

Buổi 2

Thiết kế giao diện với ASP.NET

I. ASP.NET WebForm Control

1. Các control cơ bản

ASP.NET WebForm (gọi tắt là WebForm) cung cấp rất nhiều control ở phía server để phục vụ việc thiết kế giao diện web. Tất cả các control này đều bắt đầu bằng tiền tố <asp:...> trong tên thẻ và bắt buộc có 2 thuộc tính ID và runat, trong đó ID do người dùng đặt theo quy chuẩn đặt id của control HTML, và runat mang giá trị là "server":

```
<asp:..... ID="....." runat="server">...
```

Khi biên dịch, các control này sẽ được chuyển thành các control HTML tương ứng.

Danh sách một số control thường dùng trong WebForm:

| Tên control WebForm | Thẻ ASP.NET | Thẻ HTML (sau khi biên dịch) |
|---------------------|--------------------|--|
| Button | <asp:Button> | <input type="button"> <input type="submit"> |
| CheckBox | <asp:CheckBox> | <input type="checkbox"> |
| DropDownList | <asp:DropDownList> | <select> |
| FileUpload | <asp:FileUpload> | <input type="file"> |
| HyperLink | <asp:HyperLink> | <a> |
| Image | <asp:Image> | |
| ImageButton | <asp:ImageButton> | <input type="image"> |
| Label | <asp:Label> | |
| LinkButton | <asp:LinkButton> | <a> |
| Panel | <asp:Panel> | <div> |
| RadioButton | <asp:RadioButton> | <input type="radio"> |
| TextBox | <asp:TextBox> | <input type="text"> |

2. Các thuộc tính cơ bản

Các control WebForm có khá nhiều thuộc tính chung bên cạnh các thuộc tính riêng đặc thù cho một số control. Các thuộc tính này có thể được tìm thấy tại mục Properties của mỗi control.

Danh sách một số thuộc tính chung cơ bản:

| Tên thuộc tính | Ý nghĩa |
|----------------------|--|
| BackColor, ForeColor | Màu nền và màu chữ |
| CausesValidation | Control có trigger việc kiểm tra dữ liệu (data validation) hay không |
| Controls | (Collection) Danh sách các control con nằm trong control này |
| CssClass | Tên class CSS (tương đương thuộc tính class của HTML) |
| Enabled | Control có thể tương tác được hay không |
| Height, Width | Chiều cao, chiều rộng (pixel hoặc %) |

| | |
|----------|---|
| ID | ID của control |
| TabIndex | Thứ tự control khi nhấn Tab |
| ToolTip | Text gợi ý hiển thị khi di chuyển chuột lên control |
| Visible | Control đang hiện hay ẩn |

3. Các sự kiện cơ bản

Các sự kiện (*event*) của WebForm control đều có thể được tìm thấy tại mục Properties → Events của mỗi control. Một số sự kiện sẽ khiến form gửi dữ liệu lên server, những sự kiện này được gọi là *postback event*. Những sự kiện không khiến form gửi dữ liệu lên server được gọi là *non-postback event*.

Lưu ý: Khi thuộc tính AutoPostBack của control là True thì control đó sẽ luôn kích hoạt việc postback dữ liệu.

Danh sách một số sự kiện cơ bản:

| Tên sự kiện | Tên thuộc tính của control | Control áp dụng |
|----------------------|----------------------------|---|
| Click | OnClick | Button, ImageButton, LinkButton, Image... |
| Command | OnCommand | Button, ImageButton, LinkButton |
| TextChanged | OnTextChanged | TextBox |
| SelectedIndexChanged | OnSelectedIndexChanged | DropDownList, ListBox, RadioButtonList, CheckBoxList. |
| CheckedChanged | OnCheckedChanged | CheckBox, RadioButton |

Đối tượng Page có sự kiện mặc định là Load. Tương tự, mỗi control đều có 1 sự kiện mặc định. Sự kiện này có thể được tạo bằng cách double click vào control trong chế độ design.

