



# Chương 9

---

## **Cấu hình ứng dụng trong ASP.NET**

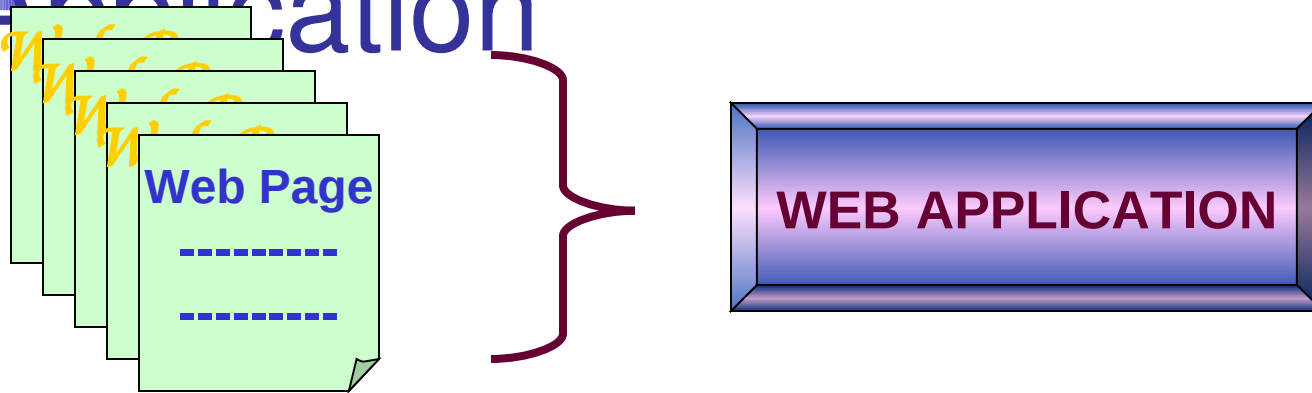


# Mục đích

---

- *Machine.Config*
- *eb.Config*
- *Cấu trúc của một tập tin cấu hình*
- *Bảo mật trang Web dùng chứng thực*

# Web Pages – Web Application



- Ứng dụng Web bao gồm nhiều trang Web
- Thiết lập các thuộc tính cho ứng dụng sẽ tác động lên tất cả các trang
- Thiết lập thuộc tính sẽ kiểm soát ứng dụng trong thời gian thực thi → cấu hình ứng dụng ASP.NET



# Tập tin cấu hình

---

- Tất cả cấu hình được lưu trong tập tin **web.config**, và lưu trong thư mục gốc của ứng dụng
- web.config là một tập tin HTML

```
<configuration>
  <system.web>
    <sessionState timeout="20" />
  </system.web>
</configuration>
```

ASP.NET cung cấp tập tin cấu hình để tùy biến web site →  
cung cấp sự linh động cho các ứng dụng Web



# Đặc trưng của tập tin cấu hình

---

- Lưu trữ dạng văn bản
- Viết trong XML, quy tắc gán qua tên thẻ và thuộc tính
- Không cần khởi động server khi thay đổi tập tin
- Các thư mục khi đề tập tin cấu hình trước đó
- Client không thể xem nội dung của nó từ trình duyệt



# Quy tắc đặt tên thẻ

---

Kiểu thẻ	Quy tắc
Thẻ và thuộc tính	<b>Camel-cased</b>
Giá trị thuộc tính	<b>Pascal-case</b> : ngoại trừ giá trị <b>true</b> , <b>false</b> .



# Các kiểu tập tin cấu hình

---

## Machine.config

- Mô tả cấu hình cho tất cả ứng dụng trên Server
- Dựa trên XML
- Lưu trữ trong *C:\WinNT\Microsoft.NET\Framework\v.1.xxxx\config*
- Chỉ một tập tin cho một cài đặt ASP.NET trên một máy

Build number  
of .net CLR

## Web.config

- Thiết lập được áp dụng cho từng ứng dụng web trên server
- Dựa trên XML lưu trữ trong thư mục ứng dụng Web của Web Server
- Một tập tin cho mỗi thư mục của ứng dụng Web

# Các kiểu tập tin cấu hình...

**Machine.config**

- One per machine

**Web.config**

- One per application
- Override settings of machine.config

**Web Applications**





# Cấu trúc của tập tin cấu hình

---


- Tập tin cấu hình bao hàm tất cả các thẻ bên trong thẻ configuration, nếu không đúng định dạng, trình biên dịch sẽ ném ra một biệt lệ
- Thuộc tính cấu hình của máy được thiết lập giữa các thẻ này
- Bên trong các thẻ configuration, thông tin cấu hình được nhóm thành 2 loại:
  - Mục khai báo trình quản lý mục cấu hình
  - Vùng thiết lập vùng cấu hình
- Mục cấu hình được định nghĩa giữa thẻ `<configSections>` và `</configSections>`
- Các mục đơn được định nghĩa dùng thẻ `<section>`.



