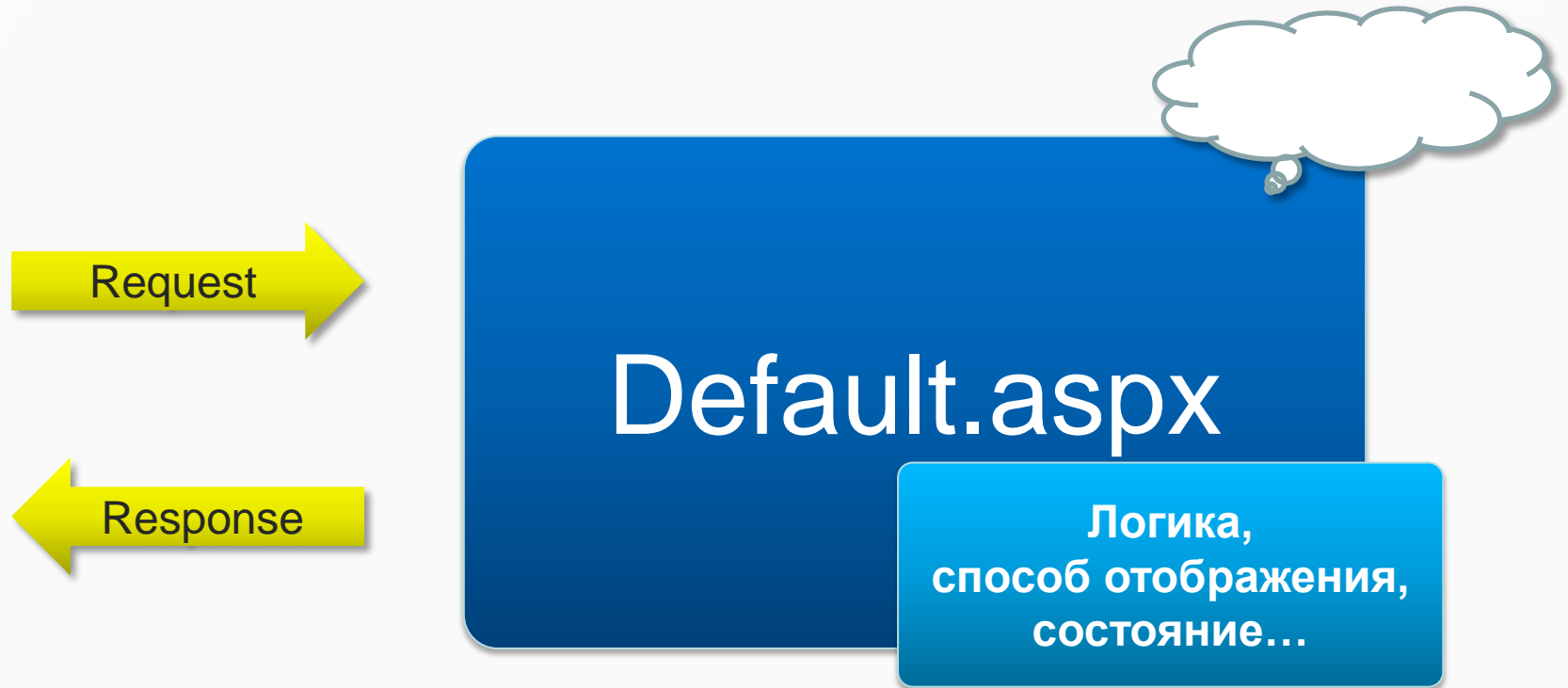




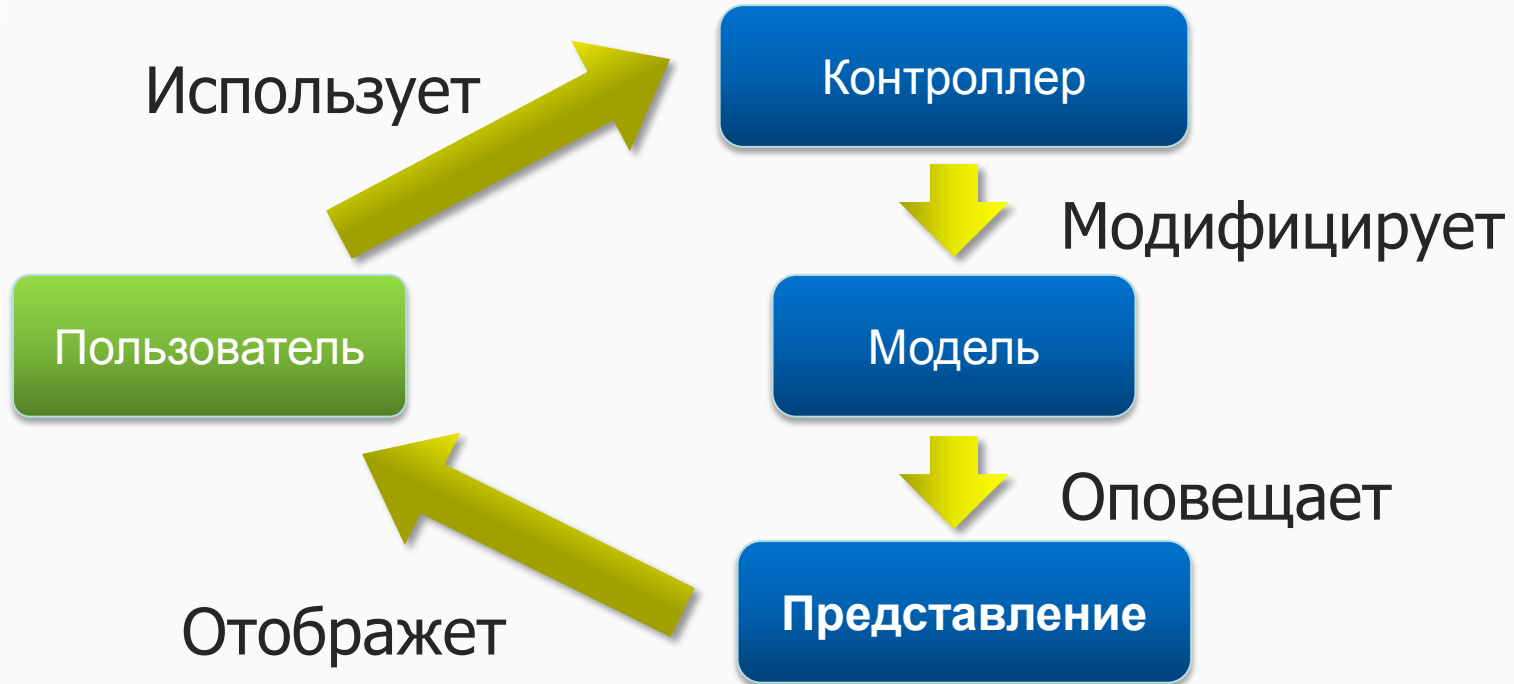
Тема: ASP .Net MVC

Докладчик: Пыхтин Павел

Что не так с Web Forms?



Паттерн MVC



ASP.Net MVC



- Является классом, наследником базового класса *System.Web.Mvc.Controller*
- Выполняет основную обработку запроса
- Логика обработки описывается отдельными методами (Action)

- Отвечают за непосредственное формирование ответа на запрос
- Является шаблоном страницы или ее фрагмента
- Для описания разметки используется Razor
- Представление ищется по пути:
 - **`\Views\{Controller}\{View}.cshtml`**,
- Если оно не было найдено, проверяется путь:
 - **`\Views\Shared\{View}.cshtml`**

- Конфигурируются в `RouteConfig.cs`
- Задаются при помощи шаблона

```
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  
        } );  
}
```


Демонстрация

```
<form action='...'>  
  <label>Email</label>  
  <input type="text"/>  
  
  <label>Password</label>  
  <input type="password"/>  
  
  <label>Password confirmation</label>  
  <input type="password"/>  
  
  <input type='submit' value='OK' />  
</form>
```

- Asp.Net MVC предоставляет набор методов для генерации разметки форм
 - `HtmlHelper.BeginForm`
 - `HtmlHelper.TextBoxFor`
 - `HtmlHelper.DropDownFor`
 - `HtmlHelper.EditorFor`
 - ...

- **Форматирование**
 - `DataTypeAttribute`
 - `DisplayAttribute`
- **Валидация**
 - `RequiredAttribute`
 - `StringLengthAttribute`

Тестирование контроллера

```
[TestClass]
```

```
public class MyControllerTest {  
    private MyController _controller;  
    private Mock<IServiceProvider> _serviceProvider;  
    private Mock<CheckerService> _checkerService;  
  
    [TestInitialize]  
    public void Init() {  
        _checkerService = new Mock<CheckerService>();  
  
        _serviceProvider = new Mock<IServiceProvider>();  
        _serviceProvider.Setup(x => x.GetCheckerService()).Returns(_checkerService.Object);  
        ServiceProvider.SetCurrent(_serviceProvider.Object);  
  
