

# Alpha Project Maintenance Manual

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

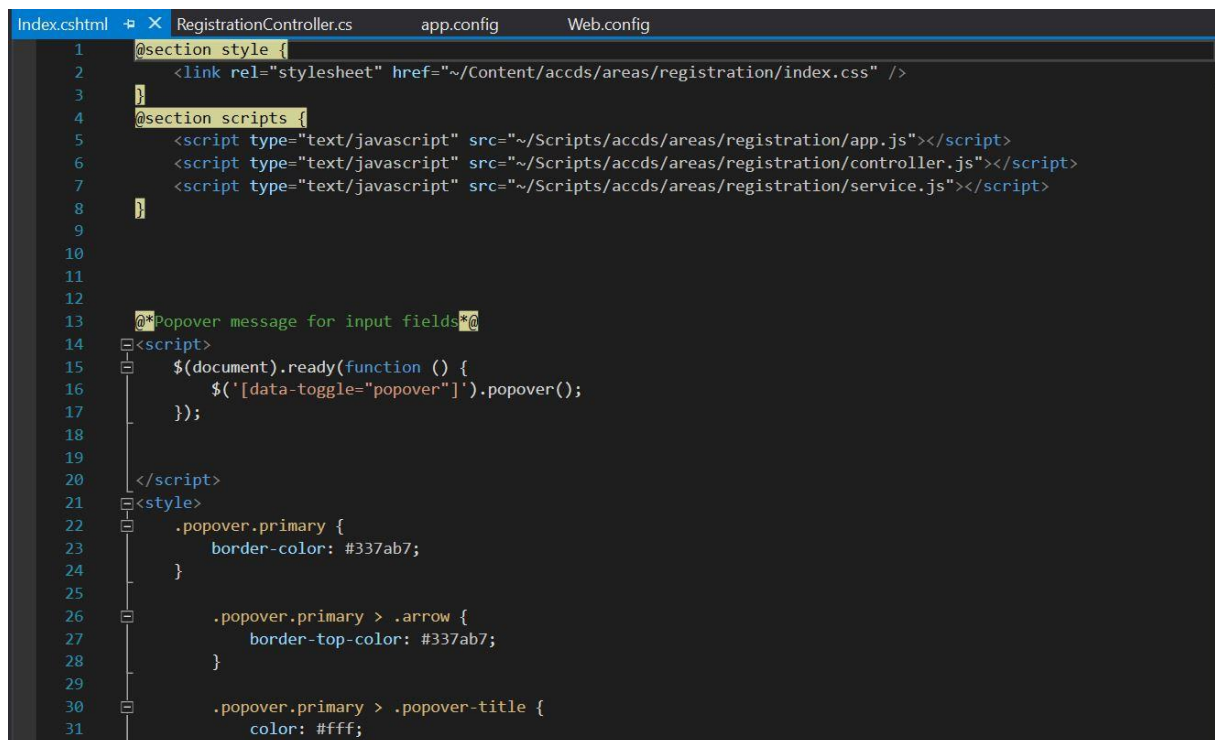
## 1.System description and third party dependencies:

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).

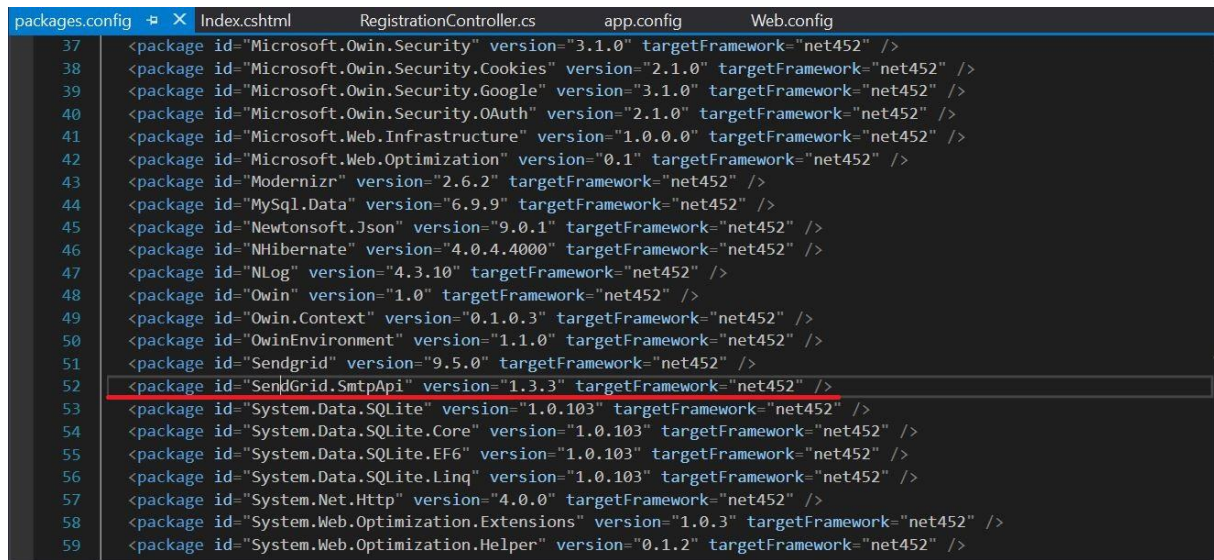
The frameworks BootStrap and AngularJS are linked to the application with a script tag in the html code. See Figure 1.



```
1  @section style {
2      <link rel="stylesheet" href="~/Content/accds/areas/registration/index.css" />
3  }
4  @section scripts {
5      <script type="text/javascript" src="~/Scripts/accds/areas/registration/app.js"></script>
6      <script type="text/javascript" src="~/Scripts/accds/areas/registration/controller.js"></script>
7      <script type="text/javascript" src="~/Scripts/accds/areas/registration/service.js"></script>
8  }
9
10
11
12
13  @*Popover message for input fields*@
14  <script>
15      $(document).ready(function () {
16          $('[data-toggle="popover"]').popover();
17      });
18  }
19  </script>
20  <style>
21      .popover.primary {
22          border-color: #337ab7;
23      }
24  }
25
26  .popover.primary > .arrow {
27      border-top-color: #337ab7;
28  }
29
30  .popover.primary > .popover-title {
31      color: #fff;
```

