



ASP.NET Core

Day 2

**ASP.NET Core MVC –
Model, View, Controller & Actions**

Session Overview

- Concept of Middleware
- Understanding Model, View, Controller
- Creating Controller
- Understanding Actions
- Actions and Non-Actions Methods
- Understanding Action-Result
- Communication Between View & Controller

Middleware : Configuration

```
Public void Configure (IApplicationBuilder app)
```

```
{
```

```
    app.UseA();
```

A thick blue arrow points from the text 'app.UseA();' to a green box labeled 'Middleware A'.

```
    app.UseB();
```

A thick blue arrow points from the text 'app.UseB();' to a green box labeled 'Middleware B'.

```
    app.UseC();
```

A thick blue arrow points from the text 'app.UseC();' to a green box labeled 'Middleware C'.

```
}
```

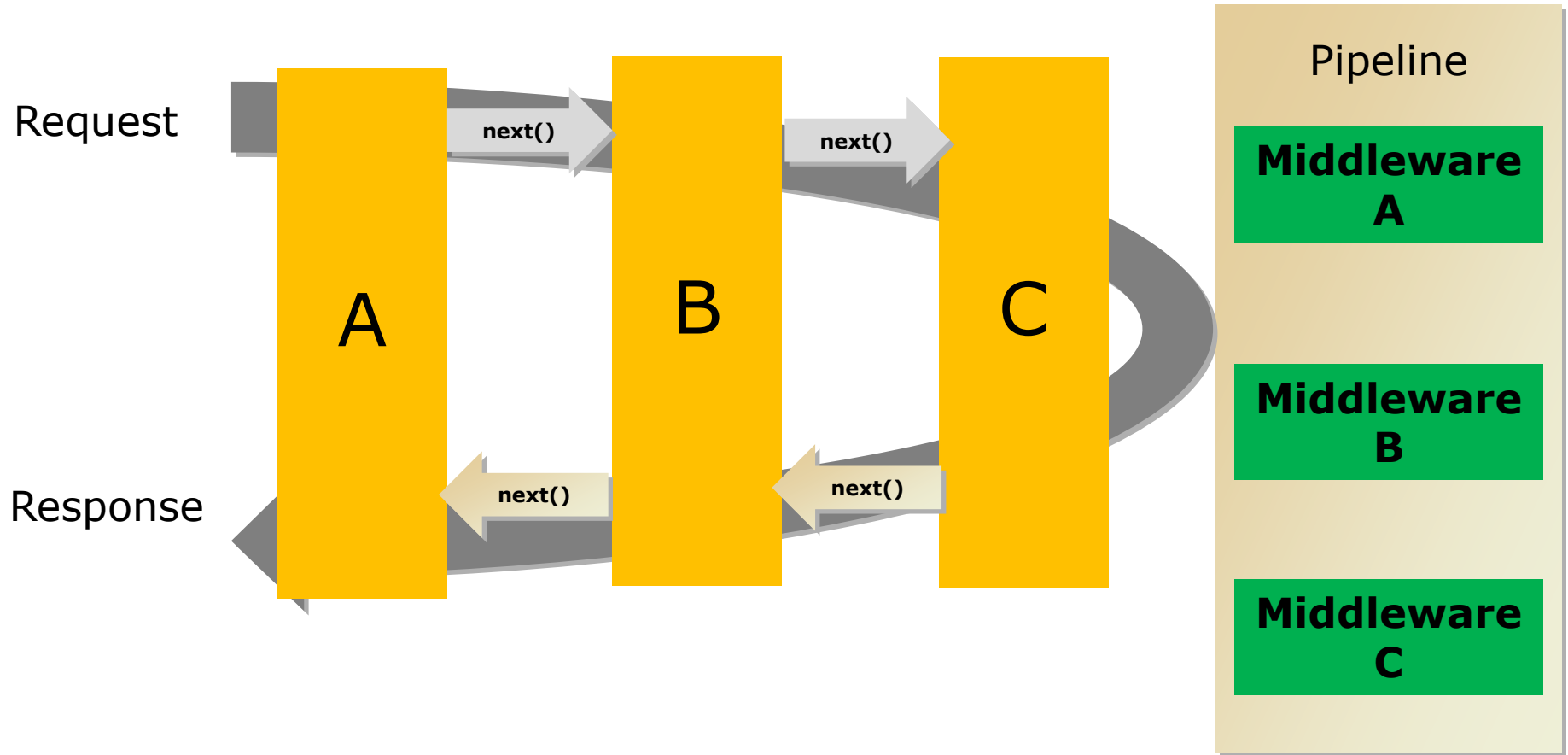
Pipeline

**Middleware
A**

**Middleware
B**

**Middleware
C**

Middleware : Request Handling



MVC Applications

- Models encapsulate objects and data
- Views generate the user interface
- Controllers interact with user actions
- Code in .cshtml and .cs files
- Precise control of HTML and URLs
- Easy to use unit tests

Model-View-Controller (MVC) Pattern

Framework

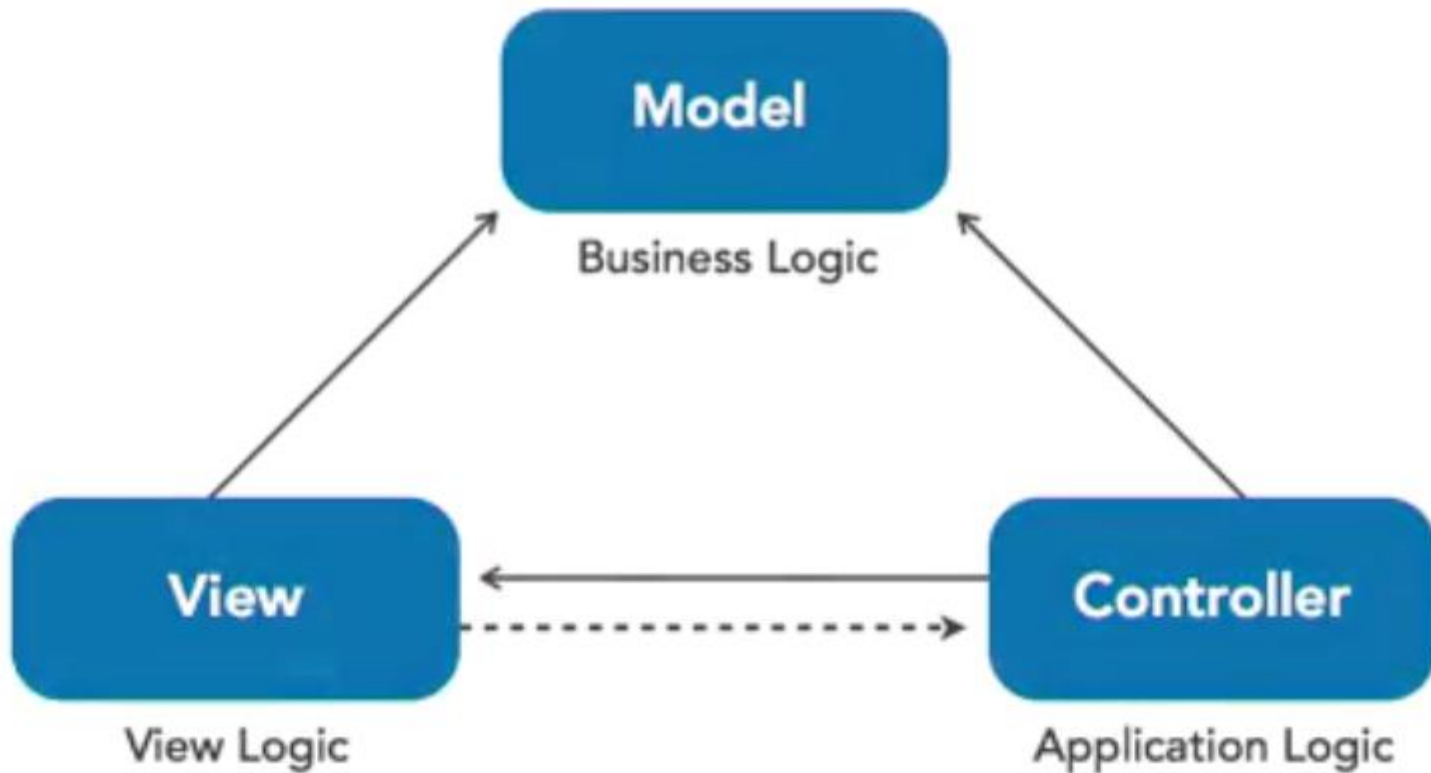
- **Handles Middleware Code**
- **Focus on Application Logic**
- **ASP.NET Core MVC Framework**