        _controller = new MyController();  
    }  
}
```

• • •

Тестирование контроллера

• • •

```
[TestMethod]
public void RedirectsToAnotherPageIfCheckPassed() {
    const int value = 11;
    _checkerService.Setup(x => x.Check(value)).Returns(true);

    var result = (RedirectToRouteResult)_controller.DoSomething(value);
    Assert.AreEqual("AnotherController", result.RouteValues["Controller"]);
    Assert.AreEqual("AnotherPage", result.RouteValues["Action"]);
}
```

```
[TestMethod]
public void ShowsErrorPageIfCheckFailed() {
    const int value = 11;
    _checkerService.Setup(x => x.Check(value)).Returns(true);

    var result = (ViewResult)_controller.DoSomething(value);

    Assert.AreEqual("Error", result.View);
}
```

Демонстрация

- GET
- POST (POST A + POST A != POST A)
- PUT (PUT A + PUT A == PUT A)
- DELETE


```
[HttpGet]
```

```
public ActionResult Edit(int id) {  
    return View(new PresentationEntity());  
}
```

```
[HttpPost]
```

```
public ActionResult Edit(PresentationEntity pe) {  
    return RedirectToAction(  
        "Index",  
        new {id = pe.id});  
}
```

- Позволяют дополнить методы контроллера дополнительной логикой
- Применяются как к отдельным действиям так и к контроллеру в целом

Демонстрация

- Являются шаблонами верхнего уровня
- Корневой лэйаут задается в разметке страницы `_ViewStart.cshtml`
- Могут быть вложенными
- Позволяют модифицировать разметку сайта, не затрагивая содержимое отдельных страниц

Лэйаут

```
...  
<div>  
  @RenderSection("Menu")  
</div>  
...
```

Страница

```
...  
@section Menu{  
  <li id='miAdd'>Add</li>  
  <li id='miDelete'>Delete</li>  
}  
...
```

- Являются шаблонами «нижнего» уровня
- Не имеют лэйаута
- Выводятся при помощи методов
 - `@Html.RenderPartial("Contact", @Model.ContatctInfo)`
 - `@Html.Partial("Contact", @Model.ContatctInfo)`

Демонстрация

- <http://www.codeproject.com/Articles/741228/MVC-Application-Lifecycle>
- <http://andrey.moveax.ru/mvc3-in-depth/>

Спасибо за внимание