Social Login Assignment

V1.0

SalesPoint Inc.

Purpose

This document is created for SalesPoint Inc. and covers all the details of the technical details and setup requirements. It defines all the steps being taken for the development.

Development details

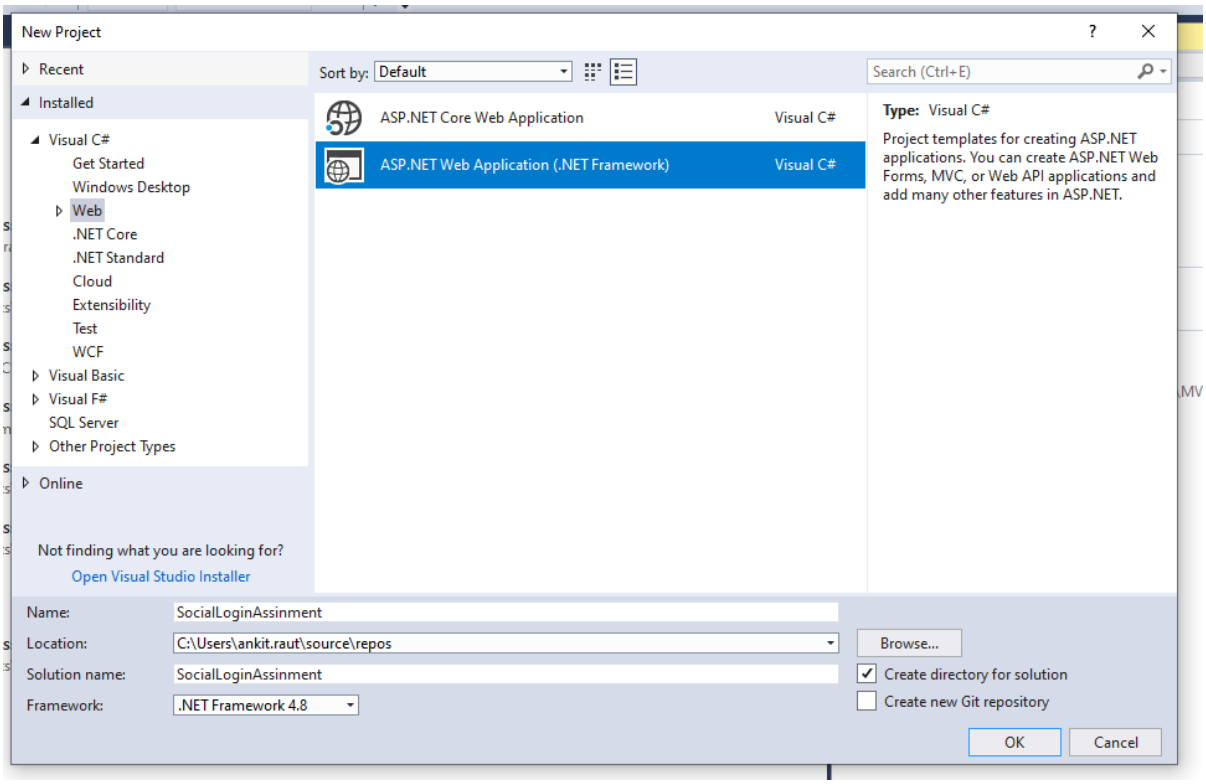
Problem statement :

Create a project in MVC (C#) with MySQL which executes the following functions:

