Table of Contents

[Pre-requisites 2](#_Toc510392362)

[Installation guide 2](#_Toc510392363)

[Seed the database. 3](#_Toc510392364)

[Build the back-end solution 4](#_Toc510392365)

[Configuring the Application 5](#_Toc510392366)

[Changing the ConnectionString 6](#_Toc510392367)

[Changing the PageSize 6](#_Toc510392368)

[Changing the search suggestion count 6](#_Toc510392369)

[\*Task 1 Back-end – Create a Web Service 7](#_Toc510392370)

[WebApi documentation via swagger 7](#_Toc510392371)

[\*Task 2 Front-end– Paging 7](#_Toc510392372)

[Angular site 7](#_Toc510392373)

[Mobile View 7](#_Toc510392374)

[\*Task 3 Front-end – Header Sorting 8](#_Toc510392375)

[Solution Architecture Overview 9](#_Toc510392376)

[Hosts 9](#_Toc510392377)

[EcomEngineTechChallenge.ApiHost 9](#_Toc510392378)

[EcomEngineTechChallenge.ConsoleSeeder 9](#_Toc510392379)

[EcomEngineTechChallenge.Spa 10](#_Toc510392380)

[Data 10](#_Toc510392381)

[EcomEngineTechChallenge.Data 10](#_Toc510392382)

[Business 11](#_Toc510392383)

[EcomEngineTechChallenge.Business.Common 11](#_Toc510392384)

[Infrastructure 11](#_Toc510392385)

[EcomEngineTechChallenge.Contracts 11](#_Toc510392386)

[Tests 12](#_Toc510392387)

[EcomEngineTechChallenge.Tests.Integration 12](#_Toc510392388)

[EcomEngineTechChallenge.Tests.Unit 12](#_Toc510392389)

# Pre-requisites

* Visual Studio 2017
* .Net core 2.0
* NodeJS (https://nodejs.org/en/download/current/)

