



I'm not a big fan of .Net and Microsoft! Anyway, Let's create a simple blog CMS using C# and ASP.NET Core. This will be a basic implementation to get you started.

Step 1: Set Up Your Project

1. Create a new ASP.NET Core Web Application:

```
dotnet new mvc -n SimpleBlogCMS
cd SimpleBlogCMS
```

2. Install Entity Framework Core:

```
dotnet add package Microsoft.EntityFrameworkCore.SqlServer
dotnet add package Microsoft.EntityFrameworkCore.Tools
```

Step 2: Define Your Models

Create a Models folder and add a Post class:

```
namespace SimpleBlogCMS.Models
{
    public class Post
    {
        public int Id { get; set; }
        public string Title { get; set; }
        public string Content { get; set; }
        public DateTime CreatedAt { get; set; }
    }
}
```

Step 3: Set Up Your Database Context

Create a Data folder and add a BlogContext class:

```
options.UseSqlServer(Configuration.GetConnectionString("DefaultConnection"));
services.AddControllersWithViews(); } public void Configure(IApplicationBuilder app,
IWebHostEnvironment env) { if (env.IsDevelopment()) { app.UseDeveloperExceptionPage();
} else { app.UseExceptionHandler("/Home/Error"); app.UseHsts(); }
```



```
app.UseHttpsRedirection(); app.UseStaticFiles(); app.UseRouting(); app.UseAuthorization();
app.UseEndpoints(endpoints => { endpoints.MapControllerRoute( name: "default", pattern:
"{controller=Home}/{action=Index}/{id?}"); }); } ">
```

```
public void ConfigureServices(IServiceCollection services)
{
    services.AddDbContext<BlogContext>(options =>
options.UseSqlServer(Configuration.GetConnectionString("DefaultConnect
ion")));
    services.AddControllersWithViews();
}
```

```
public void Configure(IApplicationBuilder app, IWebHostEnvironment
env)
{
    if (env.IsDevelopment())
    {
        app.UseDeveloperExceptionPage();
    }
    else
    {
        app.UseExceptionHandler("/Home/Error");
        app.UseHsts();
    }
    app.UseHttpsRedirection();
    app.UseStaticFiles();
    app.UseRouting();
    app.UseAuthorization();
    app.UseEndpoints(endpoints =>
    {
        endpoints.MapControllerRoute(
            name: "default",
            pattern: "{controller=Home}/{action=Index}/{id?}");
    });
}
```

Add your connection string to appsettings.json:

```
"ConnectionStrings": {
```



```
"DefaultConnection":  
"Server=(localdb)\\mssqllocaldb;Database=SimpleBlogCMS;Trusted_Connection=True;MultipleActiveResultSets=true"  
}
```

Step 5: Create Your Controllers

Create a Controllers folder and add a PostsController:

```
@model IEnumerable<SimpleBlogCMS.Models.Post>
```

```
<h1>Blog Posts</h1>  
<p>  
    <a asp-action="Create">Create New</a>  
</p>  
<table class="table">  
    <thead>  
        <tr>  
            <th>Title</th>  
            <th>Content</th>  
            <th>Created At</th>  
        </tr>  
    </thead>  
    <tbody>  
        @foreach (var item in Model)  
        {  
            <tr>  
                <td>@item.Title</td>  
                <td>@item.Content</td>  
                <td>@item.CreatedAt</td>  
            </tr>  
        }  
    </tbody>  
</table>
```

Create.cshtml: