

Namespace SewingManager.CoreBusiness

Classes

[Material](#)

Клас, який представляє матеріал у системі.

[MaterialCategory](#)

Клас, який представляє категорію матеріалу в системі.

Enums

[MaterialType](#)

Перелік типів матеріалів.

Class Material

Namespace: [SewingManager.CoreBusiness](#)

Assembly: SewingManager.CoreBusiness.dll

Клас, який представляє матеріал у системі.

```
public class Material
```

Inheritance

[object](#) ← Material

Inherited Members

[object.Equals\(object\)](#) , [object.Equals\(object, object\)](#) , [object.GetHashCode\(\)](#) , [object.GetType\(\)](#) ,
[object.MemberwiseClone\(\)](#) , [object.ReferenceEquals\(object, object\)](#) , [object.ToString\(\)](#)

Properties

CategoryId

Ідентифікатор категорії матеріалу. За замовчуванням порожній рядок.

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

Property Value

[string](#)

Description

Опис матеріалу. За замовчуванням порожній рядок.

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

Property Value

[string](#)

ImageUrl

URL зображення матеріалу (якщо є).

```
public string? ImageUrl { get; set; }
```

Property Value

[string](#)

MaterialCategory

Категорія матеріалу. Цей властивість не серіалізується в JSON.

```
[JsonIgnore]  
public MaterialCategory MaterialCategory { get; set; }
```

Property Value

[MaterialCategory](#)

MaterialCode

Код матеріалу, що є обов'язковим полем для кожного матеріалу.

```
[Required]  
public string MaterialCode { get; set; }
```

Property Value

[string](#)

MaterialId

Унікальний ідентифікатор матеріалу.

```
public int MaterialId { get; set; }
```

Property Value

[int ↗](#)

Name

Назва матеріалу. Це обов'язкове поле з обмеженням довжини до 150 символів.

```
[Required]  
[StringLength(150)]  
public string Name { get; set; }
```

Property Value

[string ↗](#)

Price

Ціна за одиницю матеріалу. Це обов'язкове поле.

```
[Required]  
public double Price { get; set; }
```

Property Value

[double ↗](#)

Quantity

Кількість одиниць матеріалу на складі. Це обов'язкове поле.

```
[Required]
```

```
public double Quantity { get; set; }
```

Property Value

[double](#)

Type

Тип матеріалу. Це обов'язкове поле.

[Required]

```
public MaterialType Type { get; set; }
```

Property Value

[MaterialType](#)

Class MaterialCategory

Namespace: [SewingManager.CoreBusiness](#)

Assembly: SewingManager.CoreBusiness.dll

Клас, який представляє категорію матеріалу в системі.

```
public class MaterialCategory
```

Inheritance

[object](#) ← MaterialCategory

Inherited Members

[object.Equals\(object\)](#) , [object.Equals\(object, object\)](#) , [object.GetHashCode\(\)](#) , [object.GetType\(\)](#) ,
[object.MemberwiseClone\(\)](#) , [object.ReferenceEquals\(object, object\)](#) , [object.ToString\(\)](#)

Properties

MaterialCategoryId

Унікальний ідентифікатор категорії матеріалу.

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

Property Value

[string](#)

Materials

Колекція матеріалів, що належать до цієї категорії.

```
public ICollection<Material> Materials { get; set; }
```

Property Value

Name

Назва категорії матеріалу.

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

Property Value

[string](#) ↴

Enum MaterialType

Namespace: [SewingManager.CoreBusiness](#)

Assembly: SewingManager.CoreBusiness.dll

Перелік типів матеріалів.

```
public enum MaterialType
```

Fields

[Display(Name = "Відрізний")] Cut = 1

Тип матеріалу: Відрізний (Cut).

[Display(Name = "Штучний")] Piece = 0

Тип матеріалу: Штучний (Piece).

Namespace SewingManager.Infrastructure

Classes

[FileStorageService](#)

Сервіс для збереження зображень на файловій системі сервера.

Class FileStorageService

Namespace: [SewingManager.Infrastructure](#)

Assembly: SewingManager.Infrastructure.dll

Сервіс для збереження зображень на файловій системі сервера.

```
public class FileStorageService : IFileStorageService
```

Inheritance

[object](#) ← FileStorageService

Implements

[IFileStorageService](#)

Inherited Members

[object.Equals\(object\)](#) , [object.Equals\(object, object\)](#) , [object.GetHashCode\(\)](#) , [object.GetType\(\)](#) ,
[object.MemberwiseClone\(\)](#) , [object.ReferenceEquals\(object, object\)](#) , [object.ToString\(\)](#)

Methods

SaveImageAsync(IBrowserFile, int, int)

Зберігає завантажене зображення на сервері після його зміни розміру відповідно до заданих параметрів.

```
public Task<string?> SaveImageAsync(IBrowserFile file, int maxWidth, int maxHeight)
```

Parameters

file [IBrowserFile](#)

Завантажений файл зображення з браузера.

maxWidth [int](#)

Максимальна ширина для зміненого зображення.

maxHeight [int](#)

Максимальна висота для зміненого зображення.

Returns

[Task](#) <[string](#)>

Відносна URL-адреса до збереженого зображення (наприклад, "/images/унікальне-ім'я.jpg"), або [null](#), якщо файл є [null](#).

Remarks

Цей метод генерує унікальне ім'я для файлу, щоб уникнути перезапису існуючих файлів. Файл зберігається в директорії "wwwroot/images".

Namespace SewingManager.UseCases. Materials

Classes

[AddMaterialUseCase](#)

Використовуваний випадок для додавання нового матеріалу до системи.

[DeleteMaterialUseCase](#)

Використовуваний випадок для видалення матеріалу з системи.

[EditMaterialUseCase](#)

Використовується для редагування матеріалів, включаючи оновлення зображень.

[ViewMaterialByIdUseCase](#)

Використовуваний випадок для перегляду матеріалу за його ідентифікатором.

[ViewMaterialCategoriesByNameUseCase](#)

Використовуваний випадок для перегляду категорій матеріалів за їх ім'ям.

[ViewMaterialsByNameUseCase](#)

Використовуваний випадок для перегляду матеріалів за їх іменем.

Class AddMaterialUseCase

Namespace: [SewingManager.UseCases.Materials](#)

Assembly: SewingManager.UseCases.dll

Використовуваний випадок для додавання нового матеріалу до системи.

```
public class AddMaterialUseCase : IAddMaterialUseCase
```

Inheritance

[object](#) ← AddMaterialUseCase

Implements

[IAddMaterialUseCase](#)

Inherited Members

[object.Equals\(object\)](#) , [object.Equals\(object, object\)](#) , [object.GetHashCode\(\)](#) , [object.GetType\(\)](#) ,
[object.MemberwiseClone\(\)](#) , [object.ReferenceEquals\(object, object\)](#) , [object.ToString\(\)](#)

Constructors

AddMaterialUseCase(IRepository<IMaterial>, IFileStorageService)

Ініціалізує новий випадок додавання матеріалу.

```
public AddMaterialUseCase(IRepository<IMaterial> materialRepository,  
IFileStorageService fileStorageService)
```

Parameters

materialRepository [IMaterialRepository](#)

Репозиторій для роботи з матеріалами.

fileStorageService [IFileStorageService](#)

Сервіс для зберігання файлів.

Methods

ExecuteAsync(Material, IBrowserFile?)

Додає новий матеріал до системи.

```
public Task ExecuteAsync(Material material, IBrowserFile? file)
```

Parameters

material [Material](#)

Об'єкт матеріалу, який потрібно додати.

file [IBrowserFile](#)

Файл, що супроводжує матеріал (наприклад, зображення або інший документ), або `null`, якщо файл не надано.

Returns

[Task](#)

Завдання, яке виконується асинхронно.

Remarks

Цей метод додає новий матеріал до системи. Якщо файл надано, то він зберігається за допомогою сервісу зберігання файлів, і його URL записується в об'єкт матеріалу.

Class DeleteMaterialUseCase

Namespace: [SewingManager.UseCases.Materials](#)

Assembly: SewingManager.UseCases.dll

Використовуваний випадок для видалення матеріалу з системи.

```
public class DeleteMaterialUseCase : IDeleteMaterialUseCase
```

Inheritance

[object](#) ← DeleteMaterialUseCase

Implements

[IDeleteMaterialUseCase](#)

Inherited Members

[object.Equals\(object\)](#) , [object.Equals\(object, object\)](#) , [object.GetHashCode\(\)](#) , [object.GetType\(\)](#) ,
[object.MemberwiseClone\(\)](#) , [object.ReferenceEquals\(object, object\)](#) , [object.ToString\(\)](#)

Constructors

DeleteMaterialUseCase(IMaterialRepository)

Ініціалізує новий випадок для видалення матеріалу.

```
public DeleteMaterialUseCase(IMaterialRepository materialRepository)
```

Parameters

materialRepository [IMaterialRepository](#)

Репозиторій для роботи з матеріалами.