# Installation guide

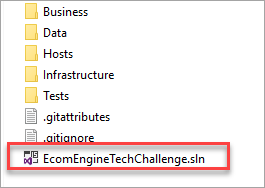
1. Download the source codes

<https://drive.google.com/open?id=1UXrMhEhob4JSZFT2ZcKOoZvzFtYZWi5o>

1. Extract EcomEngineTechChallenge.zipx

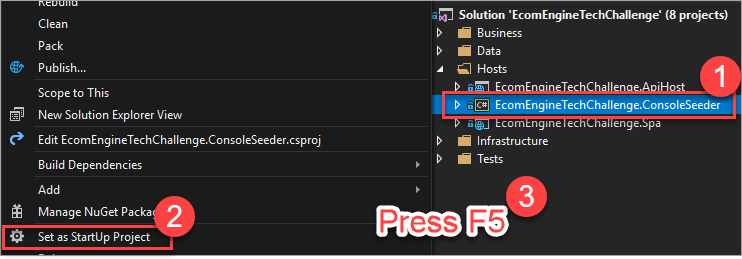


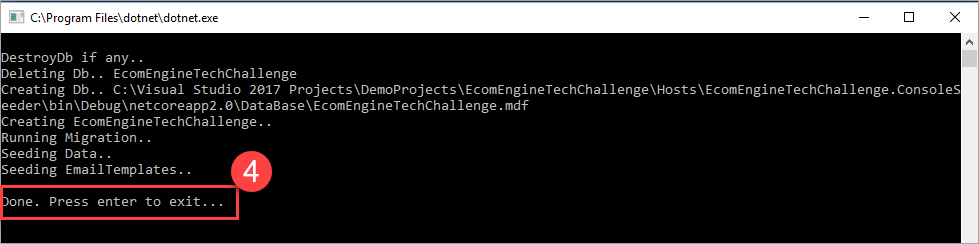
1. Open the solution



## Seed the database.

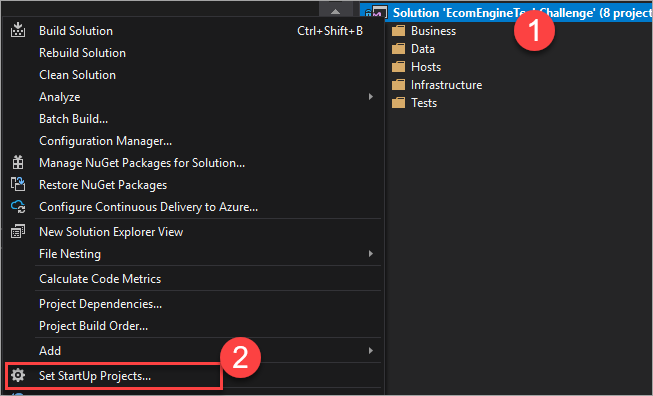
Run the EcomEngineTechChallenge.ConsoleSeeder project.

