Assignment 2

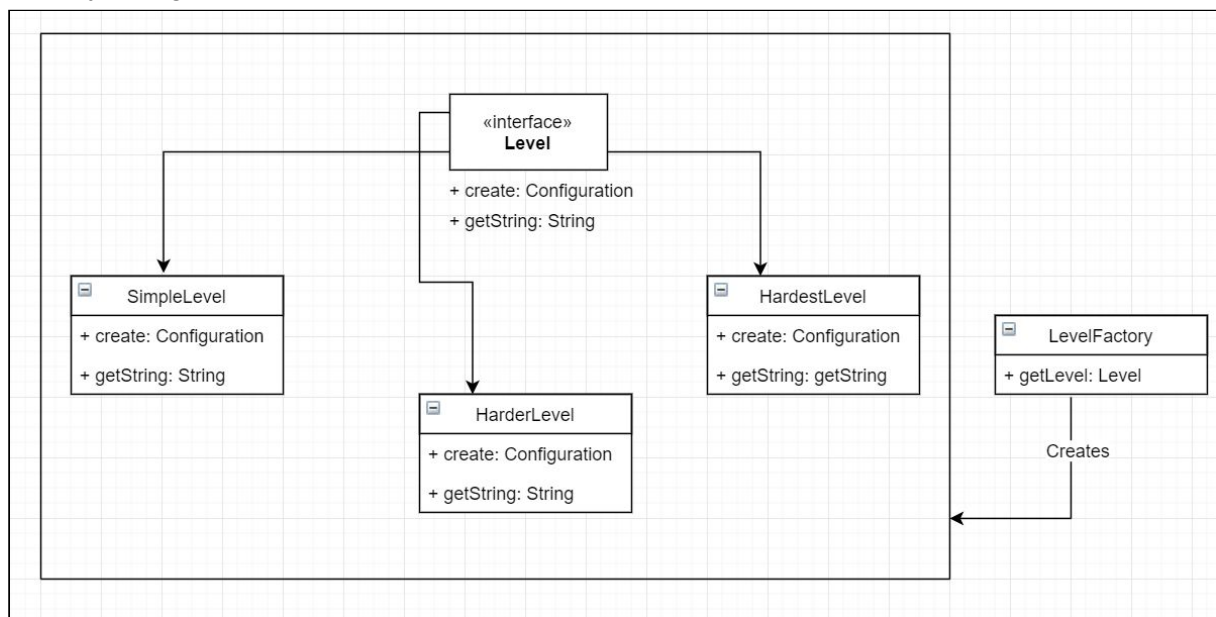TI2206: Software Engineering Methods

Exercise 1 - Design patterns

Description of design patterns

First, a decorator pattern was used in the creation of an one-way track and a filter track. By doing this, the needless creation of a class for every type of track is prevented. Instead of that, there is one abstract 'track', from which a horizontal, vertical, one-way and filter track can be created. This prevents a lot of code duplication.
