

8 DE OCTUBRE DE 2023

# GAME DESIGN DOCUMENT

## RALLY TEAM TACTICS

*BARRACUDA INTERACTIVE*

ADRIÁN RUBIO GARRIDO  
ALEJANDRO ASENSIO PÉREZ  
ÁLVARO MARTÍN HITA  
DANIEL HERNÁNDEZ TAMAYO  
ERIC MARTÍNEZ GAMERO  
SERGIO MONTES VEREDAS



## TABLA DE CONTENIDO

Índice de figuras .....	3
índice de ilustraciones.....	3
Índice de tablas .....	5
1 Introducción .....	6
1.1 Introducción al juego y su enfoque al aprendizaje pensamiento computacional .....	6
1.2 Contenido del documento .....	6
2 Referencias y concepto de juego .....	7
2.1 Principales referencias .....	7
2.2 Concepto de juego y bases de la jugabilidad .....	11
2.2.1 Rol del jugador .....	11
2.2.2 Objetivo del jugador, jugabilidad y gestión de recursos .....	11
2.2.3 Gameplay loop .....	12
3 Objetivo del proyecto: enseñanza del Pensamiento Computacional a niños pequeños.....	14
3.1 Los videojuegos en el aprendizaje del Pensamiento Computacional .....	14
3.2 Público objetivo.....	14
3.3 Destrezas del Pensamiento Computacional en el juego .....	14
3.4 Plataformas .....	15
4 Trasfondo .....	16
4.1 Ambientación .....	16
4.2 Narrativa.....	16
5 Mecánicas y elementos de juego .....	17
5.1 Cámara y perspectiva .....	17
5.2 Gameplay loop en detalle .....	18
5.2.1 Fase de reconocimiento en detalle .....	18
5.2.2 Fase de ejecución en detalle .....	20
5.3 Mecánicas de juego.....	21
5.3.1 Comportamiento del vehículo en cada tipo de sección .....	21
5.3.2 Relieve en el trazado .....	25
5.3.3 Gestión de recursos: combustible y daño sufrido.....	29
5.4 Controles y periféricos .....	31
6 Progresión .....	32
6.1 Objetivos del jugador .....	32
6.2 Estructura del juego .....	32
6.3 Clasificación en línea .....	35

7 Arte.....	36
7.1 Referencias y estética del juego.....	36
7.2 Arte final.....	36
8 Interfaz de usuario .....	37
8.1 Requisitos de la interfaz .....	37
8.2 Diagrama de flujo de navegación.....	37
8.3 Diseño visual de las pantallas.....	37
9 Sonido.....	38
9.1 Efectos de sonido .....	38
9.2 Música .....	38
10 Producción.....	39
10.1 Modelo de negocio del juego y plan de financiación.....	39
10.2 Marketing .....	39
10.3 Versiones preliminares del producto .....	39
10.4 Producto final.....	39
11 Conclusiones.....	40
Referencias y bibliografía .....	41
Anexo .....	42

## ÍNDICE DE FIGURAS

### ÍNDICE DE ILUSTRACIONES

Ilustración 1. Colin McRae Rally 3 (PS2) .....	7
Ilustración 2. DiRT Rally (PS4) .....	7
Ilustración 3. WRC 10 (PC) .....	8
Ilustración 4. Virtual Racing (Arcade) .....	8
Ilustración 5. Ridge Racer (Arcade) .....	9
Ilustración 6. Daytona USA (Arcade) .....	9
Ilustración 7. Art of Rally (PC) .....	10
Ilustración 8. You Suck at Parking (Xbox One) .....	10
Ilustración 9. Gameplay loop de un sector .....	12
Ilustración 10. Gameplay loop de una etapa .....	12
Ilustración 11. Leyenda para diagramas de flujo sobre el gameplay .....	13
Ilustración 12. Perspectiva y movimiento de la cámara .....	17
Ilustración 13. Gameplay loop de la fase de reconocimiento .....	19
Ilustración 14. Gameplay loop de la fase de ejecución .....	20
Ilustración 15. Leyenda para el comportamiento del vehículo .....	21
Ilustración 16. Iconos de las instrucciones de tipo de trazado .....	21
Ilustración 17. Comportamiento del coche en recta .....	22
Ilustración 18. Comportamiento del coche en curva poco cerrada .....	23
Ilustración 19. Comportamiento del coche en curva cerrada .....	23
Ilustración 20. Comportamiento del vehículo en curva muy cerrada .....	24
Ilustración 21. Comportamiento del coche en horquilla .....	24
Ilustración 22. Sketch de una rampa .....	25
Ilustración 23. Sketch de una meseta .....	26
Ilustración 24. Sketch de un valle .....	26
Ilustración 25. Sketch de un hundimiento .....	27
Ilustración 26. Sketch de un charco .....	27
Ilustración 27. Sketch de un bache seguido de un charco en una curva poco cerrada ..	28
Ilustración 28. Sketch de las barras de combustible y daño .....	29
Ilustración 29. Sketch de los recursos: sin combustible .....	30

Ilustración 30. Sketch de los recursos: coche destrozado..... 30

Ilustración 31. Estructura de un tramo cronometrado (días 1 a 3 -fases 1, 2 y 3-) ..... 33

Ilustración 32., Estructura de una etapa completa cronometrada (último día -fase 4-) 33

Ilustración 33. Estructura del juego..... 34

Ilustración 34. Estructura de un evento ..... 34

ÍNDICE DE TABLAS

Tabla 1. Destrezas del Pensamiento Computacional en el juego ..... 15

## 1 INTRODUCCIÓN

Este documento cubre todos los aspectos referentes al diseño del juego **Rally Team Tactics (RTT)**, **en desarrollo** por el equipo de *Barracuda Interactive* para PC, navegadores web y dispositivos Android y cuyo lanzamiento está previsto para finales de 2023.

### 1.1 INTRODUCCIÓN AL JUEGO Y SU ENFOQUE AL APRENDIZAJE PENSAMIENTO COMPUTACIONAL

Como breve introducción al juego, **Rally Team Tactics (RTT)** es un *arcade racer* donde el jugador asume el rol del copiloto de un equipo de rally. Como tal, este debe analizar el trazado del circuito para dar las instrucciones adecuadas al piloto antes de que comience la prueba.

El juego pone gran énfasis en el ensayo y error gracias a la posibilidad de rebobinar hasta el punto de control previo. El jugador tiene así la posibilidad de resolver el problema basándose en lo acontecido anteriormente. La identificación de patrones (respuesta de las físicas del vehículo al relieve y el itinerario) le permiten mejorar la elección de instrucciones y hacer así una gestión de los recursos (gasolina y daño recibido) más eficiente en su siguiente intento.

