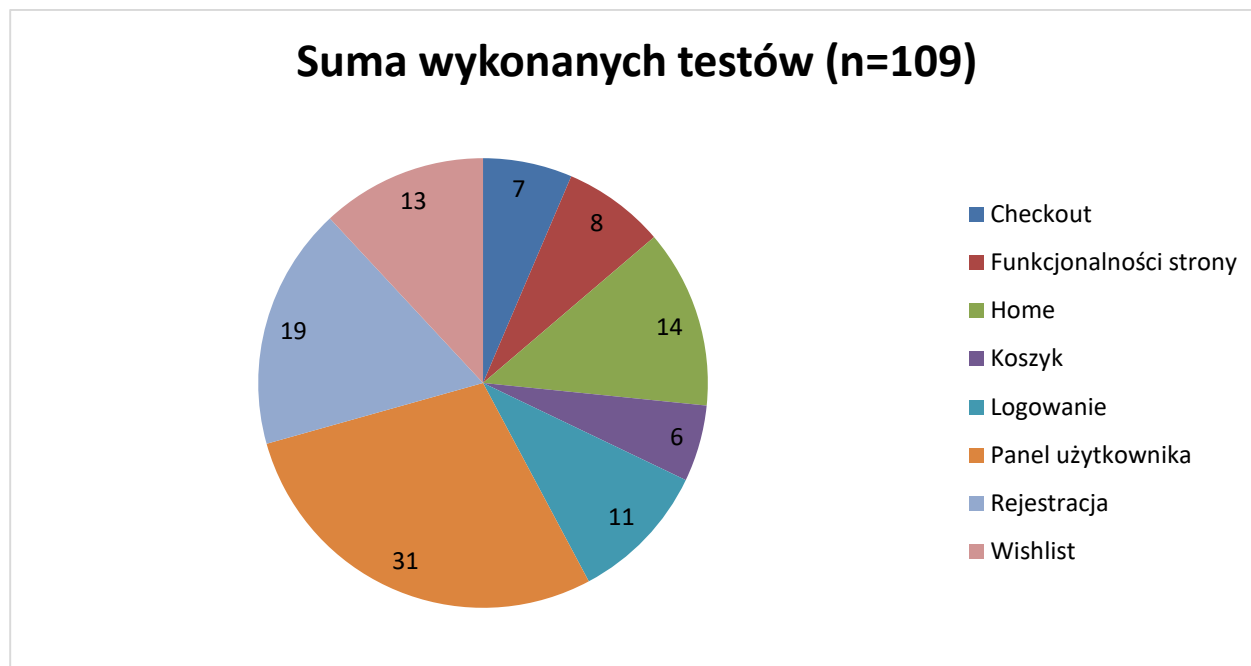
**Tabela 8. Podział testów (n=109).**

Rodzaj wykonanych testów	n	%
Manualne	72	66,06%
Automatyczne	37	33,94%
Łącznie	109	100,00%

Źródło: Opracowanie własne

## 4. Wyniki testów

Ta część pracy przedstawia wyniki projektu uzyskane podczas testów strony <https://sanibelsupply.eu/>, łącznie wykonano 109 testów. Analiza przeprowadzonych badań umożliwia ocenę funkcjonalności, a także weryfikację odpowiedniego wdrożenia serwisu.



**Ryc. 12. Suma wykonanych testów (n=109).**

Źródło: Opracowanie własne

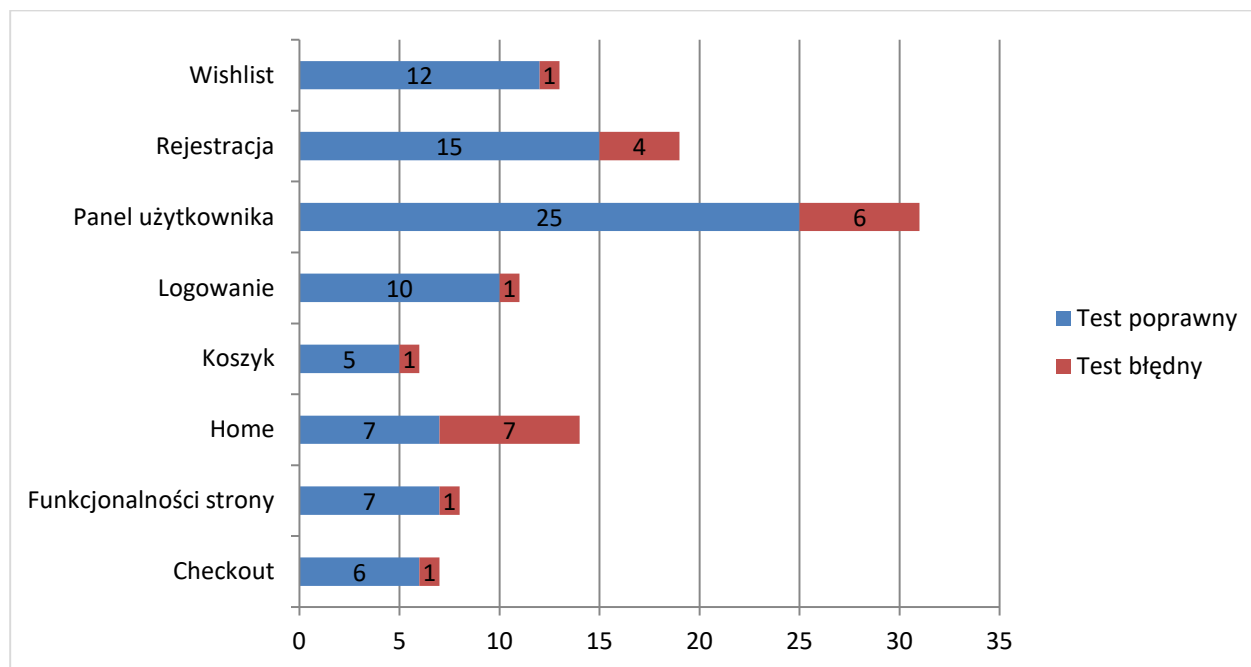
Jak wynika z analizy ryciny 12, najliczniejszą grupę testów w projekcie stanowią testy w panelu użytkownika  $n=31$  (28,44%), następnie testy w module rejestracji  $n=19$  (17,43%), w home, wishlist oraz w module logowania wykonano odpowiednio testów  $n=14$  (12,84%),  $n=13$  (11,93%) oraz  $n=11$  (10,09%). W funkcjonalnościach strony wykonano  $n=8$  (7,34%), checkout  $n=7$  (6,42%) testów, natomiast najmniej testów wykonano w module koszyk  $n=6$  (5,50%).

**Tabela 9. Podział testów ze względu na wynik (n=109).**

Wynik testu	Ilość	%
Pozytywny	87	79,82%
Negatywny	22	20,18%

Źródło: Opracowanie własne

Tabela 9 przedstawia podział ilości testów ze względu na uzyskany wynik. Na 109 przeprowadzonych testów w projekcie,  $n=87$  (79,82%) wypadło poprawnie, natomiast  $n=22$  (20,18%) wykazało negatywny rezultat testu.



**Ryc. 13. Podział testów ze względu na kategorię.**

Źródło: Opracowanie własne

W poszczególnych kategoriach wyniki testów kształtowały się następująco: wishlist udział testów poprawnych wynosi  $n=12$ , błędnych  $n=1$ . Moduł rejestracji to  $n=15$  poprawnych testów, błędne  $n=4$ , w panelu użytkownika aż  $n=25$  testów uzyskało wynik poprawny, błędny natomiast  $n=6$ . W panelu logowania poprawnie wypadło  $n=10$  testów, zaś  $n=1$  test błędny. W module koszyk poprawny wynik testu uzyskało  $n=5$ , błędny  $n=1$ , home to  $n=7$  testów poprawnych, a  $n=7$  błędnych. Funkcjonalności strony w testach wypadły  $n=7$  poprawnie,  $n=1$  błędnie, checkout  $n=6$  poprawnych testów, a jedynie  $n=1$  błędny.

## 4.1 Kwestie niejasne

Testy porównania opcji „wyszukiwania” oraz „wyszukiwania zaawansowanego” przebiegły w dość niejasny sposób. Wyniki testów są niejednoznaczne co przedstawia tabela 10.

*Tabela 10. Wyniki testów „wyszukiwania”.*

Kryteria wyszukiwania	Search	Advanced search
Wpisanie frazy nieprzypisanej do żadnego produktu ("szukam")	"Your search returned no results."	"We can't find any items matching these search criteria. Modify your search."
Wklejenie pełnej nazwy produktu "Ambu® Blue Sensor™ P-00-S, bag/50 pcs"	Proces wyszukiwania zakończył się wyświetleniem produktów z różnych grup. Szukany produkt nie zostanie wyświetlony na pierwszym miejscu.	Produkt nie został znaleziony.
Wklejenie pełnej nazwy produktu "Sanibel™ disposable prewired electrodes 50cm, 300 pcs"	-	Szukany produkt został wyświetlony.
Wyszukanie produktu podając tylko jeden człon nazwy produktu ("ambu")	Proces wyszukiwania zakończył się wyświetleniem produktów posiadających w nazwie szukaną frazę.	Proces wyszukiwania zakończył się wyświetleniem produktów posiadających w nazwie szukaną frazę.
Wyszukanie produktu podając nazwę, numer przedmiotu, opis, "widelki cenowe" (opcja tylko dla advanced search)	-	Produkt został poprawnie wyszukany.

Źródło: Opracowanie własne

**Przypadek 1** – wyszukiwanie dowolnego, niezwiązanego z żadnym produktem słowa – w obu przypadkach proces przebiegł prawidłowo. Żaden produkt nie został wyszukany.

**Przypadek 2** – rezultat wyszukiwania produktu o nazwie "Ambu® Blue Sensor™ P-00-S, bag/50 pcs" w „Search” zakończył się wyświetleniem produktów z różnych grup, szukany produkt nie został wyświetlony na pierwszym miejscu. Natomiast w „Advanced search” produkt nie został w ogóle znaleziony.

**Przypadek 3** – wyszukiwanie frazy "Sanibel™ disposable prewired electrodes 50cm, 300 pcs" w „Advanced search” przebiegło pomyślnie.

**Przypadek 4** – w obu opcjach wyszukiwania po wpisaniu frazy „ambu” rezultat był taki sam, zostały poprawnie wyświetlone produkty spełniające powyższe kryterium.

**Przypadek 5** – tylko wykonany w opcji w „Advanced search”, ze względu na rozbudowane opcje wyszukiwania.



### **Podsumowanie**

Powyższa analiza wyników wyszukiwania wskazuje na niepoprawne funkcjonowanie opcji „Advanced search”. Możemy to zauważyć analizując przypadek 2, w którym po wklejeniu pełnej nazwy szukanego produktu, opcja wyszukiwania zaawansowanego nie wyświetla szukanego rezultatu. Natomiast rozpatrując przypadek 3 - wklejenie pełnej nazwy innego produktu w „Advanced search” zakończyło się poprawnym wynikiem.

## **4.2 Raport błędów**

W niniejszym podrozdziale przedstawiono wyniki z raportu defektów. Łączna liczba wykrytych błędów wynosi  $n=22$ .



**Ryc. 14. Podział błędów ze względu na kategorię (n=22).**

Źródło: Opracowanie własne

**Tabela 11. Podział błędów ze względu na kategorię (n=22).**

<b>Kategoria</b>	<b>n</b>	<b>%</b>
Home	7	31,82%
Panel użytkownika	6	27,27%
Checkout	4	18,18%
Koszyk	1	4,55%
Rejestracja	1	4,55%
Logowanie	1	4,55%
Funkcjonalności strony	1	4,55%
Wishlist	1	4,55%

Źródło: Opracowanie własne

Tabela 11 przedstawia podział defektów ze względu na kategorię, w której wystąpiły. Wyniki kształtują się następująco: n=7 (31,82%) błędów w module Home, n=6 (27,27%) w panelu użytkownika, w module checkout n=4 (18,18%) oraz po n=1 (4,55%) w pozostałych modułach serwisu.

**Tabela 12. Podział defektów ze względu na poziom krytyczności błędu (n=22).**

<b>Poziom błędu</b>	<b>n</b>	<b>%</b>
Krytyczny	1	4,55%
Wysoki	9	40,91%
Średni	2	9,09%
Niski	10	45,45%

Źródło: Opracowanie własne

Jak wynika z danych w tabeli 12 najliczniejszą grupę defektów stanowiły błędy na poziomie niskim n=10 (45,45%) oraz wysokim n=9 (40,91%). Najmniej wykazano błędów na poziomie średnim n=2 (9,09%) i krytycznym n=1 (4,55%).

