



Dokumentacja Projektu

MICHAŁ SAŁEK | GRUPA: L/23 N LAB5/2/PROG | INDEKS: 13809

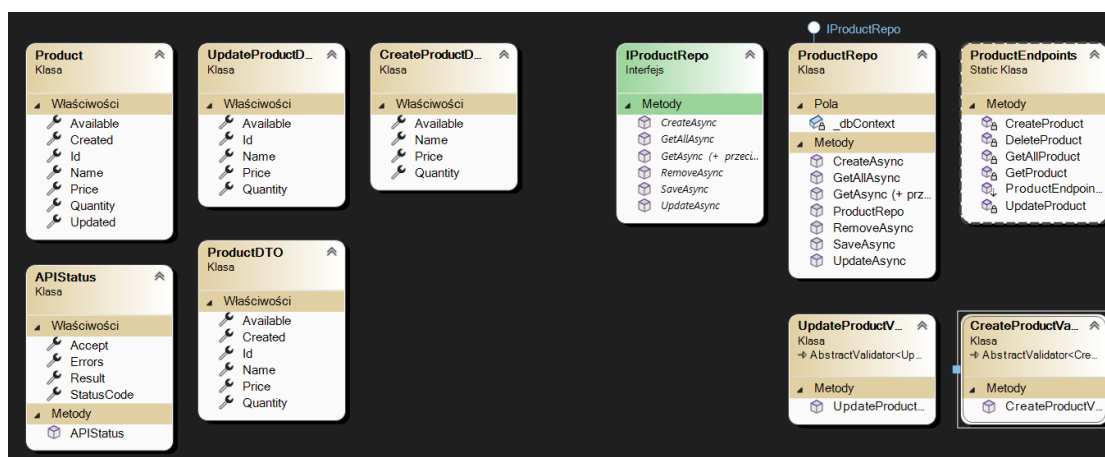
Opis

Założeniem aplikacji jest zaliczenie przedmiotu. Projekt jest to prosta aplikacja, umożliwiająca rejestrowanie użytkowników, logowania i tworzenie, odczytywanie, aktualizowanie i usuwanie elementów w bazie danych.

Rejestracja i logowanie odbywa się w bezpieczny sposób z udziałem autoryzacji oraz dostępem do ról, "admin", "customer"

