

GDD Ejemplo Aplicado

Diseño de Videojuego II: Andrés Rossi – Ramiro Rodríguez

Dragon Rush (v1.26)

1. [Dragon Rush \(GDD v1.26\)](#)
 - a. [Control de versiones](#)
 - b. [Datos Tecnicos](#)
 - c. [Descripcion del juego](#)
 - d. [0.Navegacion de escenas](#)
 - e. [1.Escena de preloader](#)
 - f. [2.Escena de Transicion](#)
 - g. [3.Escena de inicio](#)
 - h. [4. Escena de ayuda](#)
 - i. [5. Escena de juego](#)
 - i. [5.2. Estructura general de etapas](#)
 - ii. [5.3. Escena de Etapa superada](#)
 - iii. [5.6. Escena de juego perdido](#)
 - iv. [5.7. Escena de juego superado](#)
 - v. [5.8. Elementos del escenario](#)
 1. [5.8.1. Camara](#)
 2. [5.8.2. Iman de camara](#)
 3. [5.8.3. Superficie normal](#)
 4. [5.8.4. Huecos](#)
 5. [5.8.5. Vehiculo del jugador](#)
 6. [5.8.6. Efectos](#)
 7. [5.8.7. Carteles informativos.](#)
 - vi. [5.9. Filosofia de diseño de etapas](#)
 - vii. [5.10. Piezas para creacion de terreno, colisiones](#)
 - viii. [5.11. GUI](#)
 - ix. [5.12. Aparicion del Dragon](#)
 - x. [5.13. Powerups](#)
 - xi. [5.20. Bocetos etapas](#)
 - xii. [5.14. Score](#)
 - xiii. [5.15. Musica](#)
 - xiv. [5.16. Sonido](#)
 - xv. [5.17. Controles](#)

Control de versiones

Version 1.1 (27/11/2010). Editado por Andrés Rossi:

- Se cambio la forma en que se relaciona el resumen de score con las distintas animaciones de resultados (ganador, perdedor, juego superado). La animacion de nivel superado ya no es necesaria.
- Se quitaron las superficies destruibles del juego.
- Se agrego sonido de dragon masticando

Version 1.26 (10/12/2010). Editado por Andrés Rossi:

- Agregada estructura de trazabilidad
- Agregado concepto de speed powerup
- Agregado concepto de huecos en el escenario y sus consecuencias.
- Agregada filosofia de diseño de etapas.
- Modificada la forma en que se suceden las etapas y se muestra el score.
- Agregado el elemento "iman de camara" (+boceto) (opcional)
- Agregada muestra de piezas para creacion de terreno
- Agregada escena de transicion, como funciona y en que momentos aparece
- Quitada ayuda in game (cortaria el flujo de juego ya que la idea es que el jugador no se detenga)
- Cambiada la GUI por una version mas simple, que obstruye menos la vision del escenario.
- Agregado mas detalle a la descripcion de la relacion lives-extra lives (corazones rojos/verdes en la GUI)
- Agregado comportamiento del dragon cuando el vehiculo se desplaza hacia la izquierda.
- Agregados bocetos de etapas 1,2 y 3.

Datos Tecnicos

Genero de juego: Autos sidescroller con fisica.

Modos de juego: Single player.

Plataforma: Flash / Web

Target de publico: hombres entre 13 y 18 años.

Objetivo: Mantener enganchado al jugador por el mayor tiempo posible tratando de completar el juego y promover que comparta el link con la mayor cantidad posible de conocidos y amigos.

Persistencia de datos: No habrá persistencia de datos

Descripcion del juego

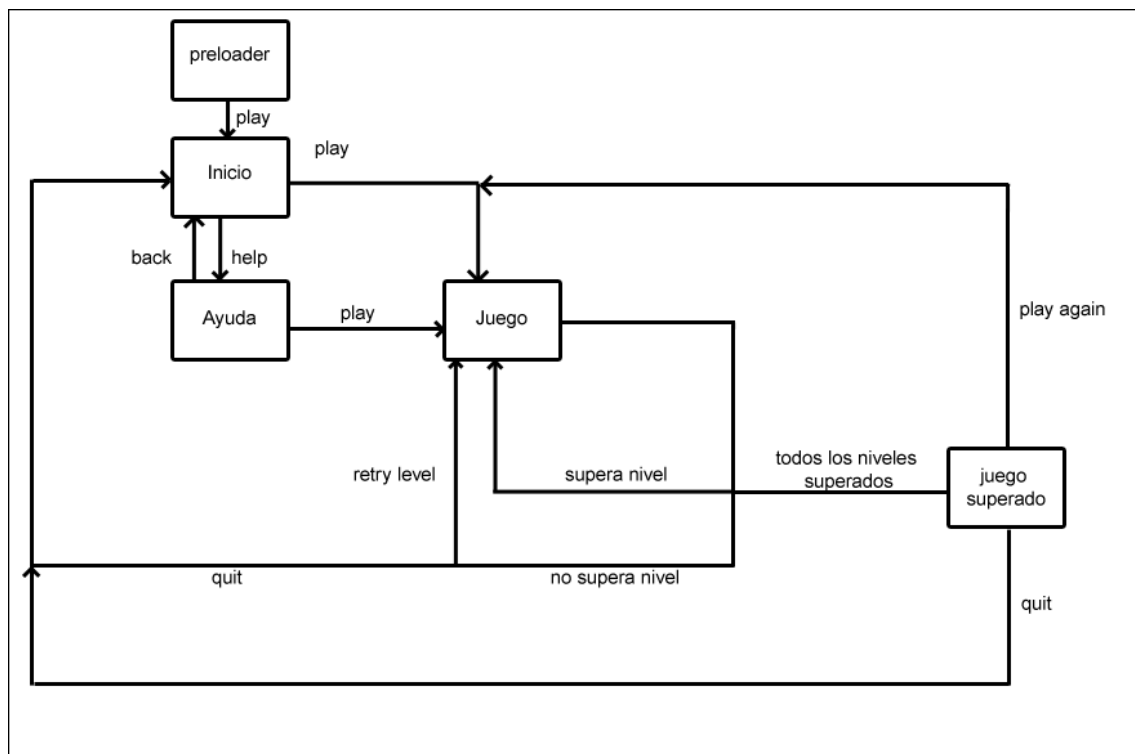
Es un juego estilo rider, desplazamiento lateral, con fisica. El jugador debe llegar al final de cada etapa conduciendo un buggy arenero por un mundo salvaje, perseguido por un dragon.

