

SPRAWOZDANIE Z PROJEKTU: ASP.NET MVC

Jacek Kowalczyk i Jakub Szczepaniak

Grupa L2

Temat: Aplikacja do zarządzania zamówieniami i magazynem

Technologia: ASP.NET Core 8 MVC, Entity Framework Core, SQL Server (SQLEXPRESS).

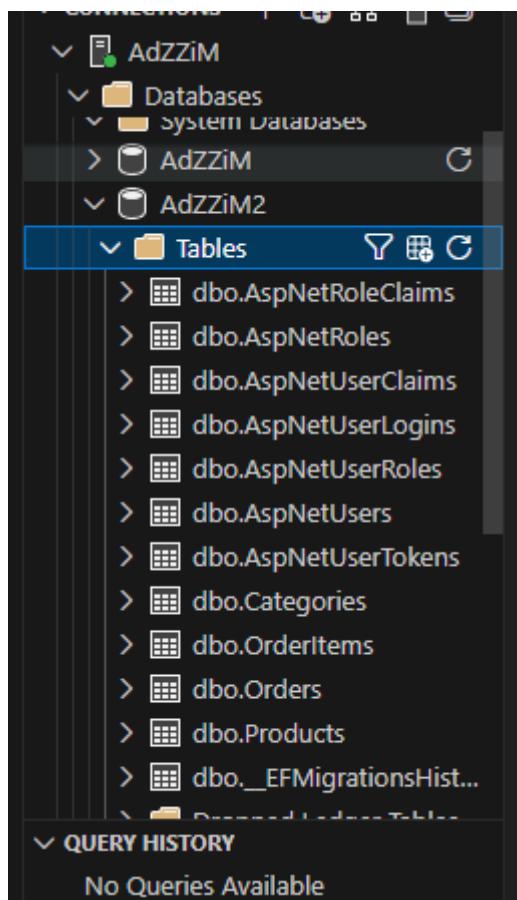
1. Wstęp i architektura systemu

Projekt AdZZiM został zrealizowany zgodnie z wzorcem Model-View-Controller (MVC). Zapewnia to separację logiki biznesowej od warstwy prezentacji. Aplikacja uruchamia się bez błędów i korzysta z lokalnej instancji bazy danych.