### 1.2 CONTENIDO DEL DOCUMENTO

A lo largo de los siguientes capítulos, se cubre en este documento:

- Principales referencias y jugabilidad
- Propósito del proyecto: enseñanza del Pensamiento Computacional a niños pequeños
- Herramientas y programas utilizados
- Proceso de diseño del juego en todos sus ámbitos de manera extensiva
  - Trasfondo: ambientación, narrativa y entornos
  - Mecánicas y elementos de juego
  - Progresión
  - Arte
  - **Música y sonido**
  - Interfaces y experiencia de usuario (UI y UX)
  - Producción y modelo de negocio

## 2 REFERENCIAS Y CONCEPTO DE JUEGO

### 2.1 PRINCIPALES REFERENCIAS

**Rally Team Tactics (RTT)** es un *arcade racer* inspirado por otros juegos de simulación de carreras o de conducción *arcade*, con ejemplos provenientes desde los 90 hasta la actualidad. Entre estas referencias, encontramos:

- Franquicias de rally y carreras *off-road* realistas:
  - **Colin McRae Rally**, conocida actualmente como **DiRT** y **DiRT Rally** (Codemasters, EA, 1998-2020). Ver Ilustración 1 e Ilustración 2.
  - **WRC** (múltiples desarrolladores, EA, 2001-2023). Ver Ilustración 3.

De estas se toman su énfasis en la simulación de las físicas de los vehículos sobre diferentes terrenos.



Ilustración 1. Colin McRae Rally 3 (PS2)



Ilustración 2. DiRT Rally (PS4)





Ilustración 3. WRC 10 (PC)

- Juegos de carreras para recreativas de los años 90:
  - **Virtua Racing** (Sega AM2, 1992). Ver Ilustración 4.
  - **Ridge Racer** (Namco, 1993). Ver Ilustración 5.
  - **Daytona USA** (Sega AM2, 1993). Ver Ilustración 6.

Estos juegos sirven como inspiración tanto para el estilo artístico colorido y retro de *RTT* como para mecánicas arcade como las pruebas contrarreloj, el uso del turbo o *leaderboards*.



Ilustración 4. Virtual Racing (Arcade)



Ilustración 5. Ridge Racer (Arcade)



Ilustración 6. Daytona USA (Arcade)

- Juegos *indie* de conducción *arcade* contemporáneos:
    - **Art of Rally** (Funselektor Labs, 2020): las principales influencias provenientes de este título son su *setting* en la época dorada del rally y su arte 3D *low-poly* de gran calidad y su iluminación. Ver Ilustración 7.
- Otros aspectos por destacar:
- Su perspectiva de cámara aérea.
  - Sus físicas que no llegan al realismo de los títulos del primer punto pero que encuentran un buen balance entre el control *arcade* y la simulación centrada en las características del terreno (relieve, superficie) y las condiciones climatológicas.

- ***You Suck at Parking*** (Happy Volcano, 2022): otra gran inspiración, especialmente por su naturaleza ensayo-error (reinicio rápido). El juego introduce además elementos de puzles que ponen a prueba la memoria y el reconocimiento de patrones del jugador. También cuenta con una cámara aérea, con la posibilidad de visualizar el nivel libremente antes de empezar un intento. Este último elemento es una fuerte influencia para RTT. Ver Ilustración 8.



Ilustración 7. Art of Rally (PC)



Ilustración 8. You Suck at Parking (Xbox One)



## 2.2 CONCEPTO DE JUEGO Y BASES DE LA JUGABILIDAD

**Rally Team Tactics (RTT)** se concibe como un *arcade racer* en 3D ambientado en la época dorada del rally.

### 2.2.1 ROL DEL JUGADOR

El jugador toma el control de un **copiloto de rally** en lugar del piloto. Este giro le da una nueva perspectiva al género, pues en lugar de tener agencia directa sobre el vehículo, **se deben “cantar” las notas que contienen código nemotécnico y que describen las características de la carrera**. Además, como a los copilotos en la realidad, al jugador se le otorga la **responsabilidad de supervisar aspectos del coche** como el nivel de combustible restante o los daños sufridos.

Como no controlamos el coche directamente, qué notas “cantemos” determinarán el comportamiento de este. Para situar las anotaciones del copiloto en el trayecto, se permite en todo momento tener una “vista de pájaro” con una **cámara libre para navegar por toda la etapa**, como si tuviésemos acceso a las cámaras del helicóptero de televisión.

### 2.2.2 OBJETIVO DEL JUGADOR, JUGABILIDAD Y GESTIÓN DE RECURSOS

El objetivo es lograr el **mejor tiempo en cada etapa de rally**, que se divide en **tramos cronometrados**. Antes del día de la etapa completa, **el jugador se debe enfrentar a cada tramo cronometrado por separado**, refinando su estrategia de cara al día de la etapa.

Solo podemos **situar las instrucciones mientras el vehículo se encuentre parado**, por lo que para cambiarlas se debe reiniciar el tramo o etapa mediante **un reinicio que conserva las instrucciones seleccionadas**.

La experimentación puede llevar al jugador de chocar en la primera curva a conseguir su mejor tiempo en el tramo o etapa a base de ensayo y error. **Si el coche choca o sufre demasiados daños, es obligatorio regresar al inicio** para idear una nueva estrategia.

Siempre que se finaliza un tramo o etapa **se muestra el tiempo conseguido**, y si se ha completado el tramo o etapa en el pasado también si se ha superado al histórico y por cuánto.

La gestión de recursos juega un papel importante. Tanto el daño sufrido por el vehículo como el combustible restante deben tenerse en cuenta. Al inicio del tramo o etapa, el coche tiene 0% de daños y 100% de combustible.

- Si el coche sufre una caída o impacta con algún objeto su integridad se ve comprometida, y el **medidor de daño (damage)** incrementa en diferentes cantidades.

- El tanque de combustible siempre se va consumiendo, pero utilizar el turbo o acelerar tras una frenada importante producen un mayor consumo del **medidor de gasolina (fuel)**.

2.2.3 GAMEPLAY LOOP

El *gameplay loop* básico se refleja en el siguiente diagrama para un tramo:

Gameplay Loop de un tramo

En un tramo los jugadores solo pueden elegir instrucciones entre dos puntos de control, que determinan el tramo. Pueden buscar mejorar su tiempo en este tramo para prepararse para la etapa final

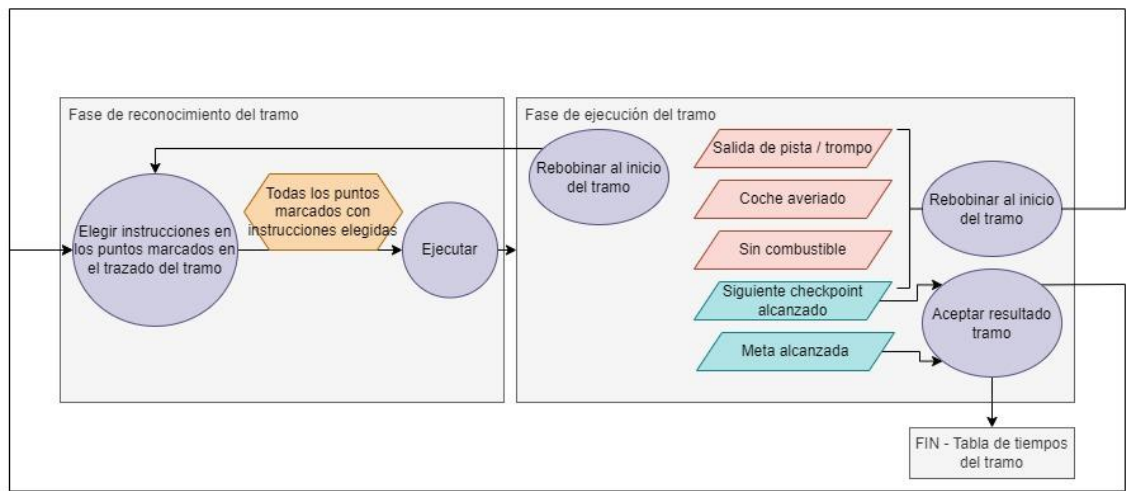


Ilustración 9. Gameplay loop de un sector

Y el siguiente para una etapa en su totalidad:

Gameplay Loop de una etapa completa

En la etapa completa los jugadores comienzan con las instrucciones que eligieron en los tramos en los días anteriores, ahora pueden variarlas para enlazar dichos tramos de manera óptima

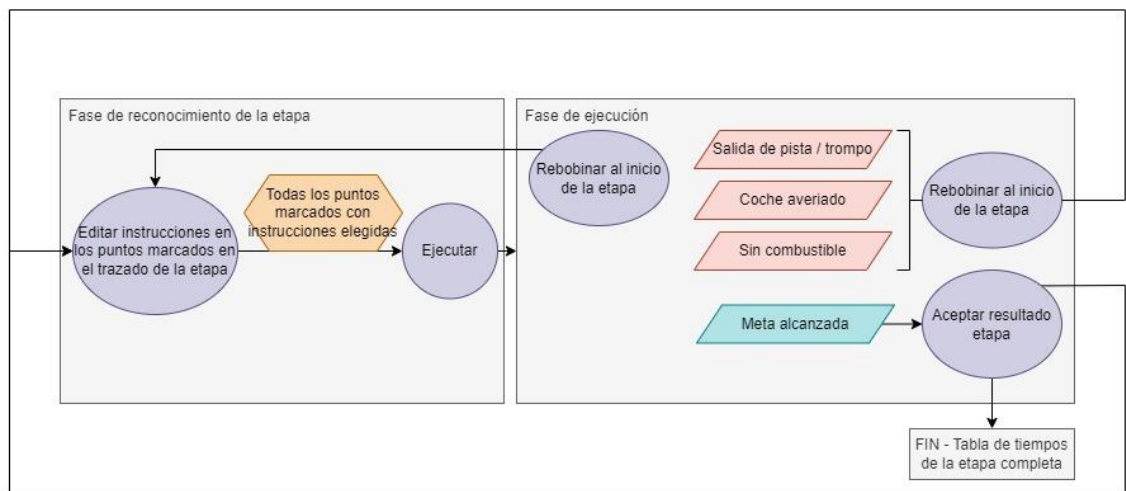


Ilustración 10. Gameplay loop de una etapa

En la leyenda que se muestra seguidamente se explica el significado de las formas en los diagramas de flujo referentes a *gameplay*:



Ilustración 11. Leyenda para diagramas de flujo sobre el gameplay

---

## FASE DE RECONOCIMIENTO

Como se plasma en el diagrama, el jugador puede elegir anotaciones (instrucciones) durante la **fase de reconocimiento**. Una vez se hayan seleccionado las instrucciones deseadas para todos los puntos señalizados, se puede proceder a ejecutar.

---

## FASE DE EJECUCIÓN

El paso a la **fase de ejecución** arranca el vehículo desde el punto de salida. Ahora podemos ver el vehículo realizar el trayecto de acuerdo a las órdenes situadas en reconocimiento. Durante la ejecución seguimos teniendo el control de la cámara y podemos reiniciar en cualquier momento, volviendo a la fase de reconocimiento.

---

## POSIBLES EVENTOS Y SUS CONSECUENCIAS

Se pueden producir una serie de eventos negativos, que nos forzarán a reiniciar, como una salida de pista completa o el consumo del tanque de gasolina.

Por otro lado, los eventos positivos posibles son alcanzar el siguiente punto de control o la meta. La meta es el final solo en el último tramo y en la etapa completa. Si el jugador acepta el resultado (no reinicia) se daría por finalizado el tramo o etapa. Además. De tratarse de una etapa completa, se mostraría la **pantalla de clasificación (*leaderboards*)** correspondiente; de lo contrario, se pasaría directamente a la selección de fase.

### 3 OBJETIVO DEL PROYECTO: ENSEÑANZA DEL PENSAMIENTO COMPUTACIONAL A NIÑOS PEQUEÑOS

#### 3.1 LOS VIDEOJUEGOS EN EL APRENDIZAJE DEL PENSAMIENTO COMPUTACIONAL

Llamamos Pensamiento Computacional (PC) a la habilidad cognitiva relacionada con el uso de estrategias computacionales para la resolución de problemas. Su aprendizaje desde una edad temprana puede proporcionar amplios beneficios en la vida cotidiana. Por este motivo, la enseñanza del PC es parte del currículo escolar en España y otros países desde la etapa de educación infantil.

Los videojuegos orientados a potenciar el PC pueden tener una mayor efectividad que otras actividades como la programación o los juegos de mesa. Los niños y niñas en la actualidad tienen una relación estrecha con el mundo de los videojuegos, y las posibilidades que estos ofrecen en la enseñanza apenas se han explotado.

Por los motivos citados anteriormente, el juego **en desarrollo** pretende explorar ese nicho, creando una experiencia lúdica que impulse el uso del PC dentro de un entorno de entretenimiento.

#### 3.2 PÚBLICO OBJETIVO

El *target* del videojuego son niños y niñas desde 8 hasta 12 años. Esta demográfica juega con frecuencia a títulos de cierta complejidad como *Fortnite* o *Roblox*, por lo que se tiene en cuenta este factor a la hora de diseñar el juego: pese a tratarse de un producto educativo, no subestima las capacidades de los niños y evita las características habituales de los *serious games*. Se busca reforzar el PC sin hacerlo explícito, comprometer la diversión ni la posibilidad de que el juego pueda ser disfrutado por audiencias de mayor edad.

#### 3.3 DESTREZAS DEL PENSAMIENTO COMPUTACIONAL EN EL JUEGO

Desde la concepción del juego se busca potenciar múltiples facultades propias del PC. Las destrezas particulares que se pretenden ejercitan jugando RTT son las siguientes:

Destreza del PC	Cómo se pone en práctica en el juego
<b>Abstracción</b>	Esta habilidad se instruye gracias a la decisión de diseño que pone al jugador en los pies del copiloto. El jugador debe abstraerse de la reacción inmediata de la conducción, situándose en una <b>capa superior de planificación y toma de decisiones</b> . La “vista de pájaro” pone énfasis en esta capacidad de <b>ver el problema en su totalidad y reconocer así sus elementos clave</b> .

<b>Pensamiento algorítmico</b>	En RTT se pone a disposición del jugador un conjunto de notas que el copiloto “canta” en determinados momentos, según su colocación y orden. Se debe <b>crear un algoritmo</b> , al fin y al cabo, una <b>secuencia de acciones</b> concretas que da lugar al mejor resultado posible.
<b>Descomposición</b>	En base a la respuesta del vehículo en un determinado tramo con una determinada serie de instrucciones, el jugador puede adquirir la capacidad de <b>deducir</b> cómo se va a comportar el coche en el mismo escenario u otro similar en función de sus decisiones en la fase de reconocimiento.
<b>Evaluación</b>	<p>El <i>RTT</i> se presta especial atención a la <b>gestión de recursos</b>, es este caso, el combustible y el daño sufrido. Se deben considerar a corto y medio plazo la consumición de combustible y el desgaste del coche. Esto implica no solo tener en mente el tramo inmediatamente posterior, sino la etapa en su totalidad. Reservar combustible para un momento concreto o arriesgar la integridad del vehículo en el punto adecuado de la pista son ejemplos de decisiones que pueden dar lugar a mejores tiempos.</p> <p>Además, mediante el ensayo y error, el jugador asimila la <b>detección de fallos</b>, y toma <b>decisiones ajustadas al objetivo</b>. Si el coche responde de una determinada forma a una secuencia de instrucciones que producen un consumo de recursos correspondiente, puede usar ese conocimiento para siguientes intentos.</p>
<b>Generalización</b>	<p><i>RTT</i> pone el foco en la repetición para encontrar soluciones mejores. <b>Resolver el problema en base a soluciones anteriores</b> es parte fundamental del diseño. Cuando el jugador <b>encuentra patrones y similitudes</b> en la reacción del vehículo ante distintos obstáculos o trazados, aprende a utilizar las estrategias que llevan a un resultado más satisfactorio.</p> <p>Aunque el jugador no conoce al detalle las físicas del coche (ni se pretende que lo haga), a medida que se va familiarizando con su comportamiento, toma decisiones que le resultan más favorables.</p>

Tabla 1. Destrezas del Pensamiento Computacional en el juego

### 3.4 PLATAFORMAS

Como los niños suelen disponer de tabletas en los centros educativos o en su hogar, es fundamental que el juego se desarrolle no solo para **escritorio** y **navegadores Web**, sino también para dispositivos táctiles Android (en este caso **tablets Android**).



## 4 TRASFONDO

### 4.1 AMBIENTACIÓN

### 4.2 NARRATIVA

## 5 MECÁNICAS Y ELEMENTOS DE JUEGO

### 5.1 CÁMARA Y PERSPECTIVA

La perspectiva de la cámara es **aérea**, emula la vista desde un helicóptero de televisión. El entorno tridimensional se puede navegar tanto en fase de reconocimiento como en fase de ejecución.

Las limitaciones de movimiento de la cámara son:

- Imposibilidad de hacer *zoom*
- Imposibilidad de regular la altitud
- La inclinación es fija, de tal manera que se observen con suficiente claridad las características del terreno y el trayecto del circuito

El jugador tiene control, por tanto, sobre:

- La posición en el plano horizontal definido a la altitud predeterminada
- El giro de la cámara, 360 grados de rotación

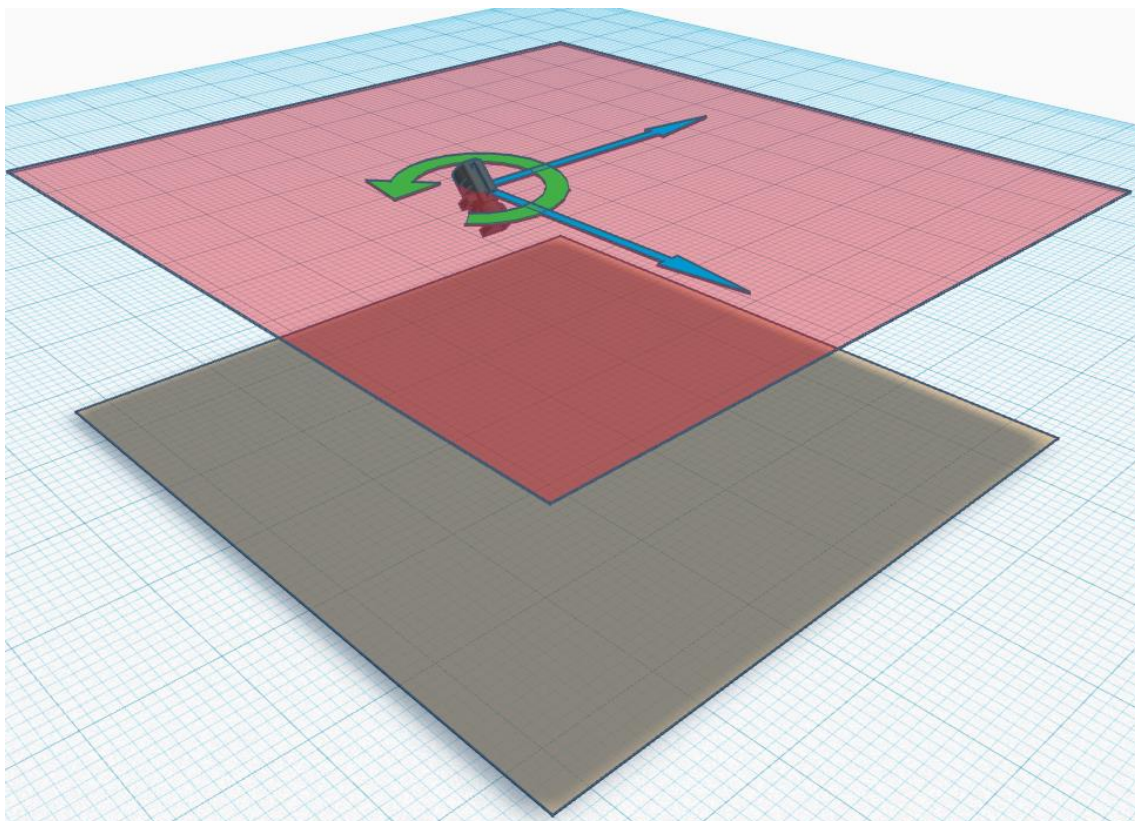


Ilustración 12. Perspectiva y movimiento de la cámara

Las flechas azules y verdes representan el movimiento horizontal y giro de la cámara, respectivamente, mientras que el plano rojo indica la altitud fija, y como se puede observar, la inclinación de la cámara también está bloqueada (ver Ilustración 12).

## 5.2 GAMEPLAY LOOP EN DETALLE

En este apartado se van a describir con detenimiento las fases que componen el *gameplay loop* del juego.

Como se indicó en el apartado [2.2.3 Gameplay loop](#), el *gameplay loop* de *RTT* está separado en dos fases o turnos: reconocimiento y ejecución. Esto es igual en los días de preparación (los distintos tramos) y en el día final (la etapa completa).

