



C# Web – MVC

PRO/PRW - C# Web 1

**DE HOGESCHOOL
MET HET NETWERK**

Hogeschool PXL – Elfde-Liniestraat 24 – B-3500 Hasselt
www.pxl.be - www.pxl.be/facebook



Doel

- ASP.Net Core toepassing (Oefening 15)
 - MVC Web Application
 - TagHelper
 - MVCVoertuig
 - ViewComponent
 - MVCVoertuig

ASP.NET CORE

MVC Web Application

MVCVoertuig

(Blackboard – Lesmateriaal algemeen - Startbestanden)

- Partial View
- TagHelper
- ViewComponent

MVCVoertuig

1. Nuget Package Manager - packages



Microsoft.EntityFrameworkCore.SqlServer by Microsoft
Microsoft SQL Server database provider for Entity Framework Core.



Microsoft.EntityFrameworkCore.Tools by Microsoft
Entity Framework Core Tools for the NuGet Package Manager Console in Visual Studio.

MVCVoertuig

2. Data - VoertuigDbContext

```
using Microsoft.EntityFrameworkCore;
namespace MVCVoertuig.Data
{
    public class VoertuigDbContext : DbContext
    {
        public VoertuigDbContext(DbContextOptions<VoertuigDbContext> options)
            : base(options)
        { }
    }
}
```

MVCVoertuig

3. appsettings.json

```
"AllowedHosts": "*",  
"ConnectionStrings": {  
  "VoertuigConnection":  
    "Server=(localdb)\\MSSQLLocalDB;  
    Database=MVCVoertuig;  
    MultipleActiveResultSets=true"  
}
```

MVCVoertuig

4. Program.cs - AddDbContext

```
builder.Services.AddControllersWithViews();  
var connectionString =  
builder.Configuration.GetConnectionString("VoertuigConnection");  
builder.Services.AddDbContext<VoertuigDbContext>(options =>  
options.UseSqlServer(connectionString));
```

MVCVoertuig

5. Models/Data - Voertuig

```
public class Voertuig
{
    public int VoertuigId { get; set; }
    [Required]
    [StringLength(100)]
    public string Merk { get; set; }
    [Required]
    [StringLength(100)]
    public string MerkType { get; set; }
    [Required]
    public int Bouwjaar { get; set; }
    [Required]
    [Range(0, 500000)]
    public int Km { get; set; }
    [Column(TypeName = "decimal(8, 2)")]
    public decimal VerkoopPrijs { get; set; }
    public bool AanwezigInShowroom { get; set; }
}
```


MVCVoertuig

6. ModelValidation - Voertuig

```
namespace MVCVoertuig.ModelValidations
{
    public class BouwjaarValidation : Attribute, IModelValidator
    {
        public IEnumerable<ModelValidationResult> Validate(ModelValidationContext
context)
        {
            var cstMinValue=1900;
            int bouwjaar=0;
            var lst = new List<ModelValidationResult>();
            if (int.TryParse(context.Model.ToString(), out bouwjaar))
            {
                if (bouwjaar <cstMinValue || bouwjaar > DateTime.Now.Year)
                    lst.Add(new ModelValidationResult("", $"Bouwjaar moet liggen
tussen 1900 en {DateTime.Now.Year}"));
            }
            else
            {
                lst.Add(new ModelValidationResult("", "Geen geldig bouwjaar!"));
            }
            return lst;
        }
    }
}
```

MVCVoertuig

6. ModelValidation - Voertuig

```
using MVCVoertuig.ModelValidations;
using System.ComponentModel.DataAnnotations;
using System.ComponentModel.DataAnnotations.Schema;

namespace MVCVoertuig.Models.Data
{
    public class Voertuig
    {
        public int VoertuigId { get; set; }
        [Required]
        [StringLength(100)]
        public string? Merk { get; set; }
        [Required]
        [StringLength(100)]
        public string? MerkType { get; set; }
        [Required]
        [BouwjaarValidation]
        public int? Bouwjaar { get; set; }
    }
}
```

MVCVoertuig

7. VoertuigController toevoegen (empty MVC Controller)

Views/Voertuig – Index (List)

```
@model IEnumerable<MVCVoertuig.Models.Data.Voertuig>
@{
    ViewData["Title"] = "Index";
}
<p>
    <a asp-action="Create">Create New</a>
</p>
<table class="table">
```

