



ACTIVIDAD	Zumita Refactoring
Código	AA1_06_ZumitaRefactoring
Tipo	Evaluable. Diseño e implementación en grupos de 2
Objetivo	Uso de clases. Visibilidad.Constructores Organización del código Uso de STL vector Interfaz(.h) separada de la implementación (.cpp)

Añadir un nuevo proyecto a la solución “Actividades” llamado **AA1_06**.

El objetivo de esta actividad es hacer un “refactoring” de la actividad 05 (AShootBallsGame) con la finalidad de aplicar los conceptos nuevos estudiados: Clases y STL vector. Para ello, considerar, al menos, las siguientes recomendaciones:

- a) Convertir el struct Player en class Player. Considere la visibilidad de los atributos. Si lo cree conveniente implemente setter's y/o getter's.
- b) Transformar el método init de la clase Player en un constructor de la clase.
- c) Convertir el struct Panel en class Panel. Considere la visibilidad de los atributos. Si lo cree conveniente implemente setter's y/o getter's.
- d) Transformar el método init de la clase Panel en un constructor de la clase.
- e) Cambiar el array dinámico que usa el panel por un <vector> de STL.
- f) Reconsiderar la implementación de los métodos insert, verifíer, deleteThree, insertThree para que hagan uso de las funcionalidades que nos da STL vector
- g) Modificar el main para que funcione con las nuevas clases definidas.

Las buenas praxis dicen que los nombres de los métodos de las clases han de utilizar la notación UpperCamelCase.