Model-View-Controller (MVC) Pattern

ASP.NET Core MVC Framework

- **Named for Model-View-Controller(MVC) Pattern**
- **User Interface Design Pattern**
- **Promotes Separation of Concerns across multiple application layer**

Model-View-Controller (MVC) Pattern



Model-View-Controller (MVC) Pattern

HTML

```
<h1>My Blog</h1>

<section>
  <h2>Middleware</h2>
  <div class="post">
    - - - - -
  </div>
  <a href="/comments/1">
    View Comments
  </a>
</section>

<section>
  <h2>Mobile First</h2>
  <div class="post">
    - - - - -
  </div>
  <a href="/comments/2">
    View Comments
  </a>
</section>
```

My Blog

Middleware

... ..
... ..

View Comments

Mobile First

... ..
... ..

View Comments

CSS

```
H1 {
  font-weight: bold;
}

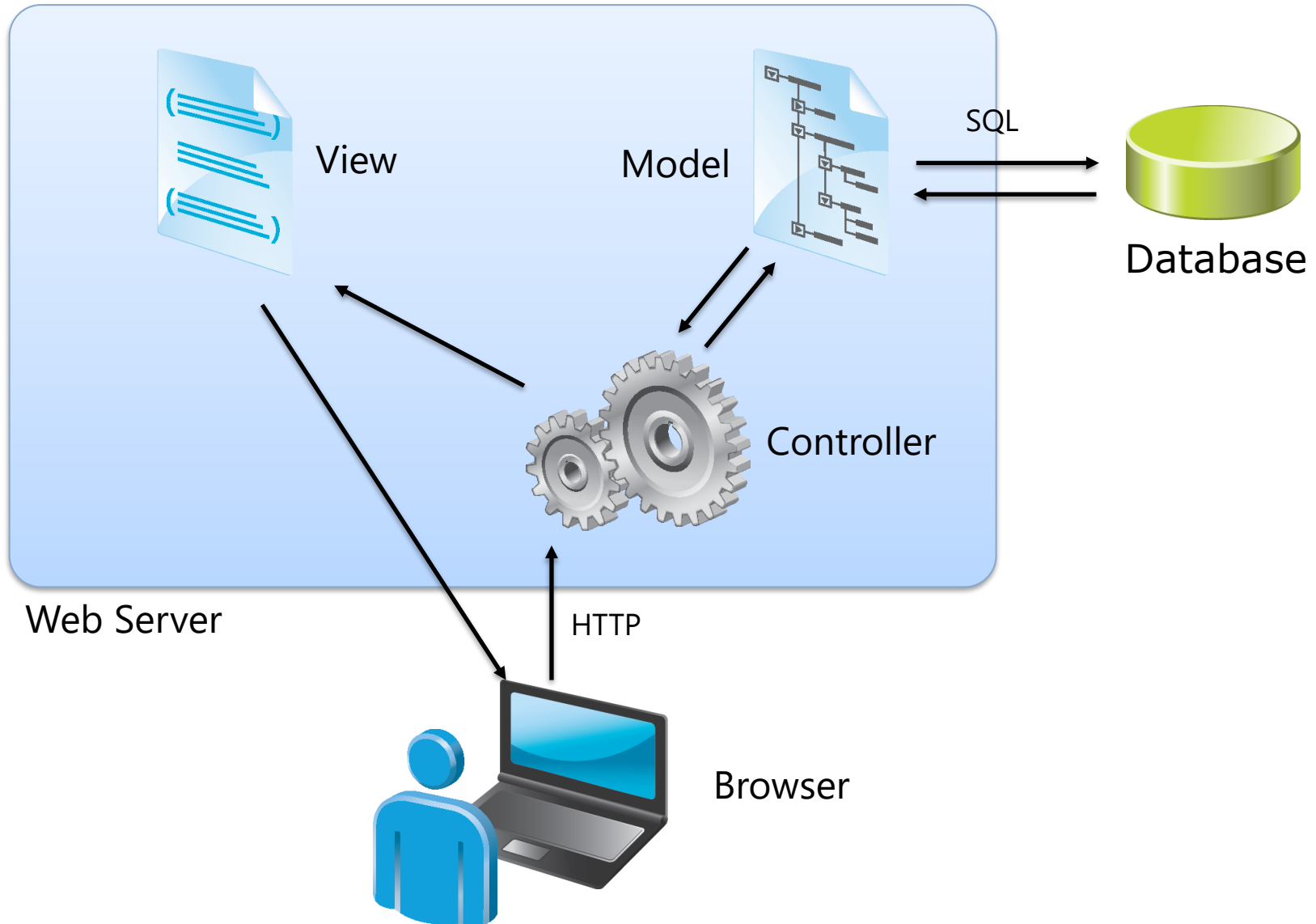
H2 {
  font-size: 110%;
}

.post {
  margin-bottom: 1em;
}
```

