**SKLEP INTERNETOWY ☺**

**Słownik**

* First name - minimum jeden znak
* Last name - minimum jeden znak
* Email - <minimum jeden znak> @ <minimum jeden znak>
* Password - minimum 8 znaków w tym wielka litera, mała litera i cyfra
* Country – 1-20 liter
* City – 1-20 liter
* Street – 1-20 liter
* ZIPCode – <cyfra><cyfra>-<cyfra><cyfra><cyfra>
* Building number – 1-4 cyfry
* Apartament number – 1-4 cyfry
* Letter of complain – minimum 10 znaków

**Funkcjonalności**

1. Rejestracja użytkownika. Wymagany email (w bazie danych nie ma konta użytkownika z podanym email’em), imię, nazwisko, hasło, powtórzenie hasła (podstrona ‘Sign in/Register’).
2. Logowanie. Wymagany email i hasło, które odpowiadają użytkownikowi zapisanemu w bazie danych (podstrona ‘Sign in/Register’).
3. Zakup. Wymagane wcześniejsze logowanie, podanie adresu lub wybranie adresu z listy adresów użytkownika, płatność (podstrona ‘Cart’).
4. Reklamacja (podstrona ‘My account’, zakładka ‘Orders’). Wymagane logowanie.
5. Filtrowanie oferty ze względu na typ produktu i zakres ceny.
6. Zapis/rezygnacja do/z newsletter’a (podstrona ‘My account’, zakładka ‘Newsletter’). Wymagane logowanie.
7. Dodanie/usunięcie adresu do/z listy adresów(podstrona ‘My account, zakładka ‘Address book’)’. Wymaganie logowanie.
8. Edycja danych użytkownika tj. imię, nazwisko (podstrona ‘My account’, zakładka ‘Profile’). Wymagane logowanie.

**Uwierzytelnianie**

**Scenariusze**

**REJESTRACJA**

|  |  |
| --- | --- |
| **Aktorzy:** | Użytkownik |
| **Zakres:** | Sklep internetowy |
| **Poziom:** | Sytemowowy ? |
| **Udziałowcy i ich cele:** | **Użytkownik** chce stworzyć konto. **Sklep** chce zebrać dane spełniające zadane warunki. |
| **Zdarzenie wyzwalające:** | Użytkownik przechodzi na podstronę ‘Sign in/Register’ |
| **Warunki wstępne:** | brak |
| **Warunki końcowe dla sukcesu:** | Konto użytkownika zostaje utworzone i zapisane w bazie danych. Użytkownik zostaje powiadomiony o sukcesie |
| **Warunki końcowe dla niepowodzenia:** | Konto nie zostaje stworzony. Użytkownik zostaje powiadomiony o niepowodzeniu i przyczynach niepowodzenia. |

|  |
| --- |
| **Scenariusz główny:** |

1. System wyświetla formularz wprowadzania danych rejestracji

2. Użytkownik wprowadza dane (zdefiniowane w słowniku): imię, nazwisko, email, hasło

4. System weryfikuje dane

5. System wyświetla powiadomienie o udanej rejestracji

|  |
| --- |
| **Scenariusz alternatywny:** |

4.a. Nie wprowadzono wymaganych danych

4.a.1 System wyświetla ponownie formularz zaznaczając, które dane powinny zostać poprawione i jakie warunki powinny spełniać zaznaczone dane

4.a.2 Następuje powrót do punktu 2 scenariusza głównego

|  |
| --- |
| **Scenariusz alternatywny:** |

4.a. W bazie danych znajduję się konto dla podanego email’a

4.a.1 System wyświetla ponownie formularz zaznaczając email i wyświetla informację o istnieniu konta dla podanego email’a

