



# Формирование страниц при помощи библиотеки компонентов MudBlazor

## Overview

Каждому хочется иметь красивый сайт или приложение, но не каждому сделать таковым это будет под силу. Для таких людей существуют готовые библиотеки компонентов, которые уже стилизованы и готовы к использованию.

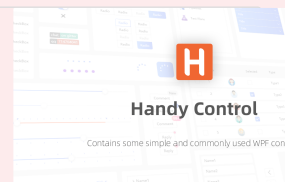


Такие библиотеки вы можете найти для любой платформы, например для WPF можно использовать следующие пакеты, чтобы сделать интерфейс более современным:

GitHub - HandyOrg/HandyControl: Contains some simple and commonly used WPF controls

Contains some simple and commonly used WPF controls - GitHub - HandyOrg/HandyControl: Contains some simple and commonly used WPF controls

<https://github.com/HandyOrg/HandyControl>



Material Design In XAML Toolkit

Material Design for WPF and the Windows Desktop

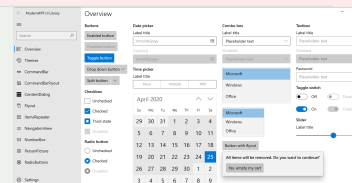
<http://materialdesigninxaml.net/>



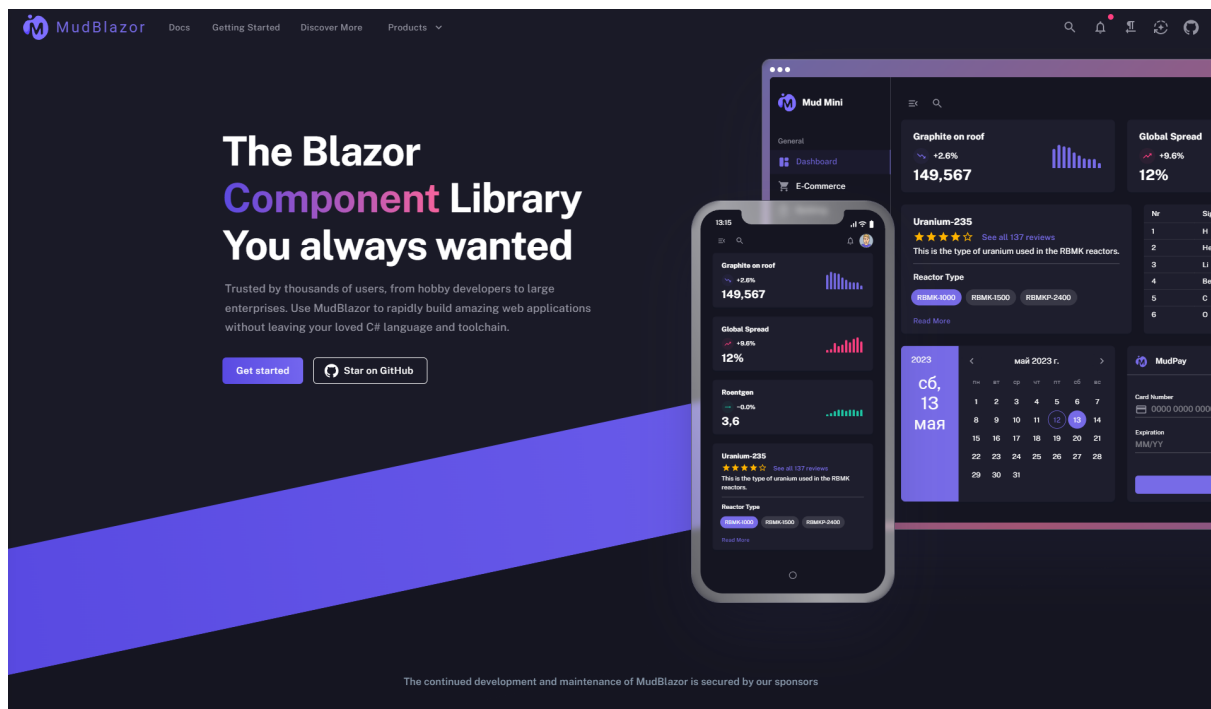
GitHub - Kinnara/ModernWpf: Modern styles and controls for your WPF applications

