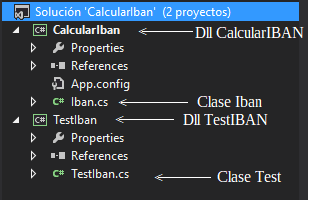
**Proyecto IBAN en C#**

Creamos el **ProyectoIBAN** con nuestro entorno gráfico Visual Studio 2012.

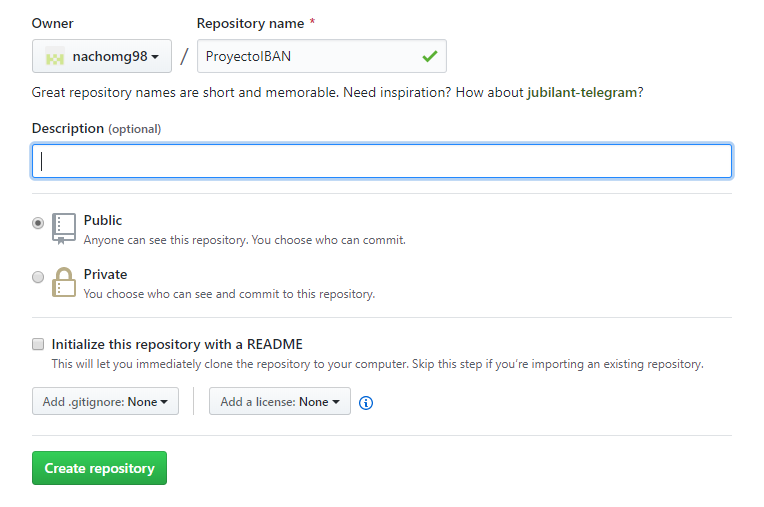
Agregamos a nuestro proyecto dos bibliotecas de clases, “**CalcularIBAN**” y “**TestIBAN**”. CalcularIBAN será la biblioteca de clase que contendrá nuestra codificación del cálculo, y TestIBAN será la biblioteca de clase que testeará la dll CalcularIBAN.

Cada Dll tendrá una clase que se encargará de lo explicado anteriormente. En la imagen siguiente se muestra como está realizado nuestro árbol de nuestro proyecto.

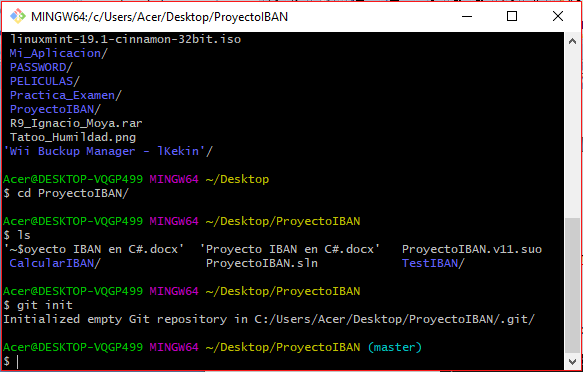


Creamos nuestro repositorio de GitHUB y Git. Van a ser nuestros repositorios donde subiremos los cambios que vayamos haciendo en nuestro proyecto y así tener un control de cambios y versiones del proyecto a realizar:

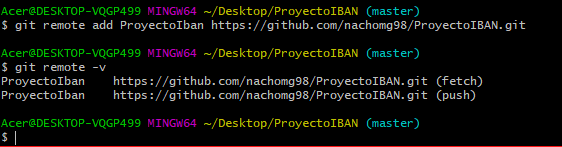
Creamos un nuevo repositorio en GitHub:



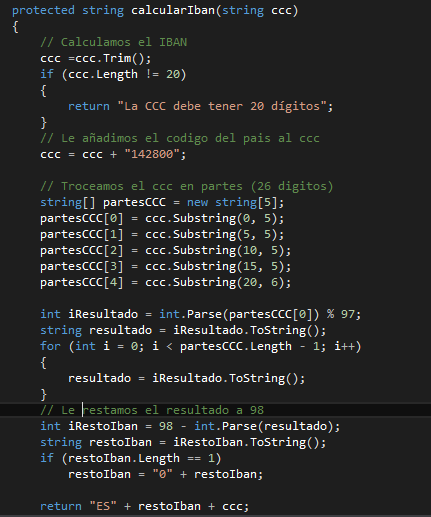
Ahora creamos un repositorio en GitBash:



