

Work Summary

Table of Contents

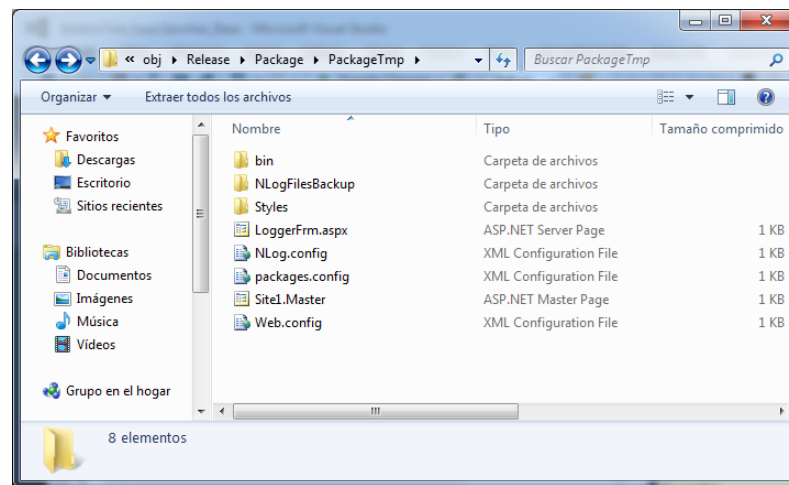
Installers.....	2
Source Code	2
BelatrixTest_IsaacSanchez_Base Application	2
UI Layer: BelatrixTest.UI.....	3
Note for BelatrixTest.UI configuration:.....	3
Business Layer: BelatrixTest.Business.....	3
Business Layer: BelatrixTest.Files	4
Access Layer: BelatrixTest.Access.Common	4
Access Layer: BelatrixTest.Access.Core	4
Production Environment Setup	5
Changing logger plugins.....	6
Changing class libraries plugins.....	6
Extend the BelatrixTest_IsaacSanchez_Base solution for new UI interfaces	6
How to use	6
Database Connection Message.....	6
Logger to a File in Disk	7
Logger to Database	7
Logger to Console	8

Installers

This is a development installer for helping you setup development local tests, which means you cannot use directly on production ATMs. But you can configure an IIS or others application windows servers in order to complete the future deployment later.

1. BelatrixTest_IsaacSanchez_Base package: Compiled UI classes and pages ready to be published as website.

It includes:

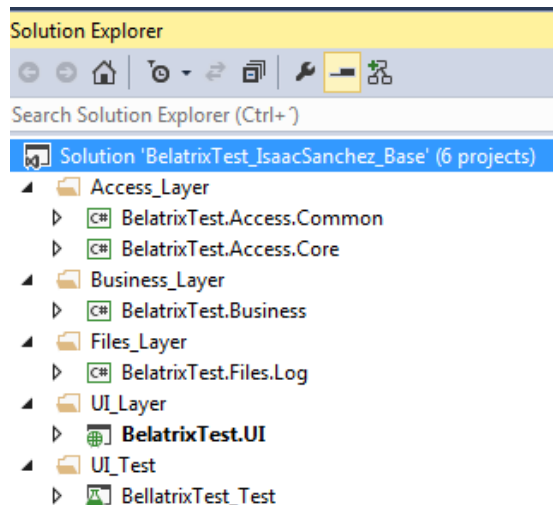


Source Code

The source code was written in C# as the main language, the architecture allows the layer's construction to make it safe and scalable applying the best practices and patterns.

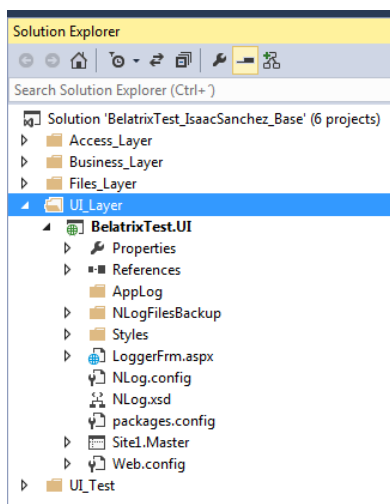
BelatrixTest_IsaacSanchez_Base Application

Please take a look into the description and main purposes for each application layer:



UI Layer: BelatrixTest.UI

Web Forms, master pages, scripts, styles and configuration are located in the UI layer; it means you can implement the UI in the way the user prefers in terms of color, layouts and user experience in general.

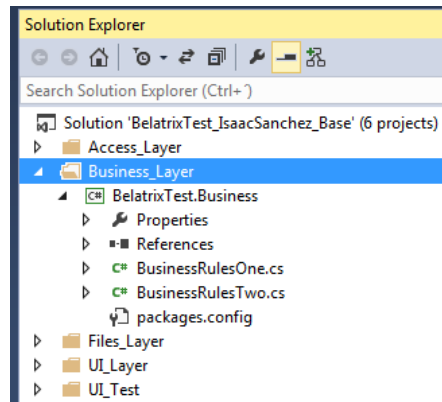


Note for BelatrixTest.UI configuration:

There are parameters you may change in .xml file, please check it before running the web application.

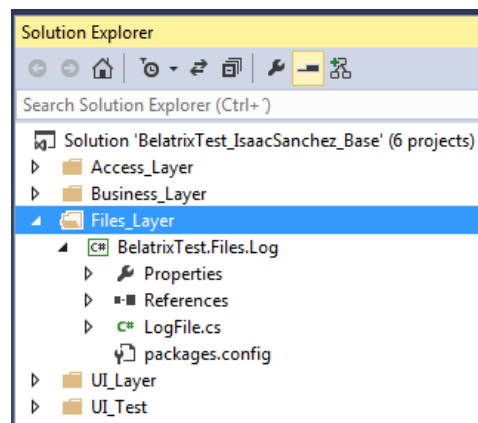
Business Layer: BelatrixTest.Business

Business rules, restrictions, entities and main logic are located into this layer.



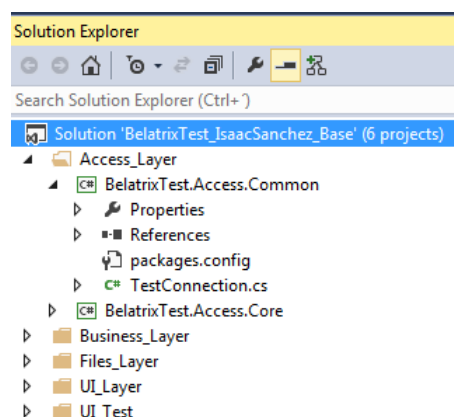
Business Layer: BelatrixTest.Files

To implement some non-functional features that work cross over the other layers or applications, it could have cryptography, customized logs, and others.



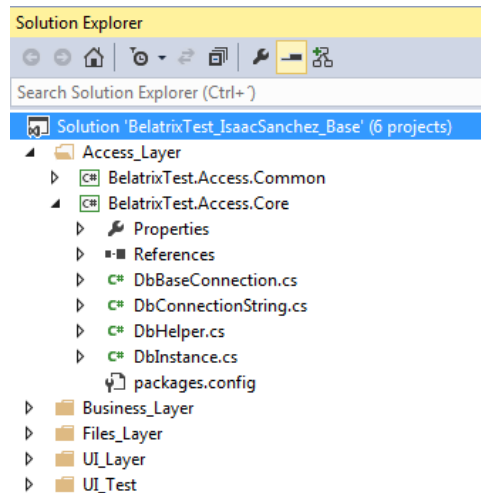
Access Layer: BelatrixTest.Access.Common

Made for adding common classes that implement directly database methods or object instances.



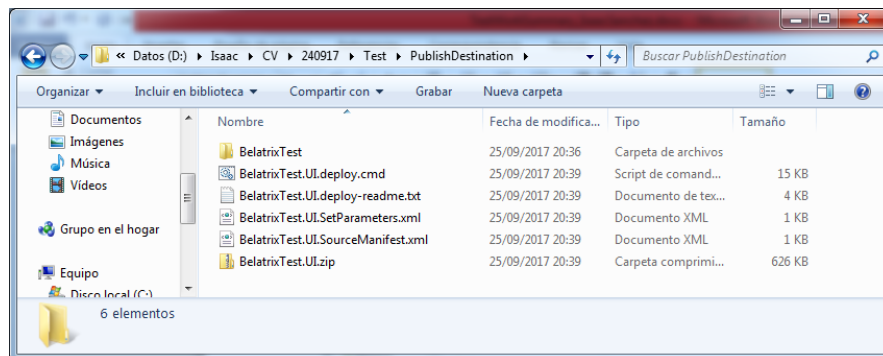
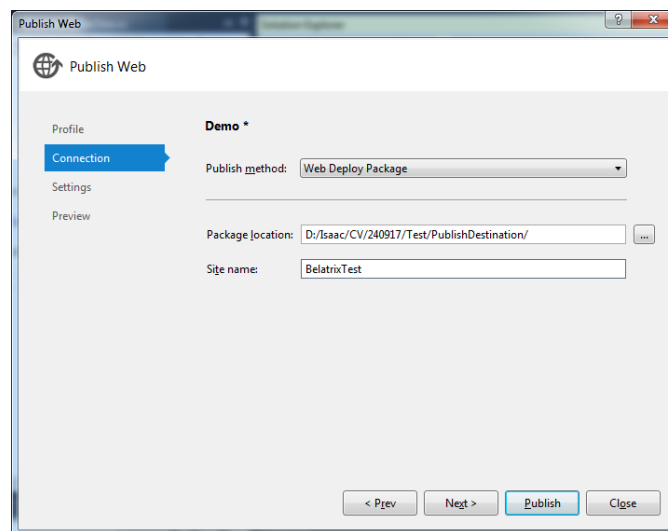
Access Layer: BelatrixTest.Access.Core

