# Logo Description automatically generated

Application Setup

Identity Framework with SQLite

## Application Setup

Since roles help you to manage user accounts, you will need to create a new **ASP.NET Core Web Application** with **Individual Accounts**. Creating a new web application and when prompted, choose these options.

|  |
| --- |
| A screenshot of a computer  Description automatically generated |

Name your application: PayPalDemo.

|  |
| --- |
|  |

In the final dialog select Individual Accounts.

|  |
| --- |
| A screenshot of a computer  Description automatically generated |

|  |
| --- |
| Create |

## Identity Framework Scaffolding

You will notice when using the identity framework that pages such as registration and account confirmation are automatically generated. These pages however, are not initially made visible in your project. To make the Identity Framework class files visible, right click the **Areas** folder,

and choose **Add | New Scaffold Item**. Then choose **Identity**, **identity**, **Add**.

|  |  |  |  |
| --- | --- | --- | --- |
| |  | | --- | | A screenshot of a computer  Description automatically generated with medium confidence | | |  | | --- | | Graphical user interface, application, website  Description automatically generated | |

In the final dialog, select **Override all files**, select your database context in the dropdown and click **Add**.

|  |
| --- |
| Graphical user interface  Description automatically generated with medium confidence |

## Converting the connection to SQLite

Follow the steps below to change your database connection from the default mssqllocalDb to SQLite.

1. Open the Package Manager Console by clicking:

**Tools 🡪 NuGet Package Manager 🡪 Manage NuGet Packages for Solution**

Install the following Entity Framework packages. Make sure the installed versions are

compatible with your application.

|  |  |
| --- | --- |
| PM>  PM> | Install-Package Microsoft.EntityFrameworkCore.Sqlite -Version 8.0  Install-Package Microsoft.EntityFrameworkCore -Version 8.0 |

1. Your installed packages should look like this:

|  |  |
| --- | --- |
| |  | | --- | | A screen shot of a computer program  Description automatically generated | |

1. Change the DefaultConnection in the **appsettings.json** file to the following.

|  |  |
| --- | --- |
| |  | | --- | | {  "ConnectionStrings": {  "DefaultConnection": "Data Source=.\\wwwroot\\sql.db"  },  "Logging": {  "LogLevel": {  "Default": "Information",  "Microsoft.AspNetCore": "Warning"  }  },  "AllowedHosts": "\*"  } | |

1. Look for the code below in the **Program.cs** file and change the reference to UseSqlite.

|  |  |
| --- | --- |
| |  | | --- | | bbuilder.Services.AddDbContext<ApplicationDbContext>(options =>  options.UseSqlite(connectionString)); | |

1. Again in the **Program.cs** file, set the register email confirmation to false.

|  |  |
| --- | --- |
| |  | | --- | | (options => options.SignIn.RequireConfirmedAccount = false) | |

1. Delete the Migrations folder found in the Data folder.
2. To create your SQLite database and add the Identity tables to it, you will need to create and apply a migration. So, delete the Migrations folder within the Data folder if it exists. Then in the **Package Manager Console** execute the following commands.

|  |  |  |
| --- | --- | --- |
| |  |  | | --- | --- | | PM>  PM> | add-migration initialCreate  update-database | |

1. The SQLite database will be created in the directory laid out in the DefaultConnection and in this case will be found in the wwwroot folder. Copy the location of your SQLite database by right clicking on the wwwroot folder in the Solution Explorer and select **Copy Full Path**.
2. Next Run SQLite and click Open Database. Paste in the copied path then select **sql.db** and Open.

|  |
| --- |
| Graphical user interface, text  Description automatically generated |

1. To view the data, click on the Browse Data tab then select the table you want in the drop down.

|  |
| --- |
| A screenshot of a computer  Description automatically generated |