



Nombre producto: Yggdras Tur

Equipo de desarrollo: Fernando Castro  
Ignacio Gajardo  
Julio León  
Katherine López  
Claudia Provoste  
Claudio Rubio

**Resumen:** Yggdras Tur es un videojuego de plataformas procedural ambientado en la mitología nórdica, donde el objetivo es subir el árbol del mundo para pedir a Odin que cese el ragnarok.

**Género(s):** Endless, Plataforma

**Plataforma(s):** Pc

Público Objetivo: 9 - 15 años, usuarios reciente acceso al computador que gusten de jugar videojuegos para navegador (referencias Subway Surfers, Temple Runner)

Temática: Subir el árbol del mundo Yggdrasil, ragnarok y el valhalla. Mitología Nórdica.

## Jugabilidad

- Mecánicas principales:
  - Salto: Es automático y aumenta la altura del jugador en x unidades
  - Cambio de (perspectiva o clima): El personaje cambia la perspectiva del mundo, lo que varía el eje de las plataformas, se realiza con el botón a y d
  - Movimiento: Se realiza presionando la flecha derecha e izquierda.
- Power Ups:
  - Hyper salto: Luego de tomar el power up, la siguiente plataforma en la que colisione el jugador obtendrá mayor impulso.
  - Escudo anti daño: Niega los siguientes tres golpes que recibe el personaje de enemigos o ciertas plataformas.
  - Warp: Agujero negro que aumenta de forma instantánea la altura a la cual se encuentra el personaje, teletransportándolo a una posición más alta en el juego.

## Flujo de Juego

### Inicio del Nivel:

El jugador inicia una plataforma, se aprecia que existen múltiples plataformas a las cuales saltar y se procede a esto, es importante explicar el objetivo del juego y las mecánicas más básicas tales como el salto y movimiento, mediante el diseño del nivel.

- **Fase Inicial:**

- Una vez comprendidas las técnicas básicas el jugador procede a subir a través de diversas plataformas, en esta fase solo encuentran los obstáculos más básicos, como son plataformas **estáticas, móviles y rompibles**.
- En esta fase aparece el primer power ups **Hyper salto**.
- Se encuentra conformada por tres niveles **Niflheim, Helheim y Jotunheim**.
- Los obstáculos se deben agregar de forma progresiva entre estos tres niveles.

- **Fase Intermedia:**

- El jugador ya se debería encontrar acostumbrado a las mecánicas, por lo que es necesario aumentar el número de obstáculos para aumentar con esto la dificultad.
- Los nuevos obstáculos a agregar son enemigos, **plataformas de fuego y rayos**, con esto se busca que los obstáculos no sean solo los saltos y la habilidad del jugador con estos, sino que además se agregan factores en externos que desafíen al mismo.
- En esta Fase aparece el power up **Escudo anti daño**
- Se encuentra conformada por cuatro niveles **Vanaheim, Nidavellir, Alfheim y Muspelheim**.
- Los obstáculos se deben agregar de forma progresiva entre estos cuatro niveles.

- **Fase avanzada:**

- Llegado a este punto el jugador ya está acostumbrado con las mecánicas de juego, razón por la cual es necesario aumentar nuevamente el desafío, para esto se aumenta de forma progresiva tanto la velocidad de movimiento de las plataformas, como la probabilidad de que éstas sean rompibles, desafiando en gran medida al jugador gracias a esto.
- En esta fase se debe agregar el ultimo power up, **Warp**.
- Se encuentra conformada por tres niveles **Midgard, Asgard y Valhalla**, siendo este último el nivel final del juego, en el cual aparece **Odin**

## Diseño de niveles

El diseño de niveles es basado en las mecánicas encontradas en el juego. Las plataformas están creadas de forma procedural, asegurándose de que ninguna partida que el jugador encuentre sea igual a la anterior e incentive a que vuelva a jugar, aumentando su *replay value*.

Los power Ups aparecen luego de que el jugador alcance cierta altura en cada nivel con tal de que no aparezcan muy seguido ni escasamente.

Los niveles y fases mencionadas anteriormente controlan la dificultad del juego, asegurando un flow seguido y dándole tiempo para que el jugador se adapte a las nuevas mecánicas que van apareciendo a lo largo de todo el juego.

