CollectionView — katalog produktów

Czas: 2 × 45 min | praca indywidualna | Wykład 2.27

Cel: Zbudujesz katalog produktów z CollectionView: szablon z obrazem, wyszukiwanie, grupowanie po kategorii, odświeżanie gestem i usuwanie przesunięciem z potwierdzeniem.

Po zajęciach potrafisz:

- projektuje szablon elementu i nagłówek grupy CollectionView
- filtruje i grupuje dane w klasie logiki
- obsługuje RefreshView i EmptyView
- stosuje SwipeView i gest dotknięcia z komendą

Sprzęt i materiały

- komputer z Windows 11 i dostępem do internetu
- pendrive lub konto w repozytorium do zapisania projektu

Oprogramowanie

- **Visual Studio Community 2026 (obciążenie „Programowanie aplikacji klasycznych platformy .NET”)** — darmowe dla uczniów i nauczycieli; w pracowni zainstalowane visualstudio.microsoft.com/pl/downloads
- **.NET 10 SDK (LTS)** — instalowany razem z Visual Studio; sprawdzenie: `dotnet --version` dotnet.microsoft.com/pl-pl/download/dotnet/10.0
- **Obciążenie VS „Programowanie aplikacji wieloplatformowych w .NET” (.NET MAUI) z emulatorem Androida** — emulator wymaga wirtualizacji sprzętowej; awaryjnie uruchamiaj jako Windows Machine learn.microsoft.com/dotnet/maui/get-started/installation
- **Pakiet NuGet CommunityToolkit.Mvvm 8.4** — instalacja: Zarządzaj pakietami NuGet nuget.org/packages/CommunityToolkit.Mvvm

Przygotowanie przed zajęciami

- Ikony kategorii `audio.png`, `akcesoria.png`, `pamieci.png` z folderu `pliki` przy tym zadaniu.

Bezpieczeństwo

- Zapisuj projekt w swoim folderze (np. `C:\Users\<login>\source\repos`), a na koniec zajęć skopiuj go na pendrive lub wyślij do repozytorium.
- Nie uruchamiaj programów i pakietów z nieznanych źródeł; pakiety NuGet instaluj tylko te, które wskazuje zadanie.

Przebieg zadania

1 Logika

Projekt MAUI `Katalog`; klasy bez MAUI.

1. Rekord `Produkt(Id, Nazwa, Kategoria, Cena, Obraz)` i 7 produktów w 3 kategoriach.
2. `GrupaProduktow : List<Produkt>` z właściwościami Nazwa i Suma.
3. `KatalogLogika.Filtruj(produkty, fraza)` (nazwa lub kategoria, bez wielkości liter) i `Grupuj` (kategorie alfabetycznie).

2 ViewModel

`ProduktyViewModel`.

1. Kolekcja `Grupy`; `Fraza` → ponowne filtrowanie i grupowanie (OnFrazaChanged).
2. `OdswiezAsync`: symulacja pobrania (Task.Delay), przywrócenie danych, `Odswiezanie = false`.
3. `UsunAsync(Produkt)` z pytaniem; `PokazSzczegolyAsync` — komunikat z ceną.
4. Podsumowanie „Produktów: N".

3 Strona

XAML.

```
<CollectionView ItemsSource="{Binding Grupy}" IsGrouped="True">
  <CollectionView.GroupHeaderTemplate>
    <DataTemplate x:DataType="m:GrupaProduktow">
      <Label Text="{Binding Nazwa}" FontAttributes="Bold" />
    </DataTemplate>
  </CollectionView.GroupHeaderTemplate>
```

1. RefreshView z CollectionView `IsGrouped="True"`; SearchBar w nagłówku; EmptyView z podpowiedzią.
2. GroupHeaderTemplate: nazwa kategorii i liczba sztuk na kolorowym tle.
3. ItemTemplate: obraz 48 × 48, nazwa, kategoria, cena; SwipeView (Reveal) z akcją Usuń; TapGestureRecognizer → szczegóły.
4. Tło elementów z `AppThemeBinding` (jasny/ciemny).

4 Sprawdzenie

Scenariusze.

1. Fraza „audio", „mysz", „xyz" — liczba wyników i EmptyView.
2. Usuń produkt, pociągnij w dół — produkt wraca.
3. Zatwierdź pracę w Git.

W sprawozdaniu umieść

- ☐ kod klas logiki i ViewModelu
- ☐ XAML strony
- ☐ tabela scenariuszy
- ☐ zrzuty: grupy, wyszukiwanie, pusta lista

Pytania kontrolne

1. Jakie wymagania ma kolekcja dla CollectionView z grupami?
2. Co się stanie, jeśli nie ustawisz Odswiezanie = false?
3. Czym różni się SwipeItems Mode="Reveal" od "Execute"?
4. Dlaczego filtrowanie umieszczamy w osobnej klasie?

Kryteria oceny

Kryterium	Punkty
Klasy logiki (filtr, grupy)	5
ViewModel	5
Strona z grupami, EmptyView, Swipe, Refresh	7
Pytania	3
Razem	**20**

Sprawozdanie przygotuj na wzorze dołączonym do zadania (plik .docx o tym samym numerze).