

Memoria



GODS VS TITANS

Tower Defense



17 DE MAYO DE 2018

GRUPO 9

Francesco Andreace, Guillermo J. Gutiérrez, Eduardo Varona, Carlos Tejeda Martínez,
Antonio Gómez

Índice

Análisis de bajo nivel.... p. 2

Análisis de alto nivel.... p. 3

Notas..... p. 4

Análisis de bajo nivel

En primer lugar, para el desarrollo de *GvsT*, hemos utilizado la interfaz *StateBasedGame* (en vez del interfaz propuesto *BasicGame*), porque hemos considerado que era más sencillo de implementar teniendo en cuenta las características y mecánicas del juego. La estrategia de desarrollo que adoptamos es la de creación de clases evolutiva, ya que el proyecto iba creciendo exponencialmente en el proceso de desarrollo, y por tanto ha sido necesario añadir modificaciones estructurales en la jerarquía de clases. El método básico que hemos utilizado en el desarrollo del proyecto ha sido una implementación basada en capas. Primero, implementamos los paquetes de las torres y los súbditos, porque confiábamos en que era lo más sencillo de desarrollar y la base sobre la cual implementar el resto de funcionalidades. Una vez establecida esta base, el método de trabajo se ha basado en ensayo y error probando distintas las funcionalidades necesarias el futuro desarrollo del videojuego.

Segundamente, hemos organizado la jerarquía de clases según la relación de métodos y campos dando como resultado la siguiente clasificación: un paquete clases de súbditos (*Subdito / Subditos.java*), uno de clases torres (*towers / Towers.java*), otra de mapas (*map / Mapa.java*), otra de tiles (*grid / Tile.java*) y una última para el funcionamiento del juego (*game / Game.java*) (en la cual se incluye la función *main*). Hemos utilizado herencia para crear los distintos tipos de súbditos y de torres mediante patrones de diseño *Abstrac Factory* y *Factory*.

Finalmente, para implementar los mapas hemos utilizado una estructura de datos array que indica el punto de inicio, los puntos de giro, y el punto final. Sobre esta estructura de datos, a cada punto se le ha asignado un tile (según si es camino o terreno edificable) para proyectar el mapa. Si el usuario lo desea, puede cambiar dichos puntos a su gusto siempre que tengan una cierta coherencia y formen un camino con principio y final modificando el archivo .txt en el paquete *mapSaves* el cual almacena la estructura array.

Análisis de Alto nivel

Manual de uso

- I. Ejecuta el programa.
- II. Una vez se haya abierto el menú de inicio, pulsa el botón *Start*.
- III. Selecciona un mapa del menú de mapas.
- IV. Se iniciará automáticamente la partida, en el mapa que hayas seleccionado.
- V. En la barra inferior encontrarás los indicadores de vida, oro, número de oleada y el botón de *Start Wave* (EL CUAL DEVERÁS PULSAR PARA LANZAR LA SIGUIENTE WAVE).
- VI. En la parte superior de la derecha, encontrarás el menú de construcción con los diferentes tipos de torres y los correspondientes botones de *upgrade*(para mejorar las torres) y *reventa*(para vender las torres) . En la esquina inferior derecha también encontrarás el botón *Exit*.
- VII. **Construcción:** Para construir una torre solo tienes que seleccionar el tipo de torre que quieres construir en cualquier momento con el ratón y después dirigir el puntero al lugar que desees colocar la torre. En el camino no se puede construir.
- VIII. **Mejora y reventa:** para llevar a cabo cualquiera de las dos acciones tienes que:
 - i. seleccionar la acción que desees realizar en el menú desplegado en la columna derecha, justo debajo de las torres.
 - ii. Hacer click encima de la torre sobre la que quieras ejecutar la acción.

Mecánicas del juego

Básicamente, GvsT v1.0 es un juego de tipo *tower defence* arcade, cuyo objetivo es resistir el mayor número de oleadas de súbditos en el mapa de juego. Para ello, tendrás que armar tus defensas mediante la construcción y mejora de diferentes tipos de torres (Torre Arquera y Torre Bombardera). Estas torres se compran con oro que lo generan los súbditos.

Notas

Para el desarrollo del juego nos hemos basado en tutoriales de Youtube y por una cuestión de economización del tiempo hemos reutilizado código de un juego, también de tipo TD, encontrado en GitHub*.

Por una cuestión de falta de tiempo no nos ha sido posible implementar las funciones siguientes:

- I. Más tipos de torres y de súbditos
- II. Más mapas y mapas más elaborados con *Tiles* (Tuvimos problemas a la hora de implementar las rutas de los NPCs)*
- III. Distintos efectos de sonido y animación*
- IV. Una GUI más elaborada e intuitiva*
- V. Un menú más elaborado con un apartado de opciones
- VI. Una opción de guardado de partida

*Presente en el proyecto de GitHub.