King Survival 3

# Team members

Дженко Пенев – dzhenko

Ясен Михайлов – Yasen

Павел Христов – lithical

Светлин Няголов – tinskiwe

Лъчезар Томов – lbtomov

Мартин Маринов – Martin\_N\_Marino

Никола Тодоров – nmtodorov

Христо Илиев - Hristo.Iliev

# Design Patterns

### Structural:

Adaptor Design Pattern – Renderer.

Decorator – Renderer.

Façade - Game Pieces.

### Creational:

Builder Design Patter – Game Pieces.

Abstract Factory – Game Pieces.

### Behavioral:

Observer – GameBoard

# Refactoring

The project was refactored the following way:

A **Game** class was introduced to handle all logic of the game with sub classes. A **Renderer** was created to handle the rendering of the game. The logic for the King and the Pawns was separated intotheir ownclasses and they inherit the **Piece** class and **Game Board** was created to keep to data for the game. **Game Engine** class was introduced to handle the game logic. All the constants were move to a static class called **GameConstants**. **Validator** class for validating properties. **Coordinates** structure for keeping coordinate of chess figures.

The **Renderer** is using the Adaptor and Decorator deign patterns.

Decorator design pattern: **RenderBase** - component, **RendererConsole** – concrete component, **TextRendererDecoratorBase** – decorator and **TextRendererDecorator** – concrete decorator.

Adapter design pattern – **GameRendererAdaptor** – adapter, **TextRendererDecorator** - adaptee, **GameEngine** – target.

**Game Pieces** are created by an Abstract Factory, a Builder and a Façade design patterns.

Builder Design pattern – PlayerGamePieceDirector – director, PlayersGamePieceBuilder – buider, Player1GamePieceBuilder and Player2GamePieceBuilder – concrete builders

Façade Design pattern – PlayersAllGamePiecesCreator

Abstract Factory – PawnsAndKingsFactory – abstract factory, PlayersAllGamePiecesCreator – concrete factory, Piece – abstract product, PieceKing and PicePawn – concrete products.

Observer – GameBoard is notifyed when a piece change her coordinates

**GameBoard:** holds game field without game pieces; Image of current game board with pieces; Constants: TotalPlayfieldSize (size of board without decorations on screen) and some for positioning game board on screen; When board is notified, calculates relation between screen and logical coordinates