4.a.2 Następuje powrót do punktu 2 scenariusza głównego

**LOGOWANIE**

|  |  |
| --- | --- |
| **Aktorzy:** | Użytkownik |
| **Zakres:** | Sklep internetowy |
| **Poziom:** | Sytemowowy ? |
| **Udziałowcy i ich cele:** | **Użytkownik** chce się zalogować. **Sklep** chce zebrać dane spełniające zadane warunki. |
| **Zdarzenie wyzwalające:** | Użytkownik przechodzi na podstronę ‘Sign in/Register’ |
| **Warunki wstępne:** | Brak |
| **Warunki końcowe dla sukcesu:** | Użytkownik zostaje zalogowany. Następuje przekierownie na podstronę ’Shop’ |
| **Warunki końcowe dla niepowodzenia:** | Użytkownik nie zostaje zalogowany. Użytkownik zostaje powiadomiony o niepowodzeniu. |

|  |
| --- |
| **Scenariusz główny:** |

1. System wyświetla formularz wprowadzania danych logowania

2. Użytkownik wprowadza dane: email, hasło

4. System weryfikuje czy w bazie danych istnieje konto, które odpowiada wprowadzonym danym

5.System przekierowuje na podstronę ‘Shop’.

6. System wyświetla ‘My account’ na pasku nawigacji

7. System zamienia opcję przekierowania na postronę ‘Sign in/Register’ na opcję wylogowania użytkownika ‘Sign out’

8. Użytkownik przechodzi na podstronę ‘My account’

9. Użytkownik wybiera opcję ‘Sign out’

10. System przekierowuje na podstonę ‘Shop’

11. System usuwa ‘My account’ z paska nawigacji

|  |
| --- |
| **Scenariusz alternatywny:** |

4.a. Nie istnieje konto dla wprowadzonych danych

4.a.1 System wyświetla informację o niepowodzeniu

4.a.2 Następuje powrót do punktu 2 scenariusza głównego

|  |
| --- |
| **Scenariusz alternatywny:** |

7.a. Następuje przejście do punktu 9 scenariusza głównego

9.a Następuje przejście do punktu 11 scenariusza głównego

**Scenariusze do testów**

*LOGIN*

1. Click ‘Sign in/Register’
2. Redirect to ‘Sign in/Register’ page
3. Enter invalid data in ‘Sign in’ form
4. Click ‘Sign in’
5. See failure message
6. Enter valid datain ‘Sign in’ form
7. Click ‘Sign in’
8. Redirect to ‘Shop’ page
9. ‘My account’ is displayed
10. ‘Sign out’ appear instead of ‘Sign in/Register’
11. Click ‘Sign out’
12. ‘My account’ disapear
13. ‘Sign in/Register’ appear instead of ‘Sign out’

*LOGIN 2*

1. Click ‘Sign in/Register’
2. Redirect to ‘Sign in/Register’ page
3. Enter valid data in ‘Sign in’ form
4. Click ‘Sign in’
5. Redirect to ‘Shop’ page
6. ‘My account’ is displayed
7. ‘Sign out’ appear instead of ‘Sign in/Register’
8. Click ‘My account’
9. Click ‘Sign out’
10. Redirect to ‘Shop’ page
11. ‘My account’ disapear
12. ‘Sign in/Register’ appear instead of ‘Sign out’

*REGISTER*

1. Click ‘Sign in/Register’
2. Redirect to ‘Sign in/Register’ page
3. Enter nothing in ‘Register’ form
4. Click ‘Register’
5. See error messages
6. Enter valid data but taken email in ‘Register’ form
7. See email taken error
8. Enter available email
9. Click ‘Register’
10. See successful message

BAZA DANYCH

Tabela „Products”

- productId: integer (PK)

- name: string[[1]](#footnote-2)

- price: integer[[2]](#footnote-3)

- imagePath: string

- type: string

- description: string

Tabela “Clients”

* clientId: integer (PK)
* email: string[[3]](#footnote-4)
* firstName: string
* lastName: string
* password: string
* newsletter: boolean

Tabela “Orders”

