

# **Systemy Zarządzania Treścią**

## **Sylwia Mieszkowska 150371**

### **2020/21**

#### **1. Wstęp**

W ramach projektu przygotowano prostą aplikację webową, umożliwiającą użytkownikowi dodawanie, edycję, usuwanie oraz wyświetlanie produktów i kategorii.

#### **2. Wykorzystane narzędzia i technologie**

Aplikacja stworzona została w języku Java w wersji 8 z użyciem narzędzia Spring Boot. Spring Boot posiada własny wbudowany serwer aplikacyjny Tomcat. Do zarządzania warstwą danych wykorzystano framework Hibernate, umożliwiający translację danych w postaci obiektowej do rekordów w bazie danych.

Warstwa prezentacji została oparta o silnik szablonów Thymeleaf. Skorzystano także z dodatkowych narzędzi ułatwiających pracę nad widokiem: Bootstrap, jQuery oraz wtyczki oparte o jQuery – Multiselect oraz DataTables.

Do przechowywania danych wykorzystano bazę danych MySQL.

#### **3. Założenia aplikacji**

Aplikacja podzielona została na dwie części: dostępną dla wszystkich użytkowników oraz panel administratora. Zwykły użytkownik ma możliwość wyświetlania produktów w postaci listy oraz po naciśnięciu w odpowiedni przycisk, wyświetlanie szczegółów wybranego produktu. Ponadto ma możliwość filtrowania produktów wg. wybranych kategorii oraz wysyłania wiadomości w formularzu kontaktowym. Administrator ma możliwość dodawania, usuwania oraz edycji produktów oraz kategorii.

Wyróżniono następujące założenia funkcjonalne:

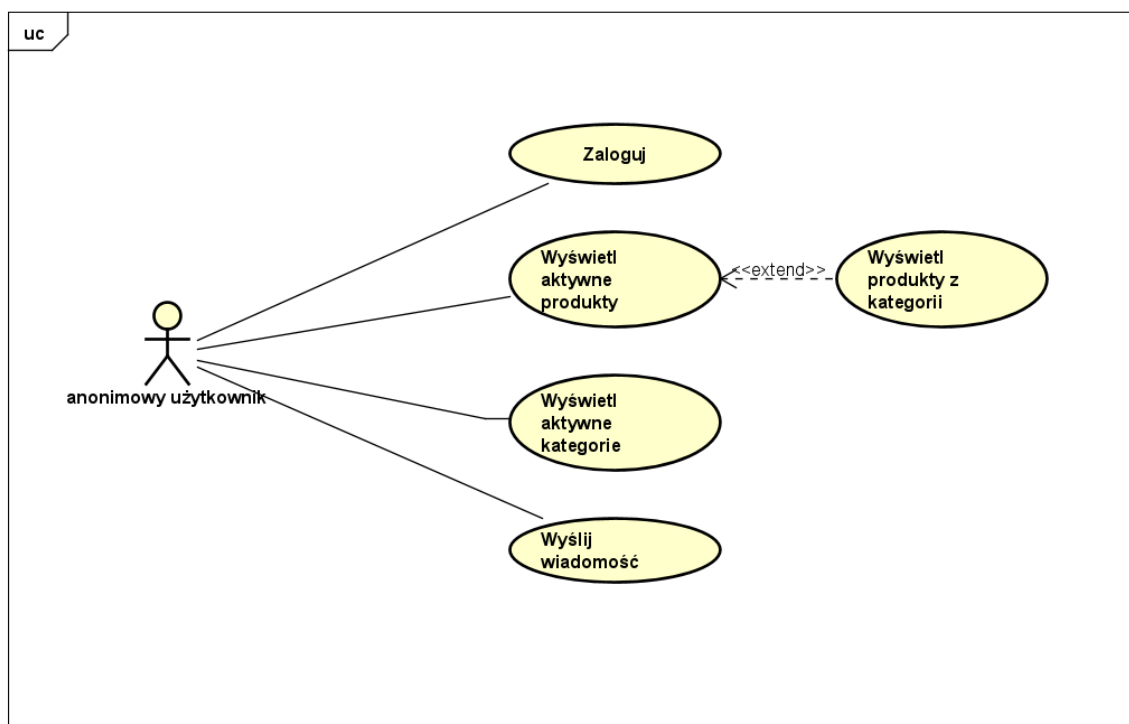
- a) Administrator:
  - dodawanie produktów,
  - edycja produktów,

- usuwanie produktów,
- dodawanie kategorii,
- edycja kategorii,
- usuwanie kategorii,
- wyświetlanie produktów,
- wyświetlanie kategorii,
- wylogowanie użytkownika.