Ejemplo de un rider, sidescroller:

<http://www.pepijuegos.com/juegos-de-coches/rocky-rider>

0.Navegacion de escenas

0.1. Imagen de referencia. Se incluye un grafico de referencia de todas las escenas y sus interrelaciones, como se accede de una a otra en el juego.



1.Escena de preloader

1.1. El preloader es un minijuego, debe ser menor a 50kb. Este minijuego carga rapidamente y luego mientras se carga el juego principal el jugador se entretiene jugando este preloader.

1.2. El gameplay del preloader consiste en clikear con el mouse sobre huevos de dragon alados que cruzan volando la pantalla. Los huevos se mueven en forma ondulatoria de izquierda a derecha o de derecha a izquierda.

1.2.1. Cada vez que destruimos un huevo sumamos un punto.

1.2.1.1. Sonido de Disparo fallado al errar un tiro.

1.2.1.2. Sonido de disparo dado al destruir un huevo.

1.3. La GUI del minijuego consiste en el icono de un huevo de dragon, un signo "x" a su lado y un numero dinamico que representa cuantos huevos han sido destruidos.

1.3.1 Mientras el juego principal esta cargando: **"Dragon Rush is loading. Shoot the eggs while you wait"**

1.3.2. Cuando el juego principal termina de cargar: **"Dragon Rush is READY."**

1.3.3. Cuando el juego principal termina de cargar: Boton **"PLAY"**. Click lleva al jugador a la escena de inicio mediante la escena de transicion.

1.3.3.1. Este boton utiliza el sonido de click al ser presionado.

2.Escena de Transicion

2.1. Es una animacion que representa la cola del dragon "limpiando" la pantalla como si se tratara de un limpiaparabrisas. Esta animacion incluye mascaras y sirve para pasar de una escena a la otra de una manera esteticamente mas agradable.

3.Escena de inicio

3.1. La escena de inicio muestra una animacion del vehiculo huyendo de un gran dragon.

3.1.1. Las ruedas del vehiculo giran para dar la impresion de movimiento.

3.1.2. El dragon esta volando y moviendose levemente arriba y abajo.

3.1.3. El resto del escenario se desplaza (loop) de derecha a izquierda para generar la ilusion de que el dragon y el vehiculo se mueven hacia la derecha.

3.2. En esta escena vemos dos botones: **PLAY** y **HELP**.

3.2.1. **PLAY:** permite acceder a la escena de juego (mediante la escena de transicion)

3.2.1.1. Este boton utiliza el sonido de click al ser presionado.

3.2.2. **HELP:** Permite acceder a la escena de ayuda (sin transicion, en este caso la escena de help es una caja flotante que hace un fade in para aparecer y fade out para desaparecer)

3.2.2.1. Este boton utiliza el sonido de click al ser presionado.

3.3. La musica de la escena de inicio es un loop del leitmotiv de Dragon rush en clave lenta.

3.4. Imagen de referencia.



4. Escena de ayuda

4.1. La escena de ayuda muestra un pequeño resumen ilustrado de los controles, objetivo del juego y powerups disponibles.

4.1.1. Se trata de una caja con contenido que aparece sobre la escena desde la cual es llamada (en el ejemplo se encuentra sobre la escena de inicio).

4.2. En esta escena vemos dos botones: **BACK** y **PLAY**

4.2.1. **BACK:** Permite regresar a la escena de inicio (la escena de help es una caja flotante sobre la escena de cover y hace un fade out para desaparecer)

4.2.1.1. Este boton utiliza el sonido de click al ser presionado.

4.2.2. **PLAY:** Permite acceder a la escena de juego (Mediante la escena de transicion)

4.2.2.1. Este boton utiliza el sonido de click al ser presionado.

4.3. La musica continua de la escena desde la cual esta escena fue llamada.

4.4. Imagen de referencia



5. Escena de juego

5.1. Es la escena del juego propiamente dicho, donde el jugador controla el vehiculo e intenta superar las etapas.

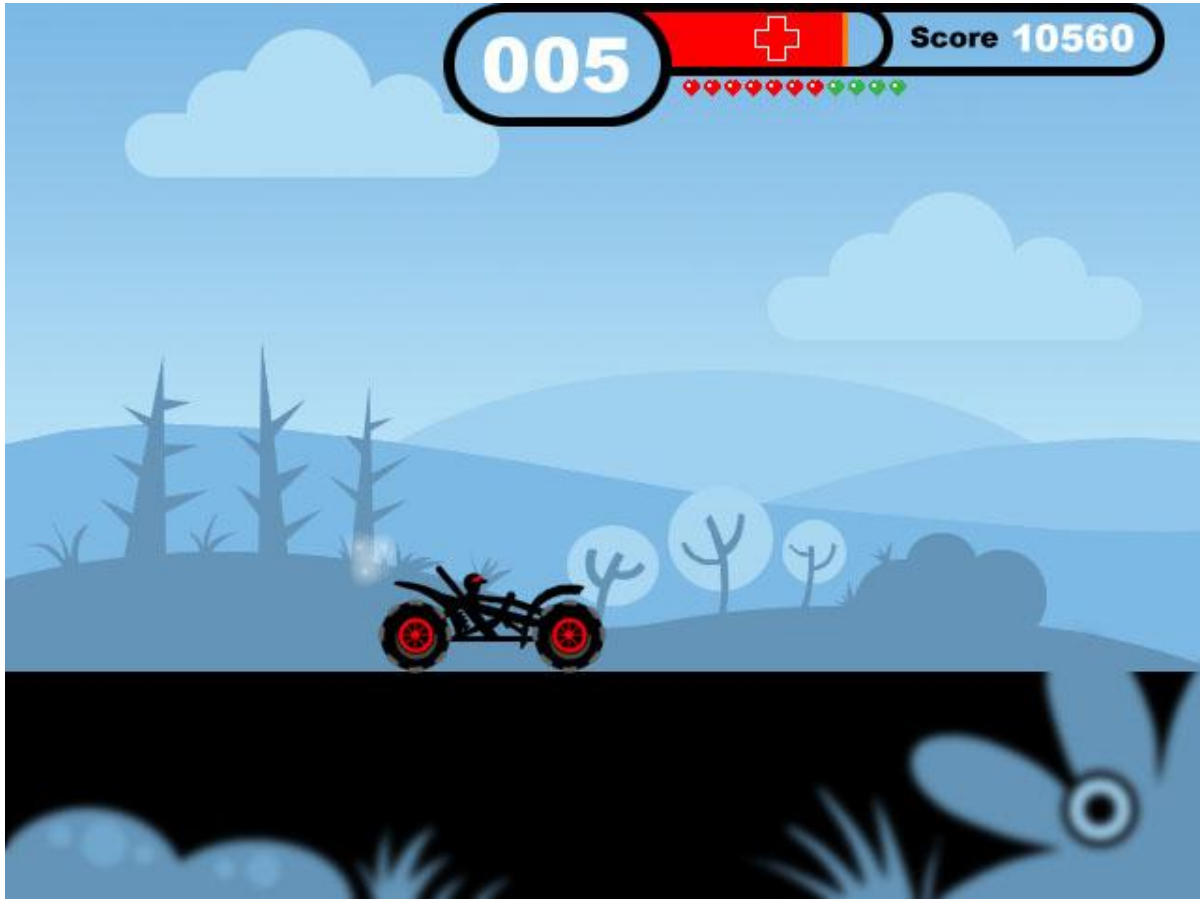
5.1.1. La vista del juego es lateral, se trata de un sidescroller.

5.1.2. El vehiculo puede avanzar hacia la derecha o hacia la izquierda y frenar.

5.1.3. El vehiculo puede rotar hacia la izquierda o la derecha.

5.1.4. El vehiculo puede dar saltos hacia arriba.

5.1.5. Imagen de referencia.



5.2. Estructura general de etapas

5.2.1. Cada etapa de la escena de juego comienza en el punto mas alejado a la izquierda del mapa.

5.2.2. El objetivo de la etapa es llegar hasta una torre que indica el final de la etapa.

5.2.2.1. Cada etapa tiene al final una torre con una llama viva en la parte superior. Al pasar la torre cae un rayo sobre ella y una animacion muestra la torre derrumbandose en una pila de ladrillos. Hay un sonido cuando cae un rayo sobre la torre y esta se derrumba

5.2.2.2. Al ser derrumbada la torre pasa a ser un obstaculo solido, impasable.

5.2.3. El objetivo del juego es superar todas las etapas antes que el dragon nos atrape.

5.2.4. El objetivo secundario del juego es recoger todos los powerups que se pueda para lograr un mayor score.

5.2.5. Otro objetivo secundario es llegar al final de la etapa lo antes posible para obtener un mayor score.

5.3. Escena de Etapa superada

5.3.2. Al pasar la torre del final de etapa aparece un cartel animado con un texto que marca el numero del nivel al cual se acaba de acceder. El juego continua sin interrupciones.

5.3.2.1. Este cartel animado tiene asociado un sonido al aparecer. Es un sonido abstracto.

5.3.3. Se updatea internamente el score y las vidas.

5.3.3.1. Los corazones verdes en la GUI se transforman en corazones rojos.

5.3.4. Aparece el cartel con el resumen de score (por detras del vehiculo/dragon/terreno pero por delante del resto del escenario). Luego de unos segundos el resumen de score desaparece.

5.6. Escena de juego perdido

5.6.1. Si el jugador pierde todas las lives, pierde el juego.

5.6.1.1. Cuando el tiempo llega a cero o el health del vehiculo se termina el jugador pierde una life.

5.6.1.2. Cuando el health del vehiculo se termina el jugador pierde una life.

5.6.2. Cuando el jugador pierde el juego aparece una animacion que muestra al jugador siendo masticado por el dragon y le informa que ha perdido el juego mediante el mensaje **"Game over"**. El sonido del dragon masticando acompaña a la animacion. La musica es el leitmotiv de dragon rush con un tono en caida.

5.6.3. Luego aparece el recuadro con el resumen del score. En la parte inferior hay dos botones. **RETRY LEVEL** y **QUIT**.

5.6.3.1. **RETRY LEVEL:** Lleva nuevamente al comienzo de la etapa que se estaba jugando. Todas las variables resetadas a sus valores iniciales (puntos, vidas, etc). De este modo se puede continuar el juego sin necesidad de volver al principio pero no se puede conseguir un gran score. (Mouseover: Try the level again but the score is reset to 0) (mediante escena de transicion). Sonido de click al presionar.

5.6.3.2. **QUIT:** Muestra un recuadro con la pregunta: **"Are you sure you want to quit?"** y los botones **YES** y **NO**. Sonido de click al presionar.

5.6.3.2.1. **YES:** Lleva nuevamente a la escena de inicio. Sonido de click al presionar.

5.6.3.2.2. **NO:** El recuadro desaparece. (Mouseover Yes: Back to cover screen, Mouseover No: Cancel) (mediante escena de transicion). Sonido de click al presionar.

5.7. Escena de juego superado

5.7.1. Luego de superar 10 niveles, el juego termina y el jugador ha logrado escapar del Dragon.

5.7.2.1. Aparece una animacion que muestra el vehiculo escapando velozmente por la derecha de la pantalla. Musica: entrada del vehiculo (accion). Sonido: El vehiculo escapando a toda velocidad de la pantalla. ruido de motor pasando y alejandose.

5.7.2.2. El dragon que lo perseguia, entra en la pantalla, parece co

nfundido, no encuentra al vehiculo. Musica: entrada del dragon (silencio/suspense/espera). Sonido: El dragon chillando enojado.

5.7.2.3. De repente unos pequeños personajes armados con lanzas y escudos entran en pantalla dando pequeños saltos y arrojan sus lanzas al dragon. Musica: entrada de los guerreros (marcha militar). Sonido: Los guerreros gritando sus gritos de batalla. (voces aceleradas)

5.7.2.4. El dragon recibe los impactos y cae detras de la montaña como un helicoptero. Detras de las montañas se ve una explosion. Musica: ataque de los guerreros, caída del dragon (accion). Sonido: Las lanzas surcando el aire hacia el dragon. Las lanzas dandole al dragon. El dragon emitiendo el sonido de un helicoptero que cae sin control. Explosion del dragon (mismo sonido que explosion del vehiculo)

5.7.2.5. Los guerreros dan saltitos de alegria. Musica: festejo de los guerreros (leitmotiv, hacia arriba). Sonido: Gritos de festejo de los guerreros (voces aceleradas)

5.7.3. Luego de la animacion aparece el recuadro con el resumen del score. Alli hay dos botones **PLAY AGAIN** y **QUIT**.

5.7.3.1. **PLAY AGAIN**: permite regresar al juego en el primer nivel (Mediante escena de transicion). Sonido de click al presionar.

5.7.3.2. **QUIT**: nos lleva a la escena de inicio (En este caso debido a que el juego ha sido superado en su totalidad, no es necesario preguntar si esta seguro)(Mouseover Play again: Restart from level 1, mouseover quit: Back to cover screen) (Mediante escena de transicion). Sonido de click al presionar.

5.8. Elementos del escenario

5.8.1. Camara

Para dar una sensacion visual de profundidad se utilizaria un efecto de parallax de varios layers que se desplazan a distintas velocidades. Cuando el vehiculo salta muy alto, la camara "lo acompaña" en el salto desplazando apropiadamente los demas layers. Mas lento los layers mas lejanos y mas rapido los layers mas cercanos.

5.8.1.1. En el layer principal tenemos el escenario de juego.

5.8.1.2. Delante del layer principal se ven objetos mas cercanos que se desplazan rapidamente y lucen desenfocados.

5.8.1.3. Detras del layer principal vemos un paisaje lejano que scrollea mas lentamente.

5.8.1.4. Finalmente, detras de todo, tenemos el fondo mas lejano que incluye el horizonte y el cielo. No se desplaza sino que esta fijo en la pantalla.

5.8.2. Iman de camara

Es una caracteristica que no es absolutmanente necesario implementar. Depende mas que nada de los tiempos de desarrollo. Es interesante y podria ser reutilizado en otros proyectos.

Funciona de la siguiente manera: los imanes de camara son invisibles, se colocan sobre el escenario y cuando el vehiculo se acerca a ellos "atraen" la camara hacia un punto especifico.

Se utilizan para mostrar algo importante del terreno antes que el vehiculo llegue a ellos para darle al jugador tiempo a reaccionar.

Tienen caracteristicas que los definen:

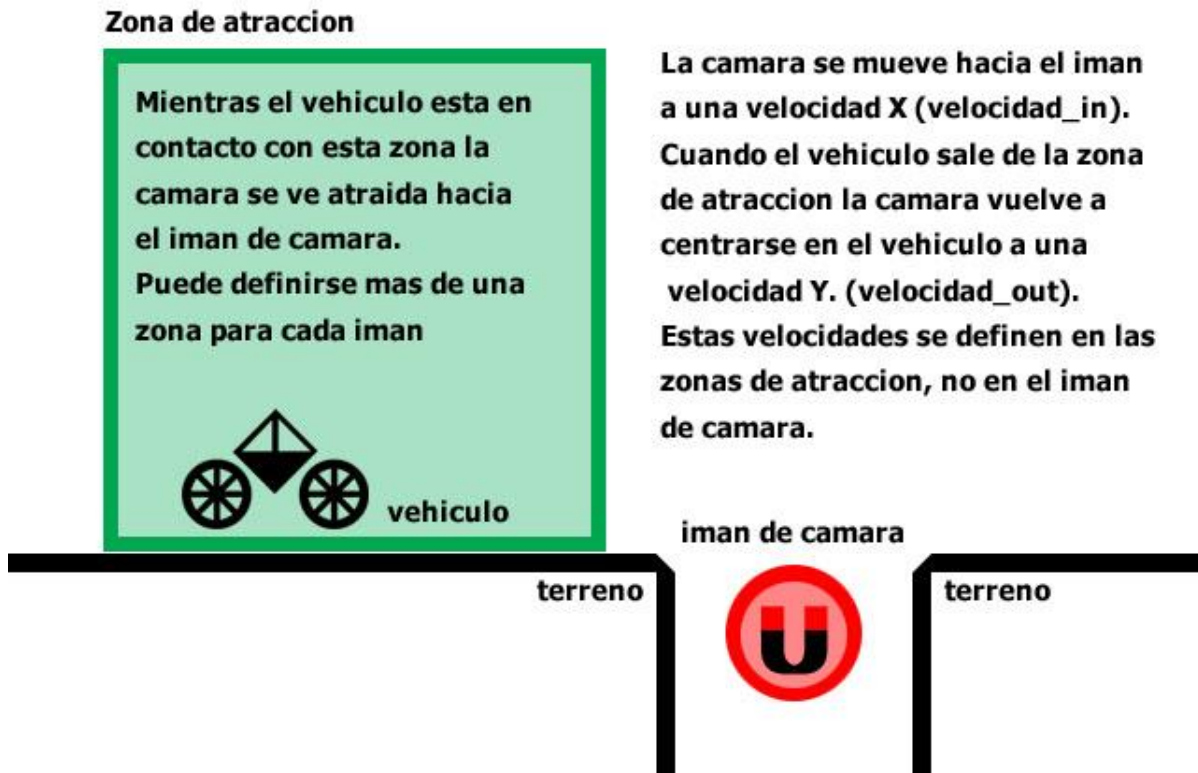
5.8.2.1. Punto de atraccion. Es un unico punto que indica hacia donde se movera la camara al ser atraida.

5.8.2.2. zona de atraccion. El area es un poligono invisible, mientras el vehiculo toca o esta dentro de este poligono el iman "atrae" la camara hacia el punto definido. Puede

haber varias zonas de atraccion para un mismo punto de atraccion.

5.8.2.3. La velocidad de atraccion. Define que tan rapido se mueve la camara hacia el iman (velocidad_in) o desde el a otro punto (velocidad_out). Estas velocidades se definen en la zona de atraccion y no en el punto de atraccion.

5.8.2.4. Imagen de referencia.



5.8.3. Superficie normal

El escenario esta compuesto mayoritariamente de una superficie inalterable sobre la cual el vehiculo se desplaza.

5.8.4. Huecos

5.8.4.1. Son simplemente interrupciones en el terreno, huecos por donde el vehiculo puede caer.

5.8.4.2. Si el vehiculo cae a un hueco pierde instantaneamente una life y debe volver al principio de la etapa en la que se encontraba.

5.8.4.3. Los huecos deberian estar señalizados mediante carteles en el terreno, tambien podrian estar afectados por un iman de camara si estos se encuentran funcionales.

