

ASP

Authorisation and Authentication

With ASP.NET Identity

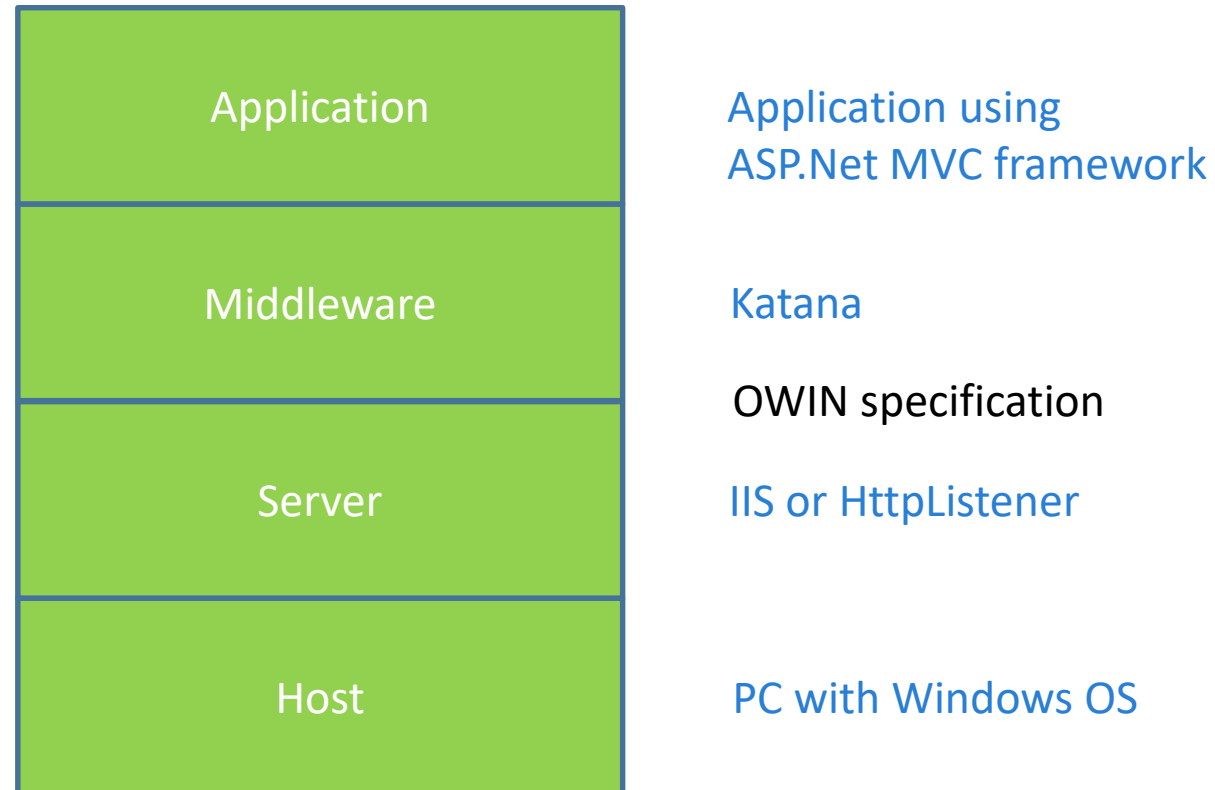
Definitions

- **Authentication** is the process of verifying that "you are who you say you are"
- **Authorization** is the process of verifying that "you are permitted to do what you are trying to do"

ASP.NET Identity

- Is Microsofts implementation of a authorisation and authentication framework
- Can be used with all of the ASP.NET frameworks, such as ASP.NET MVC, Web Forms, Web Pages, Web API, and SignalR
- Ease of plugging in profile data about the user
- Persistence control
- Unit testability
- Role provider
- Claims Based
- Social Login Providers (OAuth)
- Azure Active Directory
- OWIN Integration
- NuGet package

ASP.Net MVC 5 Architecture



Crate a New MVC project

New ASP.NET Project - ASP_Identity

Select a template:

ASP.NET 4.6 Templates

Empty Web Forms **MVC** Web API Single Page Application

Azure API App (Preview) Azure Mobile App (Preview) Azure Mobile Service

ASP.NET 5 Preview Templates

Empty Web API Web Application

A project template for creating ASP.NET MVC applications. ASP.NET MVC allows you to build applications using the Model-View-Controller architecture. ASP.NET MVC includes many features that enable fast, test-driven development for creating applications that use the latest standards.