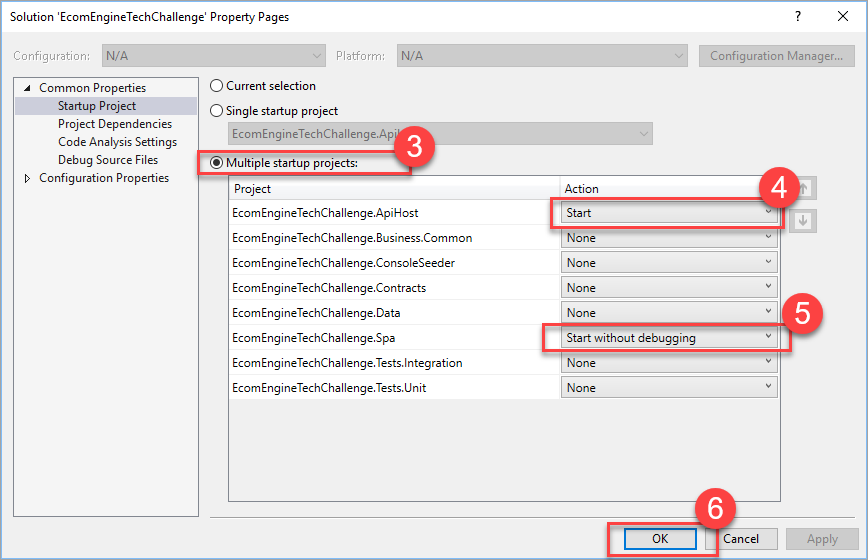




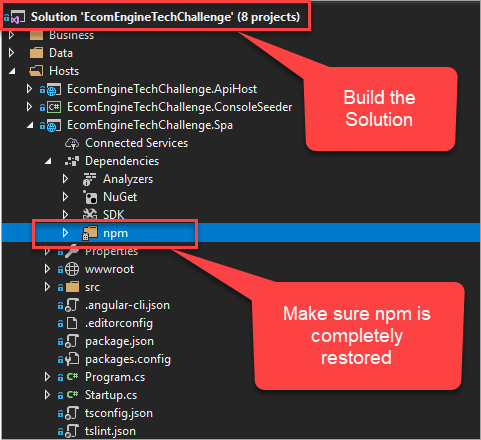
## Build the back-end solution

1. Set start-up Projects





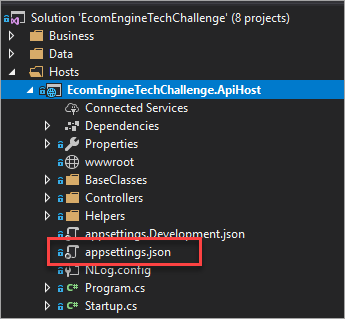
1. Build the solution



1. Run the solution by pressing F5

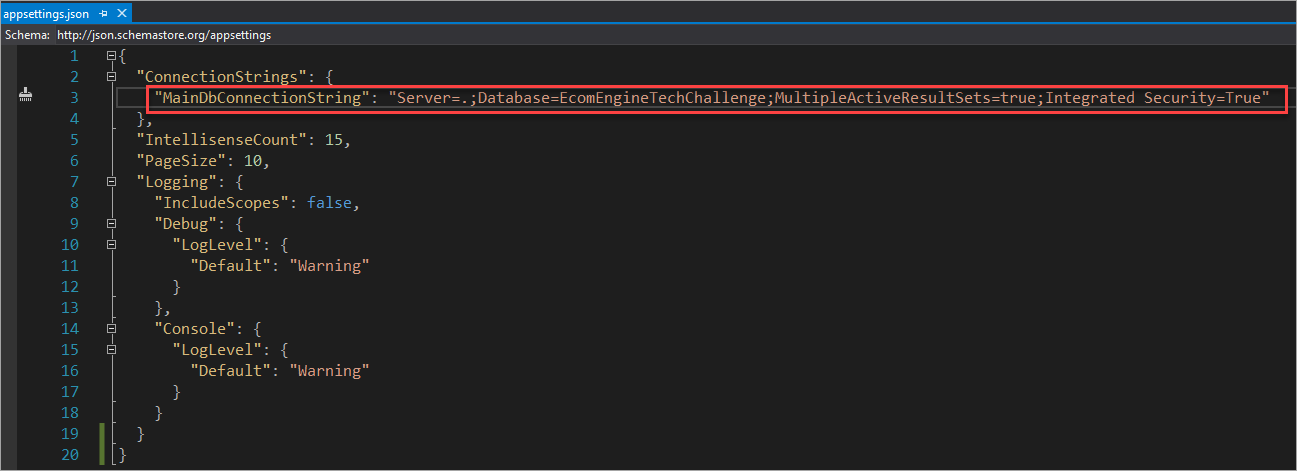
# Configuring the Application

Open *appsettings.json*



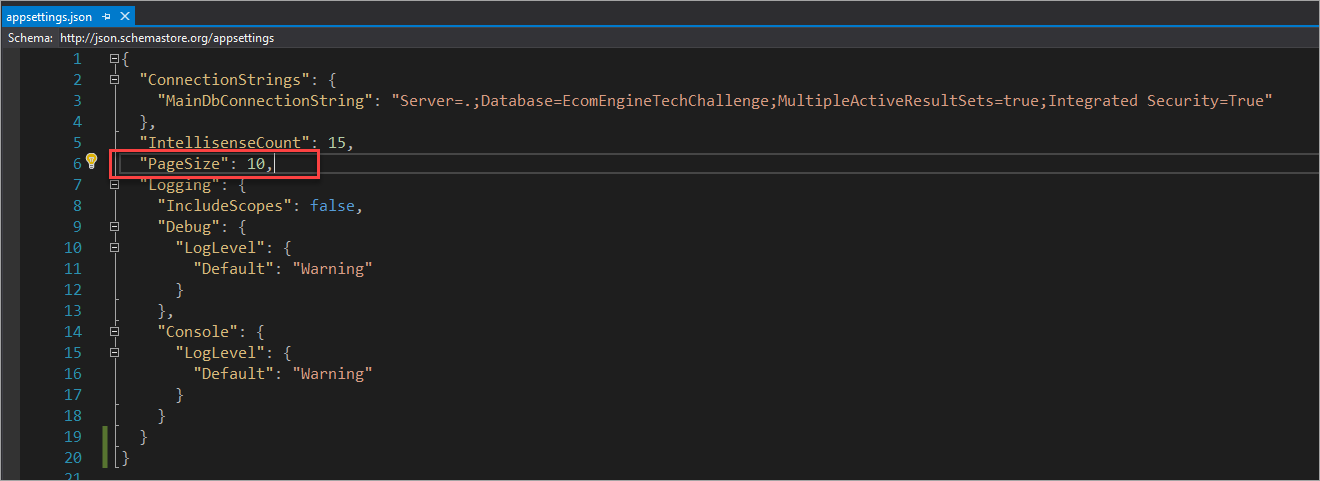
## Changing the ConnectionString

Modify the entry to change the database name, etc..



