

# Module 2 Day 13

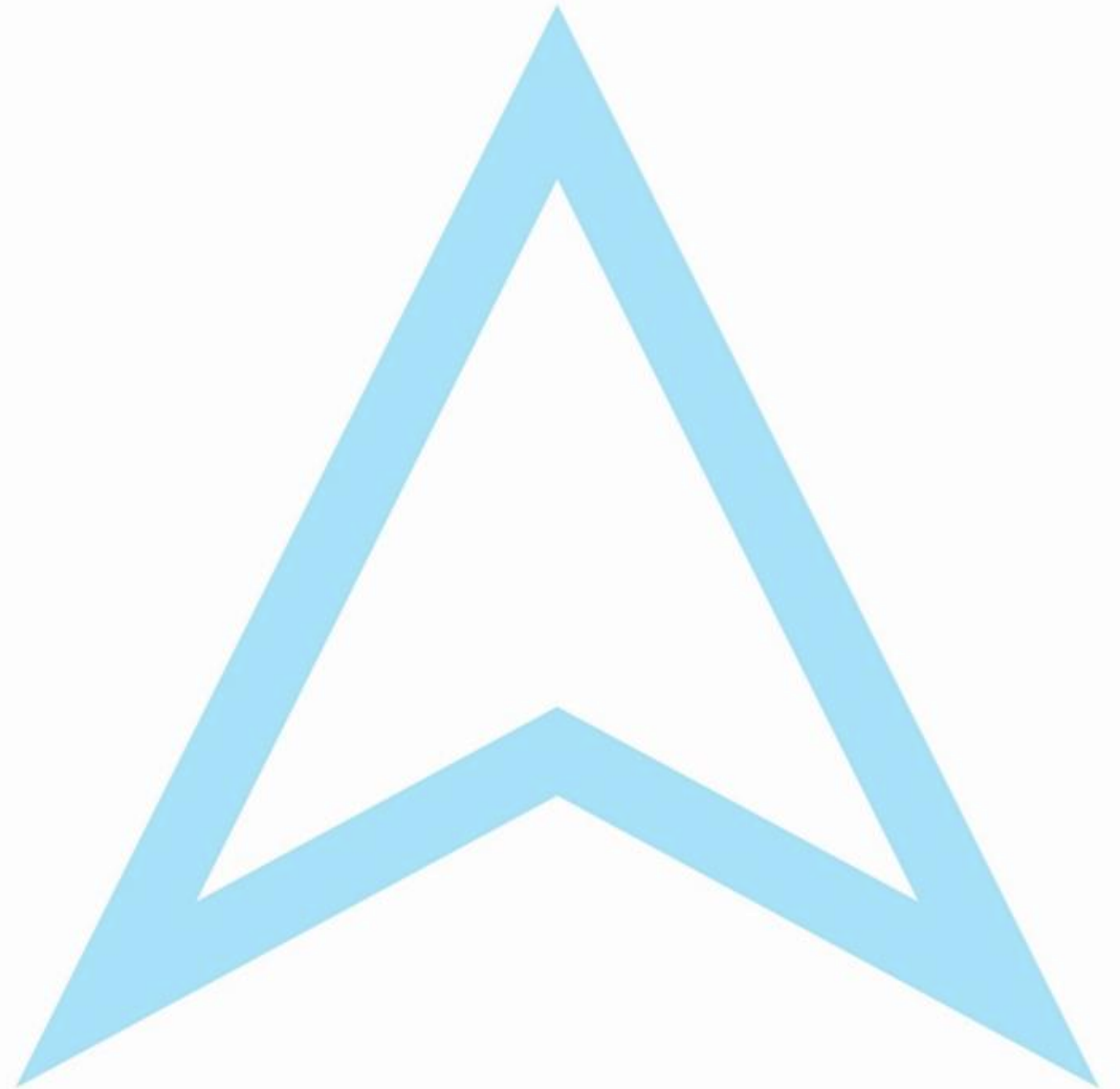
## Creating APIs - Part 1

# ASP.Net Core MVC

- Active Server Pages (ASP)
    - Framework for dynamic web applications
    - Initially released in 1996
    - ASP.Net released in 2002
    - ASP.Net Core first released 2016
      - Open-source, cross platform
  - Model-View-Controller (MVC)
    - Architectural design pattern used in 1970's and 1980's, and formalized in 1988
    - Many frameworks use this design pattern
    - Web API is based on ASP.NET MVC
- 
- [https://en.wikipedia.org/wiki/Design\\_Patterns](https://en.wikipedia.org/wiki/Design_Patterns)
  - <https://exceptionnotfound.net/tag/designpatterns/>

# MVC, classic definition

- Model
  - Business data
  - POJO
- View
  - Presents data to user
  - Presents model in different ways
  - “Knows about” Model
- Controller
  - Glue / mediator / orchestrator
  - Gets model data, selects view
  - “Knows about” Model, View



# Model

- POCO – "Plain ol' CLR object" or "Plain ol' C# object"
- Your application's data
- Properties, Methods and Data Validation
- Usually accessed using a DAO or API

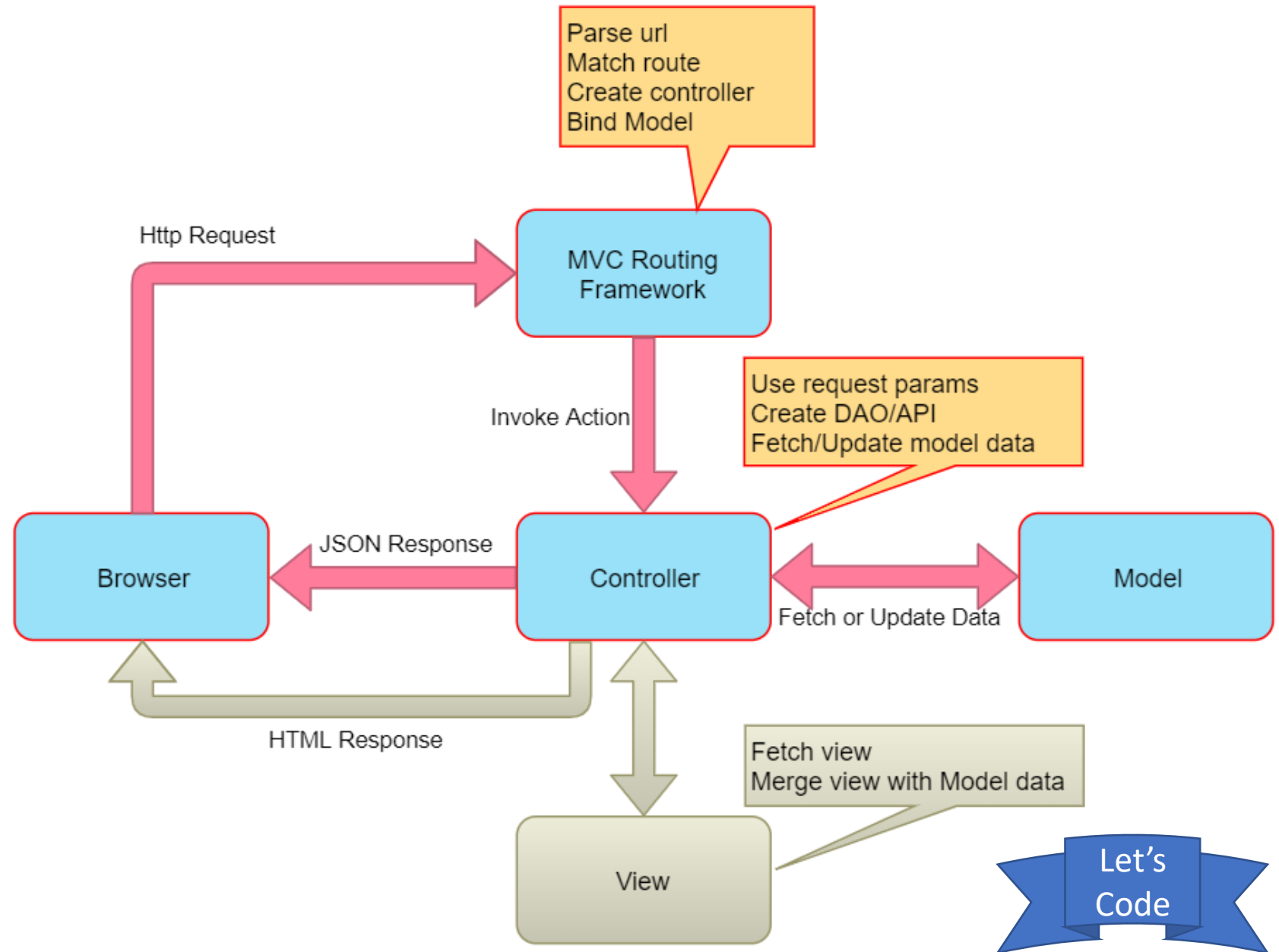
# Controller

- Inherits from ControllerBase
- Attributes
  - [ApiController]
  - [Route]
    - On class
    - On method
  - [HttpGet(sub-route)], [HttpPost], [HttpPut], [HttpDelete]
    - Can include a route or sub-route
- Actions
  - Public Methods that can be involved by clients



Let's  
Code

# MVC Workflow



# Model Binding

- Done by the framework
- Finds data in Request Body, Route, Query string (and Form fields)
- "Maps" this data to controller arguments
  - Converts data types
  - Handles case differences
  - Handles complex types



Let's  
Code