Dhruvil Dobariya

dhruvildobariya21@gmail.com

Abstract

[Draw your reader in with an engaging abstract. It is typically a short summary of the document.   
When you’re ready to add your content, just click here and start typing.]

ASP.Net Identity

Documentation



Introduction To Identity

# What is Identity?

* It is razor library which is help use to manage user role, clime, authentication, authorization and other user related task like login log out.

# Different between Identity and Identity Server or Framework

* Identity : It is user to manage user related task.
* Identity Server or Framework : It is used to manage protocol support for open id connect.

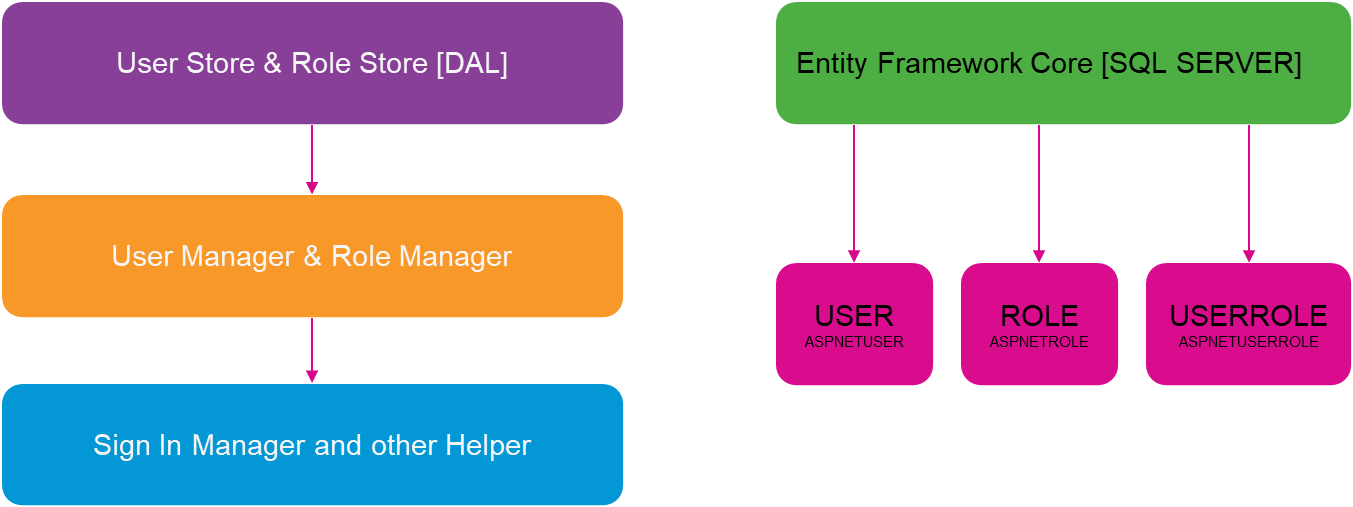
# Different between Authentication and Authorization

* Authentication : Who you are?
* Authorization : What access do you have?

# Authentication

* Cookie Based Authentication :
  + It is a default method.
  + It is used to handling user authentication for a long time.
  + It is stateful.
* Token Based Authentication
  + Token-based authentication has gained prevalence over the last few years due to the rise of single page applications, web APIs, and the Internet of Things (IoT).
  + The server does not keep a record of which users are logged in or which JWTs have been issued, hence stateless.
* Cookie Based Authentication Flow
  + User enters their login credentials.
  + Server verifies the credentials are correct and creates a session which is then stored in a database.
  + A cookie with the session ID is placed in the user's browser.
  + On subsequent requests, the session ID is verified against the database and if valid the request processed.
  + Once a user logs out of the app, the session is destroyed both client-side and server-side.
* Token Based Authentication Flow
  + User enters their login credentials.
  + Server verifies the credentials are correct and returns a signed token.
  + This token is stored client-side
  + Subsequent requests to the server includes this token
  + The server decodes the JWT and if the token is valid processes the request.
  + Once a user logs out, the token is destroyed client-side.

# Structure and Architecture of ASP.NET Core Identity



# Different between Roles and Claims

* Roles : Who the user is in aa given context?
* Claims : A piece of information about user.