## Flow de juego y condiciones de victoria

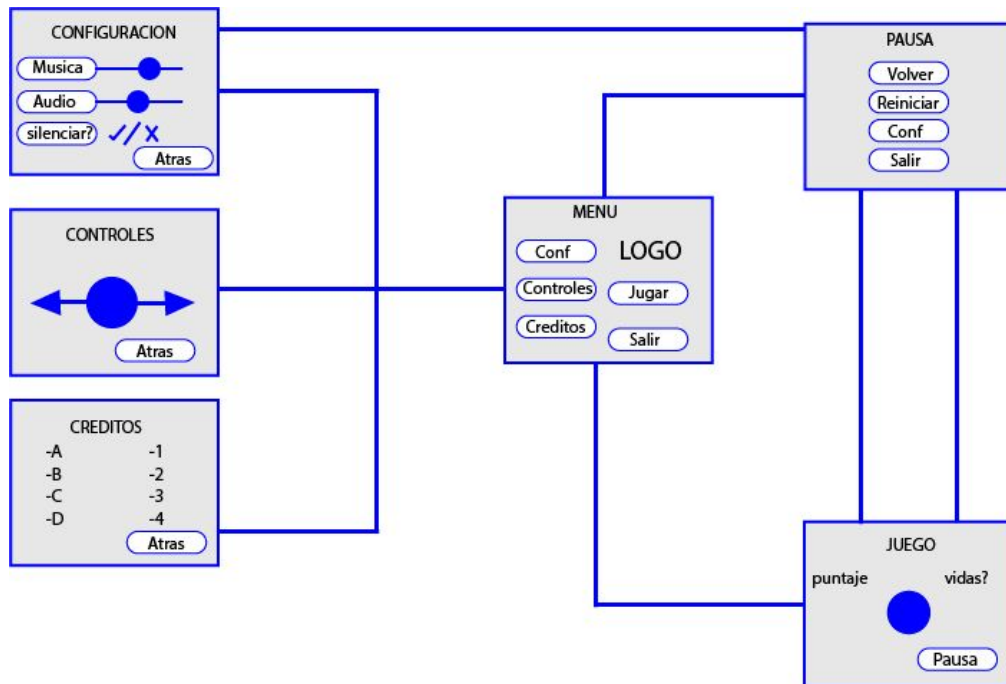
La dificultad del juego es progresiva poseyendo picos al cambiar entre las diversas fases del juego, esto debido a que en cada fase se agrega un tipo diferente de obstáculos y power ups motivando al jugador a adaptarse. Los obstáculos repartidos por fases son los siguientes:

- **Fase inicial:** pensada para acostumbrar al jugador a las mecánicas básicas.
  - Plataformas estáticas
  - Plataformas Rompibles
  - Plataformas móviles
- **Fase intermedia:** pensada para poner a prueba la habilidad del jugador.
  - Plataformas de fuego
- **Fase final:** pensada para desafiar al jugador.
  - Gran aumento velocidad plataformas
  - Gran aumento de plataformas rompibles

No existe una condición de victoria como tal, el objetivo es llegar lo más lejos posible subiendo el árbol, aún así es posible llegar a un último nivel de altura (Valhalla) donde aparecería odin, cumpliendo el objetivo del personaje sin terminar la partida.

