

# Podstawy Konteneryzacji i Architektury Mikroserwisów

Aplikacja webowa: lista książek

**Damian Opolski, 202280**

## Wstęp

Aplikacja ma na celu ułatwienie użytkownikom przeglądanie posiadanej lub interesującej ich kolekcji książek. Dzięki intuicyjnemu interfejsowi i podstawowym funkcjom, takim jak wyszukiwanie, filtrowanie i zarządzanie rekordami, aplikacja ta oferuje prosty, ale skuteczny sposób na utrzymanie porządku w cyfrowym (lub planowanym) księgozbiorze. W dalszej części wypracowania przeanalizuję poszczególne elementy aplikacji, jej funkcjonalności, a także zastanowię się nad potencjalnymi ulepszeniami.

## Kluczowe Funkcje Aplikacji

### Wyświetlanie Listy Książek:

Aplikacja prezentuje listę książek w formie tabeli, gdzie każda książka jest oddzielnym wierszem.

Poszczególne kolumny tabeli zawierają istotne informacje o książkach, takie jak tytuł, autor, liczba stron, rok wydania i rodzaj okładki.

### Wyszukiwanie Książek:

Aplikacja posiada pole wyszukiwania, które pozwala użytkownikom na szybkie odnalezienie książki po tytule (a być może po innych kryteriach, jeśli wyszukiwanie jest bardziej zaawansowane).

### Filtrowanie Książek:

Dostępny jest filtr oparty na rodzaju okładki (soft, hard), który pozwala zawęzić listę książek do konkretnego typu.

### Dodawanie Książek:

Aplikacja umożliwia dodawanie nowych książek do listy za pomocą przycisku "+ Add". (Na podstawie obrazu nie widać konkretnego formularza, ale logika podpowiada że po naciśnięciu użytkownik będzie mógł wprowadzić dane nowej książki)

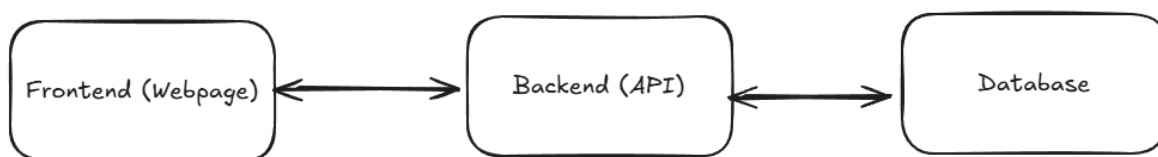
## Edycja i Usuwanie Książek:

Dla każdej książki dostępna jest opcja "Actions", która po rozwinięciu pozwala na edycję danych książki oraz jej usunięcie.

## Podstawowe Informacje o Książce:

Aplikacja wyświetla podstawowe dane o każdej książce, w tym tytuł, autora, liczbę stron, rok wydania i rodzaj okładki.

# Schemat architektury mikroserwisów



## Warstwa Prezentacji – Frontend (Webpage)

Warstwa ta jest odpowiedzialna za wizualne przedstawienie danych oraz umożliwienie użytkownikowi interakcji z aplikacją. W kontekście naszego systemu do zarządzania książkami, frontend prezentuje listę książek w czytelnej tabeli, umożliwia wyszukiwanie i filtrowanie pozycji, a także udostępnia formularze do dodawania, edytowania i usuwania książek.

Interfejs użytkownika jest zaprojektowany tak, aby był intuicyjny i przyjazny dla użytkownika. Dostępne opcje wyszukiwania i filtrowania, a także przejrzyste przedstawienie informacji, pozwala użytkownikowi na sprawne nawigowanie po systemie. Frontend nie tylko prezentuje dane, ale też odzwierciedla reakcje użytkownika. Przykładowo, po naciśnięciu przycisku "Add", użytkownikowi wyświetli się formularz do wprowadzenia danych nowej książki.

## Warstwa Logiki Biznesowej – Backend (API)

Backend obsługuje zapytania takie jak "pokaż wszystkie książki", "dodaj nową książkę", "edytuj książkę o danym identyfikatorze", "usuń wybraną książkę".

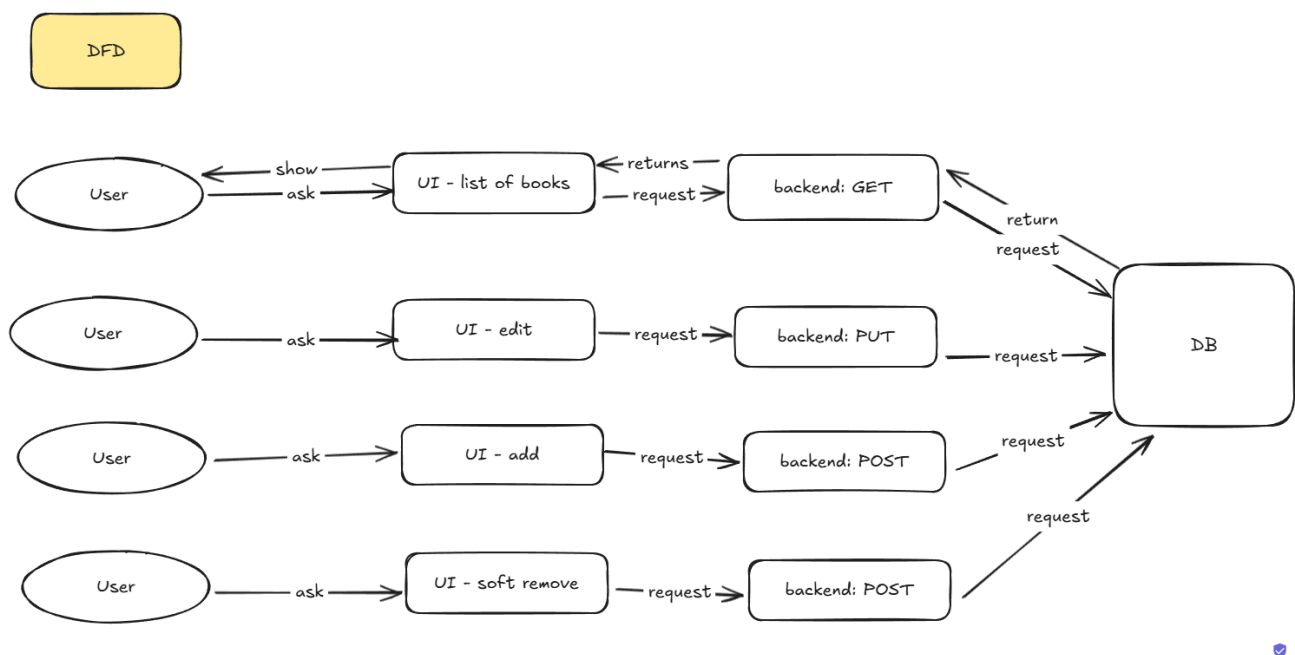
Logika biznesowa zawarta w backendzie obejmuje również walidację danych, czyli sprawdzanie poprawności informacji wprowadzanych przez użytkownika. Dodatkowo, backend zarządza komunikacją z bazą danych, wysyłając do niej zapytania i odbierając dane. Kluczowe jest, że backend ukrywa przed frontendem szczegóły implementacji bazy danych, udostępniając mu do tego celu odpowiednie interfejsy API.

## Warstwa Danych – Database

Warstwa ta to magazyn danych aplikacji. To właśnie w bazie danych przechowywane są informacje o wszystkich książkach, takie jak tytuł, autor, liczba stron, rok wydania i rodzaj okładki.

Baza danych zapewnia trwałość danych, co oznacza, że nawet po zamknięciu aplikacji, dane są bezpieczne i dostępne do ponownego wykorzystania. Warstwa ta umożliwia backendowi dostęp do danych poprzez wysyłanie i odbieranie zapytań, przy czym sama struktura danych jest zorganizowana tak, aby dostęp do nich był szybki i efektywny. Baza danych jest kluczowa dla utrzymania integralności danych i ich prawidłowego przechowywania.