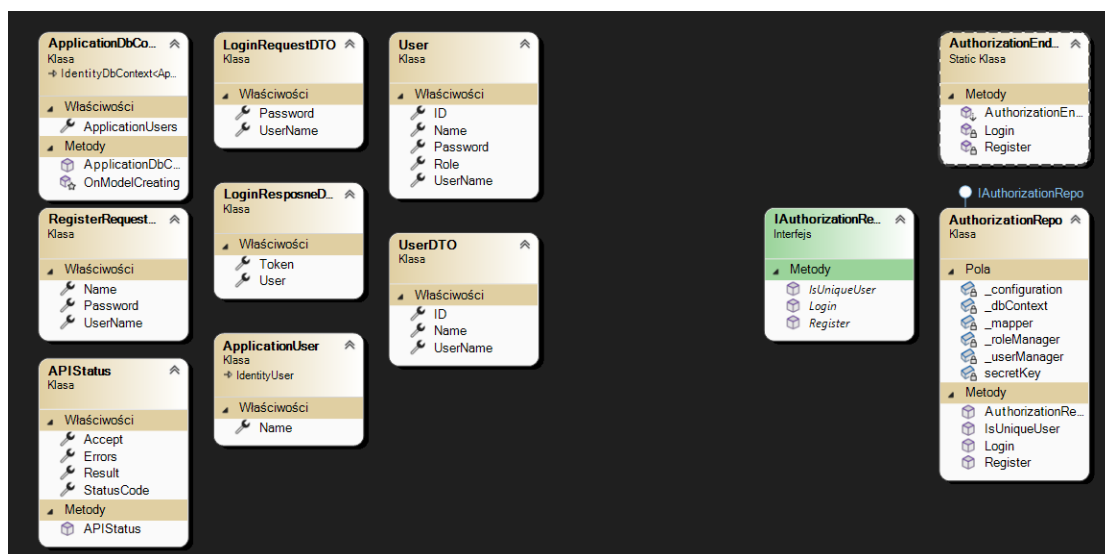
Klasy modelu serwisu ProjectAPI

Po lewej klasy obiekt transferu danych. Po prawej klasy metod, walidacji i interfejsu, klasy ProjectAPI odpowiadającej defakto za produkty



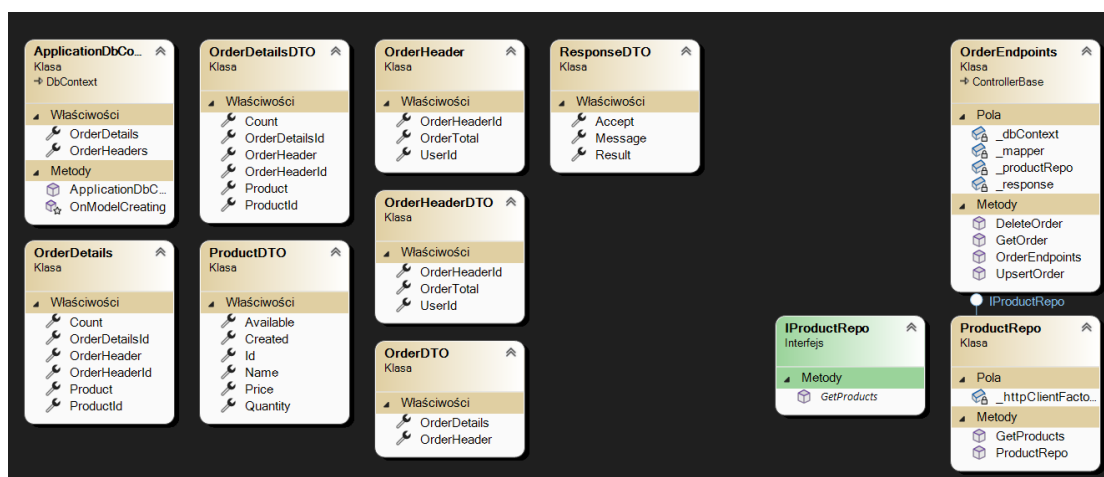
Klasy modelu serwisu ProjectAPIAuth

Po lewej klasy obiekt transferu danych. Po prawej klasy metod, interfejsu, klasy ProjectAPIAuth odpowiadającej za logowanie, rejestrację, autoryzację użytkowników



Klasy modelu serwisu ProjectAPIOrder

Po lewej klasy obiekt transferu danych. Po prawej klasy metod, interfejsu, klasy ProjectAPIOrder odpowiadającej za obsługę zamówień.



Działanie aplikacji

Do uruchomienia aplikacji należy wykonać zaktualizowanie naszej bazy poleceniem
dotnet ef database update

W terminalu w głównym katalogu projektu. Baza zostanie utworzona z migracji znajdujących się w folderze migracje.

Należy również w pliku asp settings dodać nazwę swojego serwera. Jest on wymagany do działania aplikacji. Należy to zrobić w każdym z projektów

Należy również ustawić uruchamianie projektów. Mamy trzy projekty wszystkie powinny otworzyć się przy starcie.

```
{
  "Logging": {
    "LogLevel": {
      "Default": "Information",
      "Microsoft.AspNetCore": "Warning"
    }
  },
  "AllowedHosts": "*",
  "ConnectionStrings": {
    "DefaultConnection": "Server=LAPTOP-9A79R96U;Database=ProjectAPI;TrustServerCertificate=true;Trusted_Connection=True;MultipleActiveResultSets=true"
  },
  "ApiSettings": {
    "Secret": "Weryfikacja Tokenu JWT Sekret Token Token Token"
  }
}
```

