



## **Tower Defence**

Proyecto

Debido a la gran popularidad alcanzada por la serie “Stranger Things”, la cadena Netflix está en vistas de lanzar un juego basado en dicha serie. Para esto, planean desarrollar un juego del tipo “tower defense” en el cual el jugador debe defender la humanidad de

# STRANGER — THINGS —

los Demogorgons, Demodogs y el Mindflyer. Es de vital importancia para la empresa la correctitud del diseño y la implementación, ya que pretenden usar este juego como molde para próximas implementaciones. Por otra parte, ya que Netflix está asociada con múltiples cadenas y produce un sin fin de series y películas, la descripción de este juego podría servir como base para otros basados en otros temas, por ejemplo “Vikings” o “Altered Carbon”. A fines prácticos, describiremos las partes de vital importancia en el juego que usted deberá diseñar e implementar.

El objetivo del juego es defender la base, representada por uno de los márgenes del mapa. Los enemigos avanzan partiendo desde el margen opuesto y el jugador deberá interponer personajes (todos ellos estáticos) en su camino a fin de evitar que lleguen a la base e ir destruyendo los enemigos. Al ser derrotados, los enemigos entregan puntos, monedas y ocasionalmente “powerups”. El objetivo del juego es sobrevivir el mayor tiempo/niveles posible acumulando la mayor cantidad de puntos. El jugador pierde si los enemigos llegan a la base.

### El mapa

El mapa donde se juega se encuentra dividido en una grilla imaginaria de un tamaño mínimo de 10 “cuadros” a lo largo y 6 a lo ancho. Uno de sus márgenes se define como “el lugar a defender” y el otro el lugar de donde provienen los enemigos. En cada “cuadro” de esta grilla solo puede estar presente un personaje, ya sea enemigo o del jugador, un obstáculo o un premio

### El nivel

Un nivel está compuesto de un grupo definido de enemigos separados en tres tandas. Si ninguna de las tandas llegó a la base y todos los enemigos fueron destruidos, el jugador gana el nivel.

Al comenzar a jugar un nivel, el jugador dispone de un “presupuesto” en monedas las cuales puede utilizar para “comprar” personajes y objetos, los cuales colocará en el mapa para evitar la llegada de los enemigos a la base.

En cualquier momento el jugador puede “vender” un personaje y recuperar todas sus monedas, al menos que el personaje se encuentre herido, motivo por el cual se le retornarán solo la mitad de sus monedas al venderlo.

Entre hordas, pueden llegar a aparecer objetos propios del mapa en el tablero que ayuden al jugador a ralentizar a los enemigos, como ser pequeños lagos o piedras (o agujeros negros o meteoritos dependiendo del tema del juego).

## Los personajes y objetos

Todos los personajes y objetos (premios u obstáculos) del juego, tienen una determinada cantidad de puntos de vida. Al terminarse estos puntos, el objeto o personaje se destruye.

Por otra parte, los personajes (todos ellos) tienen además una fuerza de impacto y un alcance. Estos parámetros determinan cuánta vida consume al atacar y cuantos “cuadros” de alcance tienen.

Los enemigos tienen además una velocidad que determina qué tan rápido pasa de un cuadro al otro, un puntaje y un rango de cantidad de monedas, pudiendo variar dentro de él, que entregan al ser destruidos. Por su parte, los objetos y personajes del jugador pueden ocupar más de un “cuadro” a lo ancho simultáneamente.

Los objetos pueden ser de dos tipos, con vida (como podrían ser barricadas o piedras) o por tiempo, los cuales tienen vida infinita pero desaparecen después de un tiempo (como los objetos mágicos, lagos o fuego). Entre las hordas, pueden aparecer en el mapa objetos propios del mismo para ayudar al jugador.

## Los premios

Al ser destruidos, algunos de los enemigos dejarán un premio, el cual el jugador deberá “tocar” (click) para obtener.

Estos premios pueden ser de dos tipos:

- Magia temporal: Este tipo de premios otorga efectos temporales a todas o a algunas de las unidades del jugador pudiendo por ejemplo duplicar la fuerza. Uno de los efectos deberá ser una suerte de “campo de protección” que destruya al siguiente enemigo que lo toque.
- Objetos preciosos: Este tipo de premios le confiere al jugador objetos especiales los cuales puede colocar en el mapa, pudiendo estos ser bombas o barricadas. Hay que tener cuidado con las bombas porque estas afectan a todos, tanto enemigos como personajes del jugador.

Cada tanto, algún enemigo aparecerá con magia aplicada sobre el.

## Lineamientos generales de implementación y diseño

Como se ha mencionado anteriormente, usted deberá diseñar de manera completa el juego pudiendo elegir la temática que desee.