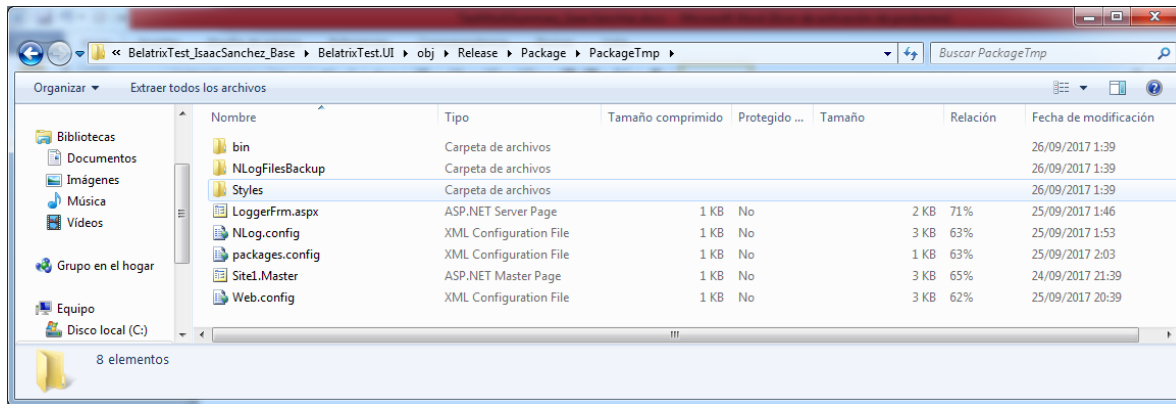
Most of the functions have been implemented to centralize the database classes and methods into a unique safe class.



Production Environment Setup

The current solution will be easily pluggable in production environments, you can replace the files to modify dynamic configuration while it is still running.





Changing logger plugins

Replace the NLog.config file to change paths, parameters and connection strings.

Changing class libraries plugins

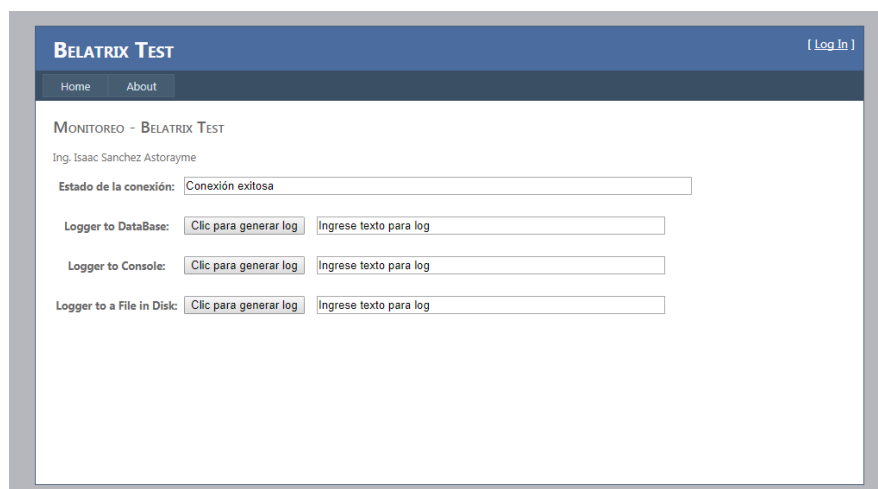
Clean and built libraries separately and replace only the layer or assembly you need to change.

Extend the BelatrixTest_IsaacSanchez_Base solution for new UI interfaces

The idea of using the layers oriented architecture is to deploy it in different servers across different development teams whose are able to add code as they could need, just using the solution as a base and referenced it.

How to use

The below instructions explain the main UI functionalities you can use with “BelatrixTest_IsaacSanchez_Base” application:



Database Connection Message

The Database Connection Message shows the connection status, it will show “Conexión Exitosa” or “No se pudo conectar” whether the database connection is already open or not.

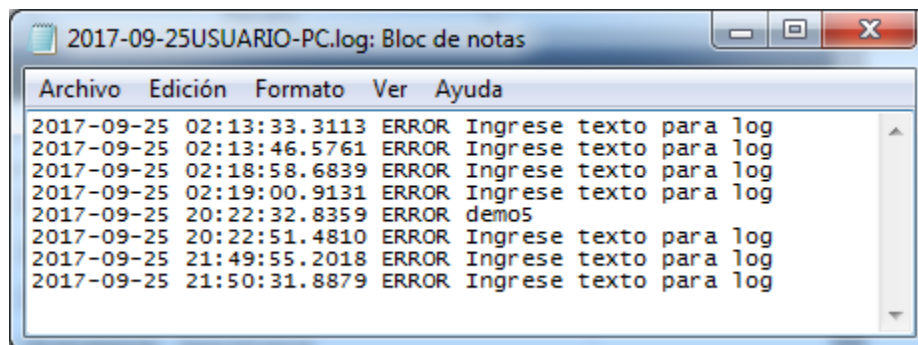
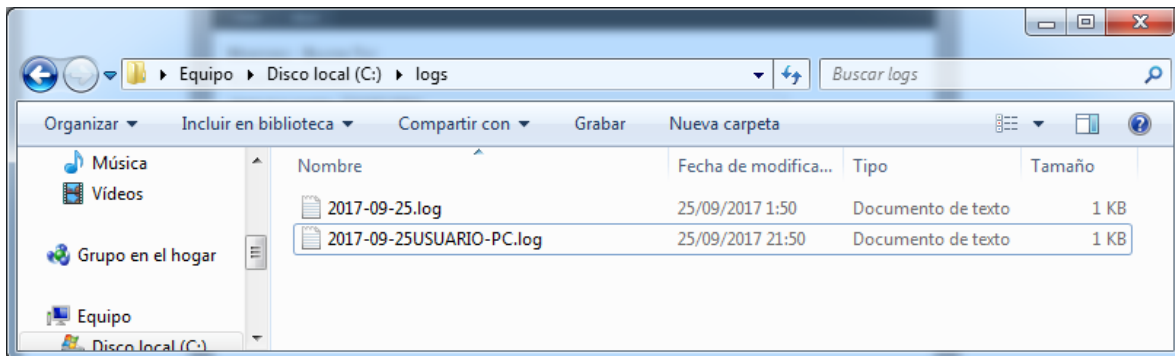
Estado de la conexión: Conexión exitosa

Logger to a File in Disk

Press this button to log an exception in a text file, the path configured is: C:\logs

Logger to a File in Disk:

The text you insert in the textbox will appear in the log:

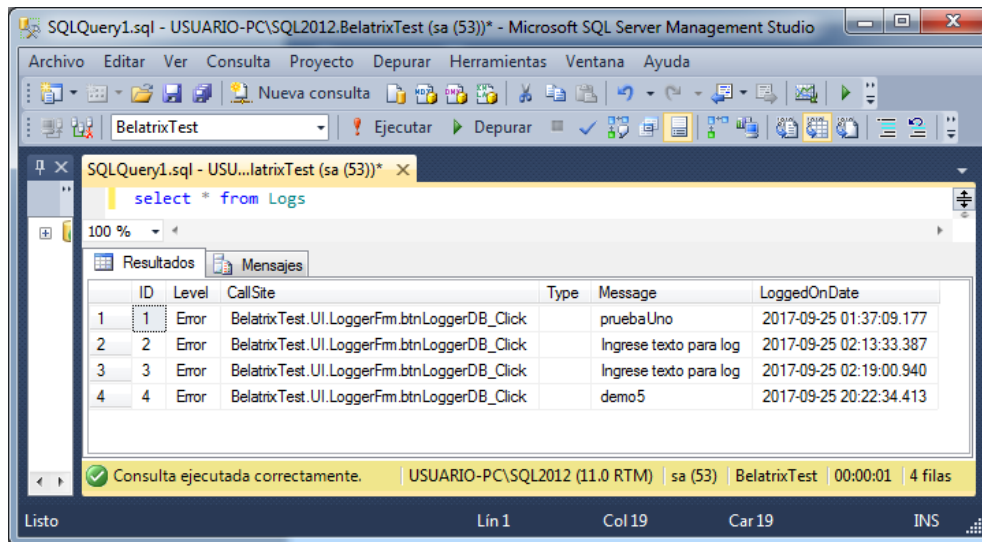


Logger to Database

Press this button to log an exception to a database table called "logs" in "BelatrixTest" database.

Logger to DataBase:

The text you insert in the textbox will appear in the log:



Logger to Console

Press this button to log an exception to console:

Logger to Console: