Towards the .NET Junior Developer

The extremely solid course

Lesson 11

Application UI development

Agenda

Towards the .NET Junior Developer

UI libraries and frameworks

Windows Forms


```csharp
using System;
using System.Windows.Forms;


namespace HelloWorld
{
    public partial class Form1 : Form
    {
        public Form1()
        {
            InitializeComponent();
        }

        private void btnClickThis_Click(object sender, EventArgs e)
        {
            lblHelloWorld.Text = "Hello World!";
        }
    }
}
```

WPF – Windows Presentation Foundation

```xml
<DataTemplate>
  <Ellipse Width="30" Height="30">
    <Ellipse.Resources>
      <Style TargetType="Ellipse">
        <Setter Property="Fill" Value="Transparent"/>
        <Style.Triggers>
          <DataTrigger Binding="{Binding State}" Value="-1">
            <Setter Property="Fill" Value="LightPink"/>
          </DataTrigger>
          <DataTrigger Binding="{Binding State}" Value="0">
            <Setter Property="Fill" Value="LightBlue"/>
          </DataTrigger>
          <DataTrigger Binding="{Binding State}" Value="1">
            <Setter Property="Fill" Value="Green"/>
          </DataTrigger>
          <DataTrigger Binding="{Binding State}" Value="2">
            <Setter Property="Fill" Value="Azure"/>
          </DataTrigger>
          <DataTrigger Binding="{Binding State}" Value="3">
            <Setter Property="Fill" Value="Red"/>
          </DataTrigger>
          <DataTrigger Binding="{Binding State}" Value="4">
            <Setter Property="Fill" Value="RosyBrown"/>
          </DataTrigger>
          <DataTrigger Binding="{Binding State}" Value="5">
            <Setter Property="Fill" Value="SaddleBrown"/>
          </DataTrigger>
        </Style.Triggers>
      </Style>
    </Ellipse.Resources>
  </Ellipse>
</Ellipse>
```
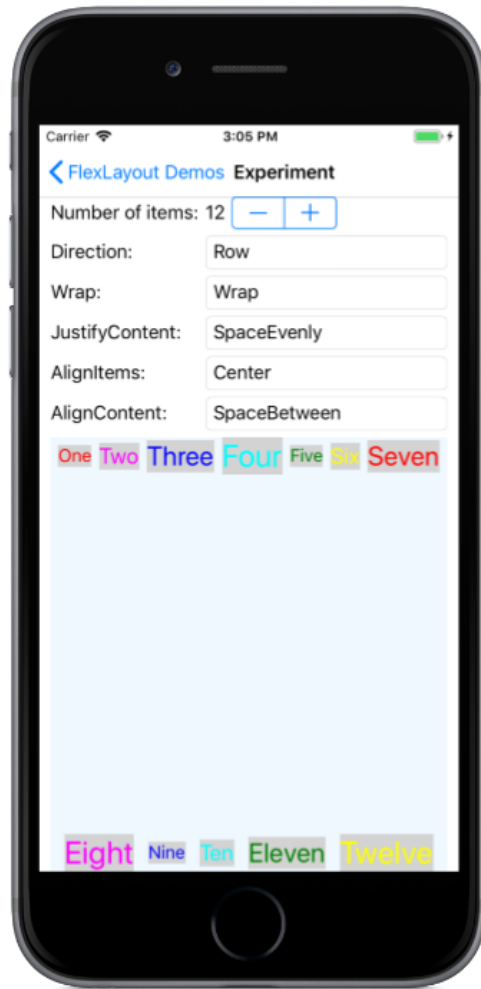
```csharp
48  public class Data
49  {
        1 reference
50      public int ID { get; set; }
        1 reference
51      public ModeState State { get; set; }
52  }
53
    2 references
54  public enum ModeState
55  {
56      NotDefine = -1,
57      Stop = 0,
58      Automatic = 1,
59      Wait = 2,
60      Error = 3,
61      Jam = 4,
62      Block = 5
63  }
64  }
```

Window61

Ellipse	ID	State
	1	NotDefine
	2	Stop
	3	Automatic
	4	Wait
	5	Error
	6	Jam
	7	Block

Xamarin.Forms

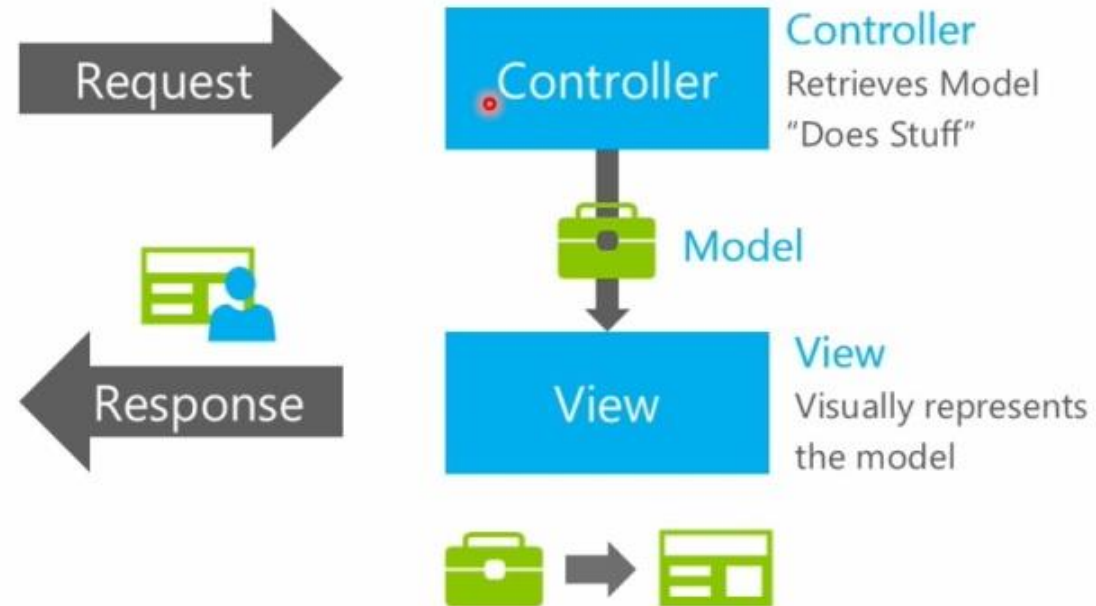# MAUI – Multi-platform Application UI
