

Bài 5

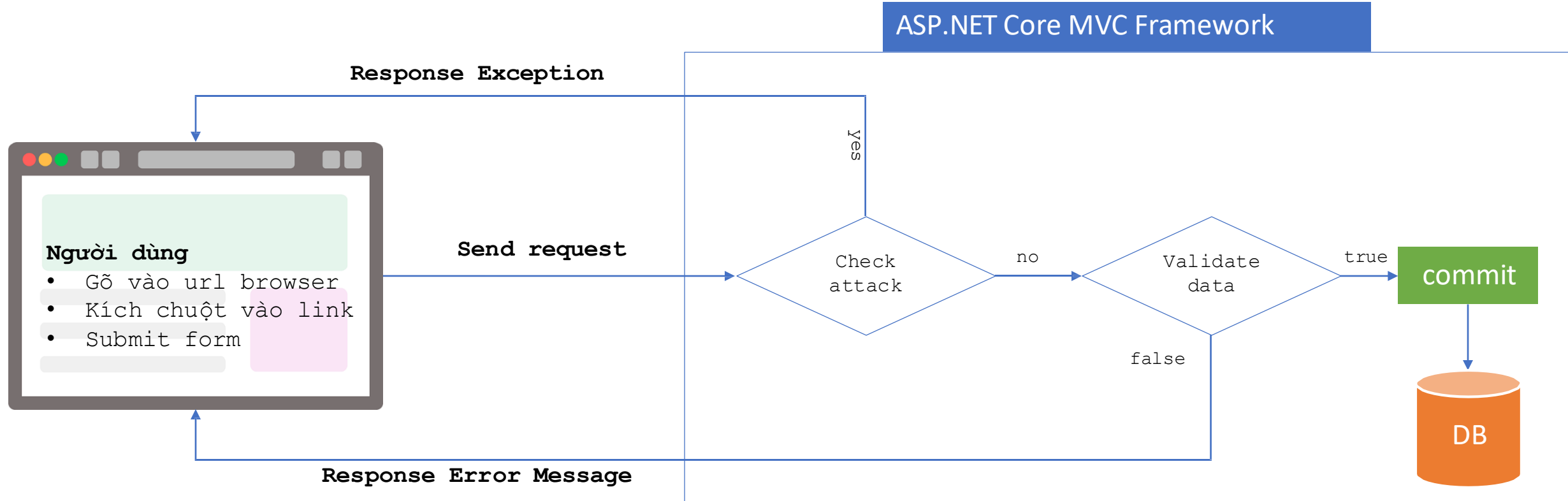
Data validation và Annotation

Mục tiêu

- Giới thiệu về Data Validation
- Manual Validation
- Data Validation sử dụng Data Annotations
- Sử dụng thuộc tính ModelState

Giới thiệu về Data Validation

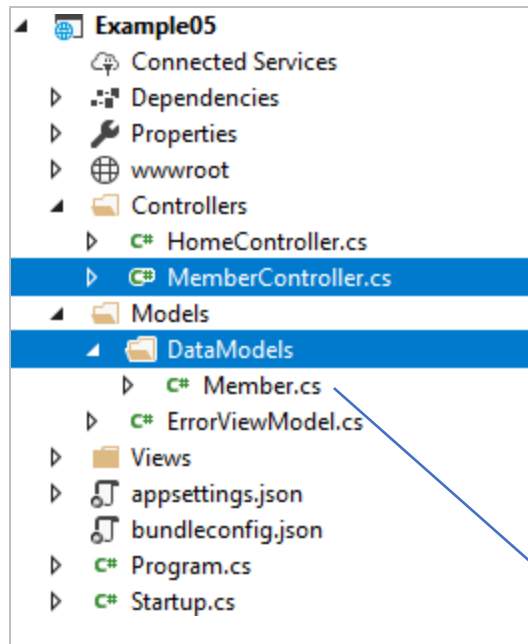
- Data validation là quá trình kiểm tra tính hợp lệ của dữ liệu gửi từ người dùng đến ứng dụng Web.



Manual Validation

- Đây là cách tự code logic trong **Action** của **Controller** để kiểm tra tính hợp lệ của dữ liệu.
 - Kiểm tra độ dài chuỗi dữ liệu
 - Kiểm tra số điện thoại
 - Kiểm tra email
 - Kiểm tra giữa Password và Confirm Password
 -

Manual Vadtation



```
public class MemberController : Controller
{
    public static readonly List<Member> members = new List<Member>();

    public IActionResult Index()
    {
        return View(members);
    }

    public IActionResult Create()
    {
        return View();
    }

    [HttpPost]
    public IActionResult Create(Member member) ...
}

public class Member
{
    public string MemberId { get; set; }
    public string Username { get; set; }
    public string FullName { get; set; }
    public string Password { get; set; }
    public string Email { get; set; }
    public string Phone { get; set; }
    public DateTime Birthday { get; set; }
}
```

Manual Validation

Code Razor cho View "Create"

```
@model Example05.Models.DataModels.Member
<h2>Thêm mới thành viên</h2>
@Html.Raw(ViewBag.msg)
<hr />
<div class="row">
    <div class="col-md-4">
        <form asp-action="Create">
            <div class="form-group">
                <label asp-for="MemberId" class="control-label"></label>
                <input asp-for="MemberId" class="form-control" readonly />
            </div>
            <div class="form-group">
                <label asp-for="Username" class="control-label"></label>
                <input asp-for="Username" class="form-control" />
            </div>
            <div class="form-group">
                <label asp-for="FullName" class="control-label"></label>
                <input asp-for="FullName" class="form-control" />
            </div>
        </form>
    </div>
</div>
```

Manual Vadilation

Code Razor cho View "Create"

```
<div class="form-group">
    <label asp-for="Password" class="control-label"></label>
    <input asp-for="Password" type="password" class="form-control" />
</div>
<div class="form-group">
    <label asp-for="Email" class="control-label"></label>
    <input asp-for="Email" class="form-control" />
</div>
<div class="form-group">
    <label asp-for="Phone" class="control-label"></label>
    <input asp-for="Phone" class="form-control" />
</div>
<div class="form-group">
    <label asp-for="Birthday" class="control-label"></label>
    <input asp-for="Birthday" class="form-control" />
</div>
<div class="form-group">
    <input type="submit" value="Lưu" class="btn btn-primary" />
</div>
</form>
</div>
</div>
```

Manual Validation

Code logic cho Action "Create" với method là POST

```
[HttpPost]
public IActionResult Create(Member member)
{
    string msg = null;
    bool validate = true;
    if (member.Username.Length < 3 || member.Username.Length > 20)
    {
        msg = "<li>Tên đăng nhập phải có độ dài từ 3-20 ký tự</li>";
        validate = false;
    }
    string patternemail = @"^[a-z][a-z0-9_\.\-]{5,32}@([a-z0-9]{2,}(\.[a-z0-9]{2,4}))^{1,2}$";
    if (!Regex.IsMatch(member.Email, patternemail))
    {
        msg += "<li>Email không đúng định dạng</li>";
        validate = false;
    }
    if (member.Birthday.AddYears(18) > DateTime.Today)
    {
        msg += "<li>Bạn chưa đủ 18 tuổi</li>";
        validate = false;
    }
}
```


Manual Validation

Code logic cho Action "Create" với method là POST

```
string patternphone = @"^0\d{9,12}$";
if (!Regex.IsMatch(member.Phone, patternphone))
{
    msg += "<li>Số điện thoại không hợp lệ</li>";
    validate = false;
}
if (validate)
{
    member.MemberId = Guid.NewGuid().ToString();
    members.Add(member);
    return RedirectToAction("Index");
}
else
{
    ViewBag.msg = "<div class='alert alert-danger'>" + msg + "</div>";
    return View(member);
}
```

Manual Vadilation

Thêm mới thành viên

- Email không đúng định dạng
- Bạn chưa đủ 18 tuổi
- Số điện thoại không hợp lệ

MemberId

Username

FullName

Password

Email

Phone

Birthday

Lưu

Data Annotation

- Data Annotation là các lớp thuộc tính được sử dụng để định nghĩa MetaData cho ASP.NET MVC và ASP.NET Data Controls.
- Data Annotation được sử dụng với 3 mục đích

Front-End: Validation Attributes

Sử dụng để kiểm tra tính hợp lệ của dữ liệu trên View như require, email, phone,...