Modern styles and controls for your WPF applications - GitHub - Kinnara/ModernWpf: Modern styles and controls for your WPF applications

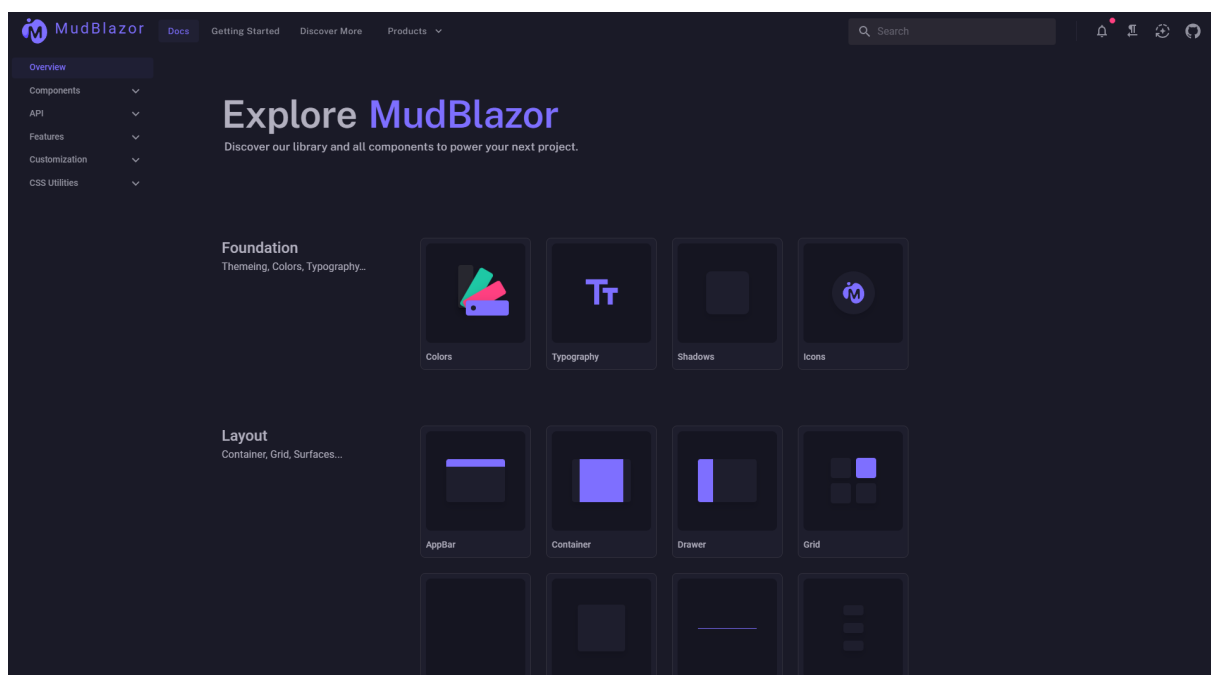
<https://github.com/Kinnara/ModernWpf>



Так как наше веб-приложение основано на платформе Blazor, то наиболее интересной библиотекой компонентов будет MudBlazor.



**MudBlazor** — одна из самых популярных библиотек для приложений под Blazor. Во многом из-за количества компонентов и красоты их реализации. Попробовать все компоненты вы можете их документации:



#### MudBlazor - Blazor Component Library

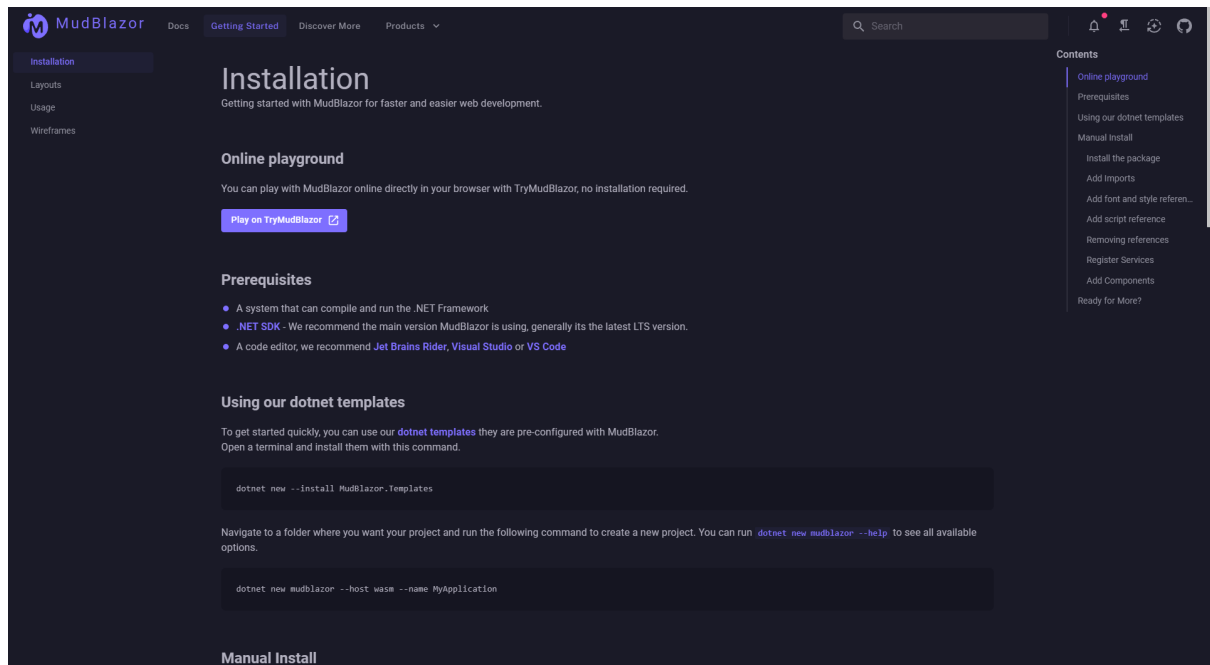
Blazor Component Library based on Material Design. MudBlazor is easy to use and extend, especially for .NET devs because it uses almost no Javascript.

 <https://mudblazor.com/docs/overview>




# Установка и применение библиотеки

Приступим к установке данной библиотеки. В документации MudBlazor есть специальный раздел с руководством по установке данной библиотеки, выполните необходимые действия для корректной работы данного пакета.



## MudBlazor - Blazor Component Library

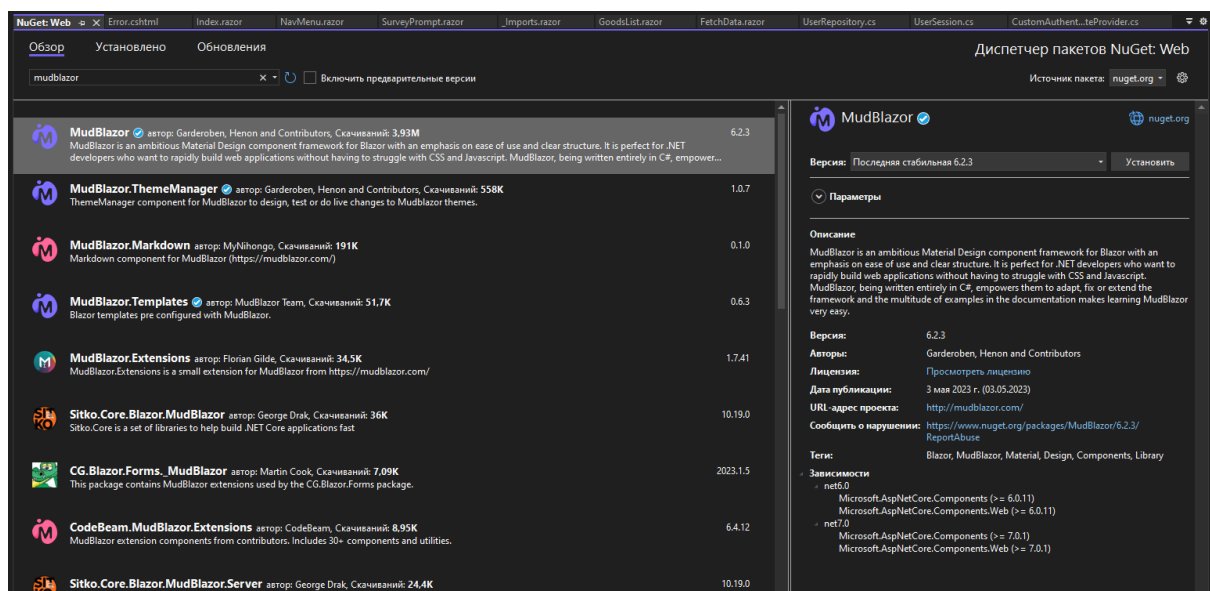
Blazor Component Library based on Material Design. MudBlazor is easy to use and extend, especially for .NET devs because it uses almost no Javascript.