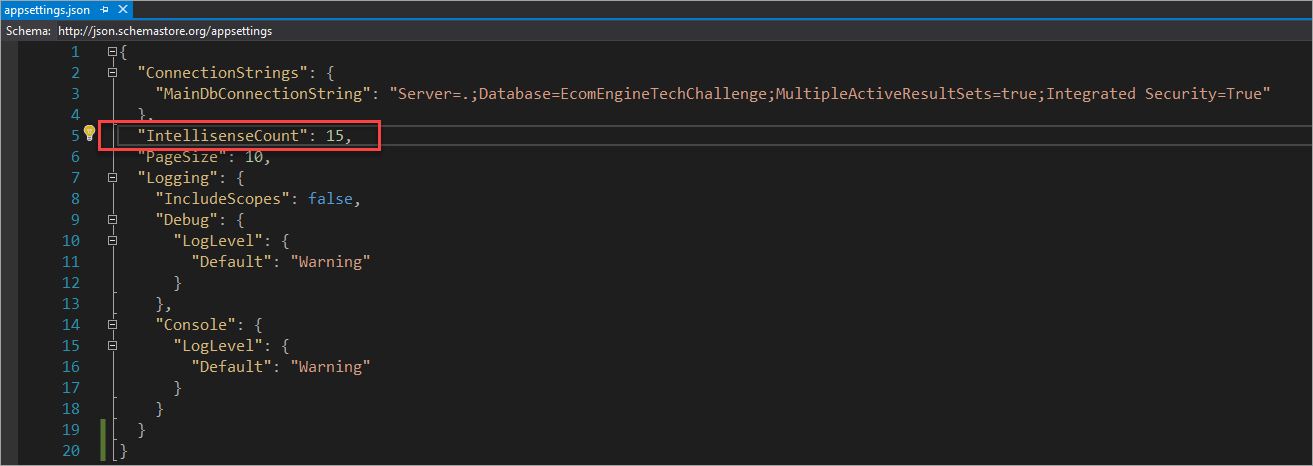
## Changing the PageSize

Increase or decrease values here. Default is 10 rows.



## Changing the search suggestion count

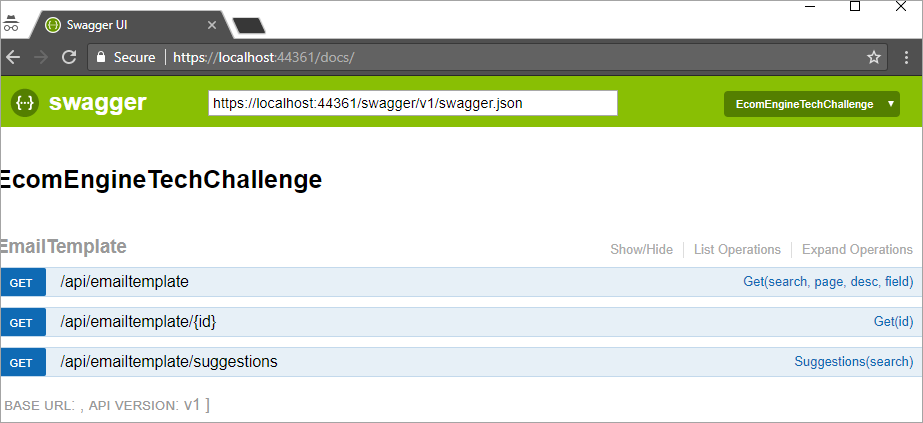
Increase or decrease values here. Default is 15 suggestions..



