Alpha Project Maintenance Manual

This document is to support the long term maintenance and extension of the Alpha project application.

1.System description and third party dependencies:

The Alpha team Project is a web application developed in C#(v7), JavaScript(ES6) and also the frameworks .Net(v4.6.1), AngularJS(v1.5.8) and BootStrap(v3.3.7). Miscellaneous packages were installed, the list can be found in the section ACCDataStore.Web/References. Key packages include EntityFramework(v6.1.3),Microsoft.ApplicationInsights.Web(v2.1.0),Microsoft.AspNet.WebApi.Core(v5.2.3) and MySql.Data(v6.9.9).

The frameworks BootStrap and AngularJS are linked to the application with a script tag in the html code. See Figure 1.

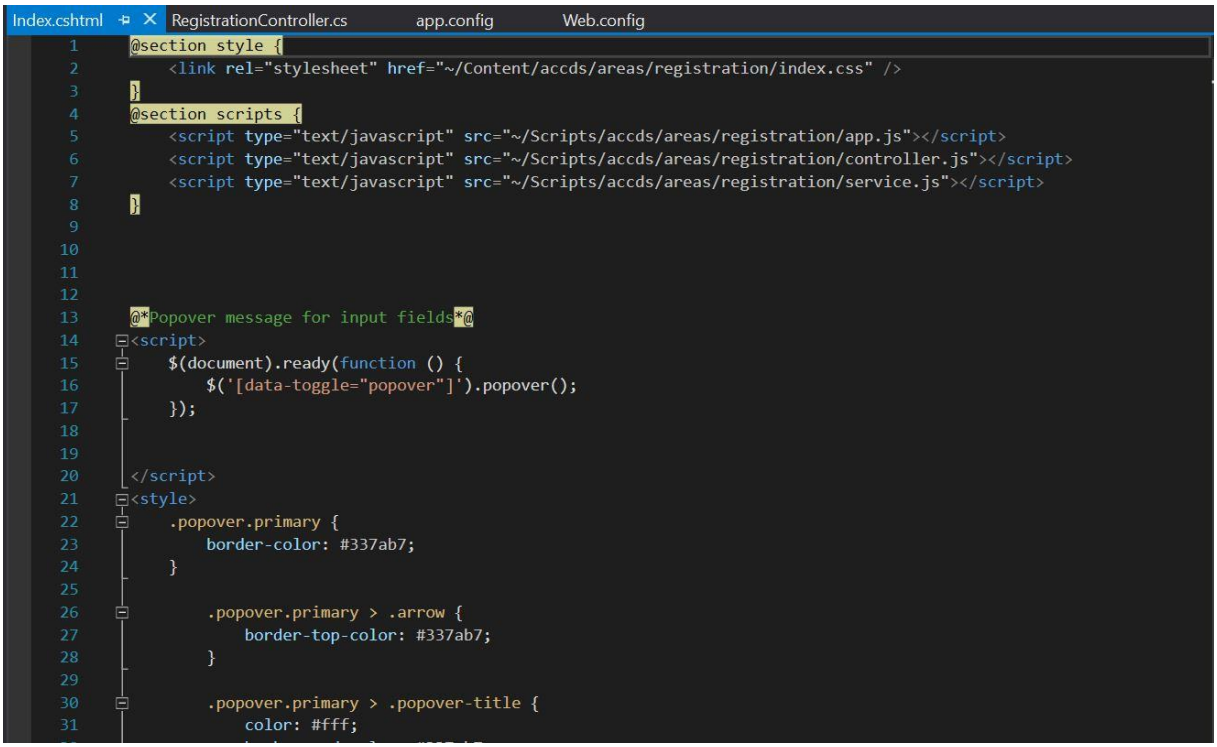

Figure 1 Bootstrap AngularJS script

The application is created to work with SendGrid which is a customer communication platform for transactional and marketing emails. See Figure 2.

