

Les formulaires en Blazor

les formulaires : création et validation

Bien évidemment en Blazor nous avons des formulaires. ces derniers vont nous permettre d'ajouter ou modifier des données à notre application en se basant sur les modèles.

0.1 Les composants d'un formulaire

- **EditForm** : équivalent à la balise **form** en html, a un paramètre **Model**, va nous permettre de préciser le modèle auquel est rattaché notre formulaire.

Dans un formulaire nous avons :

1. **les composants d'entrée :**

ces composants sont équivalents à nos champs dans un formulaire en html :

- **InputText**
- **InputArea**

2. **DataAnnotationsValidator** : pour indiquer à blazor que dans ce formulaire on veut prendre en compte les attributs de validations

3. **ValidationSummary** : permet d'afficher les message de validation du formulaire

4. **Les évènements :**

- **OnValidSubmit** : la fonction à laquelle est associée ce formulaire s'exécute uniquement lorsque les champs sont valides.
- **OnInvalidSubmit** : la fonction à laquelle est associée ce formulaire s'exécute uniquement lorsque les champs sont invalides.
- **OnSubmit** : Ici la fonction est exécutée dans les 2 cas.

0.2 Exemple de composant

```
@page"/creer"
```

```
<PageTitle>Nouvelle tâche</PageTitle>
```

```
<EditForm Model="@tache" OnvalidSubmit="@Creer">
```

```
<DataAnnotationsValidator/>
```

```
<ValidationSummary/>
```

```
<InputText id="nom" @bind-value="tache.nom" />
<button class="btn btn-primary" type="submit">Creer</button>
</EditForm>
```

```
@code {
private Todo tache = new Todo();
private void Creer()
{
//
}
}
```

Ce formulaire permet d'ajouter des instances de tâches à notre application, il utilise le modèle ci-dessous :

```
public class Todo
{
[Required]

[StringLength(100,ErrorMessage)="Le nom est trop long."]]

public string Nom { get; set;}

}
```

La validation du formulaire de création d'une tâche se base sur les différentes annotations définies sur les champs du modèle.

Data Annotations :

- **Required** : champ requis
- **StringLenth** : taille de la chaîne
- **Range** : valeur comprise dans un ensemble

Travaux Pratiques :