## Menús

- Flujo de menús



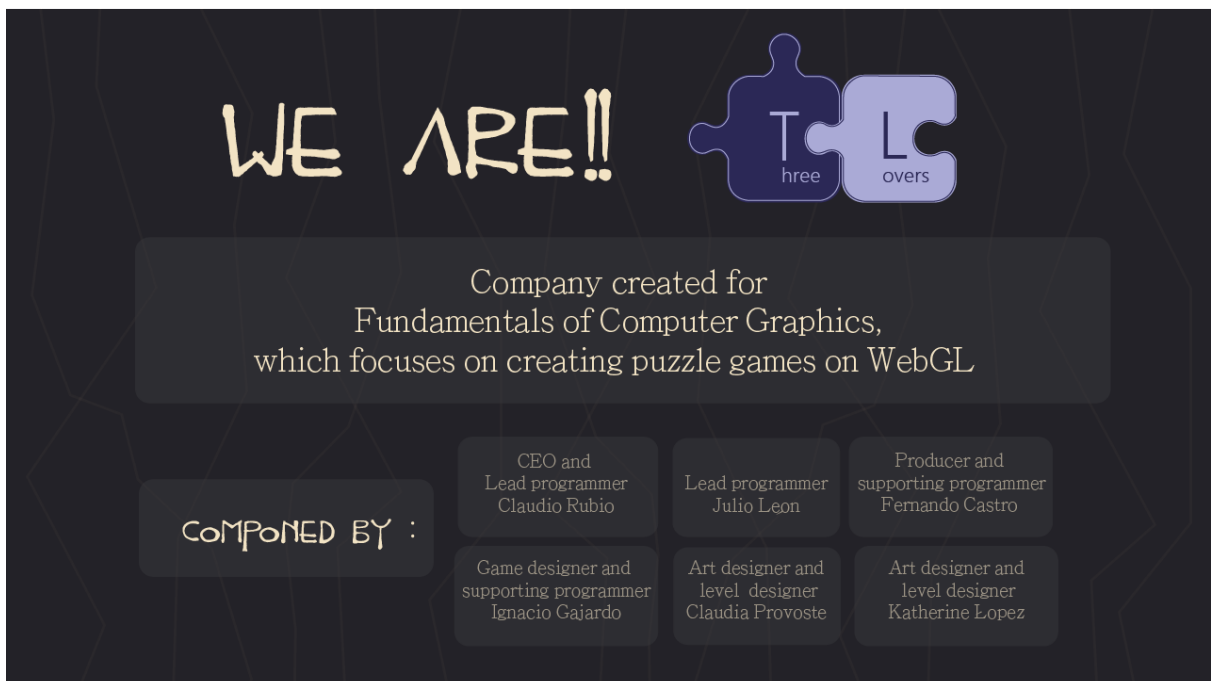
- Menú



- Instrucciones



- Créditos



## Arte

Intención visual: Dar la sensación al jugador de ir subiendo continuamente mientras está en el entorno de los 9 mundos de la mitología nórdica.

Estilo visual: Minimalista, Flat.

Estilo musical: Épica.

Referencias: En sección de referencias

Paleta de colores:



NIFLEHEIM



JOTUNHEIM



HELHEIM 1



HELHEIM 2



MIDGARD



MUSPELHEIM 1



MUSPELHEIM 2



NIDAVELLIR



VANAHEIM 1



VANAHEIM 2



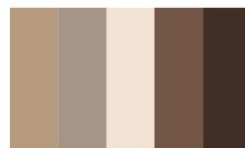
VALHALLA 1



VALHALLA 2



ASGARD 1



ASGARD 2



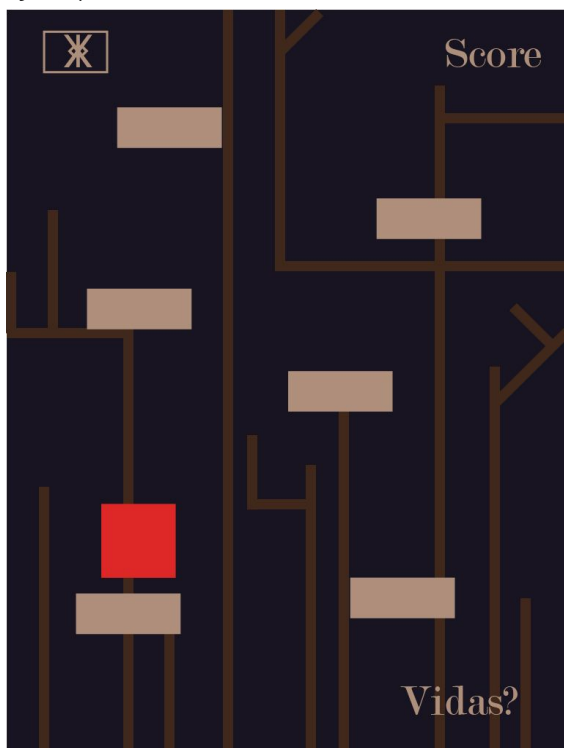
ASGARD 3

Niveles :

Distribución:

