

Universidad de Buenos Aires
Facultad de Ingeniería



96.08- Taller de programación 1

1^{er} Cuatrimestre de 2018

Trabajo Práctico Final

Manual de usuario

Grupo 2

Bourbon Navarro, Rodrigo- #99999

Gómez Peter, Federico- #96091

Lidjens, Axel- #99999

Índice

1. Introducción	2
2. Instalación	2
2.1. Requisitos de software	2
2.2. Requisitos de instalación	2
3. Configuración	2
4. Forma de uso	3
4.1. Servidor	3
4.2. Cliente	3
4.2.1. Controles	4
4.2.2. Empezar una partida	5
4.3. Editor	8

1. Introducción

En el siguiente manual se detallarán los requerimientos para poder jugar, instrucciones de instalación e instrucciones sobre como jugar al *Worms*.

2. Instalación

2.1. Requisitos de software

El juego se probó en distintos sistemas operativos linux, de los cuales se destacan *deepin 15.5 64bits* y *Ubuntu 16.04 64bits*. Como requisitos para poder compilar el cliente, servidor y el editor se encuentran:

- SDL2
- SDL2 image
- SDL2 ttf
- SDL2 mixer
- Qt-5

Además, se utilizan las librerías *Box2d* como motor físico del juego y *jbeder-yaml-cpp* para las comunicaciones y la configuración del mismo, que se encuentran incluidas en la entrega.

2.2. Requisitos de instalación

Para instalar el juego se debe preinstalar previamente las dependencias. Para esto, se debe ejecutar los comandos que se observan en el código 1 (para distribuciones debian).

Código 1: Instalación de dependencias

```
1      # installs SDL2 dependencies
2      sudo apt-get install libsdl2-dev libsdl2-image-dev libsdl2-ttf-dev libsdl2-mixer-
      dev
3      # installs Qt5 dependencies
4      sudo apt-get install qt5-default qtdeclarative5-dev
```

Una vez instaladas las dependencias, se procederá a instalar el juego. Para esto, primero se debe descrompimir el paquete. Seguido de esto, se debe abrir una consola en la carpeta recientemente creada, y ejecutar las líneas del código 2.

Código 2: Instalación del juego

```
1      # creates a build folder
2      mkdir build
3      # prepares project to compile
4      cmake ..
5      # compiles and moves binaries to /usr/bin, configuration file to /etc/Worms and
      assets to /var/Worms
6      make install
```

Una vez terminado, se pueden ejecutar los distintos binarios ejecutando por consola `worms_client`, `worms_server <port>`, y `worms_editor` para levantar el cliente, el servidor y el editor respectivamente.

3. Configuración

El juego es completamente configurable. Para esto, se requiere modificar el archivo de formato *YAML*, que se encuentra en `/etc/Worms/serverConfig.yaml`. En este archivo se pueden modificar cuestiones físicas del juego (velocidad de saltos, de caminar, fuerza del viento) como cuestiones del juego (tiempo de turnos, tiempo para mover el

worm después de haber disparado, tiempo de espera hasta el siguiente turno, daño de las distintas armas, si las balas son afectadas por el viento, etc).

Para configurar otros aspectos, como lo puede ser el mapa, las municiones de cada arma o la vida de los gusanos, se puede utilizar el editor.

4. Forma de uso

4.1. Servidor

Para levantar un servidor, se requiere que se ejecute la siguiente linea:

Código 3: Instalación de dependencias

```
1 worms_server <PORT>
```

Donde el puerto debe ser mayor a 1000 si es que no se quiere ejecutar el comando como **sudo**. Además, si se quiere realizar cambios en la configuración del juego, es necesario reiniciar le servidor, para que este vuelva a leer el archivo correspondiente y cargar los datos necesarios.

El archivo **serverConfig.yaml**, que se encuentra en **/etc/Worms**, tiene la información para configurar la experiencia de juego. En este archivo se pueden modificar parámetros como los tiempos de los turnos (*turnTime*, *extraTurnTime*, *waitForNextTurnTime*, *teleportTime*), como también la distancia de caída que puede realizar el gusano sin recibir daño (*safeFallDistance*), su daño maximo, etc. También se pueden modificar parámetros de las armas, como lo pueden ser el daño de esta, el radio del impacto, el factor de amortiguamiento de las explosiones (cuanto impulso se le asigna al gusano impactado), el tiempo inicial para que explote, la fricción y rebote de la bala (efecto que se apreciará si es un arma que no explota al primer contacto), etc.

