MANUAL PROYECTO DESARROLLO INTERFACES

CONEXIÓN A BASE DE DATOS MYSQL

Carlos Prieto

Carlos Gabriel

Mariano de la Cerda

INDICE

Contenido

[1. INTRODUCCIÓN 3](#_Toc503946154)

[2. CREACIÓN BASE DE DATOS 3](#_Toc503946155)

[3. ESTRUCTURA DEL PROYECTO 6](#_Toc503946156)

[4. CONECTANDO A LA BASE DE DATOS DESDE VISUAL STUDIO 7](#_Toc503946157)

[5. CREAMOS LAS INTERFACES/VENTANAS QUE USAREMOS 8](#_Toc503946158)

[6. CREACIÓN DE METODOS Y FUNCIONES 11](#_Toc503946159)

[7. CREACIÓN DE INFORMES 17](#_Toc503946160)

[8. CONCLUSIÓN 17](#_Toc503946161)

# INTRODUCCIÓN

El objetivo de este programa es crear un CRUD de alumnos donde podremos crear, actualizar y borrar un alumno haciendo referencia a una base de datos creada en MySQL y llamándola desde Visual Studio usando WPF C#.

Nuestra interfaz está orientada a la facilidad, que el usuario sea capaz a simple vista de controlar el programa sin mucha complicación.

Para ello, disponemos de 2 ventanas para el login y el registro del usuario, otra ventana para el programa principal y una última ventana para añadir un nuevo alumno a la base de datos.

# CREACIÓN BASE DE DATOS

***INTRODUCCIÓN***

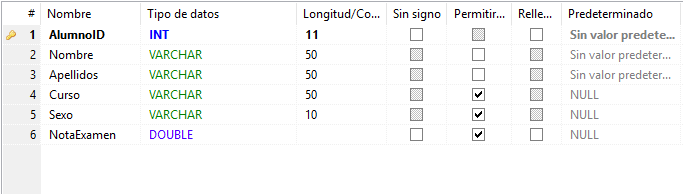
Lo primero que tenemos que hacer es crear nuestra base de datos que la hemos llamado colegio la cual, contiene cuatro tablas diferenciadas:

* Alumno: la tabla Alumno tendrá los siguientes datos:
* Nombre
* Apellidos
* Curso
* Sexo
* NotaExamen
* Examen: la tabla Examen tendrá los siguientes datos:
* ExamenID
* ProfesorID
* Asignatura
* FechaExamen
* Profesor: la tabla Profesor tendrá los siguientes datos:
* ProfesorID
* Asignatura
* Nombre
* Apellidos
* Curso
* LoginAlumno: la tabla LoginAlumno tendrá los siguientes datos:
* AlumnoID
* Usuario
* Clave

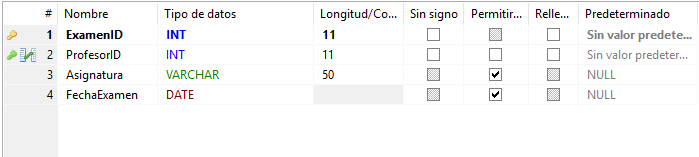
Para comenzar el proyecto lo que hemos hecho es preparar las bases de datos que vayamos a utilizar.

Tenemos cuatro bases de datos:

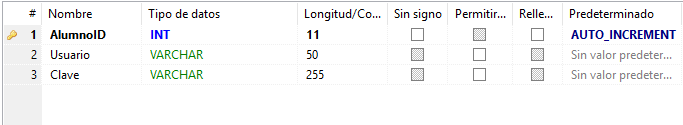
* **Alumno**



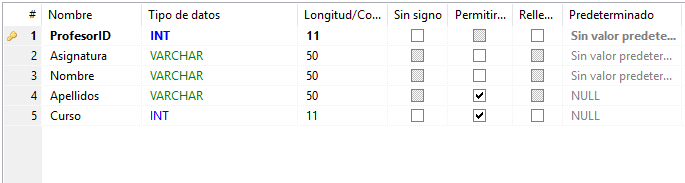
* **Examen**



* **LoginAlumno**

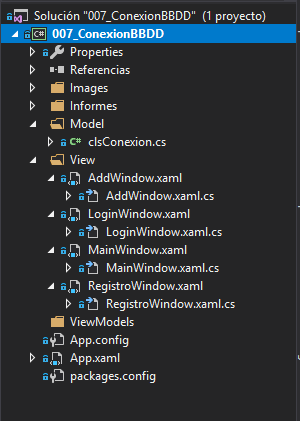


* **Profesor**



# ESTRUCTURA DEL PROYECTO

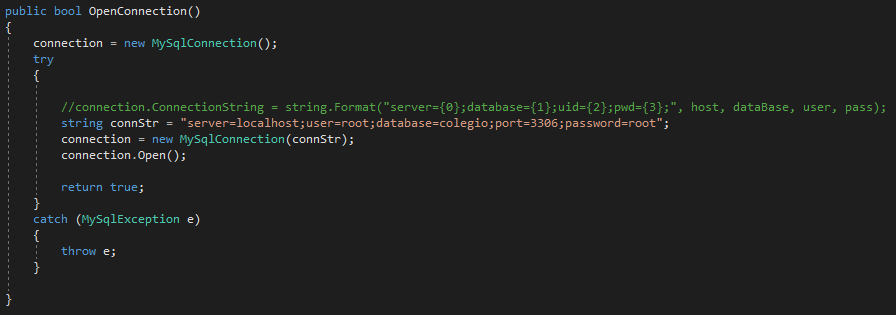
Para nuestro proyecto hemos utilizado el patrón Modelo Vista Controlador y a parte tendremos los recursos, tales como las imágenes, informes, … Como vemos en la imagen:



# CONECTANDO A LA BASE DE DATOS DESDE VISUAL STUDIO

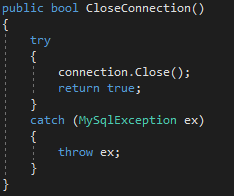
Lo primero que debemos hacer es configurar el plugin de mysql para Visual Studio sqlite (mysql-connector-net-6.10.5.msi) y lo podemos encontrar en: <https://dev.mysql.com/downloads/windows/visualstudio/> .

Una vez ya configurado todo tendremos que establecer la conexión:



Creamos una instancia de MySqlConnection; creamos un string donde le pasemos todos los parámetros (server, user, password, database, port), y le pasamos tal string al objeto MySqlConnection creado anteriormente y la abrimos.

Luego para cerrarla debemos de llamar al método Close():



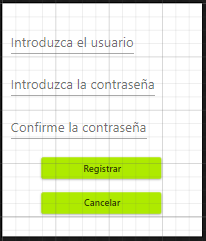
# CREAMOS LAS INTERFACES/VENTANAS QUE USAREMOS

En nuestro caso usaremos 4 diferentes ventanas.

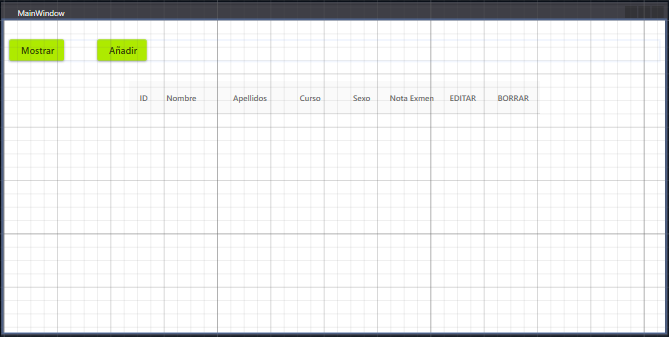
* **LoginWindow.** Esta es la primera ventana que verá el usuario al iniciar el programa, es un simple login donde comprobará los credenciales introducidos con los existentes en la base de datos, en caso que coincidan se cerrará la ventana y pasará al MainWindow; en caso contrario saldrá un mensaje de error para que el usuario vuelva a introducir los datos. Tiene otra opción para crearte una cuenta nueva en caso de que no tengas una, lo cual llamará a otra ventana para hacer el registro.



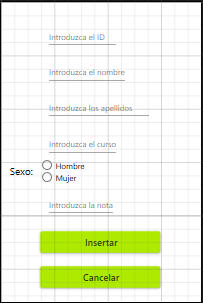
* **RegistroWindow.** Esta ventana se abre al clickar en “Registrarse” en la ventana LoginWindow. El usuario introducirá tanto el usuario como la contraseña (tiene que volver a introducirla para verificarla) y esto creará un nuevo campo en la tabla LoginAlumno de la base de datos donde aparecerá el usuario y la contraseña encriptada para que disponga de más seguridad la aplicación.



