

# Adding Login Capabilities

---



**Gill Cleeren**

@gillcleeren [www.snowball.be](http://www.snowball.be)



# Overview



**Exploring ASP.NET Core Identity**

**Adding authentication to the site**

**Enabling authorization**



# Exploring ASP.NET Core Identity

---





## ASP.NET Identity

- Membership system
- Authentication and authorization
- Supports external providers
- SQL Server support built-in
- Since ASP.NET Core 2.1: scaffolding support



## Adding ASP.NET Identity

- IdentityDbContext
- Configuration changes
  - Password length
  - Cookies
  - User options
- Scaffolding

# Configuration Options

```
services.Configure<IdentityOptions>(options =>
{
    options.Password.RequireDigit = true;
    options.Password.RequiredLength = 8;
    options.Password.RequireNonAlphanumeric = true;
    options.User.RequireUniqueEmail = true;
});
```



# Demo



## Adding ASP.NET Identity



# Adding Authentication to the Site

---





# Differences Between ASP.NET Core Versions

## **ASP.NET Core 2.0**

Manual approach based on  
AccountController

## **ASP.NET Core 2.1 and higher**

Support for Razor Class  
Library and scaffolding



# Side Note: Razor Class Library



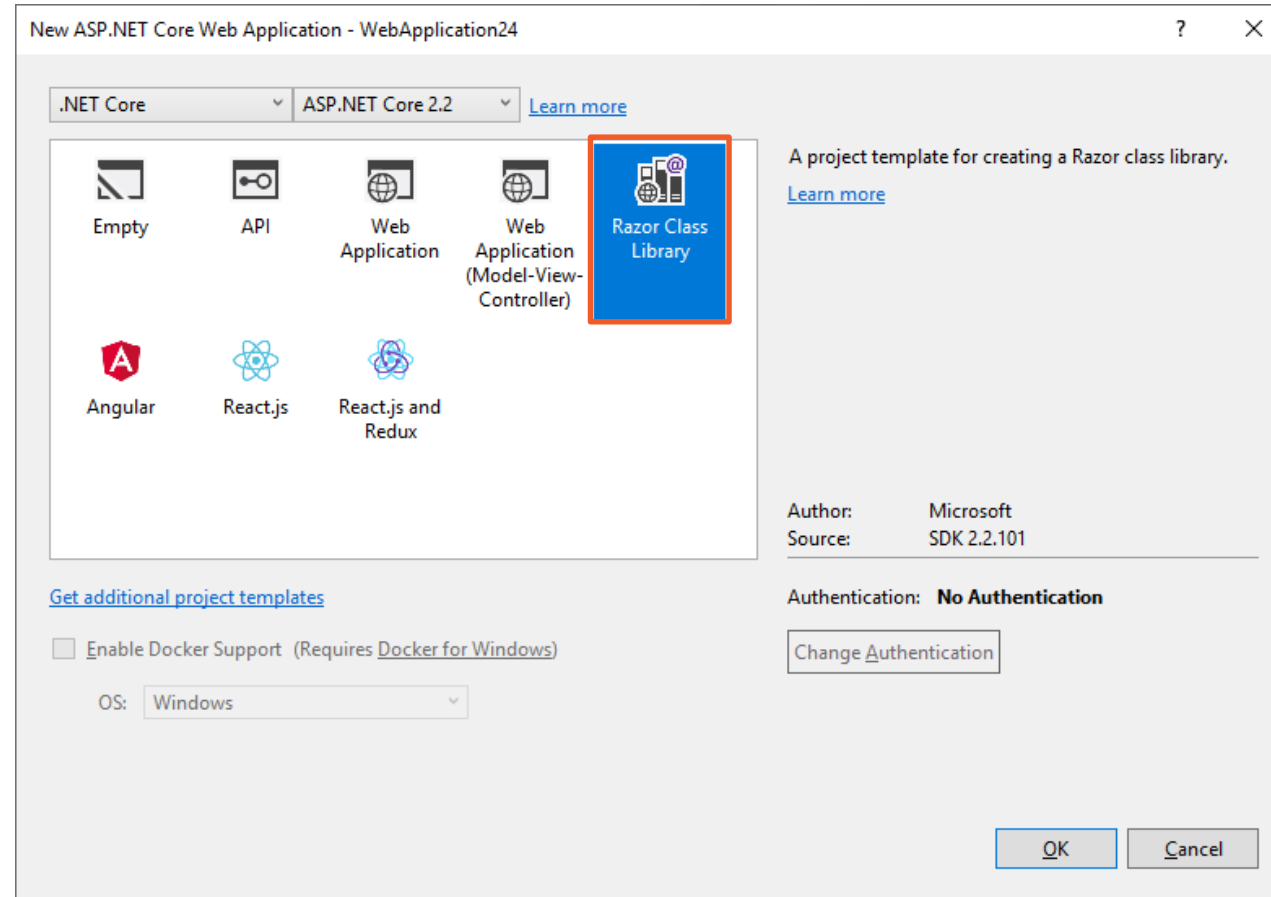
