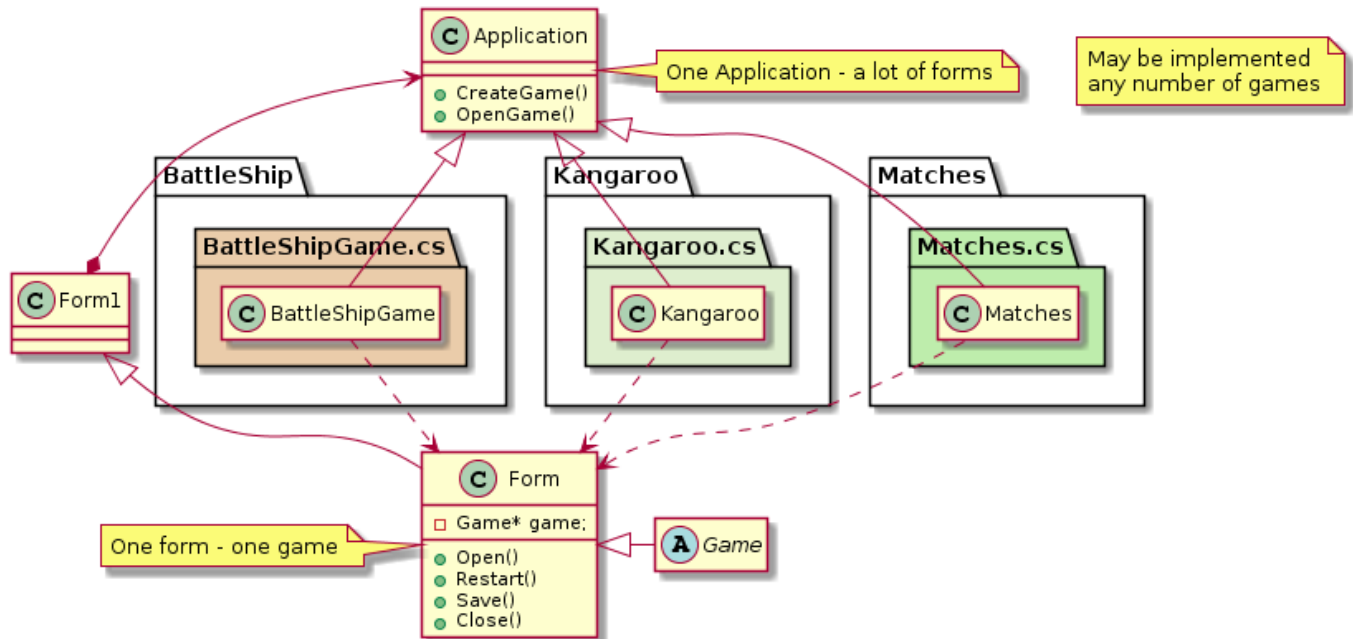


# Design Patterns project

In this repo will be described the TMPS' project which will include different patterns such as Creational, Structural and Behavioral.

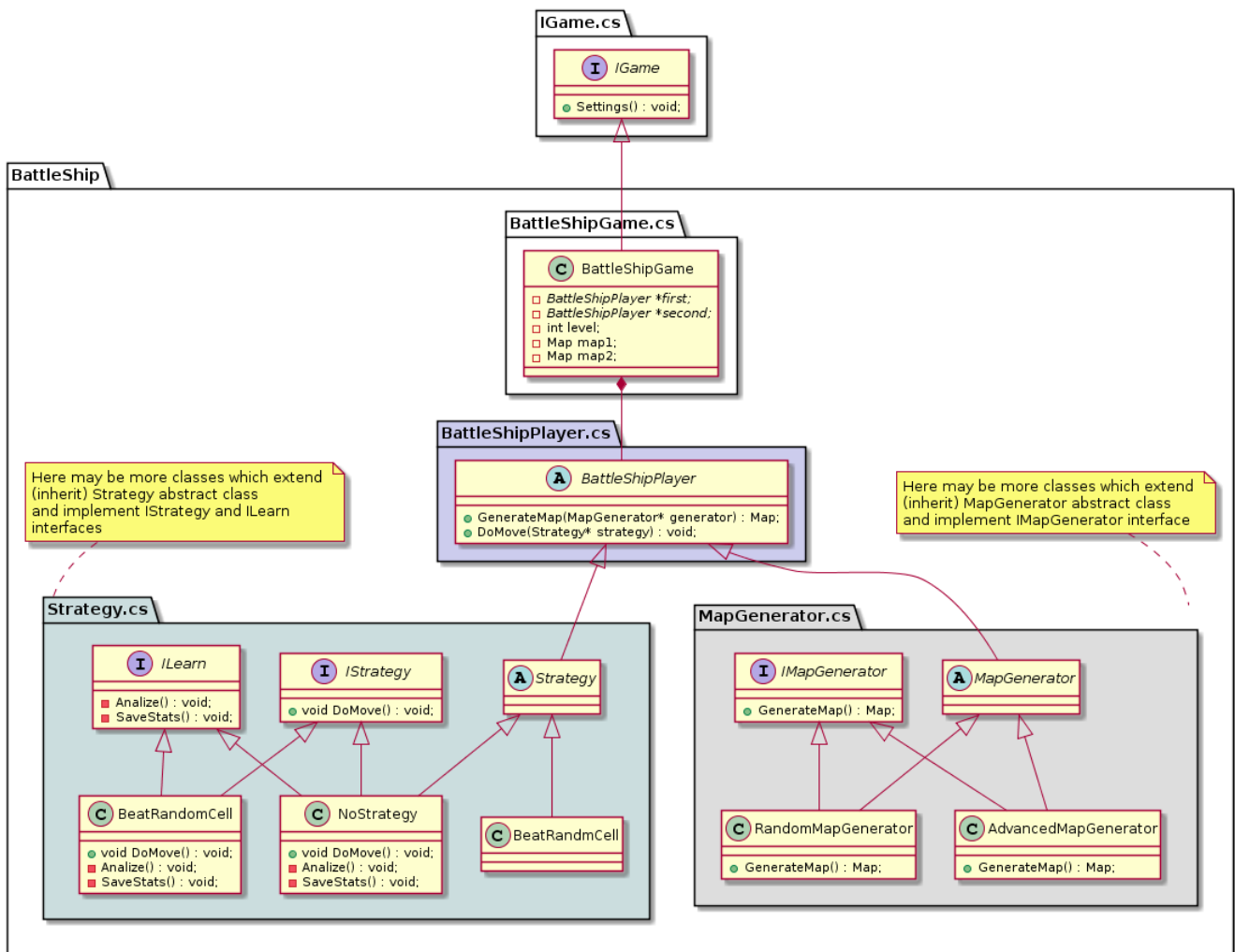
First of all, below are illustrated all initial requirements of the project. The project is called Games.

Application - Class Diagram



As it can be seen from this image the application has different set of Forms, but at each Form int the concrete moment could be initialized only one game. Moreover, each name\_of\_game class inherits the abstract class Game and implements the IGame interface. It gives the possibility to use the Game abstract class as a protector in that scence that besides Form1 (default window class of application) could be opened only one Form in which there is only one game. The backend of the corresponding Form will depend on implementation of respective class.

## Battleship Game - Class Diagram



Here is presented the content of BattleShip folder, and in it there are implemented several Creational design patterns. All these patterns make the process of Runtime initializations of the objects easy as it possible. It's useful, because the user doesn't need to know what and how exactly is implemented in code - all he/she needs is frindly multifunctional GUI.

The reference to my Github project: <https://github.com/VladGanuscheak/TMPS>