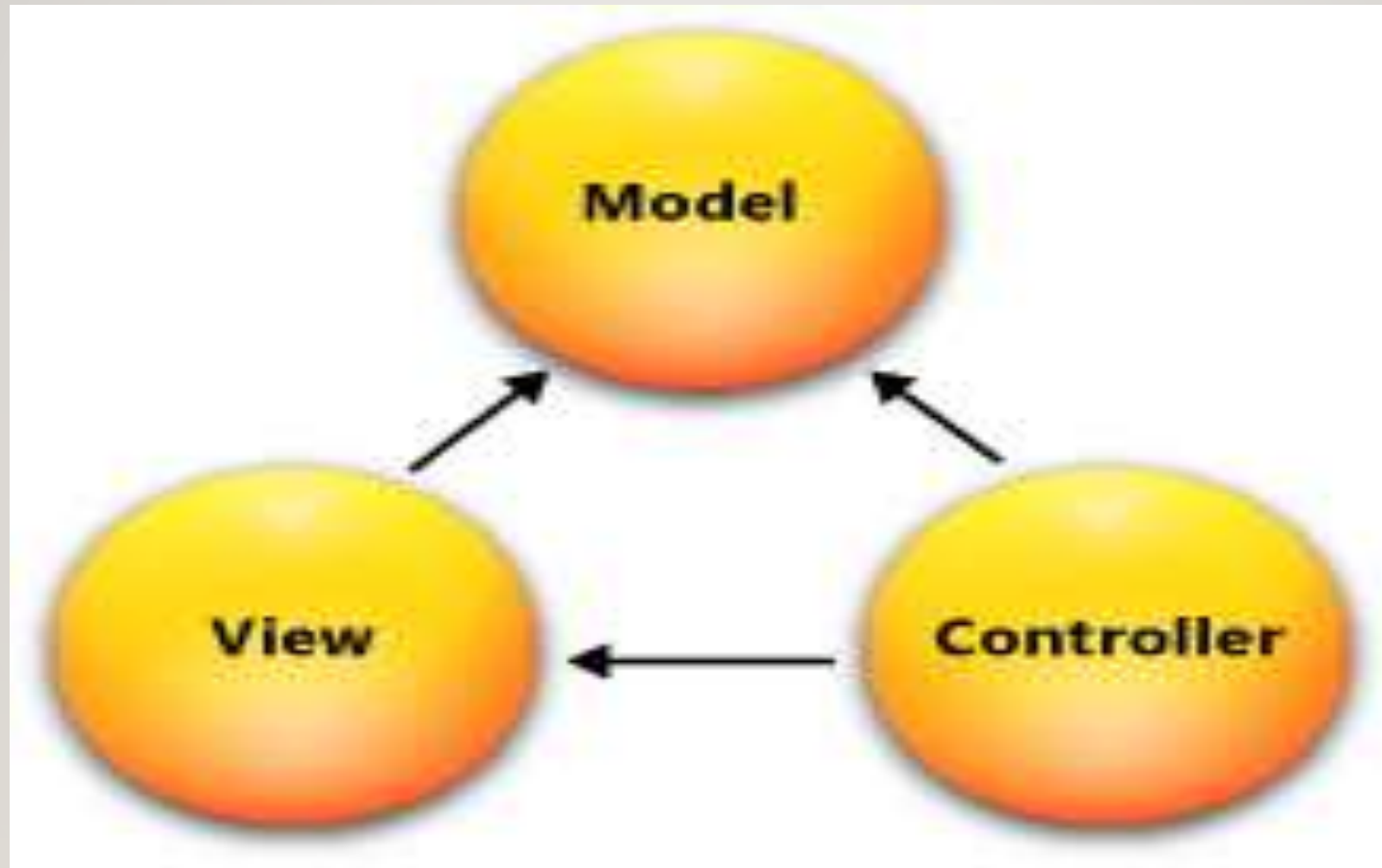


MVC (MODEL-VIEW-CONTROLLER)

- MVC is a software design pattern for developing web application.
- The MVC architectural pattern separates an app into three main components: **M**odel, **V**iew, and **C**ontroller.

MVC DESIGN PATTERN



CONTD...

Models:

- Classes that represent the data of the app. The model classes use validation logic to enforce business rules for that data.
- Model objects retrieve and store model state in a database.
- It is responsible for maintaining data.

Views:

- Views are the components that display the application's user interface (UI).
- Generally, this UI displays the model data.

