# Refactoring Documentation for Project “GameFifteen”

Cherokee

Team “Chlorine”

1. Redesigned the project structure:
   * Renamed the project to **GameFifteen**.
   * Renamed the main class **Game** to **GameFifteen**.
   * Extracted each class in a separate file with a good name:
     1. **Position.cs**
   * Full redesign of program to work with **GameBoard** class and not with static methods.
2. Changes in class **Position** 
   * Renamed constructor to **Position.**
   * Added **‘this’** to properties used in constructor.
   * Added private fields representing the properties.
   * Added checks for incorect values for properties in setters – negative values -> throw **ArgumentOutOfRange**.
   * Added comments to public members.
   * Added test pattern for **Position** class.
3. Changes in class **GameFifteen**
   * Rename old named **Pozicia** class references to new one **Position.**
   * Mehod **findPrazno** renamed to **FindEmptyCell** and moved to class **GameBoard.**
   * Extracted **Drawmatrica,** renamed to **Draw()** and moved to class **GameBoard.**
   * Extracted **Proverka,** renamed to **IsSolved** and moved to **GameBoard.**
   * Extracted **GenerateGameFild,** renamed to **GenerateFild** and moved to **GameBoard.**
   * Extracted **FillOutGameField,** renamed to **FillOut** and moved to **GameBoard.**
   * Made various identifier name changes - indexer variables (i, j -> row, col), matrix identifiers, renamed Cyrillic names to their correct English counterpart, removed some unnecessary comments, changed row- and column-related names to expose their real usage
   * Fixed spacings around control structures, brackets, and array initializers.
   * Changes in the main game controller - magical numbers fixed, formatting and renaming, unclear naming fixed, unnecessarily complex control structures simplified and fixed
   * Introduced constants - game field size, number of players in the highscores table
   * Fixed output message errors (putted, invalid spacing, inconsistent or improper messages)
   * Changed some exceptions (e. g. ArgumentException for null game board should be ArgumentNullException) and fixed the expected exceptions in the unit tests to show that behaviour
   * Fixed about 250 Stylecop warnings (mostly whitespaces left where there shouldn't be any and wrong placement of brackets
   * Fixed method **UpdateAndDrawBoard** - does two things and accepts too many parameters
   * Refactored the GameBoard class - too many instance methods
   * Added instances of Position class instead of integers wherever possible
   * Moved some methods which accept a board to the GameBoard class and made them instance methods
   * Fixed the console output throughout the game
   * Bug fix - the highscores do not print
   * Big fix - agter Good bye! nothing must be printed
   * Documented the public methods which lacked documentation
   * Added unit tests for the GameBoard class
   * Added exception handling for the GameBoard class
   * Made the GameBoard accept a string array as body
   * Removed all unneeded empty lines, e.g. in the method **FillOutGameField**.
   * Inserted empty lines between the methods.
   * Split the lines containing several statements into several simple lines
   * Formatted the curly braces **{** and **}** according to the best practices for the C# language.
   * Put **{** and **}** after all conditionals and loops (when missing).
   * Character casing: variables and fields made **camelCase**; types and methods made **PascalCase**.
   * In class **GameFifteen, FillOutGameField(…)**: **rnd** 🡪 **rand**; **matricata** 🡪 **gameField**; **used** 🡪 **usedNumbers**; **isFilled** 🡪 **isPositionFilled**.
   * -In **Main(string[] args)**: **g** 🡪 **gameFifteen**.
   * Created method **GenerateGameField()** in class **GameFifteen**.
   * Created method **IsSolvable(string[,] gameField)**.
   * Fixed a bug in method (**proverka(string[,] matrica)**).
   * Added some notes in code as comments – will be removed later (**// TODO: …**).
   * Fixed a game logic issue concerning introduction of Exceptions and logic that didn’t expect exceptions (Position(-1, -1)).
4. Changes in class **GameBoard**
   * New array representing game board coded as class with geter/setter, indexer.
   * Added readonly field for board size **GameBoardSize.**
   * Removed magic number in method **FindEmptyCell** (replaced with **GameBoardSize**).
   * Added a few tests for **IsSolvable(…)**.
   * Fixed a bug in **IsSolvable(…)**.
   * “Fixed” some “magic” numbers
5. Changes in class **HighScores**
   * Refactored class **Highscores**, moved all related functionality in it
   * Wrote tests for **Highscores**’ methods. Found and fixed a bug.
6. Changes in class **Scores**
   * Created class **Score** with **Moves** and **PlayerName** properties.
   * Wrote tests for **Score**