[Learn more](#)

Change Authentication

Authentication: **Individual User Accounts**

Microsoft Azure

☐ Host in the cloud

Web App

Add folders and core references for:

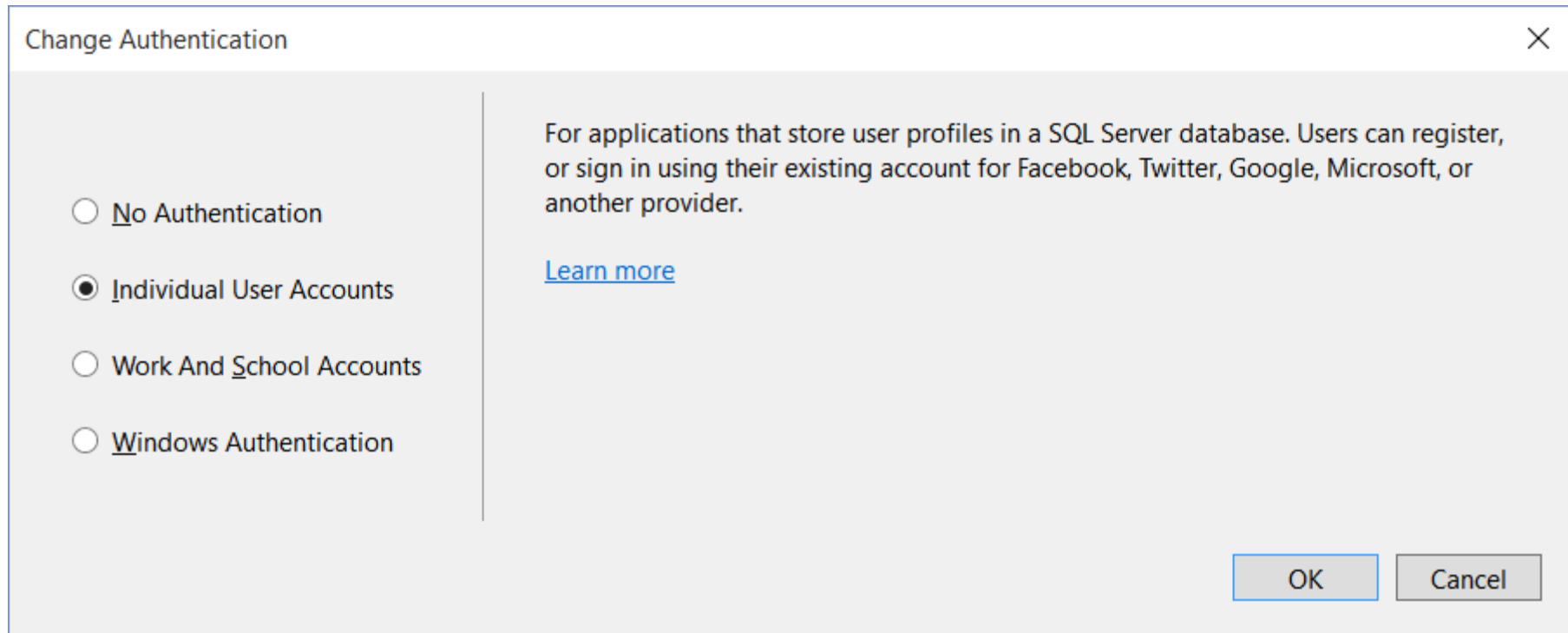
☐ Web Forms ☒ MVC ☐ Web API

☐ Add unit tests

Test project name: ASP_Identity.Tests

OK Cancel

Select Authentication Scheme



Change Authentication

☐ No Authentication

☒ Individual User Accounts

☐ Work And School Accounts



☐ Windows Authentication

For applications that store user profiles in a SQL Server database. Users can register, or sign in using their existing account for Facebook, Twitter, Google, Microsoft, or another provider.

[Learn more](#)

OK Cancel

How to Use?



```
[Authorize]
public class AccountController : Controller
{
    [AllowAnonymous]
    public ActionResult Login(string returnUrl)
    {
        ViewBag.ReturnUrl = returnUrl;
        return View();
    }

    //
    // POST: /Account/Login
    [HttpPost]
    [AllowAnonymous]
    [ValidateAntiForgeryToken]
    public async Task<ActionResult> Login(LoginViewModel
model, string returnUrl)
    {
        ApplicationUser user = null;
        if (!ModelState.IsValid)
```

How to require a Role?

Define your roles

```
public enum Role
{
    Student,
    Staff,
    Admin
}
```

```
[Authorize(Roles = "Admin")]
public class ComponentTypeController : Controller
{
```

When initializing IdentityDbContext

```
//Create Roles if they do not exist
foreach (var role in roles)
{
    if (!RoleManager.RoleExists(role))
    {
        RoleManager.Create(new IdentityRole(role));
    }
}
```


Oauth Specification

- Is an open standard for authorization
- It specifies a process for resource owners to authorize third-party access to their server resources without sharing their credentials
- Focuses on client developer simplicity while providing specific authorization flows for web applications, desktop applications, mobile phones, and living room devices
- Is being developed within the IETF OAuth WG
- **OAuth is commonly used as a way for Internet users to log into third party websites using their Microsoft, Google, Facebook or Twitter accounts without exposing their password**
- The OAuth 2.0 framework was published as RFC 6749 in October 2012

References & Links

- Introduction to ASP.NET Identity
<http://www.asp.net/identity/overview/getting-started/introduction-to-aspnet-identity>
- **Identity by K. Scott Allen** (must see)
<http://pluralsight.com/training/Player?author=scott-allen&name=aspdotnet-mvc5-fundamentals-m6-ef6&mode=live&clip=0&course=aspdotnet-mvc5-fundamentals>
- **OAuth**
<http://oauth.net/2/>