Figure 1 Bootstrap AngularJS script

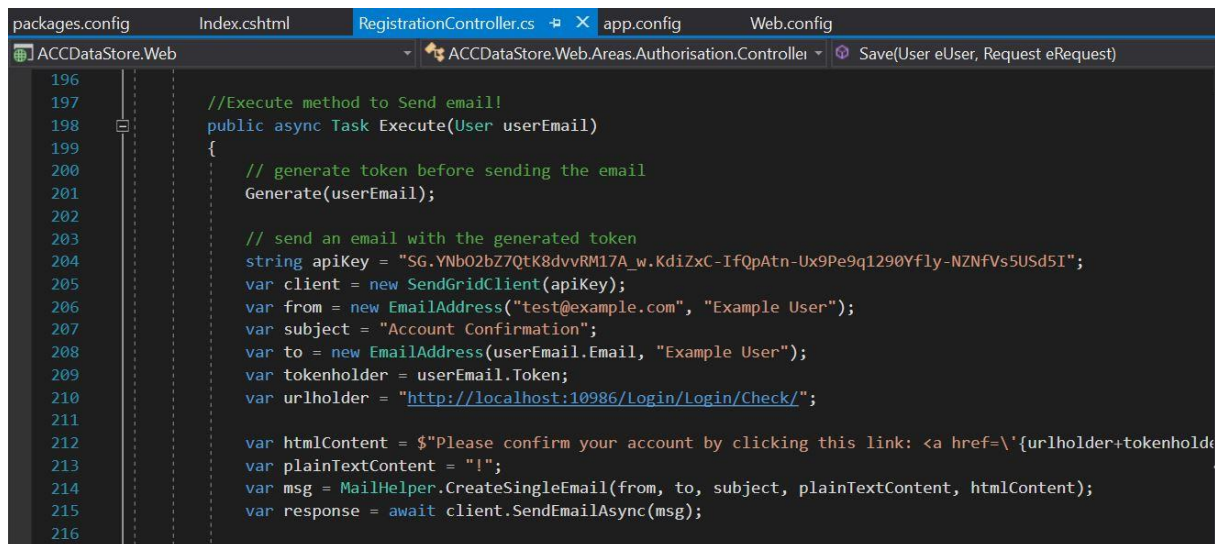
The application is created to work with SendGrid which is a customer communication platform for transactional and marketing emails. See Figure 2.



```
packages.config  Index.cshtml  RegistrationController.cs  app.config  Web.config
37 <package id="Microsoft.Owin.Security" version="3.1.0" targetFramework="net452" />
38 <package id="Microsoft.Owin.Security.Cookies" version="2.1.0" targetFramework="net452" />
39 <package id="Microsoft.Owin.Security.Google" version="3.1.0" targetFramework="net452" />
40 <package id="Microsoft.Owin.Security.OAuth" version="2.1.0" targetFramework="net452" />
41 <package id="Microsoft.Web.Infrastructure" version="1.0.0.0" targetFramework="net452" />
42 <package id="Microsoft.Web.Optimization" version="0.1" targetFramework="net452" />
43 <package id="Modernizr" version="2.6.2" targetFramework="net452" />
44 <package id="MySQL.Data" version="6.9.9" targetFramework="net452" />
45 <package id="Newtonsoft.Json" version="9.0.1" targetFramework="net452" />
46 <package id="NHibernate" version="4.0.4.4000" targetFramework="net452" />
47 <package id="NLog" version="4.3.10" targetFramework="net452" />
48 <package id="Owin" version="1.0" targetFramework="net452" />
49 <package id="Owin.Context" version="0.1.0.3" targetFramework="net452" />
50 <package id="OwinEnvironment" version="1.1.0" targetFramework="net452" />
51 <package id="Sendgrid" version="9.5.0" targetFramework="net452" />
52 <package id="SendGrid.SmtpApi" version="1.3.3" targetFramework="net452" />
53 <package id="System.Data.SQLite" version="1.0.103" targetFramework="net452" />
54 <package id="System.Data.SQLite.Core" version="1.0.103" targetFramework="net452" />
55 <package id="System.Data.SQLite.EF6" version="1.0.103" targetFramework="net452" />
56 <package id="System.Data.SQLite.Linq" version="1.0.103" targetFramework="net452" />
57 <package id="System.Net.Http" version="4.0.0" targetFramework="net452" />
58 <package id="System.Web.Optimization.Extensions" version="1.0.3" targetFramework="net452" />
59 <package id="System.Web.Optimization.Helper" version="0.1.2" targetFramework="net452" />
```

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.



```
196
197
198 //Execute method to Send email!
199 public async Task Execute(User userEmail)
200 {
201     // generate token before sending the email
202     Generate(userEmail);
203
204     // send an email with the generated token
205     string apiKey = "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";
206     var client = new SendGridClient(apiKey);
207     var from = new EmailAddress("test@example.com", "Example User");
208     var subject = "Account Confirmation";
209     var to = new EmailAddress(userEmail.Email, "Example User");
210     var tokenholder = userEmail.Token;
211     var urlholder = "http://localhost:10986/Login/Login/Check/";
212
213     var htmlContent = $"Please confirm your account by clicking this link: <a href='{urlholder+tokenholder}'>Click here</a>";
214     var plainTextContent = "!";
215     var msg = MailHelper.CreateSingleEmail(from, to, subject, plainTextContent, htmlContent);
216     var response = await client.SendEmailAsync(msg);
```

Figure 3 Sendgrid API key

The future deployment of the application will require a new ApiKey. The cheapest and most reliable option is SendGrid. More details about SendGrid can be found here:

[https://sendgrid.com/docs/Integrate/Code\\_Examples/v2\\_Mail/csharp.html](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 MS SQL 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 MS SQL 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.

# Welcome to MySQL Workbench

MySQL Workbench is the official graphical user interface (GUI) tool for MySQL. It allows you to design, create and browse your database schemas, work with database objects and insert data as well as design and run SQL queries to work with stored data. You can also migrate schemas and data from other database vendors to your MySQL database.

[Browse Documentation >](#)

[Read the Blog >](#)

[Discuss on the Forums >](#)

MySQL Connections 

Figure 4 Create a new connection

- Select any name for the “Connection Name” and leave the other details by default