MVCVoertuig

7. Views/Voertuig - Create

```
@model MVCVoertuig.Models.Data.Voertuig
@{
    ViewData["Title"] = "Create";
}
<h4>Voertuig</h4>
<hr />
<div class="row">
    <div class="col-md-4">
        <form asp-action="Create">
```

MVCVoertuig

8. Controllers – VoertuigController – Dependency Injection

```
namespace MVCVoertuig.Controllers
{
    public class VoertuigController : Controller
    {
        private readonly VoertuigDbContext _context;

        public VoertuigController(VoertuigDbContext context)
        {
            _context = context;
        }
    }
}
```

MVCVoertuig

8. Controllers – VoertuigController – Index (List) & Create

```
public async Task<IActionResult> Index()
{
    return View(await _context.Voertuigen.ToListAsync());
}
public IActionResult Create()
{
    return View();
}
[HttpPost]
[ValidateAntiForgeryToken]
public async Task<IActionResult> Create(Voertuig voertuig)
{
    if (ModelState.IsValid)
    {
        _context.Voertuigen.Add(voertuig);
        await _context.SaveChangesAsync();
        return RedirectToAction(nameof(Index));
    }
    return View(voertuig);
}
```

MVCVoertuig

9. DbContext – Nieuwe tabel - Voertuigen

```
public class VoertuigDbContext : DbContext
{
    public VoertuigDbContext(DbContextOptions<VoertuigDbContext> options) : base(options)
    { }
    public DbSet<Voertuig> Voertuigen { get; set; }
}
```

MVCVoertuig

10. SeedData – Default Data

```
namespace MVCVoertuig.Data.DefaultData
{
    public static class SeedData
    {
        public static void EnsurePopulated(WebApplication app)
        {
            using (var scope = app.Services.CreateScope())
            {
                var context =
scope.ServiceProvider.GetRequiredService<VoertuigDbContext>();
                if (!context.Voertuigen.Any())
                {
                    foreach (var voertuig in GetVoertuigen())
                    {
                        context.Voertuigen.Add(voertuig);
                    }
                    context.SaveChanges();
                }
            }
        }
    }
}
```


MVCVoertuig

10. SeedData – Default Data – Program.cs

```
app.MapControllerRoute(  
    name: "default",  
    pattern: "{controller=Home}/{action=Index}/{id?}");  
  
SeedData.EnsurePopulated(app);  
  
app.Run();
```

