Agenda 04 - Desenvolvimento de Sistemas III

Parte 1:

**1. Você conhece o .NET MAUI?**

Sim. O **.NET MAUI (Multi-platform App UI)** é um framework da Microsoft que permite criar aplicativos multiplataforma (Android, iOS, Windows e macOS) com uma única base de código em **C# e XAML**. Ele é a evolução do Xamarin.Forms, trazendo melhor performance, integração com recursos nativos e suporte ampliado a MVVM.

**2. Você conhece sobre mecanismos de buscas?**

Sim. Em termos de UI, um mecanismo de busca é um recurso que permite ao usuário filtrar e localizar itens em tempo real.  
No caso de listas em aplicativos, isso geralmente é feito através de um **SearchBar** que dispara eventos sempre que o texto muda, aplicando um **filtro sobre a coleção de dados** exibida.

**3. Como implementar o evento TextChanged em um SearchBar?**

O SearchBar do .NET MAUI possui o evento **TextChanged**, que é disparado toda vez que o usuário digita ou apaga caracteres. Esse evento pode ser usado para chamar um método que atualiza a lista com base no texto digitado.

Exemplo em C#:

private void SearchBar\_TextChanged(object sender, TextChangedEventArgs e)

{

var filtro = e.NewTextValue.ToLower();

var itensFiltrados = Produtos.Where(p => p.Descricao.ToLower().Contains(filtro)).ToList();

ProdutosFiltrados.Clear();

foreach (var item in itensFiltrados)

ProdutosFiltrados.Add(item);

}

**Como filtrar dados em uma ObservableCollection em tempo real?**

A ObservableCollection<T> é usada porque ela **notifica automaticamente a interface** quando há alterações (adição, remoção ou substituição de itens).  
Para implementar o filtro, é comum ter **duas coleções**:

* **Produtos** → coleção original (todos os itens).
* **ProdutosFiltrados** → coleção que é exibida na interface.

Assim, sempre que o usuário digita no SearchBar, a coleção ProdutosFiltrados é atualizada com os itens correspondentes.

**5. Exemplo prático de utilização do SearchBar para busca dinâmica de produtos**

**Model**

**public class Produto**

**{**

**public string Descricao { get; set; }**

**public double Preco { get; set; }**

**}**

**ViewModel**

**using System.Collections.ObjectModel;**

**using System.Linq;**

**public class ProdutoViewModel**

**{**

**public ObservableCollection<Produto> Produtos { get; set; }**

**public ObservableCollection<Produto> ProdutosFiltrados { get; set; }**

**public ProdutoViewModel()**

**{**

**Produtos = new ObservableCollection<Produto>**

**{**

**new Produto { Descricao = "Arroz", Preco = 5.50 },**

**new Produto { Descricao = "Feijão", Preco = 7.20 },**

**new Produto { Descricao = "Macarrão", Preco = 4.10 },**

**new Produto { Descricao = "Café", Preco = 12.00 }**

**};**

**ProdutosFiltrados = new ObservableCollection<Produto>(Produtos);**

**}**

**public void FiltrarProdutos(string textoBusca)**

**{**

**var filtro = textoBusca?.ToLower() ?? "";**

**var itensFiltrados = Produtos**

**.Where(p => p.Descricao.ToLower().Contains(filtro))**

**.ToList();**

**ProdutosFiltrados.Clear();**

**foreach (var item in itensFiltrados)**

**ProdutosFiltrados.Add(item);**

**}**

**}**

**View (XAML)**

**<ContentPage xmlns="http://schemas.microsoft.com/dotnet/2021/maui"**

**xmlns:x="http://schemas.microsoft.com/winfx/2009/xaml"**

**xmlns:local="clr-namespace:MauiApp"**

**x:Class="MauiApp.MainPage">**

**<ContentPage.BindingContext>**

**<local:ProdutoViewModel />**

**</ContentPage.BindingContext>**

**<VerticalStackLayout Padding="20">**

**<!-- Barra de busca -->**

**<SearchBar Placeholder="Buscar produto..."**

**TextChanged="SearchBar\_TextChanged" />**

**<!-- Lista de produtos -->**

**<CollectionView ItemsSource="{Binding ProdutosFiltrados}">**

**<CollectionView.ItemTemplate>**

**<DataTemplate>**

**<Frame BorderColor="LightGray" Padding="10" Margin="5">**

**<VerticalStackLayout>**

**<Label Text="{Binding Descricao}" FontAttributes="Bold"/>**

**<Label Text="{Binding Preco, StringFormat='Preço: {0:C}'}"/>**

**</VerticalStackLayout>**

**</Frame>**

**</DataTemplate>**

**</CollectionView.ItemTemplate>**

**</CollectionView>**

**</VerticalStackLayout>**

**</ContentPage>**

**Code-behind**

**private void SearchBar\_TextChanged(object sender, TextChangedEventArgs e)**

**{**

**var vm = BindingContext as ProdutoViewModel;**

**vm?.FiltrarProdutos(e.NewTextValue);**

**}**