## Data flow diagram



## Entity relation diagram(ERD)

posts	
id 🔗	integer
title	varchar
author	text
pages	integer
release_year	varchar
cover	text
is_deleted	bool


## Tech Stack

- Angular
- Material UI
- .NET Core
- Entity Framework
- MariaDB

## Widok użytkownika

Search Cover ▾

+ Add

Title	Author	Pages	Release Year	Cover	Actions
Harry Potter	J.K. Rowling	724	2001	hard	<div></div>
Hobbit	J. R. R. Tolkien	1088	1955	soft	<div><div> Edit</div><div> Delete</div></div>
Wiedźmin: Ostatnie życzenie	Andrzej Sapkowski	328	1993	soft	<div></div>
Metro 2033	Dmitry Glukhovsky	480	2005	soft	<div></div>
Diuna	Frank Herbert	604	1965	hard	<div></div>
Trylogia Husycka	Andrzej Sapkowski	988	1996	hard	<div></div>
Rok 1984	George Orwell	328	1949	soft	<div></div>
Zabić drozda	Harper Lee	281	1960	hard	<div></div>

Search Cover ▾

Title
Harry Potter
Hobbit
Wiedźmin: Ostatnie życzenie
Metro 2033
Diuna
Trylogia Husycka
Rok 1984
Zabić drozda
Duma i uprzedzenie
Sto lat samotności
Lśnienie
I nie było już nikogo

### Editing existing book

Title
Hobbit

Author  
J. R. R. Tolkien

Pages  
1088

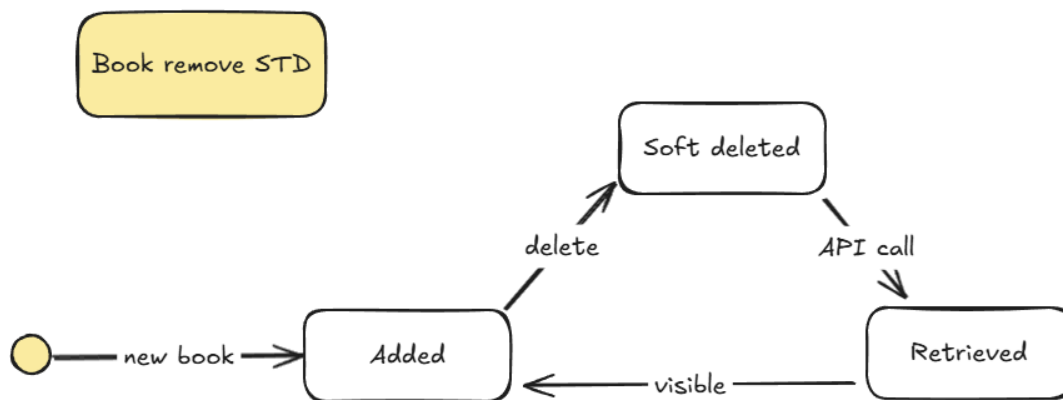
Release Year  
1955

Cover  
Soft

+ Edit

I nie było już nikogo	Agatha Christie	256	1939
-----------------------	-----------------	-----	------

## State transition diagram (STD)



## Widok kontenerów w Docker

DC400CTIUDD1						
<input type="checkbox"/>		<a href="#">angularkurs7</a>		Running (3/3)	0.02%	23 minutes ago
<input type="checkbox"/>		<a href="#">front-app</a> 85e8665bca96	<a href="#">angularkurs7-front-app</a>	Running	0% <a href="#">8080:80</a>	23 minutes ago
<input type="checkbox"/>		<a href="#">mariadb-container</a> 2c40eccc3375	<a href="#">mariadb:latest</a>	Running	0.01% <a href="#">3306:3306</a>	23 minutes ago
<input type="checkbox"/>		<a href="#">dotnet-app</a> f8609ceed24e	<a href="#">angularkurs7-dotnet-ap</a>	Running	0.01% <a href="#">5241:80</a>	23 minutes ago