- Niflheim: desde 0-10000 unidades
  - *plataforma de piedra.*
- Jotunheim: 10000-20000
  - *plataforma de hielo cristalino.*
- Helheim: 20000-30000
  - *plataforma de piedra gris con nieve.*
- Vanaheim: desde 30000-40000 unidades
  - *plataforma de piedra con musgo.*
- Nidavellir: desde 40000-50000 unidades
  - *plataforma de hierro [herrería enana].*
- Muspelheim: desde 50000-60000 unidades
  - *plataforma de roca con fuego.*
- Midgard: desde 60000-70000 unidades
  - *plataforma de tierra.*
- Asgard: desde 70000-80000 unidades
  - *plataforma de oro.*
- Valhalla: desde 80000-90000 unidades
  - *plataforma de nube*




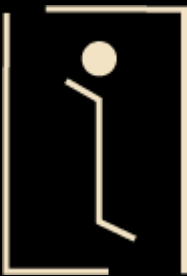



Ejemplo Front View:










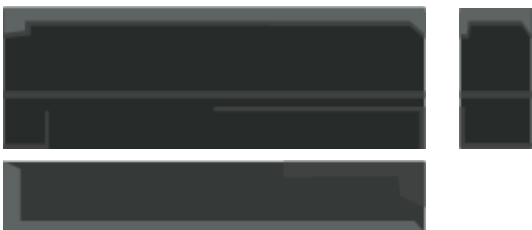
## Assets

Botones:

	Inicia el juego
	Cierra el navegador
	Permite Acceder a los créditos
	Permite acceder a las instrucciones
 	Primero representa musica encendida, el segundo representa música apagada
	Permite volver a la sección anterior del menú

## Lista de texturas:

- Plataformas:

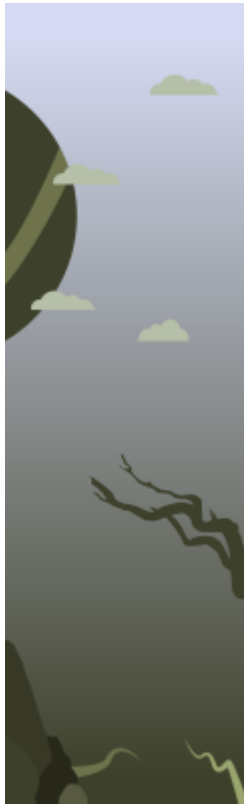
Niflheim	
Helheim	
Jotunheim	
Vanaheim	
Muspelheim	
Nidavellir	

Midgard	
Asgard	
Valhalla	
Plataforma de aviso de peligro.	
Plataforma Rompible	

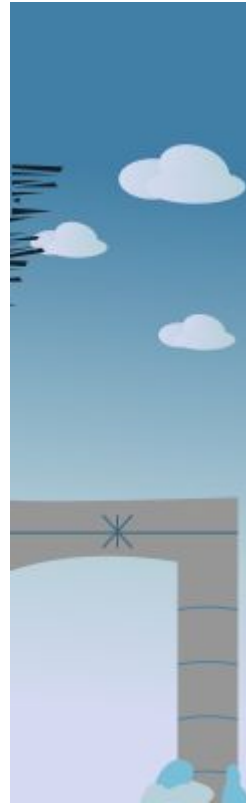
- Power Ups:

Hyper salto		
Escudo anti daño		
Warp		

- Fondos: cuatro imágenes por fondo

Niflheim	
----------	---

Jotunheim



Helheim



Vanaheim



Nidavellir



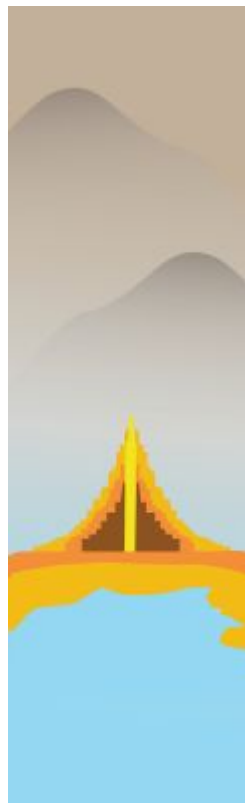
Muspelheim



Midgard



Asgard



Valhalla





Logo del juego:



