|  |
| --- |
| Refactoring Documentation for Project “Minesweeper – 5”Team "Minesweeper-3"  1. Providing a new Solution folder. 2. Separating the project into three main subprojects.   2.1 Class Library Application  2.2 Console Client Application  2.3 Unit Test Application   1. Dividing the game classes into separate files, stored into the Class Library Application project 2. Separating the static methods into new static class 3. Refactoring MinesInitializer class main methods - > PlayGame method (removing goto command) 4. Providing refactoring and code comments 5. MinesInitializer class refactored according to Singleton design pattern. 6. private static MinesInitializer onlyInstance field contains the only instance of the class 7. private MinesInitializer() disables external instantiation of the class 8. public static MinesInitializer Instance() creates the only instance with lazy loading and returns it to the clients 9. Basic interfaces created 10. IScoreBoard for the score board implementations 11. IMinesGenerator for the random mines generator implementations 12. IMinesGenerator.FillWithRandomMines(…) generates the mines using random deployment algorithm 13. IDrawer for the UI related operations 14. IMinesweeperFactory for the game factory; through this interface implementations of the ones above must be obtained 15. IMinesweeperFactory implements the “Abstract factory” design pattern 16. public class MinesGenerator : IMinesGenerator is the generator of mines used by the game 17. class ScoreBoard : IScoreBoard moved to dedicated folder |
|  |