* **MainWindow.** Esta es la ventana principal del programa, la que contiene toda la lógica del mismo. Tiene un estilo simplificado para que al usuario le resulte fácil acceder a todas sus funciones. Nada mas entrar nos encontraremos con toda la información de los alumnos mostrada en una tabla. En la parte superior tendremos un botón “mostrar” que actualizará la información de la tabla, seguido de un botón “añadir” que abrirá una nueva ventana para insertar un nuevo alumno. Dentro de la tabla nos encontramos con dos botones, uno para borrar un alumno y otro para editar el mismo alumno.



* **AddWindow.** Esta ventana se abre al pulsar en “añadir” en el MainWindow. El usuario deberá introducir todos los campos para luego insertarlos en una nueva fila de la tabla Alumno, en caso de que haya introducido un campo erróneo saltará un error y no se insertará dicho lumno. Al terminar de insertar el alumno se cerrará la ventana y se actualizará la tabla con la información de los alumnos.



# CREACIÓN DE METODOS Y FUNCIONES

A continuación, creamos los métodos que vamos a usar a lo largo de nuestra aplicación:

* **Encriptar.** Este método encripta la contraseña introducida en un campo de texto de la ventana LoginWindow.

El string hash es el patrón para encriptar/desencriptar la contraseña.

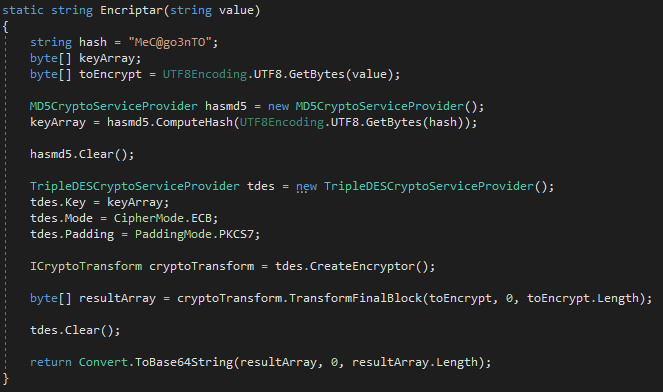
Instanciamos un byte de arrays vacio, y un un byte de arrays que contiene los bytes de la contraseña introducida por teclado.

Instancia el servicio que computa el hash que hemos introducido y los guardamos en nuestro keyArray, y limpiamos el flujo.

Luego definimos un objeto para obtener acceso a los servicios criptograficos. En key introducimos el keyArray, en mode el tipo de cifrado y en padding se establece el modo de relleno usado en el algoritmo.

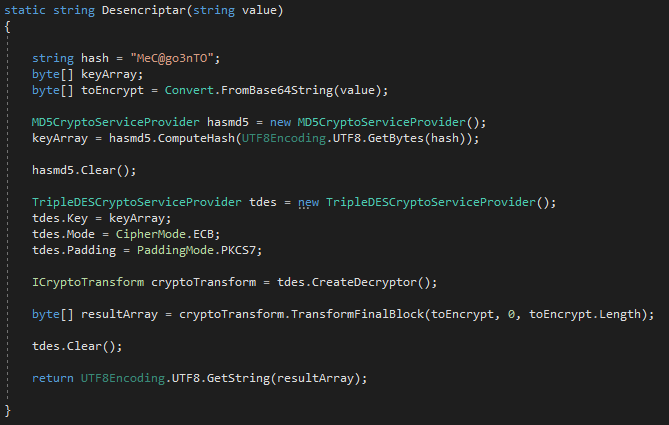
ICryptoTransform define las operaciones basicas de las transformaciones criptograficas. Despues convertimos la region especificada, en este caso el array de bytes keyArray y la guardamos en otro array de bytes resultArray.

Limpiamos el flujo y devolvemos el resultados convertido a String.