Methods

ExecuteAsync(int)

Видаляє матеріал з системи за його ідентифікатором.

```
public Task ExecuteAsync(int materialId)
```

Parameters

materialId [int](#)

Ідентифікатор матеріалу, який потрібно видалити.

Returns

[Task](#)

Завдання, яке виконується асинхронно.

Remarks

Цей метод видаляє матеріал із системи, використовуючи його унікальний ідентифікатор.

Class EditMaterialUseCase

Namespace: [SewingManager.UseCases.Materials](#)

Assembly: SewingManager.UseCases.dll

Використовується для редагування матеріалів, включаючи оновлення зображень.

```
public class EditMaterialUseCase : IEditMaterialUseCase
```

Inheritance

[object](#) ← EditMaterialUseCase

Implements

[IEditMaterialUseCase](#)

Inherited Members

[object.Equals\(object\)](#) , [object.Equals\(object, object\)](#) , [object.GetHashCode\(\)](#) , [object.GetType\(\)](#) ,
[object.MemberwiseClone\(\)](#) , [object.ReferenceEquals\(object, object\)](#) , [object.ToString\(\)](#)

Constructors

EditMaterialUseCase([IMaterialRepository](#), [IFileStorageService](#))

Ініціалізує новий екземпляр класу EditMaterialUseCase.

```
public EditMaterialUseCase(IMaterialRepository materialRepository,  
IFileStorageService fileStorageService)
```

Parameters

materialRepository [IMaterialRepository](#)

Репозиторій для роботи з матеріалами.

fileStorageService [IFileStorageService](#)

Сервіс для збереження зображень.

Methods

ExecuteAsync(Material, IBrowserFile?)

Оновлює матеріал у репозиторії та, за наявності, зберігає нове зображення.

```
public Task ExecuteAsync(Material material, IBrowserFile? file)
```

Parameters

material [Material](#)

Модель матеріалу з оновленими даними.

file [IBrowserFile](#)

Файл зображення для збереження (може бути null).

Returns

[Task](#)

Завдання, що завершується після оновлення матеріалу.

Examples

```
var material = new Material { MaterialId = 1, Name = "Бавовна оновлена" }; var file = await
GetFileFromInput(); // Отримати файл із Blazor InputFile await
editMaterialUseCase.ExecuteAsync(material, file);
```

Class ViewMaterialByIdUseCase

Namespace: [SewingManager.UseCases.Materials](#)

Assembly: SewingManager.UseCases.dll

Використовуваний випадок для перегляду матеріалу за його ідентифікатором.

```
public class ViewMaterialByIdUseCase : IViewMaterialByIdUseCase
```

Inheritance

[object](#) ← ViewMaterialByIdUseCase

Implements

[IViewMaterialByIdUseCase](#)

Inherited Members

[object.Equals\(object\)](#) , [object.Equals\(object, object\)](#) , [object.GetHashCode\(\)](#) , [object.GetType\(\)](#) ,
[object.MemberwiseClone\(\)](#) , [object.ReferenceEquals\(object, object\)](#) , [object.ToString\(\)](#)

Constructors

ViewMaterialByIdUseCase(IMaterialRepository)

Ініціалізує новий випадок для перегляду матеріалу за його ідентифікатором.

```
public ViewMaterialByIdUseCase(IMaterialRepository materialRepository)
```

Parameters

materialRepository [IMaterialRepository](#)

Репозиторій для роботи з матеріалами.

Methods

ExecuteAsync(int)

Отримує матеріал за його ідентифікатором.

```
public Task<Material> ExecuteAsync(int materialId)
```

Parameters

materialId [int](#)

Ідентифікатор матеріалу, який потрібно переглянути.

Returns

[Task](#) <[Material](#)>

Завдання, яке повертає об'єкт матеріалу, якщо матеріал знайдений.

Remarks

Цей метод повертає матеріал з системи, використовуючи його унікальний ідентифікатор.

Class ViewMaterialCategoriesByNameUseCase

Namespace: [SewingManager.UseCases.Materials](#)

Assembly: SewingManager.UseCases.dll

Використовуваний випадок для перегляду категорій матеріалів за їх ім'ям.

```
public class ViewMaterialCategoriesByNameUseCase : IViewMaterialCategoriesByNameUseCase
```

Inheritance

[object](#) ← ViewMaterialCategoriesByNameUseCase

Implements

[IViewMaterialCategoriesByNameUseCase](#)

Inherited Members

[object.Equals\(object\)](#) , [object.Equals\(object, object\)](#) , [object.GetHashCode\(\)](#) , [object.GetType\(\)](#) ,
[object.MemberwiseClone\(\)](#) , [object.ReferenceEquals\(object, object\)](#) , [object.ToString\(\)](#)

Constructors

ViewMaterialCategoriesByNameUseCase(IMaterialRepository)

Ініціалізує новий випадок для перегляду категорій матеріалів за їх ім'ям.

```
public ViewMaterialCategoriesByNameUseCase(IMaterialRepository materialRepository)
```

Parameters

materialRepository [IMaterialRepository](#)

Репозиторій для роботи з категоріями матеріалів.

Methods

ExecuteAsync(string)

Отримує категорії матеріалів за їх іменем.

```
public Task<IEnumerable<MaterialCategory>> ExecuteAsync(string name = "")
```

Parameters

name [string](#)

Ім'я категорії для фільтрації результатів. Якщо не вказано, повертаються всі категорії.

Returns

[Task](#)<[IEnumerable](#)<[MaterialCategory](#)>>

Завдання, яке повертає колекцію категорій матеріалів, що відповідають вказаному імені.

Remarks

Цей метод дозволяє отримати категорії матеріалів, відфільтровані за ім'ям. Якщо параметр **name** не вказано або порожній, будуть повернуті всі категорії.

Class ViewMaterialsByNameUseCase

Namespace: [SewingManager.UseCases.Materials](#)

Assembly: SewingManager.UseCases.dll

Використовуваний випадок для перегляду матеріалів за їх іменем.

```
public class ViewMaterialsByNameUseCase : IViewMaterialsByNameUseCase
```

Inheritance

[object](#) ← ViewMaterialsByNameUseCase

Implements

[IViewMaterialsByNameUseCase](#)

Inherited Members

[object.Equals\(object\)](#) , [object.Equals\(object, object\)](#) , [object.GetHashCode\(\)](#) , [object.GetType\(\)](#) ,
[object.MemberwiseClone\(\)](#) , [object.ReferenceEquals\(object, object\)](#) , [object.ToString\(\)](#)

Constructors

ViewMaterialsByNameUseCase(IMaterialRepository)

Ініціалізує новий випадок для перегляду матеріалів за їх іменем.

```
public ViewMaterialsByNameUseCase(IMaterialRepository materialRepository)
```

Parameters

materialRepository [IMaterialRepository](#)

Репозиторій для роботи з матеріалами.

Methods

ExecuteAsync(string)

Отримує матеріали за їх іменем.

```
public Task<IEnumerable<Material>> ExecuteAsync(string name = "")
```

Parameters

name [string](#)

Ім'я матеріалу для фільтрації результатів. Якщо не вказано, повертаються всі матеріали.

Returns

[Task](#)<[IEnumerable](#)<[Material](#)>>

Завдання, яке повертає колекцію матеріалів, що відповідають вказаному імені.

Remarks

Цей метод дозволяє отримати матеріали, відфільтровані за ім'ям. Якщо параметр **name** не вказано або порожній, будуть повернуті всі матеріали.

Namespace SewingManager.UseCases. Materials.Interfaces

Interfaces

[IAddMaterialUseCase](#)

Інтерфейс для додавання нового матеріалу в систему.

[IDeleteMaterialUseCase](#)

Інтерфейс для видалення матеріалу з системи.

[IEditMaterialUseCase](#)

Інтерфейс для редагування інформації про матеріал у системі.

[IFileStorageService](#)

Інтерфейс для зберігання файлів на сервері.

[IViewMaterialByIdUseCase](#)

Інтерфейс для перегляду матеріалу за його ідентифікатором.

[IViewMaterialCategoriesByNameUseCase](#)

Інтерфейс для перегляду категорій матеріалів за їх ім'ям.