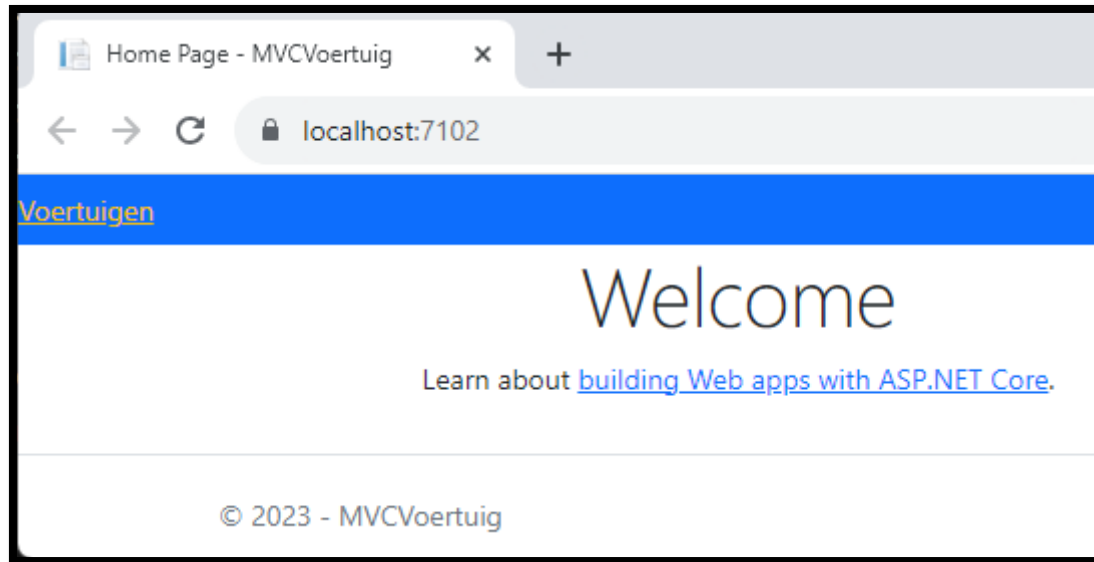
MVCVoertuig

11. Styling – Views/Shared/_Layout.cshtml

```
<header>
  <nav class="navbar navbar-expand-sm bg-primary navbar-light">
    <a class="link-warning" asp-controller="Voertuig" asp-action="Index">Voertuigen</a>
  </nav>
</header>
<div class="container">
  <main role="main" class="pb-3">
    @RenderBody()
  </main>
</div>
<footer class="border-top footer text-muted">
  <div class="container">
    &copy; PXL – MVCVoertuig
  </div>
</footer>
```

MVCVoertuig

11b Database Migratie & update-database



ASP.NET CORE

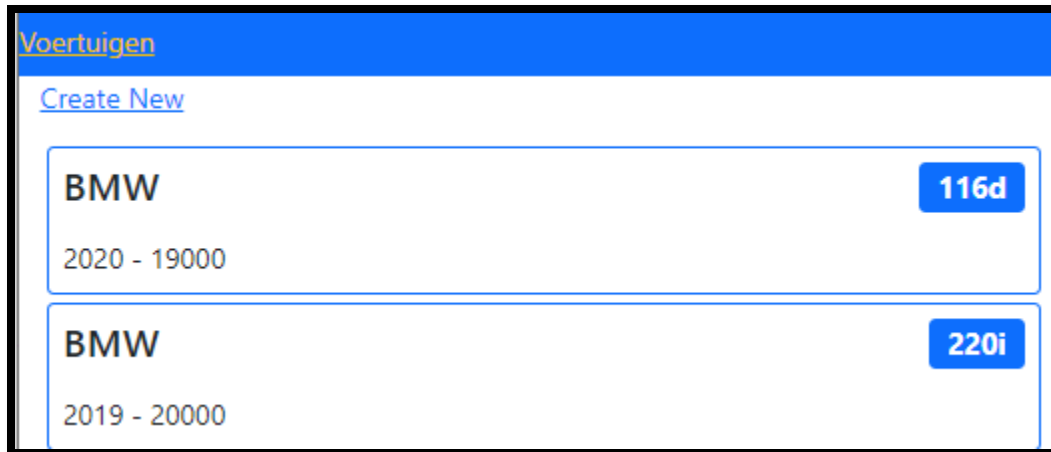
MVC Web Application

MVCVoertuig

- Partial View

MVCVoertuig

12. Partial View - VoertuigCard



MVCVoertuig

12. Partial View – VoertuigCard – Views/Shared/VoertuigCard.cshtml

```
@model MVCVoertuig.Models.Voertuig

<div class="card card-outline-primary border-primary m-1 p-1">
  <div class="bg-faded p-1">
    <h4>
      @Model.Merk
      <span class="badge bg-primary" style="float:right">
        @Model.MerkType
      </span>
    </h4>
  </div>
  <div class="card-text p-1">@Model.Bouwjaar – @Model.Km</div>
</div>
```

MVCVoertuig

12. Partial View – VoertuigCard – Views/Voertuig/Index.cshtml

```
@model IEnumerable<MVCVoertuig.Models.Voertuig>
@{
    ViewData["Title"] = "Index";
}
<p>
    <a asp-action="Create">Create New</a>
</p>
@foreach (var voertuig in Model)
{
    <partial name="VoertuigCard" model="voertuig" />
}
```

ASP.NET CORE

MVC Web Application

MVCVoertuig

- TagHelper

MVCVoertuig

13. TagHelper – Views/_ViewImports.cshtml

```
@using MVCVoertuig  
@using MVCVoertuig.Models  
@addTagHelper *, Microsoft.AspNetCore.Mvc.TagHelpers  
@addTagHelper *, MVCVoertuig
```

MVCVoertuig

13. TagHelper – HyperlinkButtonTagHelper

```
namespace MVCVoertuig.TagHelpers
{
    [HtmlTargetElement("a", Attributes = "button-color")]
    public class HyperlinkButtonTagHelper : TagHelper
    {
        public string ButtonColor { get; set; }
        public override void Process(TagHelperContext context,
                                     TagHelperOutput output)
        {
            output.Attributes.SetAttribute("class", $"btn btn-{ButtonColor}");
        }
    }
}
```

