Table of Contents

[Section 1: Attendance Register Solution 1](#_Toc74919500)

[Projects 1](#_Toc74919501)

[Attendance Register API 1](#_Toc74919502)

[Attendance Register Library 1](#_Toc74919503)

[Attendance Register Data 1](#_Toc74919504)

[Attendance Register UI 1](#_Toc74919505)

[Section 2: Environments 1](#_Toc74919506)

[Section 3: Deployment 1](#_Toc74919507)

[Section 4: Process Flow 1](#_Toc74919508)

[Register 1](#_Toc74919509)

[Login 1](#_Toc74919510)

[Section 5: Running the solution/project 1](#_Toc74919511)

[Section 6 : Summary 1](#_Toc74919512)

# Section 1: Attendance Register Solution

## Projects

### Attendance Register API

#### Function

Rest API exposing methods which is to be used to:

* Register new users/teachers.
* Logging in users/teachers.
* Configuring classes.
  + Add classes.
  + Delete classes.
* Configuring students.
  + Adding student to classes.
  + Removing students from classes.
* Maintaining Attendance Register.
  + Add or update student’s attendance for specific class.
* Attendance Register reporting.
  + Daily report.
  + Termly report.

### Attendance Register Library

#### Function

This class library contains the ORM data models which is used for CRUD operations.

### Attendance Register Data

#### Function

This is a data project used for the design and creation of the SQL database.

### Attendance Register UI

#### Function

The UI consist out of an AngularJs single page application used for:

* Registering new users/teachers.
* Login in registered users.
* Creating new classes.
* Removing existing classed.
* Adding and removing students to and from classes.
* Registering the attendance of students for specified classes.
* Presenting daily and term reports.

# Section 2: Environments

#### Database

SQL Database Server.

#### Backend

.Net Web API 2 to be hosted in IIS.

#### Frontend

AngularJs SPA to be hosted in IIS.

# Section 3: Deployment

#### Steps

1. Creating Database
   * SQL script location: AttendanceRegister\SQLScripts.
   * Execute “CreateDbScript” script in SQL Server Management Studio.
2. Deploying Web API
   * Published API location: AttendanceRegister\Deployment\API
   * Copy API folder to: C:\inetpub\wwwroot on webserver and rename to “AttendanceRegisterAPI”.
   * In IIS, add new ApplicationPool by right clicking and selecting “Add Application Pool”.

Graphical user interface, text

Description automatically generated

* + In IIS, right click on “API” folder and select “Convert to Application”

Graphical user interface

Description automatically generated

* + Select the newly created Application pool

Graphical user interface, text, application

Description automatically generated

* + Ensure that your connection string in the API’s “web.config” file is pointing to the database you created in step 1.

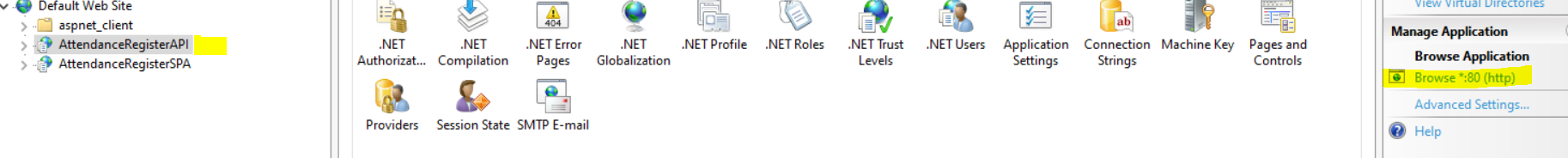
Graphical user interface, table

Description automatically generated

Text

Description automatically generated

* + Browse the API to ensure its working:



Graphical user interface, text

Description automatically generated

1. Deploying the UI
   * Published UI location: AttendanceRegister\Deployment\SPA
   * Copy SPA folder to: C:\inetpub\wwwroot on webserver and rename to “AttendanceRegisterUI”.
   * In IIS, right click on “SPA” folder and select “Convert to Application”

Graphical user interface, text, application

Description automatically generated

* + Select the newly created Application pool

Graphical user interface, text, application

Description automatically generated

* + Browse to test UI

Graphical user interface, application

Description automatically generated

Text

Description automatically generated

* + Correct URL:

From <http://someserver/AttendanceRegisterUI#!/AR>

To <http://someserver/AttendanceRegisterUI/#!/AR>

Graphical user interface, text, website

Description automatically generated

# Section 4: Process Flow

### Register

Graphical user interface

Description automatically generated

### Login

Graphical user interface, application

Description automatically generated

# Section 5: Running the solution/project

#### Steps:

1. Creating Database
   1. SQL script location: AttendanceRegister\SQLScripts.
   2. Execute “CreateDbScript” script in SQL Server Management Studio.
2. Run Visual Studio as **Administrator**.
3. Open the solution file from the folder: AttendanceRegister/.
4. Because the API and SPA does not exists in your local IIS the projects might not load correctly.
   1. Foreach project that doesn’t load correctly open the “Someproject.csproj” file from: \AttendanceRegister\AttendanceRegisterAPI\AttendanceRegisterAPI.csproj

Table

Description automatically generated

* 1. Do a search for “IIS” in the file:

Graphical user interface, text, application

Description automatically generated

* 1. Set the entry to “False” and save the file.

Text, letter

Description automatically generated

* 1. The project should now load correctly.
  2. Right click on the project in Visual Studio and select properties.

Graphical user interface, text, application, email

Description automatically generated

* 1. On the Web tab, select Local IIS and click “Create Virtual Directory”, this will create the project in your Local IIS.
  2. Rebuild the solution.

1. Setup start-up projects.
   1. Right click on the solution and select “Set StartUp Projects”.
   2. Select “Multiple startup projects”
   3. Under “Action”, select “Start” for both the API and UI projects:
   4. Graphical user interface

      Description automatically generated
   5. Click Apply and Ok.
   6. Confirm that your connection string point to the correct Database.
   7. You should now be able to run the solution locally.

# Section 6 : Summary

Using a Web API allows for multiple types of UI applications i.e., mobile apps, web apps, WPF apps Windows Forms apps.

Seeing that Mrs. Masuku is locked down and she is unable to install software, a web application seems the logical option.

AngularJs utilizing Bootstrap’s adaptive layout will allow the web pages to adapt the screen size of the device they’re browsed from and show work relatively well on an iPhone.