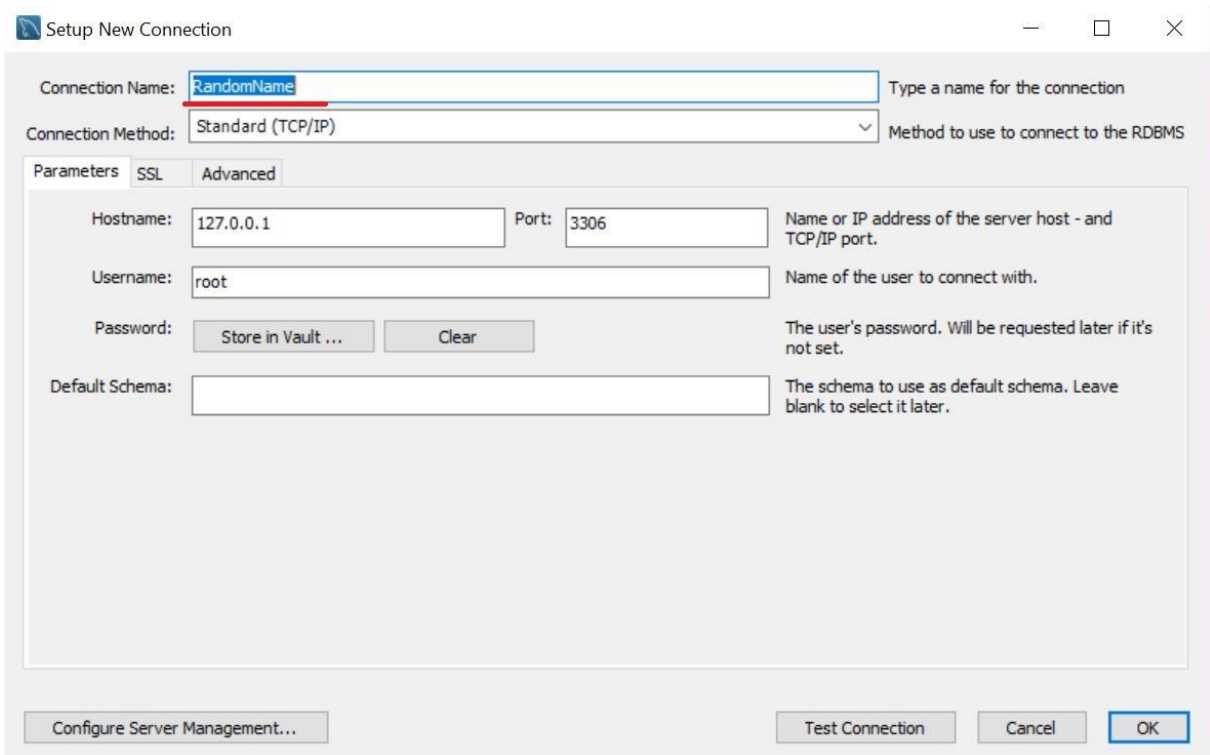


Figure 5 Create a new connection

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

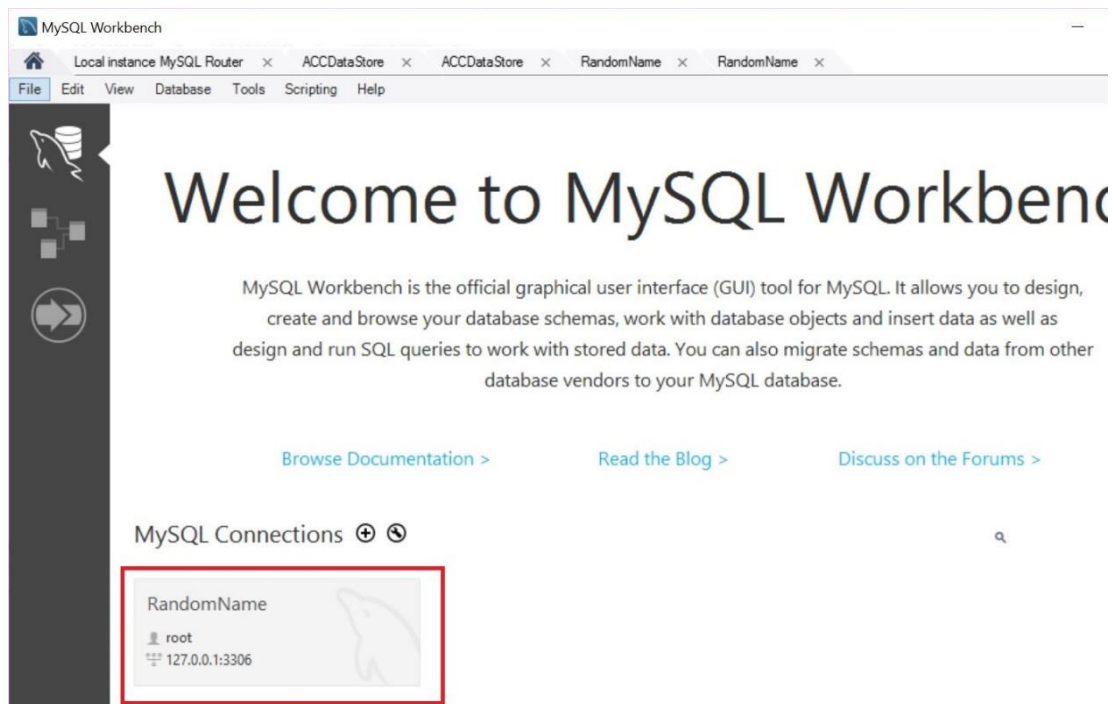


Figure 6 Open the newly created MySQL connection