Finalmente, para realizar un cierre con su liberación ordenada de los recursos, le usuario solo debe tipear la tecla **q** en la consola que se haya ejecutado el servidor. En la figura 1 se puede ver como se cierra dicho ejecutable.

```
14:04:43 X fedemgp:(bigMerge2)~/CLionProjects/4Worms/build$ worms_server 1052
Se bindeo
hubo una conexión
hubo una conexión
Worms::Game::inputWorker:Conexión cerrada.
Worms::Game::inputWorker:Conexión cerrada.
q
14:14:47 ✓ fedemgp:(bigMerge2)~/CLionProjects/4Worms/build$
```

Figura 1: Cierre ordenado del servidor

4.2. Cliente

En la figura 2 puede verse un cliente en pleno juego. En este pueden verse distintas cosas que merecen ser destacadas:

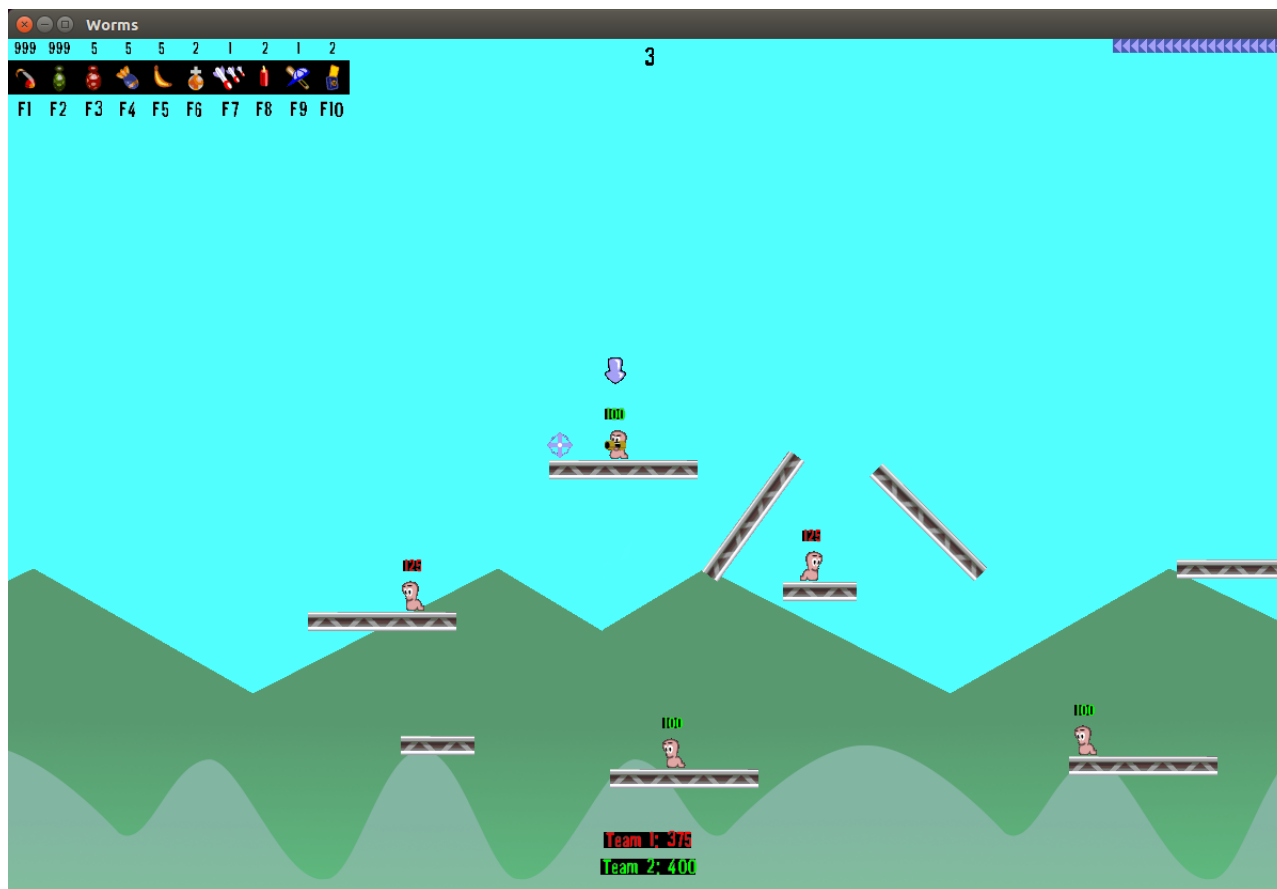


Figura 2: Captura de una partida del Worms

- En la esquina superior izquierda, puede verse el arsenal, en este se informa al usuario las armas disponibles, que botón se debe teclear para usar dicha arma (abajo del ícono), y por encima del ícono, la cantidad de municiones que posee dicha arma. Cuando el contador llega a cero, el equipo no podrá usar mas esa arma.
- En la esquina superior derecha, puede verse una barra con flechas. Esa barra indica la intensidad del viento. Cuando mas larga la barra, mas fuerte será el viento. Además la flecha indica la dirección de este. Hay ciertas armas cuyas balas son afectadas por el viento. La intensidad de este y si la bala de un determinado arma es afectado por el viento puede editarse en el archivo mencionado previamente en la sección 3.
- Arriba al centro puede ver un contador. Este marca el tiempo de cada turno. Cuando un turno termina, se cuenta un periodo extra, donde se indica quien tiene el siguiente turno. El usuario, durante este período, no podrá moverse. Por default este período es de 3 segundos. Finalmente, cuando un gusano dispara, también