

LEKCE 11

Refactoring, Tidbits, Delegates



Závěrečný úkol

Jak to jde?

Závěrečný projekt - Flow Odevzdávání

ToDoList Categories
Backend

2.12.

Úprava backendu,
aby podporoval kategorie

- Models
- Entity + Context
- Database Schema
 - Migrations
- DTOs
- Endpoint Controller
- Testy

ToDoList Categories
Frontend

9.12.

Úprava frontendu,
aby podporoval kategorie

- Dashboard component
- EditItem component
- Opravy BE na základě zpětné vazby

ToDoList Categories
Final

16.12.

Prostor pro vychytání chyb a
vylepšení programu a kódu.

- Vylepšení vzhledu pro uživatele frontendu
- Opravy FE na základě zpětné vazby
- Refactoring
 - jednoduchost
 - přehlednost
 - pojmenování



Konfigurace

Proč je konfigurace důležitá?

- Odděluje kód od nastavení
- Usnadňuje údržbu a nastavování parametrů
- Flexibilita nasazení

Prakticky v ToDoList aplikaci

- Nastavení logování
- Nastavení databázového připojení (vývojová x produkční)
- Nastavení připojení HTTP klienta na backend

The background is a solid pink color with various abstract geometric patterns. In the top left, there is a grid of small dots. In the top right, there is a large, faint, light pink arrow pointing upwards and to the right. In the bottom left, there is a faint line-art illustration of a plant with leaves and a flower. In the bottom right, there is a large, faint, light pink 'X' shape. A semi-circular area in the bottom center contains a pattern of small dots, and a rectangular area to its right contains diagonal stripes.

Validace

Prakticky v ToDoList aplikaci

- Klientská validace v EditToDoItem (ToDoItemView)
- Serverová validace - REST API (DTOs)
- Serverová validace - databáze (EF Core ToDoItem Entity)

Dokumentace OpenAPI



ProducesResponseType<Type>(StatusCode)

The background is a solid pink color with various abstract geometric patterns. In the top left, there is a grid of small dots. In the top right, there is a large, faint arrow pointing upwards and to the right. In the bottom left, there is a faint outline of a leaf or flower. In the bottom right, there is a large, faint 'X' shape and a circular area with diagonal stripes.

Delegáti

```
public delegate void MessageTakingDelegate(string message);
```

```
public delegate int NumberOperationDelegate(int n1, int n2);
```

Jak se používají?

Deklarace delegáta

```
public delegate void MyDelegate(string input);
```

Deklarace a Inicializace delegátového typu

```
public MyDelegate mujDelegat = MojeMetodaSplnujiciDelegat;
```

Invokace (volání)

```
mujDelegat("zprava");
```

```
mujDelegat.Invoke("zprava");
```

Action, Func

(předpřípravení delegáti)

Action<ParameterType1, ...>

Func<ParameterType1, ..., ReturnType>

Lambda výrazy

(anonymní funkce)

Context



(input-parameters) => expression

(input-parameters) => { <sequence-of-statements> }

Kdy použít co?

Použití	Delegáti	Anonymní funkce (lambdy)
Jednorázové volání	✗ Příliš složité	✓ Jednoduché a přehledné
Více metod (multicast)	✓ Delegát zvládá připojení více metod	✗ Není vhodné
Znovupoužitelnost	✓ Lepší pro logiku používanou na více místech	✗ Není ideální, pokud je třeba opakovaně používat
Jednoduchost	✗ Vyžaduje více kódu	✓ Krátké a snadné na psaní
Podpora closure	✗ Delegáti nezachycují okolní kontext	✓ Lambdy umožňují použití closure