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

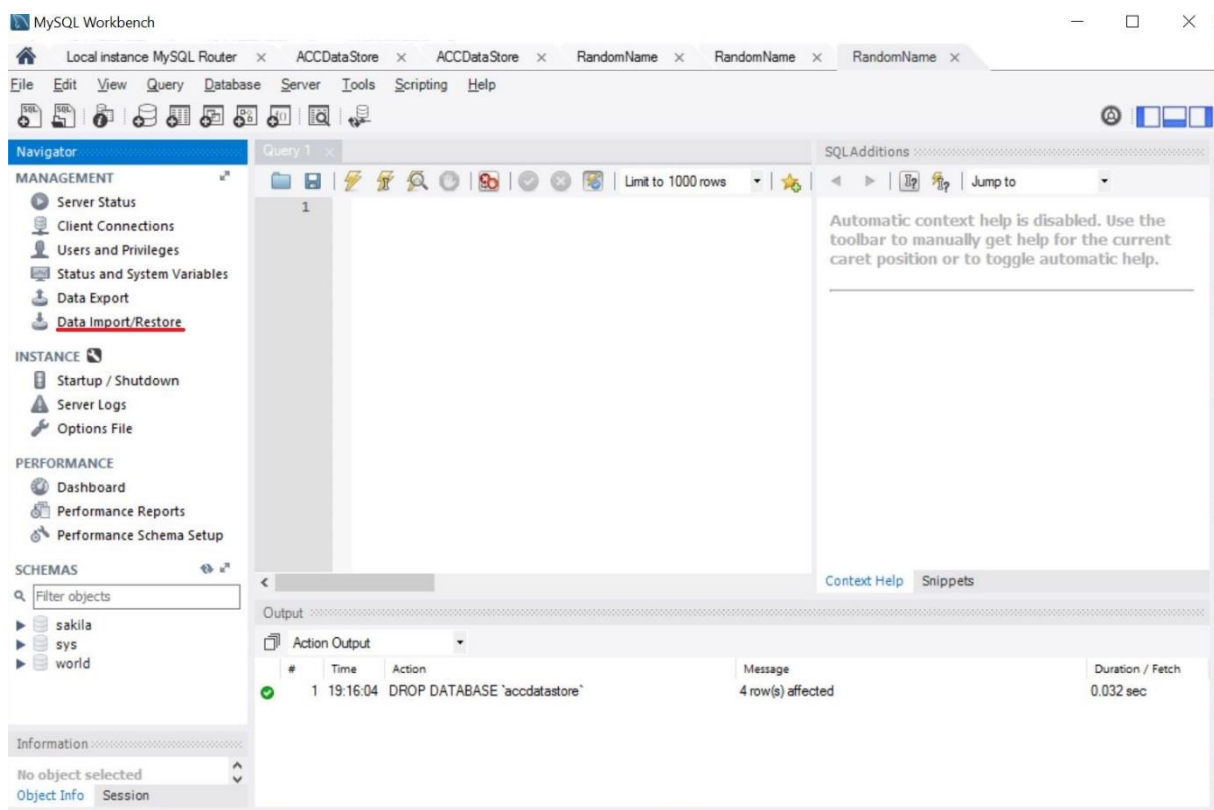


Figure 7 Import Database

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

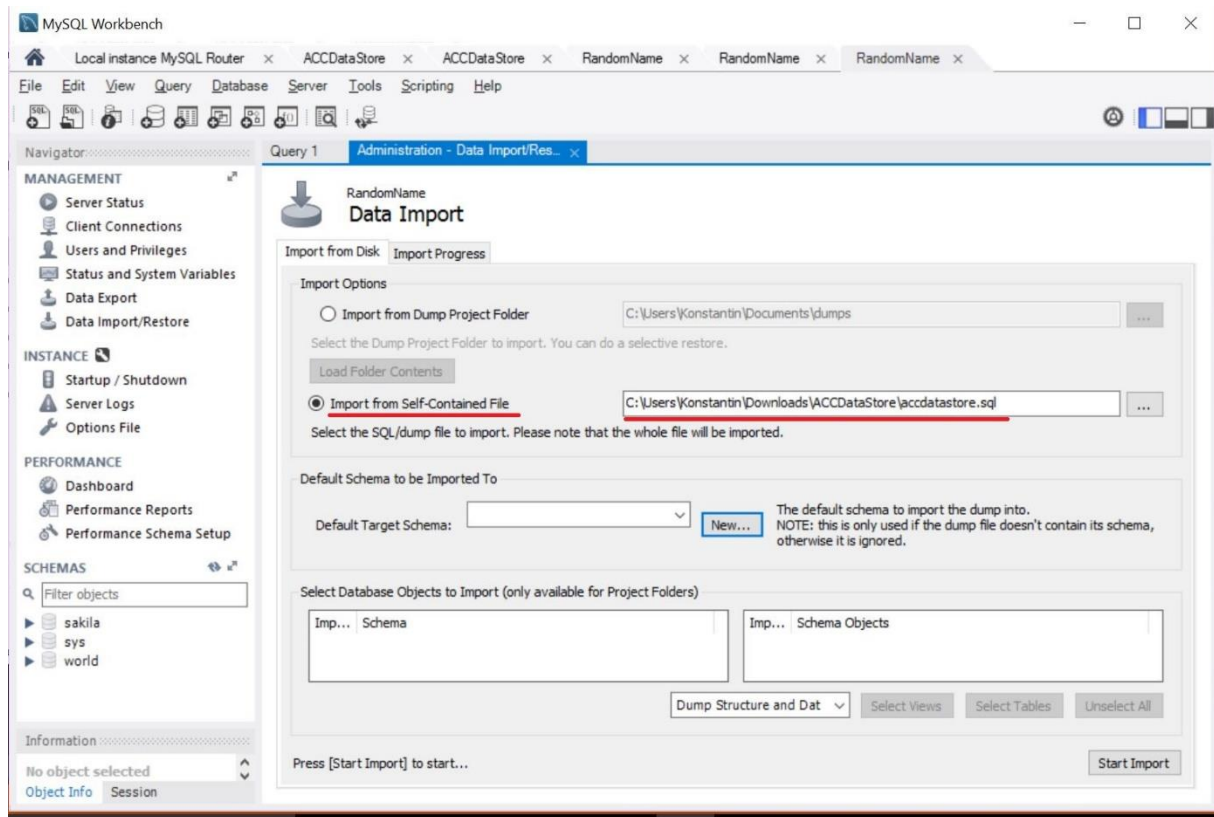


Figure 8 Import Database