[IViewMaterialsByNameUseCase](#)

Інтерфейс для перегляду матеріалів за їх іменем.

Interface IAddMaterialUseCase

Namespace: [SewingManager.UseCases.Materials.Interfaces](#)

Assembly: SewingManager.UseCases.dll

Інтерфейс для додавання нового матеріалу в систему.

```
public interface IAddMaterialUseCase
```

Methods

ExecuteAsync(Material, IBrowserFile?)

Виконує операцію додавання нового матеріалу в систему.

```
Task ExecuteAsync(Material material, IBrowserFile? file)
```

Parameters

material [Material](#)

Об'єкт матеріалу, який потрібно додати.

file [IBrowserFile](#)

Файл, що супроводжує матеріал (наприклад, зображення або інший документ), або **null**, якщо файл не надано.

Returns

[Task](#)

Завдання, яке виконується асинхронно.

Remarks

Цей метод обробляє додавання матеріалу до системи, включаючи можливість додавання супутнього файлу.

Interface IDeleteMaterialUseCase

Namespace: [SewingManager.UseCases.Materials.Interfaces](#)

Assembly: SewingManager.UseCases.dll

Інтерфейс для видалення матеріалу з системи.

```
public interface IDeleteMaterialUseCase
```

Methods

ExecuteAsync(int)

Виконує операцію видалення матеріалу за його ідентифікатором.

```
Task ExecuteAsync(int materialId)
```

Parameters

materialId [int](#)

Ідентифікатор матеріалу, який потрібно видалити.

Returns

[Task](#)

Завдання, яке виконується асинхронно.

Remarks

Цей метод забезпечує видалення матеріалу з системи за допомогою наданого ідентифікатора.

Interface IEditMaterialUseCase

Namespace: [SewingManager.UseCases.Materials.Interfaces](#)

Assembly: SewingManager.UseCases.dll

Інтерфейс для редагування інформації про матеріал у системі.

```
public interface IEditMaterialUseCase
```

Methods

ExecuteAsync(Material, IBrowserFile?)

Виконує операцію редагування існуючого матеріалу в системі.

```
Task ExecuteAsync(Material material, IBrowserFile? file)
```

Parameters

material [Material](#)

Об'єкт матеріалу, який потрібно оновити.

file [IBrowserFile](#)

Файл, що супроводжує матеріал (наприклад, зображення або інший документ), або **null**, якщо файл не надано.

Returns

[Task](#)

Завдання, яке виконується асинхронно.

Remarks

Цей метод дозволяє редагувати дані матеріалу, а також оновлювати супутні файли, якщо вони надані.

Interface IFileStorageService

Namespace: [SewingManager.UseCases.Materials.Interfaces](#)

Assembly: SewingManager.UseCases.dll

Інтерфейс для зберігання файлів на сервері.

```
public interface IFileStorageService
```

Methods

SaveImageAsync(IBrowserFile, int, int)

Зберігає зображення на сервері після його зміни розміру до заданих розмірів.

```
Task<string?> SaveImageAsync(IBrowserFile file, int maxWidth, int maxHeight)
```

Parameters

file [IBrowserFile](#)

Завантажене зображення з браузера.

maxWidth [int](#)

Максимальна ширина для зміненої картинки.

maxHeight [int](#)

Максимальна висота для зміненої картинки.

Returns

[Task](#) <[string](#)>

Відносна URL-адреса збереженого зображення (наприклад, "/images/унікальне-ім'я.jpg"), або **null**, якщо файл є **null**.

Remarks

Цей метод зберігає файл з унікальним ім'ям у директорії для зображень на сервері.

Interface IViewMaterialByIdUseCase

Namespace: [SewingManager.UseCases.Materials.Interfaces](#)

Assembly: SewingManager.UseCases.dll

Інтерфейс для перегляду матеріалу за його ідентифікатором.

```
public interface IViewMaterialByIdUseCase
```

Methods

ExecuteAsync(int)

Виконує операцію отримання матеріалу за його ідентифікатором.

```
Task<Material> ExecuteAsync(int materialId)
```

Parameters

materialId [int](#)

Ідентифікатор матеріалу, який потрібно переглянути.

Returns

[Task](#) <[Material](#)>

Завдання, яке повертає об'єкт матеріалу, якщо матеріал знайдений.

Remarks

Цей метод дозволяє отримати інформацію про конкретний матеріал із системи за його унікальним ідентифікатором.

Interface IViewMaterialCategoriesByNameUseCase

Namespace: [SewingManager.UseCases.Materials.Interfaces](#)

Assembly: SewingManager.UseCases.dll

Інтерфейс для перегляду категорій матеріалів за їх ім'ям.

```
public interface IViewMaterialCategoriesByNameUseCase
```

Methods

ExecuteAsync(string)

Виконує операцію отримання категорій матеріалів за їх ім'ям.

```
Task<IEnumerable<MaterialCategory>> ExecuteAsync(string name = "")
```

Parameters

name [string](#)

Ім'я категорії для фільтрації результатів. Якщо не вказано, повертаються всі категорії.

Returns

[Task](#) <[IEnumerable](#) <[MaterialCategory](#)>>

Завдання, яке повертає колекцію категорій матеріалів, що відповідають вказаному імені.

Remarks

Цей метод дозволяє отримати категорії матеріалів, відфільтровані за ім'ям. Якщо параметр `name` не вказано або порожній, будуть повернуті всі категорії.

Interface IViewMaterialsByNameUseCase

Namespace: [SewingManager.UseCases.Materials.Interfaces](#)

Assembly: SewingManager.UseCases.dll

Інтерфейс для перегляду матеріалів за їх іменем.

```
public interface IViewMaterialsByNameUseCase
```

Methods

ExecuteAsync(string)

Виконує операцію отримання матеріалів за їх іменем.

```
Task<IEnumerable<Material>> ExecuteAsync(string name = "")
```

Parameters

`name` [string](#)

Ім'я матеріалу для фільтрації результатів. Якщо не вказано, повертаються всі матеріали.

Returns

[Task](#) <[IEnumerable](#) <[Material](#)>>

Завдання, яке повертає колекцію матеріалів, що відповідають вказаному імені.

Remarks

Цей метод дозволяє отримати матеріали, відфільтровані за ім'ям. Якщо параметр `name` не вказано або порожній, будуть повернуті всі матеріали.

Namespace SewingManager.UseCases.Plugin Interfaces

Interfaces

[IMaterialRepository](#)

Репозиторій для роботи з матеріалами.

Interface IMaterialRepository

Namespace: [SewingManager.UseCases.PluginInterfaces](#)

Assembly: SewingManager.UseCases.dll

Репозиторій для роботи з матеріалами.

```
public interface IMaterialRepository
```

Methods

AddMaterialAsync(Material)

Додає новий матеріал.

```
Task AddMaterialAsync(Material material)
```

Parameters

material [Material](#)

Об'єкт матеріалу, який потрібно додати.

Returns

[Task](#) ↗

Завдання, яке виконується асинхронно.

DeleteMaterialByIdAsync(int)

Видаляє матеріал за його ідентифікатором.

```
Task DeleteMaterialByIdAsync(int materialId)
```

Parameters

materialId [int](#)

Ідентифікатор матеріалу, який потрібно видалити.

Returns

[Task](#)

Завдання, яке виконується асинхронно.

GetMaterialByIdAsync(int)

Отримує матеріал за його ідентифікатором.

`Task<Material> GetMaterialByIdAsync(int materialId)`

Parameters

materialId [int](#)

Ідентифікатор матеріалу, який потрібно переглянути.

Returns

[Task](#) <[Material](#)>

Завдання, яке повертає матеріал за вказаним ідентифікатором.

GetMaterialCategoriesByNameUseCase(string)

Отримує категорії матеріалів за їх іменем.

`Task<IEnumerable<MaterialCategory>> GetMaterialCategoriesByNameUseCase(string name)`

Parameters

name [string](#)

Ім'я категорії для фільтрації результатів. Якщо не вказано, повертаються всі категорії.

Returns

[Task](#) <IEnumerable<[MaterialCategory](#)>>

Завдання, яке повертає колекцію категорій матеріалів, що відповідають вказаному імені.

GetMaterialsByNameAsync(string)

Отримує матеріали за їх іменем.

`Task<IEnumerable<Material>> GetMaterialsByNameAsync(string name)`

Parameters

`name` [string](#)

Ім'я матеріалу для фільтрації результатів. Якщо не вказано, повертаються всі матеріали.

Returns

[Task](#) <IEnumerable<[Material](#)>>

Завдання, яке повертає колекцію матеріалів, що відповідають вказаному імені.