Front-End: Display Attributes

Sử dụng hiển thị tiêu đề các thuộc tính trong Model, có thể sử dụng Resource với Multiple language

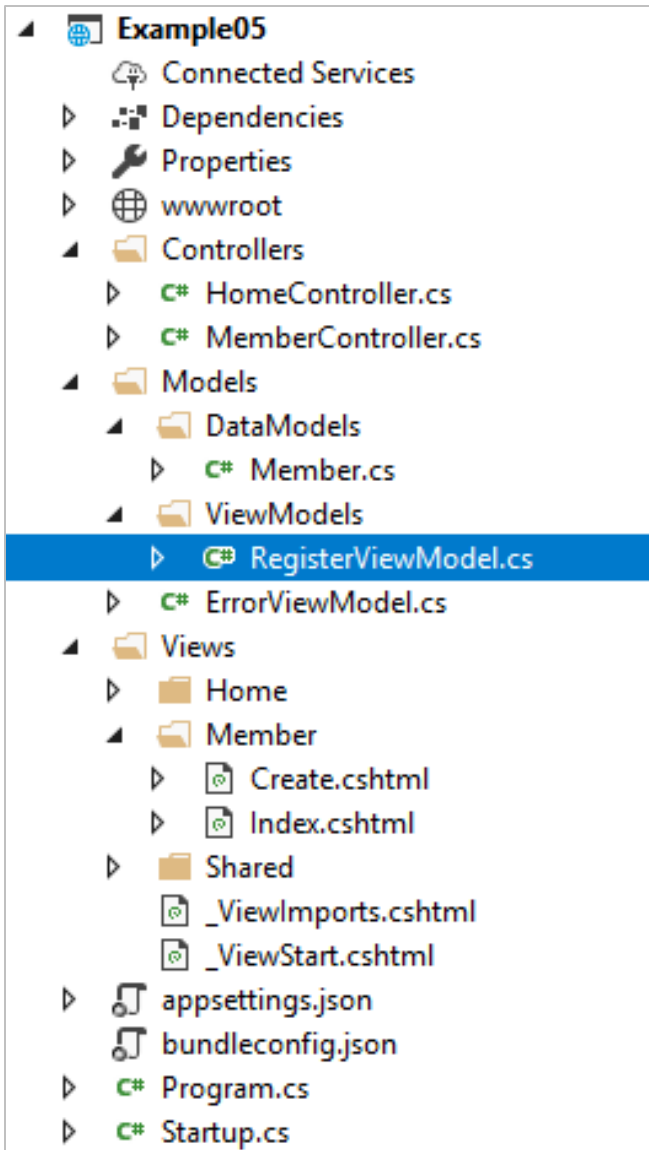
Back-End: Modelling Attributes

Sử dụng trong thiết kế database theo phương pháp CodeFirst để xác định tên bảng, khóa chính, và các quan hệ giữa các bảng, loại dữ liệu, chiều dài...

Attributes

- [Required]
- [Compare]
- [StringLength]
- [Range]
- [RegularExpression]
- [DataType]
- [DisplayName]
- ...

Attributes



```
public class RegisterViewModel
{
    [DisplayName("Tên đăng nhập")]
    [Required(ErrorMessage = "Tên đăng nhập không được trống")]
    [StringLength(20, MinimumLength = 3, ErrorMessage = "Độ dài tên từ 3-20 ký tự")]
    public string Username { get; set; }
    [DisplayName("Họ và tên")]
    [Required(ErrorMessage = "Họ và tên không được trống")]
    public string FullName { get; set; }
    [DisplayName("Mật khẩu")]
    [Required(ErrorMessage = "Hãy nhập password")]
    [DataType(DataType.Password)]
    public string Password { get; set; }
    [DisplayName("Gõ lại mật khẩu")]
    [Compare("Password", ErrorMessage = "Mật khẩu không khớp")]
    [DataType(DataType.Password)]
    public string ConfirmPassword { get; set; }
    [DisplayName("Hòm thư")]
    [DataType(DataType.EmailAddress)]
    [Required(ErrorMessage = "Email không bỏ trống")]
    public string Email { get; set; }
    [DisplayName("Điện thoại")]
    [RegularExpression(@"^\d{9,12}$", ErrorMessage = "Phải bắt đầu bằng 0 và dài 10-12 số")]
    public string Phone { get; set; }
    [DisplayName("Ngày sinh")]
    public DateTime Birthday { get; set; }
}
```

