**Section03 Token based authentication in web API**

**Notes: -**

**1-Token based authentication : which contains enough data to identify user , its loosely coupling and the token is generated and perform authentication by the server**

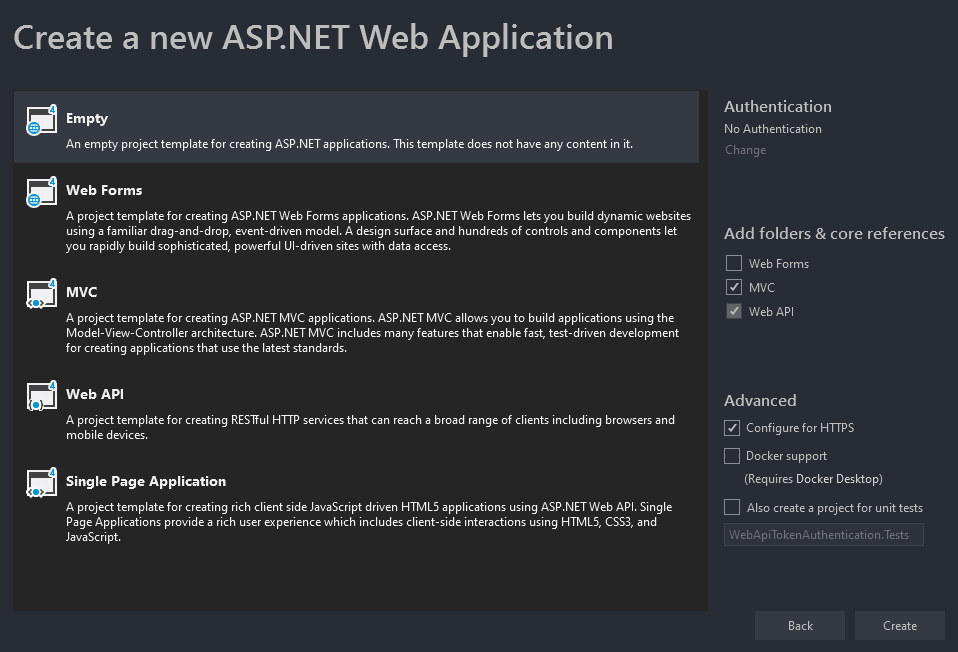
**2-Token based authentication is mobile friendly**

**3-Token based authentication send request to authentication server endpoint with appreciate credential and if the info is correct it will send to the client the generated token that contains all the enough information**

**4-owin handle generate Token and implement Authenticate which detect that the passed token is valid or not and implement Authorize to make sure that the user role is matched**

**Steps:-**

**1-on VS 2019 > create empty project with specify MVC and Web API as below**



**2-we install the following nuget packages:-**

**Microsoft.Owin.Host.SystemWeb //enable owin to be accessible from the Asp.Net**

**Microsoft.Owin.Security.OAuth //contains the OAuth V2.0 middleware authentication**

**Microsoft.Owin.Cors //to allow CROS origin to access to web app owin**

**3-on the project > right click > add new item and select owin startup**

**using Microsoft.Owin;**

**using Microsoft.Owin.Cors;**

**using Microsoft.Owin.Security.OAuth;**

**using Owin;**

**using System;**

**using System.Web.Http;**

**using WebApiTokenAuthentication.Providers;**

**[assembly: OwinStartup(typeof(WebApiTokenAuthentication.Startup))]**

**namespace WebApiTokenAuthentication{**

**public class Startup{**

**public void Configuration(IAppBuilder app){**

**// For more information on how to configure your application, visit https://go.microsoft.com/fwlink/?LinkID=316888**

**//enable cros origin request**

**app.UseCors(CorsOptions.AllowAll);**

**//configure oAuth server configuration**

**var myProvider = new MyAuthorizationServerProvider();**

**OAuthAuthorizationServerOptions options = new OAuthAuthorizationServerOptions{**

**AllowInsecureHttp = true,**

**//this is the path that the user can request Token**

**TokenEndpointPath = new PathString("/token"),**

**//the time that the Token will be valid**

**AccessTokenExpireTimeSpan = TimeSpan.FromDays(1),**

**//pass the provider we created**

**Provider = myProvider};**

**app.UseOAuthAuthorizationServer(options);**

**app.UseOAuthBearerAuthentication(new OAuthBearerAuthenticationOptions());**

**//register HttpConfiguration**

**HttpConfiguration config = new HttpConfiguration();**

**WebApiConfig.Register(config);}}}**

**4-on the project 🡪 add folder providers > class called**

**using Microsoft.Owin.Security.OAuth;**

**using System.Security.Claims;**

**using System.Threading.Tasks;**

**namespace WebApiTokenAuthentication.Providers{**

**public class MyAuthorizationServerProvider : OAuthAuthorizationServerProvider{**

**public override async Task ValidateClientAuthentication(OAuthValidateClientAuthenticationContext context){**

**context.Validated(); // it will validate the client application}**

**//we will validate the credential of user and generate the signed Token from the information passed on register**

**public override async Task GrantResourceOwnerCredentials(OAuthGrantResourceOwnerCredentialsContext context)**

**{**

**//we create**

**var identity = new ClaimsIdentity(context.Options.AuthenticationType);**

**if (context.UserName == "admin" && context.Password == "admin")**

**{**

**identity.AddClaim(new Claim(ClaimTypes.Role, "admin"));**

**identity.AddClaim(new Claim("username", "admin"));**

**identity.AddClaim(new Claim(ClaimTypes.Name, "testing admin"));**

**context.Validated(identity);**

**}**

**else if (context.UserName == "user" && context.Password == "user")**

**{**

**identity.AddClaim(new Claim(ClaimTypes.Role, "user"));**

**identity.AddClaim(new Claim("username", "user"));**

**identity.AddClaim(new Claim(ClaimTypes.Name, "testing user"));**

**context.Validated(identity);**

**}**

**else**

**{**

**context.SetError("invalid\_grant", "provided username and password is incorrect");**

**return;**

**}**

**}**

**}**

**}**