# \*Task 1 Back-end – Create a Web Service

## WebApi documentation via swagger

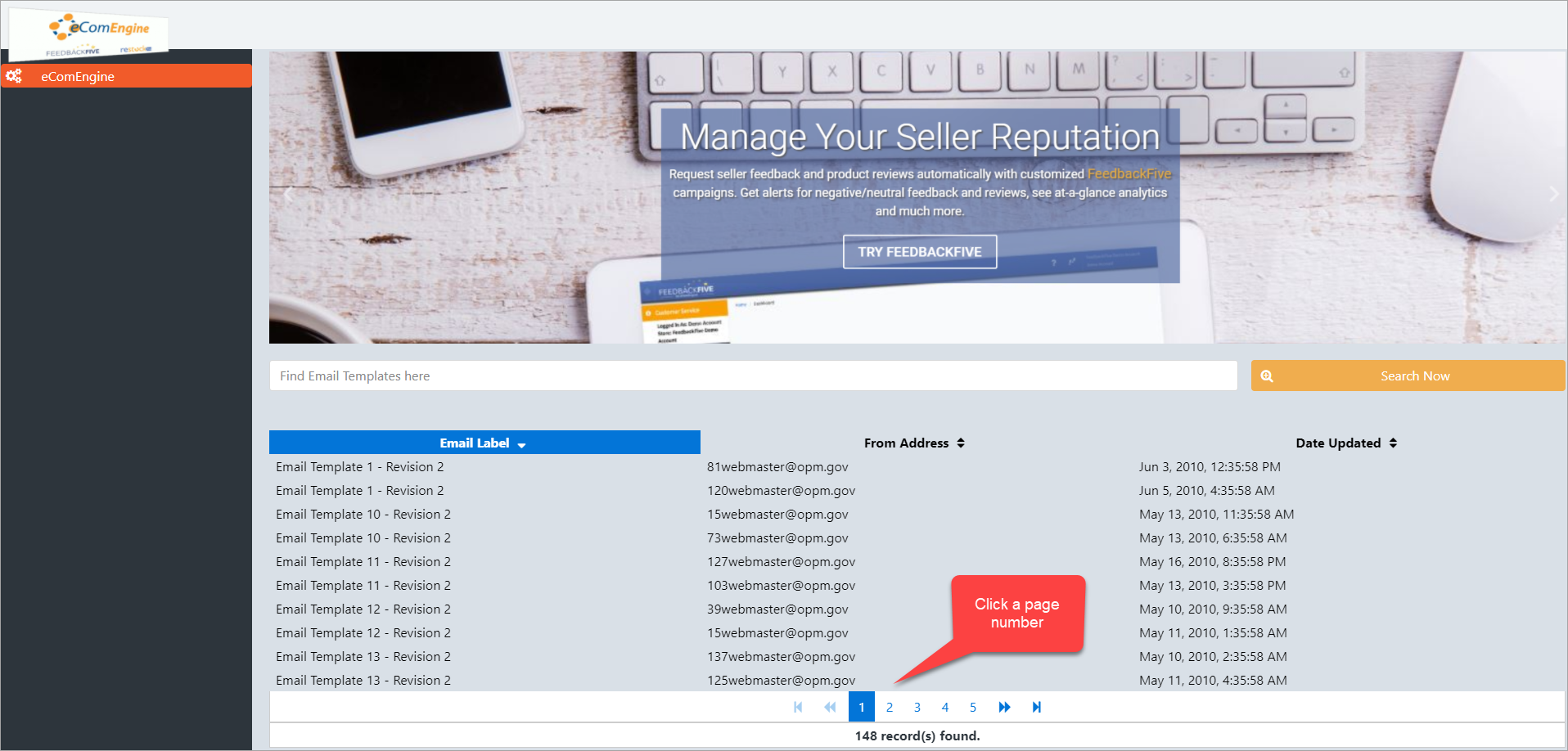
An optional component that will allow you to play around with the eComEngine endpoints.



# \*Task 2 Front-end– Paging

## Angular site

Click on the page indicators to navigate from one page to another.