## 5. Podsumowanie

### 5.1 Realizacja założeń

Realizacja projektu przebiegła zgodnie z harmonogramem. W wyniku procesu testowego uzyskano 100% pokrycie zaplanowanego zakresu testów. W tym na 109 przypadków testowych  $n=87$  (79,82%) wypadło poprawnie, natomiast  $n=22$  (20,18%) wykazało negatywny rezultat testu. Zgłoszono  $n=22$  defekty. Jak wynika z danych w tabeli 12 najliczniejszą grupę defektów stanowiły błędy o poziomie niskim  $n=10$  (45,45%) oraz wysokim  $n=9$  (40,91%) co przekłada się na ponad 85% wykrytych defektów. Najmniej wykazano błędów na poziomie średnim  $n=2$  (9,09%) i krytycznym  $n=1$  (4,55%). Zaproponowano 6 propozycji ulepszeń serwisu.

### 5.2 Rekomendacje ulepszeń

Listę rekomendowanych zmian w serwisie przedstawia tabela 13.

**Tabela 13. Propozycja ulepszeń**

Propozycja ulepszeń	Opis
PU_001	Propozycja zmiany przycisku „akcji” – widoczne po zalogowaniu do serwisu
PU_002	Propozycja zmiany przycisku zamknięcia podglądu zdjęcia
PU_003	Propozycja dodania opisu i ikony „serca” oraz do ikony „wykresu”, które nawigują do „listy życzeń” oraz do „listy porównawczej”
PU_004	Propozycja dodania przycisku „usuń wszystko” w koszyku
PU_005	Propozycja dodania odnośnika/zakładki kierującej do listy porównawczej
PU_006	Propozycja zmiany wyświetlania sumarycznej ilości produktów w koszyku

Źródło: Opracowanie własne

### 5.3 Ocena przedmiotu testów

Po przeprowadzonym procesie testowym, serwis <https://sanibelsupply.eu/> wykazuje błędy w wybranych przez Zleceniodawcę modułach strony. Defekty te, mogą mieć wpływ na odbiór serwisu przez potencjalnych klientów. Niemalże 60% znalezionych błędów znajduje się na stronie głównej „HOME” oraz w panelu użytkownika, co może być przyczyną zmniejszonego zainteresowania serwisem, a także niechęcią do zakładania konta użytkownika w sklepie.

W celu poprawy jakości świadczonych usług, rekomendujemy naprawę znalezionych usterek, szczególnie na poziomie krytycznym oraz wysokim, a także po ich naprawie wykonanie testów potwierdzających oraz regresywnych. Zaproponowane ulepszenia mogą korzystnie wpłynąć na odbiór serwisu przez użytkowników, co przełoży się na zwiększony ruch na stronie, a także może znacząco poprawić zainteresowanie firmą.

## 6. Spis tabel i rycin

Ryc. 1. Strona główna serwisu <a href="https://sanibelsupply.eu/">https://sanibelsupply.eu/</a> .....	4
Ryc. 2. Najpopularniejsze przeglądarki w Polsce. ....	6
Ryc. 3. Harmonogram prac .....	7
Tabela 1. Plan testów.....	8
Ryc. 4. Priorytet modułu. ....	8
Tabela 2. Priorytet modułu.....	9
Tabela 3. Zakres testów.....	9
Tabela 4. Charakterystyka przypadku testowego.....	11
Ryc. 5. Przykładowy przypadek testowy. ....	12
Tabela 5. Ważność defektów.....	13
Tabela 6. Priorytet defektów.....	14
Tabela 7. Charakterystyka zgłoszenia błędu.....	15
Ryc. 6. Szablon zgłoszenia defektów.....	16
Ryc. 7. Treść pliku „Helper_zalogowany.py” .....	18
Ryc. 8. TC_3.0 My Account - edycja imienia i nazwiska użytkownika.....	19
Ryc. 9. TC_3.2 My Account - edycja imienia i nazwiska - sprawdzenie możliwości wpisania nieograniczonej ilości znaków.....	20
Ryc. 10. Zewnętrzny plik „.csv”.....	20
Ryc. 11. Przykładowy przypadek testowy z weryfikacją.....	21
Tabela 8. Podział testów (n=109).....	21
Ryc. 12. Suma wykonanych testów (n=109).....	22
Tabela 9. Podział testów ze względu na wynik (n=109).....	22
Ryc. 13. Podział testów ze względu na kategorię.....	23
Tabela 10. Wyniki testów „wyszukiwania”.....	24
Ryc. 14. Podział błędów ze względu na kategorię (n=22).....	25
Tabela 11. Podział błędów ze względu na kategorię (n=22).....	26
Tabela 12. Podział defektów ze względu na poziom krytyczności błędu (n=22).....	26
Tabela 13. Propozycja ulepszeń.....	27

## 7. Załączniki