Preparing for Client-side Blazor



Gill Cleeren CTO XPIRIT BELGIUM @gillcleeren www.snowball.be



Overview



Client-side Blazor

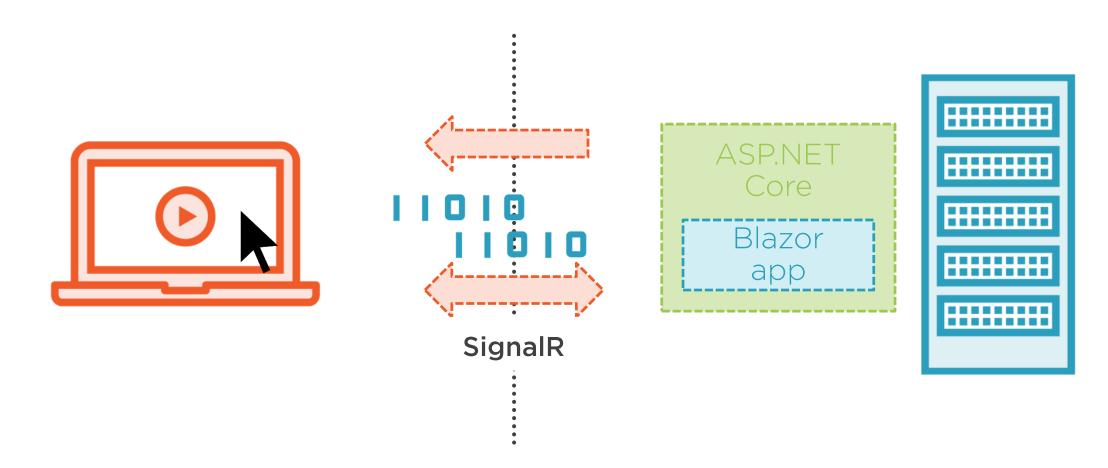
Converting the app to client-side Blazor



Client-side Blazor



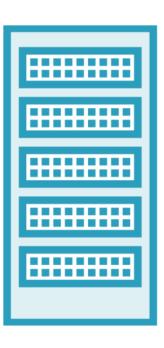
Server-side Blazor Architecture



Client-side Blazor Architecture





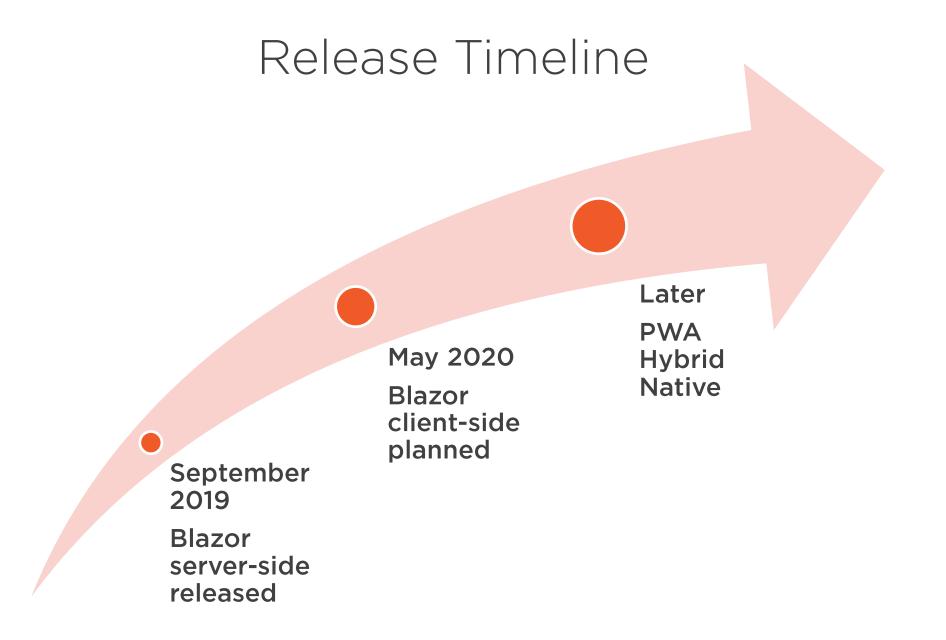




Blazor Hosting Models

Server-side Client-side







Server-side to Client-side

Server-side Blazor app



Client-side Blazor app



Converting the App to Client-side Blazor





Start preparing

- Ensure your app can be easily switched
- Extract what is different
- Share most of the code

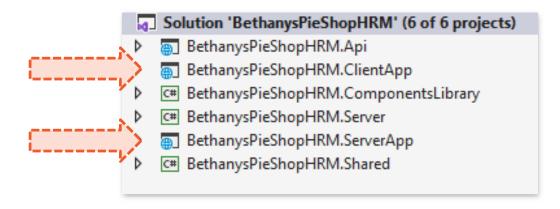


Things to Consider

HttpClient **EF Core REST APIS**



Proposed Approach





Server-side Project

■ BethanysPieShopHRM.ServerApp

Connected Services

Dependencies

Properties

www.root

Pages

Host.cshtml

Appsettings.json

C# Program.cs

C# Startup.cs



Client-side Project

■ BethanysPieShopHRM.ClientApp

C Connected Services

Dependencies

Properties

www.root
index.html

C Program.cs

Startup.cs



C# BethanysPieShopHRM.Server Dependencies Properties Components Pages EmployeeDetail.razor ▶ C# EmployeeDetailBase.cs EmployeeEdit.razor ▶ C# EmployeeEditBase.cs EmployeeOverview.razor ▶ C# EmployeeOverviewBase.cs Error.razor Index.razor Services Shared www.root _Imports.razor App.razor

App Project

- _Host.cshtml is removed
- Project is now a Razor Class Library
- Code can be fully reused



Demo



Extracting the server-side specific files

Adding a client-side project

Converting our original project



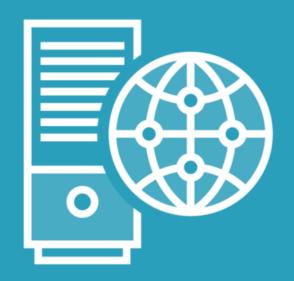
Summary



Server-side Blazor projects can easily be converted to client-side by following some patterns

Makes app-building with Blazor even more interesting





Up next:

Deploying the application