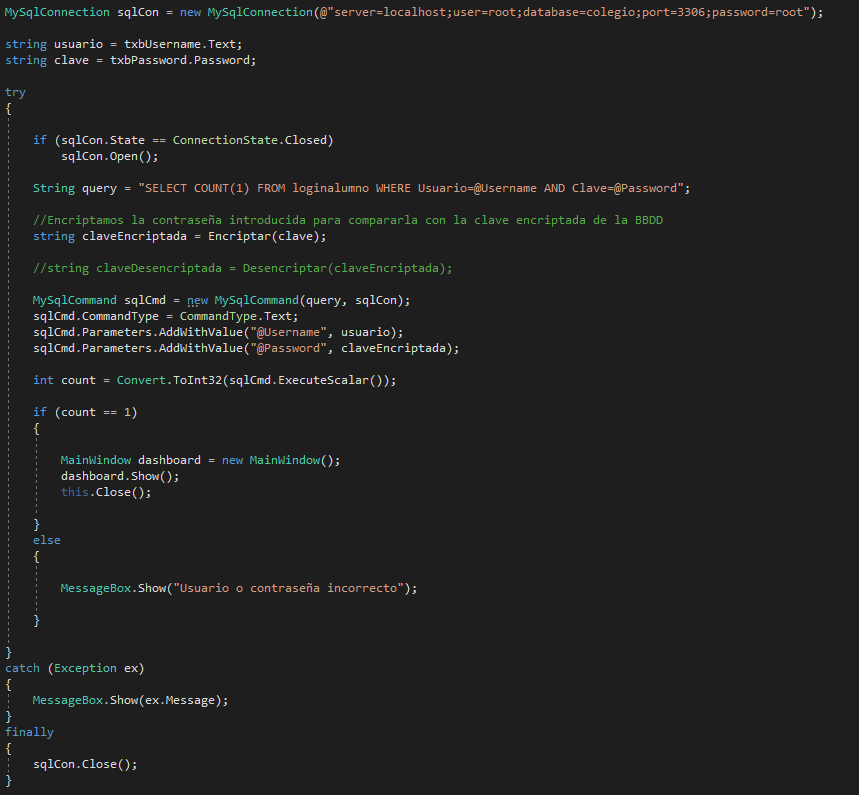


* **Desencriptar.** Este método encripta la contraseña encriptada anteriormente para poder ser vista por el usuario.

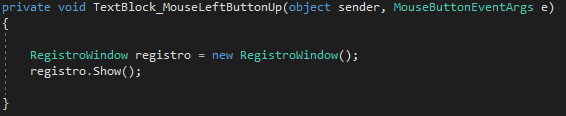
Desencriptando utilizamos practicamente los mismos pasos que enctriptando, ya que la metodologia es la misma pero en el paso “IcryptoTransform” utilizamos “CreateDecryptor” en vez de “CreateEncryptor” y asi desencriptamos la contraseña hasheada.



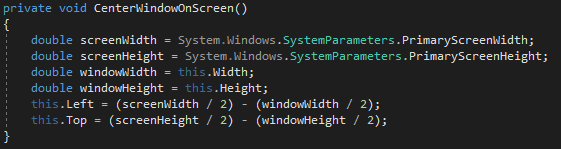
* **Login.** Este método comprueba que los campos introducidos (usuario y clave) correspondan con los campos (usuario y clave) de la base de datos, si es así llama a la ventana MainWindow, en caso contrario muestra un mensaje de error.



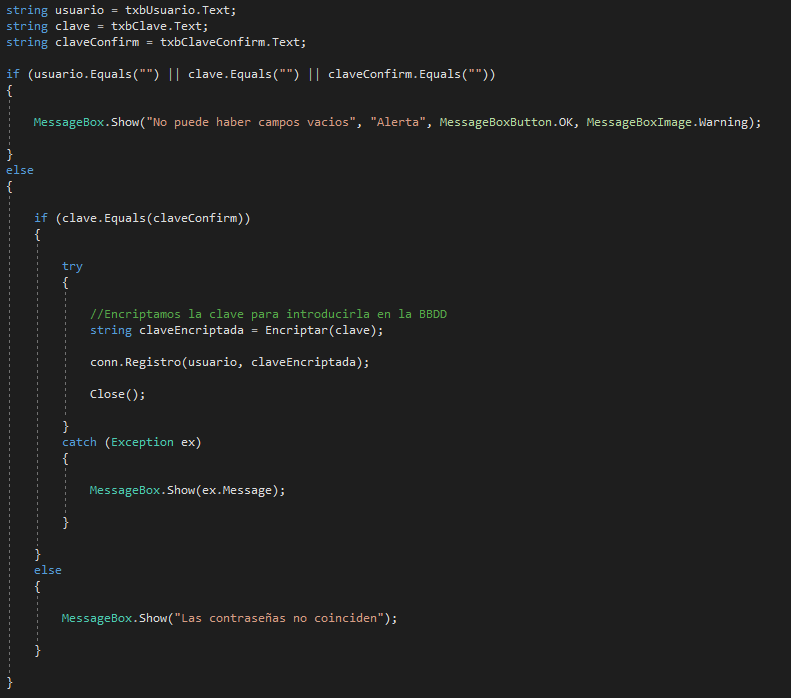
* **Llamar a Registro.** Este método llama a la ventana RegistroWindow al hacer click en registrar, para crear un nuevo usuario.