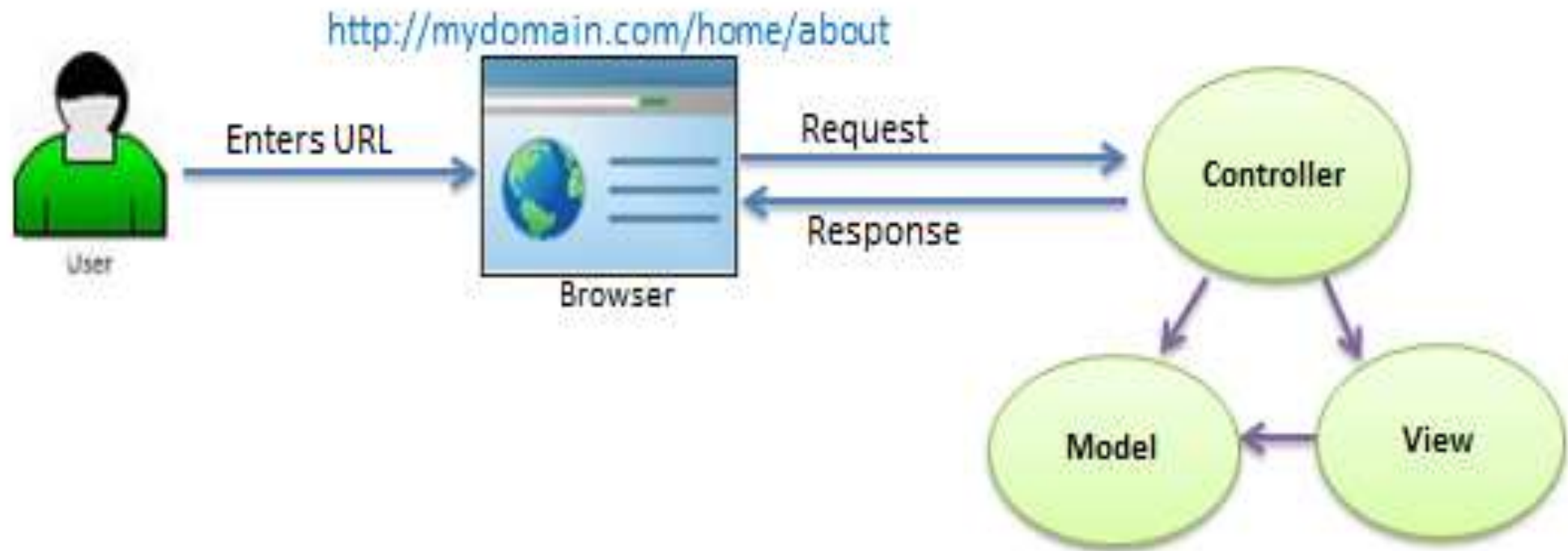
CONTD...

- The view only displays information.

Controller:

- Controllers are the components that handle user interaction, work with the model, and ultimately select a view to render that displays UI.
- It controls the interactions between the Model and View.
- They retrieve model data and call view templates that return a response.
- The controller handles and responds to user input and interaction.

MVC ARCHITECTURE



CONTD...

- When the user enters a URL in the browser, it goes to the server and calls appropriate controller.
- Then, the Controller uses the appropriate View and Model and creates the response and sends it back to the user.

ASP.NET CORE

- It is a cross-platform, high-performance, open-source framework for building modern, cloud-based, Internet-connected applications.
- It ships entirely as NuGet packages.
- Using NuGet packages allows to optimize app to include only the necessary dependencies.

ASP.NET CORE PROVIDES

- Ability to build and run on Windows, macOS, and Linux.
- A unified story for building web UI and web APIs.
- Built-in dependency injection.
- A cloud-ready, environment-based configuration system.
- A lightweight, high-performance, and modular HTTP request pipeline.
- Ability to host on IIS, Nginx, Apache, Docker, or self-host in your own process.
- Open-source and community-focused.
- Deploy to the cloud or on-premises.

FRAMEWORK SELECTION

ASP.NET Core	ASP.NET
Build for Windows, macOS, or Linux	Build for Windows
Razor Pages is the recommended approach to create a Web UI as of ASP.NET Core 2.x	Use Web Forms, SignalR, MVC, Web API, WebHooks, or Web Pages
Multiple versions per machine	One version per machine
Higher performance than ASP.NET	Good performance
Choose .NET Framework or .NET Core runtime	Use .NET Framework runtime
Develop with Visual Studio, Visual Studio for Mac, or Visual Studio Code using C# or F#	Develop with Visual Studio using C#, VB, or F#