JavaScript

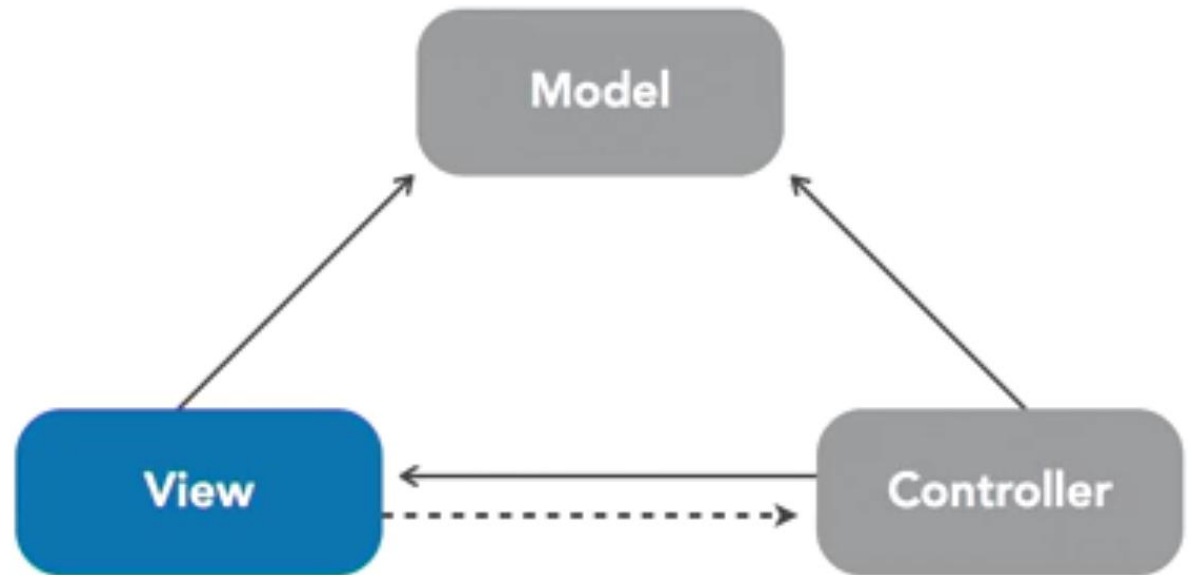
```
$(function() {
  $('.post a')
    .on('click', loadComments)
})
```

Models, Views, and Controllers



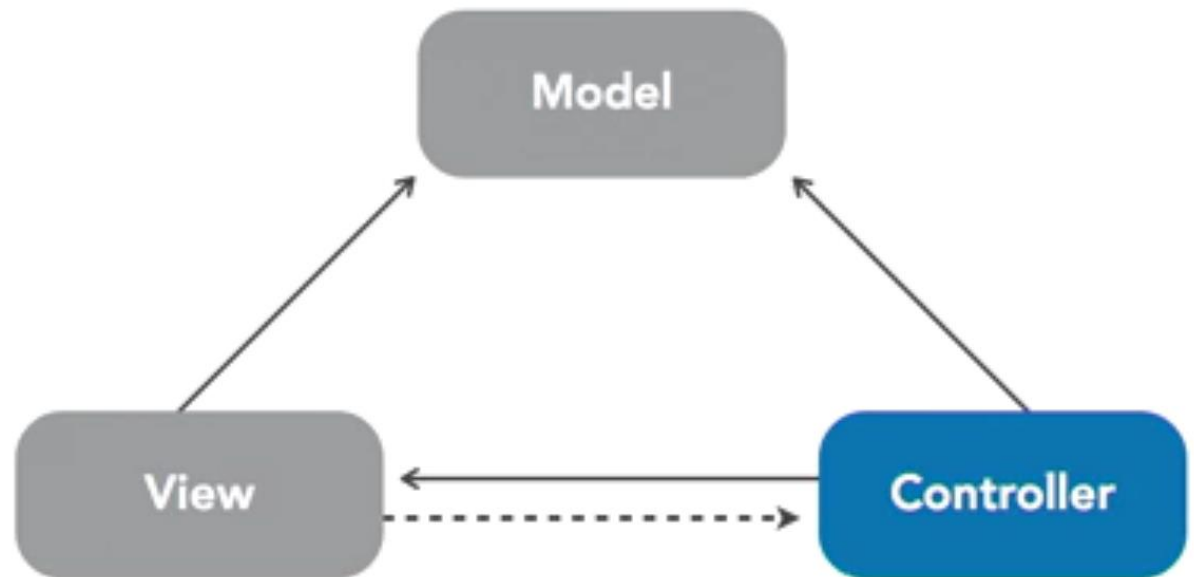
MVC : View

- Good Looking User Experience
- Very Little Logic



MVC : Controller

- Controls Application Workflow
- Orchestrates Interaction between the model and the view



MVC : Model

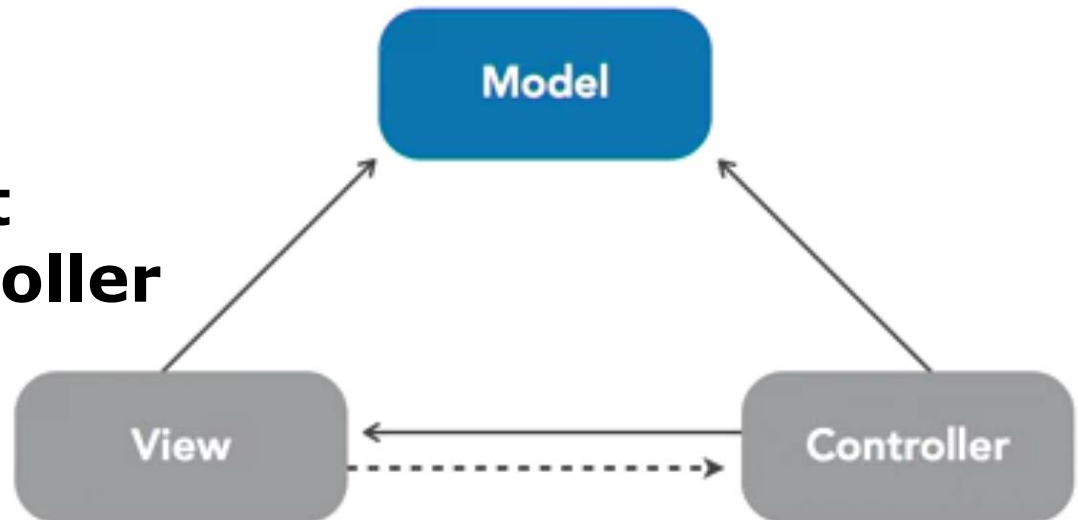
- All Business Logic

Business Processes

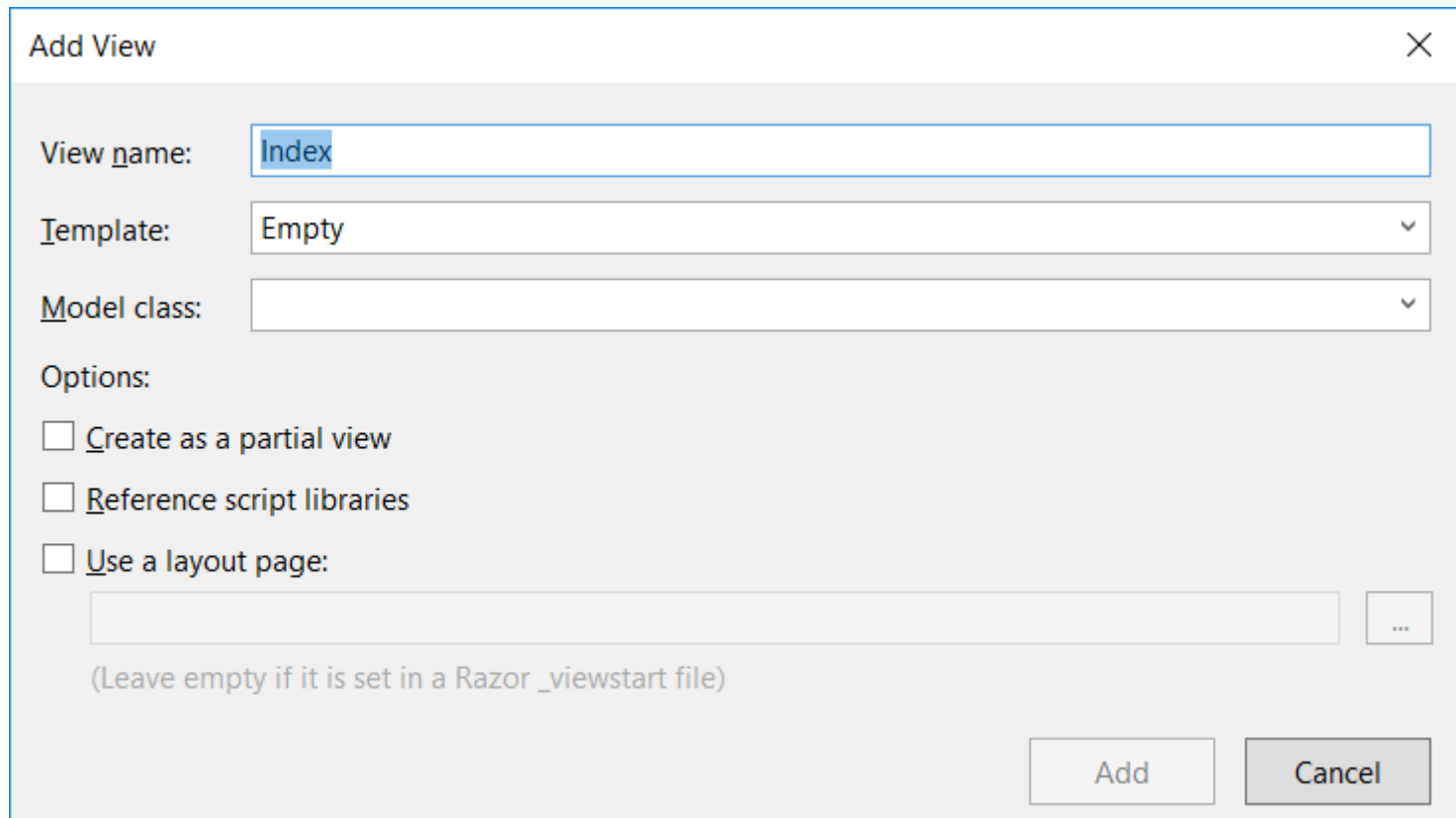
Validation Rules

System Integration

**Does not know
anything about
view and controller**



Adding Views



The image shows a 'Add View' dialog box with a title bar containing a close button (X). The dialog contains several input fields and checkboxes. The 'View name' field has 'Index' entered. The 'Template' dropdown is set to 'Empty'. The 'Model class' dropdown is empty. Under the 'Options' section, three checkboxes are present: 'Create as a partial view', 'Reference script libraries', and 'Use a layout page:'. The third checkbox is selected. Below it is a text field for the layout page name, which is empty, followed by a small button with three dots. A note in parentheses says '(Leave empty if it is set in a Razor _viewstart file)'. At the bottom right are 'Add' and 'Cancel' buttons.

Add View

View name: Index

Template: Empty

Model class:

Options:

☐ Create as a partial view

☐ Reference script libraries

☒ Use a layout page:

(Leave empty if it is set in a Razor _viewstart file)

Add Cancel