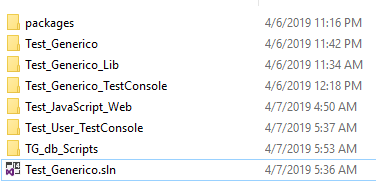
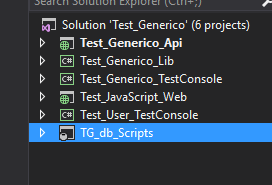
Project Description:

The main goal of this project is to test my abilities in .net and javascript. So I create the following solution using vs2015 framework 4.5.2



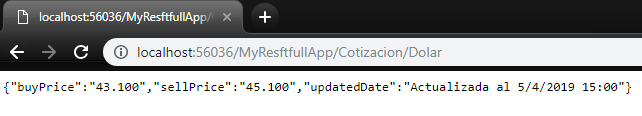


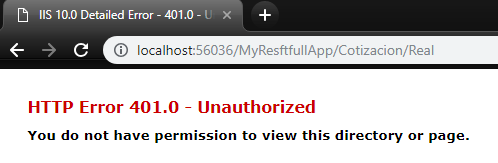
The main solution is **Test\_Generico**

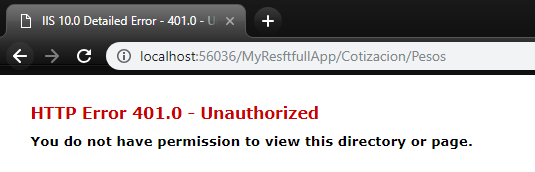
**Test\_Generico\_Api**

Is the rest api solution for the 2 services.

The first service MyResftfullAPP have a Cotizacion method that return a Json object if is a dolar quote price. If is Real and Pesos it will give the 401 error.







Se second service is for the User.

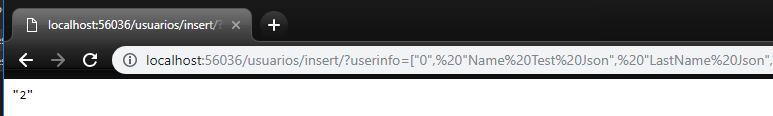
**Insert method:** will receive a json object with the user information and return the userid

jsonObject:

["0", "Name Test Json", "LastName Json", "email@email.com", "password"]

Example of the call:

/usuarios/insert/?userinfo=["0", "Name Test Json", "LastName Json", "email@email.com", "password"]

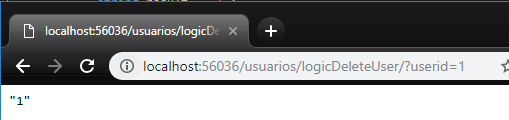


**logicDeleteUser method:** will receive userid and will do a logical delete

userid:1

Example of the call:

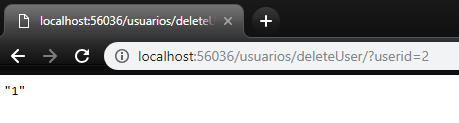
/usuarios/logicDeleteUser/?userid=1



deleteUser method: permanent delete of the user will receive an userid

userid:2

Example of the call:

/usuarios/deleteUser/?userid=2

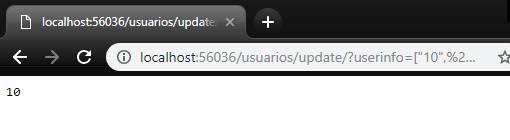
Update method: Will update the user information and return the iduser. This method receive an json object with the user information. The first parameter in the json is the iduser to be updated.

jsonObject:

["1", "Name Test Json", "LastName Json", "email@email.com", "password"]

Example of the call:

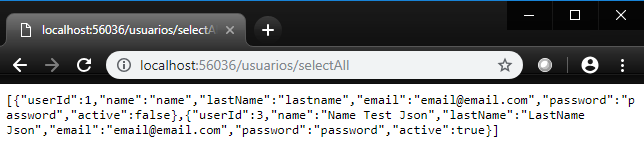
/usuarios/update/?userinfo=["10", "Name Test Json", "LastName Json", "email@email.com", "password"]



selectAll method: this method will return all the users that are in the database as a json.

Example of the call:

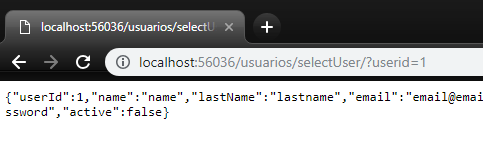
/usuarios/selectAll



selectUser method: will return the information of a specific user as json and receive and iduser

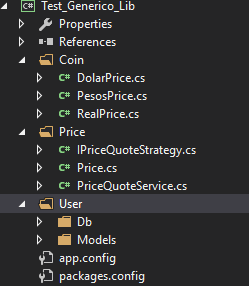
Example of the call:

/usuarios/selectUser/?userid=1



**Test\_Generico\_Lib**

This is a class library for all the strategy pattern and the calls to database. This lib can be use in other projects. For this project I use this lib in the console application app for the unit test and the rest api.