UpdateMaterialAsync(Material)

Оновлює існуючий матеріал.

`Task UpdateMaterialAsync(Material material)`

Parameters

`material` [Material](#)

Об'єкт матеріалу з оновленими даними.

Returns

[Task](#)

Завдання, яке виконується асинхронно.

Namespace SewingManager.Webapp.

Components

Classes

[App](#)

[Routes](#)

[Imports](#)

Class App

Namespace: [SewingManager.Webapp.Components](#)

Assembly: SewingManager.Webapp.dll

```
public class App : ComponentBase, IComponent, IHandleEvent, IHandleAfterRender
```

Inheritance

[object](#) ← [ComponentBase](#) ← App

Implements

[IComponent](#), [IHandleEvent](#), [IHandleAfterRender](#)

Inherited Members

[ComponentBase.OnInitialized\(\)](#), [ComponentBase.OnInitializedAsync\(\)](#),
[ComponentBase.OnParametersSet\(\)](#), [ComponentBase.OnParametersSetAsync\(\)](#),
[ComponentBase.StateHasChanged\(\)](#), [ComponentBase.ShouldRender\(\)](#),
[ComponentBase.OnAfterRender\(bool\)](#), [ComponentBase.OnAfterRenderAsync\(bool\)](#),
[ComponentBase.InvokeAsync\(Action\)](#), [ComponentBase.InvokeAsync\(Func<Task>\)](#),
[ComponentBase.DispatchExceptionAsync\(Exception\)](#),
[ComponentBase.SetParametersAsync\(ParameterView\)](#), [object.Equals\(object\)](#),
[object.Equals\(object, object\)](#), [object.GetHashCode\(\)](#), [object.GetType\(\)](#),
[object.MemberwiseClone\(\)](#), [object.ReferenceEquals\(object, object\)](#), [object.ToString\(\)](#)

Methods

BuildRenderTree(RenderTreeBuilder)

Renders the component to the supplied [RenderTreeBuilder](#).

```
protected override void BuildRenderTree(RenderTreeBuilder __builder)
```

Parameters

__builder [RenderTreeBuilder](#)

Class Routes

Namespace: [SewingManager.Webapp.Components](#)

Assembly: SewingManager.Webapp.dll

```
public class Routes : ComponentBase, IComponent, IHandleEvent, IHandleAfterRender
```

Inheritance

[object](#) ← [ComponentBase](#) ← Routes

Implements

[IComponent](#), [IHandleEvent](#), [IHandleAfterRender](#)

Inherited Members

[ComponentBase.OnInitialized\(\)](#), [ComponentBase.OnInitializedAsync\(\)](#),
[ComponentBase.OnParametersSet\(\)](#), [ComponentBase.OnParametersSetAsync\(\)](#),
[ComponentBase.StateHasChanged\(\)](#), [ComponentBase.ShouldRender\(\)](#),
[ComponentBase.OnAfterRender\(bool\)](#), [ComponentBase.OnAfterRenderAsync\(bool\)](#),
[ComponentBase.InvokeAsync\(Action\)](#), [ComponentBase.InvokeAsync\(Func<Task>\)](#),
[ComponentBase.DispatchExceptionAsync\(Exception\)](#),
[ComponentBase.SetParametersAsync\(ParameterView\)](#), [object.Equals\(object\)](#),
[object.Equals\(object, object\)](#), [object.GetHashCode\(\)](#), [object.GetType\(\)](#),
[object.MemberwiseClone\(\)](#), [object.ReferenceEquals\(object, object\)](#), [object.ToString\(\)](#)

Methods

BuildRenderTree(RenderTreeBuilder)

Renders the component to the supplied [RenderTreeBuilder](#).

```
protected override void BuildRenderTree(RenderTreeBuilder __builder)
```

Parameters

__builder [RenderTreeBuilder](#)

Class _Imports

Namespace: [SewingManager.Webapp.Components](#)

Assembly: SewingManager.Webapp.dll

```
public class _Imports : ComponentBase, IComponent, IHandleEvent, IHandleAfterRender
```

Inheritance

[object](#) ← [ComponentBase](#) ← [_Imports](#)

Implements

[IComponent](#), [IHandleEvent](#), [IHandleAfterRender](#)

Inherited Members

[ComponentBase.OnInitialized\(\)](#), [ComponentBase.OnInitializedAsync\(\)](#),
[ComponentBase.OnParametersSet\(\)](#), [ComponentBase.OnParametersSetAsync\(\)](#),
[ComponentBase.StateHasChanged\(\)](#), [ComponentBase.ShouldRender\(\)](#),
[ComponentBase.OnAfterRender\(bool\)](#), [ComponentBase.OnAfterRenderAsync\(bool\)](#),
[ComponentBase.InvokeAsync\(Action\)](#), [ComponentBase.InvokeAsync\(Func<Task>\)](#),
[ComponentBase.DispatchExceptionAsync\(Exception\)](#),
[ComponentBase.SetParametersAsync\(ParameterView\)](#), [object.Equals\(object\)](#),
[object.Equals\(object, object\)](#), [object.GetHashCode\(\)](#), [object.GetType\(\)](#),
[object.MemberwiseClone\(\)](#), [object.ReferenceEquals\(object, object\)](#), [object.ToString\(\)](#)

Methods

BuildRenderTree(RenderTreeBuilder)

Renders the component to the supplied [RenderTreeBuilder](#).

```
protected override void BuildRenderTree(RenderTreeBuilder __builder)
```

Parameters

[__builder](#) [RenderTreeBuilder](#)

Namespace SewingManager.Webapp. Components.Controls.Common Classes

[EnumToggleComponent<TEnum>](#)

[ImageUploaderComponent](#)

[MaterialListComponent](#)

[MaterialListItemComponent](#)

[SearchComponent](#)

Class EnumToggleComponent<TEnum>

Namespace: [SewingManager.Webapp.Components.Controls.Common](#)

Assembly: SewingManager.Webapp.dll

```
public class EnumToggleComponent<TEnum> : ComponentBase, IComponent, IHandleEvent,  
IHandleAfterRender where TEnum : struct, Enum
```

Type Parameters

TEnum

Inheritance

[object](#) ← [ComponentBase](#) ← EnumToggleComponent<TEnum>

Implements

[IComponent](#), [IHandleEvent](#), [IHandleAfterRender](#)

Inherited Members

[ComponentBase.OnInitialized\(\)](#), [ComponentBase.OnInitializedAsync\(\)](#),
[ComponentBase.OnParametersSet\(\)](#), [ComponentBase.OnParametersSetAsync\(\)](#),
[ComponentBase.StateHasChanged\(\)](#), [ComponentBase.ShouldRender\(\)](#),
[ComponentBase.OnAfterRender\(bool\)](#), [ComponentBase.OnAfterRenderAsync\(bool\)](#),
[ComponentBase.InvokeAsync\(Action\)](#), [ComponentBase.InvokeAsync\(Func<Task>\)](#),
[ComponentBase.DispatchExceptionAsync\(Exception\)](#),
[ComponentBase.SetParametersAsync\(ParameterView\)](#), [object.Equals\(object\)](#),
[object.Equals\(object, object\)](#), [object.GetHashCode\(\)](#), [object.GetType\(\)](#),
[object.MemberwiseClone\(\)](#), [object.ReferenceEquals\(object, object\)](#), [object.ToString\(\)](#)

Properties

GroupName

```
[Parameter]  
public string GroupName { get; set; }
```

Property Value

[string](#)

Value

```
[Parameter]  
public TEnum? Value { get; set; }
```

Property Value

TEnum?

