

String MinLength and MaxLength validation don't work (asp.net mvc)

I have this class

44

```
using System.ComponentModel.DataAnnotations;
using Argussoft.BI.DAL.Domain.Users;
```

```
namespace Argussoft.BI.DAL.DTOS.UserDTOS
```

```
{
```

```
    public class CreateUserDto
```

```
    {
```

```
        [Required(ErrorMessage = "Введите логин")]
```

```
        [MaxLength(User.Emaillength, ErrorMessage = "Максимальная длина логина 40  
символов")]
```

```
        [RegularExpression(User.NameRegularExpression, ErrorMessage = "Логин может  
содержать только латинские символы, дефисы, подчеркивания, точки")]
```

```
        public string Name { get; set; }
```

```
        [Required(ErrorMessage = "Введите Email")]
```

```
        [MaxLength(User.Emaillength, ErrorMessage = "Максимальная длина адреса  
электронной почты 100 символов")]
```

```
        [RegularExpression(User.EmailRegularExpression, ErrorMessage = "Введите  
корректный адрес электронной почты")]
```

```
        public virtual string Email { get; set; }
```

```
        [Required(ErrorMessage = "Введите имя пользователя")]
```

```
        [MaxLength(User.FullNameLength, ErrorMessage = "Максимальная длина имени  
пользователя 100 символов")]
```

```
        [RegularExpression(User.NameRegularExpression, ErrorMessage = "Имя пользователя  
может содержать только латинские символы, дефисы, подчеркивания, точки")]
```

```
        public virtual string FullName { get; set; }
```

```
        public virtual int Role { get; set; }
```

```
        [RegularExpression(User.PhoneRegularExpression, ErrorMessage = "Введите  
корректный номер телефона")]
```

```
        public virtual string Phone { get; set; }
```

```
        public virtual int Status { get; set; }
```

```
        [Required(ErrorMessage = "Введите пароль")]
```

By using our site, you acknowledge that you have read and understand our [Cookie Policy](#), [Privacy Policy](#), and our [Terms of Service](#).

```

символов"]
    [RegularExpression(User.PasswordRegularExpression, ErrorMessage = "Пароль может
содержать только латинские символы, дефисы, подчеркивания, точки")]
    public virtual string Password { get; set; }
}
}

```

and this form

```

@model Argussoft.BI.DAL.DTOS.UserDTOS.CreateUserDto

@using (Ajax.BeginForm("CreateUser", "User", new AjaxOptions { OnSuccess =
"onSuccessCreateUser" }, new { id = "dialog_form", @class = "form-horizontal" }))
{
    <div class="modal-header">
        <button type="button" class="close" data-dismiss="modal" aria-
hidden="true">&times;</button>
        <h4>Добавить пользователя</h4>
    </div>
    <div class="modal-body">

        <!-- Name -->
        <div class="control-group">
            <label class="control-label" for="@Html.NameFor(model =>
model.Name)">Логин</label>
            <div class="controls">
                <div class="input-prepend">
                    @Html.TextBoxFor(model => model.Name, new { @class = "span3", id =
"input_name" })
                    <br />
                    @Html.ValidationMessageFor(model => model.Name)
                </div>
            </div>
        </div>

        <!-- Email -->
        <div class="control-group">
            <label class="control-label" for="@Html.NameFor(model =>
model.Email)">Email</label>
            <div class="controls">
                <div class="input-prepend">
                    @Html.TextBoxFor(model => model.Email, new { @class = "span3", id =
"input_email" })
                    <br />

```

By using our site, you acknowledge that you have read and understand our Cookie Policy, Privacy Policy, and our Terms of Service.

```

</div>

<!-- FullName -->
<div class="control-group">
  <label class="control-label" for="@Html.NameFor(model =>
model.FullName)">Имя пользователя</label>
  <div class="controls">
    <div class="input-prepend">
      @Html.TextBoxFor(model => model.FullName, new { @class = "span3", id
= "input_full_name" })
      <br />
      @Html.ValidationMessageFor(model => model.FullName)
    </div>
  </div>
</div>

<!-- Role -->
<div class="control-group">
  <label class="control-label" for="@Html.NameFor(model => model.Role)">Роль
пользователя</label>
  <div class="controls">
    <div class="input-prepend">
      @Html.DropDownList("Role", (SelectList)ViewBag.Roles, new{id
="input_role"})
      <br />
      @Html.ValidationMessageFor(model => model.Role)
    </div>
  </div>
</div>

<!-- Phone -->
<div class="control-group">
  <label class="control-label" for="@Html.NameFor(model =>
model.Phone)">Контактный телефон</label>
  <div class="controls">
    <div class="input-prepend">
      @Html.TextBoxFor(model => model.Phone, new { @class = "span3", id =
"input_phone" })
      <br />
      @Html.ValidationMessageFor(model => model.Phone)
    </div>
  </div>
</div>

<!-- Status -->
<div class="control-group">

```

By using our site, you acknowledge that you have read and understand our [Cookie Policy](#), [Privacy Policy](#), and our [Terms of Service](#).

```

        <div class="input-prepend">
            @Html.DropDownList("Status", (SelectList)ViewBag.UserStatuses, new { id
="input_status" })
            <br />
            @Html.ValidationMessageFor(model => model.Status)
        </div>
    </div>
</div>

<!-- Password -->
<div class="control-group">
    <label class="control-label" for="@Html.NameFor(model =>
model.Password)">Пароль пользователя</label>
    <div class="controls">
        <div class="input-prepend">
            @Html.TextBoxFor(model => model.Password, new { @class = "span3", id
= "input_password" })
            <br />
            @Html.ValidationMessageFor(model => model.Password)
        </div>
    </div>
</div>

</div>
<div class="modal-footer">
    <a href="@Url.Action("UserIndex", "User")" class="btn">Отменить</a>
    <button type="submit" id="save_button" class="btn btn-
primary">Сохранить</button>
</div>
}

```

User class:

```

public class User : BaseEntity
{
    public const int NameLength = 40;
    public const int PasswordMinLength = 5;
    public const int PasswordMaxLength = 20;
    public const int FullNameLength = 100;
    public const int EmailLength = 100;
    public const int PhoneLength = 40; //...
}

```

But my validation works only for Name and Password and doesn't work for FullName, Email and Phone. I don't understand why.

By using our site, you acknowledge that you have read and understand our [Cookie Policy](#), [Privacy Policy](#), and our [Terms of Service](#).

What can be a reason?

[c#](#)[asp.net-mvc](#)[asp.net-mvc-3](#)[asp.net-mvc-4](#)

edited Jul 3 '18 at 21:41



Tieson T.

17.9k

3

57

79

asked Aug 16 '13 at 15:23



Heidel

1,307

9

35

60

What is the value of User.EmailLength? – [PW Kad](#) Aug 16 '13 at 15:25

```
public class User : BaseEntity { public const int NameLength = 40; public const int PasswordMinLength = 5; public const int PasswordMaxLength = 20; public const int FullNameLength = 100; public const int EmailLength = 100; public const int PhoneLength = 40; //...} – Heidel Aug 16 '13 at 15:27
```

5 Answers

▲ [MaxLength](#) is used for the Entity Framework to decide how large to make a string value field when it creates the database.

115

From MSDN:



Specifies the maximum length of array or string data allowed in a property.



✓ [StringLength](#) is a data annotation that will be used for validation of user input.

From MSDN:

Specifies the minimum and maximum length of characters that are allowed in a data field.

Non Customized

Use [StringLength]

```
[RegularExpression(@"^[a-zA-Z ]{2,}$", ErrorMessage = "Minimum 2 characters required")]
```

By using our site, you acknowledge that you have read and understand our [Cookie Policy](#), [Privacy Policy](#), and our [Terms of Service](#).

30 is the Max Length
Minimum length = 3

Customized StringLengthAttribute Class

```
public class MyStringLengthAttribute : StringLengthAttribute
{
    public MyStringLengthAttribute(int maximumLength)
        : base(maximumLength)
    {
    }

    public override bool IsValid(object value)
    {
        string val = Convert.ToString(value);
        if (val.Length < base.MinimumLength)
            base.ErrorMessage = "Minimum length should be 3";
        if (val.Length > base.MaximumLength)
            base.ErrorMessage = "Maximum length should be 6";
        return base.IsValid(value);
    }
}

public class MyViewModel
{
    [MyStringLength(6, MinimumLength = 3)]
    public String MyProperty { get; set; }
}
```

edited Aug 16 '13 at 16:03

answered Aug 16 '13 at 15:28



Imad Alazani

5,595 5 28 54

How can I set min and max values in this case? – [Heidel](#) Aug 16 '13 at 15:30

-
- 1 Thank you, it works. But if I need to set different error messages for MinimumLength and MaximumLength, how can I do that? – [Heidel](#) Aug 16 '13 at 15:39

I tried to use your Customized StringLengthAttribute Class, I don't get compilation errors, but it doesn't work again and I don't get any validation error. stackoverflow.com/questions/18307385/... What can be wrong? – [Heidel](#) Aug 19 '13 at 6:15

By using our site, you acknowledge that you have read and understand our [Cookie Policy](#), [Privacy Policy](#), and our [Terms of Service](#).

I post my model and view controller here stackoverflow.com/questions/18307385/... – [Heidel](#) Aug 19 '13 at 6:24

Try using this attribute, for example for password min length:

8

```
[StringLength(100, ErrorMessage = "Максимальная длина пароля 20 символов", MinimumLength = User.PasswordMinLength)]
```

answered Aug 16 '13 at 15:27



[Daniele](#)

1,780 12 24

If I need to set different error messages for MinimumLength and MaximumLength, how can I do that? – [Heidel](#) Aug 16 '13 at 15:43

They do now, with latest version of MVC (and jquery validate packages). [mvc51-release-notes#Unobtrusive](#)

3

Thanks to this [answer](#) for pointing it out!

edited May 23 '17 at 10:31



Community ♦

1 1

answered Jun 26 '14 at 3:42



[Mrchief](#)

62k 16 115 172

Silent downvoters lurk amongst us. – [Mrchief](#) Apr 20 '16 at 14:17

2

```
[StringLength(16, ErrorMessageResourceName =  
    "PasswordMustBeBetweenMinAndMaxCharacters", ErrorMessageResourceType =  
    typeof(Resources.Resource), MinimumLength = 6)]  
[Display(Name = "Password", ResourceType = typeof(Resources.Resource))]  
public string Password { get; set; }
```

Save resource like this

By using our site, you acknowledge that you have read and understand our [Cookie Policy](#), [Privacy Policy](#), and our [Terms of Service](#).

```
"ThePasswordMustBeAtLeastCharactersLong" | "The password must be {1} at least {2}
characters long"
```

answered Dec 12 '15 at 10:33

[Ruben.sar](#)

266 2 8

▲ This can replace the MaxLength and the MinLength

2

```
[StringLength(40, MinimumLength = 10 , ErrorMessage = "Password cannot be longer than 40
characters and less than 10 characters")]
```

answered Nov 19 '16 at 3:35

[amal50](#)

457 2 12 29

1 vb.net syntax <StringLength(20, MinimumLength:=5)> – [JoshYates1980](#) Mar 21 '17 at 21:30 