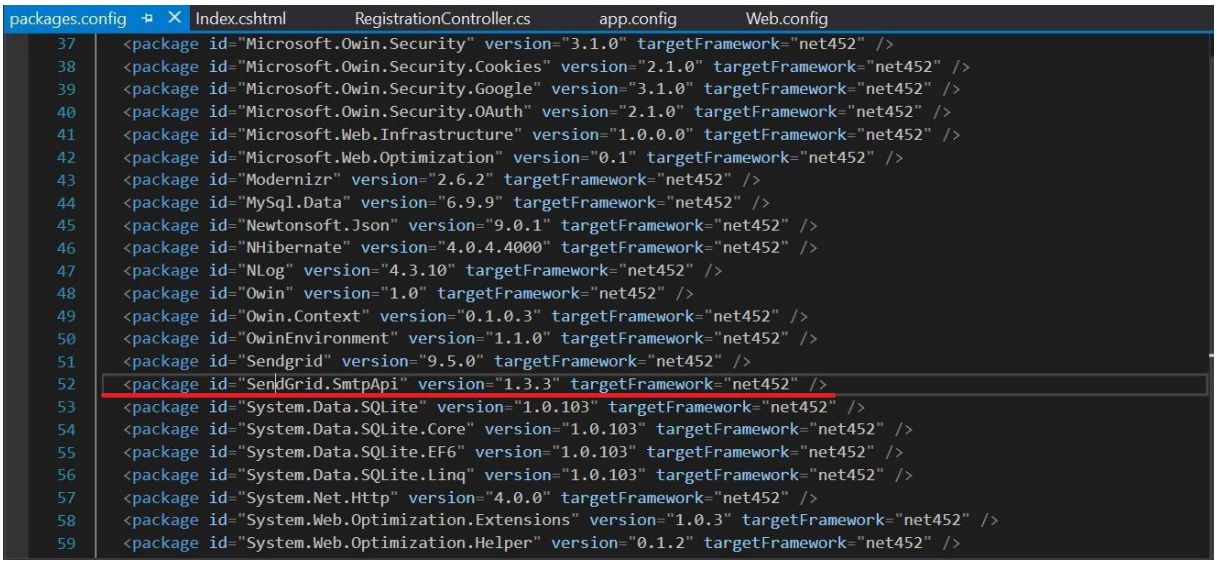

Figure 2 The Sendgrid package

The SendGrid ApiKey which is used in the application for sending emails can be used for free and provides only 100 emails per day. See Figure 3.

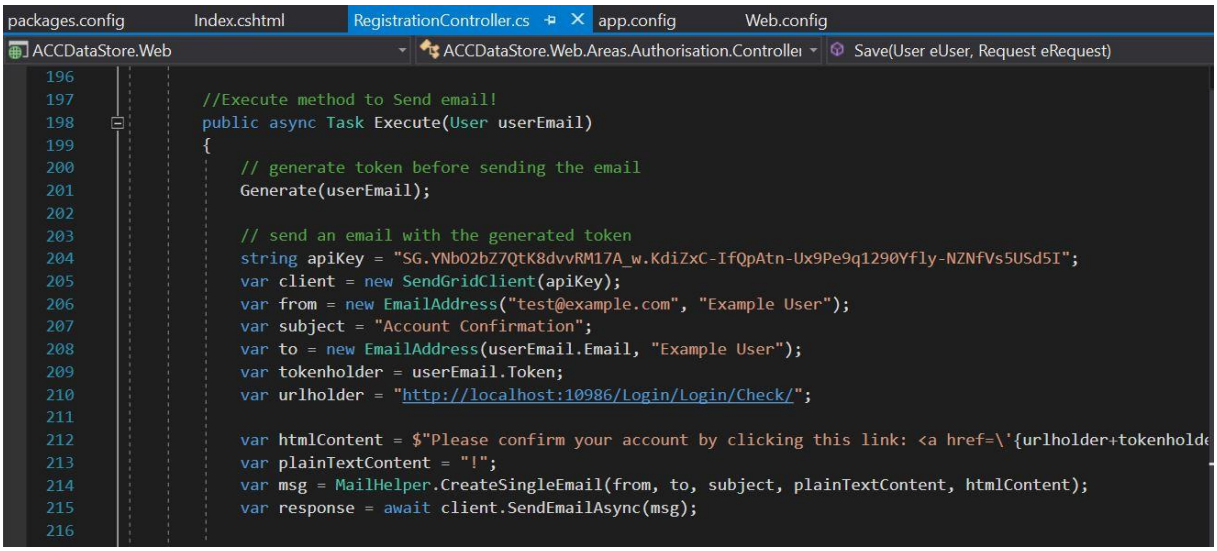

Figure 3 Sendgrid API key

The future deployment of the application will require a new ApiKey. The cheapest and most realiable option is SendGrid. More details about SendGrid can be found here: **https://sendgrid.com/docs/Integrate/Code_Examples/v2_Mail/csharp.html**

2.Installation

In order for the application to run, Visual Studio 2017(VS) has to be installed. VS is a program which helps for producing efficient and reliable software, most of the Microsoft programs and applications are build with VS. It can be downloaded from the following link: https://www.visualstudio.com/downloads/

The second software needed for running the application is MSQL WorkBench 6.3 CE, it is a visual database design tool that provides an integrated environment for database design, modeling, SQL development, database administration and database migration. It can be downloaded from the following link: https://dev.mysql.com/downloads/workbench/

After installing Visual Studio and MSQL WorkBench, the following steps must be done, in order for the application to be run:

1. Open the application in Visual Studio.
2. Open MySQL Workbench.
3. Create a new connection in MySQL workbench. See Figure 4 and 5.

