There are many ways to structure and name projects and solutions, for example, using a folder hierarchy as well as a naming convention. If you work in a team, make sure you know how your team does it.

Structuring projects in a solution or workspace

It is good to have a naming convention for your projects in a solution or workspace so that any developer can tell what each one does instantly. A common choice is to use the type of project, for example, class library, console app, website, and so on, as shown in the following table:

Name	Description
`Northwind.Common`	A class library project for common types like interfaces, enums, classes, records, and structs, used across multiple projects.
`Northwind.Common.EntityModels`	A class library project for common EF Core entity models. Entity models are often used on both the server and client side, so it is best to separate dependencies on specific database providers.
`Northwind.Common.DataContext`	A class library project for the EF Core database context with dependencies on specific database providers.
`Northwind.Web`	An ASP.NET Core project for a simple website that uses a mixture of static HTML files and dynamic Razor Pages.
`Northwind.Razor.Component`	A class library project for Razor Pages used in multiple projects.
`Northwind.Mvc`	An ASP.NET Core project for a complex website that uses the MVC pattern and can be more easily unit tested.
`Northwind.WebApi`	An ASP.NET Core project for an HTTP API service. A good choice for integrating with websites because they can use any JavaScript library or Blazor to interact with the service.
`Northwind.OData`	An ASP.NET Core project for an HTTP API service that implements the OData standard to enable a client to control queries.
`Northwind.GraphQL`	An ASP.NET Core project for an HTTP API service that implements the GraphQL standard to enable a client to control queries.
`Northwind.gRPC`	An ASP.NET Core project for a gRPC service. A good choice for integrating with apps built with any language and platform since gRPC has wide support and is highly efficient and performant.
`Northwind.SignalR`	An ASP.NET Core project for real-time communication.
`Northwind.AzureFuncs`	An ASP.NET Core project for implementing a serverless nanoservice for hosting in Azure Functions.
`Northwind.BlazorServer`	An ASP.NET Core Blazor Server project.
`Northwind.BlazorWasm.Client`	An ASP.NET Core Blazor WebAssembly client-side project.
`Northwind.BlazorWasm.Server`	An ASP.NET Core Blazor WebAssembly server-side project.
`Northwind.Maui`	A .NET MAUI project for a cross-platform desktop/mobile app.
`Northwind.MauiBlazor`	A .NET MAUI project for hosting Blazor components with native integrations with the OS.