Widok swagger

Na początku możemy dodać użytkownika rejestrując się. Bez autoryzacji po zalogowaniu nie będziemy mogli np. usunąć rekordu, dostaniemy kod błędu 401, natomiast będziemy mogli wyświetlić produkt po id

DELETE /api/product/{id}

Parameters

Name Description

id * required integer(\$int32) (path) 1

Execute Clear

Responses

Curl

```
curl -X 'DELETE' \
'https://localhost:7223/api/product/1' \
-H 'accept: */*'
```

Request URL

https://localhost:7223/api/product/1

Server response

Code Details

401 Undocumented Error: response status is 401

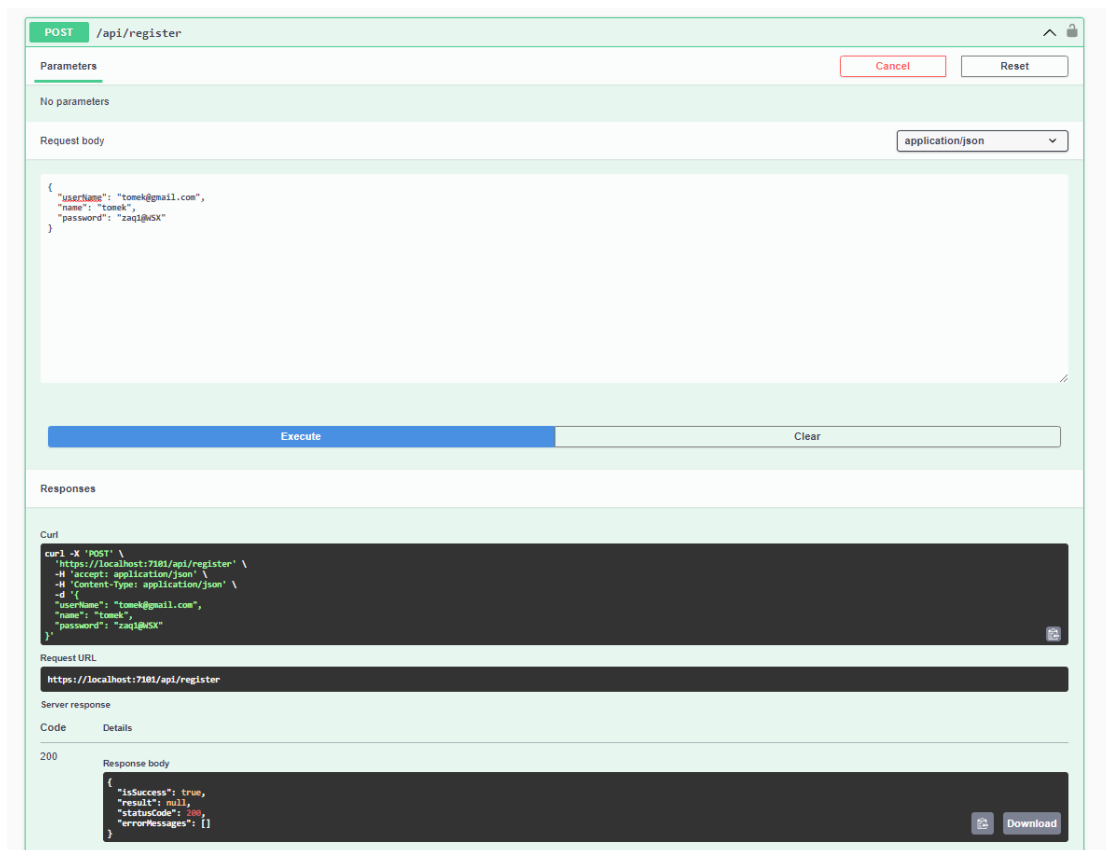
Response headers

```
content-length: 0
date: Sun, 30 Apr 2023 17:04:47 GMT
server: Kestrel
www-authenticate: Bearer
```

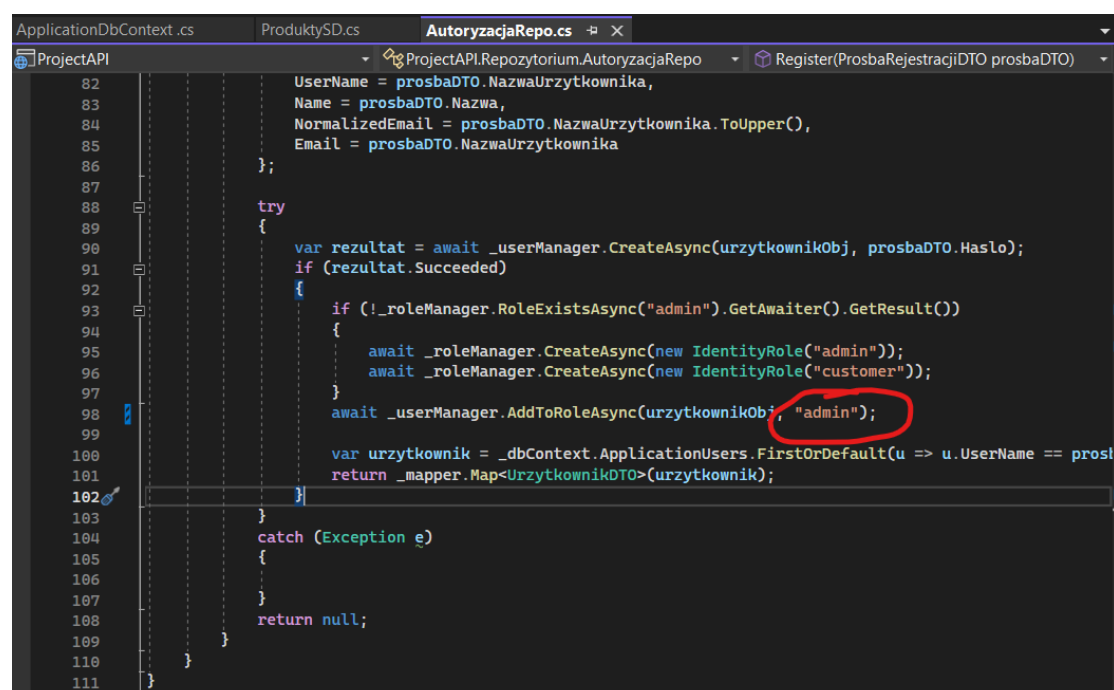
Responses

Results Messages									
	Id	Name	UserName	NormalizedUserName	Email	NormalizedEmail	EmailConfirmed	PasswordHash	
1	620fc407-1486-492f-b019-3aa640ac5894	marek	marek@gmail.com	MAREK@GMAIL.COM	marek@gmail.com	MAREK@GMAIL.COM	0	AQAAAAIAAYagAAAAEF	
2	98a4c861-22cd-433b-8045-f96495d8d5ff	lolek	lolek@gmail.com	LOLEK@GMAIL.COM	lolek@gmail.com	LOLEK@GMAIL.COM	0	AQAAAAIAAYagAAAAEH	
3	e36e1942-5693-4cba-a1a7-744d410bb0c9	tomek	romek@gmail.com	ROMEK@GMAIL.COM	romek@gmail.com	ROMEK@GMAIL.COM	0	AQAAAAIAAYagAAAAEH	
4	f53321f3-99bd-4037-8355-5bb7cde83838	tomek	tomek@gmail.com	TOMEK@GMAIL.COM	tomek@gmail.com	TOMEK@GMAIL.COM	0	AQAAAAIAAYagAAAAEC	
5	fc279012-37db-444b-9a21-d3097dc00b78	bolek	bolek@gmail.com	BOLEK@GMAIL.COM	bolek@gmail.com	BOLEK@GMAIL.COM	0	AQAAAAIAAYagAAAAEI	

