



Bài 2

Controller trong ASP.NET MVC



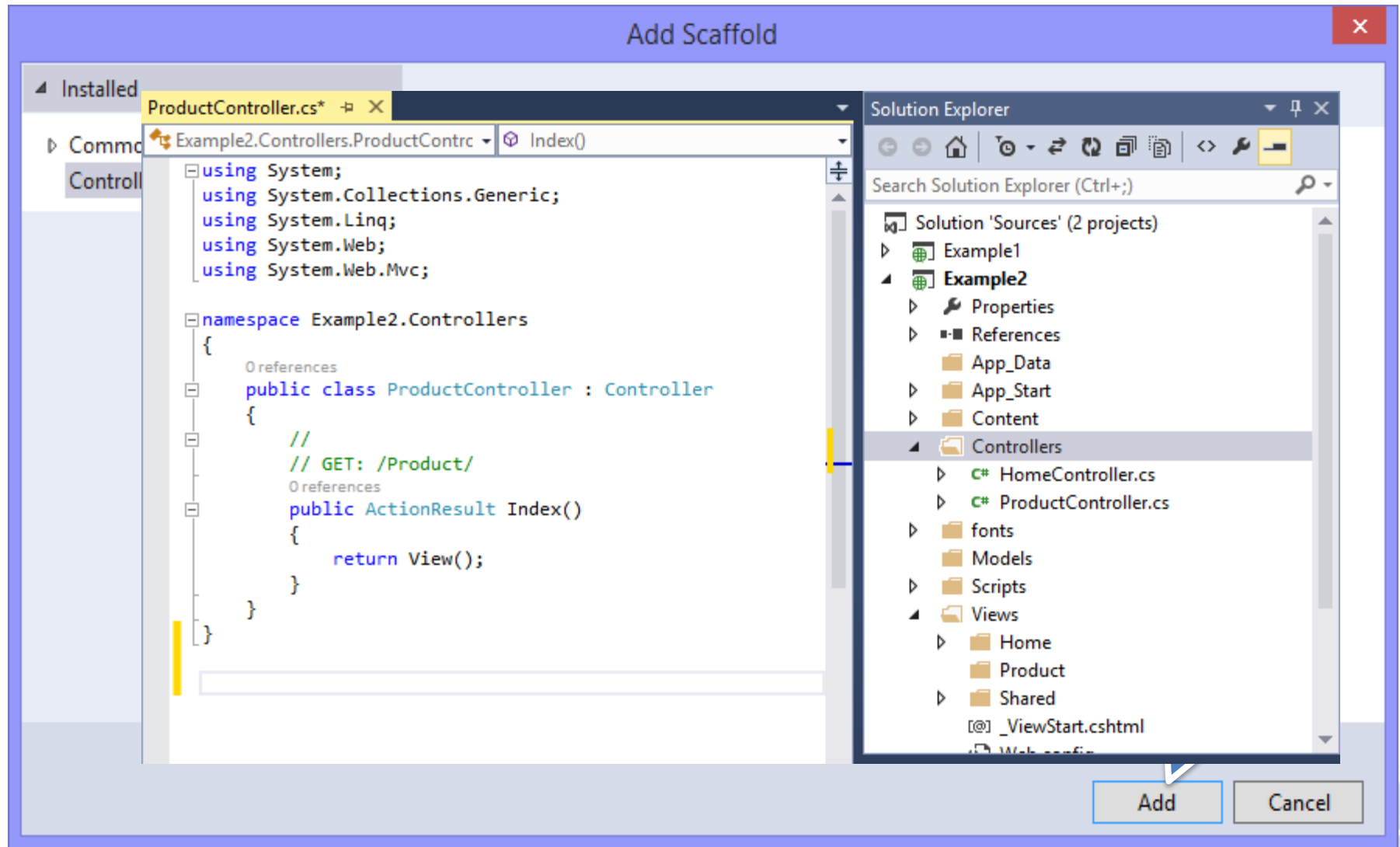
- ◆ Nhiệm vụ của Controller
- ◆ Cách tạo Controller
- ◆ Làm việc với các action trong controller
- ◆ Các kiểu ActionResult
- ◆ Giới thiệu về Routing
- ◆ Đăng ký Routing
- ◆ Các mẫu URL