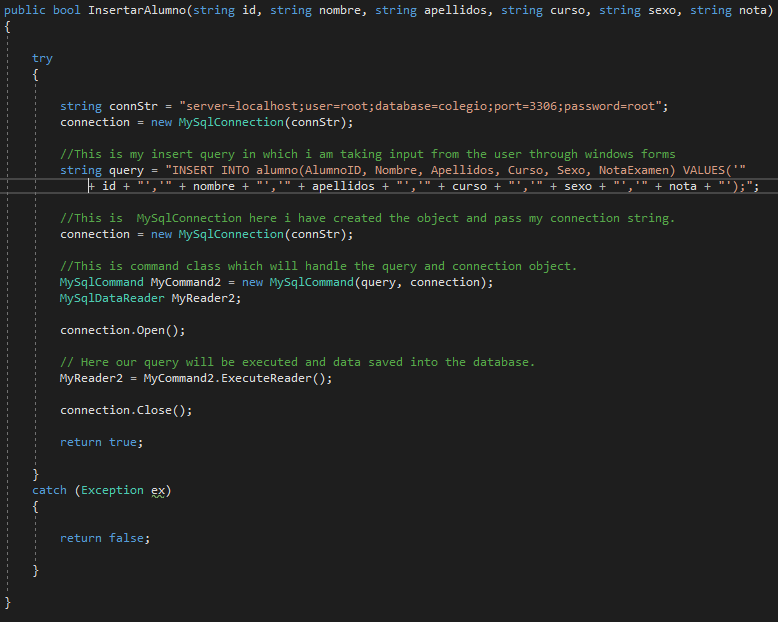
* **Centrar Ventana.** Este método centra la ventana al centro de la pantalla.



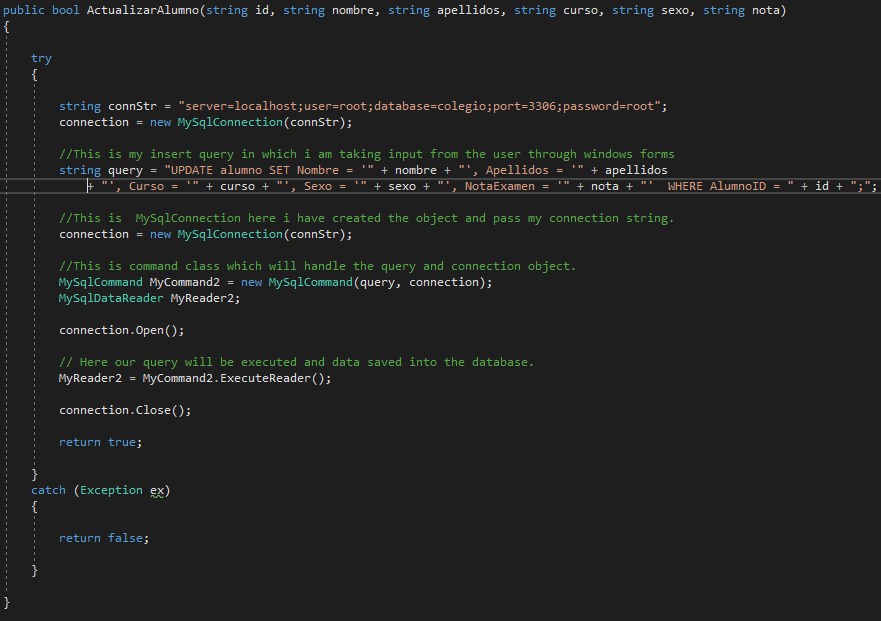
* **Registro.** Este método recoge el nuevo usuario introducido, la nueva contraseña (antes hacemos una confirmación de contraseña para asegurarnos que es la clave que desea el usuario) y a continuación se llama al método Encriptar para introducirlo en la base de datos encriptado.