ValueChanged

```
[Parameter]  
public EventCallback<TEnum> ValueChanged { get; set; }
```

Property Value

[EventCallback](#)<TEnum>

Methods

BuildRenderTree(RenderTreeBuilder)

Renders the component to the supplied [RenderTreeBuilder](#).

```
protected override void BuildRenderTree(RenderTreeBuilder __builder)
```

Parameters

__builder [RenderTreeBuilder](#)

Class ImageUploaderComponent

Namespace: [SewingManager.Webapp.Components.Controls.Common](#)

Assembly: SewingManager.Webapp.dll

```
public class ImageUploaderComponent : ComponentBase, IComponent,
IHandleEvent, IHandleAfterRender
```

Inheritance

[object](#) ← [ComponentBase](#) ← ImageUploaderComponent

Implements

[IComponent](#), [IHandleEvent](#), [IHandleAfterRender](#)

Inherited Members

[ComponentBase.OnInitialized\(\)](#), [ComponentBase.OnInitializedAsync\(\)](#),
[ComponentBase.OnParametersSet\(\)](#), [ComponentBase.OnParametersSetAsync\(\)](#),
[ComponentBase.StateHasChanged\(\)](#), [ComponentBase.ShouldRender\(\)](#),
[ComponentBase.OnAfterRender\(bool\)](#), [ComponentBase.OnAfterRenderAsync\(bool\)](#),
[ComponentBase.InvokeAsync\(Action\)](#), [ComponentBase.InvokeAsync\(Func<Task>\)](#),
[ComponentBase.DispatchExceptionAsync\(Exception\)](#),
[ComponentBase.SetParametersAsync\(ParameterView\)](#), [object.Equals\(object\)](#),
[object.Equals\(object, object\)](#), [object.GetHashCode\(\)](#), [object.GetType\(\)](#),
[object.MemberwiseClone\(\)](#), [object.ReferenceEquals\(object, object\)](#), [object.ToString\(\)](#)

Properties

ImageUrl

URL раніше завантаженого зображення.

```
[Parameter]
public string? ImageUrl { get; set; }
```

Property Value

[string](#)

OnImageSelected

Подія, що викликається при виборі або очищенні зображення.

[Parameter]

```
public EventCallback<IBrowserFile> OnImageSelected { get; set; }
```

Property Value

[EventCallback](#)<[IBrowserFile](#)>

Methods

BuildRenderTree(RenderTreeBuilder)

Renders the component to the supplied [RenderTreeBuilder](#).

```
protected override void BuildRenderTree(RenderTreeBuilder __builder)
```

Parameters

__builder [RenderTreeBuilder](#)

Class MaterialListComponent

Namespace: [SewingManager.Webapp.Components.Controls.Common](#)

Assembly: SewingManager.Webapp.dll

```
public class MaterialListComponent : ComponentBase, IComponent,
IHandleEvent, IHandleAfterRender
```

Inheritance

[object](#) ← [ComponentBase](#) ← MaterialListComponent

Implements

[IComponent](#), [IHandleEvent](#), [IHandleAfterRender](#)

Inherited Members

[ComponentBase.OnInitialized\(\)](#), [ComponentBase.OnInitializedAsync\(\)](#),
[ComponentBase.OnParametersSet\(\)](#), [ComponentBase.StateHasChanged\(\)](#),
[ComponentBase.ShouldRender\(\)](#), [ComponentBase.OnAfterRender\(bool\)](#),
[ComponentBase.OnAfterRenderAsync\(bool\)](#), [ComponentBase.InvokeAsync\(Action\)](#),
[ComponentBase.InvokeAsync\(Func<Task>\)](#), [ComponentBase.DispatchExceptionAsync\(Exception\)](#),
[ComponentBase.SetParametersAsync\(ParameterView\)](#), [object.Equals\(object\)](#),
[object.Equals\(object, object\)](#), [object.GetHashCode\(\)](#), [object.GetType\(\)](#),
[object.MemberwiseClone\(\)](#), [object.ReferenceEquals\(object, object\)](#), [object.ToString\(\)](#)

Properties

SearchInventoryFilter

Фільтр для пошуку матеріалів за назвою, який передається з батьківського компонента.

```
[Parameter]
public string? SearchInventoryFilter { get; set; }
```

Property Value

[string](#)

Methods

BuildRenderTree(RenderTreeBuilder)

Renders the component to the supplied [RenderTreeBuilder](#).

```
protected override void BuildRenderTree(RenderTreeBuilder __builder)
```

Parameters

__builder [RenderTreeBuilder](#)

OnParametersSetAsync()

Метод, який викликається при оновленні параметрів компонента. Використовується для завантаження списку матеріалів на основі пошукового фільтра.

```
protected override Task OnParametersSetAsync()
```

Returns

[Task](#)

Завдання, що виконується асинхронно.

Class MaterialListItemComponent

Namespace: [SewingManager.Webapp.Components.Controls.Common](#)

Assembly: SewingManager.Webapp.dll

```
public class MaterialListItemComponent : ComponentBase, IComponent,
IHandleEvent, IHandleAfterRender
```

Inheritance

[object](#) ← [ComponentBase](#) ← MaterialListItemComponent

Implements

[IComponent](#), [IHandleEvent](#), [IHandleAfterRender](#)

Inherited Members

[ComponentBase.OnInitialized\(\)](#), [ComponentBase.OnInitializedAsync\(\)](#),
[ComponentBase.OnParametersSet\(\)](#), [ComponentBase.OnParametersSetAsync\(\)](#),
[ComponentBase.StateHasChanged\(\)](#), [ComponentBase.ShouldRender\(\)](#),
[ComponentBase.OnAfterRender\(bool\)](#), [ComponentBase.OnAfterRenderAsync\(bool\)](#),
[ComponentBase.InvokeAsync\(Action\)](#), [ComponentBase.InvokeAsync\(Func<Task>\)](#),
[ComponentBase.DispatchExceptionAsync\(Exception\)](#),
[ComponentBase.SetParametersAsync\(ParameterView\)](#), [object.Equals\(object\)](#),
[object.Equals\(object, object\)](#), [object.GetHashCode\(\)](#), [object.GetType\(\)](#),
[object.MemberwiseClone\(\)](#), [object.ReferenceEquals\(object, object\)](#), [object.ToString\(\)](#)

Properties

Material

Матеріал, який відображається в таблиці.

```
[Parameter]
public Material? Material { get; set; }
```

Property Value

[Material](#)

Methods

BuildRenderTree(RenderTreeBuilder)

Renders the component to the supplied [RenderTreeBuilder](#).

```
protected override void BuildRenderTree(RenderTreeBuilder __builder)
```

Parameters

__builder [RenderTreeBuilder](#)

Class SearchComponent

Namespace: [SewingManager.Webapp.Components.Controls.Common](#)

Assembly: SewingManager.Webapp.dll

```
public class SearchComponent : ComponentBase, IComponent, IHandleEvent, IHandleAfterRender
```

Inheritance

[object](#) ← [ComponentBase](#) ← SearchComponent

Implements

[IComponent](#), [IHandleEvent](#), [IHandleAfterRender](#)

Inherited Members

[ComponentBase.OnInitialized\(\)](#), [ComponentBase.OnInitializedAsync\(\)](#),
[ComponentBase.OnParametersSet\(\)](#), [ComponentBase.OnParametersSetAsync\(\)](#),
[ComponentBase.StateHasChanged\(\)](#), [ComponentBase.ShouldRender\(\)](#),
[ComponentBase.OnAfterRender\(bool\)](#), [ComponentBase.OnAfterRenderAsync\(bool\)](#),
[ComponentBase.InvokeAsync\(Action\)](#), [ComponentBase.InvokeAsync\(Func<Task>\)](#),
[ComponentBase.DispatchExceptionAsync\(Exception\)](#),
[ComponentBase.SetParametersAsync\(ParameterView\)](#), [object.Equals\(object\)](#),
[object.Equals\(object, object\)](#), [object.GetHashCode\(\)](#), [object.GetType\(\)](#),
[object.MemberwiseClone\(\)](#), [object.ReferenceEquals\(object, object\)](#), [object.ToString\(\)](#)

Properties

OnSearch

Подія, яка викликається при натисканні на кнопку пошуку або натисканні Enter в полі вводу.

```
[Parameter]  
public EventCallback<string> OnSearch { get; set; }
```

Property Value

[EventCallback](#) <[string](#)>

Methods

BuildRenderTree(RenderTreeBuilder)

Renders the component to the supplied [RenderTreeBuilder](#).

```
protected override void BuildRenderTree(RenderTreeBuilder __builder)
```

Parameters

__builder [RenderTreeBuilder](#)

Namespace SewingManager.Webapp. Components.Layout

Classes

[MainLayout](#)

[NavMenu](#)

Class MainLayout

Namespace: [SewingManager.Webapp.Components.Layout](#)

Assembly: SewingManager.Webapp.dll

```
public class MainLayout : LayoutComponentBase, IComponent, IHandleEvent, IHandleAfterRender
```

Inheritance

[object](#) ← [ComponentBase](#) ← [LayoutComponentBase](#) ← MainLayout

Implements

[IComponent](#), [IHandleEvent](#), [IHandleAfterRender](#)

Inherited Members

[LayoutComponentBase.SetParametersAsync\(ParameterView\)](#), [LayoutComponentBase.Body](#),
[ComponentBase.OnInitializedAsync\(\)](#), [ComponentBase.OnParametersSet\(\)](#),
[ComponentBase.OnParametersSetAsync\(\)](#), [ComponentBase.StateHasChanged\(\)](#),
[ComponentBase.ShouldRender\(\)](#), [ComponentBase.OnAfterRenderAsync\(bool\)](#),
[ComponentBase.InvokeAsync\(Action\)](#), [ComponentBase.InvokeAsync\(Func<Task>\)](#),
[ComponentBase.DispatchExceptionAsync\(Exception\)](#), [object.Equals\(object\)](#),
[object.Equals\(object, object\)](#), [object.GetHashCode\(\)](#), [object.GetType\(\)](#),
[object.MemberwiseClone\(\)](#), [object.ReferenceEquals\(object, object\)](#), [object.ToString\(\)](#)

Methods

BuildRenderTree(RenderTreeBuilder)

Renders the component to the supplied [RenderTreeBuilder](#).

```
protected override void BuildRenderTree(RenderTreeBuilder __builder)
```

Parameters

__builder [RenderTreeBuilder](#)

OnAfterRender(bool)

Method invoked after each time the component has rendered interactively and the UI has finished updating (for example, after elements have been added to the browser DOM). Any [ElementReference](#) fields will be populated by the time this runs.