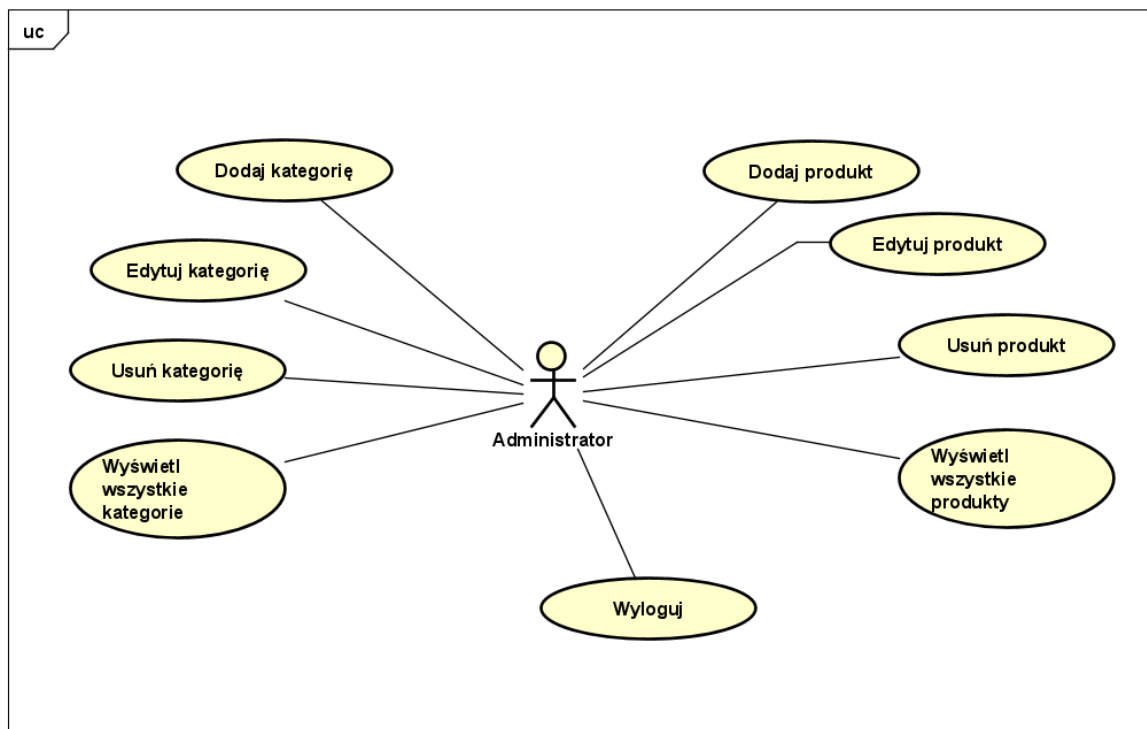
b) Anonimowy użytkownik:

- logowanie użytkownika,
- wyświetlanie aktywnych produktów,
- wyświetlanie aktywnych kategorii,
- filtrowanie aktywnych produktów wg. kategorii,
- wysyłanie wiadomości w formularzu kontaktowym.

Na rysunkach 1. i 2. przedstawiono diagramy przypadków użycia dla użytkownika anonimowego oraz administratora.



Rys 1. Diagram przypadków użycia dla użytkownika anonimowego



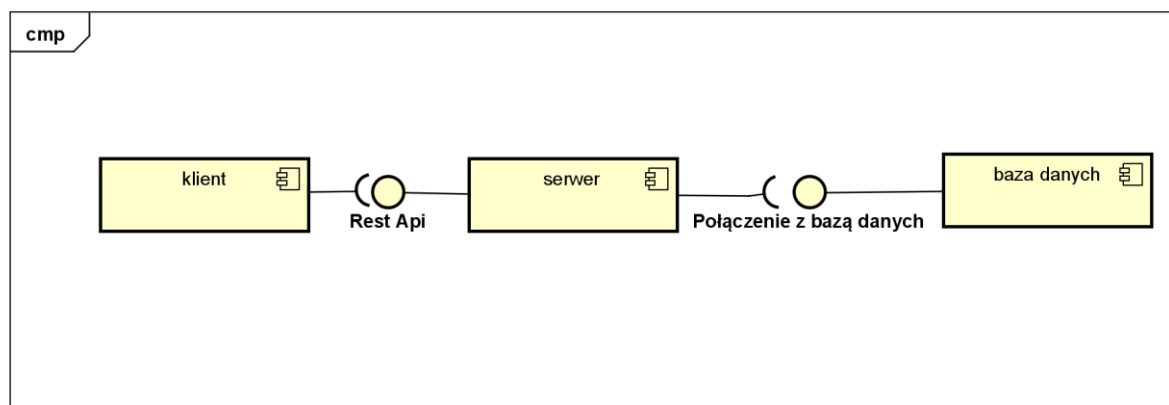
Rys 2. Diagram przypadków użycia dla administratora

#### 4. Architektura systemu

System złożony jest z trzech komponentów:

- aplikacja-serwer (pakiet „ecommerce-server”) – udostępniająca REST API,
- aplikacja-klient (pakiet „web”) – przetwarzająca dane odebrane od serwera i generująca warstwę widoku,
- baza danych.