Danh sách các sự kiện mặc định của một số control:

| Tên control | Tên sự kiện mặc định |
|-----------------|----------------------|
| Button | Click |
| CheckBox | CheckedChanged |
| CheckBoxList | SelectedIndexChanged |
| DataGrid | |
| DataList | |
| DropDownList | |
| HyperLink | Click |
| ImageButton | |
| LinkButton | |
| ListBox | SelectedIndexChanged |
| RadioButton | CheckedChanged |
| RadioButtonList | SelectedIndexChanged |

II. Cấu trúc thư mục của 1 hệ thống web

Khi làm việc với website, chúng ta cần tổ chức cấu trúc thư mục như sau:

- Tất cả trang web (.html, .aspx, .php...) đặt ở thư mục gốc của thư mục web.
- Thư mục *css* chứa các file mã nguồn CSS (.css).
- Thư mục *js* chứa các file mã nguồn JavaScript (.js).
- Thư mục *img* chứa các hình ảnh.
- Thư mục *font* chứa các font chữ sử dụng trong trang web.
- Thư mục *assets* chứa các nội dung, tài nguyên khác như video, âm thanh...

III. Master Page & Content Page

Các trang web trong cùng 1 hệ thống thường sẽ có bố cục, giao diện tương tự nhau, điển hình là các thành phần như header, slider¹, navigation bar/menu, footer.

Nếu thiết kế các trang web rời rạc với nhau thì sẽ gặp một số vấn đề như sau:

- Trùng lặp code dẫn tới tốn thời gian, tốn bộ nhớ lưu trữ.
- Khi có sự thay đổi về code liên quan tới những phần giao diện chung, phải sửa code của từng trang web.
- Dễ xảy ra sự không đồng nhất giữa các trang.

Do đó, ASP.NET cung cấp một loại trang web để giúp giải quyết các vấn đề trên, đó là *Master Page*. Master Page sẽ chứa những phần giao diện chung của các trang web, và có khoảng trống (*placeholder*) để chèn những nội dung riêng của từng trang. Các trang con (*Content Page*) sẽ kế thừa phần giao diện chung từ Master Page và chứa các phần giao diện riêng của từng trang.

1. Tạo Master Page

Để tạo 1 Master Page trong VS, click chuột phải vào project, chọn *Add* → *New Items...* và chọn *Web Forms Master Page* trong cửa sổ hiện ra. Phần mở rộng của trang Master Page là *.master*.

Chú ý: Không chọn nhầm mục *Web Form with Master Page*. Ngoài ra *Web Forms Master Page (Nested)* dùng ra trường hợp Master Page phân cấp.

Mã nguồn của 1 Master Page:

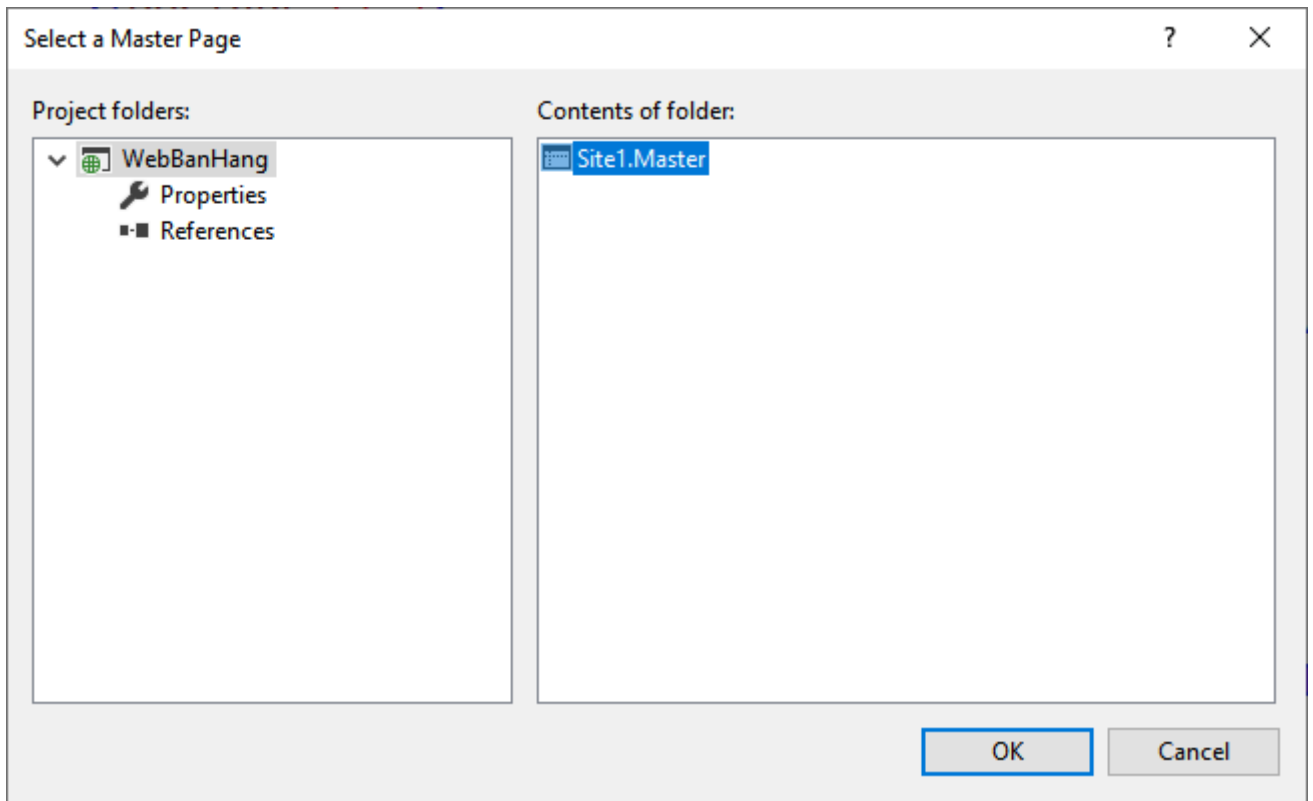
```
<head runat="server">
  <title></title>
  <asp:ContentPlaceholder ID="head" runat="server">
  </asp:ContentPlaceholder>
</head>
<body>
  <form id="form1" runat="server">
    <div>
      <asp:ContentPlaceholder ID="ContentPlaceholder1"
runat="server">
      </asp:ContentPlaceholder>
    </div>
  </form>
</body>
```

Mặc định, Master Page có 2 thẻ `<asp:ContentPlaceholder>` (trong thẻ `<head>` và thẻ `<body>`). Thẻ này là nơi đặt nội dung riêng của các trang con. Tất cả những thành phần giao diện chung đều phải được đặt ngoài 2 thẻ này.

¹ Còn gọi là slideshow hoặc carousel

2. Tạo Content Page

Để tạo 1 Content Page trong VS, click chuột phải vào project, chọn *Add* → *New Items...* và chọn *Web Form with Master Page* trong cửa sổ hiện ra, sau đó chọn Master Page muốn sử dụng cho trang này và nhấn OK:



Mã nguồn của 1 Content Page:

```
<asp:Content ID="Content1" ContentPlaceHolderID="head" runat="server">
</asp:Content>
<asp:Content ID="Content2" ContentPlaceHolderID="ContentPlaceHolder1"
runat="server">
</asp:Content>
```

Do Content Page sử dụng những thành phần chung từ Master Page nên sẽ không có các thẻ <html>, <head>, <body>... Trong đoạn mã trên, những gì được đặt vào thẻ có ID là Content1 sẽ xuất hiện trong thẻ <head> lấy từ Master Page, những gì được đặt vào thẻ có ID là Content2 sẽ xuất hiện trong thẻ <body> lấy từ MasterPage.

Những control nào nằm ở Master Page thì khi tạo sự kiện phải tạo ở Master Page. Những control nào nằm ở Content Page thì khi tạo sự kiện phải tạo ở Content Page tương ứng.