se empieza a contar un tiempo determinado en donde se le permite al usuario moverse momentaneamente luego de disparar. Tanto el tiempo del turno, como el tiempo de espera al siguiente turno y el tiempo extra luego de disparar, puede configurarse en el archivo mencionado en la sección 3.
- En la zona inferior de la ventana, se observa la vida total de cada equipo. Es la sumatoria de las vidas individuales de cada gusano. Cuando ese contador llega a cero, ese equipo pierde.

4.2.1. Controles

Para poder jugar al *Worms*, el usuario puede usar las siguientes teclas:

- **Espacio:** presionando dicha tecla, el worm comienza a cargar potencia para que la bala alcance una distancia mayor. Al soltar la barra, o al llegar al máximo de potencia, el disparo se realiza. Dicha tecla no tiene efecto en armas teledirigidas como la teleportación o el ataque aereo.
- **Enter:** realiza un salto hacia delante. La velocidad de este salto puede cambiarse en el archivo de configuración.

- **Retroceso:** realiza un salto mortal hacia atrás.
- **F1 - F10:** cambia el arma seleccionada de acuerdo al arsenal mencionado previamente.
- **1-5:** Cambia el tiempo de explosión de las distintas grandas (granada verde, cluster, holy, dinamita).¹
- **Flechas:** permiten mover al worm de izquierda a derecha, y con las flechas arriba y abajo se puede cambiar el ángulo de disparo de las distintas armas (si estas lo permiten).
- **Click izquierdo:** util para armas teledirigidas. Selecciona el punto a donde deben ir dichas herramientas / armas.

4.2.2. Empezar una partida

Para comenzar una partida, primero es necesario abrir un cliente. Esto se logra ejecutando desde consola el comando `worms_client`. Luego, se debe conectar al servidor, para esto hay que ingresar el HOST y el PORT en la pantalla que se muestra en la figura 3.