Na rysunku 3. Przedstawiono diagram komponentów systemu.



Rys 3. Diagram komponentów systemu

W ramach warstwy danych utworzono kilka encji:

- category – przechowuje informacje o dostępnych kategoriach;
- product\_category – tabela „wiele do wielu”. Przechowuje informacje o kategoriach przypisanych do produktu;
- image – przechowuje informacje o zdjęciach;
- product\_image – tabela „wiele do wielu”. Przechowuje informacje o zdjęciach przypisanych do produktu;
- product – przechowuje informacje o utworzonych produktach;
- role – przechowuje dostępne w systemie role użytkowników (domyślnie występuje jedna rola – „ADMIN”),
- user – przechowuje informacje o zarejestrowanych użytkownikach.

Na rysunku 4. Przedstawiono diagram związków encji.



Rys 4. Diagram związków encji w systemie.

## 5. RestApi

Aplikacja-serwer udostępnia REST API. W tabeli 1. zestawiono wszystkie punkty dostępowe udostępniane przez aplikację.

Aby uzyskać dostęp do endpoint'ów przeznaczonych dla administratora (URL rozpoczynające się na „/admin”) należy najpierw uzyskać autoryzację wykonując zapytanie na endpoint „/login” zgodnie ze szczegółami przedstawionymi w tabeli 1. Po poprawnej autoryzacji w nagłówku „Authorization” zwracany jest token, który należy wykorzystać, aby uzyskać dostęp do chronionych endpoint'ów. Należy każdorazowo dodać do zapytania nagłówek „Authorization” z wartością „Bearer <token>”.

Tabela 1. Zestawienie wszystkich punktów dostępowych udostępnianych przez aplikację

Adres	Metoda HTTP	Ciało żądania	Wymagane nagłówki	Parametry URL	Zawracana wartość	Opis
/common/categories	GET	-	-	-	Lista kategorii	Zwraca listę aktywnych kategorii
/common/products	GET	-	-	category={id} (opcjonalny)	Lista produktów; jeśli użyty parametr „category” - lista produktów z kategorii o danym identyfikatorze, np.: [ { "id": 16, "version": 0, "name": "Polska", "active": true } ]	Zwraca listę aktywnych produktów; Zwraca listę aktywnych produktów z kategorii o podanym identyfikatorze
/common/products/{id}	GET	-	-	-	Produkt o podanym identyfikatorze, np.: { "id": 17, "version": 0, "name": "Tajlandia Marzeń", "sku": "T-1", "price": 5000.00, "quantity": 20, "shortDescription": "Lorem ipsum "description": "Aliquam eu congue felis "createDate": "2021-01-13T20:39:45.000+00:00", "active": true, "creator": { }, "images": [], "categories": [ { "id": 16, "version": 0, "name": "Tajlandia", "active": true } ], }	Zwraca produkt o podanym identyfikatorze
/common/contact/send	POST	{ "email": "adres email nadawcy", "message": "treść wiadomości" }	-	-	-	Wysła wiadomość na adres email podany w konfiguracji aplikacji
/login	POST	{ "login": "login użytkownika", "password": "hasło użytkownika" }	-	-	Login uwierzytelnionego użytkownika; w nagłówku Authorization wygenerowany token JWT	Uwierzytelnia użytkownika o podanym loginie i hasle i zwraca token autoryzacyjny w przypadku powodzenia
/login/me	GET	-	Authorization Bearer<token>	-	Login uwierzytelnionego użytkownika	Zwraca login uwierzytelnionego użytkownika
/admin/categories	GET	-	Authorization Bearer<token>	-	Lista wszystkich kategorii, np.: [ {	Zwraca listę wszystkich kategorii