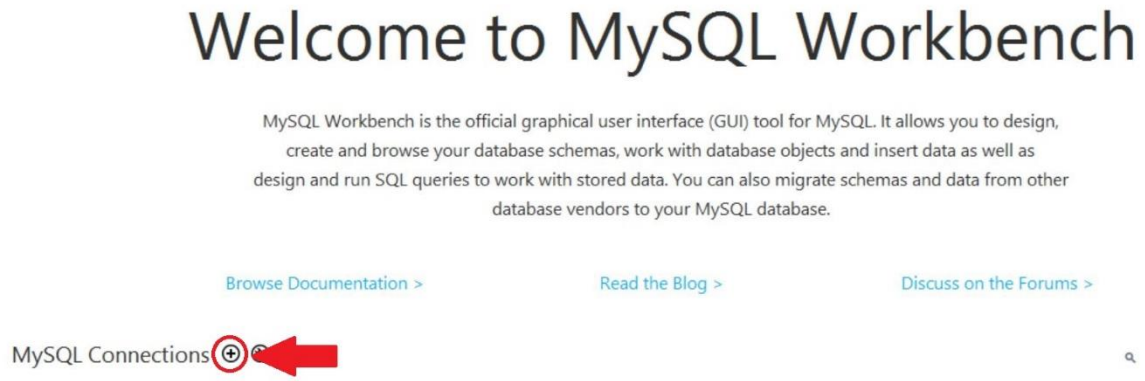

Figure 4 Create a new connection

- Select any name for the *"Connection Name"* and leave the other details by default

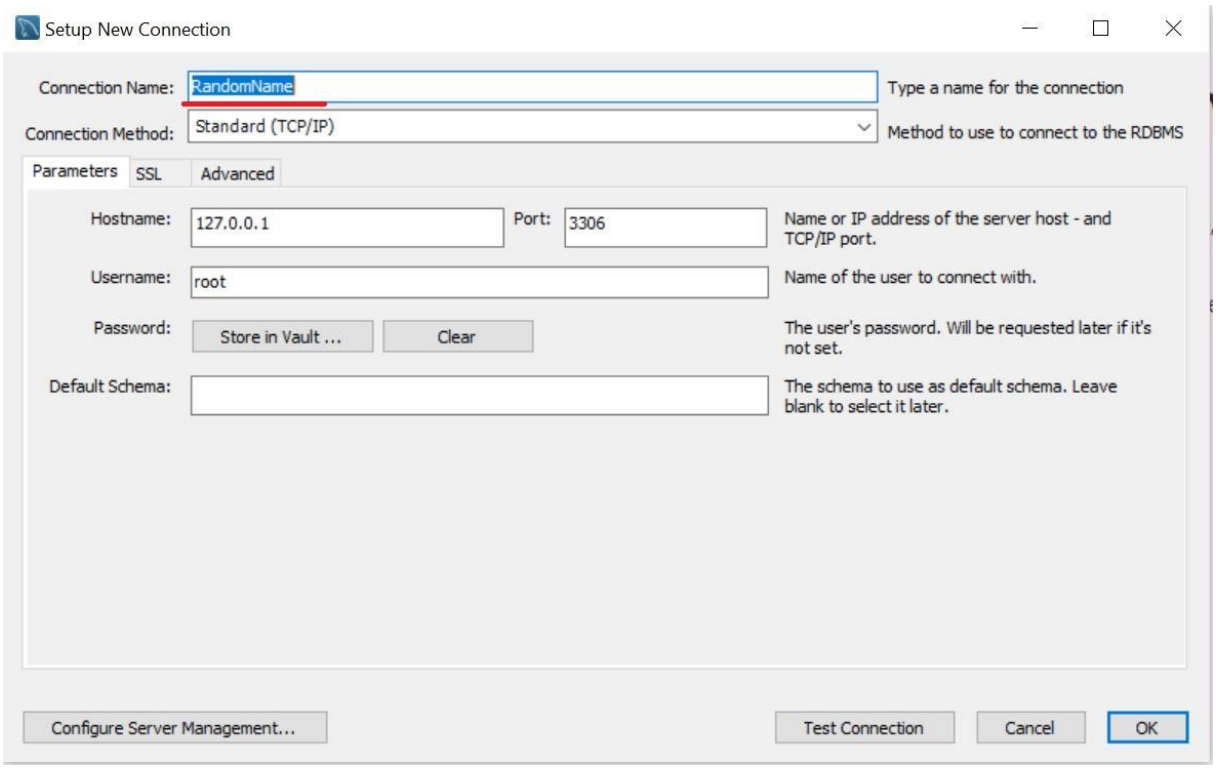

Figure 5 Create a new connection

- Select the connection you have created by double clicking it. See Figure 6.