Attributes



```
public class MembersController : Controller
{
    public static readonly List<Member> members = new List<Member>();

    public IActionResult Index()
    {
        return View(members);
    }

    public IActionResult Create()
    {
        return View();
    }

    [HttpPost]
    public IActionResult Create(RegisterViewModel register)...
```

Add View cho 2 Action “Index” và “Create”

Add MVC View

View name:

Template:

Model class:

Options:

☐ Create as a partial view

☒ Reference script libraries

☒ Use a layout page:

(Leave empty if it is set in a Razor _viewstart file)

Add Cancel

Add MVC View

View name:

Template:

Model class:

Options:

☐ Create as a partial view

☒ Reference script libraries

☒ Use a layout page:

(Leave empty if it is set in a Razor _viewstart file)

Add Cancel

Attributes

Code Razor cho View "Create"

```
@model Example05.Models.ViewModels.RegisterViewModel
<h2>ĐĂNG KÝ THÀNH VIÊN</h2>
<hr />
<div class="row">
    <div class="col-md-4">
        <form asp-action="Create">
            <div asp-validation-summary="ModelOnly" class="text-danger"></div>
            <div class="form-group">
                <label asp-for="Username" class="control-label"></label>
                <input asp-for="Username" class="form-control" />
                <span asp-validation-for="Username" class="text-danger"></span>
            </div>
            <div class="form-group">
                <label asp-for="FullName" class="control-label"></label>
                <input asp-for="FullName" class="form-control" />
                <span asp-validation-for="FullName" class="text-danger"></span>
            </div>
            <div class="form-group">
                <label asp-for="Password" class="control-label"></label>
                <input asp-for="Password" class="form-control" />
                <span asp-validation-for="Password" class="text-danger"></span>
            </div>
            <div class="form-group">
                <label asp-for="ConfirmPassword" class="control-label"></label>
                <input asp-for="ConfirmPassword" class="form-control" />
                <span asp-validation-for="ConfirmPassword" class="text-danger"></span>
            </div>
        </form>
    </div>
</div>
```


Attributes

Code Razor cho View "Create"

```
<div class="form-group">
  <label asp-for="Email" class="control-label"></label>
  <input asp-for="Email" class="form-control" />
  <span asp-validation-for="Email" class="text-danger"></span>
</div>
<div class="form-group">
  <label asp-for="Phone" class="control-label"></label>
  <input asp-for="Phone" class="form-control" />
  <span asp-validation-for="Phone" class="text-danger"></span>
</div>
<div class="form-group">
  <label asp-for="Birthday" class="control-label"></label>
  <input asp-for="Birthday" class="form-control" />
  <span asp-validation-for="Birthday" class="text-danger"></span>
</div>
<div class="form-group">
  <input type="submit" value="Lưu" class="btn btn-primary" />
</div>
</form>
</div>
</div>
```

Attributes

Code logic cho Action "Create" với method là POST

```
[HttpPost]
public IActionResult Create(RegisterViewModel register)
{
    if(ModelState.IsValid)
    {
        Member m = new Member
        {
            MemberId = Guid.NewGuid().ToString(),
            Username = register.Username,
            FullName = register.FullName,
            Email = register.Email,
            Password = register.Password,
            Phone = register.Phone,
            Birthday=register.Birthday
        };
        members.Add(m);
        return RedirectToAction("Index");
    }else
    {
        return View(register);
    }
}
```

ĐĂNG KÝ THÀNH VIÊN

Tên đăng nhập

Tên đăng nhập không được trống

Họ và tên

Họ và tên không được trống

Mật khẩu

Hãy nhập password

Gõ lại mật khẩu

Hòm thư

Email không bỏ trống

Điện thoại

Phải bắt đầu bằng 0 và dài 10-12 số

Ngày sinh



Lưu

Lưu ý: trong quá trình validate dữ liệu đầu vào, bạn nên kết hợp cả phía client và server, tùy vào từng nghiệp vụ ứng dụng