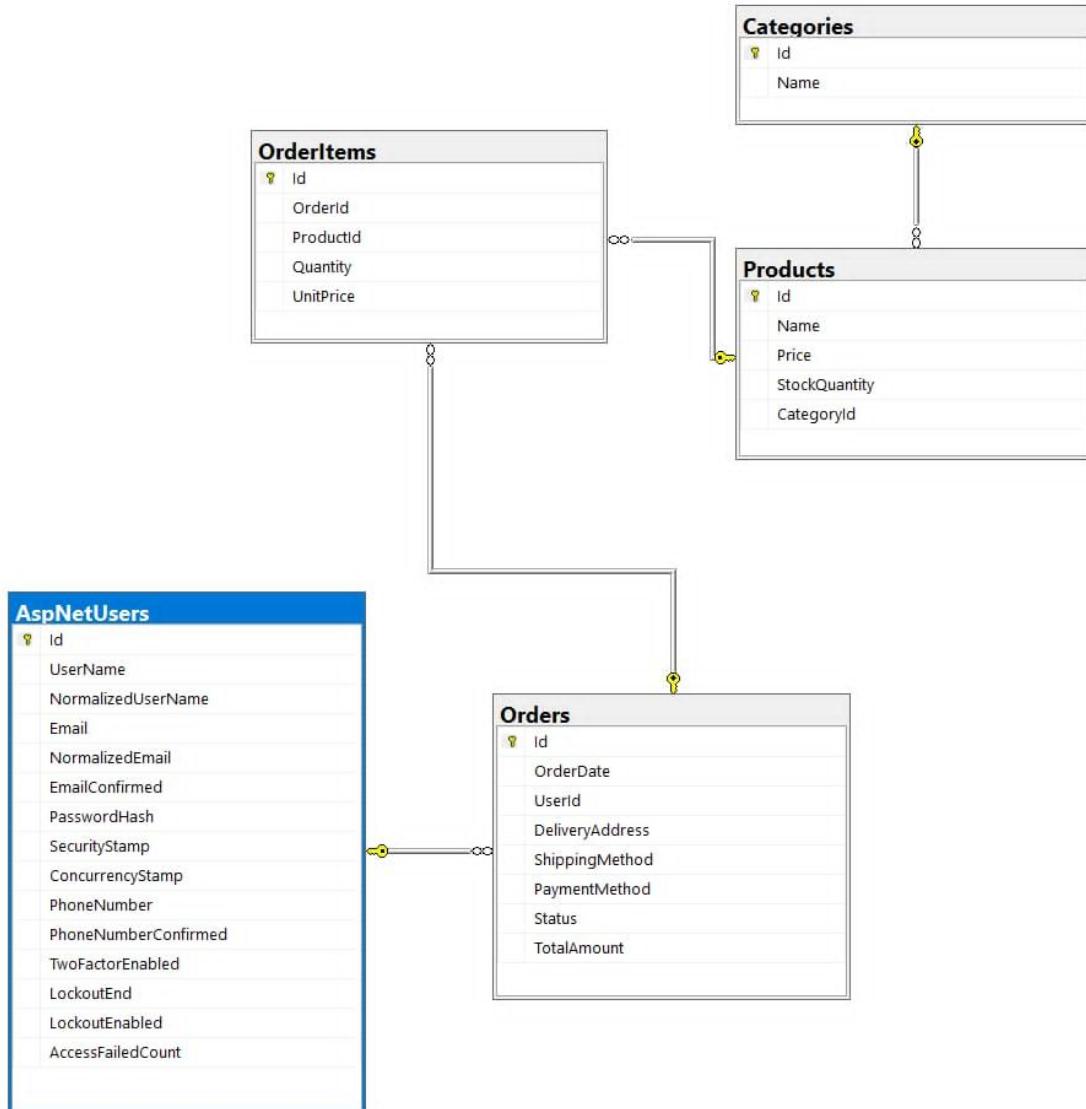
2. Warstwa danych i modele

System wykorzystuje Entity Framework Core do utrwalania danych. W projekcie zdefiniowano 9 klas modeli, z czego 4 stanowią rdzeń biznesowy aplikacji i występują w relacjach:

- **Product:** Zawiera pola Name, Price, StockQuantity oraz klucz obcy CategoryId.
- **Category:** Grupuje produkty (Relacja 1:N).
- **Order:** Zarządza danymi zamówienia, takimi jak DeliveryAddress czy PaymentMethod.
- **OrderItem:** Encja łącząca produkty z zamówieniami (szczegółły pozycji).



Poniższy diagram przedstawia relacje między tabelami w systemie AdZZiM, w tym kluczowe powiązania jeden-do-wielu między kategoriami a produktami oraz zamówieniami a ich pozycjami



3. Walidacja i Formularze

Aplikacja wykorzystuje mechanizm Model Binding oraz atrybuty z przestrzeni System.ComponentModel.DataAnnotations do zapewnienia integralności danych. Walidacja odbywa się dwutorowo: po stronie klienta (przez skrypty jQuery Validation) oraz po stronie serwera w kontrolerach.

Model	Zastosowane atrybuty walidacji	Szczegółowe przeznaczenie
Product	[Required], [StringLength(100)], [Range(0.01, 100000)]	Blokuje dodanie produktu bez nazwy lub z ceną ujemną. Chroni bazę przed zbyt długimi ciągami znaków.
Order	[Required], [EmailAddress], [Phone]	Gwarantuje, że dane do wysyłki są kompletne, a format e-maila i telefonu jest poprawny dla kuriera.
Category	[Required], [Display(Name = "Nazwa Kategorii")]	Wymusza unikalną nazwę grupy i mapuje ją na czytelny dla użytkownika tekst w widoku.
Identity	[DataType(DataType.Password)], [Compare]	Obsługuje bezpieczną rejestrację użytkowników, wymuszając potwierdzenie hasła.