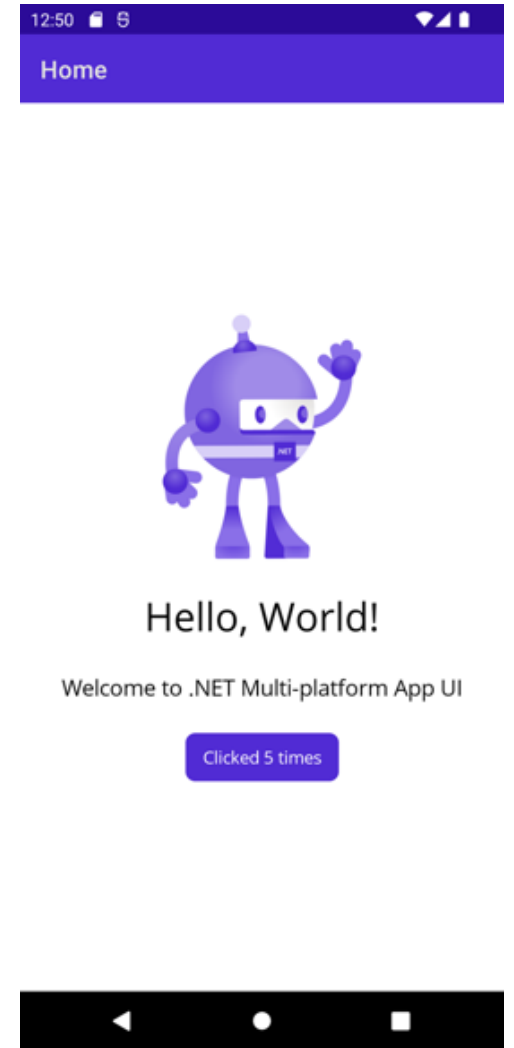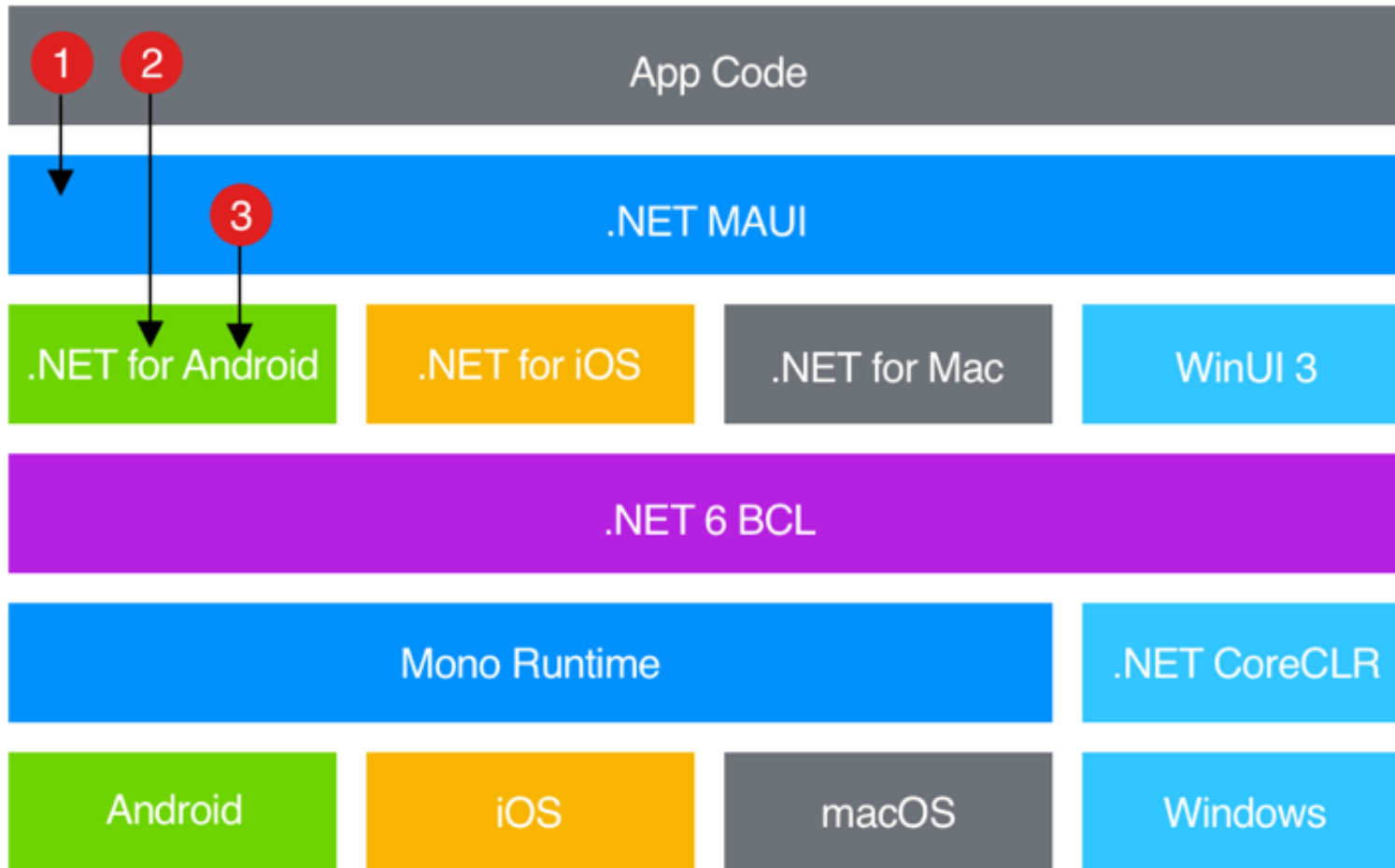
Blazor development basics