## Música y sonidos

- Música:
  - Cancion de fondo:
  - Canción Para el menú:
- Efectos de Sonido:
  - Sonido de click: debe usarse al presionar cualquier botón
  - Sonido de impacto con plataforma
  - Sonido de derrota
  - Sonido al recibir daño
  - Sonido al usar power ups
    - Sonido al usar Hyper Salto
    - Sonido al conseguir Escudo anti daño:
    - Sonido al usar Warp
  - Sonido Rayo

## Complicaciones en el desarrollo

Se presentaron cinco complicaciones principales en el desarrollo del proyecto en el ámbito de la programación, la creación de un sistema de colisiones por ejes separados, el movimiento del personaje separado por ejes, la aplicación de css a la hora de crear los menús e interfaces, la acción de warp conforme a la creación de plataformas y la creación del desastre natural rayo. Las soluciones para estas problemáticas fueron:

- **Colisiones por ejes separado:** la creación de un sistema de colisiones propios que calcula si existe o no una colisión, basado en el eje a evaluar (x o z)
- **Movimiento del personaje separado por ejes:** se generaron controles basados en la dirección actual del personaje, para esto se genero un pivot que guarda las coordenadas del pj y sigue su posición en el eje y, por lo que al girar el pivot se rota la camara.
- **Uso de css:** este problema se genera debido a desconocimiento del lenguaje, ya que está era la primera vez que se trabaja con el mismo, se soluciono leyendo la documentación de css y buscando en paginas como stackoverflow respuestas para problemas específicos, tales como crear multiples botones en un panel y luego moverlos.
- **Warp en base a altura:** El power Up de warp que consiste en el cambio instantáneo de posición del personaje en Y, lo cual traía problemas al encontrarse en una altura sobre la creación de plataformas impidiendo la aparición de estas. Al inicio se pensó en disminuir la altura del warp, pero esto hizo el power up casi inservible al ser muy cercano al salto normal, por lo que su solución fue aumentar la altura en la cual se crean las plataformas en base a la altura máxima alcanzada por el jugador.
- **Rayo:** Debido a complicaciones con el uso de raycast, no fue posible crear el desastre natural rayo

En el ámbito artístico fue complicado el diseñar un entorno que refleje la perspectiva que queríamos darle al jugador con el desarrollo de los fondos, y el entorno en general para que se sintiera como si fuesen los 9 mundos. Además, hubo problemas con el mesh del personaje. La forma de desarrollo fue:

- **Perspectiva:** Habían 2 factores base a considerar, lo que estaba lejos (y por ende, estático) y lo que estaba cerca (y debía girar). Dado que el fondo ayuda al jugador a orientarse por los objetos cercanos, debía haber al menos uno por cada fondo, lo que significó 4 fondos distintos por cada mundo.
- **Inmersión:** Si bien los 9 mundos son completamente distintos debíamos lograr que se sintiera una unión entre todos; por lo que todos los

fondos cuentan con un degradado que une un mundo con el otro; haciendo así, incluso, endless.

- **Mesh:** El modelo está compuesto por la cara del personaje y un cubo por detrás para facilitar las colisiones. La unión entre la cara y el cubo aumentó la cantidad de subdivisiones, lo cual fue resuelto haciendo las caras del cubo una sola y sin divisiones.

## Supuestos a Incorporar (Lo que podría haber estado)

- Powers Ups:
  - Disminución zona de peligro
- Niveles:
  - Alfheim
  - El árbol Yggdrasil como tal para que se sintiese como si fueras subiendo.
  - Detalle en el background.
- Enemigos:
  - en un inicio se tenía la intención de agregar enemigos que se movieran y atacarán al jugador, pero por temas de tiempo no fue posible

