



PROGRAMACIÓN ORIENTADA A OBJETOS

TRABAJO PRÁCTICO ESPECIAL

Labyrinth

Autor:

Daniel Lobo - 51171

Abstract

El objetivo de este trabajo práctico consistió en implementar una variante de un juego de tipo laberinto. El nombre mismo del juego es *Labyrinth* y fue desarrollado en el lenguaje Java.

November 2, 2014

Contents

1	Introduccion	2
2	Decisiones tomadas	2
3	UML	2
4	Problema encontrado	2

1 Introduccion

En este juego, se dispone de un tablero con un jugador, fuegos, cajas, cajas fuego, destino, interruptor y paredes.

La idea del juego es mover alguna de las cajas hasta donde se encuentra el switcher. Una vez puesta una de las cajas encima del switcher, aparece en el tablero la ubicacion del destino, al que *Link* (el personaje) se tiene que dirigir para conseguir su espada perdida. Una vez que *Link* alcanza su espada, la partida esta ganada.

A su vez, sobre el tablero se presentan fuegos, si *Link* camina sobre alguno de ellos, pierde la partida. De todas maneras, *Link* puede mover alguna de las cajas sobre un fuego como para poder caminar sobre el lugar donde está el fuego y así acceder a otras areas del tablero.

Este trabajo cuenta con la finalidad de aplicar todos los conocimientos aprendidos en la materia, aprovechando las ventajas de un diseño orientado a objetos, como son la herencia y la reusabilidad de código. Asimismo, se pone en práctica la implementación de la interfaz gráfica mediante el uso del GUI provisto por la cátedra, y el uso de la biblioteca gráfica *Swing*.

2 Decisiones tomadas

En este juego, se dispone de celdas con contenido sobre ellas. Por eso hay *Wall* y *Floor* que extienden de la clase abstracta *Cell*. De todas maneras, como las *Wall* no pueden tener contenido, no se define una variable de tipo *Content* dentro de *Cell*, se la define simplemente en *Floor*.

Se creo toda una jerarquia de packages para mantener el codigo del backend lo mas ordenado posible y claro para el desarrollador.

Se decidio hacer que la clase *Game* sea *Singleton* para mantener un mejor diseño orientado a objetos y evitar el acomplamiento de clases y funcionalidades en el backend de este proyecto.

3 UML

Adjunto en la carpeta *uml* de este repositorio se encuentran algunos graficos de UML que contienen la relaciones de las clases principales del backend, para un mejor entendimiento.

4 Problema encontrado

Durante el desarrollo y testeo de la aplicacion, se noto que cuando se inicializa desde el *Eclipse* en *Linux*, puede ocurrir que no se muestre nada en distintos

intentos seguidos. Probando esta falla en *Windows*, no se encontro tal problema y la aplicacion corre normalmente.

Si bien se ha hecho una larga investigacion en distinto foros de la web, no se logro hasta el momento encontrar ni el origen ni la solucion del mencionado problema.