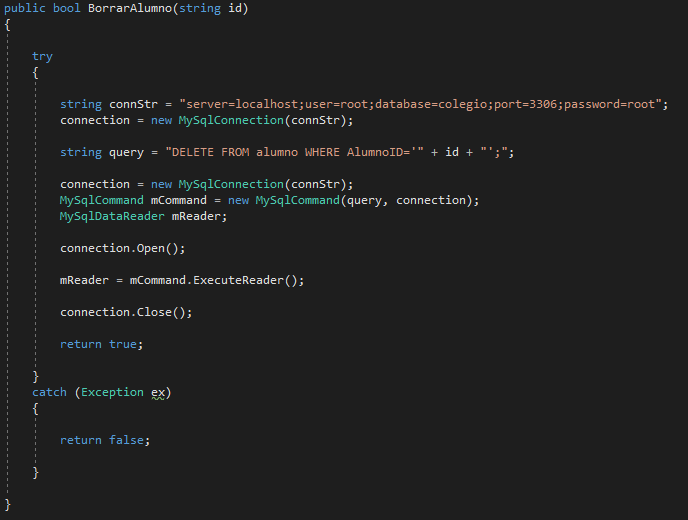
* **Insertar.** En este método insertamos un nuevo alumno a la base de datos, para ello abrimos la conexión, creamos la query, ejecutamos la query y cerramos la conexión.



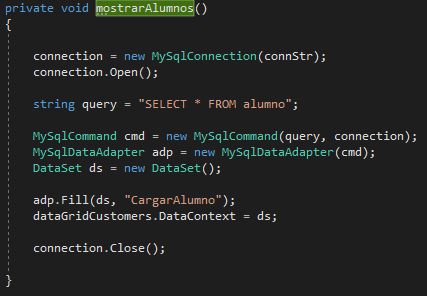
* **Actualizar.** En este método actualizamos un campo de alumnos en la base de datos. Lo único que cambia respecto al Insertar es la query.



* **Borrar.** En este método borramos un alumno de la base de datos.



* **Mostrar.** En este método hacemos un select a la base de datos para recoger todos los alumnos que existan y así cargarlos en un datagrid para que el usuario los vea.



# CREACIÓN DE INFORMES

Para la creación de informes lo que debemos hacer es descargar el ODBC de MySql <https://dev.mysql.com/downloads/file/?id=472464> y el CrystalReport para conectarlo con el Visual Studio <https://wiki.scn.sap.com/wiki/display/BOBJ/Crystal+Reports%2C+Developer+for+Visual+Studio+Downloads> .

A continuación, instalamos el CrystalReport. Durante la instalación deberemos seleccionar la base de datos que queramos crear el informe. Más adelante tendremos que seleccionar todas las tablas seguidas de sus atributos que queramos mostrar en el informe.

Po último seleccionamos el estilo que le queremos aplicar al informe, y ya estaría creado, nos creará un nuevo archivo en el proyecto donde podremos visualizar el informe de nuestra base de datos.

# CONCLUSIÓN

PUNTOS POSITIVOS

El proyecto ha tenido un desarrollo adecuado porque se han mezclado conceptos a largo de este por ejemplo hemos utilizado c#, hemos usado SQL y hemos utilizado xml.

PUNTOS NEGATIVOS

Un gran punto negativo es que hemos tenido muchos problemas para hacer los informes porque nos es fácil implementar los informes y a parte también es difícil implementarlos en el IDE.

* Opinión:

Carlos Prieto: en mi opinión podríamos haber utilizado una herramienta más avanzada en cuanto a tecnología porque la hemos usado no era una herramienta compatible y por ello hemos tenido problemas.

Mariano de la Cerda: yo opino que en general el proyecto ha estado bien, los puntos que se tendrían que pulir y mejorar sería el trabajo en equipo, es muy importante, y en este grupo hemos clasificado el trabajo en partes que cada uno hemos hecho.

Carlos Gabriel: el proyecto me ha parecido bastante simple pero me ha gustado la forma de hacerlo porque al menos hemos podido trabajar en equipo.