

UserStories Part 1 – model klasse

Du skal lave et nyt projekt med en model klasse til UserStories:

Lav et nyt projekt / solution

1. Nyt projekt/solution: **'ScrumBoard'**
Opret et nyt projekt: **'ScrumBoard'** af typen 'ASP.Net Core Web App'
- benyt .Net 5.0 og klik 'configure for https' tjekboksen fra
2. Afprøvning Kør projektet og verificer at templateen virker.
3. Opret følgende mapper i dit projekt:
 - Services
 - MockData

Lav et class library inde i solution

1. Højre klik på din solution (ikke dit projekt) og lav et nyt projekt (**'ScrumBoardLib'**), som er af typen 'Class Library' – benyt .Net 5.0 – igen.
2. Lav en mappe 'model' i libreriet.
3. Inden i mappen model, lav en klasse **'UserStory'** med følgende elementer (husk klassen skal være public):
 - a. Properties:
 - i. Id (int)
 - ii. Title (String)
 - iii. Description (String)
 - iv. StoryPoints (int)
 - v. BusinessValue (int)
 - b. To konstruktører:
 - i. En default konstruktør (dvs. med tom parameter liste)
 - ii. En Konstruktør med Title, Description som parametre (altså ikke id, StoryPoints eller BusinessValue)
 - iii. Id skal genereres automatisk (se neden under) samt StoryPoint og BusinessValue sættes begge til 0, Ved Default Konstruktøren sættes Title og Description til "dummy"
 - c. En ToString metode, til at returnere aktuelle property værdier i en tekststreng (string)
 - d. Der skal være et private static nextId felt til automatisk at optælle Id.
dvs. Id = nextId++; Det første objekt skal have Id værdien 1.

4. Der er nogle krav / betingelser til de forskellige properties:
 - a. Title, skal indeholde mindst 3 tegn
 - b. Description, skal indeholde mindst 10 tegn
 - c. StoryPoints må ikke være negative
 - d. BusinessValue skal være en værdi mellem 0-10, begge inklusive

Hvis ikke betingelserne er opfyldt, skal der kastes en passende exception.

Tilbage til ScrumBoardProjektet

1. Efter du har oversat (build) dit ScrumBoardLib, skal du tilføje en reference til dette lib i dit ScrumBoardProjekt.

2. Opret nogle MockData til første prototype

- a. I mappen MockData oprettes en ny klasse '**MockUserStories**'.
Klassen skal indeholde en statisk liste af UserStory objekter og initialiseres ala:

```
private static List<UserStory> userStories = new List<UserStory>() {  
    new UserStory("Create Story", "As P.O I want to create a new Story So ..."),  
    new UserStory("Edit Story", "As P.O I want to edit a Story So ..."),  
    new UserStory("Move Story", "As team member I want to move a Story So ..."),  
    new UserStory("Delete Story", "As team member I want to delete a Story So ...") };
```

- b. Lav en statisk metode GetMockUserStories(), der returnere listen.

3. Opret en UserStoryService

- a. I mappen Service oprettes en ny klasse '**UserStoryService**'.
Klassen skal indeholde en liste af UserStory objekter:
 - i. private List<UserStory> _userStories;
 - ii. En default konstruktør der initialiserer userStories ved et kald af GetMockUserStories() fra MockUserStories.
 - iii. En metode GetUserStories(), der returnerer userStories.

4. Konfigurer Projekt (Startup.cs)

- a. Åben Startup.cs for at registrere UserStoryService i ConfigureServices-metoden
 - i. services.AddSingleton<UserStoryService, UserStoryService>();
under linjen: services.AddRazorPages();

5. Lav en side(page) til at vise UserStories

- a. I mappen Pages lav en undermappe '**UserStories**'
- b. I undermappen UserStories lav en ny Razor Page '**index.cshtml**' (dette er default navn)
- c. I index.cshtml indsæt:
 - i. '<h1>User Stories</h1>'
- d. Tilføj et nyt link i _layout.cshtml filen til den nye side
- e. Kør og afprøv at link og page virker

6. Vise listen af UserStories på siden

a. I controller-klassen index.cshtml.cs

Vi skal nu have injected vores UserStoryService, så servicen kan benyttes til at hente listen af UserStories i MockData:

i. Tilføj et instance field til servicen:

```
private UserStoryService _userStoryService;
```

ii. Tilføj en property til Listen af UserStories:

```
public List<UserStory> UserStories { get; private set; }
```

iii. Inject servicen i konstruktøren:

```
public IndexModel(UserStoryService userStoryService) {  
    _userStoryService = userStoryService;  
}
```

iv. Opdater OnGet() metoden så den initialiserer listen UserStories med Data (mock) ved at kalde GetUserStories() på servicen:

```
UserStories = _userStoryService.GetUserStories();
```

b. Opdater Razor Page – 'UserStories/Index'

Vi skal have opdateret siden, så den viser alle vores UserStory.

Tilføj en foreach løkke der indsætter et oversigt af alle UserStory i en list:

```
<ul>  
    @foreach (var us in @Model.UserStories)  
    {  
        <li>  
            <div>(@us.Id) @us.Title</div>  
            <div>@us.Description</div>  
        </li>  
    }  
</ul>
```

7. Afprøvning Kør og afprøv at der er "hul igennem" og du kan få vist alle UserStories fra MockData