### 5.2.1 FASE DE RECONOCIMIENTO EN DETALLE

Durante el reconocimiento, el jugador puede navegar con libertad tal como se ha descrito en [5.1 Cámara y perspectiva](#), y, además:

Seleccionar un  
punto de  
instrucción

Limpiar todo  
el circuito de  
instrucciones

Comenzar  
prueba

Acceder al  
menú de  
ajustes

Los puntos de instrucción son puntos del circuito donde se debe especificar la anotación que se va a “cantar” al piloto cuando llegue a cada uno de ellos. Al **seleccionar un punto de instrucción**, el jugador tiene las siguientes opciones:

- A. **Seleccionar la nota nemotécnica** que ve más conveniente en función de el trazado inmediatamente posterior al punto de instrucción. Entre estas notas encontramos, por ejemplo, “curva muy cerrada” o “recta”.
- B. **Elegir el temperamento** con el que realizar la acción. Esto es posible gracias a un *slider* que permite determinar si se quiere un acercamiento **más cauteloso o más agresivo**.
  - Por defecto, cuando se coloca una nota por primera vez en un punto, el *slider* se encuentra en una posición intermedia, moderada.
  - Para ilustrar esta idea, supongamos que tenemos la nota “curva muy cerrada”: cuanto más agresivos seamos, el frenado se producirá más tarde y con mayor fuerza, mientras que, si optamos por un acercamiento más cauteloso, el frenado será más largo y comenzará mucho antes. En el caso de una nota de tipo “recta”, los dos extremos se corresponderían con reducir la velocidad y acelerar, siendo el punto medio del *slider* mantener la velocidad. Todos los casos se explican al detalle más adelante ([5.3 Mecánicas de juego](#)).
- C. **Deseleccionar un punto de instrucción**, que es tan sencillo como pulsar o clicar en otro lugar del circuito.

Por otro lado, en todo momento se puede **limpiar el circuito**, que sería equivalente a comenzar de cero el nivel: todas las instrucciones situadas se eliminarían.

La acción que daría paso a la fase de ejecución es **comenzar la prueba**, eso sí, esto solo se puede hacer si se han colocado notas en todos los puntos de instrucción.

Además de todo lo anterior, siempre es posible **entrar en el menú de ajustes**, pausando el juego. Aquí, entre otras cosas (configuración), se permite salir de la prueba.

El siguiente diagrama muestra el *gameplay loop* correspondiente a la fase de reconocimiento en mayor detalle:

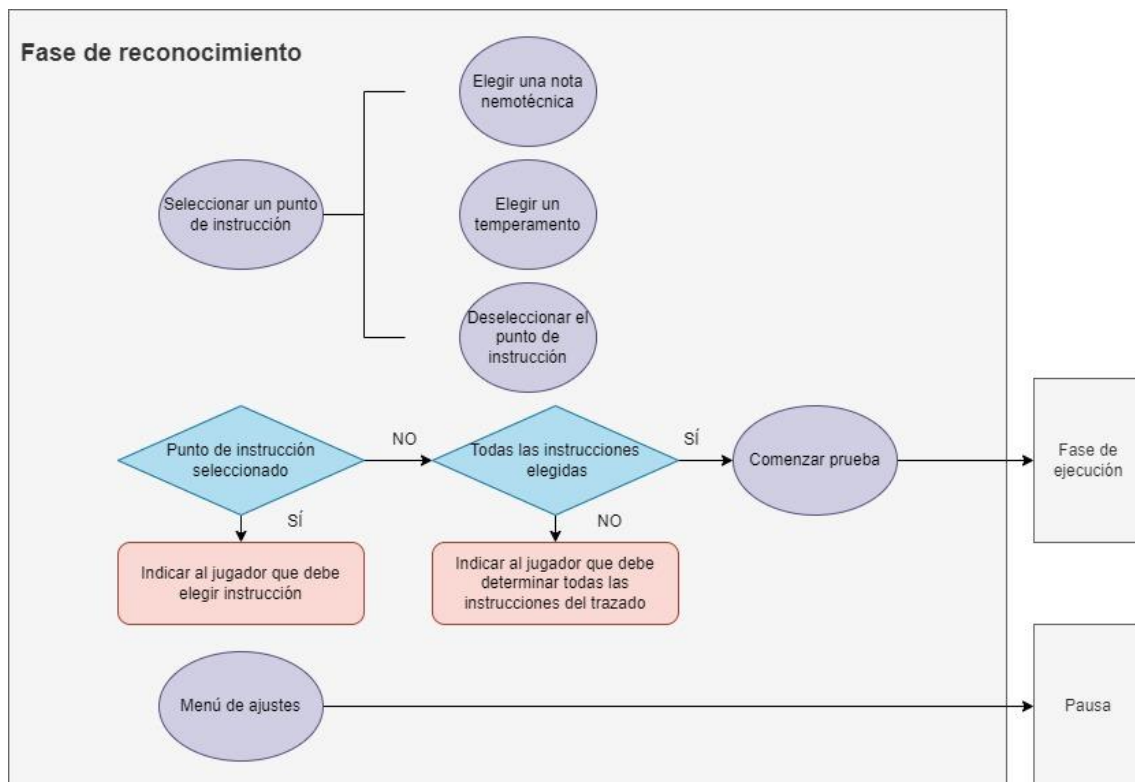


Ilustración 13. Gameplay loop de la fase de reconocimiento

Ver la leyenda de la Ilustración 11 para el significado de las diferentes formas.

### 5.2.2 FASE DE EJECUCIÓN EN DETALLE

Durante la fase de ejecución, se simula la prueba de *rally* de acuerdo a las notas nemotécnicas y temperamento elegidos a lo largo del trazado. El vehículo recorre el trayecto automáticamente, reaccionando según sus físicas con el relieve y las curvas de la carretera. Además, el copiloto “canta” las notas cuando el coche pasa por cada una de ellas (*voiceover*) para dotar de mayor realismo a la carrera.

Además de mover libremente la cámara, hay dos acciones que se pueden realizar siempre en esta fase:

- A. **Acceder al menú de ajustes** supone lo mismo que en la fase de reconocimiento
- B. **Reiniciar** implica que el coche volverá a la salida del tramo/etapa, conservándose todas las notas (instrucciones) elegidas previamente

Si el coche choca quedando inmóvil, se queda sin combustible o sufre daños irreparables, estas son las únicas opciones disponibles.

No obstante, si se completa el tramo o etapa de principio a fin, también se nos da la posibilidad de **aceptar el resultado**, que se muestra en el temporizador junto con la diferencia de tiempo respecto al histórico. En caso de que el jugador vea el resultado satisfactorio, entonces se le reconducirá a una pantalla donde podrá ver una clasificación con otros jugadores.

Veamos un diagrama al detalle de esta fase:

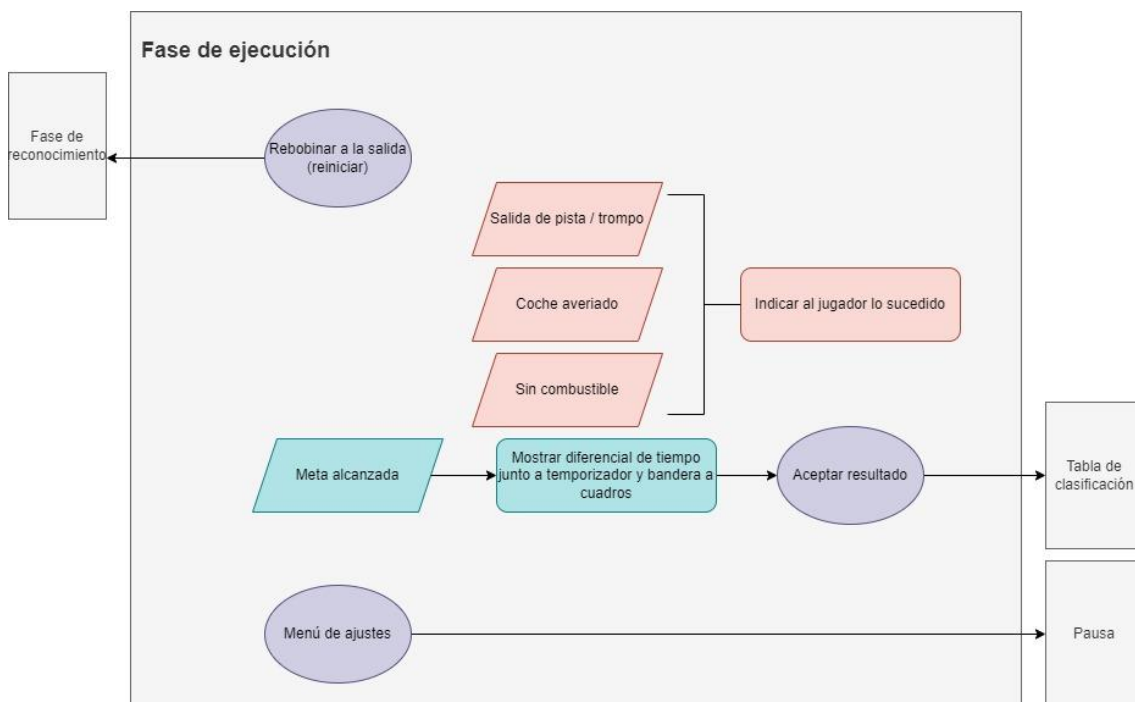


Ilustración 14. Gameplay loop de la fase de ejecución

Ver la leyenda de la Ilustración 11 para el significado de las diferentes formas.

## 5.3 MECÁNICAS DE JUEGO

En este apartado se describen las mecánicas de *Rally Team Tactics* con detenimiento.

En primer lugar, se va a explicar el comportamiento del vehículo ante las diferentes curvas y obstáculos si se elige su nota nemotécnica correcta, así como el efecto del temperamento en cada caso. En caso de no elegirse la nota correspondiente, el coche seguirá las directrices de la seleccionada, dando lugar a resultados que normalmente serán negativos, saliendo de la pista o chocando, pero que en raras ocasiones pueden optimizar el recorrido. La clave al fin y al cabo es la experimentación.

*RTT* presenta una serie de secciones de carretera predefinidas por su curvatura, todas ellas de la misma longitud. Es justo entre estas secciones en el trazado donde se eligen las notas y el temperamento.

### 5.3.1 COMPORTAMIENTO DEL VEHÍCULO EN CADA TIPO DE SECCIÓN

A continuación, se mostrarán bocetos de todos los tipos de sección y se explica cómo se comportaría el vehículo seleccionada la nota correcta, en tres casos concretos en el *slider* de temperamento: cauteloso, moderado y agresivo. La interpolación entre estos casos daría lugar a los comportamientos intermedios.

La siguiente leyenda indica el significado del código de colores de las flechas situadas en el trazado:

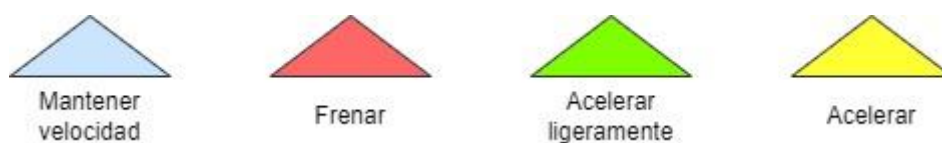


Ilustración 15. Leyenda para el comportamiento del vehículo

La imagen inferior muestra los iconos de las notas del código nemotécnico creados para los bocetos:

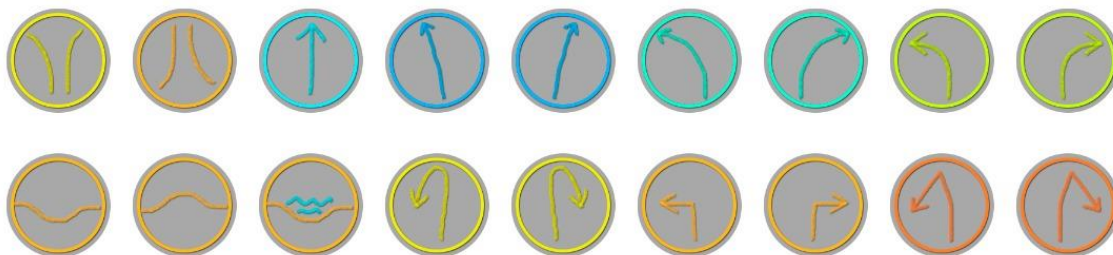


Ilustración 16. Iconos de las instrucciones de tipo de trazado



---

## OBSERVACIONES IMPORTANTES

- Las líneas guía no se muestran al jugador en ningún momento, son para *debug* y diseño.
- No hay solo siete o tres posiciones en el *slider*, no es discreto: el resto de posibles puntos son intercalaciones entre los extremos y el centro.
- Otro detalle de importancia es que en todos los casos presentados en cada sección se da por hecho que el coche empieza en el centro, lo cual no será así en la mayor parte de los casos. Siempre se va a tender progresivamente a adoptar esa posición intermedia. Por ejemplo, si se acaba de salir de una curva pegado a la izquierda y a continuación hay una recta, el vehículo buscará el centro. En resumen, el coche va a buscar siempre acercarse lo más posible al trayecto ideal.
- No se especifican valores de los parámetros como la velocidad en el diseño, se deben probar diferentes números durante la programación del juego hasta que se encuentren los valores de velocidad, frenado (reducción de la velocidad) y aceleración (aumento de la velocidad) que mejor plasman el desempeño de un coche de *rally in-game*.

---

## RECTA

En este primer caso se van a mostrar ejemplos más graduales, para que se vea cómo se interpola entre un extremo del *slider* de temperamento y el opuesto.