The screenshot shows a web browser displaying a form titled 'Dodaj produkt' (Add product). The form fields are as follows:

- Nazwa**: An input field with a red border, indicating it is required. A validation message 'Nazwa produktu jest wymagana' is displayed below it.
- Cena**: An input field with a red border, indicating it is required. A validation message 'The Price field is required.' is displayed below it.
- Ilość**: An input field with a red border, indicating it is required. A validation message 'The StockQuantity field is required.' is displayed below it.
- Kategoria**: A dropdown menu showing the value '1'.

Below the form are two buttons: a blue 'Dodaj' (Add) button and a blue 'Powrót' (Back) button.

4. Autoryzacja i API (Punkt 6c, 6d)

- Autoryzacja :** System wykorzystuje Microsoft.AspNetCore.Identity. Gość ma dostęp do przeglądania produktów i wyszukiwania. Administrator (admin@adzzim.pl) posiada pełne uprawnienia CRUD (edyja, usuwanie, dodawanie).

Admin

AdZZiM Start Oferta Kategorie [Złożyć Zamówienie](#) Wszystkie zamówienia [Panel Administratora](#) Witaj, admin@adzzim.pl! Wyloguj

Witaj w AdZZiM

System zarządzania zamówieniami i magazynem.

Przeglądaj Ofertę
Sprawdź produkty
[Zobacz Produkty](#)

Twoje Zamówienia
Złoż nowe zamówienie lub sprawdź statusy swoich poprzednich zakupów.
[Złożyć Zamówienie](#)

Panel Administracyjny
Pełna kontrola nad magazynem, edycja produktów i zarządzanie zamówieniami.

[Dodaj produkt](#)
[Dodaj kategorie](#)
[Zarządzaj zamówieniami](#)
[Zarządzaj użytkownikami](#)

© 2025 - AdZZiM - [Polityka Prywatności](#)

Użytkownik

AdZZiM Start Oferta Kategorie [Złożyć Zamówienie](#) [Zarejestruj się](#) [Zaloguj się](#)

Witaj w AdZZiM

System zarządzania zamówieniami i magazynem.

Przeglądaj Ofertę
Sprawdź produkty
[Zobacz Produkty](#)

Twoje Zamówienia
Zaloguj się, aby móc składać zamówienia i śledzić ich status.
[Zaloguj się](#)
[Zarejestruj się](#)

Panel Administracyjny
Sekcja dostępna tylko dla użytkowników z uprawnieniami Administratora.

© 2025 - AdZZiM - [Polityka Prywatności](#)

- **API CRUD (6d):** Udostępniono endpoint ProductsApiController, który zwraca listę produktów w formacie JSON.

Nazwa	Cena	Ilość	Kategoria
test	50,00	1	1

© 2025 - AdZZiM - [Polityka Prywatności](#)

`< > C localhost:5101/api/productsapi`

`Bootstrap 5 Grid B... Upper-intermediat... Software Engineer... Cisco Networking...`

`Zastosuj formatowanie stylistyczne`

```
[{"id":1,"name":"test","price":50.00,"stockQuantity":1,"categoryId":1,"category":null}]
```

5. Dokumentacja instalacyjna

Projekt został przygotowany do uruchomienia w środowisku lokalnym przy wykorzystaniu .NET 8 SDK.

- Konfiguracja bazy: Aplikacja automatycznie tworzy bazę danych AdZZiM2 na lokalnym serwerze .\SQLEXPRESS dzięki mechanizmowi migracji Entity Framework Core.
- Łąćuch połączenia:
`Server=.\SQLEXPRESS;Database=AdZZiM2;Trusted_Connection=True;MultipleActiveResultSets=true;TrustServerCertificate=True;`.
- Dane uwierzytelniające (Administrator):
 - Login: admin@adzzim.pl
 - Hasło: admin222.
- Instrukcja uruchomienia:
 1. Sklonuj repozytorium: `git clone https://github.com/Jacek126d/AdZZiM`.
 2. Otwórz folder projektu w terminalu (np. w VS Code).
 3. Przywróć pakiety i stwórz bazę polecienniem: `dotnet ef database update`.
 4. Uruchom aplikację: `dotnet run`.
- Adres API: Po uruchomieniu dane JSON są dostępne pod adresem: `https://localhost:7105/api/productsapi`.