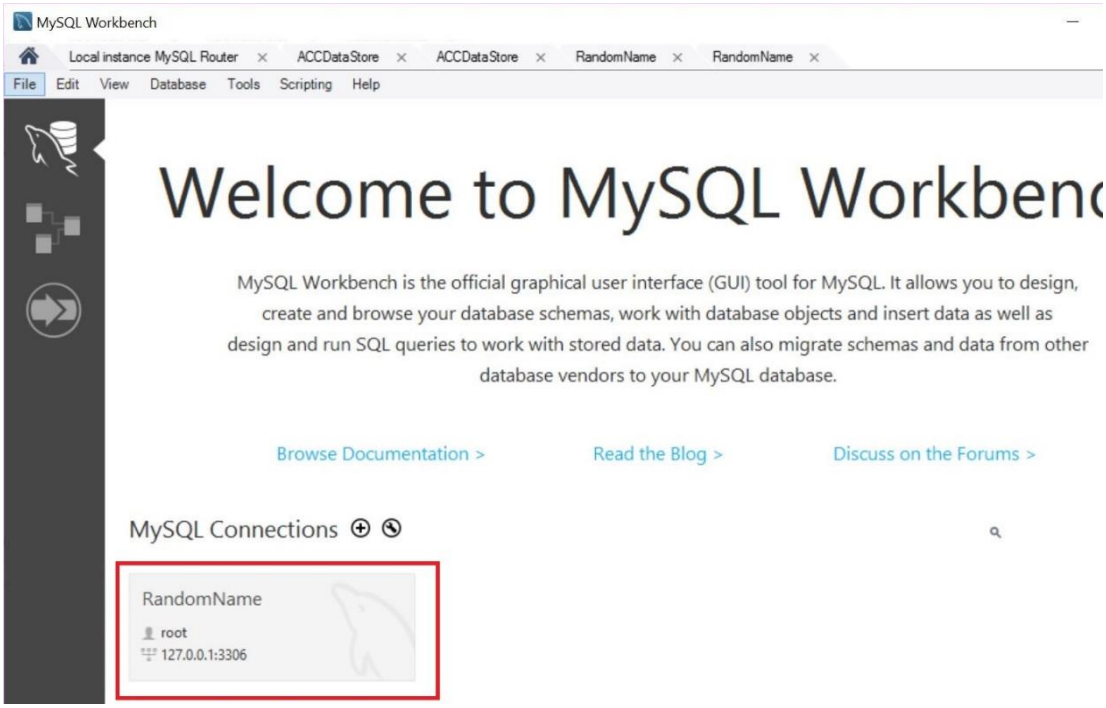

Figure 6 Open the newly created MySQL connection

- After the connection has been opened, import the Database, located in the main folder(*accdatastore.sql*). See Figure 7.

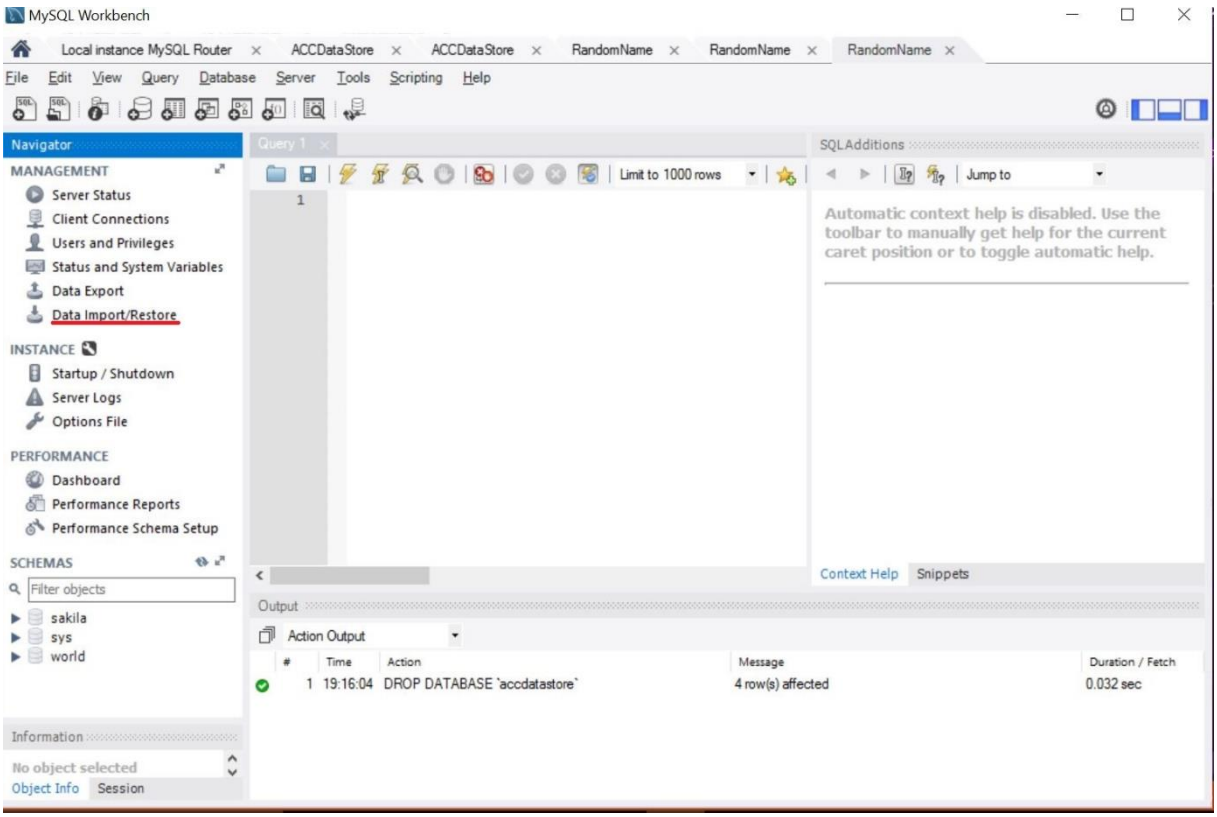

Figure 7 Import Database

- Select “Import from Self-Contained File” and find the Database you would like to import. See Figure 8.