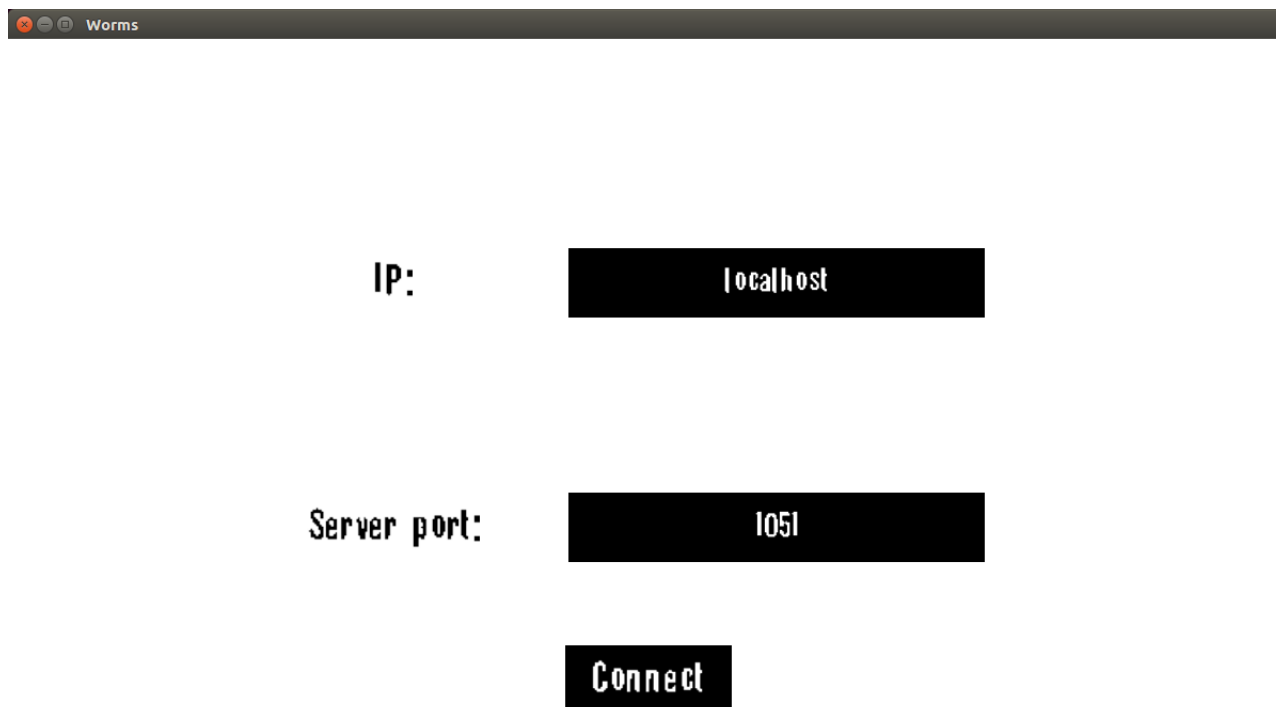


Figura 3: Pantalla de conexión al servidor

Una vez conectado al servidor, este te da la opción de crear una partida nueva (figura 4), o de unirse a alguna ya existente (figura 5). Para la creación, se puede observar que cada nivel tiene prefijado la cantidad de jugadores que pueden conectarse a dicho nivel. Por otro lado, para unirse a una partida, solo se podrá hacerlo en aquellas que no estén completas.

¹Vale la aclaración que no hay una notificación evidente cuando se cambia el tiempo de explosión. Por cuestiones de tiempo y prioridades se dejó ese feature para mas adelante. Sin embargo el timeout funciona.