## Especificaciones Técnicas

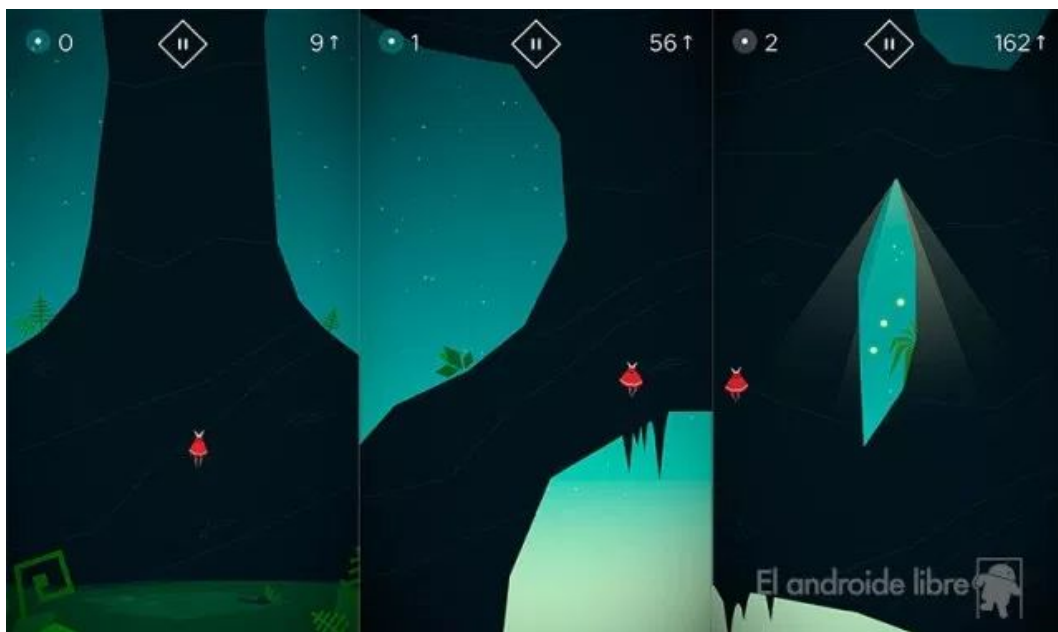
- FrameWork:
  - Three.js:
    - Usado debido a las exigencias impuestas al proyecto
- Editor de Código:
  - Note Pad++
  - Sublime 3
- Herramientas Creación de Arte Visual
  - Adobe Illustrator:
    - Versión: CC 2018
    - Uso:
      - Creación de figuras
      - Creación de fondos
      - Creación de UI
      - Creación de wireframes
  - Blender:
    - Versión: 2.79b
    - Uso:
      - Creación de modelo 3D

## Referencias

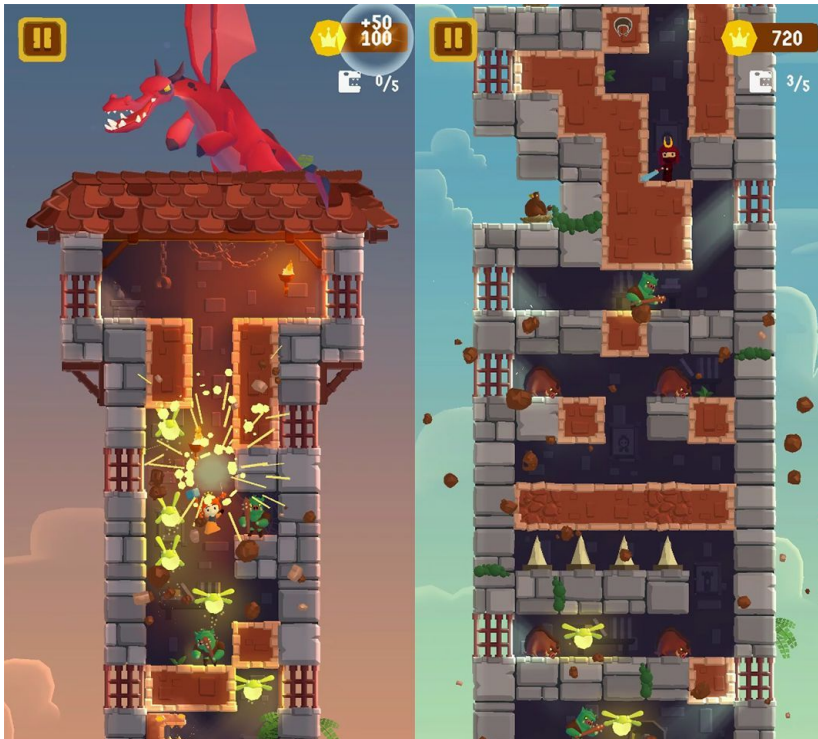
Referencias visuales:  
Plataformas



Intención visual similar



Fern Flower



Once upon a tower

Juegos similares:



Planet tower



Doodle jump