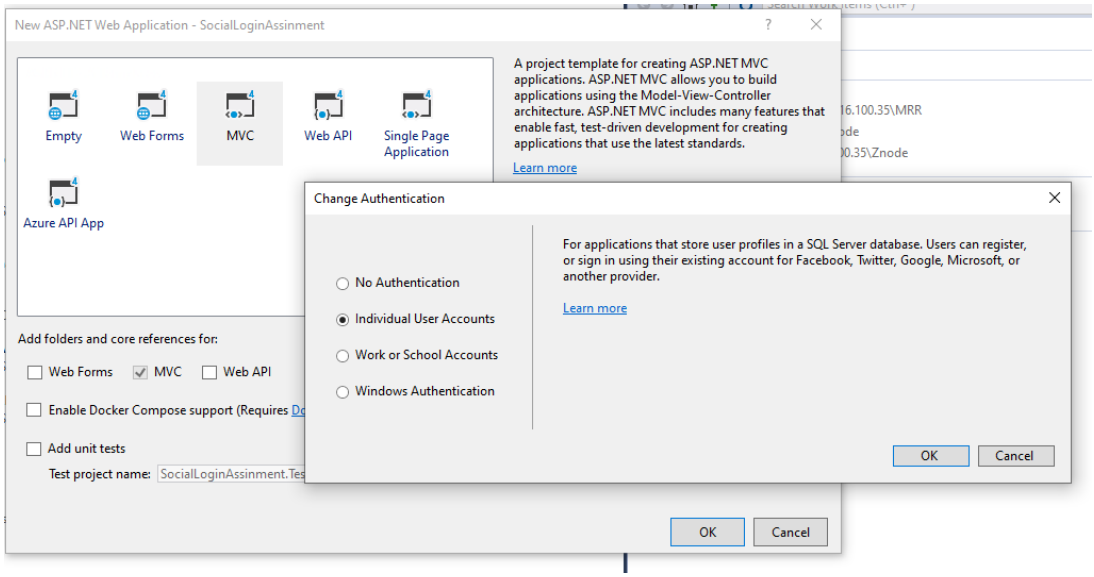
- Code first approach for database creation.
- Registration with Facebook, twitter, LinkedIn, Gmail.
- Response data should be stored in a database table having name user registration.
- Login with Facebook, twitter, LinkedIn, Gmail.
- Create a document which explains all the steps of coding.

<u>Step 1 : Project creation and initial setup :</u>

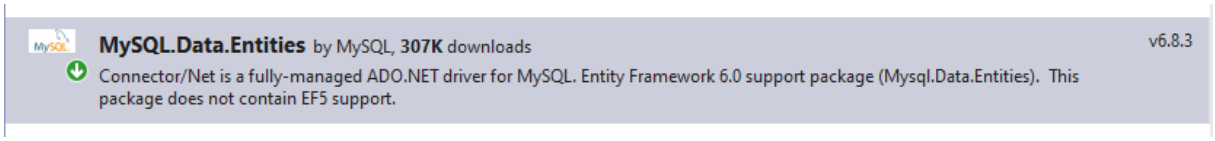
1. Create an ASP.NET MVC 5 Web Application on Visual Studio.
   Go to File — > New — > Project.



2. In the next prompt select MVC from the list of templates, and added Identity framework by clicking on “Change Authentication” button and selecting “Individual User Accounts”.



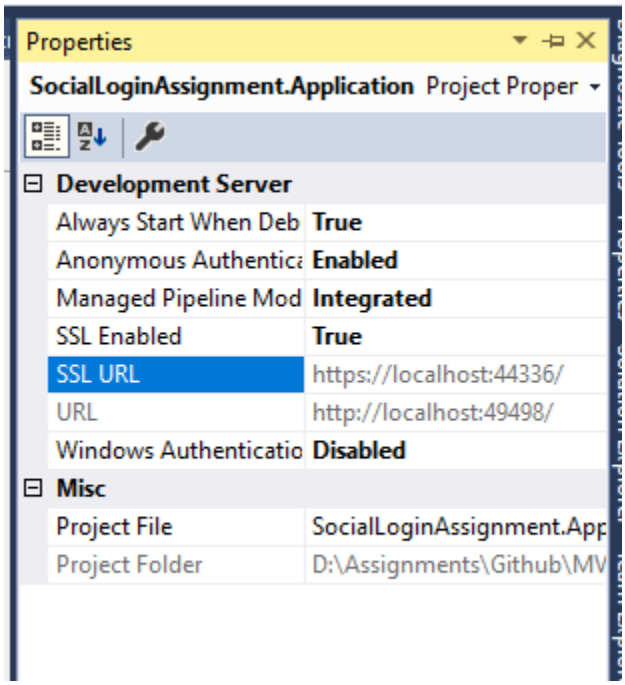
3. Now to work with MySql database I need to install a package namely “**MySql.Data.Entities**”.



4. Once it is done it should appear in the providers under entity framework in Web.config file. If it does not appear under Web.Config file then copy paste below code.

```
<entityFramework>
    <defaultConnectionFactory
type="System.Data.Entity.Infrastructure.LocalDbConnectionFactory,
EntityFramework">
      <parameters><parameter value="mssqllocaldb" /></parameters>
    </defaultConnectionFactory>
    <providers>
      <provider invariantName="System.Data.SqlClient"
type="System.Data.Entity.SqlServer.SqlProviderServices,
EntityFramework.SqlServer" />
      <provider invariantName="MySql.Data.MySqlClient"
type="MySql.Data.MySqlClient.MySqlProviderServices, MySql.Data.Entity.EF6,
Version=6.8.3.0, Culture=neutral, PublicKeyToken=c5687fc88969c44d">
      </provider>
    </providers>
  </entityFramework>
```

5. Now press “**F4**” on the project and make “**SSL Enabled**” value to “**True**”. Also add an SSL enabled port in the URL like “**https://localhost:44336/**”.



6. Now right-click on project and goto “**Web**” tab from the left side panel and change project URL to “**https://localhost:44336/**”.

Step 2 : MySql Database creation and initial setup :

Pre-requested :

Install MySql database and MyPhpAdmin/MySQL Workbench

1. Now Create a Database in MySQL namely “SocialLogin”.

```
CREATE DATABASE SocialLogin;
```

2. Add connectionstring as below in Web.Config file. Also change userId and password field as per your connection configuration.

```
 <connectionStrings>
    <add name="DefaultConnection" providerName="MySql.Data.MySqlClient"
connectionString="server=localhost;port=3306;userid=root;password=admin1234
5;database=sociallogin;persistsecurityinfo=True" />
  </connectionStrings>
```

3. As we are using IdentityDbContext, so need to map it with MySql configuration. To do this, add below code in IdentityModel.cs file.

```
    [DbConfigurationType(typeof(MySqlEFConfiguration))]
    public class ApplicationDbContext : IdentityDbContext<ApplicationUser>
    {
     .....
        public static ApplicationDbContext Create()
        {
            DbConfiguration.SetConfiguration(new MySqlEFConfiguration());
            return new ApplicationDbContext();
        }
     .....
     }
```

4. To map SQL Server database model to MySQL database model added below code in ApplicationDbContext class.

```
protected override void OnModelCreating(DbModelBuilder modelBuilder)
        {

modelBuilder.Conventions.Remove<System.Data.Entity.ModelConfiguration.Conve
ntions.PluralizingTableNameConvention>();
            base.OnModelCreating(modelBuilder);

            modelBuilder.Properties().Where(x =>
                    x.PropertyType.FullName != null &&
                    (x.PropertyType.FullName.Equals("System.String") &&

!x.GetCustomAttributes(false).OfType<ColumnAttribute>().Any(q => q.TypeName
!= null &&
                    q.TypeName.Equals("varchar(max)",
StringComparison.InvariantCultureIgnoreCase)))).Configure(c =>
                    c.HasColumnType("varchar(65000)"));

            modelBuilder.Properties().Where(x =>
                    x.PropertyType.FullName != null &&
                    (x.PropertyType.FullName.Equals("System.String") &&

!x.GetCustomAttributes(false).OfType<ColumnAttribute>().Any(q => q.TypeName
!= null &&
                    q.TypeName.Equals("nvarchar",
StringComparison.InvariantCultureIgnoreCase)))).Configure(c =>
                    c.HasColumnType("varchar"));
        }
```

Step 3 : Database Migration :

To perform code first approach follow below steps :

1. go to Tools -> NuGet Package Manager -> Package Manager Console. This will open up a console window and execute below command. This will create a “Migrations” folder which is used to store migration file.

```
enable-migrations
```

2. Now execute the below command to create a migration file which is used to initial setup the database.

```
add-migration IntialDBMigration
```

3. To update the database according to this DataModel and change, we need to execute below command.

```
update-database
```

Step 4 : Setup Social Media site for Login :

Please follow the below URL and steps to create developer APP and configure oAuth API authentication.

1. Facebook : https://docs.microsoft.com/en-us/aspnet/mvc/overview/security/create-an-aspnet-mvc-5-app-with-facebook-and-google-oauth2-and-openid-sign-on
2. Google : https://docs.microsoft.com/en-us/aspnet/mvc/overview/security/create-an-aspnet-mvc-5-app-with-facebook-and-google-oauth2-and-openid-sign-on
3. Twitter : https://www.tutlane.com/tutorial/aspnet-mvc/oauth-login-with-twitter-in-asp-net-mvc-website-with-example
4. LinkedIn : https://www.codeproject.com/Articles/874207/LinkedIn-Authentication-in-ASP-NET-MVC

Note :

By default, external authentication in ASP.NET supports only MicrosoftAccount, TwitterAuthentication, FacebookAuthentication, and GoogleAuthentication but we need to use LinkedIn authentication for that we need to install a package from the nuget called Owin.Security.Providers.

```
Install-Package Owin.Security.Providers
```

Step 4 : Configure APP to add Social Login :

1. Go to Startup.Auth.Cs file and below code under **ConfigureAuth** method

```
app.UseTwitterAuthentication(
                consumerKey:
ConfigurationManager.AppSettings["TwitterConsumerKey"].ToString(),
                consumerSecret:
ConfigurationManager.AppSettings["TwitterConsumerSecret"].ToString()
                );

            app.UseFacebookAuthentication(
                appId:
ConfigurationManager.AppSettings["FacebookAppId"].ToString(),
                appSecret:
ConfigurationManager.AppSettings["FacebookAppSecret"].ToString()
                );

            app.UseGoogleAuthentication(new
GoogleOAuth2AuthenticationOptions()
            {
                ClientId =
ConfigurationManager.AppSettings["GoogleClientId"].ToString(),
                ClientSecret =
ConfigurationManager.AppSettings["GoogleClientSecret"].ToString()
            });

            app.UseLinkedInAuthentication
            (
                clientId:
ConfigurationManager.AppSettings["LinkedinClientId"].ToString(),
                clientSecret:
ConfigurationManager.AppSettings["LinkedinClientSecret"].ToString()
            );
```

2. Add Keys and Value of Apps created in Step 3 into **Web.Config**.

```
 <!--Twitter Auth Key-->
<add key="TwitterConsumerKey" value="BmZvFP51JeMMt2Iy1f9reP44r" />
    <add key="TwitterConsumerSecret"
value="d5Dn8cesi6TYySHr8gnaqDttrzLdOpZ2Ho7lXBKIEyfajY1Npj" />

<!--Facebook Auth Key-->
    <add key="FacebookAppId" value="526118978766719" />
    <add key="FacebookAppSecret"
value="7c4bc9e55a7a0f23b99a6bf7c3781335" />

 <!--Google Auth Key-->
    <add key="GoogleClientId"
value="221163763668-7k3b70op8jjfqck1o04qnl1uh2qmhb9f.apps.googleuserc
ontent.com" />
    <add key="GoogleClientSecret" value="v3pCGW3hdlZu0Vu1eyoAXLEJ" />

<!--LinkedIn Auth Key-->
    <add key="LinkedinClientId" value="784bcvaiv2mj61" />
    <add key="LinkedinClientSecret" value="8qy3uqbF7j0kqMlu" />
```