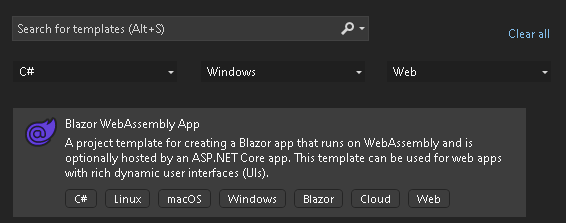
Blazor webapplikáció

* [Blazor University - What is Blazor? (blazor-university.com)](https://blazor-university.com/overview/what-is-blazor/)
* [Blazor alkalmazásüzemeltési modellek - .NET | Microsoft Learn](https://learn.microsoft.com/hu-hu/dotnet/architecture/blazor-for-web-forms-developers/hosting-models)
* [Blazor WebAssembly Series - Ultimate Guide (code-maze.com)](https://code-maze.com/blazor-webassembly-series/)

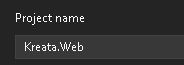
Készítsünk először egy blazor frontend alkalmazást. A backend (server) rész is része lehet, de később is hozzáadhatjuk.

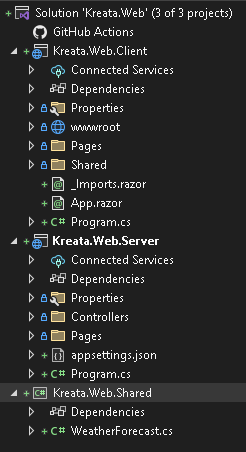
Az alkalmazásunk egy elképzelt „Kréta” iskolai adminisztrációs program lesz.

Válasszuk ki a projektet:



Alkalmazásunk viselje a Kreta.Web nevet:



Az itteni kódok .Net 7.0 verzióban készültek.

# Elkészült a projekt

A projekt most Client, Server és Shared projekteket is tartalmaz. Ezeket majd később kicseréljük a backend órákon megírt kódra.

A Shared projektben olyan osztályok kerülnek, amelyek a frontend és backend oldalon is szükségesek. Ezt a részt is majd később fejlesztjük.

# MudBlazor csomag

* [MudBlazor - Blazor Component Library](https://mudblazor.com/)

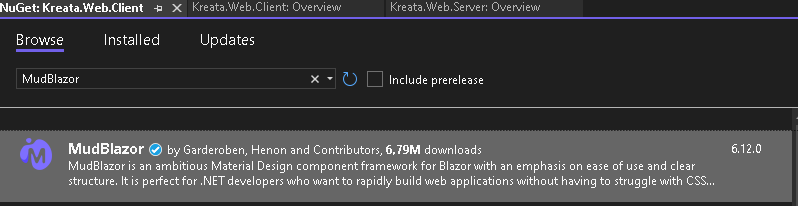
Az első cél a MudBlazor használva egy modernebb weboldal elkészítése.

Használjuk a MudBlazor telepítésével kapcsolatos leírást:

* [Installation - MudBlazor](https://mudblazor.com/getting-started/installation)

Adjuk hozzá a Kreata.WebClient projekthez a Mudblazor NuGet csomagot. A verzió száma lehet mindig az aktuális legnagyobb.

MudBlazor NuGet csomag hozzáadása:



## Kövessük a telepítési leírást

* Az \_Imports.razor fájlba azokat a névtereket töltjük be amelyek minden razor oldalon használni szeretnénk

@using MudBlazor

* Ezután a stílusok és fontok hozzáadása következik az index.html fájlba

<link href="https://fonts.googleapis.com/css?family=Roboto:300,400,500,700&display=swap" rel="stylesheet" />

<link href="\_content/MudBlazor/MudBlazor.min.css" rel="stylesheet" />

* A blazor java script kódok betöltése

<script src="\_content/MudBlazor/MudBlazor.min.js"></script>

* Regisztráljuk a szükséges szolgáltatást (Program.cs)

builder.Services.AddMudServices();

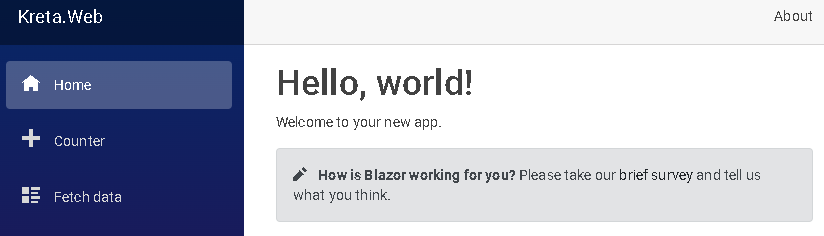
* A MudBlazor szükséges komponenseit betöltjük (MainLayout)

<MudThemeProvider/>

<MudDialogProvider/>

<MudSnackbarProvider/>

A weboldalunk jelenlegi elrendezése:



## MudBlazor weboldal elrendezések

* [Layouts - MudBlazor](https://mudblazor.com/getting-started/layouts#basic-layout)
* [Wireframes - MudBlazor](https://mudblazor.com/getting-started/wireframes#custom-content)

A MudBlazor weboldalán különböző elrendezésekre látunk lehetőséget, ezek közül válaszunk!

A következők közül választhatunk még:

* [App Bar - MudBlazor](https://mudblazor.com/components/appbar#api) (app sáv)

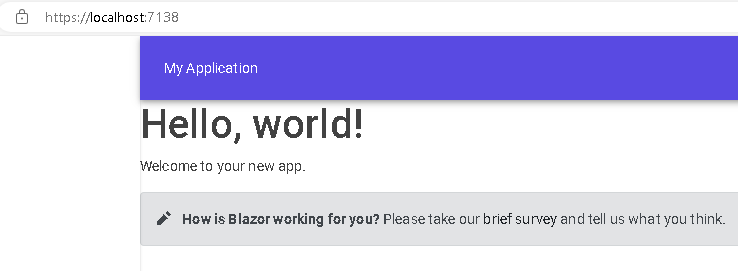
Az alkalmazássáv a műveletek, a márkajelzés, a navigáció és címek megjelenítésére szolgál. Ha az alkalmazássávot a különböző oldalak böngészése közben állandóan jelen van, megkönnyíti a navigációt és a műveletekhez való hozzáférést a felhasználók számára.

* [Drawer - MudBlazor](https://mudblazor.com/components/drawer#usage)

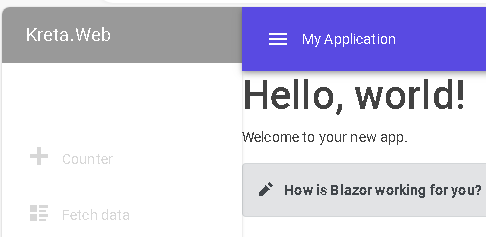
A navigációs ablak, hozzáférést biztosít az alkalmazásban található menüpontokhoz és tartalmakhoz.

### Különböző nézetek és a webapp felépítések

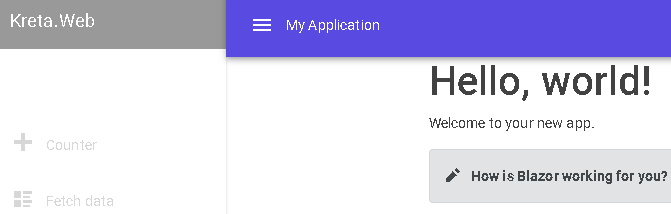
* Drawer és AppBar



* Eltüntethető drawer menüpontokkal



* [Container - MudBlazor](https://mudblazor.com/components/container#fluid)



* Elkészíthetjük a menu-t is a MudBlazorral