**ASP.NET Core 2.1 feature**

**Reuse functionality**

- Views, pages, controllers, view components...

**Can be overridden**

# Razor Class Library (RCL)



# Allowing Users to Log In

**Use built-in RCL**

**Scaffold items**

**Layout changes**



# Important Classes in Identity

**UserManager<IdentityUser>**

**SignInManager<IdentityUser>**



# Demo



**Scaffolding Login and Register  
functionality**

**Layout changes**



# Enabling Authorization

---



```
[Authorize]  
public class FeedbackController : Controller  
{  
}
```

## Authorizing Users





```
public class FeedbackController : Controller
{
    [Authorize]
    public IActionResult Send()
    { }
}
```

Action Level



```
[Authorize(Roles = "Administrator")]  
public class FeedbackController : Controller  
{  
}
```

Combining with Roles



# Demo



## Authorizing users to the Feedback view



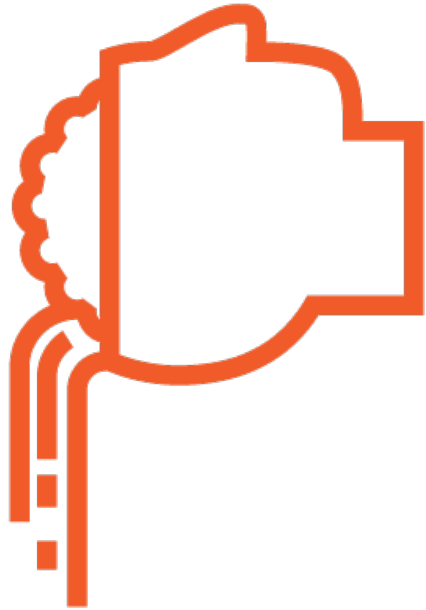
# Summary



## **ASP.NET Identity works for**

- authentication
- authorization

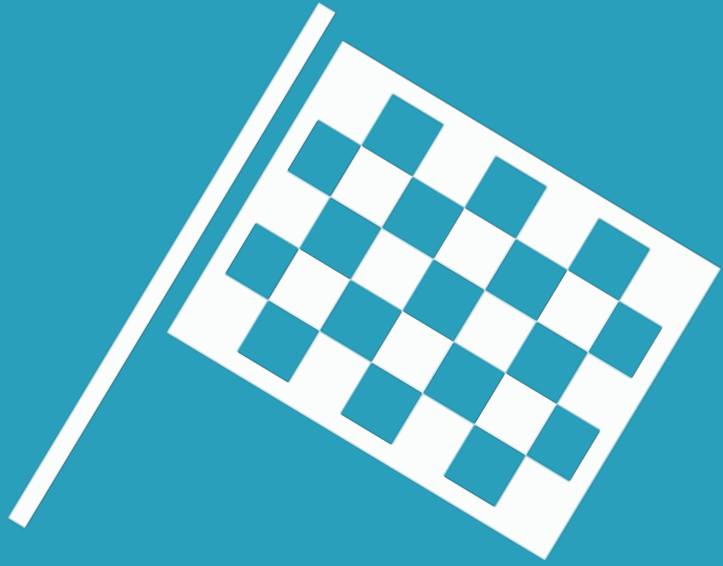




### Related, deep-dive course:

- Building an Enterprise Application with ASP.NET Core MVC





Congratulations  
on finishing this course!