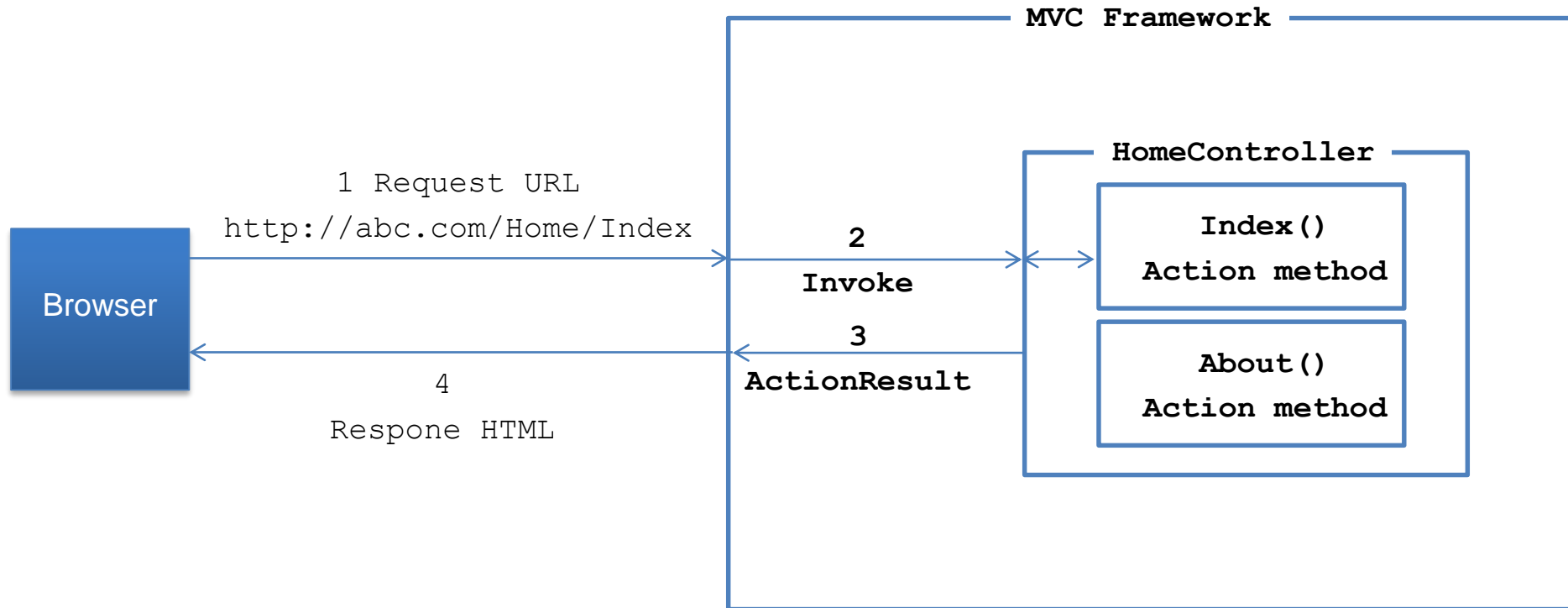


- ◆ Đảm nhiệm việc nhận các request tới và thực thi code của ứng dụng phù hợp
- ◆ Giao tiếp với Model của ứng dụng và chọn ra view cần thiết để sinh ra cho request
- ◆ Cho phép phân tách tầng business logic và tầng presentation
- ◆ Xác định phương thức phù hợp để gọi cho request tới.
- ◆ Kiểm tra dữ liệu của request tới trước khi phương thức yêu cầu được gọi..
- ◆ Nhận request dữ liệu và chuyển nó tới phương thức được request.
- ◆ Điều khiển bất kỳ ngoại lệ nào khi phương thức request tung ra.
- ◆ Trợ giúp việc sinh ra view dựa trên kết quả của phương thức yêu cầu.