MVCVoertuig

13. TagHelper – CarInfoTagHelper

```
namespace MVCVoertuig.TagHelpers
{
    [HtmlTargetElement("car-info")]
    public class CarInfoTagHelper : TagHelper
    {
        [HtmlAttributeName("merk")]
        public string Merk { get; set; }
        VoertuigDbContext _context;
        public CarInfoTagHelper(VoertuigDbContext context)
        {
            _context = context;
        }
    }
}
```

MVCVoertuig

13. TagHelper – CarInfoTagHelper

```
private int GetCarCount()
{
    int count = 0;
    if (string.IsNullOrEmpty(Merk))
    {
        count = _context.Voertuigen.Count();
    }
    else
    {
        count = _context.Voertuigen.Where(x => x.Merk == Merk).Count();
    }
    return count;
}
```

MVCVoertuig

13. TagHelper – CarInfoTagHelper

```
public override void Process(TagHelperContext context,
                             TagHelperOutput output)
{
    output.Content.SetHtmlContent(GetHtmlText());
}
private string GetHtmlText()
{
    StringBuilder html = new StringBuilder();
    html.Append("<button type='button' class='btn border-primary m-2'>");
    html.Append("Aantal voertuigen ");
    html.Append("<span class='badge bg-primary'>");
    html.Append($"{GetCarCount()}");
    html.Append("</span>");
    html.Append("</button>");
    return html.ToString();
}
```

MVCVoertuig

13. TagHelper – CarInfoTagHelper – Views/Home/Index



Views/Home/Index.cshtml

```
@{  
    ViewData["Title"] = "Home Page";  
}  
<car-info></car-info>
```

ASP.NET CORE

MVC Web Application

MVCVoertuig

- ViewComponent

MVCVoertuig

14. ViewComponent – HeaderMenuViewComponent

- Create folder Components
 - Add class HeaderMenuViewComponent
- Create folder Views/Shared/Components/HeaderMenu
 - Add empty razor view
 - Default.cshtml

MVCVoertuig

14. ViewComponent – HeaderMenuViewComponent Views/Shared/Components/HeaderMenu/Default.cshtml

```
@model IEnumerable<string>

<ul class="navbar-nav">
  <li class="nav-item">
    <a button-color="info" asp-controller="Home" asp-action="Index">Home</a>
  </li>
  <li class="nav-item">
    <a button-color="info" asp-controller="Voertuig" asp-action="Index">Voertuigen</a>
  </li>
  @foreach (string merk in Model)
  {
    <li class="nav-item">
      <a button-color="info" asp-controller="Voertuig" asp-action="Merk" asp-route-merk="@merk">@merk</a>
    </li>
  }
</ul>
```

MVCVoertuig

14. ViewComponent – HeaderMenuViewComponent

- VoertuigController

```
public IActionResult Merk(string merk)
{
    ViewBag.Merk = merk;
    return View(_context.Voertuigen.Where(x => x.Merk == merk));
}
```

- Add razor view in Views/Voertuig

- List template
- Name: Merk.cshtml

```
@model IEnumerable<MVCVoertuig.Models.Data.Voertuig>
@{
    ViewData["Title"] = "View";
    var merk = ViewBag.Merk;
}
<br />
<car-info merk="@merk"></car-info>
<br />
```

MVCVoertuig

14. ViewComponent – HeaderMenuViewComponent Components/HeaderMenuViewComponent.cs

```
public class HeaderMenuViewComponent : ViewComponent
{
    VoertuigDbContext _context;
    public HeaderMenuViewComponent(VoertuigDbContext context)
    {
        _context = context;
    }
    public IActionResult Invoke()
    {
        return View(
            _context.Voertuigen
                .Select(x => x.Merk)
                .Distinct()
                .OrderBy(x => x));
    }
}
```

MVCVoertuig

14. ViewComponent – HeaderMenuViewComponent

- Views/Shared/_Layout.cshtml

```
<header>  
  <nav class="navbar navbar-expand-sm bg-primary navbar-light">  
    <vc:header-menu></vc:header-menu>  
  </nav>  
</header>
```