 <https://mudblazor.com/getting-started/installation#online-playground>

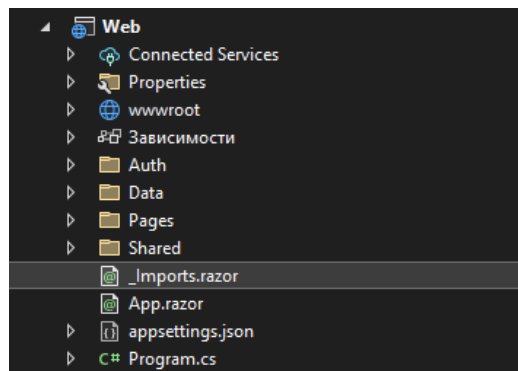


Так как мы используем платформу Blazor Server, то выполним следующие шаги:

В диспетчере пакетов Nuget установите пакет MudBlazor для вашего проекта Blazor.



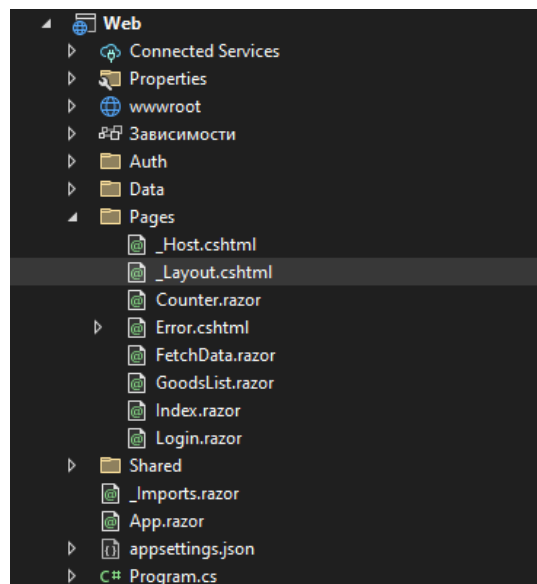
Откройте файл `_Imports.razor`



И добавьте директиву `using` для импорта пакета `MudBlazor`

```
1  @using System.Net.Http
2  @using Microsoft.AspNetCore.Authorization
3  @using Microsoft.AspNetCore.Components.Authorization
4  @using Microsoft.AspNetCore.Components.Forms
5  @using Microsoft.AspNetCore.Components.Routing
6  @using Microsoft.AspNetCore.Components.Web
7  @using Microsoft.AspNetCore.Components.Web.Virtualization
8  @using Microsoft.JSInterop
9  @using Web
10 @using Web.Shared
11 @using Domain.Interfaces
12 @using Domain.Models
13 @using MudBlazor
```

Перейдите в файл `Layout.cshtml`



И добавьте эти строки кода для подключения таблиц стилей пакета `MudBlazor`

```

1  @using Microsoft.AspNetCore.Components.Web
2  @namespace Web.Pages
3  @addTagHelper *, Microsoft.AspNetCore.Mvc.TagHelpers
4
5  <!DOCTYPE html>
6  <html lang="en">
7  <head>
8      <meta charset="utf-8" />
9      <meta name="viewport" content="width=device-width, initial-scale=1.0" />
10     <base href="/" />
11     <link rel="stylesheet" href="css/bootstrap/bootstrap.min.css" />
12     <link href="css/site.css" rel="stylesheet" />
13     <link href="Web.styles.css" rel="stylesheet" />
14     <link href="https://fonts.googleapis.com/css?family=Roboto:300,400,500,700&display=swap" rel="stylesheet" />
15     <link href="_content/MudBlazor/MudBlazor.min.css" rel="stylesheet" />
16     <component type="typeof(HeadOutlet)" render-mode="ServerPrerendered" />
17 </head>
18 <body>
19     @RenderBody()
20
21     <div id="blazor-error-ui">
22         <environment include="Staging,Production">
23             An error has occurred. This application may no longer respond until reloaded.
24         </environment>
25         <environment include="Development">
26             An unhandled exception has occurred. See browser dev tools for details.
27         </environment>
28         <a href="" class="reload">Reload</a>
29         <a class="dismiss">X</a>
30     </div>
31
32     <script src="_framework/blazor.server.js"></script>
33 </body>
34 </html>
35

```

Для того, чтобы работали различные скрипты данного пакета, добавьте ниже строчку кода для подключения файла со скриптами

```

1  @using Microsoft.AspNetCore.Components.Web
2  @namespace Web.Pages
3  @addTagHelper *, Microsoft.AspNetCore.Mvc.TagHelpers
4
5  <!DOCTYPE html>
6  <html lang="en">
7  <head>
8      <meta charset="utf-8" />
9      <meta name="viewport" content="width=device-width, initial-scale=1.0" />
10     <base href="/" />
11     <link rel="stylesheet" href="css/bootstrap/bootstrap.min.css" />
12     <link href="css/site.css" rel="stylesheet" />
13     <link href="Web.styles.css" rel="stylesheet" />
14     <link href="https://fonts.googleapis.com/css?family=Roboto:300,400,500,700&display=swap" rel="stylesheet" />
15     <link href="_content/MudBlazor/MudBlazor.min.css" rel="stylesheet" />
16     <component type="typeof(HeadOutlet)" render-mode="ServerPrerendered" />
17 </head>
18 <body>
19     @RenderBody()
20
21     <div id="blazor-error-ui">
22         <environment include="Staging,Production">
23             An error has occurred. This application may no longer respond until reloaded.
24         </environment>
25         <environment include="Development">
26             An unhandled exception has occurred. See browser dev tools for details.
27         </environment>
28         <a href="" class="reload">Reload</a>
29         <a class="dismiss">X</a>
30     </div>
31
32     <script src="_framework/blazor.server.js"></script>
33     <script src="_content/MudBlazor/MudBlazor.min.js"></script>
34 </body>
35 </html>
36

```

Перейдите в файл Program.cs и добавьте данную строчку кода

```

        builder.Services.AddSingleton<WeatherForecastService>();

        builder.Services.AddScoped<AuthenticationStateProvider, CustomAuth>();
        builder.Services.AddScoped<ProtectedSessionStorage>();
        builder.Services.AddScoped<ProtectedLocalStorage>();

        builder.Services.AddMudServices();

        var app = builder.Build();

```

В файле MainLayout.razor добавьте строчки кода, которые активируют функционал для работы:

- Менеджера тем
- Диалоговых окон
- Навигационной панели

```

1  @using Web.Auth;
2  @inherits LayoutComponentBase
3
4  @inject AuthenticationStateProvider authStateProvider
5  @inject NavigationManager navManager
6
7  <PageTitle>Web</PageTitle>
8
9  <MudThemeProvider />
10 <MudDialogProvider />
11 <MudSnackbarProvider />
12
13 <div class="page">
14     <div class="sidebar">
15         <NavMenu />
16     </div>
17
18     <main>

```

Основные этапы подготовки были завершены

## Задание:

При помощи документации MudBlazor:

1. Создайте страницу профиля пользователя
2. Оформите главную страницу (Файл Index.razor) и наполните его в соответствии с тематикой вашего магазина