Skill Tracker

Instructions.

Database Objects Creation:

Prerequisites: Microsoft SQL Server Management Studio 2008 and above installation and Windows 7 and above machine.

Step 1: Unzip or extract DB.zip file to local machine follow instructions in db\_Instructions.txt file.

Step 2: Open installed version Microsoft SQL Server Management Studio then open SkillTracker\_1 SQL file from extracted location and execute it.

Step 3: Verify WorkoutTracker database created successfully.

Step 4: Open SkillTracker\_DDL\_2 SQL file from extracted location and execute it.

Step 5: Verify user, database, workout\_category, workout\_collection, workout\_active tables are created in WorkoutTracker database successfully.

Step 6: Open SkillTracker\_DML\_3 SQL file from extracted location and execute it.

Step 7: Verify data is inserted in user table successfully.

Configurations to run front-end application:

Prerequisites: Microsoft Visual Studio 2017 and above installation and Windows 7 and above machine, Node.js version 9.7.1, Node Package Manager(npm) version 5.8.0, Angular CLI version 1.7.0, Typescript version 2.5.3 installations.

Step 1: Unzip or extract SkillTrackerService.zip file to local machine.

Step 2: Find SkillTrackerService from extracted location and open it in visual studio 2017 or higher.

Step 3: Rebuild the solution to nuget package manger to install necessary packages to application run.

Step 4: Verify application builds successfully without error.

Step 5: Verify application correctly pointed to right database in Web.config connection string part or follow below instructions.

Step 6: Verify application correctly pointed to right folder in Web.config or App.config app settings part in all projects or follow below instructions.

Step 7: Open App.config file in any editor for the below mentioned projects

1. SkillTracker.Repository.Tests.
2. SkillTracker.Services.Tests.
3. SkillTracker.WebApi.Tests.

And find “SkillTrackerEntities” configuration in <connectionStrings></connectionStrings> part and replace below highlighted part

Example connection string.

<add name="SkillTrackerEntities" connectionString="metadata=res://\*/ SkillTrackerModel.csdl|res://\*/ SkillTrackerModel.ssdl|res://\*/ SkillTrackerModel.msl;provider=System.Data.SqlClient;provider connection string=&quot;data source={PlaceServerName};initial catalog=SkillTracker;integrated security=True;MultipleActiveResultSets=True;App=EntityFramework&quot;" providerName="System.Data.EntityClient" />

With current SQL Server Management Studio name where SkillTracker database created in each projects one by one.

Step 8: Open web.config file in any editor for the below project

1. SkillTracker.WebApi.

And find “SkillTrackerEntities” configuration in <connectionStrings></connectionStrings> part and replace below highlighted part

Example connection string.

<add name="SkillTrackerEntities" connectionString="metadata=res://\*/ SkillTrackerModel.csdl|res://\*/ SkillTrackerModel.ssdl|res://\*/ SkillTrackerModel.msl;provider=System.Data.SqlClient;provider connection string=&quot;data source={PlaceServerName};initial catalog=SkillTracker;integrated security=True;MultipleActiveResultSets=True;App=EntityFramework&quot;" providerName="System.Data.EntityClient" />

With current SQL Server Management Studio name where SkillTracker database created.

Step 9: Open web.config file in any editor for the below project.

1. SkillTracker.WebApi.

<appSettings>

<add key="ErrorLogFileName" value=" SkillTrackerLog.txt" />

<add key="ErrorLogFilePath" value="C:\Final\_SBA\SkillTracker\SkillTrackerLogs\" />

<add key="Audience" value="http://localhost:55804/" />

<add key="Issuer" value="http://localhost:50499/" />

<add key="JWTSecurityKey" value="401b09eab3c013d4ca54922bb802bec8fd5318192b0a75f201d8b3727429090fb337591abd3e44453b954555b7a0812e1081c39b740293f765eae731f5a65ed1" />

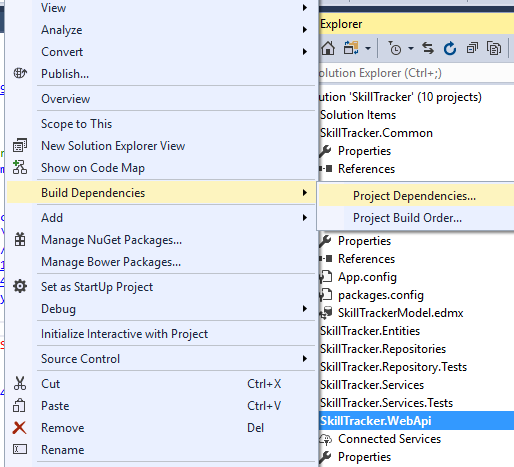
<add key="TokenExpiryInMinutes" value="30" />

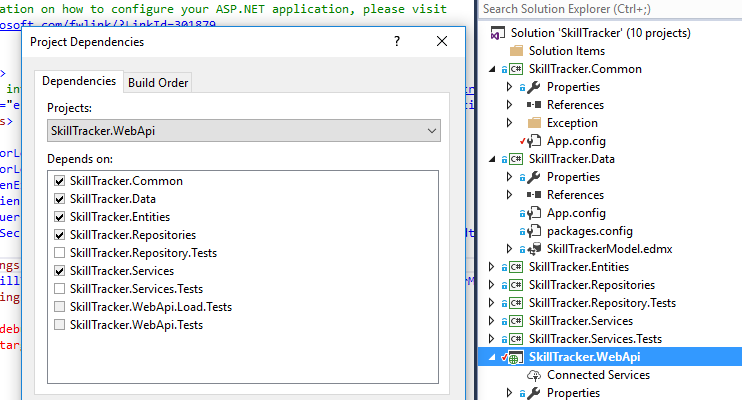
</appSettings>

And change the value of highlighted parts according to the solution.

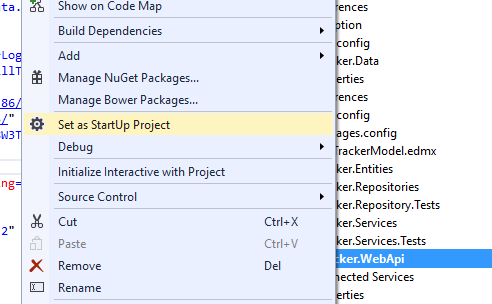
Note: Audience denotes SkillTracker.Web project url, and Issuer denotes SkillTracker.WebApi project url which will get it from project properties. And also change the ErrorLogFilePath according to the solution path.

Step 10: Then Right click on SkillTracker.Web project then select Build Dependencies -> Project Dependencies it will show list of projects in the solution, then check SkillTracker.WebApi project and give ok.





Step 11: Then Right click on SkillTracker.Web select Set as Startup project.



Step 15: Hit F5 to run the service.

Step 16: Unzip or extract SkillTrackerUI.zip file to local machine.

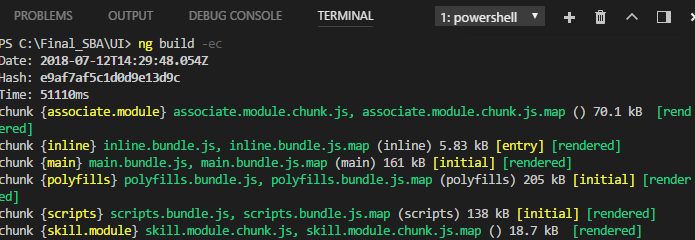
Step 17: Find SkillTracker from extracted location and open it in visual studio code.

Step 18: Open Integrated Terminal in visual studio code and command node --version, npm --version, ng --version to verify Node.js, NPM, Angular CLI installations.

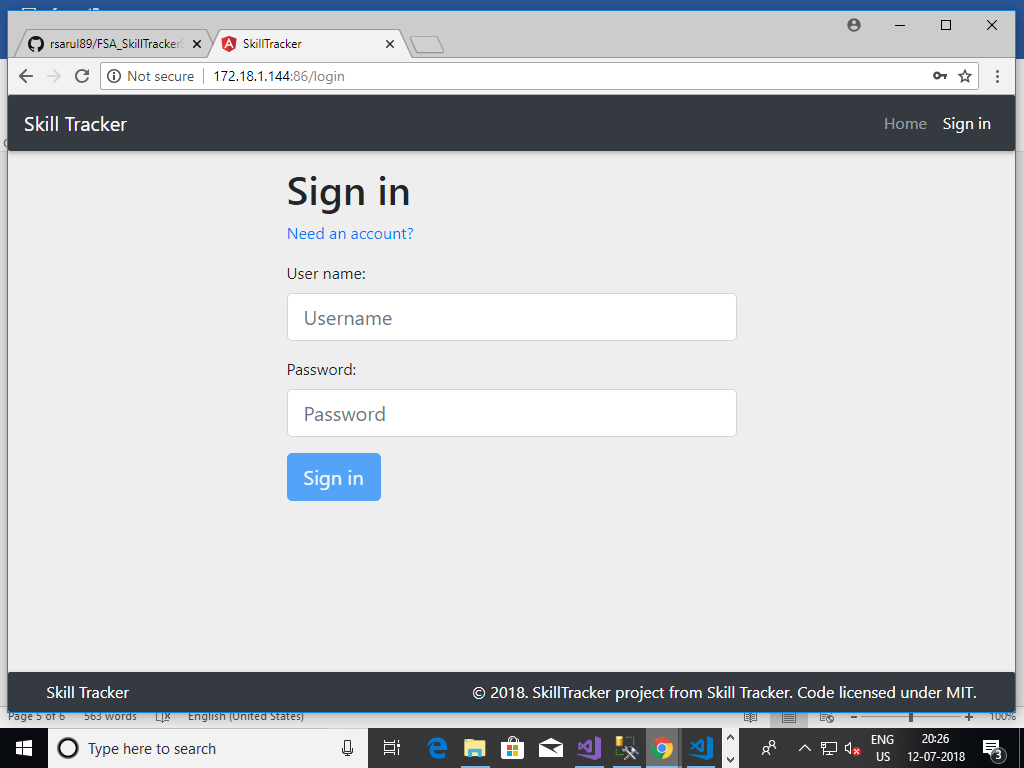
Step 19: Open Integrated Terminal in visual studio code and command npm install, it will install all node modules required and wait till installation gets complete.

Step 20: Verify all node modules created under node\_modules in above mentioned path.

Step 21: Then provide ng build --ec or ng build --prod command in command prompt it will build the application then Rebuild the entire solution again and verify solution gets build without error. Example successful build screenshot.



Step 22: Then click F5 to run the application and application opens in browser or open <http://localhost:4200> any browser and default page is user login page .



Step 23: Demo user credentials to login

User name: demouser

Password: test123

Or user can also register themselves by navigate Register page by clicking Register button.