This method is not invoked during prerendering or server-side rendering, because those processes are not attached to any live browser DOM and are already complete before the DOM is updated.

```
protected override void OnAfterRender(bool firstRender)
```

Parameters

firstRender [bool](#)

Set to `true` if this is the first time [OnAfterRender\(bool\)](#) has been invoked on this component instance; otherwise `false`.

Remarks

The [OnAfterRender\(bool\)](#) and [OnAfterRenderAsync\(bool\)](#) lifecycle methods are useful for performing interop, or interacting with values received from `@ref`. Use the `firstRender` parameter to ensure that initialization work is only performed once.

OnInitialized()

Method invoked when the component is ready to start, having received its initial parameters from its parent in the render tree.

```
protected override void OnInitialized()
```

Class NavMenu

Namespace: [SewingManager.Webapp.Components.Layout](#)

Assembly: SewingManager.Webapp.dll

```
public class NavMenu : ComponentBase, IComponent, IHandleEvent, IHandleAfterRender
```

Inheritance

[object](#) ← [ComponentBase](#) ← NavMenu

Implements

[IComponent](#), [IHandleEvent](#), [IHandleAfterRender](#)

Inherited Members

[ComponentBase.OnInitialized\(\)](#), [ComponentBase.OnInitializedAsync\(\)](#),
[ComponentBase.OnParametersSet\(\)](#), [ComponentBase.OnParametersSetAsync\(\)](#),
[ComponentBase.StateHasChanged\(\)](#), [ComponentBase.ShouldRender\(\)](#),
[ComponentBase.OnAfterRender\(bool\)](#), [ComponentBase.OnAfterRenderAsync\(bool\)](#),
[ComponentBase.InvokeAsync\(Action\)](#), [ComponentBase.InvokeAsync\(Func<Task>\)](#),
[ComponentBase.DispatchExceptionAsync\(Exception\)](#),
[ComponentBase.SetParametersAsync\(ParameterView\)](#), [object.Equals\(object\)](#),
[object.Equals\(object, object\)](#), [object.GetHashCode\(\)](#), [object.GetType\(\)](#),
[object.MemberwiseClone\(\)](#), [object.ReferenceEquals\(object, object\)](#), [object.ToString\(\)](#)

Methods

BuildRenderTree(RenderTreeBuilder)

Renders the component to the supplied [RenderTreeBuilder](#).

```
protected override void BuildRenderTree(RenderTreeBuilder __builder)
```

Parameters

__builder [RenderTreeBuilder](#)

Namespace SewingManager.Webapp. Components.Pages

Classes

[Error](#)

[Home](#)

[Weather](#)

Class Error

Namespace: [SewingManager.Webapp.Components.Pages](#)

Assembly: SewingManager.Webapp.dll

```
[Route("/Error")]
public class Error : ComponentBase, IComponent, IHandleEvent, IHandleAfterRender
```

Inheritance

[object](#) ← [ComponentBase](#) ← Error

Implements

[IComponent](#), [IHandleEvent](#), [IHandleAfterRender](#)

Inherited Members

[ComponentBase.OnInitializedAsync\(\)](#), [ComponentBase.OnParametersSet\(\)](#),
[ComponentBase.OnParametersSetAsync\(\)](#), [ComponentBase.StateHasChanged\(\)](#),
[ComponentBase.ShouldRender\(\)](#), [ComponentBase.OnAfterRender\(bool\)](#),
[ComponentBase.OnAfterRenderAsync\(bool\)](#), [ComponentBase.InvokeAsync\(Action\)](#),
[ComponentBase.InvokeAsync\(Func<Task>\)](#), [ComponentBase.DispatchExceptionAsync\(Exception\)](#),
[ComponentBase.SetParametersAsync\(ParameterView\)](#), [object.Equals\(object\)](#),
[object.Equals\(object, object\)](#), [object.GetHashCode\(\)](#), [object.GetType\(\)](#),
[object.MemberwiseClone\(\)](#), [object.ReferenceEquals\(object, object\)](#), [object.ToString\(\)](#)

Methods

BuildRenderTree(RenderTreeBuilder)

Renders the component to the supplied [RenderTreeBuilder](#).

```
protected override void BuildRenderTree(RenderTreeBuilder __builder)
```

Parameters

[__builder](#) [RenderTreeBuilder](#)

OnInitialized()

Method invoked when the component is ready to start, having received its initial parameters from its parent in the render tree.

```
protected override void OnInitialized()
```

Class Home

Namespace: [SewingManager.Webapp.Components.Pages](#)

Assembly: SewingManager.Webapp.dll

```
[Route("/")]
public class Home : ComponentBase, IComponent, IHandleEvent, IHandleAfterRender
```

Inheritance

[object](#) ← [ComponentBase](#) ← Home

Implements

[IComponent](#), [IHandleEvent](#), [IHandleAfterRender](#)

Inherited Members

[ComponentBase.OnInitialized\(\)](#), [ComponentBase.OnInitializedAsync\(\)](#),
[ComponentBase.OnParametersSet\(\)](#), [ComponentBase.OnParametersSetAsync\(\)](#),
[ComponentBase.StateHasChanged\(\)](#), [ComponentBase.ShouldRender\(\)](#),
[ComponentBase.OnAfterRender\(bool\)](#), [ComponentBase.OnAfterRenderAsync\(bool\)](#),
[ComponentBase.InvokeAsync\(Action\)](#), [ComponentBase.InvokeAsync\(Func<Task>\)](#),
[ComponentBase.DispatchExceptionAsync\(Exception\)](#),
[ComponentBase.SetParametersAsync\(ParameterView\)](#), [object.Equals\(object\)](#),
[object.Equals\(object, object\)](#), [object.GetHashCode\(\)](#), [object.GetType\(\)](#),
[object.MemberwiseClone\(\)](#), [object.ReferenceEquals\(object, object\)](#), [object.ToString\(\)](#)

Methods

BuildRenderTree(RenderTreeBuilder)

Renders the component to the supplied [RenderTreeBuilder](#).

```
protected override void BuildRenderTree(RenderTreeBuilder __builder)
```

Parameters

[__builder](#) [RenderTreeBuilder](#)

Class Weather

Namespace: [SewingManager.Webapp.Components.Pages](#)

Assembly: SewingManager.Webapp.dll

```
[StreamRendering(true)]
[Route("/weather")]
public class Weather : ComponentBase, IComponent, IHandleEvent, IHandleAfterRender
```

Inheritance

[object](#) ← [ComponentBase](#) ← Weather

Implements

[IComponent](#), [IHandleEvent](#), [IHandleAfterRender](#)

Inherited Members

[ComponentBase.OnInitialized\(\)](#), [ComponentBase.OnParametersSet\(\)](#),
[ComponentBase.OnParametersSetAsync\(\)](#), [ComponentBase.StateHasChanged\(\)](#),
[ComponentBase.ShouldRender\(\)](#), [ComponentBase.OnAfterRender\(bool\)](#),
[ComponentBase.OnAfterRenderAsync\(bool\)](#), [ComponentBase.InvokeAsync\(Action\)](#),
[ComponentBase.InvokeAsync\(Func<Task>\)](#), [ComponentBase.DispatchExceptionAsync\(Exception\)](#),
[ComponentBase.SetParametersAsync\(ParameterView\)](#), [object.Equals\(object\)](#),
[object.Equals\(object, object\)](#), [object.GetHashCode\(\)](#), [object.GetType\(\)](#),
[object.MemberwiseClone\(\)](#), [object.ReferenceEquals\(object, object\)](#), [object.ToString\(\)](#)

Methods

BuildRenderTree(RenderTreeBuilder)

Renders the component to the supplied [RenderTreeBuilder](#).

```
protected override void BuildRenderTree(RenderTreeBuilder __builder)
```

Parameters

[__builder](#) [RenderTreeBuilder](#)

OnInitializedAsync()

Method invoked when the component is ready to start, having received its initial parameters from its parent in the render tree.