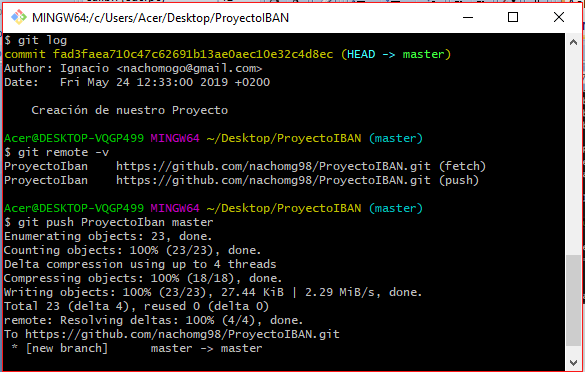
Ahora vinculamos nuestro repositorio Local con nuestro repositorio Remoto, porque los cambios lo realizaremos en nuestro repositorio local y luego lo subiremos al remoto.



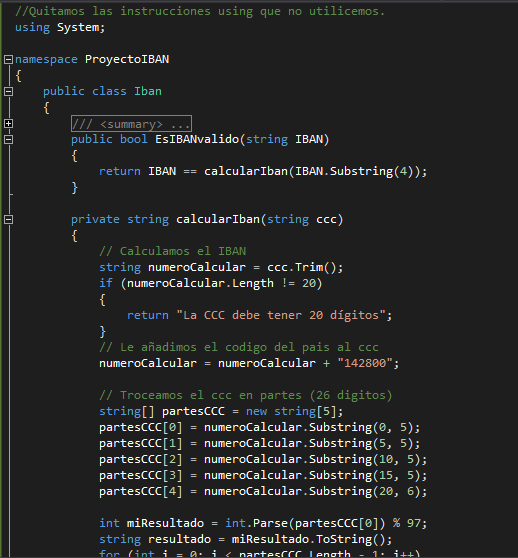
Implementamos el código C# para el cálculo del iban en nuestra clase Iban.cs:

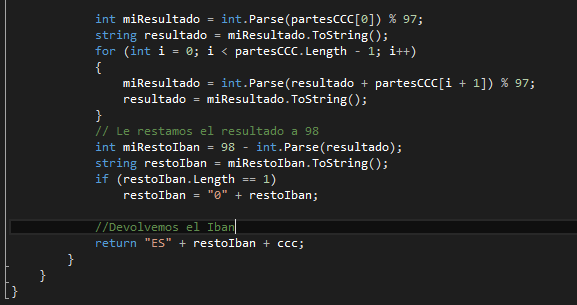


Subimos los cambios a Git y GitHub. Añadimos el proyecto a git y realizamos un commit con el mensaje “Creación del nuestro Proyecto". Luego lo subimos a nuestro repositorio remoto con git push.

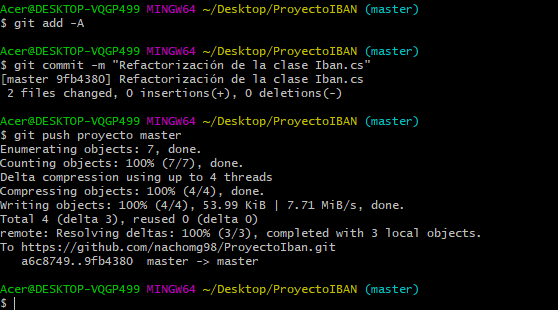


Como vemos el código no está refactorizado, por ello pasamos al proceso de la refactorización.

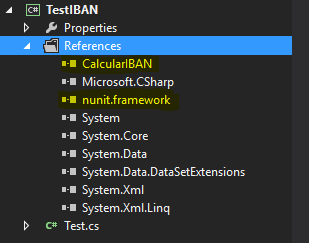




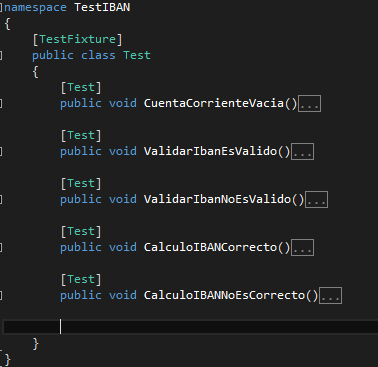
Volvemos a subir los cambios al git:



Una vez que tenemos la clase Iban.cs implementada y refactorizada, implementamos en la dll TestIBAN, los test unitarios del testeo que tenemos que hacer a la clase Iban.cs. Para ello lo primero que tenemos que hacer es referenciar la dll CalcularIban en Test y agregar la referencia Nunit.Framework.



Implementamos la clase Test.cs:



Subimos los cambios a Git: