**Sample Refactoring Documentation for Project “Game “King Survival”**

**Team "King-Survival-7"**

1. Used design patterns:

* Singleton for initialization of the Chess board
* Factory for initialization of Figures
* Facada - including all the logic of the game
* .

1. Redesigned the project structure:

* Renamed the project to **KingSurvivalRefactored**.
* Renamed the main class **KingSurvival.cs** to **KingSurvivalConsole.cs**.
* Create folders to keep the logic separated: Enum, Exceptions, GameCore, GameRenderer, RefactoringDocumentation, UserInteraction
* Extracted each class in a separate file with a good name in the proper folder
* Create Class Diagram: **KingSurvivalClassDiagram**.**cd**
* Created Unitests corresponding to the game in project **TestKingSurvival**

1. Reformatted the source code:

* Naming conventions: Character casing: variables and fields made camelCase; types and methods made PascalCase. - changed class names king and paw, other
* All names converted to English meaningful names: deleted peshkaA – peshkaD, car, other
* Removed all unneeded empty lines, e.g. in the method, Inserted empty lines between the methods. In method Print in class KingSurvival
* Split the lines containing several statements into several simple lines, e.g.:
* DRY – changed and extracted all repeated code in all methods (Try(), PawwAMove()….)
* Single responsibility – all functionality that prints on the console moved in different class

1. Introduced constants:

* BoardSize = 8

1. Introduces 3 new Exception implementations all inheritance of **: ApplicationException**

* GameExitException.cs , IllegalMoveExeception.cs, InvalidCommandException.cs

1. Created Enumerations: FigureSymbol.cs where it holds the letters of the Figures. And Movement – which holds all directions: UpLeft, UpRight, DownLeft, DownRight
2. Deleted classes kingg and paww. The classes were almost identical – DRY principal. And created structure Position that holds the X and Y coordinates. And class Figure that initiate in its constructor properties: (enum) FigureType and char Name.
3. Introduced class ChessBoard and moved all related functionality in it.

* Instantiate ChessBoard with Singleton Design Pattern.
* Changes method Print(char[,] matrix) in public class KingSurvival to

Method ToString() in public class ChessBoard. Included Stringbuilder for optimization.

1. Deleted class KingSurvival and initialization of the figures created with Factory design in class FigureFactory.cs
2. All game logic is moved to class GameEngine.cs with all printing on the console moved to ConsoleRenderer.cs
3. Introdused UnitTest Project , plus Mocked objects