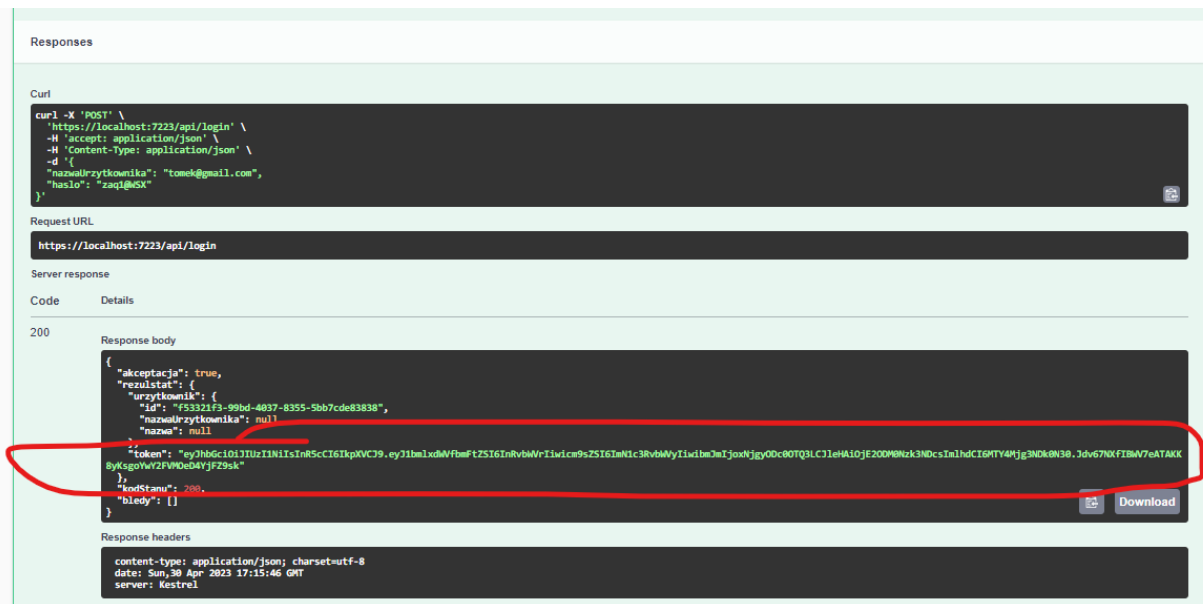
Rejestracja użytkownika. Identity przyjmuje hasła z minimum 1 małą, wielką literą, znakiem specjalnym oraz cyfrą, minimum 8 znaków



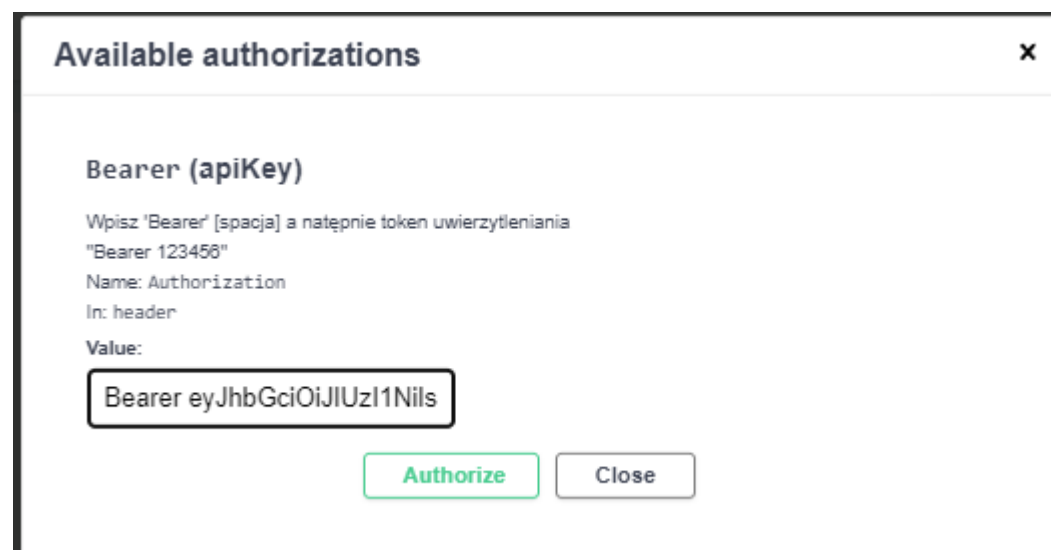
Aby zarejestrować użytkownika o innej roli należy w pliku zmienić role na admin lub centromer



