

Trabajo Práctico N°2 - AlgoCraft

[75.07/95.02] Algoritmos y Programación III

Curso 1

Primer Cuatrimestre 2019

Grupo “T1”

|  |  |
| --- | --- |
| Alumno | Padrón |
| ARRUA, Rocío Ayelén | 100960 |
| LEDESMA, Dylan | 102876 |
| SAYOS, Alberto Daniel | 101050 |
| VILCA VARGAS, James Joseph | 97240 |

Contenido

[1. Introducción 3](#_Toc12548684)

[2. Supuestos 3](#_Toc12548685)

[3. Diagramas de clase 4](#_Toc12548686)

[4. Diagramas de secuencia 7](#_Toc12548687)

[5. Diagramas de paquete 8](#_Toc12548688)

[6. Diagramas de estado 10](#_Toc12548689)

[7. Detalles de implementación 10](#_Toc12548690)

[8. Excepciones 11](#_Toc12548691)

# Introducción

El presente informe reúne la documentación de la solución del trabajo práctico 2 de la materia Algoritmos y Programación III que consiste en desarrollar en Java una versión 2D del juego “Minecraft”, incluyendo una interfaz gráfica basada en JavaFX, utilizando los conceptos del paradigma de la orientación a objetos vistos hasta ahora en el curso.

# Supuestos

2.1 Herramientas  
\*PicoFino puede reducir la durabilidad de los Diamantes y Piedra  
\*Sólo el PicoDePiedra reduce la durabilidad del Metal

2.2 Materiales  
\*Sólo las hachas reducen la durabilidad de la Madera  
\*Sólo los picos reducen la durabilidad de la Piedra

2.3 Jugador

\*El Jugador sólo puede romper materiales que se encuentren a su izquierda, derecha, arriba o debajo de su posición

2.4 Mapa  
\*El Mapa es una matriz de 22x13 posiciones

# Diagramas de clase

Imagen que contiene captura de pantalla, texto

Descripción generada con confianza alta

Imagen que contiene captura de pantalla

Descripción generada con confianza muy alta

Imagen que contiene captura de pantalla

Descripción generada con confianza muy alta

Imagen que contiene captura de pantalla

Descripción generada con confianza muy alta

# Diagramas de secuencia

Imagen que contiene captura de pantalla

Descripción generada con confianza muy alta

Ilustración 1:Hacha es usada contra el material Madera, el cual puede romper

Imagen que contiene captura de pantalla

Descripción generada con confianza muy alta

Ilustración 2: PicoFino es usado contra el material Diamante, el cual puede romper

Imagen que contiene captura de pantalla

Descripción generada con confianza muy alta

# Diagramas de paquete

Imagen que contiene captura de pantalla

Descripción generada con confianza muy alta

Ilustración 3: diagrama de paquetes involucrados en Juego

Imagen que contiene captura de pantalla

Descripción generada con confianza muy alta

Ilustración 4: Diagrama de paquetes involucrados con Materiales y Herramientas

# Diagramas de estado

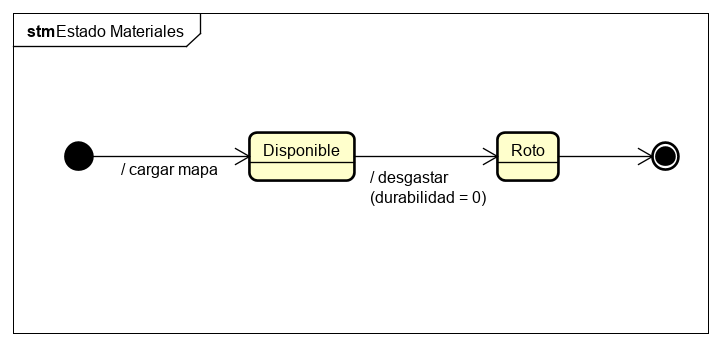


Ilustración 5: estados de los Materiales a lo largo del juego

# Detalles de implementación

Patrones:

Utilizamos DOUBLEDISPATCH y MVC.

*Doubledispatch*: para usar Herramienta contra Material (método *usar* de Herramienta).

*MVC*: para ordenamiento de paquetes básicos.

Clase ControladorDelJuego:

Se encarga de actualizar la vista del juego; a partir del modelo, se representan los movimientos del jugador en la vista.

Al comunicarse con el Modelo (*Juego*), actualiza la vista basándose en el modelo.

A través de la comunicación con *ControladorDeInventario*, se encarga de actualizar la vista del inventario.

# Excepciones

***GolpeFueraDeRangoException***:

Utilizada para que el Jugador no pueda golpear por fuera de los límites del Mapa.

***JugarSinHerramientaEquipadaException:***

Utilizada para que el Jugador siempre tenga una Herramienta equipada (no puede golpear sin ellas). Se lanza cuando la herramienta se rompe y pasa a ser NULL.