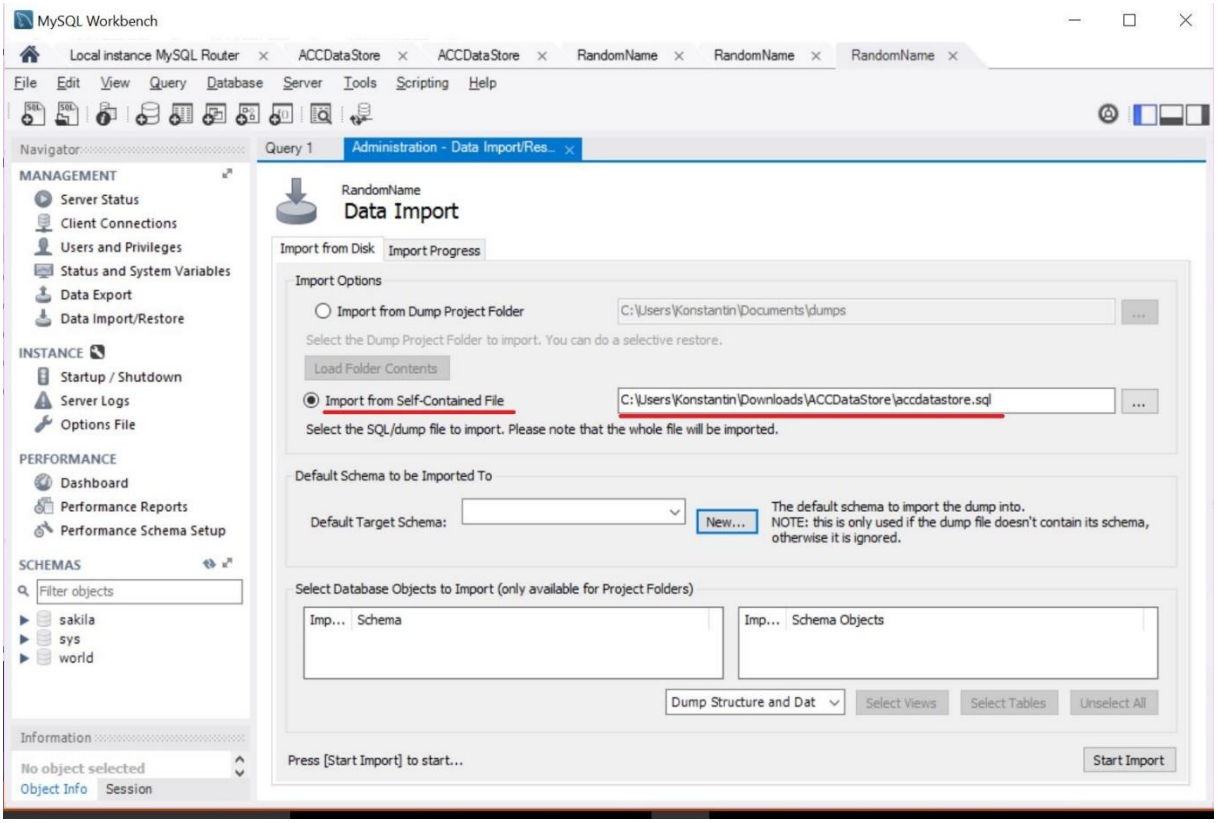

Figure 8 Import Database

- Select “New” and type in accdatastore as the DB schema. Then select “Start Import”

