

FastReport .NET for Blazor

Версия 2021.2 Beta

В вашем распоряжении пакет FastReport.Web.Blazor с компонентами для встраивания в ваше веб-приложение. Данная библиотека основана на технологии [Blazor Server](#) (server-side) и означает, что все операции будут выполняться на стороне сервера, который будет затем передавать клиенту готовое отображение.

Минимальная целевая платформа (target framework) на данный момент .Net Core 3.1 для обеспечения максимально возможной совместимости с последней на данный момент LTS (long-term support) версией с длительной поддержкой. Также данная версия есть у большинства пользователей и она совместима с последним фреймворком .NET 5 (в рамках данного пакета).

Настройки перед использованием

Для использования FastReport.Web.Blazor вам необходимо добавить зависимость в ваш файл проекта (csproj) PackageReference с указанием id данного пакета и пакета FastReport.Core (версии могут отличаться):

```
<ItemGroup>
  <PackageReference Include="FastReport.Core" Version="2021.2.11-demo"/>
  <PackageReference Include="FastReport.Web.Blazor" Version="2021.2.11-demo"/>
</ItemGroup>
```

Затем, для упрощения наименования, советуем добавить следующие namespace в импорты вашего проекта (файл _Imports.razor):

```
@using FastReport.Web
@using FastReport.Web.Blazor
@using FastReport.Web.Blazor.Components
@using FastReport.Web.Blazor.Components.Internal
```

На самом деле, хватит только добавления FastReport.Web.Blazor.Components, однако, для некоторых случаев вам, возможно, понадобятся и другие namespace. Также некоторые компоненты, вероятно, будут перемещаться в рамках этих namespace во время стадии beta.

В конфигураторе вашего веб приложения необходимо вызвать метод UseFastReport с необязательным лямбда выражением настройки FastReportOptions. Также, для работы некоторых встроенных общих стилей и svg изображений иконок в Toolbar и Tab, необходимо использовать вызов UseStaticFiles (если вы не собираетесь использовать Toolbar и Tabs, вызов UseStaticFiles для использования данного пакета необязателен):

```
public void Configure(IApplicationBuilder app, IWebHostEnvironment env)
{
```

```
// ...
app.UseStaticFiles();
// ...
app.UseFastReport();
}
```

Описание встроенных компонентов

Далее идёт описание компонентов, входящих в состав FastReport.Web.Blazor. Мы не рекомендуем использование компонентов, отличных от WebReportContainer, т.к. они могут вести себя нестабильно вне стандартного дерева компонентов. Однако, для более точной настройки отображения, вам может потребоваться использование других компонентов.

WebReportContainer

Основной и самый универсальный Blazor компонент, выполняющий отрисовку WebReport-а, является **<WebReportContainer/>**. Он находится в namespace FastReport.Web.Blazor.Components. Единственный параметр, который он принимает - объект класса WebReport. Это означает, что для использования данного компонента, вы должны создать объект класса WebReport, присвоить ему отчёт (Report), другие необходимые параметры и передать данный объект в параметры **WebReportContainer**.

Пример:

```
<WebReportContainer WebReport="@UserWebReport" />

@code {
    public WebReport UserWebReport { get; set; }

    protected override void OnParametersSet()
    {
        var report = Report.FromFile(
            Path.Combine(
                directory,
                "My report.frx"));

        // Registers the application dataset
        Report.RegisterData(DataSet, "NorthWind");

        UserWebReport = new WebReport();
        UserWebReport.Report = Report;
    }
}
```

Данный компонент умеет определять различный Mode (Designer, Dialog и обычный Preview). Умеет подготавливать отчет, встраивать стили, определенные по умолчанию, и

индивидуальные стили, отображать Toolbar, Outline и Tabs, работать с интерактивными отчетами и др.

WebReportPreview

Аналогичен предыдущему компоненту, однако не учитывает Designer Mode. Т.е. всегда пытается подготовить и отрисовать отчёт.

ReportContainer

Аналогичен предыдущему компоненту, однако не включает загрузку стилей WebReport-а (общих и индивидуальных для Toolbar/Tabs/Outline).

Занимается подготовкой отчёта и последующим отображением вместе с Toolbar и Tabs (по необходимости).

При работе какой-либо интерактивности (клики по отчёту/работа с диалоговыми формами) обновляется именно этот компонент.

ReportBody / ExportComponent

ReportBody вызывает отрисовку Outline (по необходимости) и “вкладывает” в себя компонент, который является отрисовкой самого отчёта (ExportComponent), который ему передаёт ReportContainer. Не рекомендуется для использования.

BlazorExport

Самым “нижним” уровнем компонента является вовсе не компонент, а сам BlazorExport - инструмент экспорта подготовленного отчёта в формат построения [RenderTreeBuilder](#). Находится в FastReport.Web.Blazor.Export namespace.

Для построения данного экспорта необходимо:

- 1) Подготовить данный отчёт
- 2) Быть уверенным, что в этом отчёте не используются диалоговые формы (они отображаются с помощью DialogPageComponent и не рассматриваются в данном руководстве)
- 3) Создать свой компонент и в нём явно определить способ построения (вызвать переопределение метода BuildRenderTree)
- 4) В этом методе построения создать экземпляр BlazorExport, задать ему необходимые свойства и вызвать Export с передачей следующих параметров: отчёта (Report) и экземпляр builder, являющимся аргументом данного переопределенного метода.

```
/// Main function
protected override void BuildRenderTree(RenderTreeBuilder builder)
{
    using (BlazorExport blazor = new BlazorExport())
    {
```

```
        blazor.StylePrefix = $"fr{WebReport.ID}";
        blazor.EmbedPictures = true;
        blazor.OnClick += ProcessClick;
        blazor.EnableMargins = WebReport.EnableMargins;
        blazor.SinglePage = true;
        blazor.CurPage = WebReport.CurrentPageIndex;

        blazor.Export(myReport, builder);
    }
}
```

Online Designer

В данный момент Online Designer умеет работать в элементе iframe средствами javascript и он полностью совместим со сборкой Online Designer для Core, однако, вызов Preview пока не работает.

Для использования только возможностей дизайнера, вы можете вызвать компонент **<IFrameDesigner/>** с передачей ему параметра WebReport с настроенным свойством Report и необязательными DesignerLocale и DesignerPath:

```
<IFrameDesigner WebReport="CurrentWebReport" />
```

Однако напоминаем, компонент WebReportContainer понимает, с каким Mode он в данный момент работает и вовсе не обязательно вызывать IFrameDesigner именно в таком виде.

Настройка общих стилей и SVG

В отличие от FastReport.Web for Core, SVG изображения для Toolbar и Tabs, а также некоторые общие стили отображения Tabs, Outline и др. были вынесены в staticWebAssets для возможной настройки в вашем веб-приложении (изменения цвета, размеров, замены изображения).

Располагаются данные ресурсы на вашем локальном хранилище. В момент разработки/сборки вашего приложения они находятся по адресу:
{UserName}/.nuget/packages/fastreport.web.blazor/{version}/staticwebassets

В момент публикации вашего веб-приложения (dotnet publish) эти ресурсы копируются в директорию: wwwroot/_content/FastReport.Web.Blazor

Демо проект

Проект для демонстрации работы с пакетом FastReport.Web.Blazor вы можете найти [на нашем GitHub](#). Пример использования **WebReportContainer** находится в пользовательском компоненте по пути Pages/Index.razor и Pages/Index.razor.cs.