* orderId: integer (PK)
* clientId: integer (FK)
* workerId: integer (FK)
* addressed: integer (FK)
* status: integer[[4]](#footnote-5)
* opendate: date
* closedate: date

Tabela “Order Entries”

* orderId: integer (FK)
* productId: integer (FK)
* amount: integer

Tabela “Addresses”

* addressId: integer (PK)
* clientId: integer (FK)
* country: string
* city: string
* street: string
* ZIPCode: integer
* buildingNumber: integer
* apartmentNumber: integer

Tabela “Payments”

* paymentId: integer (PK)
* orderId: integer (FK)
* amount: integer[[5]](#footnote-6)
* date: date

Tabela “Complaints”

* orderId: integer (FK)
* workerId: integer (FK)
* description: string
* date: date
* isClosed: boolean[[6]](#footnote-7)

Tabela “Workers”

* workerId: integer (PK)
* role: enum[[7]](#footnote-8)
* email: string
* password: string

DODAWANIE PRODUKTÓW DO KOSZYKA (Z POZIOMU PODSTRONY PRODUKTU)

|  |  |
| --- | --- |
| **Aktorzy:** | Użytkownik |
| **Zakres:** | Sklep internetowy |
| **Poziom:** | Systemowy |
| **Udziałowcy i ich cele:** | **Użytkownik** chce dodawać produkty do koszyka. **Sklep** chce zebrać i zapisać informacje o zawartości koszyka. |
| **Zdarzenie wyzwalające (trigger):** | Użytkownik przechodzi na podstronę produktu |
| **Warunki wstępne:** | Brak |
| **Warunki końcowe dla sukcesu:** | dodanie produktów do koszyka |
| **Warunki końcowe dla niepowodzenia:** | Informacja o niepowodzeniu |

|  |
| --- |
| **Scenariusz główny:** |

1. System wyświetla podstronę produktu

2. Użytkownik wprowadza ilość produktów (domyślna wartość=1)

4. System weryfikuje dane (ilość>0, 0-ignoruj)

5. Użytkownik wybiera opcję ‘Add to Cart’

6. System zmienia stan koszyka

|  |
| --- |
| **Scenariusz alternatywny:** |

4.a. Ilość nie jest liczbą naturalną

4.a.1 System wyświetla ponownie formularz zaznaczając, które dane powinny zostać poprawione

4.a.2 Następuje powrót do punktu 2 scenariusza głównego

komicik

ZMIANA ZAWARTOŚCI KOSZYKA (Z POZIOMU KOSZYKA)

|  |  |
| --- | --- |
| **Aktorzy:** | Użytkownik |
| **Zakres:** | Sklep internetowy |
| **Poziom:** | Sytemowowy |
| **Udziałowcy i ich cele:** | **Użytkownik** chce zmienić zawartość koszyka. **Sklep** chce zaktualizować informacje o zawartości koszyka. |
| **Zdarzenie wyzwalające (trigger):** | Użytkownik przechodzi na podstronę koszyka. |
| **Warunki wstępne:** | Brak |
| **Warunki końcowe dla sukcesu:** | aktualizacja stanu koszyka |
| **Warunki końcowe dla niepowodzenia:** | Informacja o niepowodzeniu |

|  |
| --- |
| **Scenariusz główny:** |

1. System wyświetla podstronę koszyka

3. Użytkownik zmienia ilość produktów

4. System weryfikuje dane

5. System aktualizuje stan koszyka

|  |
| --- |
| **Scenariusz alternatywny:** |

4.a.Ilość nie jest liczbą naturalną

4.a.1 System wyświetla ponownie formularz zaznaczając, które dane powinny zostać poprawione

4.a.2 Następuje powrót do punktu 2 scenariusza głównego

Request’y do serwera:

GetClientCredentials : pobierz imię, nazwisko, adresy (flaga przy adresie domyślnym tzn. tym który został użyty przy ostatnim zamówieniu i jednocześnie jest zapisany w książce adresów) zalogowanego klienta

