Dokumentacja Projektu Spring Boot: MojSklep

**Widoki**

* home.html: Strona główna z listą produktów dostępnych w katalogu.
* register.html: Formularz rejestracji nowego użytkownika.
* login.html: Formularz logowania do systemu.
* koszyk.html: Podgląd koszyka z możliwością usuwania produktów.
* myOrders.html: Historia zamówień zalogowanego użytkownika.
* admin-orders.html: Podgląd wszystkich zamówień w systemie – tylko dla admina.

# Serwisy aplikacji

Aplikacja korzysta z czterech głównych serwisów odpowiedzialnych za różne aspekty logiki biznesowej:

* ProductService: Zarządza katalogiem produktów. Przechowuje dane w mapie w pamięci.
* UserService: Obsługuje rejestrację i logowanie użytkowników. Dane zapisuje w pliku 'users.csv'.
* CartService: Zarządza koszykiem użytkownika i zapisuje zamówienia do 'orders.csv'.
* OrderService: Odczytuje zamówienia z pliku 'orders.csv' i przekształca je w obiekty Order.

# Schemat działania aplikacji (krok po kroku)

1. 1. Użytkownik wchodzi na stronę główną (home.html) i widzi produkty.
2. 2. Klikając 'Dodaj do koszyka', produkt zostaje dodany do sesji przez CartService.
3. 3. Użytkownik może się zarejestrować (register.html), dane trafiają do UserService i pliku users.csv.
4. 4. Po zalogowaniu (login.html), dane sesji są ustawiane.
5. 5. Przechodząc do koszyka (koszyk.html), użytkownik widzi zawartość z sesji.
6. 6. Po kliknięciu 'Zamów', zamówienie zapisuje się do pliku orders.csv przez CartService.
7. 7. Użytkownik może zobaczyć swoje zamówienia w myOrders.html (CartService / OrderService).
8. 8. Admin (np. 'admin') może podejrzeć wszystkie zamówienia w admin-orders.html.

# Schemat tekstowy aplikacji

+----------------------+  
 | home.html |  
 +----------+-----------+  
 |  
 +----------v-----------+  
 | ProductService | ---> Katalog produktów  
 +----------------------+  
  
 +-------------+ +-------------+ +---------------------+  
 | login.html | -------> | UserService | <------ | register.html |  
 +-------------+ +-------------+ +---------------------+  
 | |  
 Sesja logowania Zapis do users.csv  
  
 +--------------+ +----------------+  
 | koszyk.html | <------ | CartService |  
 +------+-------+ +----------------+  
 | |  
 Dodawanie produktów Zapis zamówienia  
  
 +-------------+ +-------------------+  
 | myOrders.html | <------ | OrderService / CartService  
 +-------------+ +-------------------+  
 |  
 +-----------v------------+  
 | orders.csv (historia) |  
 +------------------------+  
  
 +-----------------+  
 | admin-orders.html |  
 +-----------------+  
 |  
 Widzi wszystkie zamówienia

**Wykorzystane funkcje w wszystkich plikach**

"AuthController":

@PostMapping("/login")  
Logowanie użytkownika. Sprawdza poprawność loginu i hasła. Po poprawnej autoryzacji zapisuje nazwę użytkownika w sesji i przekierowuje na stronę główną. W przypadku błędu zwraca login.html z komunikatem.

@PostMapping("/register")  
Rejestracja nowego użytkownika. Sprawdza, czy użytkownik już istnieje. Jeśli nie – zapisuje go i wyświetla komunikat o sukcesie.

("loginPage()", "Zwraca widok strony logowania."),

("login()", "Obsługuje przesyłanie formularza logowania i autoryzuje użytkownika."),

("logout()", "Wylogowuje użytkownika i czyści dane z sesji."),

("registerPage()", "Zwraca widok strony rejestracji."),

("register()", "Rejestruje nowego użytkownika, jeśli nie istnieje."),

"HomeController": [

("home()", "Zwraca stronę główną z listą produktów oraz zawartością koszyka."),

"OrderController": [

@PostMapping("/order/confirm")  
 Obsługuje potwierdzenie zamówienia z koszyka. Zapisuje dane zamówienia do pliku orders.csv, czyści koszyk i przekierowuje na stronę główną.

("showCart()", "Wyświetla koszyk użytkownika."),

("removeFromCart()", "Usuwa produkt z koszyka według indeksu."),

("confirmOrder()", "Potwierdza i zapisuje zamówienie do pliku."),

("myOrders()", "Wyświetla historię zamówień zalogowanego użytkownika."),

("adminOrders()", "Wyświetla wszystkie zamówienia w widoku administratora."),

"AdminController": [

("adminOrders()", "Zwraca widok z listą wszystkich zamówień dla administratora."),

"UserService": [

("registerUser()", "Dodaje nowego użytkownika do mapy i zapisuje do pliku CSV."),

("getAllUsers()", "Zwraca listę wszystkich zarejestrowanych użytkowników."),

("getUserByUsername()", "Wyszukuje użytkownika po loginie."),

"ProductService": [

("getAllProducts()", "Zwraca katalog wszystkich produktów jako mapa."),

("getProductById()", "Zwraca produkt na podstawie jego ID."),

"CartService": [

("addToCart()", "Dodaje produkt do koszyka użytkownika (z sesji)."),

("removeFromCart()", "Usuwa produkt z koszyka wg indeksu."),

("getCart()", "Zwraca bieżącą zawartość koszyka."),

("getCartSize()", "Zwraca liczbę produktów w koszyku."),

("saveUserOrder()", "Zapisuje zamówienie w pamięci aplikacji."),

("saveOrderToFile()", "Zapisuje zamówienie do pliku CSV."),

("readAllOrders()", "Odczytuje wszystkie zamówienia z pliku CSV."),

("getOrdersFromFileForUser()", "Odczytuje zamówienia danego użytkownika z pliku CSV.")

"OrderService": [

("getAllOrdersDetailed()", "Odczytuje wszystkie zamówienia wraz ze szczegółami z pliku CSV."),

("getOrdersForUser()", "Zwraca zamówienia tylko dla wskazanego użytkownika."),

"Model – User": [

("getId(), setId()", "Dostęp do identyfikatora użytkownika."),

("getUsername(), setUsername()", "Dostęp do nazwy użytkownika."),

("getPassword(), setPassword()", "Dostęp do hasła."),

"Model – Product": [

("getName(), setName()", "Nazwa produktu."),

("getImageUrl(), setImageUrl()", "Link do zdjęcia produktu."),

("getDescription(), setDescription()", "Opis produktu."),

("getPrice(), setPrice()", "Cena produktu."),

"Model – Order": [

("getUsername(), setUsername()", "Dane użytkownika składającego zamówienie."),

("getAddress(), setAddress()", "Adres dostawy."),

("getPaymentMethod(), setPaymentMethod()", "Metoda płatności."),

("getDate(), setDate()", "Data zamówienia."),

("getProducts(), setProducts()", "Lista zamówionych produktów."),

("addProduct()", "Dodaje produkt do zamówienia."),