

# Building ASP.NET Core 7 Minimal APIs

## Introduction to ASP.NET Core Minimal APIs



**Kevin Dockx**

Architect

@Kevindockx | [www.kevindockx.com](http://www.kevindockx.com)

# Version Check



**This version was created by using:**

- ASP.NET Core 7.0.3
- C# 11
- Visual Studio 2022



# Version Check



**This course is 100% applicable to:**

- ASP.NET Core 7.x
- C# 11



# Version Check



**New course versions are regularly released:**

- <https://app.pluralsight.com/profile/author/kevin-dockx>



# Coming Up



**Course prerequisites, frameworks and tooling**

**Building APIs with ASP.NET Core**

- (Dis)advantages of minimal APIs

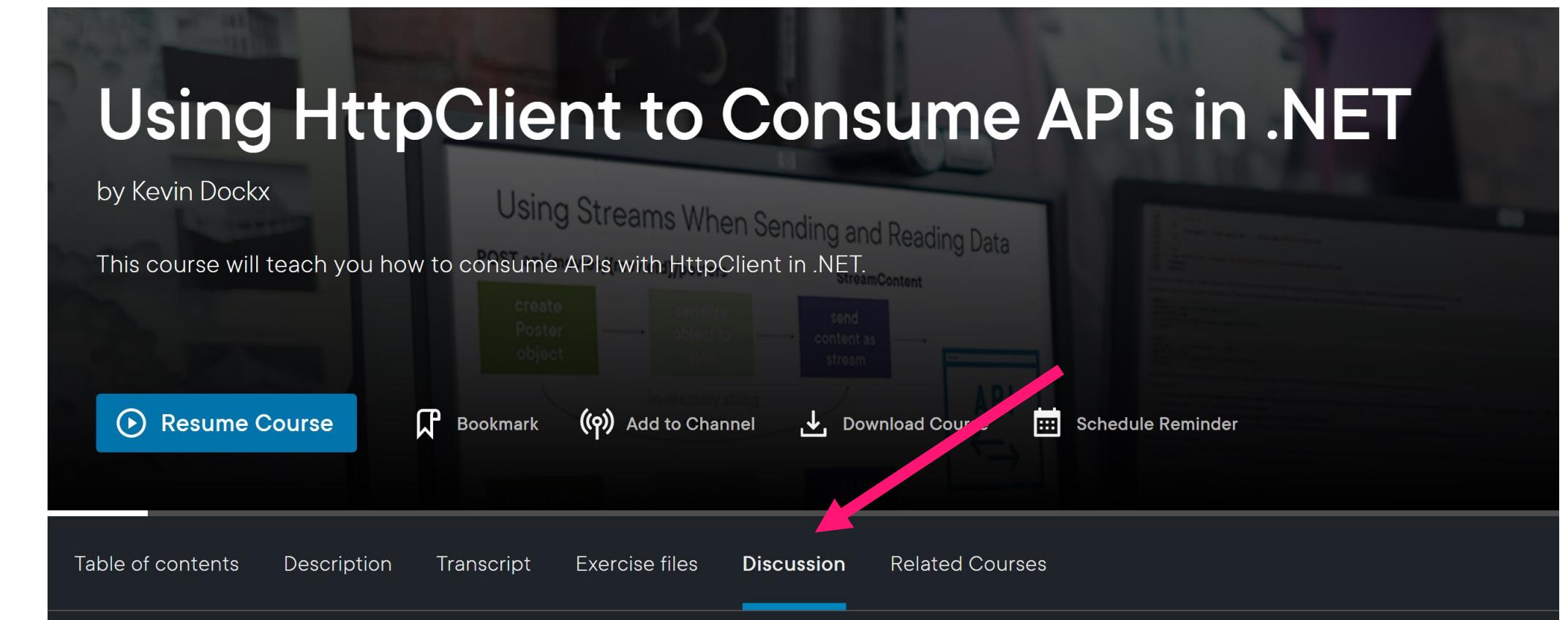
**Introducing the demo application**

- Inspecting a minimal API project
- Introducing Postman
- Adding a data layer



**Discussion tab on the course page**

**Twitter: @KevinDockx**



**(course shown is one of my other courses, not this one)**

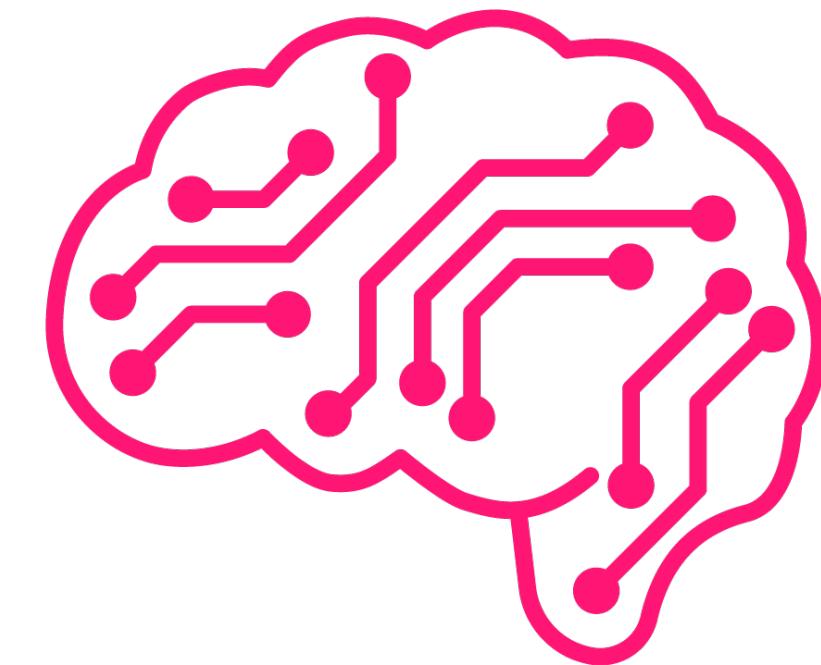
# Course Prerequisites



**Good knowledge of C#**



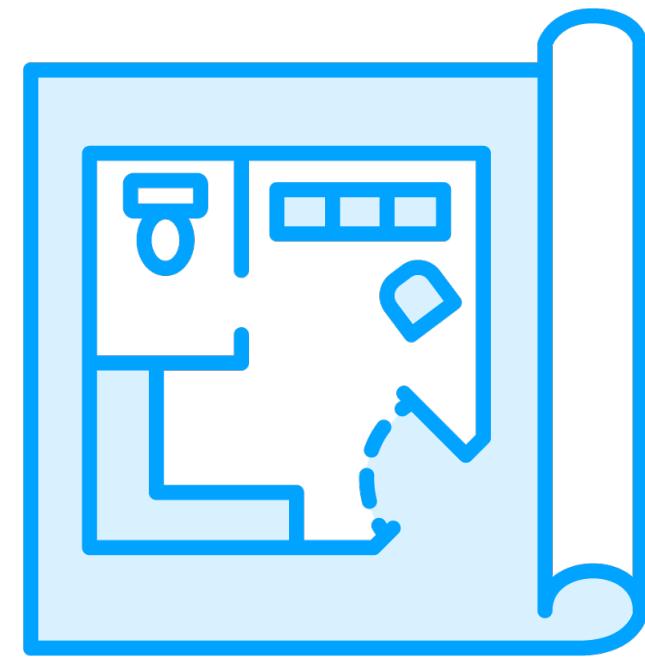
**Knowledge of  
ASP.NET Core's basic  
concepts**



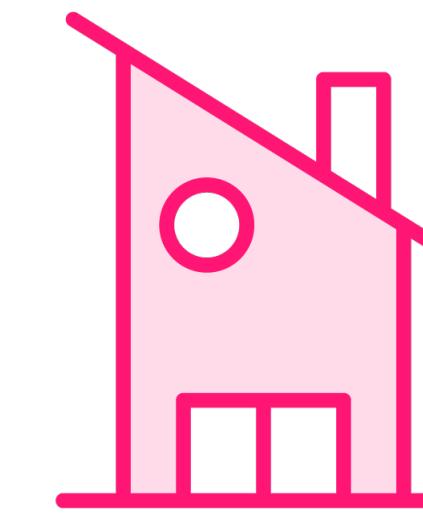
**Some knowledge of  
building an API with  
ASP.NET Core will help**