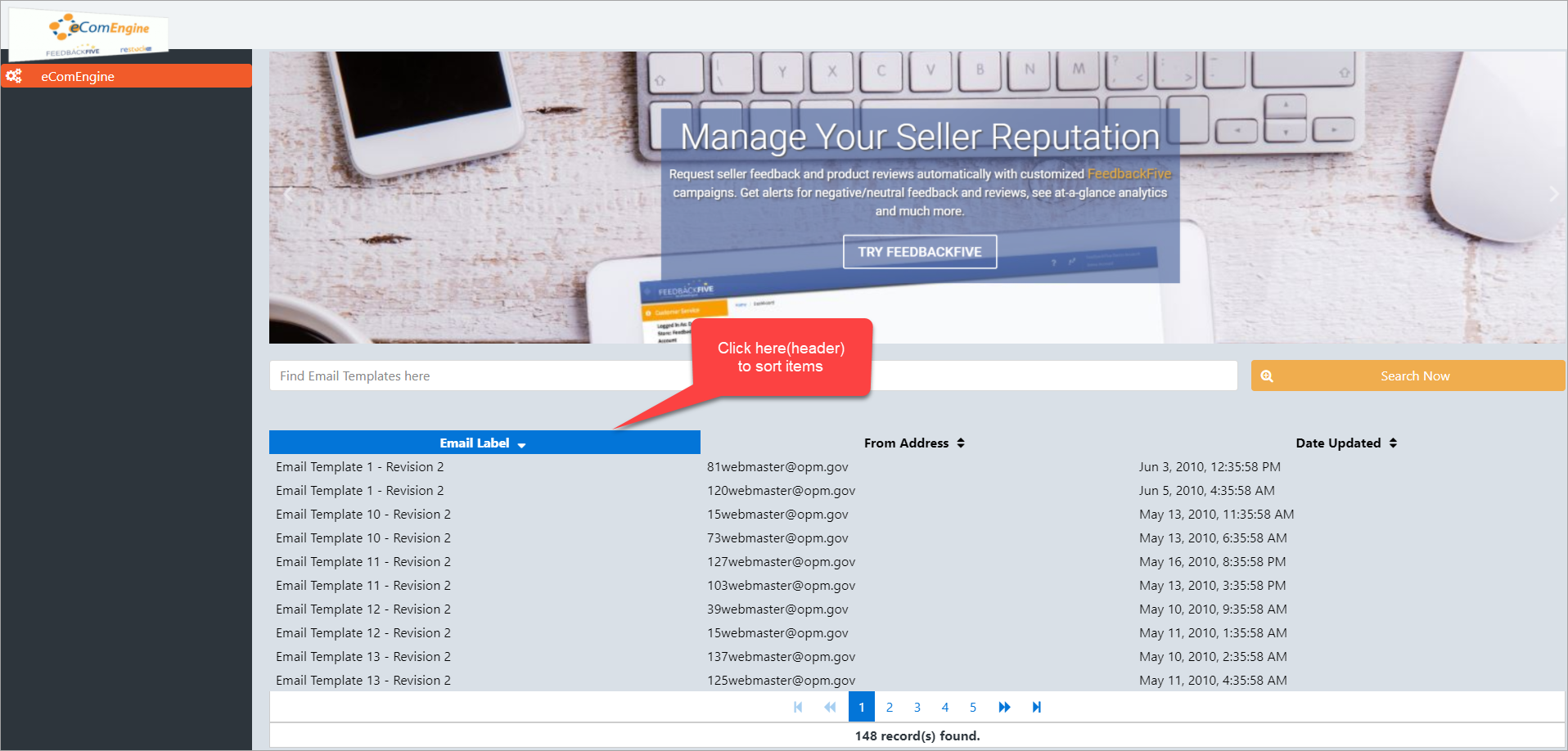


## Mobile View

This sample page is designed for mobile browsing as well.

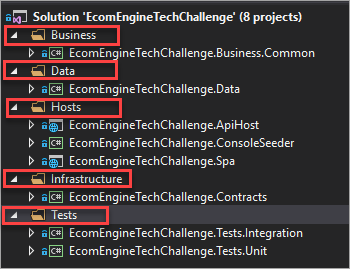
# \*Task 3 Front-end – Header Sorting

Click on the headers to toggle sorting between ascending and descending.