Los requerimientos establecidos para el diseño son:

- Debe respetar la dimensión mínima del mapa pudiendo ser más grande.
- Debe respetar el sistema de puntos y monedas para la compra de objetos.
- Debe respetar la disposición en tandas de los niveles (pueden ser tantas tandas como usted desee, incluso pueden variar con el nivel).
- Deberán aparecer al azar los “objetos del mapa” existiendo como mínimo 2 tipos de ellos: uno temporal (como un lago) y uno con vida (como una piedra).
- Deberán existir al menos:
  - 5 personajes del jugador con características y “precios” diferentes. Al menos uno de ellos deberá ocupar más de un espacio transversal a la dirección de movimiento de los enemigos (por ejemplo una barricada larga).
  - 6 enemigos con características diferentes.
  - 4 objetos con características y “precios” diferentes. Debiendo al menos ser dos temporales y dos con vida.
- Al menos dos de los personajes del jugador y al menos uno de los enemigos deberán tener un alcance mayor a 1 realizando disparos.
- El jugador no podrá comprar “objetos del mapa”
- Deberán existir al menos dos niveles diferentes.
- Las hordas/mapas no deberán estar “hardcodeadas” en el juego.
- Deberán existir al menos cinco premios diferentes: dos de magia temporal, uno siendo el “campo de protección” y dos de “objetos preciosos”, siendo al menos una bomba.
- Deberán poder aparecer enemigos con “campo de protección”.

## Donde conseguir recursos gráficos

Se les recomienda a los alumnos, a la hora de buscar recursos gráficos, el recurrir a los llamados “sprites” de juegos viejos. Estos son “plantillas” con todos los personajes de un juego en todas las posiciones posibles simplificando así la búsqueda de imágenes.

En los siguientes sitios web hay algunos recursos de este tipo:

- [https://www.sprites-resource.com/pc\\_computer/heroes3/](https://www.sprites-resource.com/pc_computer/heroes3/)
- <https://opengameart.org/content/tower-defense-prototyping-assets-4-monsters-some-tiles-a-background-image>

Ambos son sitios dedicados a “sprites” y recursos para juegos. Se les recomienda explorar las posibilidades y ser creativos a la hora de elegir.

Por otra parte, dado que pueden utilizarse a la hora de implementar los llamados “gif” manteniendo la animación, es posible, partiendo de las partes de los “sprites” generar gif animados en sitios como el siguiente:

- <http://gifmaker.me/>

Para ello, inicialmente deberán cortar las distintas partes del “sprite” por ejemplo usando “gimp” o “photoshop” para mantener la transparencia.

# Método de trabajo

Este proyecto se realizará en grupos de 3 alumnos. Habrá una entrega final con opción de re-entrega.

Durante todo este proyecto se utilizará un repositorio "Github" por equipo y se realizarán las entregas pertinentes a través del mismo.

Cada 15 días corridos habrá un "sprint". Un sprint, en el desarrollo de software, es una reunión corta donde los involucrados en los distintos niveles de diseño y desarrollo se juntan. En este caso, en cada sprint, el ayudante responsable de la corrección del proyecto se reunirá con los alumnos para que le informen el estado y el avance del proyecto. Estas reuniones serán de 5 minutos donde el equipo muestra sus avances y habrá 5 minutos más para que el ayudante pregunte. El objetivo de estas reuniones es ayudar al grupo a planificar mejor el desarrollo del proyecto.

Como resultado de cada una de las reuniones, los integrantes de la comisión deberán redactar una minuta (correspondiente a la plantilla) que contenga lo expuesto por el grupo en la reunión y subirla al repositorio.

A continuación se detalla una lista tentativa de los objetivos a cada sprint:

## Primer Sprint

- Diseño del juego
- Diseño UML del juego contemplando todos los aspectos generales del sistema: jugador, enemigos, obstáculos y power-up.
- Diseñar en pseudocódigo (general) la generación del mapa.
- Diseñar el movimiento del jugador y enemigos (pseudocódigo o diagrama de interacción).
- Tener pensado como interactuar entre el jugador, los enemigos y el mapa. Diseñar un diagrama de interacción borrador.
- Tener funcionando el GitHub con los archivos generados.

## Segundo Sprint

- Implementar gráficamente el mapa (sin colisiones).
- Implementar el jugador de manera básica.
- Implementar gráficamente el jugador (sin colisiones).
- Implementar un enemigo básico.
- Implementar gráficamente el enemigo (sin colisiones ni movimiento).
- Diseñar el disparo.

## Tercer Sprint

- Implementar el disparo (sin interacción).
- Incorporar patrones de diseño al juego.
- Incorporar detección colisiones al juego (implementar).
- Diseñar la interacción entre jugador/enemigos y disparos.
- Diseñar la interacción entre el jugador y los power up.

## Cuarto Sprint

- Implementar el disparo con interacción y gráficamente.
- Implementar la interacción entre jugador/enemigos y disparos.
- Implementar la interacción entre el jugador y los power-up.

## Quinto Sprint

- Implementar todos los enemigos.
- Implementar todos los power-up.
- Implementar el ganar/perder.
- Implementar todo gráficamente.

## MINUTA DE FIN DE SPRINT

Comisión TDP	
Docente TDP	
Fecha	
Sprint que se evalúa	
Objetivos cumplidos (máximo de 500 caracteres)	
Objetivos pendientes (máximo de 500 caracteres)	
Correcciones a realizar (máximo de 500 caracteres)	