Administration modul for Embedded Stock

Purpose:

To demonstrate fulfilment of the learning objectives:

- Explain the principles for using a MVC framework in a web server
- Design and implement a web site that include persistence of data in a database
- Basic knowledge regarding hosting of web applications including cloud based hosting

Technology requirements

The web app must use ASP.Net Core as MVC framework, Razor as view engine and must use Entity Framework Core to access a SQL database. It is recommended to use Azure as Cloud provider.

Description of the problem domain

The electronic shop at ASE uses a database to keep track of its stock of electronic components. The staffs in the electronic shop uses a PC-program to maintain this database, and students can browse the database with a web app (https://stockmanager.ase.au.dk/).

The staff wises to use other platforms (e.g. an iPad) to maintain the database, so the goal for this assignment is to develop the first iteration of the new administration module for Embedded Stock.

You can see in the model classes on the last page which data that must be stored in the database. Notice that a component type can belong to more than one category (which complicates the design).

In this first iteration there is no authentication and authorisation of users.

The next page lists a set of user stories that you must implement

User stories you must implement

Create component category
As a staff member, I can create a new component category

Delete component category
As a staff member, I can delete a component category

Update component category
As a staff member, I can update a component category

Create component type
As a staff member, I can create a new component type

Delete component type
As a staff member, I can delete a component type

Update component type
As a staff member, I can update a component type

Create component
As a staff member, I can create a new component

Delete component
As a staff member, I can delete a component

Update component
As a staff member, I can update a component

List component types for a category
As a staff member, I can see all component types for a chosen category

List components for a component type
As a staff member, I can see all components for a chosen component type

Model classes – may need some changes

```
public class Category
    {
        public Category()
        {
            ComponentTypes = new List<ComponentType>();
        }

        public int CategoryId { get; set; }
        public string Name { get; set; }
        public ICollection<ComponentType> ComponentTypes { get; protected set;
    }

public class ComponentType
    {
        public ComponentType()
        {
            Components = new List<Component>();
     .       Categories = new List<Category>();
        }

        public long ComponentTypeId { get; set; }
        public string ComponentName { get; set; }
        public string ComponentInfo { get; set; }
        public string Location { get; set; }
        public ComponentTypeStatus Status { get; set; }
        public string Datasheet { get; set; }
        public string ImageUrl { get; set; }
        public string Manufacturer { get; set; }
        public string WikiLink { get; set; }
        public string AdminComment { get; set; }
        public virtual ESImage Image { get; set; }
        public ICollection<Component> Components { get; protected set; }
        public ICollection<Category> Categories { get; protected set; }
    }


public class Component
    {
        public long ComponentId { get; set; }
        public long ComponentTypeId { get; set; }
        public int ComponentNumber { get; set; }
        public string SerialNo { get; set; }
        public ComponentStatus Status { get; set; }
        public string AdminComment { get; set; }
        public string UserComment { get; set; }
        public long? CurrentLoanInformationId { get; set; }
    }

public class ESImage
    {
        public long ESImageId { get; set; }
        [MaxLength(128)]
        public string ImageMimeType { get; set; }
        public byte[] Thumbnail { get; set; }
        public byte[] ImageData { get; set; }
    }
    public enum ComponentTypeStatus
    {
        Available,
        ReservedAdmin
    }
```

```
public enum ComponentStatus
    {
        Available,
        ReservedLoaner,
        ReservedAdmin,
        Loaned,
        Defect,
        Trashed,
        Lost,
        NeverReturned
    }
```