5.8.5. Vehiculo del jugador

Tecnicamente el vehiculo del jugador esta separado en varias partes que dependen fisicamente entre si:

5.8.5.1. **Rueda:** las ruedas son las que traccionan sobre el piso e impulsan el vehiculo.

5.8.5.2. **Chasis:** El chasis del vehiculo sirve de nexo entre las ruedas y proporciona peso y simetria al modelo fisico.

5.8.5.3. **Piloto:** El piloto esta unido al chasis y deberia "flotar" sobre el para que los cambios de direccion se reflejen en la direccion del cuerpo. (por favor programadores aportar a esto).

5.8.5.4. **Efecto de humo.** Es un efecto de humo que el vehiculo "suelta" en el escenario al acelerar

5.8.5.5. **Efecto de explosion.** Cuando el vehiculo es alcanzado por el dragon vemos este efecto de explosion.

5.8.5.6. **Efecto de speed powerup.** Es una llama que sale por la parte trasera del vehiculo mientras esta afectado por el speed powerup que le permite alcanzar una velocidad maxima mayor.

5.8.6. Efectos

5.8.6.1. **Rugido:** La pantalla tiembla y hay un sonido de rugido de dragon. Este temblor no afectara la fisica del juego, es solo visual.

5.8.6.2. **Respiracion de fuego:** Unas llamas aparecen por la izquierda de la pantalla. Estas llamas no hacen daño, solo son visuales.

5.8.6.3. **Lluvia de fuego:** Unas bolas de fuego son arrojadas desde el lado izquierdo de la pantalla. Fisicamente son pelotas fisicas que reaccionan al escenario. Luego de unos segundos "explotan" dejando llamas mas pequeñas encen

5.8.6.3. Imagen de referencia.



5.8.7. Carteles informativos.

5.8.7.1. Habrá carteles como parte del escenario que contendrán información útil para el jugador. Por ejemplo un cartel puede informarle que hay un hueco peligroso cerca.

5.9. Filosofía de diseño de etapas

5.9.1 Los niveles serán muy simples y contruidos en base a figuras estandarizadas que combinadas entre si, determinan la superficie sobre la que el vehiculo puede recorrer.

5.9.2. La unica interrupcion posible al rapido flujo del nivel es cuando el vehiculo cae en un hueco. Por lo demas no habrá trabas al recorrido sobre el escenario. Ya que se trata de escapar del dragon que nos persigue, motivamos al jugador a seguir sin pausa hacia adelante, lo mas rapido posible.

5.9.3. Por lo general deberá ser muy facil para cualquier jugador superar todas las etapas con una minima interacción.

5.9.4. Sin embargo aquellos jugadores que deseen superar los tiempos promedio y obtener mayor cantidad de puntos tendran la posibilidad de hacerlo utilizando los powerups de aceleración.

5.9.4.1. Estos powerups estan colocados estrategicamente en las etapas de manera que para atraparlos el jugador deba maniobrar el vehiculo, no solo recorrer las etapas de la forma mas simple. Por ejemplo se encontrarian entre medio de dos saltos, de manera que el jugador deberia tomar la primera rampa con menos velocidad y evitar pasar volando sobre el powerup.

5.9.4.2. Ademas muchos de los extra bonus score tambien se encontraran en ocasiones fuera del alcance del jugador a menos que este maniobre para tocarlos. Estos powerups permiten obtener mayor puntuacion al finalizar una etapa.

5.9.5. Es importante dejar al comienzo de cada etapa superficie recta suficiente para que el jugador pueda distraerse unos segundos y observar el resumen o la informacion que deseemos suministrarle. **La forma en que se cargan las etapas es diferente respecto del engine original.**

5.10. Piezas para creacion de terreno, colisiones

5.10.1. El terreno se crea en base a una secuencia de "slices", que son como grandes piezas en las cuales el terreno esta dividido.

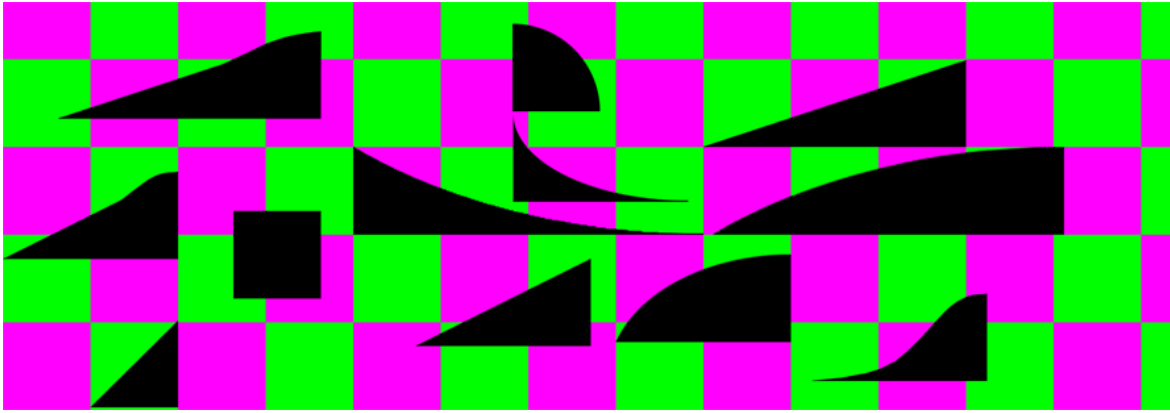
5.10.1.1. La parte visual de cada slice se crea combinando pequeñas piezas standar en photoshop. Se combinan estas piezas para armar una porcion visual de terreno. Luego se exporta la imagen en formato png y se agrega a un "slice" de terreno.