MVC pattern

Models, Views, and Controllers
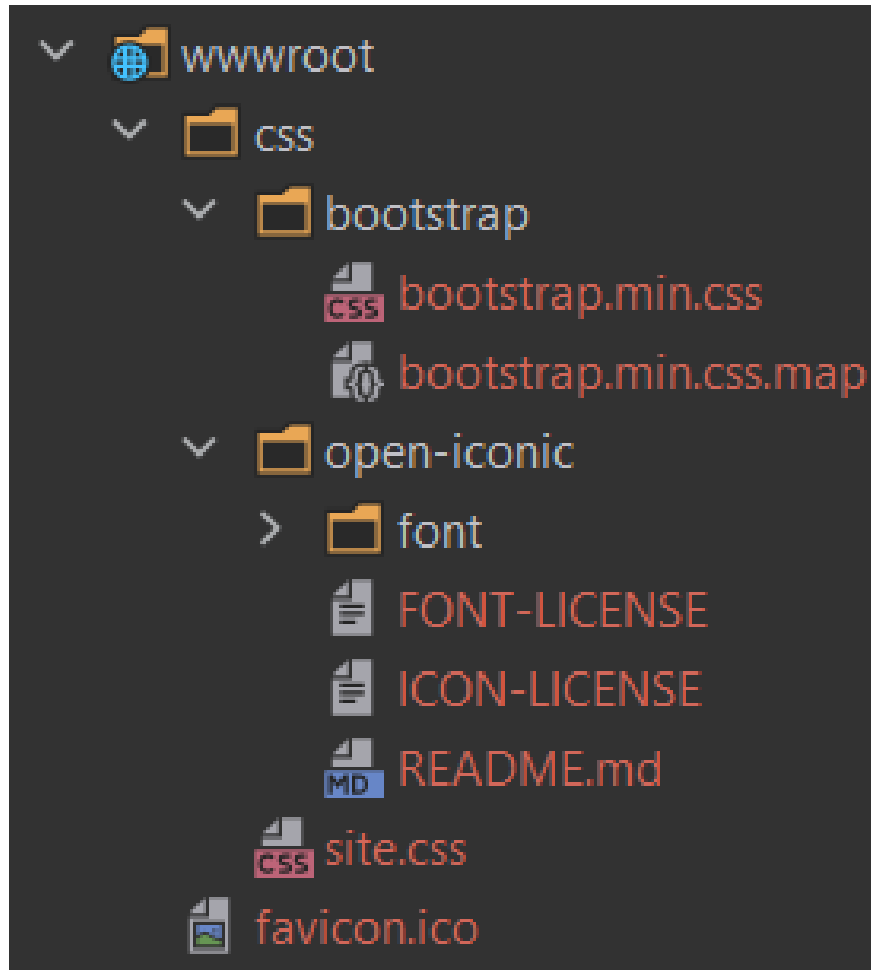
What does MVC look like?

Request → Controller

Controller
Retrieves Model
"Does Stuff"