In the Price folder I define a service and an interface to use the strategy pattern and in the coin folder are defined the coins (Dolar,Peso,Real) and all the method that need to be done.

The User folder is where all the methods of the user are implemented. We have a model user and for data access I used entity framework. All the data access method are in the SqlDataEF class.

In the app.config we can change the following values:

ConnectionString:

We can change the connection information.

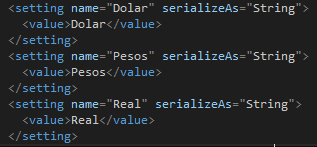


DolarUrl:

We can change the url to consume to get the dollar price.

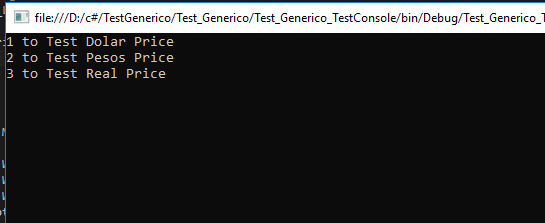


We can change the name of the parameter in the rest api, so we only need to change the value.

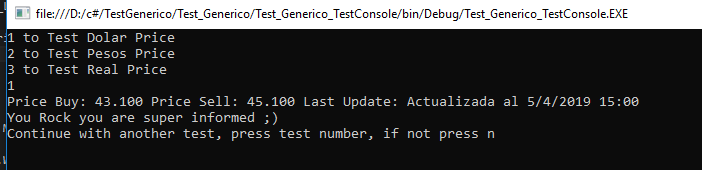


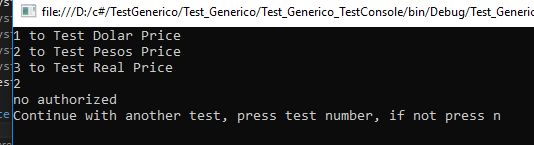
**Test\_Generico\_TestConsole**

This console app project to test the internal lib for the coin prices. Press 1 for dolar test, 2 for Pesos test and 3 for Real test.2



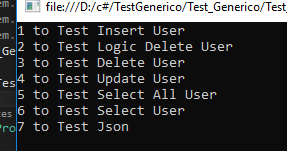
If you run the console app it will show you the following screen and you can internally test the prices methods.

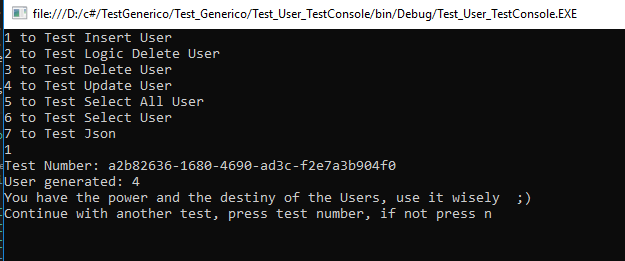


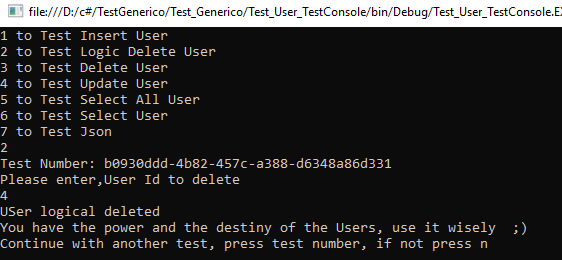


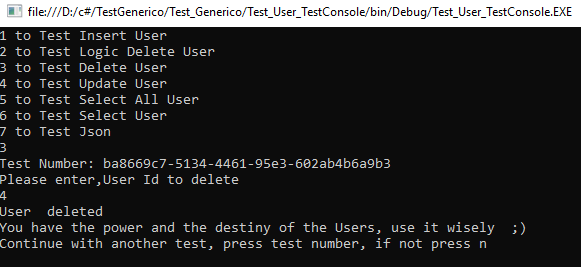
**Test\_User\_TestConsole**

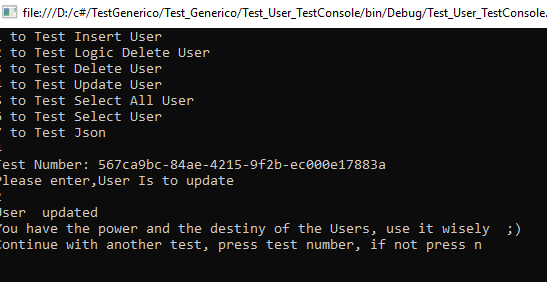
This console app project to test all the User methods. It will generate a guid number for the test and for example is you test insert it will insert the user information with the guid so you can track the test.

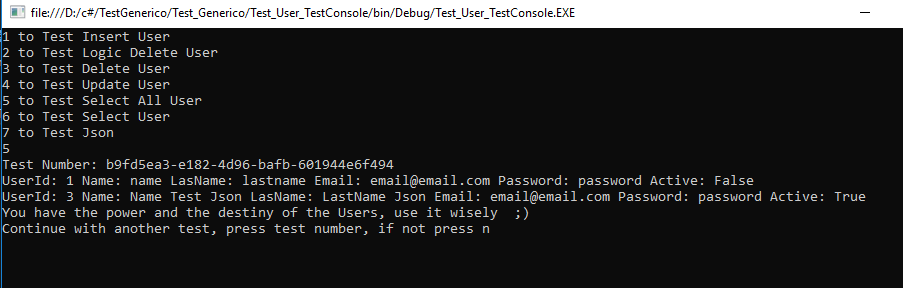


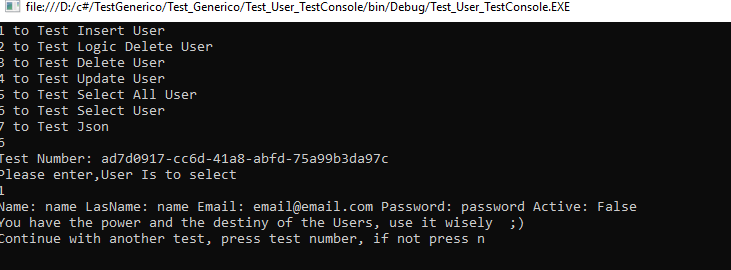








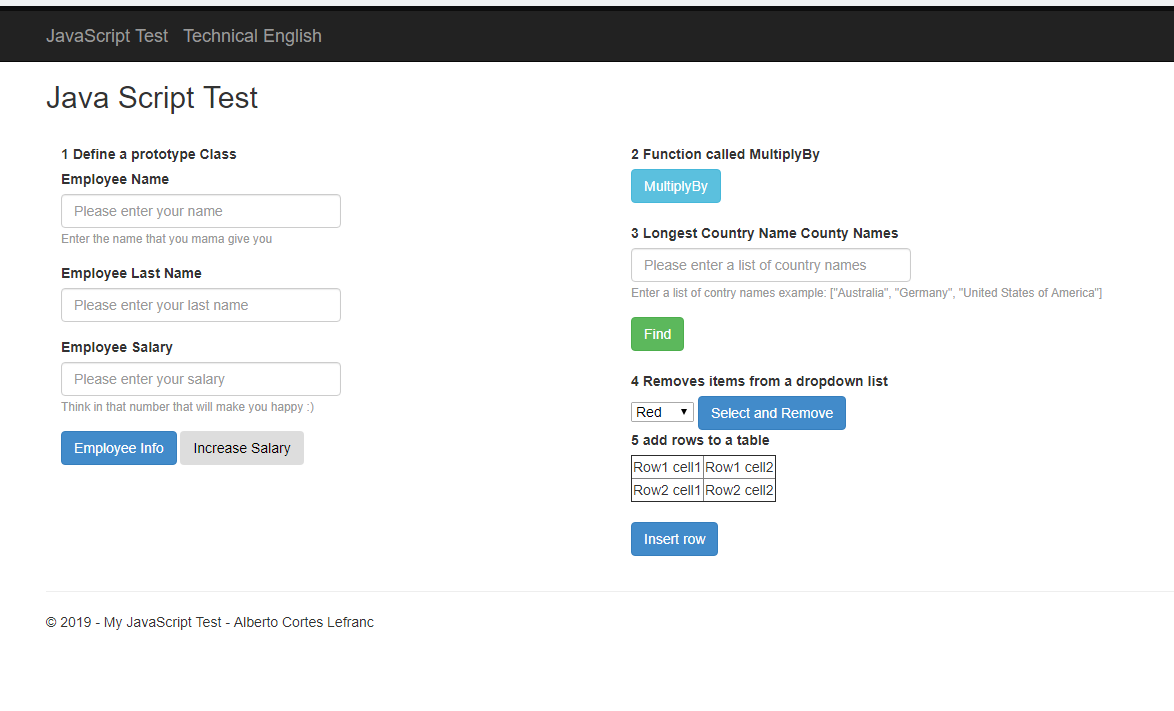




**Test\_JavaScript\_Web**

This is a project for the Javascript part has two links the JavaScript Test that is all the javascript part the Technical English is a page where I answered all the question that you send me.

This is configured to use a default route and is you run it will go this page.



In the Folder JsScripts is where the Employee prototype class is.



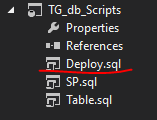
All the other javascript tests are in the view folder JavaScriptTest



**TG\_db\_Scripts**

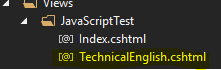
This is the db project will all the SP and table scripts that I used.

For the db deploy you can use the Deploy script in the database that you are going to use for deploy. This script create the table, sp and insert one user in the table, you can insert more using the Test\_User\_TestConsole.



**Technical Englis part**

This part is in the Test\_JavaScript\_Web project on the following view:



URL: <http://localhost:60470/JavaScriptTest/TechnicalEnglish>

