Formal Modeling of a Tetris Game

Mestrado Integrado em Engenharia Informática e Computação

Métodos Formais em Engenharia de Software

Grupo 1 Turma 4MIEIC02
Ângela Cardoso - up200204375
Tiago Galvão - up201500034
Nuno Valente - up200204376

December 24, 2016

Contents

1 Informal system description and list of requirements

1.1 Informal system description

Tetris game it's a puzzle game and one of the most recognizable and influential video game brands in the world. It's no wonder why there are hundreds of millions of Tetris products being played, worn, and enjoyed by fans in their everyday lives. The game was born in 1984 and it's living proof of a game that have truly transcended the barriers of culture and language.

The rules to play the game are very simple. Tetris game requires players to strategically rotate, move, and drop a chaining of tetrominos that fall into the rectangular board at increasing speeds. Players attempt to clear as many lines as possible by completing horizontal rows of blocks without empty space, but if the tetrominos surpass the skyline the game is over! Speed and consequent level advance can make the game ally to strategy more enthusiastic.

One meritorious reference to Alexey Pajitnov because he his the person who developed this popular game. He is a russian video game designer and computer engineer and in his spare time, he drew inspiration from his favorite puzzle board game, pentominos, and decided to create a computer game for himself. Pajitnov envisioned an electronic game that let players arrange puzzle pieces in real time as they fell from the top of the playing field. The resulting design was a game that used seven distinctive geometric playing pieces (appendixB), each made up of four squares. Pajitnov called this game "Tetris," a combination of "tetra" (the Greek word meaning "four") and "tennis" (his favorite sport).

maybe put some image here

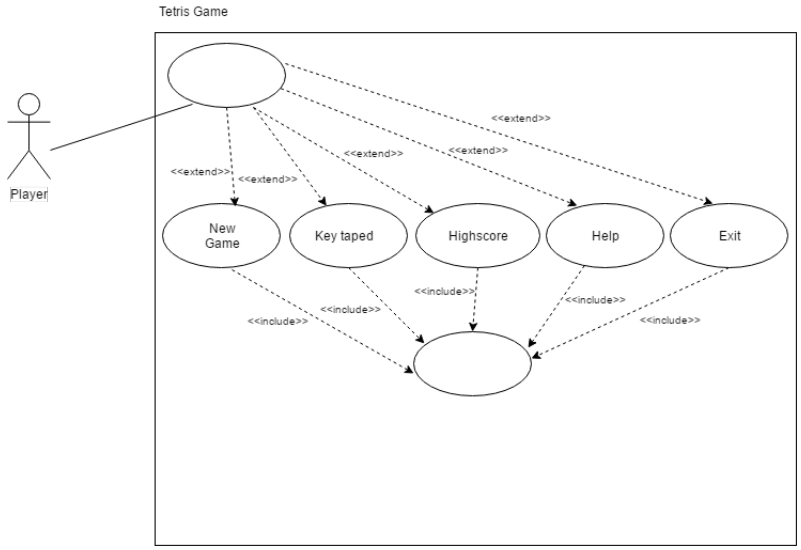
1.2 List of requirements

Id	Priority	Description
R1	Mandatory	
R2	Mandatory	
R3	Mandatory	
R4	Mandatory	
R5	Mandatory	

These requirements are directly translated onto use cases as shown next.

2 Visual UML model

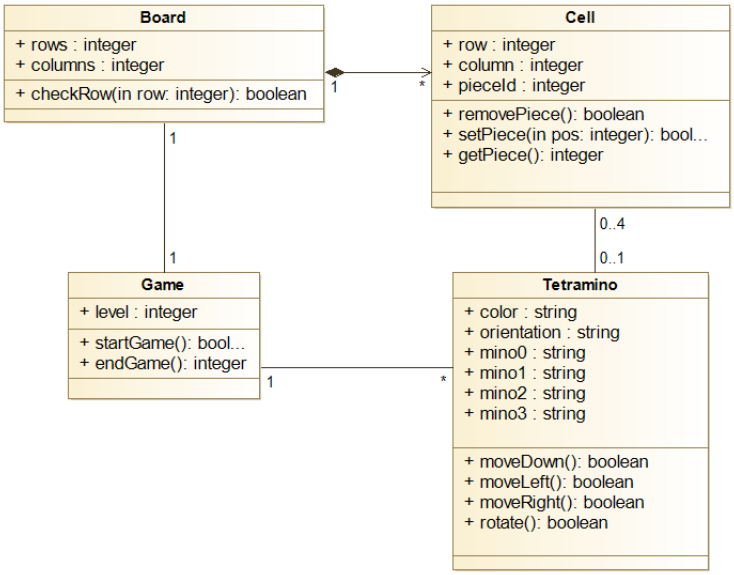
2.1 Use case model



The main use cases are described below:

Scenario	**Scenario name**
Description	
Pre-conditions	
Post-conditions	
Steps	
Exceptions	

2.2 Class model



Class	Description
Game	Core model; defines the state variables and operations available to the players.
Board	Defines a game environment for playing.
Cell	Defines the place on board where each piece of tetrominoes can stay.
Tetromino	Defines one piece to play with.

Table 1: Description of classes

3 Formal VDM++ model

todo

3.1 Class Game

todo

3.2 Class Board

todo

3.3 Class Cell

todo

3.4 Class Tetromino

todo

4 Model validation

todo

4.1 Class MyTestCase

todo

4.2 Class TestGame

todo

5 Model verification

todo

5.1 Example of domain verification

todo

5.2 Example of invariant verification

todo

6 Conclusions

The model that was developed by us covers all the requirements included implicitly on the theme project. In the final and after model verifications, we all see the game developed in VDM++ like one of the projects more consistent and safer that we have ever developed. In addition it is noticed that all the elements of the group have already had contact in the past with the game and they continue to enthuse now with this version developed by us. Maybe in the future we can all add more features to this. This project took approximately 16 hours to develop.

7 References

- https://en.wikipedia.org/wiki/Tetris
- http://tetris.com/

A Source Code

maybe not necessary because of section3

B The 7 tetrominoes