Tạo một controller





```
public class ProductController : Controller
{
    // GET: /Product/GetProducts
    0 references
    public ActionResult GetProucts()
    {
        //code logic cho action
        return View();
    }
    // GET: /Product/ProductDetail/id
    0 references
    public ActionResult ProducDetail(string id)
    {
        //code logic cho action
        return View();
    }
}
```

- ◆ Hầu hết các action của controller trả về kiểu ActionResult, tuy nhiên bạn có thể chỉ ra kiểu trả về khác như string, bool, int, ...

```
//GET: /Product/GetName  
0 references  
public string GetName()  
{  
    return "Tên sản phẩm";  
}
```



Các kiểu ActionResult

- ◆ ActionResult là lớp abstract cho tất cả các lớp result khác thực thi

Action Result Type	Examples of Use (in action methods)
ViewResult	<code>return View();</code> <code>return View("view", modelObject);</code>
PartialViewResult	<code>return PartialView();</code>

```
public JsonResult GetNames()  
{  
    string[] names = { "Mạnh", "Thắng", "Linh", "Hoa" };  
    return Json(names, JsonRequestBehavior.AllowGet);  
}
```

JsonResult	<code>return Json(someObject);</code>
JavaScriptResult	<code>Return JavaScript("\$('#table').init();");</code>
HttpUnauthorizedResult	<code>return new HttpUnauthorizedResult();</code>
EmptyResult	<code>return new EmptyResult();</code>

- ◆ Routing engine là một khái niệm độc lập với ASP.NET MVC, nó là một thành phần thuộc ASP.NET, nó có thể xử dụng bởi cả ASP.NET Web Form và ASP.NET MVC
- ◆ Routing engine có hai mục đích chính:
 - ◆ Ánh xạ các URL đầu vào cho các ứng dụng và định tuyến chúng tới Controller và Action method cụ thể để thực thi, xử lý chúng đúng cách.
 - ◆ Tạo ra các URL dựa vào Controllers và Actions mà ta chỉ ra.



- ◆ Cấu hình route default: mở thư mục App_Start -> Mở tệp RouteConfig.cs

```
public class RouteConfig
{
    1 reference
    public static void RegisterRoutes(RouteCollection routes)
    {
        routes.IgnoreRoute("{resource}.axd/{*pathInfo}");

        routes.MapRoute(
            name: "Default",
            url: "{controller}/{action}/{id}",
            defaults: new { controller = "Home", action = "Index", id = UrlParameter.Optional }
        );
    }
}
```



- ◆ Trong ứng dụng ASP.NET MVC bao gồm tệp Global.asax:
 - ◆ Khởi tạo ứng dụng với các tính năng của MVC Framework khi ứng dụng khởi động.
 - ◆ Chứa đựng lớp MvcApplication với phương thức Application_Start() mà bạn cần đăng ký route default để ứng dụng sử dụng khi nhận request tới.

```
public class MvcApplication : System.Web.HttpApplication
{
    0 references
    protected void Application_Start()
    {
        AreaRegistration.RegisterAllAreas();
        FilterConfig.RegisterGlobalFilters(GlobalFilters.Filters);
        //đăng ký route
        RouteConfig.RegisterRoutes(RouteTable.Routes);
        BundleConfig.RegisterBundles(BundleTable.Bundles);
    }
}
```



URL Patterns

```
public static void RegisterRoutes(RouteCollection routes)
{
    routes.IgnoreRoute("{resource}.axd/{*pathInfo}");

    routes.MapRoute(
        name: "Default",
        url: "{controller}/{action}/{id}",
        defaults: new { controller = "Home", action = "Index", id =
            UrlParameter.Optional }
    );
}
```

URL pattern

Route Name

Default values

Ignore route ending in axd

Thứ tự các Route

```
public static void RegisterRoutes(RouteCollection routes)
{
    routes.IgnoreRoute("{resource}.axd/{*pathInfo}");

    /// http://server/product/productlist
    routes.MapRoute(
        name: "Product List",
        url: "product/productlist",
        defaults: new { controller = "Product", action = "ProductList" });

    /// http://server/products/
    routes.MapRoute(
        name: "Product List default",
        url: "products",
        defaults: new { controller = "Product", action = "ProductList" });

    /// http://server/product/123/any-things-string.html
    routes.MapRoute(
        name: "Product detail",
        url: "product/{productId}/{name}",
        defaults: new { controller = "Product", action = "Detail" },
        constraints: new { productId = @"\d+" });
    |
    routes.MapRoute(
        name: "Default",
        url: "{controller}/{action}/{id}",
        defaults: new { controller = "Home", action = "Index", id = UrlParameter.Optional }
    );
}
```