5.10.1.2. El terreno colisionable, fisico, se crea utilizando piezas simples e invisibles en el juego (circulo perfecto, cuadrado, rectangulos) para crear las colisiones del terreno.

5.10.1.3. La unica regla a seguir es que estas piezas no pueden deformarse solo rotarse y cambiar de tamaño.

5.10.1.4. Por otra parte estas piezas deben estar centradas en el medio, no en la esquina superior izquierda.

5.10.1.5. Imagen de referencia.



5.11. GUI

5.11.1. **Time.** En la parte superior/central de la pantalla se ve un contenedor simple que muestra el time/bonus decreciendo segundo a segundo.

5.11.2. **Score.** En la parte superior/derecha se ve el score, los puntos actuales (El score obtenido hasta la ultima etapa superada).

5.11.3. **Health.** Junto al contenedor del time/bonus se ve una barra de HEALTH que representa la "salud" del personaje. Cuando el personaje recibe daño esta barra se va gastando y cuando se agota el jugador pierde una Life.

5.11.3.1. El vehiculo pierde health mientras toca el escenario con cualquier parte del chasis del vehiculo. Si el health llega a cero el vehiculo explota y jugador pierde una life.

5.11.4. **Lives.** La cantidad de lives esta representada mediante unos pequeños iconos con forma de corazon rojo. Hay un corazon rojo por cada vida disponible.

5.11.5. **Extra lives.** Tambien vemos unos pequeños iconos con forma de corazones verdes. Hay uno por cada extra life powerups que el jugador ha encontrado en este nivel en particular.

5.11.5.1. Cuando el jugador logra superar el nivel estos corazones verdes se transforman en vidas y pasan a ser corazones rojos.

5.11.5.2. Si no logra superar el nivel significa que comenzara el nivel nuevamente con una life menos y los corazones verdes desaparecen.

5.12. Aparicion del Dragon

5.12.1. Unos 10 segundos antes que el TIME se acabe una gigantesca cabeza de dragon asoma por la parte izquierda/abajo de la pantalla y se va desplazando hacia el vehiculo. El dragon se desplaza lentamente siempre alineado con el eje Y del vehiculo y la idea es que eventualmente llegue muy cerca del vehiculo para atacarlo.

5.12.1.1. Hay que aclarar que si el time llega a cero esto no implica que el jugador pierde una life.

5.12.2. La velocidad con que se desplaza desde el borde hacia el centro de la pantalla es inversamente proporcional a la velocidad del vehiculo. De ese modo cuanto mas rapido va el vehiculo mas tardara el dragon en alcanzarlo pero si el vehiculo frena totalmente el dragon llegara rapidamente a el.

5.12.3. De vez en cuando lanza unos fuertes rugidos y la pantalla tiembla (aunque este temblor no afecta fisicamente al vehiculo, es solo visual).

5.12.4. Cuando el Dragon llega al centro escupe fuego por la boca, el vehiculo explota automaticamente y el jugador pierde una life. Al perder una life:

5.12.4.1. Debe comenzar desde el principio del nivel nuevamente. El Bonus acumulado vuelve a cero, las extra lives acumuladas (corazones verdes) se pierden y debe comenzar la etapa desde el principio. Basicamente se resetea la etapa y no gana puntos ni vidas en el nivel hasta que logre superarlo.

5.12.5. Puede forzarse la aparicion del dragon desplazando el vehiculo hacia la izquierda o permaneciendo quieto durante cierta cantidad de tiempo.

5.13. Powerups

5.13.1. **Extra life.** El jugador puede juntar estos iconos en las etapas. Cada extra life que junta esta representada por un corazon verde en la GUI.

5.13.1.1. Al superar la etapa los corazones verdes se transforman en corazones rojos (lives).

5.13.1.2. Al perder una life los corazones verdes desaparecen.

5.13.2. **Health.** La barra de Health se vuelve a llenar al 100%.

5.13.3. **Extra Bonus Score.** El jugador los recoge durante la etapa y al final se multiplican por una cierta cantidad de puntos y se suma ese numero al score total. Si el jugador pierde una life el contador (es interno) de extra bonus score volvera a cero y el jugador debera recogerlos de nuevo para sumarlos al score. Es decir, solo se suman al score al superar una etapa.

5.13.4. **Extra Speed.** Este powerup permite correr al doble de velocidad durante 10 segundos.

5.13.4.1. Al tocar un speed powerup un efecto de llamas saliendo de la parte posterior del auto nos muestra que el powerup esta activo.