# Kiểu tập tin Web.config

---

```
<configuration>
  <configSections>
    <section name="sectionSettings" type="Class" />
    <sectionGroup name="sectionGroup">
      <section name="sectionSettings" type="Class" />
    </sectionGroup>
  </configSections>
  <section name="sectionSettings" type=<Class>" />
  <sectionGroup>
    <sectionSettings attribute="someValue" />
    <sectionSettings SomeAttribute="SomeValue">
      <element attribute="value"/>
    </sectionSettings>
  </sectionGroup>
</configuration>
```



# Khai báo trình quản lý mục cấu hình

---

- Định nghĩa một lớp (thường là một lớp dựa trên .Net) dùng để diễn giải dữ liệu cấu hình
- Mục cấu hình được đặt trong the `<configSections>`
- Thông thường mục này được đặt trong tập tin `Machine.Config`, và không cần đặt trong mỗi tập tin `Web.Config`, nó sẽ được kế thừa tự động



# Vùng thiết lập vùng cấu hình

---

- Định nghĩa các thiết lập thật sự cho một số tùy chọn
- Chứa trình quản lý các mục mà mã cấu hình được viết
- Mỗi trình quản lý mục được nhóm trong `<sectionGroup>`, cung cấp một cấu trúc cho tập tin cấu hình
- Có khoảng 30 thiết lập cấu hình khác nhau định nghĩa trong tập tin `Machine.config`

# Thiết lập cấu hình cho trang

Cho phép các nhà phát triển kiểm soát một số hành vi mặc định cho tất cả các trang ASP.NET trong ứng dụng hay máy

```
<configuration>
  <system.web>
    <pages buffer="true"
      enableViewState="false" />
    </system.web>
  </configuration>
```

Thuộc tính	Mô tả	Giá trị mặc định
Buffer	Các hồi báo đến client được gửi trực tiếp hay đầu tiên được cache trên server rồi gửi	True
enableViewState	Cho phép ViewState	True

# Thiết lập ứng dụng Application

- Được đặt trong thẻ `<appSettings>` và `</appSettings>`.
- Cho phép người dùng thiết lập cấu hình ứng dụng chi tiết
- Thiết lập ứng dụng cho phép lưu trữ và lấy thông tin theo cặp key-value
- Nó cũng có thể lưu trữ truy vấn SQL

```
<configuration>
  <appSettings>
    <add key="MySQLQuery" value="Select * FROM MySQLTable"/>
  </appSettings>
</configuration>
```

```
...
String GetQuery = Configuration.AppSettings ("MySQLQuery");
...
```



# Thiết lập biên dịch

---

- Dùng để chỉ ra các tùy chọn biên dịch cần thiết khi biên dịch các tập tin nguồn ứng dụng
- Tham chiếu các thư viện yêu cầu trong suốt quá trình biên dịch có thể được chỉ ra ở đây
- Nó cũng có thể chỉ ra ứng dụng đang chạy trong chế độ debug

Thuộc tính	Tùy chọn	Mô tả
Debug		Chỉ ra kiểu biên dịch
	true	debug binaries.
	false	retail binaries.
defaultLanguage		Danh sách các ngôn ngữ dùng trong quá trình biên dịch động các tập tin. Ví dụ: "C#; VB".



# Cấu hình thiết lập biên dịch

---

```
<configuration>
  <system.web>
    <compilation
      debug="false"
      defaultLanguage="C#" />
    </system.web>
  </configuration>
```

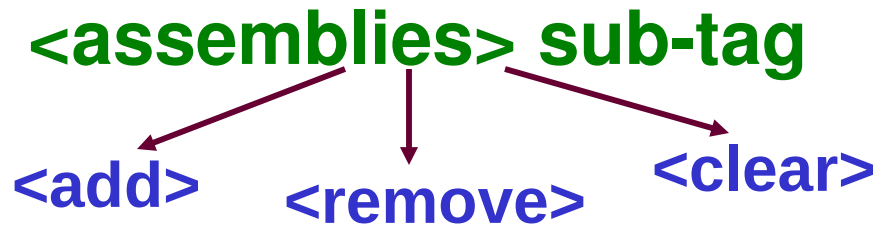




# Thẻ con của thẻ Compilation

---

1



```
<configuration>
  <system.web>
    <compilation
      debug="true" defaultLanguage="C#">
      <assemblies>
        <add assembly="System.Data" />
      </assemblies>
    </compilation>
  </system.web>
</configuration>
```

# Thẻ con của thẻ Compilation



2

## <namespaces> sub-tag

```
<configuration>
  <system.web>
    <compilation
      debug="true" defaultLanguage="C#">
      <namespaces>
        <add namespace="System.Web.UI" />
      </namespaces>
    </compilation>
  </system.web>
</configuration>
```