Po zalogowaniu na stworzonego użytkownika otrzymujemy kod autoryzacyjny.



Wprowadzamy token tak jak na schemacie



Po zalogowaniu jako customer możemy, dokonać operacji, które normalnie są zabalowane

<div> <div>Curl</div> <div> <pre>curl -X 'GET' \ 'https://localhost:7223/api/product' \ -H 'accept: application/json' \ -H 'Authorization: Bearer eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJ1bm1kdWVmbmFtZSI6Imh0bGVwcm9sZSIsImF1dG8iOiJjoxhVjgyODc1NDQ4LjE1eHA1OjE2ODM0MDAyMTgsIm1hdC16MTY4Mjg3NTQsOH0.Nwq3JmYnp03RckZ4_1x41of'</pre> </div> <div>Request URL</div> <div>https://localhost:7223/api/product</div> <div>Server response</div> <div> <div>Code</div> <div>Details</div> </div> <div>200</div> <div>Response body</div> <div> <pre>{ "akceptacja": true, "rezultat": [{ "id": 1, "nazwa": "Kubek", "cena": 10, "ilosc": 10, "dostepny": true, "utworzony": null, "zaktualizowany": null }, { "id": 2, "nazwa": "Długopis", "cena": 20, "ilosc": 0, "dostepny": false, "utworzony": null, "zaktualizowany": null }], "kodStanu": 200, "bledy": [] }</pre> <div>Download</div> </div> <div>Response headers</div> <div> <pre>content-type: application/json; charset=utf-8 date: Sun, 30 Apr 2023 17:26:09 GMT server: Kestrel</pre> </div> </div> <div>Responses</div> <div> <div>Code</div> <div>Description</div> <div>Links</div> </div>	<div> <div>Curl</div> <div> <pre>curl -X 'GET' \ 'https://localhost:7223/api/product' \ -H 'accept: application/json' \ -H 'Authorization: Bearer eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJ1bm1kdWVmbmFtZSI6Imh0bGVwcm9sZSIsImF1dG8iOiJjoxhVjgyODc1MTQ4LjE1eHA1OjE2ODM0MDAzNDAsIm1hdC16MTY4Mjg3NTQsOH0.Y4U3lV0PRjnyS7wrsfG'</pre> </div> <div>Request URL</div> <div>https://localhost:7223/api/product</div> <div>Server response</div> <div> <div>Code</div> <div>Details</div> </div> <div>403</div> <div>Error: response status is 403</div> <div>Response headers</div> <div> <pre>content-length: 0 date: Sun, 30 Apr 2023 17:28:07 GMT server: Kestrel</pre> </div> </div> <div>Responses</div> <div> <div>Code</div> <div>Description</div> <div>Links</div> </div>
--	--

Po zalogowaniu jako admin możemy dokonać operacji odblokowanych tylko dla roli administratora.

W części odpowiadającej za zamówienia, możemy tworzyć zamówienia na podstawie produktów. Serwisy komunikują się ze sobą, dlatego produkty z jednej bazy są dostępne w drugiej z zamówieniami. Automatycznie obliczana jest wartość, można też dodawać usuwać ilość i dodawać wiele produktów do jednego zamówienia.