Model

Response ←

View

View
Visually represents
the model

Blazor Server project structure
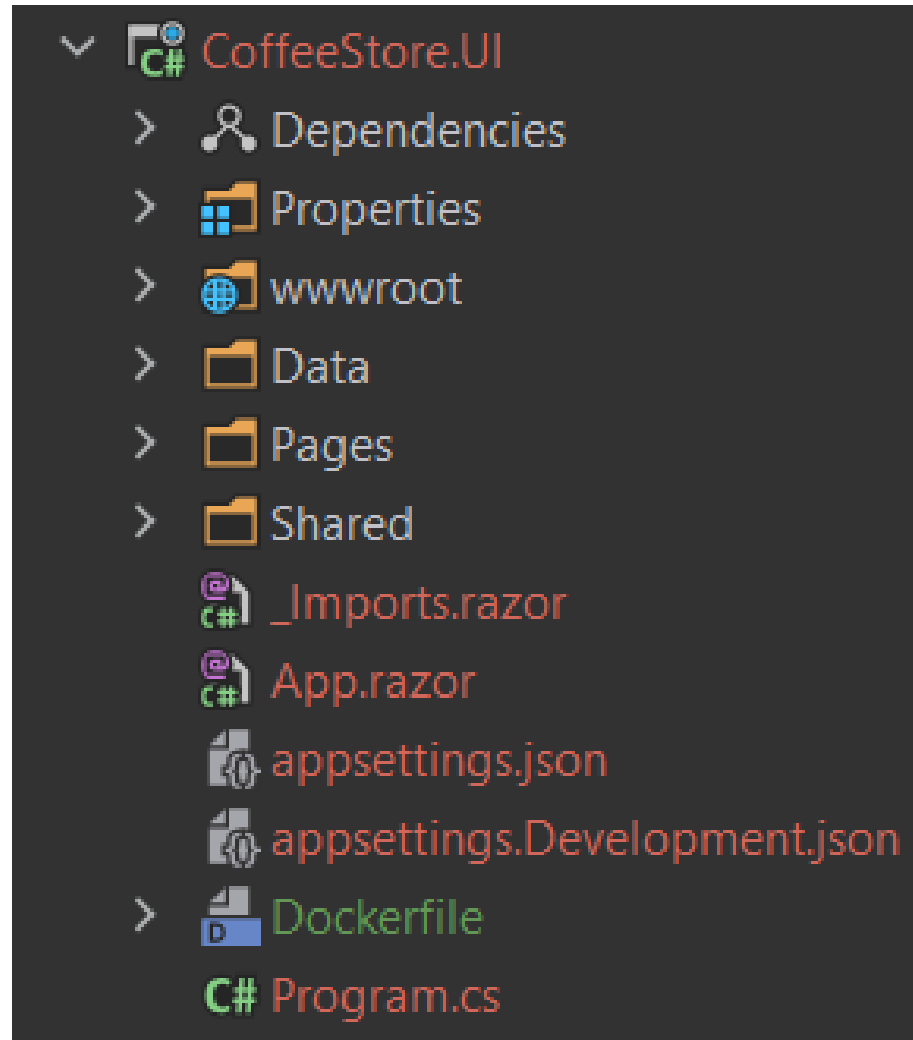
wwwroot

Data
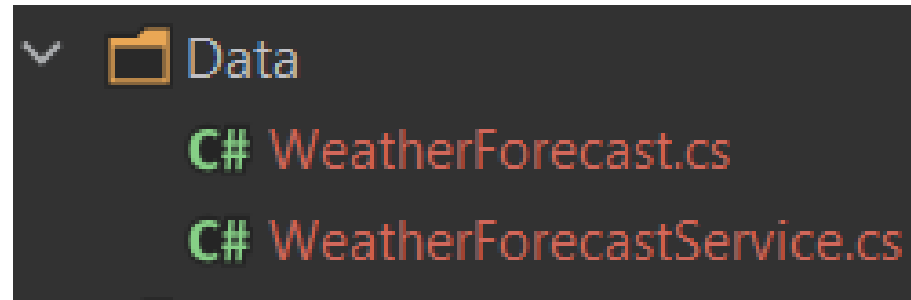
Pages

Shared

Layouts

```razor
@inherits LayoutComponentBase

<PageTitle>CoffeeStore.UI</PageTitle>

<div class="page">
    <div class="sidebar">
        <NavMenu/>
    </div>

    <main>
        <div class="top-row px-4">
            <a href="https://docs.microsoft.com/aspnet/" target="_blank">About</a>
        </div>

        <article class="content px-4">
            @Body
        </article>
    </main>
</div>
```