# Thẻ con của thẻ Compilation

---

## 3 **<compilers> sub-tag**

<compilers> dùng để chỉ ra trình biên dịch và ngôn ngữ dùng để tạo ứng dụng



# Thiết lập customErrors

---

- ASP.NET cung cấp cơ chế linh động để tùy biến các trang lỗi và điều hướng trình duyệt client khi lỗi xảy ra
- Trang lỗi có thể được thiết kế để hiển thị hay giải thích thông tin lỗi thân thiện hơn với người dùng

## Cú pháp:

```
<customErrors
  defaultRedirect="url"
  mode="On | Off | RemoteOnly">
  <error statusCode="statuscode" redirect="url" />
</customErrors>
```



# customErrors Ví dụ

---

```
<configuration>
  <system.web>
    <customErrors      defaultRedirect= "http://
localhost/Appdir/allErrors.aspx"
    mode="RemoteOnly">
      <error statusCode="404"
        redirect=" http:// localhost/ Appdir/
ErrorNo404.aspx"/>
    </customErrors>
  </system.web>
</configuration>
```



# Authentication

---

Tiến trình xác định người dùng và mật khẩu của người dùng → **Authentication**.  
Thiết lập này có thể được chỉ ra trong tập tin web.config.

```
<configuration>
  <system.web>
    <authentication mode="Windows|Forms|Passport|None">

      <forms name="name" loginUrl="url"
        protection="All|None|Encryption "
        timeout="xx" path="/" >
        <credentials passwordFormat="Clear|SHA1|MD5">

          <user name="username" password="password" />

        </credentials>
      </forms>
      <passport redirectUrl="internal"/>
    </authentication>
```



# Kiểu Authentication

---

**WINDOWS** →

Chỉ ra kiểu chứng thực Windows là kiểu chứng thực mặc định, dùng cho bất cứ dạng chứng thực nào của IIS

**FORMS** →

Chỉ ra chứng thực dựa trên form

**PASSPORT** →

Dùng Passport

**NONE** →

Không chứng thực dùng bởi người dùng anonymous users và ứng dụng cung cấp chứng thực riêng



# Thuộc tính của thẻ <forms>

---

Để chứng thực Form, kiểu chứng thực phải được chỉ ra là **Forms**, các thẻ con có các đặc tính sau:

Đặc tính	Tùy chọn	Mô tả
<b>Name</b>	None	Tên Cookie dùng cho chứng thực
<b>LoginUrl</b>	None	Trang URL đăng nhập. Client đổi hướng đến trang này nếu không có chứng thực cookie
<b>protection</b>	Encryption	Cookie được lưu trữ theo dạng mã hóa
	Validation	<b>Kiểm tra hợp lệ</b> scheme nội dung của cookie mã hóa đã được cảnh báo trước khi truyền
	ALL	Ứng dụng dùng cả hai kiểm tra hợp lệ và mã hóa cookie
	None	Ứng dụng cấm cả hai kiểm tra hợp lệ và mã hóa cookie





# Thuộc tính của thẻ <forms>

---

Đặc tính	Tùy chọn	Mô tả
<b>timeout</b>		Thời gian theo phút sau khi chứng thực hết hạn, mặc định là 30.
<b>path</b>		Đường dẫn của cookies ban hành bởi ứng dụng. Mặc định là <b>backslash (/)</b>



# Forms Authentication Ví

```
<configuration>
  <system.web>
    <authentication mode="Forms">
      <forms name="MainForm"
loginUrl="LoginPage.aspx"
      protection="None" timeout="60">
        <credentials passwordFormat="Clear">
          <user name="User1" password="user1!" />
          <user name="User2" password="user2@" />
          <user name="User3" password="user3#" />
        </credentials>
      </forms>
    </authentication>
  </system.web>
</configuration>
```



# Authorization

---

- Sau khi thiết lập chứng thực cho ứng dụng, quyền truy xuất cần được gán cho ai được phép truy cập ứng dụng → Authorization
- Dùng thiết lập Authorization, người dùng được xác nhận thật sự truy cập WebSite

```
<authorization>  
    <allow users="<userList>" />  
    <deny users="<userList>" />  
</authorization>
```



# Authorization Ví dụ

```
<configuration>  <system.web>
  <authentication mode="Forms">
    <forms name="MainForm" loginUrl="LoginPage.aspx"
      protection="None"  timeout="60">
      <credentials passwordFormat="Clear">
        <user name="User1" password="user1!" />
        <user name="User2" password="user2@" />
        <user name="User3" password="user3#" />
      </credentials>
    </forms>
  </authentication>
  <authorization>
    <allow users="User2, User3" />
    <deny users="User1" />
  </authorization>
</system.web> </configuration>
```