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

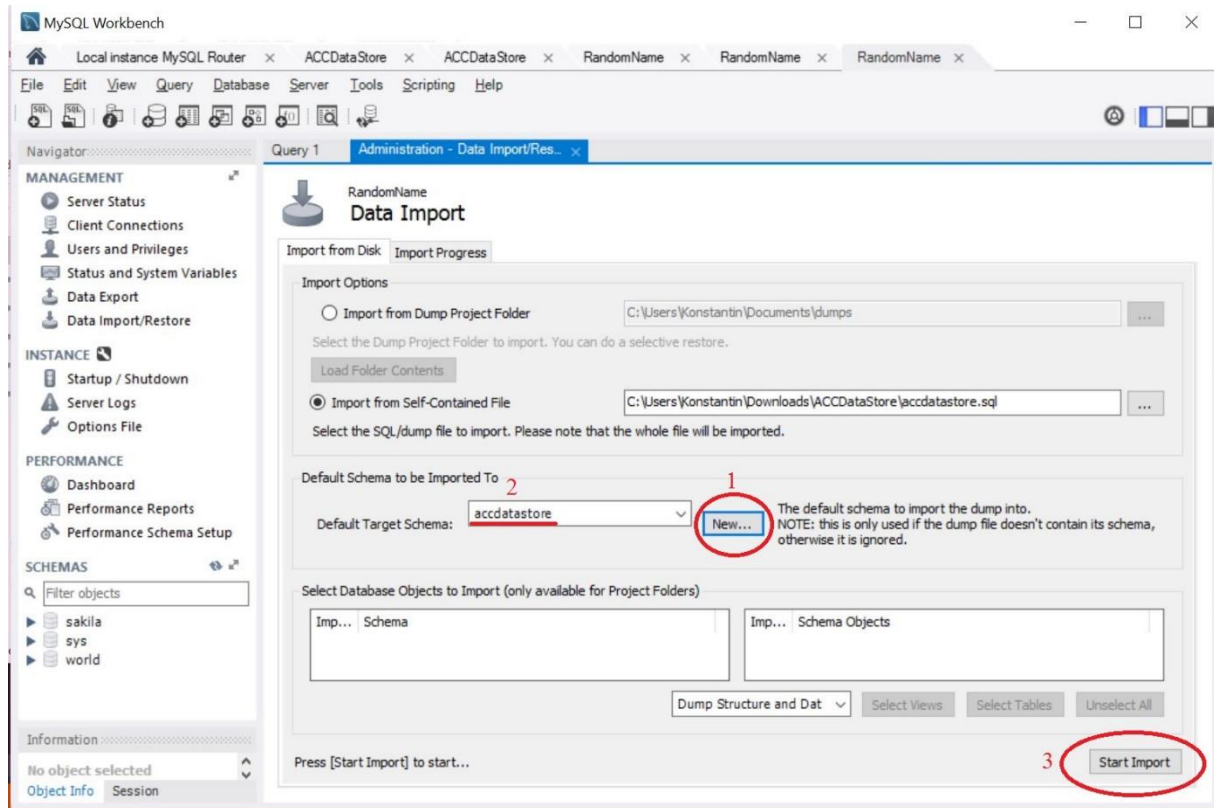


Figure 9 Import Database

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

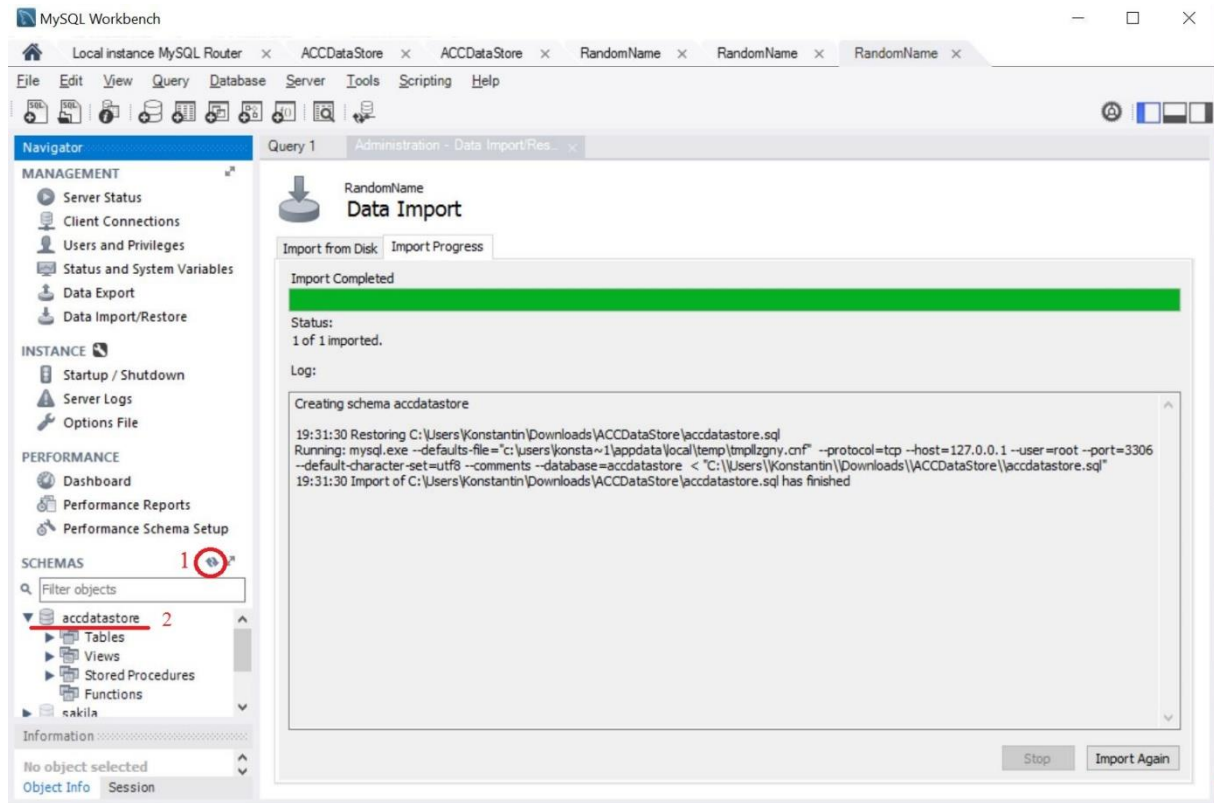
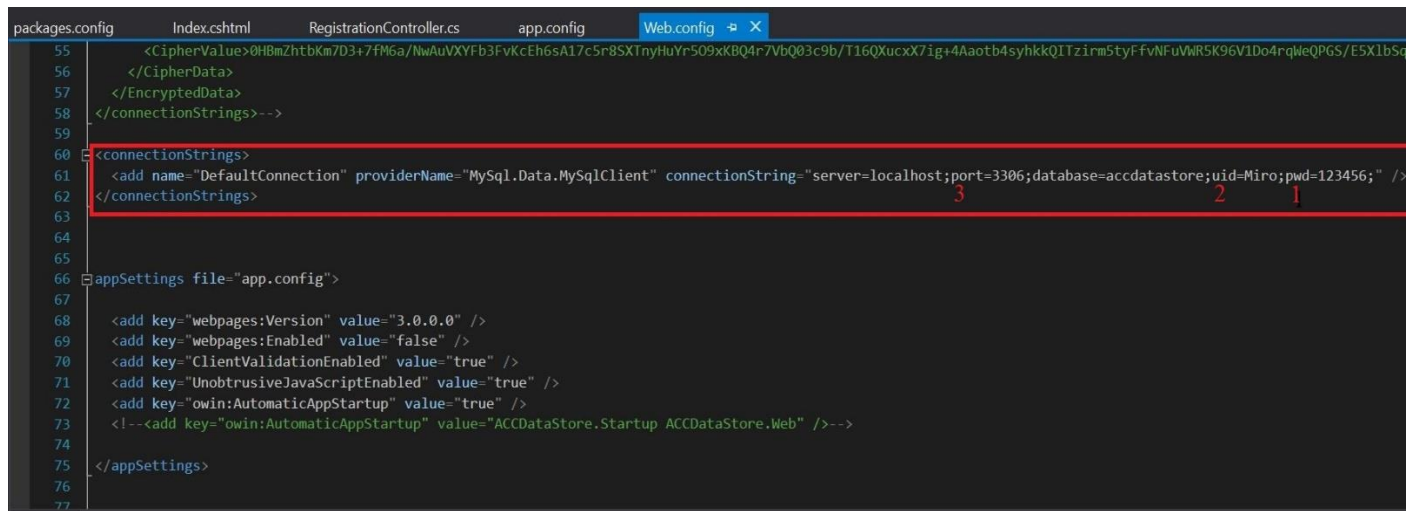


Figure 10 Refresh Schemas



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



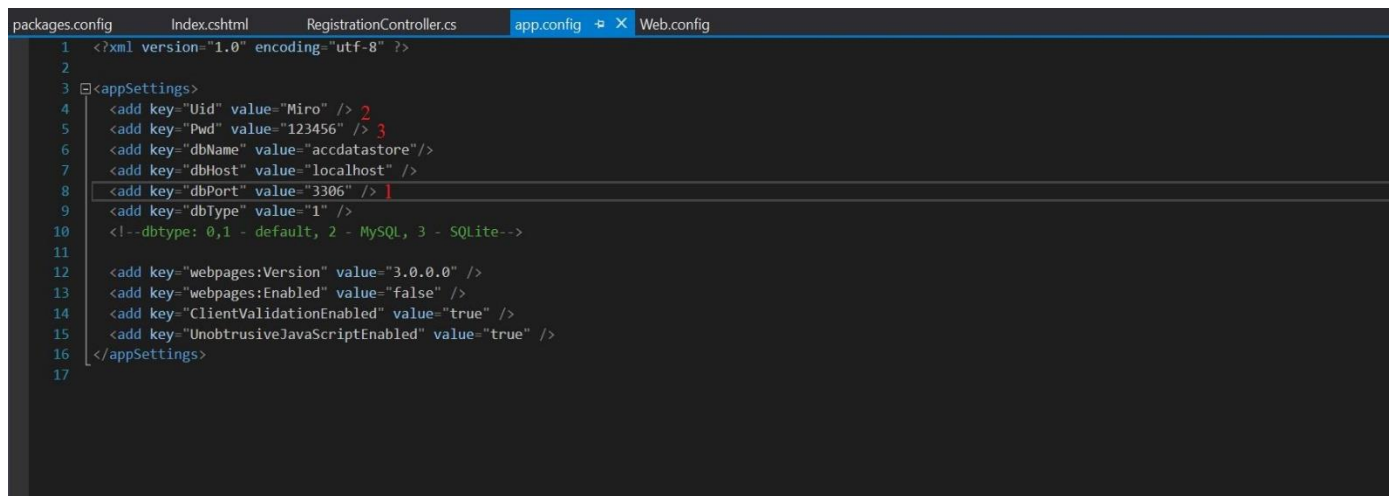
```
55 <CipherValue>0HBmZhtbKm7D3+7fM6a/NwAuVXYfb3FvKcEh6sA17c5r8SXTnyHuYr509xKBQ4r7VbQ03c9b/T16QXucxX7ig+4Aaotb4syhkkQITzirm5tyFfvNFuVwR5K96V1Do4rqWeQPGS/E5X1bS  
56 </CipherData>  
57 </EncryptedData>  
58 </connectionStrings-->  
59  
60 <connectionStrings>  
61 <add name="DefaultConnection" providerName="MySql.Data.MySqlClient" connectionString="server=localhost;port=3306;database=accdatastore;uid=Miro;pwd=123456;" />  
62 </connectionStrings>  
63  
64  
65  
66 <appSettings file="app.config">  
67  
68 <add key="webpages:Version" value="3.0.0.0" />  
69 <add key="webpages:Enabled" value="false" />  
70 <add key="ClientValidationEnabled" value="true" />  
71 <add key="UnobtrusiveJavaScriptEnabled" value="true" />  
72 <add key="owin:AutomaticAppStartup" value="true" />  
73 <!--<add key="owin:AutomaticAppStartup" value="ACCDataStore.Startup ACCDataStore.Web" />-->  
74  
75 </appSettings>  
76  
77
```