Override this method if you will perform an asynchronous operation and want the component to refresh when that operation is completed.

```
protected override Task OnInitializedAsync()
```

Returns

[Task](#)

A [Task](#) representing any asynchronous operation.

Namespace SewingManager.Webapp. Components.Pages.Materials

Classes

[AddMaterial](#)

[EditMaterial](#)

[MaterialList](#)

Class AddMaterial

Namespace: [SewingManager.Webapp.Components.Pages.Materials](#)

Assembly: SewingManager.Webapp.dll

```
[Route("/materials/add")]
public class AddMaterial : ComponentBase, IComponent, IHandleEvent, IHandleAfterRender
```

Inheritance

[object](#) ← [ComponentBase](#) ← AddMaterial

Implements

[IComponent](#), [IHandleEvent](#), [IHandleAfterRender](#)

Inherited Members

[ComponentBase.OnInitialized\(\)](#), [ComponentBase.OnParametersSet\(\)](#),
[ComponentBase.OnParametersSetAsync\(\)](#), [ComponentBase.StateHasChanged\(\)](#),
[ComponentBase.ShouldRender\(\)](#), [ComponentBase.OnAfterRender\(bool\)](#),
[ComponentBase.OnAfterRenderAsync\(bool\)](#), [ComponentBase.InvokeAsync\(Action\)](#),
[ComponentBase.InvokeAsync\(Func<Task>\)](#), [ComponentBase.DispatchExceptionAsync\(Exception\)](#),
[ComponentBase.SetParametersAsync\(ParameterView\)](#), [object.Equals\(object\)](#),
[object.Equals\(object, object\)](#), [object.GetHashCode\(\)](#), [object.GetType\(\)](#),
[object.MemberwiseClone\(\)](#), [object.ReferenceEquals\(object, object\)](#), [object.ToString\(\)](#)

Methods

BuildRenderTree(RenderTreeBuilder)

Renders the component to the supplied [RenderTreeBuilder](#).

```
protected override void BuildRenderTree(RenderTreeBuilder __builder)
```

Parameters

[__builder](#) [RenderTreeBuilder](#)

OnInitializedAsync()

Method invoked when the component is ready to start, having received its initial parameters from its parent in the render tree.

Override this method if you will perform an asynchronous operation and want the component to refresh when that operation is completed.

```
protected override Task OnInitializedAsync()
```

Returns

[Task](#)

A [Task](#) representing any asynchronous operation.

Class EditMaterial

Namespace: [SewingManager.Webapp.Components.Pages.Materials](#)

Assembly: SewingManager.Webapp.dll

```
[Route("/materials/edit/{matId:int}")]
public class EditMaterial : ComponentBase, IComponent, IHandleEvent, IHandleAfterRender
```

Inheritance

[object](#) ← [ComponentBase](#) ← EditMaterial

Implements

[IComponent](#), [IHandleEvent](#), [IHandleAfterRender](#)

Inherited Members

[ComponentBase.OnInitialized\(\)](#), [ComponentBase.OnInitializedAsync\(\)](#),
[ComponentBase.OnParametersSet\(\)](#), [ComponentBase.StateHasChanged\(\)](#),
[ComponentBase.ShouldRender\(\)](#), [ComponentBase.OnAfterRender\(bool\)](#),
[ComponentBase.OnAfterRenderAsync\(bool\)](#), [ComponentBase.InvokeAsync\(Action\)](#),
[ComponentBase.InvokeAsync\(Func<Task>\)](#), [ComponentBase.DispatchExceptionAsync\(Exception\)](#),
[ComponentBase.SetParametersAsync\(ParameterView\)](#), [object.Equals\(object\)](#),
[object.Equals\(object, object\)](#), [object.GetHashCode\(\)](#), [object.GetType\(\)](#),
[object.MemberwiseClone\(\)](#), [object.ReferenceEquals\(object, object\)](#), [object.ToString\(\)](#)

Properties

MatId

```
[Parameter]
public int MatId { get; set; }
```

Property Value

[int](#)

Methods

BuildRenderTree(RenderTreeBuilder)

Renders the component to the supplied [RenderTreeBuilder](#).

```
protected override void BuildRenderTree(RenderTreeBuilder __builder)
```

Parameters

__builder [RenderTreeBuilder](#)

OnParametersSetAsync()

Method invoked when the component has received parameters from its parent in the render tree, and the incoming values have been assigned to properties.

```
protected override Task OnParametersSetAsync()
```

Returns

[Task](#)

A [Task](#) representing any asynchronous operation.

Class MaterialList

Namespace: [SewingManager.Webapp.Components.Pages.Materials](#)

Assembly: SewingManager.Webapp.dll

```
[Route("/materials")]
public class MaterialList : ComponentBase, IComponent, IHandleEvent, IHandleAfterRender
```

Inheritance

[object](#) ← [ComponentBase](#) ← MaterialList

Implements

[IComponent](#), [IHandleEvent](#), [IHandleAfterRender](#)

Inherited Members

[ComponentBase.OnInitialized\(\)](#), [ComponentBase.OnInitializedAsync\(\)](#),
[ComponentBase.OnParametersSet\(\)](#), [ComponentBase.OnParametersSetAsync\(\)](#),
[ComponentBase.StateHasChanged\(\)](#), [ComponentBase.ShouldRender\(\)](#),
[ComponentBase.OnAfterRender\(bool\)](#), [ComponentBase.OnAfterRenderAsync\(bool\)](#),
[ComponentBase.InvokeAsync\(Action\)](#), [ComponentBase.InvokeAsync\(Func<Task>\)](#),
[ComponentBase.DispatchExceptionAsync\(Exception\)](#),
[ComponentBase.SetParametersAsync\(ParameterView\)](#), [object.Equals\(object\)](#),
[object.Equals\(object, object\)](#), [object.GetHashCode\(\)](#), [object.GetType\(\)](#),
[object.MemberwiseClone\(\)](#), [object.ReferenceEquals\(object, object\)](#), [object.ToString\(\)](#)

Methods

BuildRenderTree(RenderTreeBuilder)

Renders the component to the supplied [RenderTreeBuilder](#).

```
protected override void BuildRenderTree(RenderTreeBuilder __builder)
```

Parameters

[__builder](#) [RenderTreeBuilder](#)

Namespace SewingManager.Webapp.View Models

Classes

[MaterialViewModel](#)

Class MaterialViewModel

Namespace: [SewingManager.Webapp.ViewModels](#)

Assembly: SewingManager.Webapp.dll

```
public class MaterialViewModel
```

Inheritance

[object](#) ← MaterialViewModel

Inherited Members

[object.Equals\(object\)](#) , [object.Equals\(object, object\)](#) , [object.GetHashCode\(\)](#) , [object.GetType\(\)](#) ,
[object.MemberwiseClone\(\)](#) , [object.ReferenceEquals\(object, object\)](#) , [object.ToString\(\)](#)

Properties

CategoryId

[Required]
public string CategoryId { get; set; }

Property Value

[string](#)

Description

[StringLength(3000)]
public string Description { get; set; }

Property Value

[string](#)

ImageUrl

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

Property Value

[string ↗](#)

MaterialCode

```
[Required]  
public string MaterialCode { get; set; }
```

Property Value

[string ↗](#)

MaterialId

```
public int MaterialId { get; set; }
```

Property Value

[int ↗](#)

Name

```
[Required]  
[StringLength(150)]  
public string Name { get; set; }
```

Property Value

[string ↗](#)

Price

```
[Required(ErrorMessage = "Ціна обов'язкова")]
[Range(0, 1.7976931348623157E+308, ErrorMessage = "Ціна має бути більше або \">=\\ 0")]
public double Price { get; set; }
```

Property Value

[double](#)

Quantity

```
[Required(ErrorMessage = "Кількість обов'язкова")]
[Range(0, 1.7976931348623157E+308, ErrorMessage = "Кількість має бути більше або \">=\\ 0")]
public double Quantity { get; set; }
```

Property Value

[double](#)

Type

```
public MaterialType Type { get; set; }
```

Property Value

[MaterialType](#)

Namespace SewingManager.Webapp.View ModelsValidations

Classes

[Material_QuantityValidationByTypeAttribute](#)

Атрибут валідації, що перевіряє кількість матеріалу в залежності від його типу.