GetOrders : pobierz wszystkie zamówienia zalogowanego klienta

IsEmailAvailable : sprawdzenie czy dany email jest zapisany w bazie danych

CreateNewClient : dodaj klienta do bazy danych

IsSingInCredentialCorrect : sprawdzenie czy w bazie danych jest zapisany klient o podanym email’u i haśle

IsClientSignInForNewsletter : sprawdzenie czy zalogowany klient powinien otrzymywać newsletter

UpdateClientCredentials : zaktualizuj dane zalogowanego klienta

AddNewAddress : stwórz nowy adres i przypisz do zalogowanego klienta

RemoveAddress : usuń wybrany adres

SaveOrder : zapisz zamówienie

CreateComplain : stwórz reklamacje

**REKLAMACJA**

|  |  |
| --- | --- |
| **Aktorzy:** | Użytkownik |
| **Zakres:** | Sklep internetowy |
| **Poziom:** | Sytemowowy |
| **Udziałowcy i ich cele:** | **Użytkownik** chce dokonać reklamacji. **Sklep** chce zebrać dane spełniające zadane warunki. |
| **Zdarzenie wyzwalające:** | Użytkownik przechodzi na podstronę ‘My account’ |
| **Warunki wstępne:** | Użytkownik jest zalogowany |
| **Warunki końcowe dla sukcesu:** | Reklamacja zostaje zapisana w bazie danych. Użytkownik zostaje powiadomiony o sukcesie |
| **Warunki końcowe dla niepowodzenia:** | Użytkownik zostaje powiadomiony o niepowodzeniu i przyczynach niepowodzenia. |

|  |
| --- |
| **Scenariusz główny:** |

1. Użytkownik przechodzi na zakładkę ‘Orders’
2. Użytkownik wybiera opcję ‘Complain’
3. System otwiera podstronę ‘Complain’
4. Użytkownik wprowadza dane w polu ‘Letter of complain’
5. System weryfikuje dane (czy wprowadzono minimum 10 znaków)
6. System wyświetla powiadomienie o udanej reklamacji

|  |
| --- |
| **Scenariusz alternatywny:** |

5.a. Nie wprowadzono wymaganych danych

5a.1 System wyświetla ponownie formularz zaznaczając ile znaków należy wprowadzić

5.a.2 Następuje powrót do punktu 4 scenariusza głównego

**NEWSLETTER**

|  |  |
| --- | --- |
| **Aktorzy:** | Użytkownik |
| **Zakres:** | Sklep internetowy |
| **Poziom:** | Sytemowowy |
| **Udziałowcy i ich cele:** | **Użytkownik** chce zapisać się/wypisać z newsletter’a. **Sklep** chce zebrać informacje o zapisie do newsletter’a |
| **Zdarzenie wyzwalające:** | Użytkownik przechodzi na podstronę ‘My account’ |
| **Warunki wstępne:** | Użytkownik jest zalogowany |
| **Warunki końcowe dla sukcesu:** | Deklaracja dotycząca newsletter’ zostaje zapisana w bazie danych |

|  |
| --- |
| **Scenariusz główny:** |

1. Użytkownik przechodzi na zakładkę ‘Newsletter’
2. Użytkownik wybiera opcję zapisu do newsletter’a / rezygnacji z newsletter’a
3. System aktualizuje informacje na stronie i w bazie danych

1. „name” jest wykorzystywane do tworzenia ścieżki dla podstrony produktu [↑](#footnote-ref-2)
2. Cena w centach (¢) [↑](#footnote-ref-3)
3. „email” unikatowy w bazie danych [↑](#footnote-ref-4)
4. Enum: 0 – otwarte, 1 – zakończone [↑](#footnote-ref-5)
5. Kwota wcentach (¢) [↑](#footnote-ref-6)
6. False – reklamacja otwarta, true – reklamacja zamknięta [↑](#footnote-ref-7)
7. ADMIN | SUPPORT [↑](#footnote-ref-8)