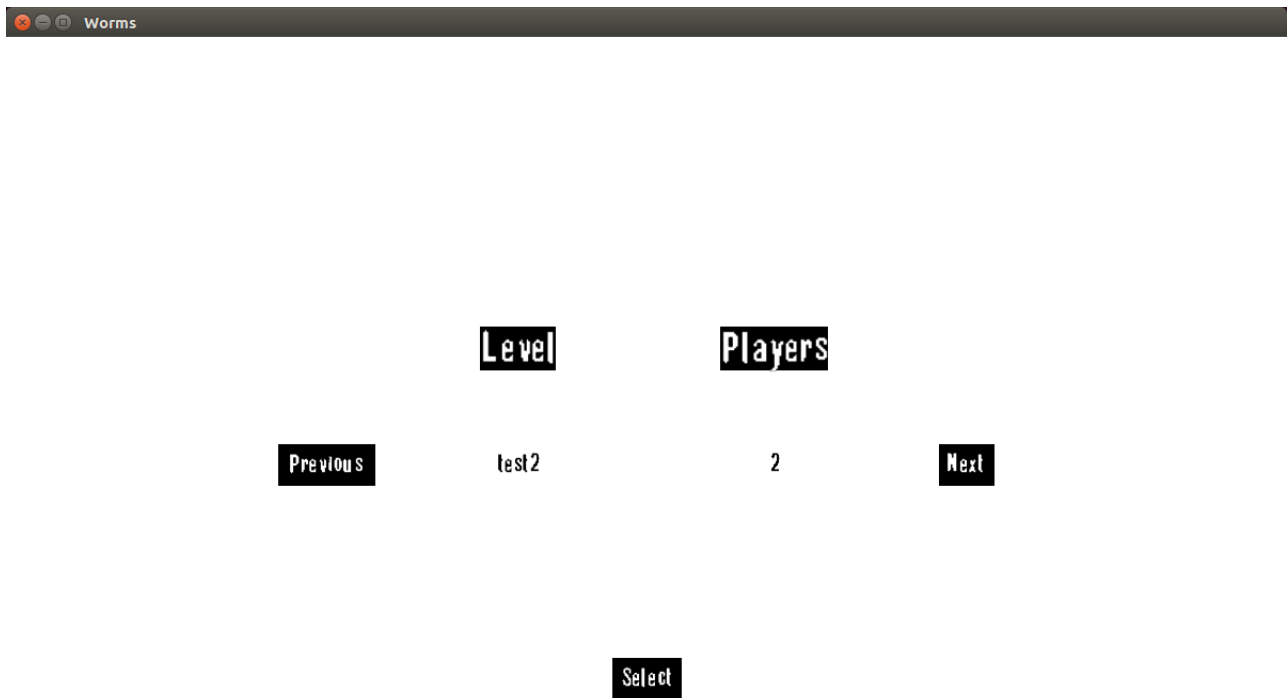


Figura 4: Pantalla de elección de nivel para crear una partida.

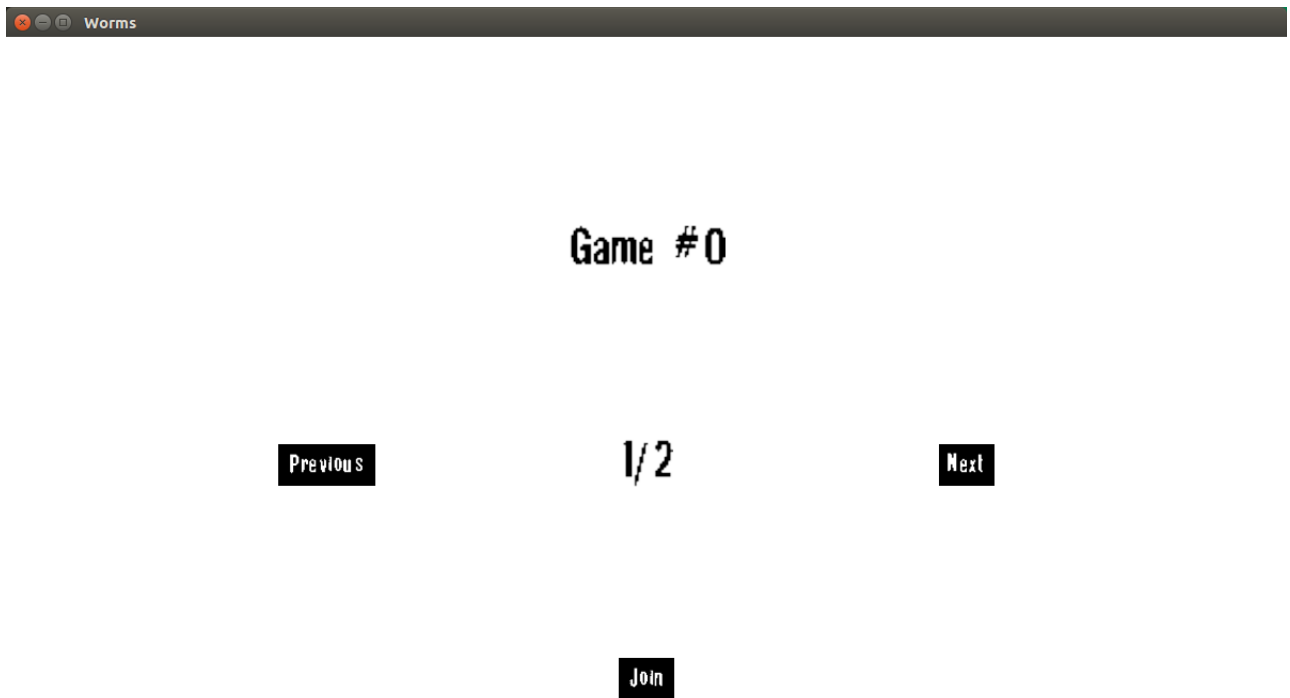


Figura 5: Pantalla de elección de sala para unirse a una partida.

Una vez finalizada la partida, aparecerá una imagen de victoria/derrota dependiendo de tu desempeño (figura 6).



Figura 6: Pantalla de victoria.