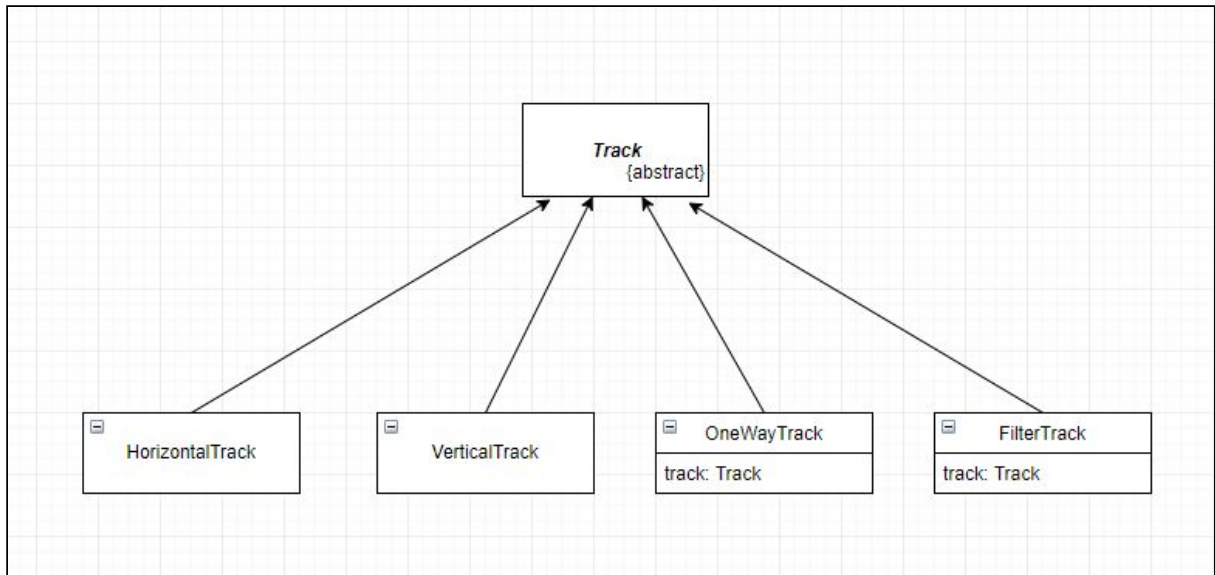
Secondly, a factory design pattern was implemented. This is done so creation of new levels is easier, as they are all derived from the interface level. Three levels have been created (to start with) deriving the methods from interface 'Level'.
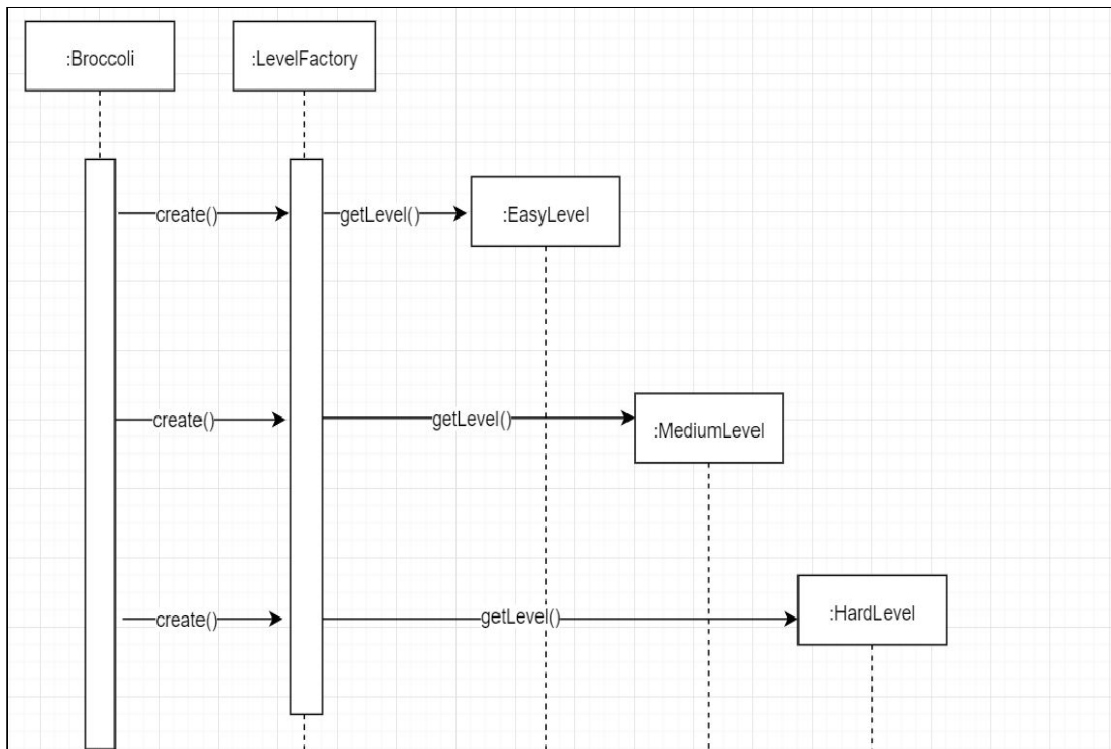
Class Diagrams
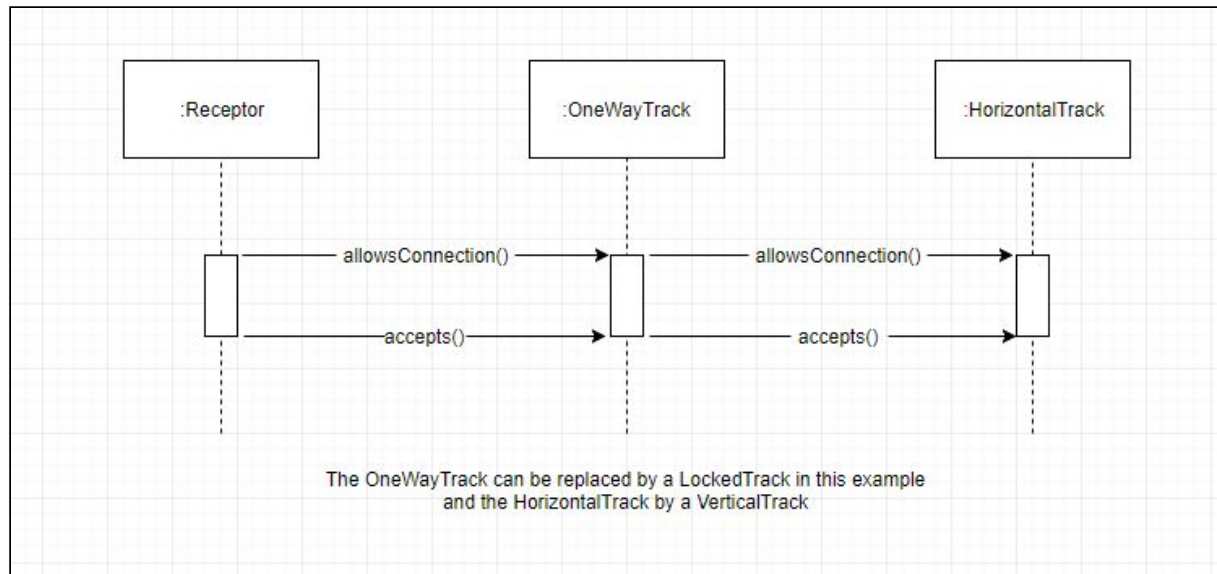
Factory Design Pattern

