<div class="ps-4 pe-4">

<div class="row c-controls">

<div class="col-md-3 ps-4">

<label>Title</label>

</div>

<div class="col-md-6">

<**StringTextBox** @bind-Value="SearchData.Title" />

</div>

</div>

<**DataGrid** @ref="CrudGrid" **TEntity**="Product" **ConditionExpr**="@(t =>

t.ProductCategoryId == CategoryId)" **Search**="SearchData" **TableHeight**="300">

<**Columns**>

<**GridColumn** **Field**="t => t.Title" />

<**GridColumn** **Field**="t => t.ActiveType.FaText()" />

<**GridColumn** **Field**="t => t.Price.Seprate3Digit()" />

<**GridColumn** **Field**="t => t.TakeoutPrice.Seprate3Digit()" />

<**GridCommandColumns**/>

</**Columns**>

</**DataGrid**>

<**Window** @bind-Status="status" @ref="UpsertWindow" **Title**="Product"

**Style**="width:500px">

<**Content**>

<**CaspianForm** @ref="UpsertForm" **Model**="UpsertData" **OnSubmit**="SetCategory"

**OnValidSubmit**="showConfirmForPrice">

<**CaspianValidationValidator** **ValidatorType**="typeof(ProductService)"

/>

<div class="row c-controls ps-4 pe-4">

<div class="col-md-4">

<label>Title</label>

</div>

<div class="col-md-8">

<**StringTextBox** @bind-Value="UpsertData.Title" />

</div>

<div class="col-md-4">

<label>Price</label>

</div>

<div class="col-md-8">

<**NumericTextBox** @bind-Value="UpsertData.Price"

**OnChange**="@SetTakeoutPrice" />

</div>

<div class="col-md-4">

<label>Take out price</label>

</div>

<div class="col-md-8">

<**NumericTextBox** @bind-Value="UpsertData.TakeoutPrice" />

</div>

<div class="col-md-4">

<label>Meal</label>

</div>

<div class="col-md-8">

<**MultiSelectBox** @bind-Value="UpsertData.Meal" />

</div>

<div class="col-md-4">

<label>Status</label>

</div>

<div class="col-md-8">

<**DropdownList** @bind-Value="UpsertData.ActiveType" />

</div>

<div class="col-md-4"></div>

<div class="col-md-8 align-content-center">

<**FormCommandButton** />

</div>

</div>

</**CaspianForm**>

</**Content**>

</**Window**>

</div>

@code

{

WindowStatus status;

[Parameter]

public int CategoryId { get; set; }

void SetTakeoutPrice()

{

if (UpsertData.TakeoutPrice == 0)

UpsertData.TakeoutPrice = UpsertData.Price;

}

void SetCategory()

{

UpsertData.ProductCategoryId = CategoryId;

}

async Task<bool> showConfirmForPrice(EditContext context)

{

if (UpsertData.Price > 0 || (await Confirm("Do you want to register the

product with zero amount?")))

return true;

return false;

}