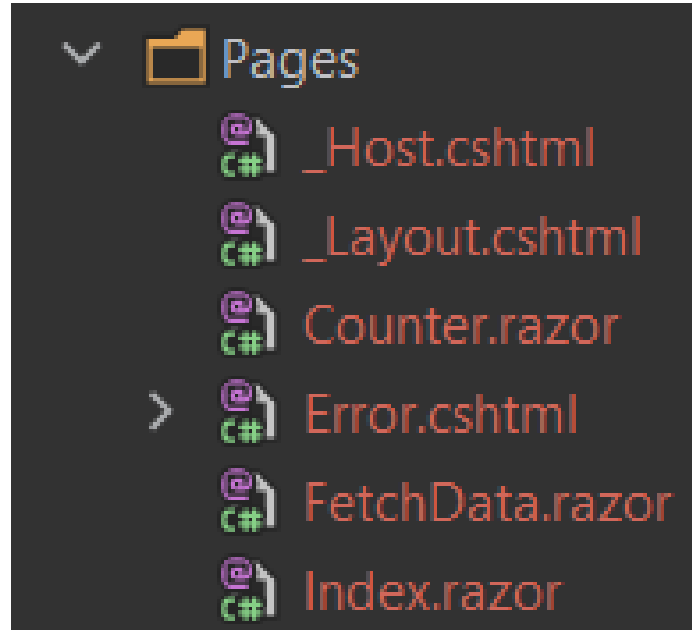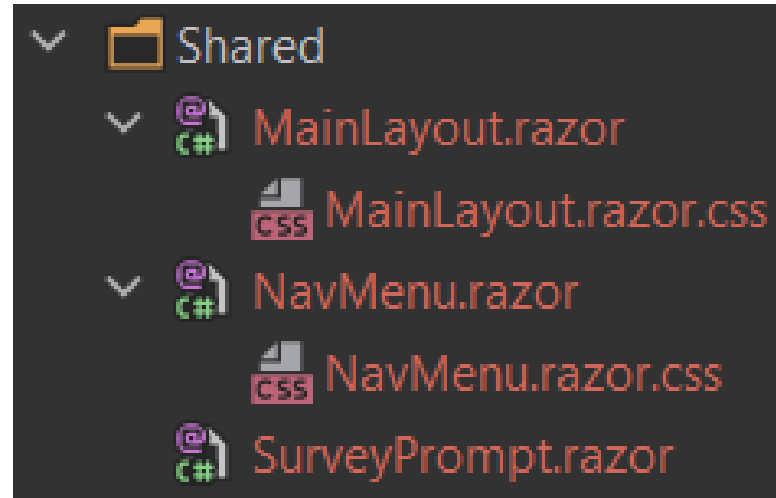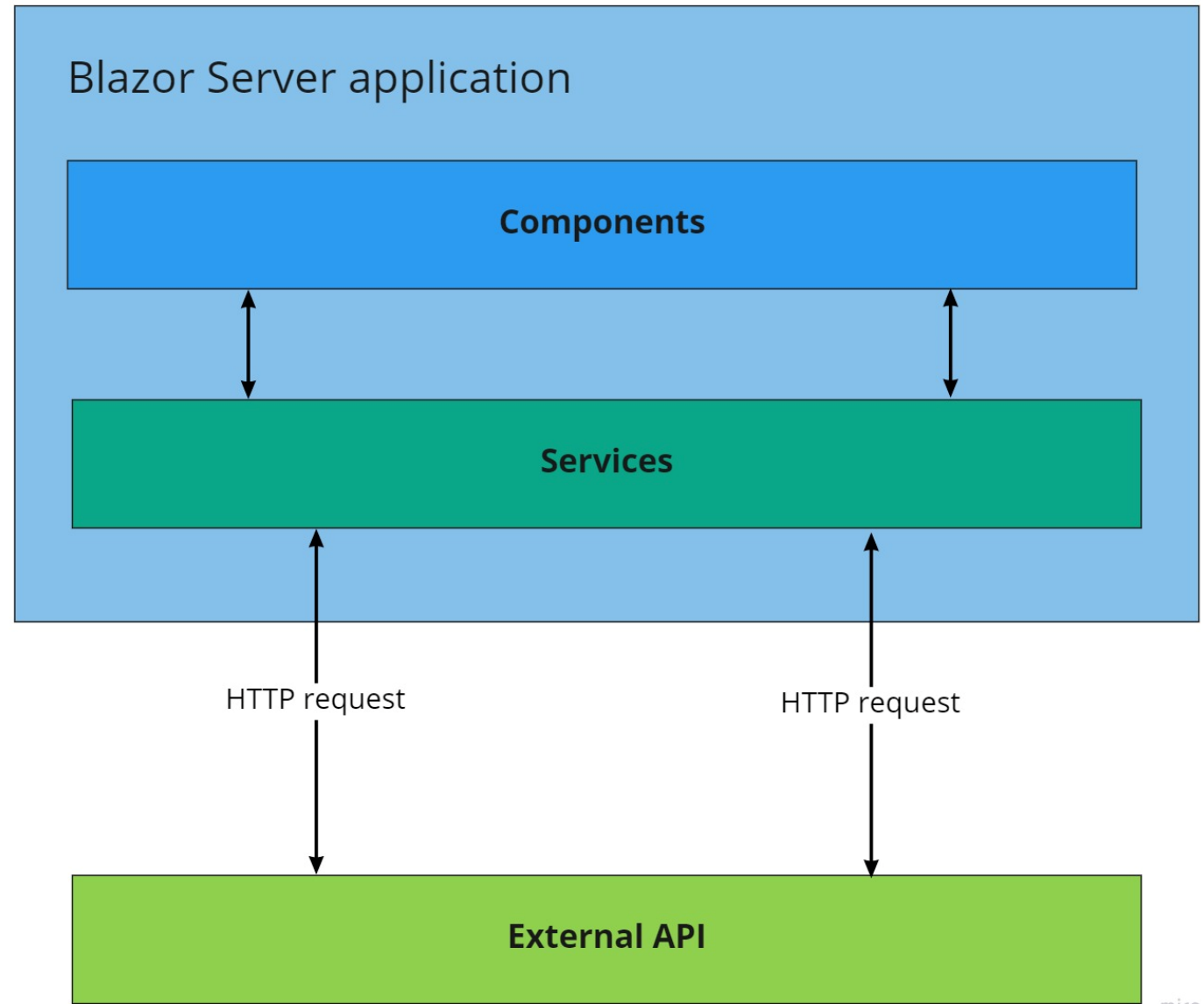
Interaction with the backend

Blazor Server application

Components

Services

HTTP request HTTP request

External API

miro

Books of the day

Freeman A. – Pro ASP.NET MVC 5

Banks A., Porcello E. – Learning React, 2e: Modern Patterns for Developing React Apps

Litvinavicius T. - Exploring Blazor. Creating Hosted, Server-side, and Client-side Applications with C#

Links of the day

MVC vs MVP vs MVVM (Habr)

Introducing Blazor (Habr)

Blazor docs (MSDN)

Blazorise Documentation

.NET MAUI (MSDN)

Hometask

Nothing today. Hooray!

That's all for this time!