					<pre>         "id": 16,         "version": 0,         "name": "Polska",         "active": true       }     ]   } </pre>	
/admin/categories	POST	<pre> {   "name": "nazwa kategorii" } </pre>	Authorization Bearer<token>	-	-	Tworzy nową kategorię o podanej nazwie
/admin/categories/{id}	GET	-	Authorization Bearer<token>	-	Kategoria o podanym identyfikatorze, np.: <pre> {   "id": 16,   "version": 0,   "name": "Tajlandia",   "active": true } </pre>	Zwraca kategorię o podanym identyfikatorze
/admin/categories/{id}	DELETE	-	Authorization Bearer<token>	-	-	Usuwa kategorię o podanym identyfikatorze
/admin/categories/{id}/activate	POST	-	Authorization Bearer<token>	-	-	Zmienia status „active” kategorii na przeciwny.
/admin/products	GET	-	Authorization Bearer<token>	-	Lista wszystkich produktów, np.: <pre> [   {     "id": 17,     "version": 0,     "name": "Tajlandia Marzeń",     "sku": "T-1",     "price": 5000.00,     "quantity": 20,     "shortDescription": "Lorem ipsum",     "description": "Aliquam eu congue felis",     "createDate": "2021-01-13T20:39:45.000+00:00",     "active": true,     "creator": {     },     "images": [],     "categories": [       {         "id": 16,         "version": 0,         "name": "Tajlandia",         "active": true       }     ]   },   ... ] </pre>	Zwraca listę wszystkich produktów
/admin/products	POST	<pre> {   "productsDto": {     "name": "nazwa produktu",     "shortDescription": "krótki opis produktu",   } } </pre>	Authorization Bearer<token>	-	-	Tworzy nowy produkt

		<pre> "description": "opis produktu", "cena": cena produktu (double), "qty": ilość produktu (int), "sku": "numer SKU produktu", "categories": [   "id" (lista id kategorii przypisanych do produktu) ], "user":{   "id": "id użytkownika przypisanego do produktu" } } "files": [   {"data": "zdjęcie w postaci base64",   "name": "nazwa pliku"} ] } </pre>				
/admin/products/{id}	GET	-	Authorization Bearer<token>	-	<p>Produkt o podanym identyfikatorze, np.:</p> <pre> {   "id": 17,   "version": 0,   "name": "Tajlandia Marzeń",   "sku": "T-1",   "price": 5000.00,   "quantity": 20,   "shortDescription": "Lorem ipsum   "description": "Aliquam eu congue felis   "createDate": "2021-01- 13T20:39:45.000+00:00",   "active": true,   "creator": {   },   "images": [],   "categories": [     {       "id": 16,       "version": 0,       "name": "Tajlandia",       "active": true     }   ], } </pre>	Zwraca produkt o podanym identyfikatorze
/admin/products/{id}	DELETE	-	Authorization Bearer<token>	-	-	Usuwa produkt o podanym identyfikatorze
/admin/products/{id}/activate	POST	-	Authorization Bearer<token>	-	-	Zmienia status „active” produktu o podanym identyfikatorze
/admin/images/decode	POST	[{"data": tablica bajtów}]	Authorization Bearer<token>	-	Lista łańcuchów znaków zawierających zdjęcia w postaci base64	Zamienia zdjęcie z postaci binarnej na postać base64
/admin/images/{id}	DELETE	-	Authorization Bearer<token>	-	-	Usuwa zdjęcie o podanym identyfikatorze



## 6. Uruchomienie aplikacji

Aby uruchomić aplikację niezbędne jest zainstalowanie Java JDK w wersji 1.8 oraz narzędzia Maven. Konieczne jest również uruchomienie serwera bazodanowego MySQL.

Po zainstalowaniu narzędzi należy odpowiednio skonfigurować parametry w plikach „application.yml”, znajdujące się w pakietach „ecommerce-server” oraz „web”. Należy skonfigurować połączenie z bazą danych podając odpowiedni adres połączenia JDBC we właściwości „spring.datasource.url”, a następnie podać nazwę użytkownika oraz hasło do bazy danych we właściwościach „spring.datasource.username” oraz „spring.datasource.password”.

W przypadku uruchamiania aplikacji w środowisku innym niż lokalne, należy skonfigurować numery portów we właściwości „server.port”, a w przypadku aplikacji „web” również adres aplikacji „ecommerce-server”, tak aby aplikacja odpowiedzialna za warstwę prezentacji mogła wykonywać zapytania do aplikacji–serwera (właściwości „app.service”).

Domyślnie aplikacja wysyła wiadomości przesłane w formularzu kontaktowym na adres email [szt.sylwia.mieszkowska@gmail.com](mailto:szt.sylwia.mieszkowska@gmail.com) (hasło SZTEcommerce). Można skonfigurować aplikację w taki sposób, aby aplikacja wysyłała wiadomości na inny, wybrany adres, modyfikując właściwości „contact.email”.

### 6.1. Wersja demonstracyjna

W celu ułatwienia przetestowania działania możliwości aplikacji, została wdrożona jej wersja demonstracyjna, która jest dostępna pod adresem:

<http://web-env.eba-ksdrwpae.eu-central-1.elasticbeanstalk.com>

Dostęp do REST API można uzyskać pod adresem:

<http://server-env.eba-v3gtm6ab.eu-central-1.elasticbeanstalk.com>

Aby przejść do panelu administratora należy przejść na adres

<http://web-env.eba-ksdrwpae.eu-central-1.elasticbeanstalk.com/login>

Dane logowania:

Login: sylwia.mieszkowska97@gmail.com

Hasło: Test1