Class Material_QuantityValidationByTypeAttribute

Namespace: [SewingManager.Webapp.ViewModelsValidations](#)

Assembly: SewingManager.Webapp.dll

Атрибут валідації, що перевіряє кількість матеріалу в залежності від його типу.

```
public class Material_QuantityValidationByTypeAttribute : ValidationAttribute
```

Inheritance

[object](#) ← [Attribute](#) ← [ValidationAttribute](#) ← [Material_QuantityValidationByTypeAttribute](#)

Inherited Members

[ValidationAttribute.FormatErrorMessage\(string\)](#) ,
[ValidationAttribute.GetValidationResult\(object, ValidationContext\)](#) ,
[ValidationAttribute.IsValid\(object\)](#) , [ValidationAttribute.Validate\(object, ValidationContext\)](#) ,
[ValidationAttribute.Validate\(object, string\)](#) , [ValidationAttribute.ErrorMessage](#) ,
[ValidationAttribute.ErrorMessageResourceName](#) , [ValidationAttribute.ErrorMessageResourceType](#) ,
[ValidationAttribute.ErrorMessageString](#) , [ValidationAttribute.RequiresValidationContext](#) ,
[Attribute.Equals\(object\)](#) , [Attribute.GetCustomAttribute\(Assembly, Type\)](#) ,
[Attribute.GetCustomAttribute\(Assembly, Type, bool\)](#) ,
[Attribute.GetCustomAttribute\(MemberInfo, Type\)](#) ,
[Attribute.GetCustomAttribute\(MemberInfo, Type, bool\)](#) ,
[Attribute.GetCustomAttribute\(Module, Type\)](#) , [Attribute.GetCustomAttribute\(Module, Type, bool\)](#) ,
[Attribute.GetCustomAttribute\(ParameterInfo, Type\)](#) ,
[Attribute.GetCustomAttribute\(ParameterInfo, Type, bool\)](#) , [Attribute.GetCustomAttributes\(Assembly\)](#) ,
[Attribute.GetCustomAttributes\(Assembly, bool\)](#) , [Attribute.GetCustomAttributes\(Assembly, Type\)](#) ,
[Attribute.GetCustomAttributes\(Assembly, Type, bool\)](#) , [Attribute.GetCustomAttributes\(MemberInfo\)](#) ,
[Attribute.GetCustomAttributes\(MemberInfo, bool\)](#) ,
[Attribute.GetCustomAttributes\(MemberInfo, Type\)](#) ,
[Attribute.GetCustomAttributes\(MemberInfo, Type, bool\)](#) , [Attribute.GetCustomAttributes\(Module\)](#) ,
[Attribute.GetCustomAttributes\(Module, bool\)](#) , [Attribute.GetCustomAttributes\(Module, Type\)](#) ,
[Attribute.GetCustomAttributes\(Module, Type, bool\)](#) , [Attribute.GetCustomAttributes\(ParameterInfo\)](#) ,
[Attribute.GetCustomAttributes\(ParameterInfo, bool\)](#) ,
[Attribute.GetCustomAttributes\(ParameterInfo, Type\)](#) ,
[Attribute.GetCustomAttributes\(ParameterInfo, Type, bool\)](#) , [Attribute.GetHashCode\(\)](#) ,
[Attribute.IsDefaultAttribute\(\)](#) , [Attribute.IsDefined\(Assembly, Type\)](#) ,

[Attribute.IsDefined\(Assembly, Type, bool\)](#) , [Attribute.IsDefined\(MemberInfo, Type\)](#) ,
[Attribute.IsDefined\(MemberInfo, Type, bool\)](#) , [Attribute.IsDefined\(Module, Type\)](#) ,
[Attribute.IsDefined\(Module, Type, bool\)](#) , [Attribute.IsDefined\(ParameterInfo, Type\)](#) ,
[Attribute.IsDefined\(ParameterInfo, Type, bool\)](#) , [Attribute.Match\(object\)](#) , [Attribute.TypeId](#) ,
[object.Equals\(object, object\)](#) , [object.GetType\(\)](#) , [object.MemberwiseClone\(\)](#) ,
[object.ReferenceEquals\(object, object\)](#) , [object.ToString\(\)](#)

Methods

IsValid(object?, ValidationContext)

Перевіряє, чи кількість матеріалу є коректною відповідно до типу матеріалу.

```
protected override ValidationResult? IsValid(object? value, ValidationContext validationContext)
```

Parameters

value [object](#)

Значення, яке потрібно перевірити (кількість матеріалу).

validationContext [ValidationContext](#)

Контекст валідації.

Returns

[ValidationResult](#)

Результат валідації: якщо кількість коректна, повертається [ValidationResult.Success](#), в іншому випадку — повідомлення про помилку.

Namespace SewingManger.Plugins.InMemory

Classes

[MaterialRepository](#)

Репозиторій для управління матеріалами та їх категоріями в пам'яті.

Class MaterialRepository

Namespace: [SewingManger.Plugins.InMemory](#).

Assembly: SewingManger.Plugins.InMemory.dll

Репозиторій для управління матеріалами та їх категоріями в пам'яті.

```
public class MaterialRepository : IMaterialRepository
```

Inheritance

[object](#) ← MaterialRepository

Implements

[IMaterialRepository](#).

Inherited Members

[object.Equals\(object\)](#) , [object.Equals\(object, object\)](#) , [object.GetHashCode\(\)](#) , [object.GetType\(\)](#) ,
[object.MemberwiseClone\(\)](#) , [object.ReferenceEquals\(object, object\)](#) , [object.ToString\(\)](#)

Constructors

MaterialRepository()

Ініціалізує новий екземпляр репозиторію з попередньо визначеними категоріями та матеріалами.

```
public MaterialRepository()
```

Methods

AddMaterialAsync(Material)

Додає новий матеріал до репозиторію.

```
public Task AddMaterialAsync(Material material)
```

Parameters

material [Material](#)

Модель матеріалу для додавання.

Returns

[Task](#) ↗

Завдання, що завершується після додавання.

Examples

```
var material = new Material { Name = "Шовк", Type = MaterialType.Cut }; await  
repository.AddMaterialAsync(material);
```

DeleteMaterialByIdAsync(int)

Видаляє матеріал за його ідентифікатором.

```
public Task DeleteMaterialByIdAsync(int materialId)
```

Parameters

[materialId](#) [int](#) ↗

Ідентифікатор матеріалу, який потрібно видалити.

Returns

[Task](#) ↗

Завдання, яке виконується асинхронно.

GetMaterialByIdAsync(int)

Отримує матеріал за його ідентифікатором.

```
public Task<Material> GetMaterialByIdAsync(int materialId)
```

Parameters

`materialId int`

Ідентифікатор матеріалу, який потрібно переглянути.

Returns

`Task<Material>`

Завдання, яке повертає матеріал за вказаним ідентифікатором.

GetMaterialCategoriesByNameUseCase(string)

Отримує категорії матеріалів за їх іменем.

```
public Task<IEnumerable<MaterialCategory>> GetMaterialCategoriesByNameUseCase(string name)
```

Parameters

`name string`

Ім'я категорії для фільтрації результатів. Якщо не вказано, повертаються всі категорії.

Returns

`Task<IEnumerable<MaterialCategory>>`

Завдання, яке повертає колекцію категорій матеріалів, що відповідають вказаному імені.

GetMaterialsByNameAsync(string)

Отримує список матеріалів за назвою.

```
public Task<IEnumerable<Material>> GetMaterialsByNameAsync(string name)
```

Parameters

`name string`

Назва матеріалу для пошуку (чутливість до реєстру ігнорується).

Returns

[Task](#) <IEnumerable<[Material](#)>>

Список матеріалів, що відповідають критерію пошуку.

Examples

```
var materials = await repository.GetMaterialsByNameAsync("Бавовна"); foreach (var material in materials) { Console.WriteLine(material.Name); }
```

UpdateMaterialAsync(Material)

Оновлює дані існуючого матеріалу.

```
public Task UpdateMaterialAsync(Material material)
```

Parameters

[material](#) [Material](#)

Модель матеріалу з оновленими даними.

Returns

[Task](#)

Завдання, що завершується після оновлення.

Examples

```
var material = await repository.GetMaterialByIdAsync(1); material.Quantity = 2000; await repository.UpdateMaterialAsync(material);
```