# Building APIs with ASP.NET Core



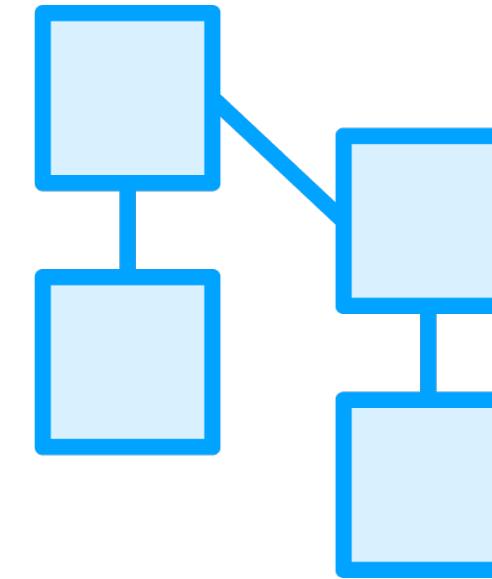
ASP.NET Core MVC



ASP.NET Core Minimal API



# Building APIs with ASP.NET Core MVC



model

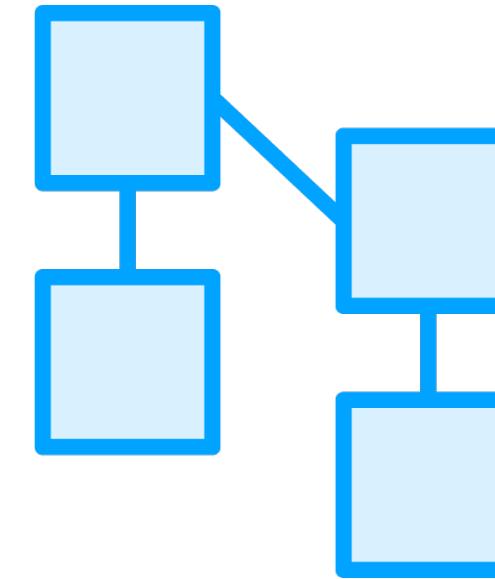


# Model

**Application data and/or business logic & rules**



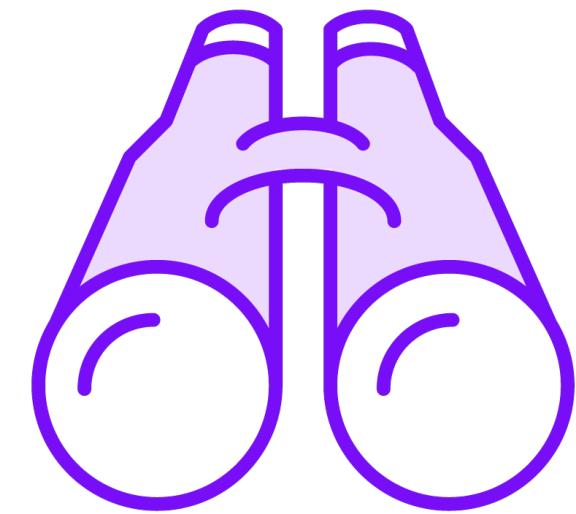
# Building APIs with ASP.NET Core MVC



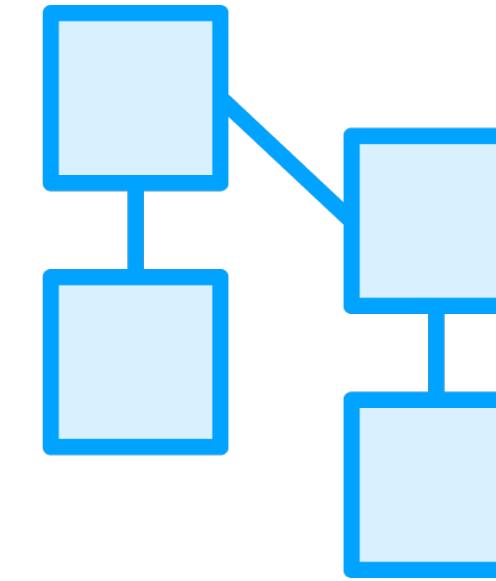
model



# Building APIs with ASP.NET Core MVC



view



model

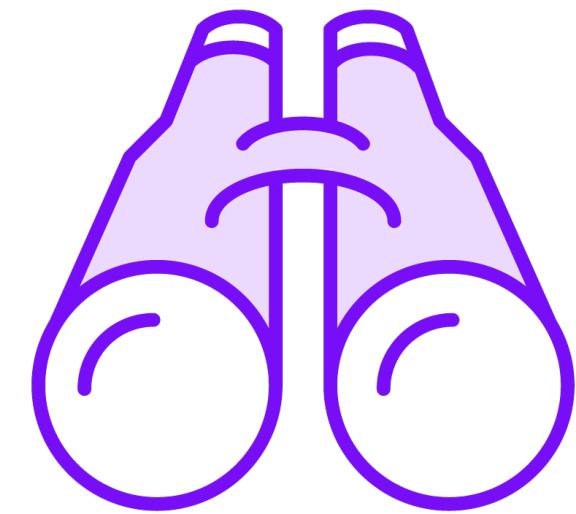


# View

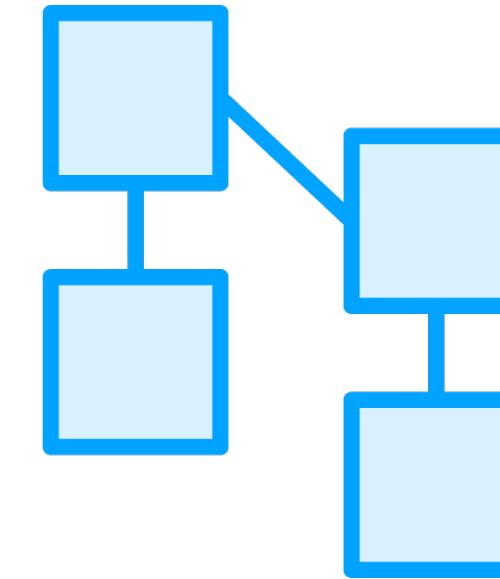
Representation of model data