5.20. Bocetos etapas

5.20.1. etapa 1. Imagen de referencia



5.20.2. etapa 2. Imagen de referencia



5.20.3. etapa 3. Imagen de referencia



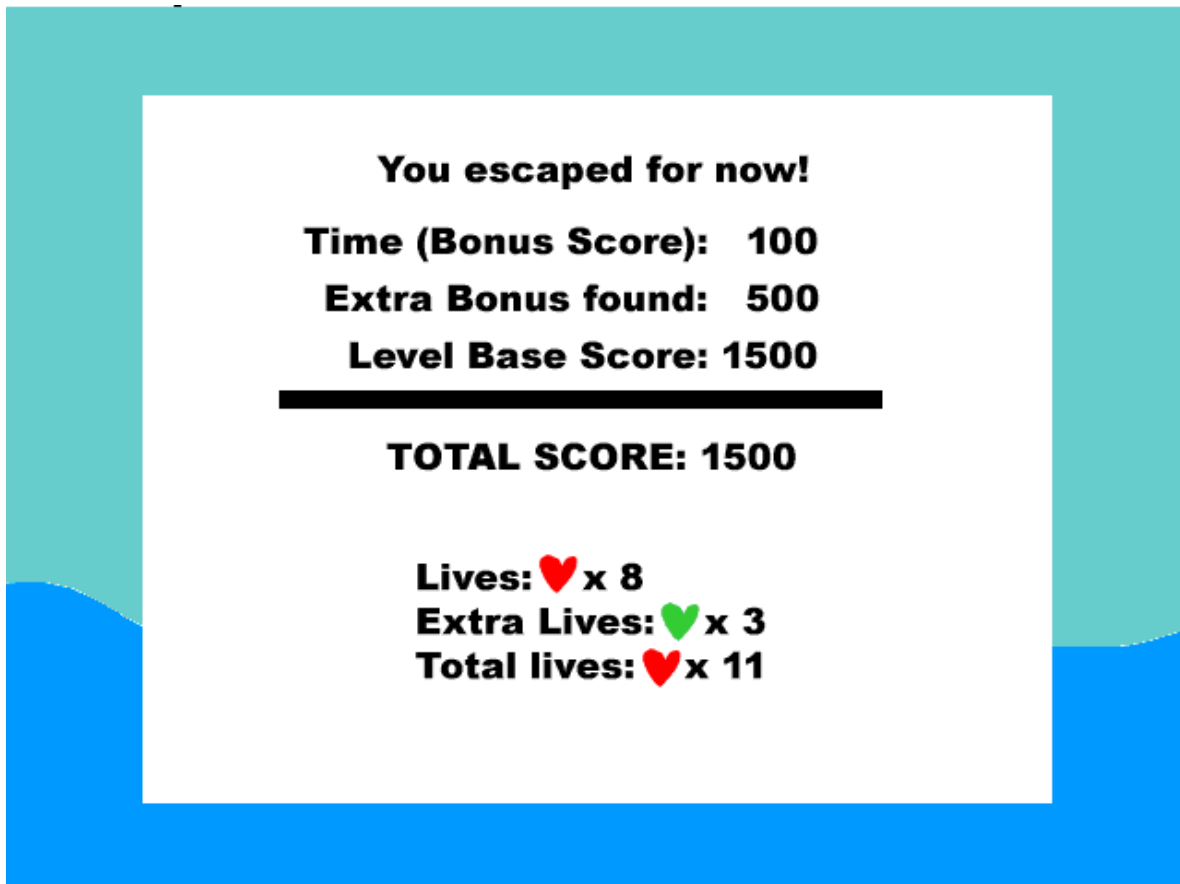
5.14. Score

5.14.1. El puntaje se calcula al superar cada etapa segun la siguiente formula: $\text{TOTAL SCORE} = \text{TIEMPO RESTANTE} + \text{EXTRA BONUS ENCONTRADOS} + \text{TOTAL SCORE}$

5.14.2. Tambien se muestra la actualizacion de las lives segun las extra lives encontradas en la etapa.

5.14.3. Se muestra visualmente el calculo del score de la siguiente manera:

5.14.3.1. Imagen



5.15. Musica

5.15.1. Durante la accion utilizamos el leitmotiv del juego en clave rapida.

5.15.2. A partir del momento en que el dragon entra en escena (acercandose al vehiculo) utilizamos una version mas compleja para indicar la cercania del dragon y el peligro.

5.16. Sonido

Resumen de los sonidos utilizados en la escena de juego.

- 5.16.1. Salto
- 5.16.2. Sonido de Conteo rapido
- 5.16.3. Sonido 1 resumen de score
- 5.16.4. Sonido 2 resumen de score
- 5.16.5. Rugido de Dragon
- 5.16.6. Click 1
- 5.16.7. Click 2
- 5.16.8. Juntar powerup
- 5.16.9. Recibir daño

5.17. Controles

- 5.17.1. Acelerar: arriba
- 5.17.2. Frenar/retroceder: abajo
- 5.17.3. Girar izquierda: izquierda
- 5.17.4. Girar derecha: derecha
- 5.17.5. Salto: espacio