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.

The image shows a code editor with several tabs at the top: 'packages.config', 'Index.cshtml', 'RegistrationController.cs', 'app.config' (which is the active tab), and 'Web.config'. The 'app.config' tab contains XML code for application settings. The code is as follows:

```
1 <?xml version="1.0" encoding="utf-8" ?>
2
3 <appSettings>
4   <add key="Uid" value="Miro" /> 2
5   <add key="Pwd" value="123456" /> 3
6   <add key="dbName" value="accdastore"/>
7   <add key="dbHost" value="localhost" />
8   <add key="dbPort" value="3306" /> 1
9   <add key="dbType" value="1" />
10  <!--dbtype: 0,1 - default, 2 - MySQL, 3 - SQLite-->
11
12  <add key="webpages:Version" value="3.0.0.0" />
13  <add key="webpages:Enabled" value="false" />
14  <add key="ClientValidationEnabled" value="true" />
15  <add key="UnobtrusiveJavaScriptEnabled" value="true" />
16 </appSettings>
17
```

Red numbers 1, 2, and 3 are placed next to the 'dbPort', 'Uid', and 'Pwd' values respectively, corresponding to the list in the caption below.

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.

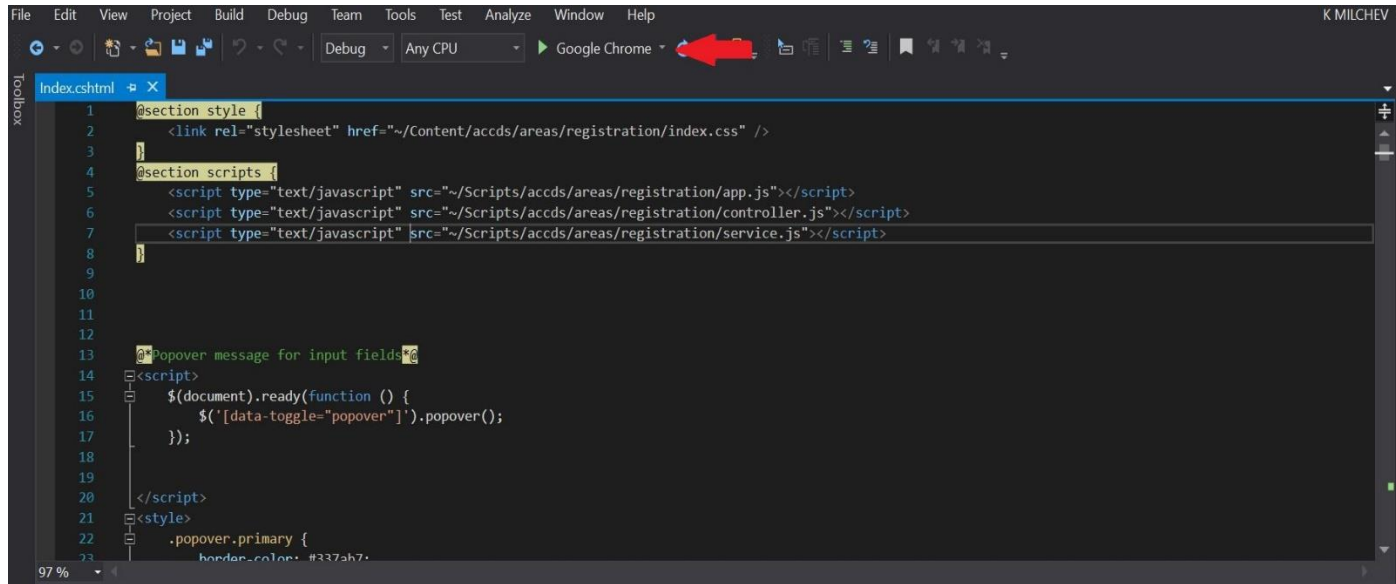


Figure 13 Run Project

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

- email: [admin@gmail.com](mailto: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 Microsoft.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