Decorator Design Pattern


Sequence Diagrams

Factory Design Pattern

Decorator Design Pattern

The OneWayTrack can be replaced by a LockedTrack in this example
and the HorizontalTrack by a VerticalTrack

Exercise 2 - Your wish is my command

Gudeballs power-up requirements.:

- The visual indicator shall indicate that some disc may give a powerup, at some point in the game.
- The powerup shall be obtained by clearing a disc.
- The "powerup" state of the disc shall be timed, so that the player must hurry to obtain the power-up.
- The "powerup" state shall go away after a certain amount of time.
- A single disc shall not have more than one power-up active
- The game shall provide, after a power-up is obtained either option 1: spawn 2 jokers
- The game shall provide, after a power-up is obtained either option 2: addition bonus points
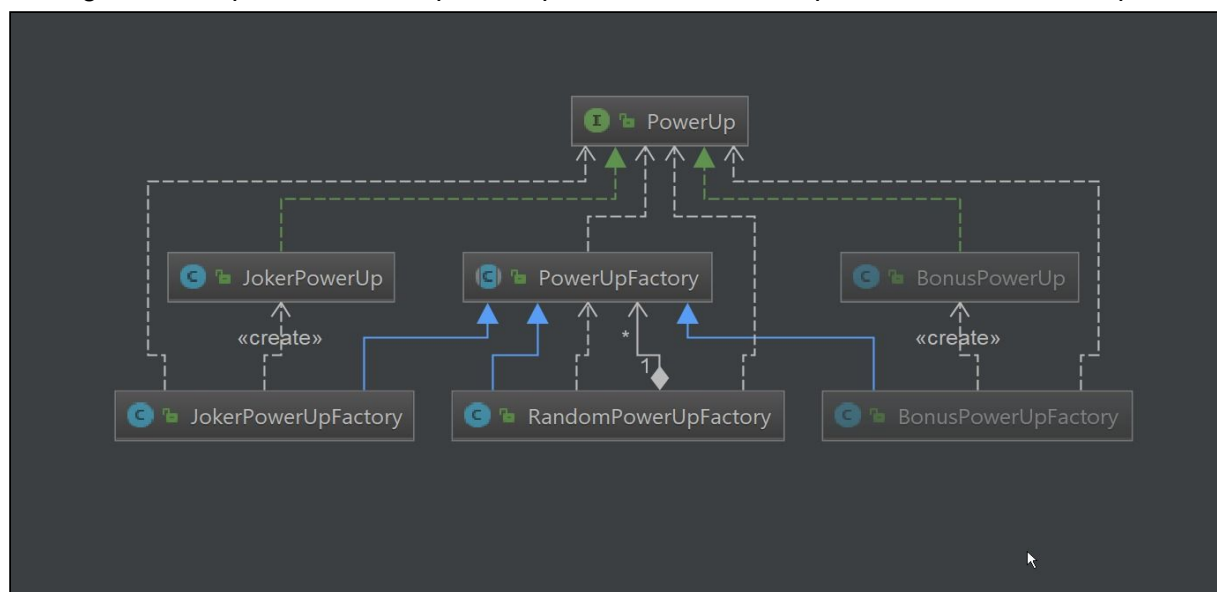
Exercise 3 - 20 time

Requirements

1. The game shall have locks, which can only be passed by balls of the same colour as the lock

2. The colour of the balls the lock shall accept shall vary

3. The joker ball shall be able to behave as any colour

5. The game shall have one-way rails

6. A ball shall be sent to the direction the arrow points to, if the ball collides with that arrow.

7. A ball shall animate when moving over the tracks

UML for this feature