# Solution Architecture Overview

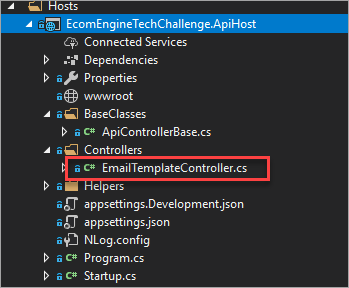
In bird’s eye view, the solution is separated into 5 different areas namely: **Business,** **Data**, **Hosts, Infrastructure** and **Tests.** This solution contains a little bit of everything of SOA.



## Hosts

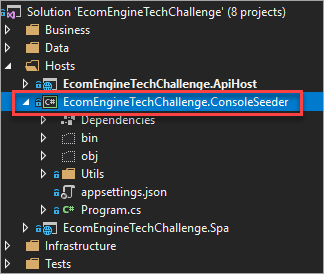
### EcomEngineTechChallenge.ApiHost

* This is the main project. It hosts the WebApi.
* Uses Dependency Injection via MEF.



### EcomEngineTechChallenge.ConsoleSeeder

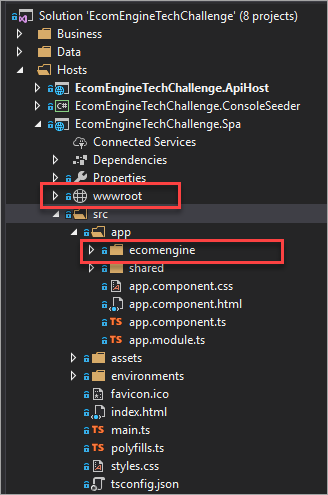
* Used to create the database and populate it with sample data.



### EcomEngineTechChallenge.Spa

Contains the Angular5 app.

* wwwroot – contains precompiled javascripts from my machine – ready to use.
* ecomengine – this is where the front-end codes are located.

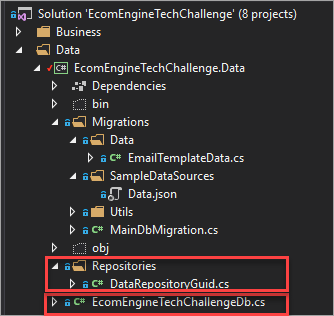


## Data

Uses EntityFramework Core 2.0

### EcomEngineTechChallenge.Data

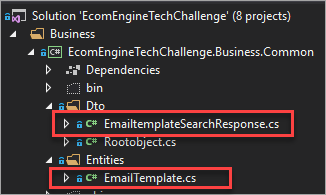
* This is where the EF Datacontext is found.
* DataRepositoryGuid.cs is a general purpose class for retrieving data – highly testable.



## Business

### EcomEngineTechChallenge.Business.Common

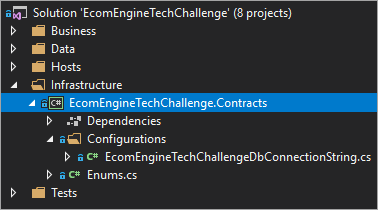
* EmailtemplateSearchResponse.cs - contains other properties needed by the Angular 5 client.
* EmailTemplate.cs – the entity representing the sample data.



## Infrastructure

### EcomEngineTechChallenge.Contracts

* A separate place for enums, contracts, configuration, etc..

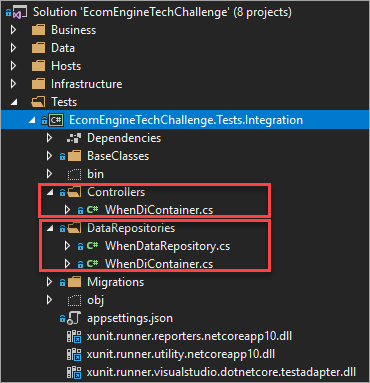


## Tests

Uses xUnit, Nsubstitute and Shouldly.

### EcomEngineTechChallenge.Tests.Integration

* Contains automated integration tests.



### EcomEngineTechChallenge.Tests.Unit

* Contains automated unit tests.

