

Main changes made to the initial project "BattleField 4":

Extracted static readonly int[,] mine fields into separate classes and used template method in order to extract same behaviour of the different mines in an abstract class called Mine. This made the mines in the game more flexible to change through inheritance and methods which can be overridden if necessary.

Added IExplodable interface to the mines so they can be used with a strategy pattern and be easily extended.

Refactored public static void PrintField method in a separate class called ConsoleRenderer, which uses an interface IDrawable, and can be easily replaced with some other renderer.

WebBrowser renderer. Not only extracting the method but decoupling the drawing logic from the game logic.

Refactored public static int GrymOtQsnoNebe method into a separate class called ExplosionHandler. This class is responsible for the way the explosion of the current mine is handled and the way the explosion takes effect on the field.

Refactored public static int TimeToPlay method into a separate class called InputHandler. The InputHandler class is responsible for the user input validation and data.

Refactored public static void InitiateGame method into several classes and methods.

Engine class – responsible for the game logic, used as a facade in order to facilitate the usage of the game.

Made several other methods private for the game engine to separate the logic of initialisation of the game and the actual gameplay.

The aforementioned changes make sure the SOLID, as well as strong cohesion, loose coupling, encapsulation, polymorphism and exception handling principles are fulfilled. Also the variable classes and method names are refactored in order to be easily understood in order to fit the naming conventions and further refactored if necessary.