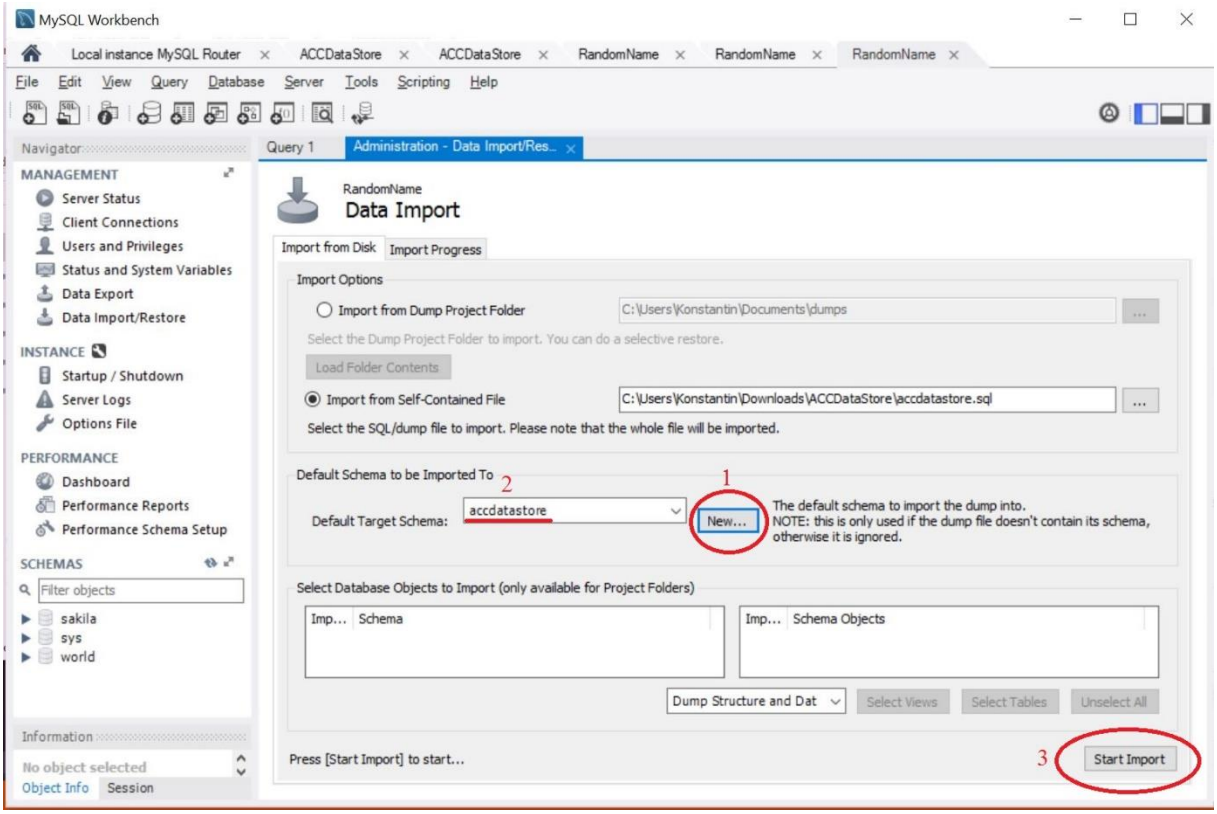

Figure 9 Import Database

- Finally, after the import has been completed “Refresh” the Schema list and you will see the accdatastore schema loaded. See Figure 10.

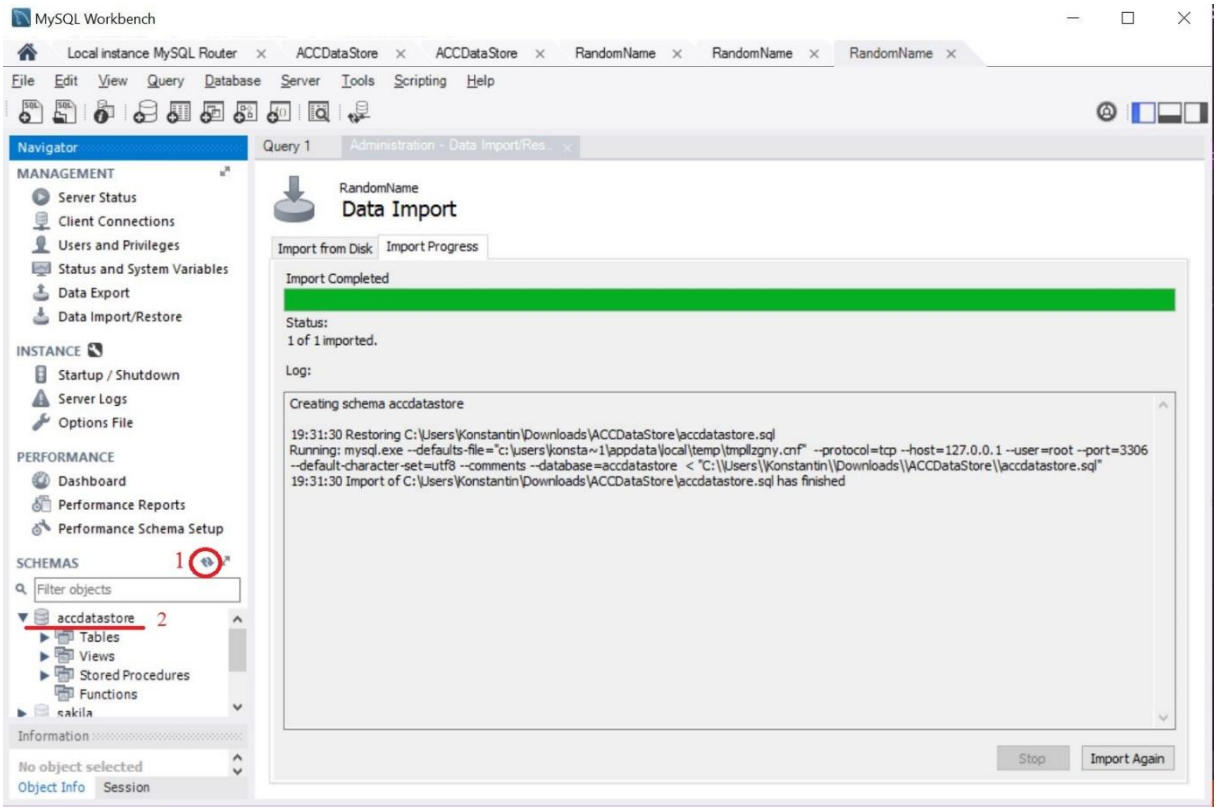

Figure 10 Refresh Schemas

- After importing the database, open the file Web.config in Visual Studio and make some changes in the html tag “ConnectionString”. See Figure 11.

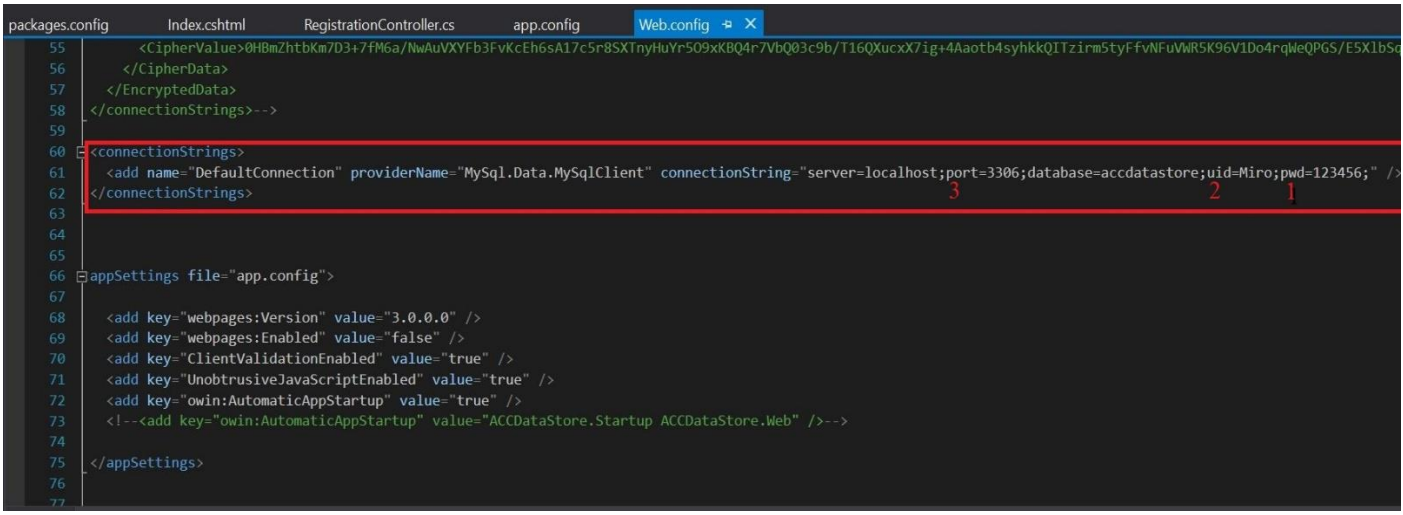

Figure 11 Webconfig Connection string

There you would need to change:

1. The password to the MySQL Workbench account password (The one you registered on MySQL Workbench)
2. MySQL username
3. MySQL port

2.1 Open the file app.config. See figure 12.

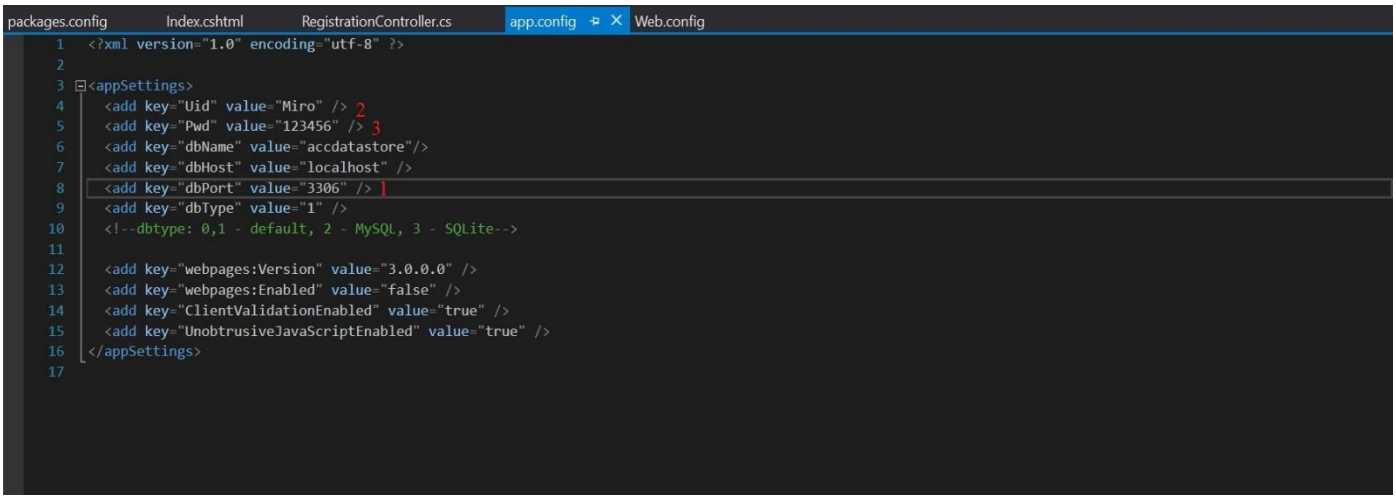

Figure 12 AppConfig Connection String

1. MySQL port
2. Account Username
3. Account Password

2.2 Run the application by pressing the Start “Google Chrome” button in VS. See Figure 13.

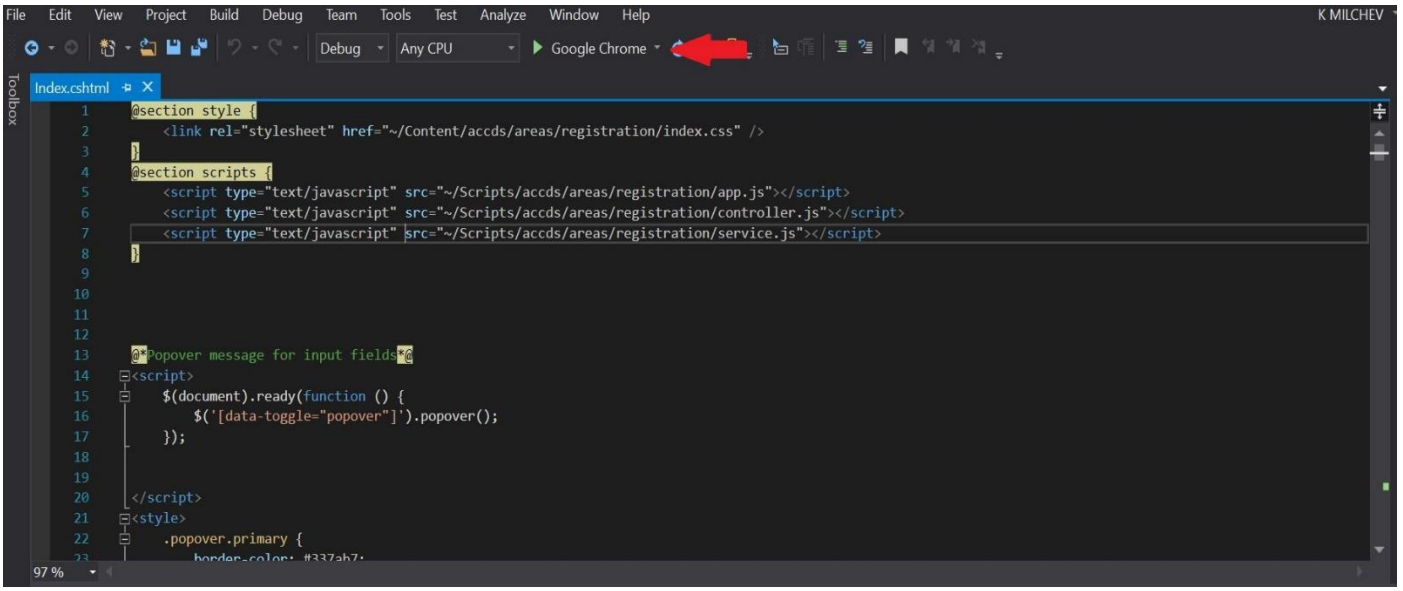

Figure 13 Run Project

2.3 In order to login with Admin controls, the following account must be used:

- email: admin@gmail.com
- password: Admin12345

3. List of other important packages installed

ID	Version	Description
Moq	{4.7.63}	Mocking Framework for .NET
MSTest.TestFramework	{1.1.11}	Discover and Execute Tests
NUnit	{3.0.1}	Unit Test .NET Framework
MySql.Data	{6.9.9}	ADO.NET driver for MySql
Antlr	{3.4.1.9004}	Language Recognition Framework
Autofac	{4.1.1}	Manages dependencies between classes
BCrypt	{1.0.0.0}	Password Hashing Function
bootstrap	{3.0.0}	A Front-End HTML, CSS and JS Framework
FluentNHibernate	{2.0.3.0}	Writing mappings through strongly types C# code
jQuery	{1.10.2}	Javascript library for Event Handling and DOM Manipulation
JSAndCSSBundling	{1.0.1}	Configuration for the Miscrosoft.Web.Optimization package
Microsoft.AspNet.Identity.Owin	{2.2.1}	Owin usage in ASP.NET
Microsoft.AspNet.Mvc	{5.2.3}	Has the runtime assemblies for ASP.NET
Microsoft.AspNet.Razor	{3.2.3}	Has the runtime assemblies for ASP.NET but utilizes the Razor syntax
Microsoft.AspNet.Web.Optimization	{1.1.3}	Presents a way to bundle together CSS and JS
Microsoft.Owin	{3.1.0}	Provides a simple way for making OWIN elements
Modernizr	{2.6.2}	Helps for the use of new components (CSS3, HTML5) while still having control over older browsers
Newtonsoft.Json	{9.0.1}	JSON framework created for .NET
NHibernate	{4.0.4.4000}	Object Relational Mapper for .NET
NLog	{4.3.10}	.NET framework for better and more optimal logging
Sendgrid	{9.5.0}	Uses the sendgrid API to send emails
WebGrease	{1.5.2}	Tool for optimizing CSS, JS
NHibernate	{4.0.4.4000}	.NET Object Relational Mapper
Entity Framework	{6.2.0}	Official Microsoft method for data access