4.3. Editor

El editor puede abrirse ejecutando desde una consola el comando `worms_editor`. En este se puede crear mapas, con dos tipos de vigas, a las cuales se las puede rotar, y se pueden colocar una cantidad determinada de gusanos. Además se pueden ajustar parámetros del juego como la vida de los worms, la cantidad de jugadores que permite este mapa, la duración del turno, y las municiones de cada arma. A continuación se listarán los distintos controles que tiene el editor:

- **Click izquierdo:** Coloca una viga o un gusano en la zona seleccionada.
- **Click derecho:** Remueve una viga o un gusano.
- **Teclas + y -:** rotan las vigas.
- **W:** shortcut para posicionar gusanos.
- **L:** shortcut para posicionar vigas largas.
- **S:** shortcut para posicionar vigas cortas.

En la barra de herramientas, en la zona superior izquierda, se puede seleccionar backgrounds para el juego (hasta 3). Hay algunos predefinidos en `/var/Worms/assets/img/background`. Finalmente, se puede exportar el mapa en formato YAML en la sección `File -> export`. Para poder utilizar dicho mapa, debe copiarse el archivo creado en la dirección `/var/Worms/res/<folder>/`, donde `folder` es una carpeta con nombre a elección. Es necesario también reiniciar el servidor para que cargue dicho mapa. En la figura 7 puede verse una imagen de un mapa creado.

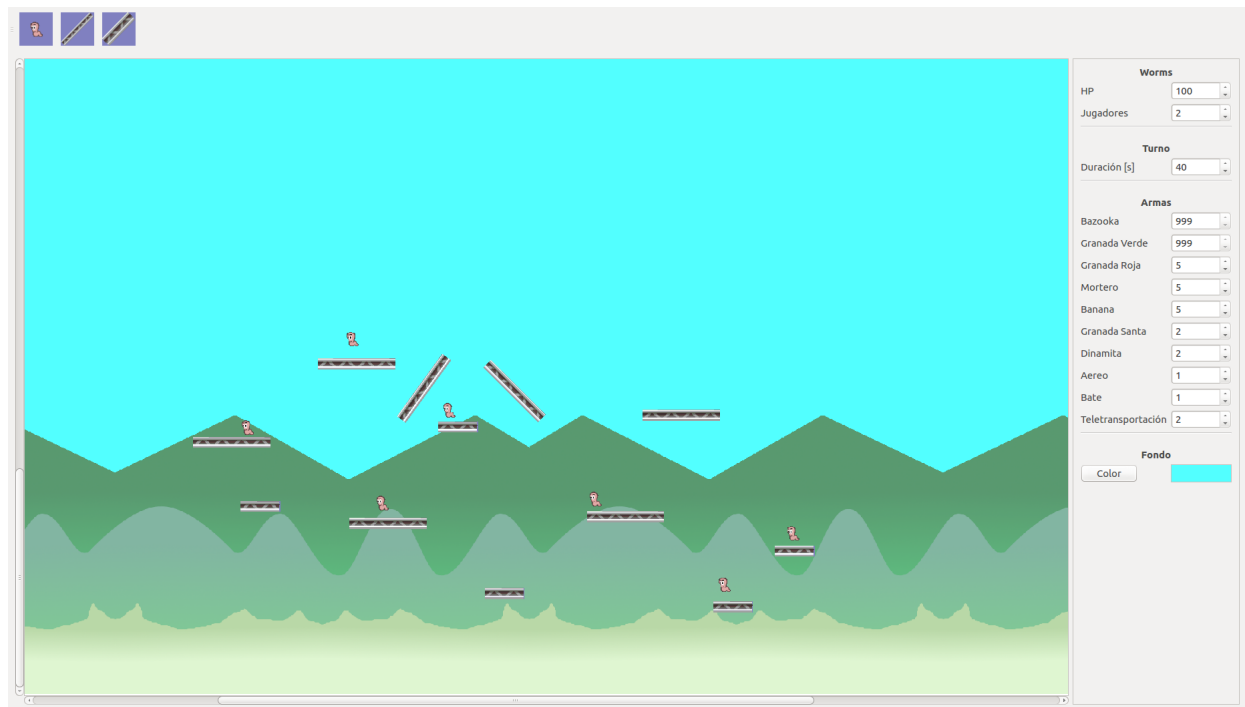


Figura 7: Vista de un mapa terminado.