# Building APIs with ASP.NET Core MVC



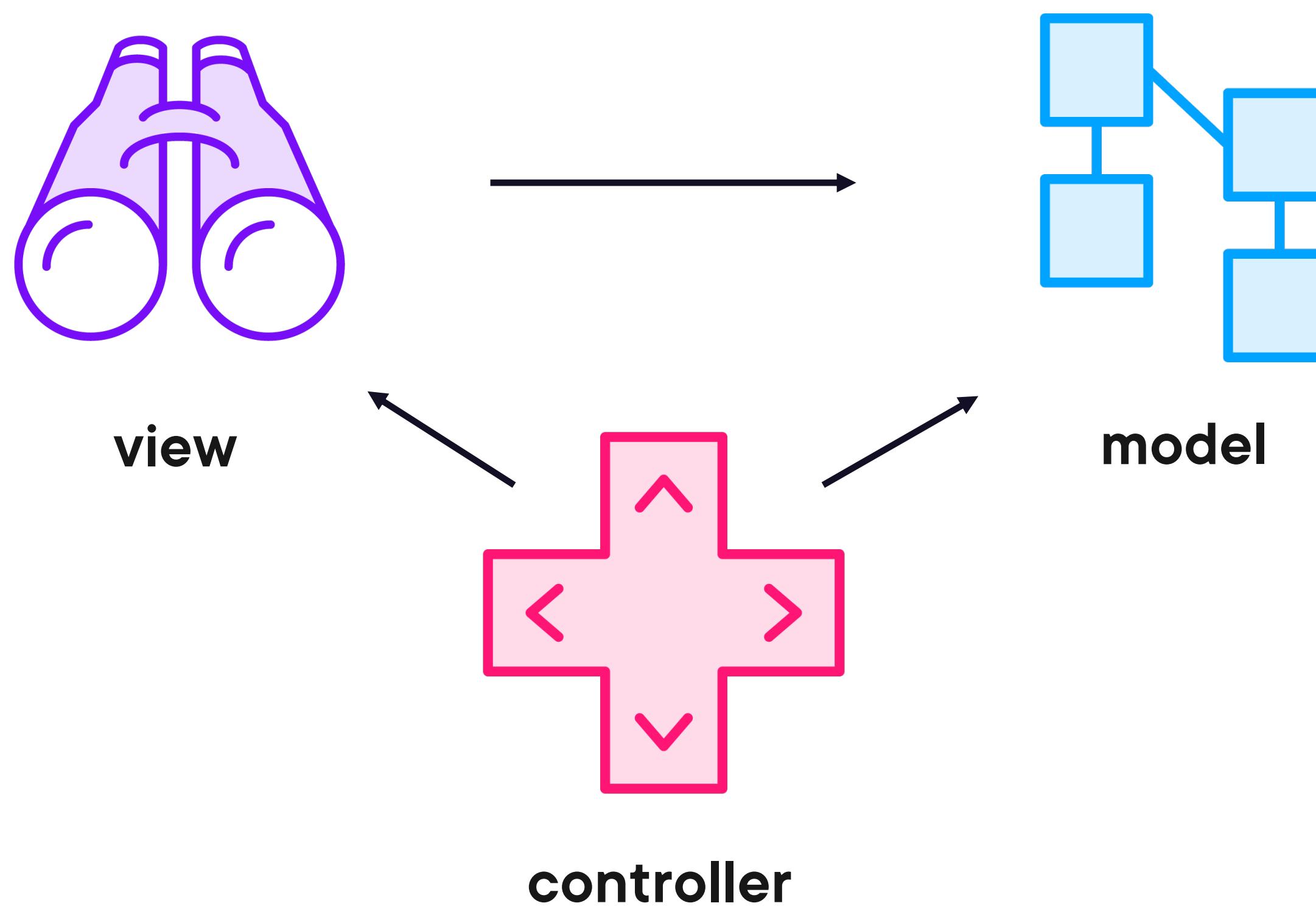
view



model



# Building APIs with ASP.NET Core MVC

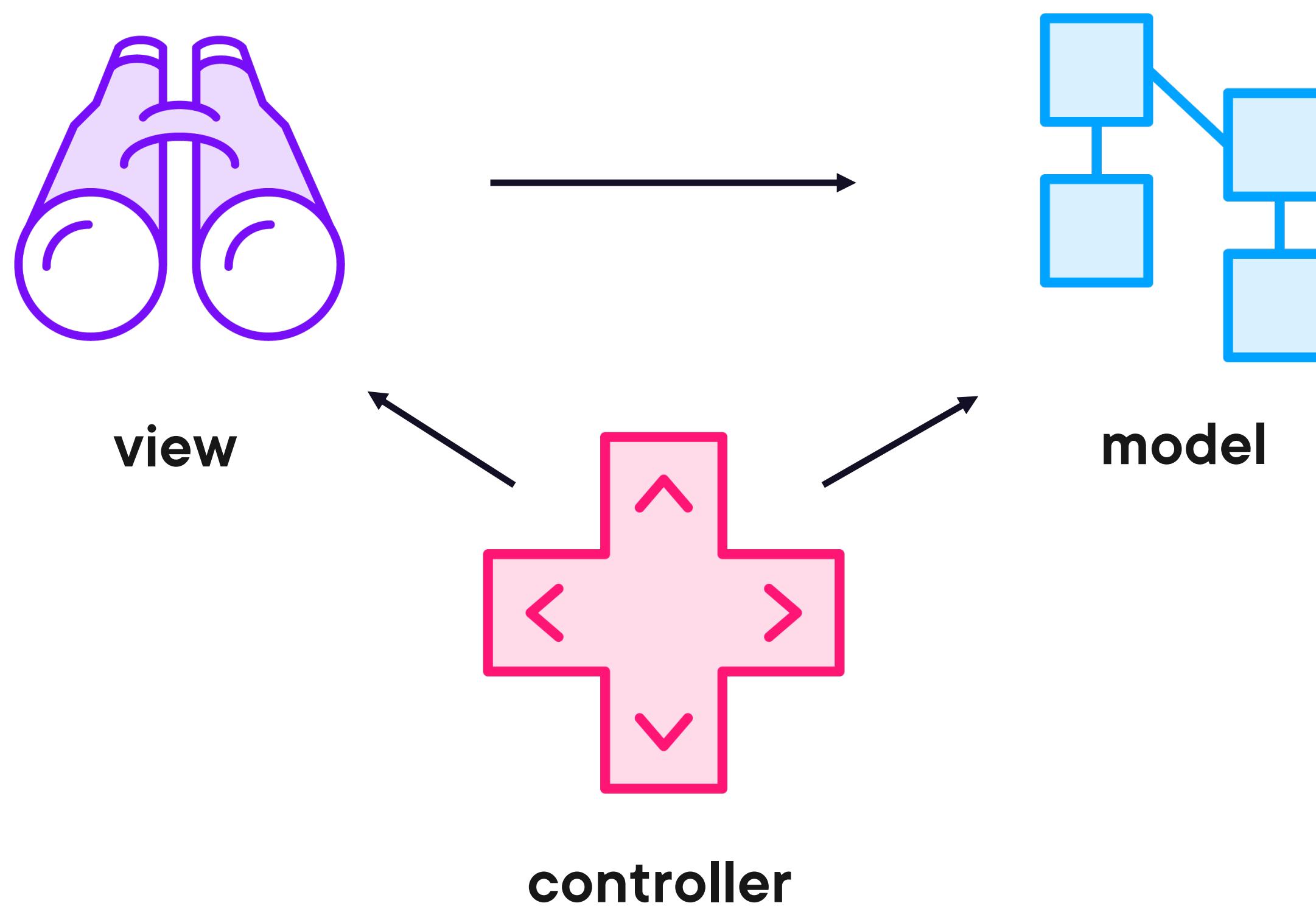


# Controller

**Enables accepting user input and translating that to model data the view can work with**



# Building APIs with ASP.NET Core MVC



# Building APIs with ASP.NET Core MVC



**Improves separation of concerns**



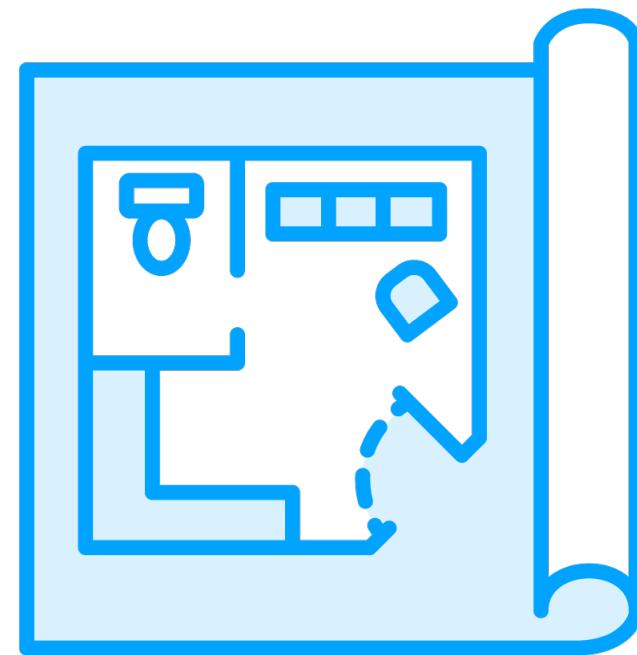
**Improves testability**



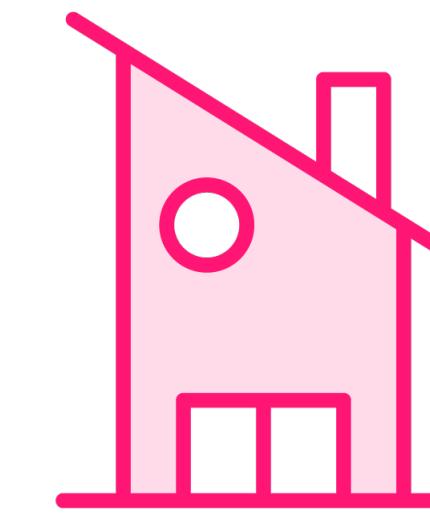
**Promotes reuse**



# Building APIs with ASP.NET Core



**ASP.NET Core MVC**  
**Fast, full-featured, proven and**  
**time-tested**



**ASP.NET Core Minimal API**  
**A bit faster, lightweight, less full-**  
**featured, relatively new**



# **Building APIs with ASP.NET Core Minimal API**

**Supports routing, parameter binding, basic serialization & basic filters**

**Intended for lightweight scenarios without all the ceremony that comes with MVC**

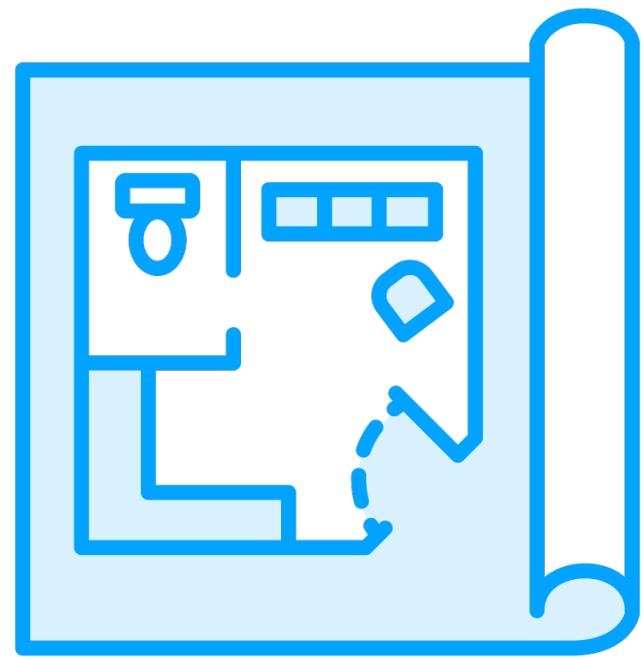
- Microservice APIs, aggregate/proxy APIs, relatively small APIs, ...**



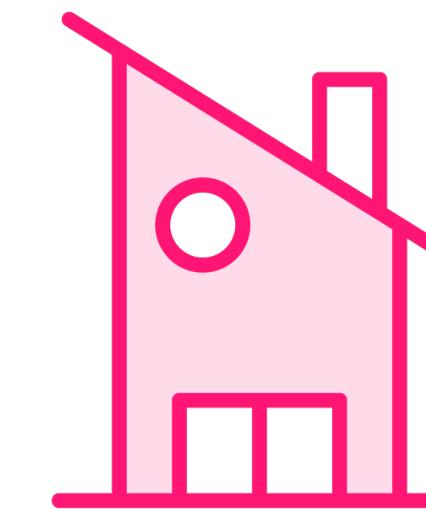
**Consider what you want to  
rely on for building your API**



# What Do You Want to Rely On?



**Built-in approach tends to lead to  
MVC approach**



**Own code or external packages  
tend to lead to minimal API  
approach**



# Introducing the Demo Application

We'll build an API that exposes dishes and their ingredients

- Start from scratch
- Dummy data is provided
- Database integration with Entity Framework Core



# Demo



**Creating and inspecting an ASP.NET Core  
minimal API project**



## Trust ASP.NET Core SSL Certificate



This project is configured to use SSL. To avoid SSL warnings in the browser you can choose to trust the self-signed certificate that ASP.NET Core has generated.

Would you like to trust the ASP.NET Core SSL certificate?

Don't ask me again

Yes

No



# Demo



## Using Postman



# Demo



## Adding the data layer



# Summary



**Different approaches for building web APIs with ASP.NET Core exist**

- Use the minimal API approach for lightweight scenarios without all the ceremony that comes with MVC

**We'll work with dishes & ingredients in this course**



**Up Next:**

# **Learning About Core Concepts and Reading Resources**

---

