

ASP.NET Identity

What is ASP.NET Identity?

ASP.NET Identity is a framework for security and user management.

- Developed by Microsoft
- ASP.NET Identity replaces the previous membership and security solutions
- ASP.NET Identity 2.0 was released in the spring 2014
- Works with ASP.NET MVC, Web API, Web Pages, SignalR and Web Forms
- Compatible from .NET 4.5

What is ASP.NET Identity?

ASP.NET Identity was developed to create a robust security solution which can cope with modern demands.

It also relieves the developers from creating a new user management system for every new application.

Giving us a framework for creating secure and modern security solutions.

Why ASP.NET Identity?

ASP.NET Identity is a security library from Microsoft, part of the .NET stack.

ASP.NET Identity is currently the preferred way to handle security in ASP.NET applications

It comes with modern safety features like:

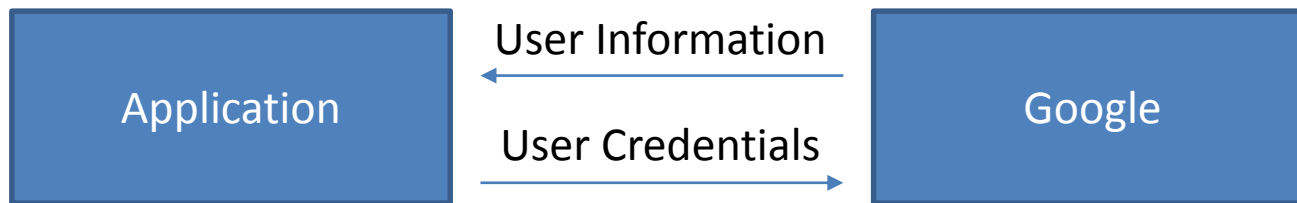
- Account confirmation
 - Both email and/or SMS confirmation
- Improved password handling
- Delegated authorization
- Two-Factor Authentication

Why ASP.NET Identity?

ASP.NET Identity supports delegated authorization, using for example OAuth which grant you:

- Easy use of social login providers thanks to OAuth
- Delegation of user authentication process

Example of such providers : Google, Facebook or Twitter



Claims-based authentication

Gives the developer the option to Create:

- More detailed identities
- Better control over what users can do

You can look at claims as a users properties, for example: name, age or IsAdmin.

ASP.NET Identity works great with following projects in .NET:

- MVC
- Web Forms
- Web API
- Single Page Application
- SignalR

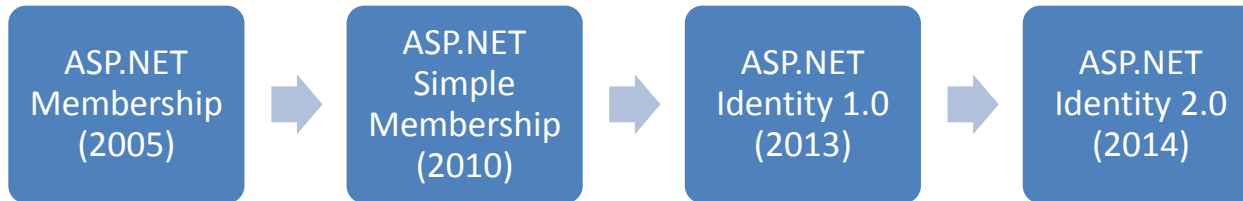
When should I use ASP.NET Identity?

- When your application will handle external users
- When you don't want to handle user authentication yourself
- When you feel the need for flexibility which claims-based authorization gives you
- When you want your application to follow the best practices of modern security

Before ASP.NET Identity

What was before ASP.NET Identity?

Before ASP.NET Identity there has been several attempts of making good libraries for handling user security



What was before ASP.NET Identity?

General flaws with the previous security libraries:

- Difficulties adding additional information to the user-identity
- No longer up to date with todays security standards
 - Not using modern password storage techniques
 - Does not support two-factor authentication
 - Does not support claims-based authentication

Think twice about using membership and simplemembership

<http://brockallen.com/2012/09/02/think-twice-about-using-membershipprovider-and-simplemembership/>

What was before ASP.NET Identity?

ASP.NET Membership

- Forms based authentication.
- Designed with only SQL servers in mind.
- No Owin integration.
- No social login providers.
- Difficult to customize user information
- Poor architecture

What was before ASP.NET Identity?

ASP.NET Simple Membership

- Built upon ASP.NET Membership
- Designed for ASP.NET MVC
- Supports OAuth/OpenId
- Tied to Microsoft databases (SQL)

What was before ASP.NET Identity?

ASP.NET Identity 1.0

- Released in 2013
- Simple & flexible way to store and handle user data
- Supports Roles & Claims
- Social Login Providers using OAuth
- Supports external Identity providers
- Supports customized databases
- Requires .NET 4.5

What was before ASP.NET Identity?

ASP.NET Identity 2.0

- Released in 2014
- Supports External Security Token Provider
- Two-Factor authentication
- Account confirmation (Email/SMS)
- Account lockout (Configure lockout time and number of attempts)
- Password reset (Email/SMS)

Resources

- ASP.NET official site
<http://www.asp.net/identity>
- ASP.NET security forum
<http://forums.asp.net/25.aspx/1?Security>
- Brock Allen (Blog)
<http://brockallen.com/category/asp-net-security/>
- Dominick Baier (Blog) Mostly focused on Web-API
<http://leastprivilege.com/category/net-security/>

- Pro ASP.NET MVC 5 Platform