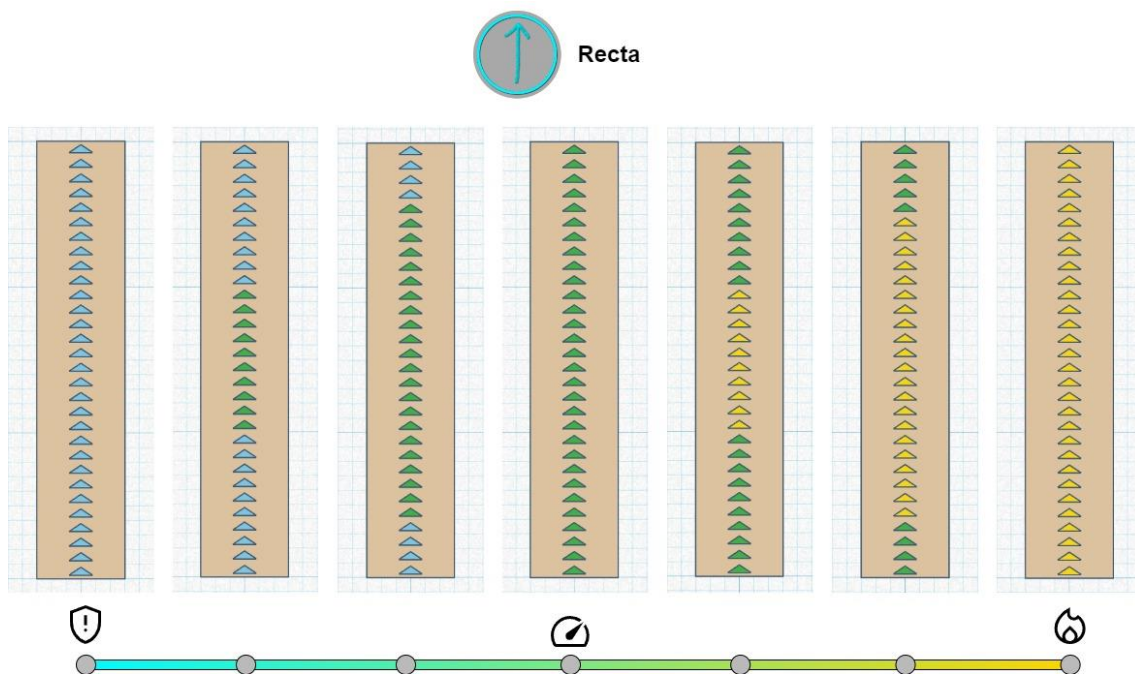


Ilustración 17. Comportamiento del coche en recta

Como se observa, a medida que se toma un perfil más agresivo, la aceleración es más fuerte y tiene una mayor duración.

---

## CURVA POCO CERRADA

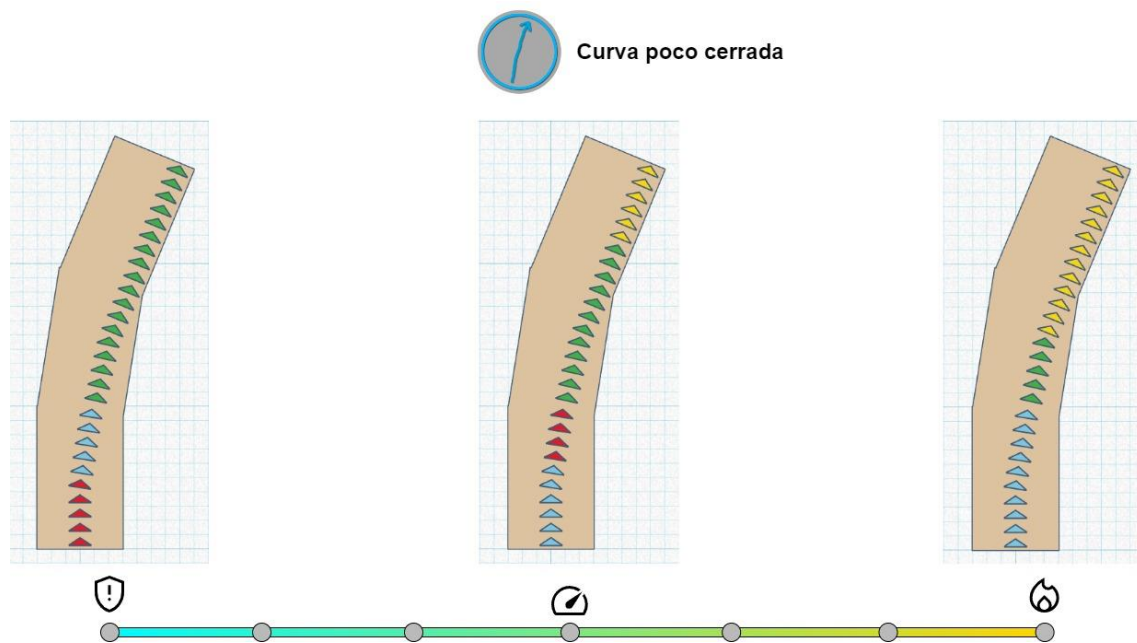


Ilustración 18. Comportamiento del coche en curva poco cerrada

Aquí se observa por primera vez la presencia de tramos de frenado. **No hay diferentes variantes de frenado como con la aceleración, solo un tipo que supone una reducción gradual de la velocidad.**

---

## CURVA CERRADA

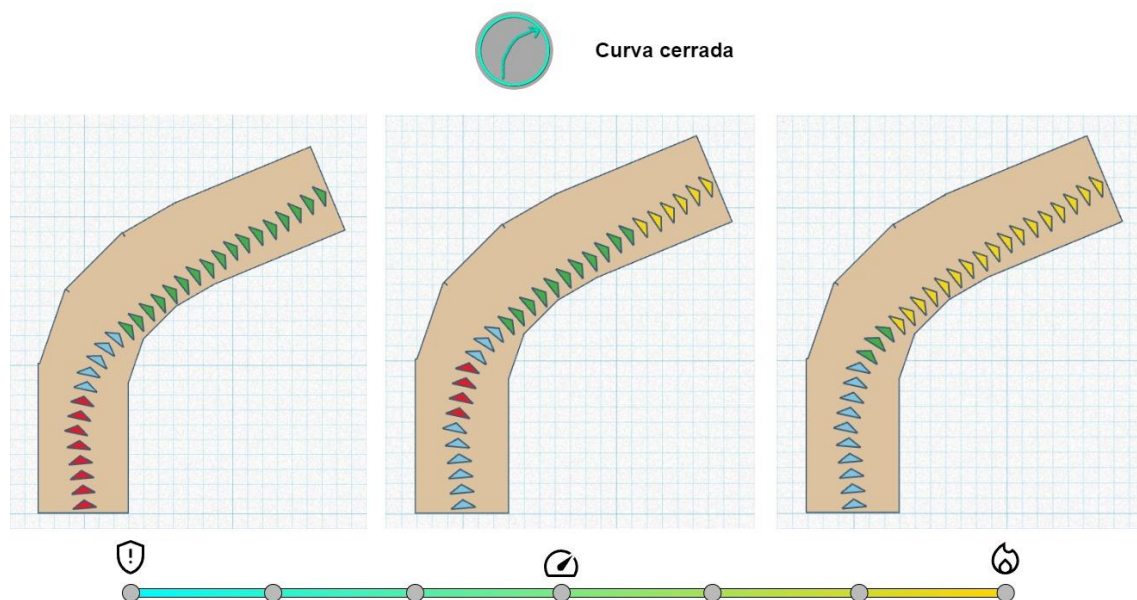


Ilustración 19. Comportamiento del coche en curva cerrada



CURVA MUY CERRADA

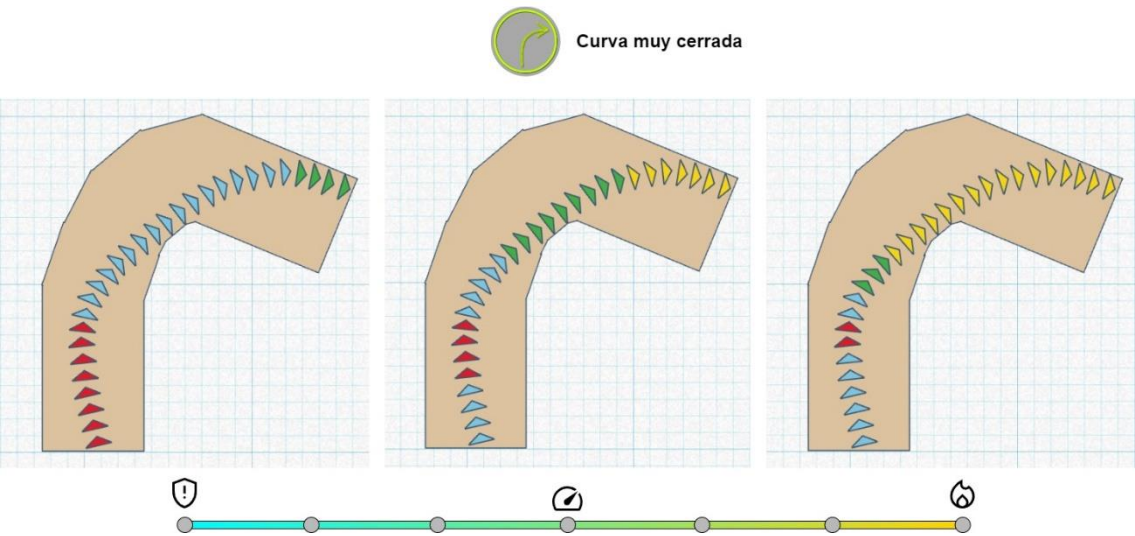


Ilustración 20. Comportamiento del vehículo en curva muy cerrada

HORQUILLA

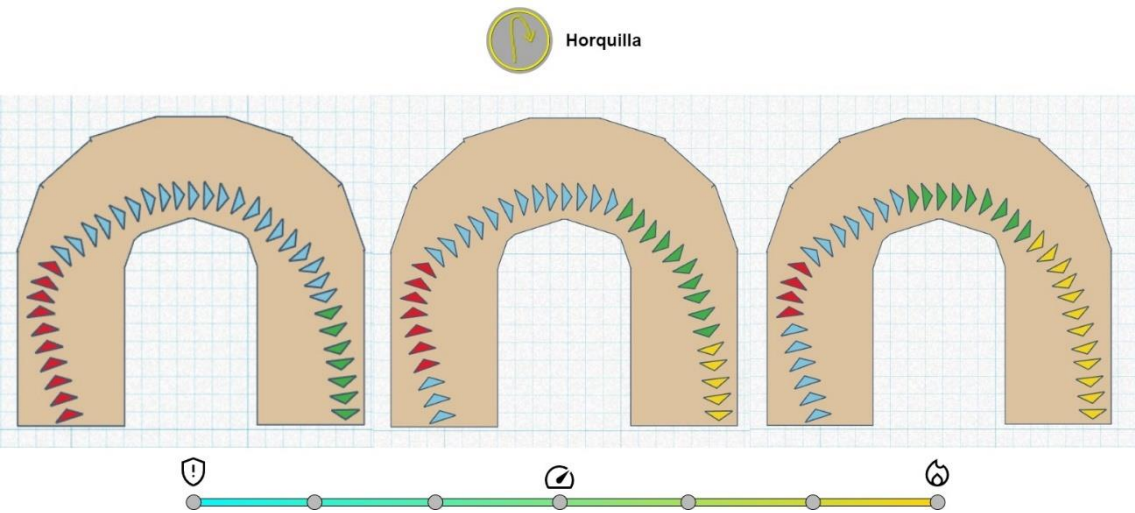


Ilustración 21. Comportamiento del coche en horquilla

### 5.3.2 RELIEVE EN EL TRAZADO

Para dar mayor diversidad a las etapas y plantear diferentes escenarios aún con secciones repetidas, se distribuyen a lo largo del itinerario obstáculos en forma de variaciones del relieve. Estas variaciones no afectan a cómo afronta el coche cada sección, pues eso está completamente determinado por la nota elegida y el temperamento seleccionado. Lo que hacen estos obstáculos es generar reacciones en las físicas del vehículo que le dificultan seguir el trayecto objetivo o provocan daños.

Es aquí donde entra en juego un buen manejo del temperamento: considerar el relieve de las secciones siguientes para determinar con qué nivel de cuidado o agresividad queremos entrar en cada una. Por ejemplo, si tenemos una recta con un bache a la mitad quizás sea más inteligente entrar con cautela, mientras que si se presenta una pendiente al inicio quizás sea mejor salir de la curva anterior con mayor agresividad para no perder mucha velocidad. Además, las colisiones del vehículo y caídas al dar saltos suponen daños, otro factor a considerar a la hora de enfrentarse a obstáculos en el circuito.

Los obstáculos se distribuyen en tres categorías: baches, depresiones y charcos. En principio estos ocupan todo el ancho de la pista, variando solo en altitud y longitud:

- Los **baches** serán porciones de una sección con una elevación seguida de un descenso. Pueden ser más o menos agudos en función del diseño de nivel elegido. Además, existen dos variantes: **rampas** y **mesetas**.

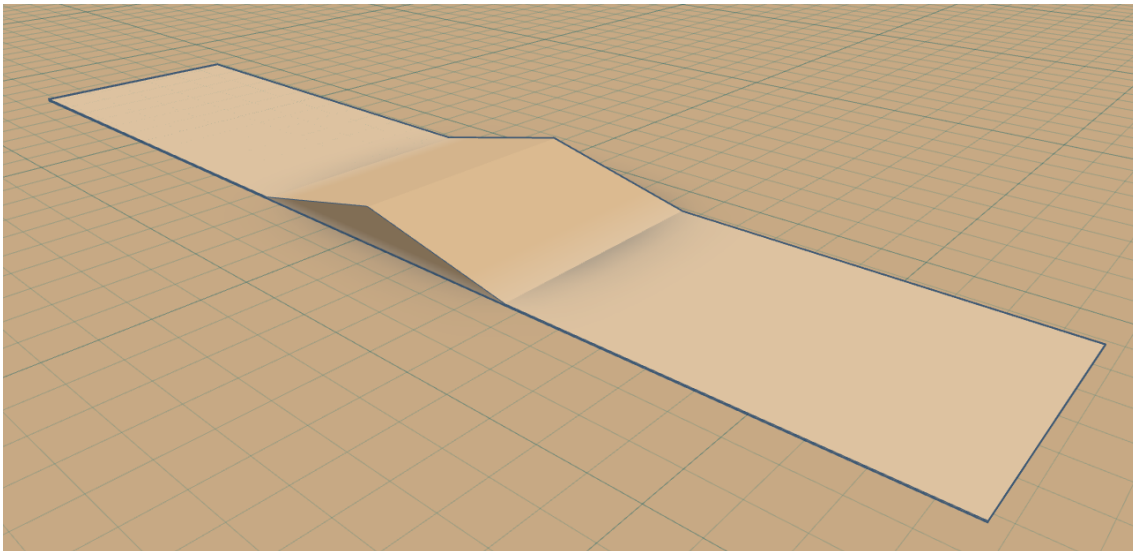


Ilustración 22. Sketch de una rampa

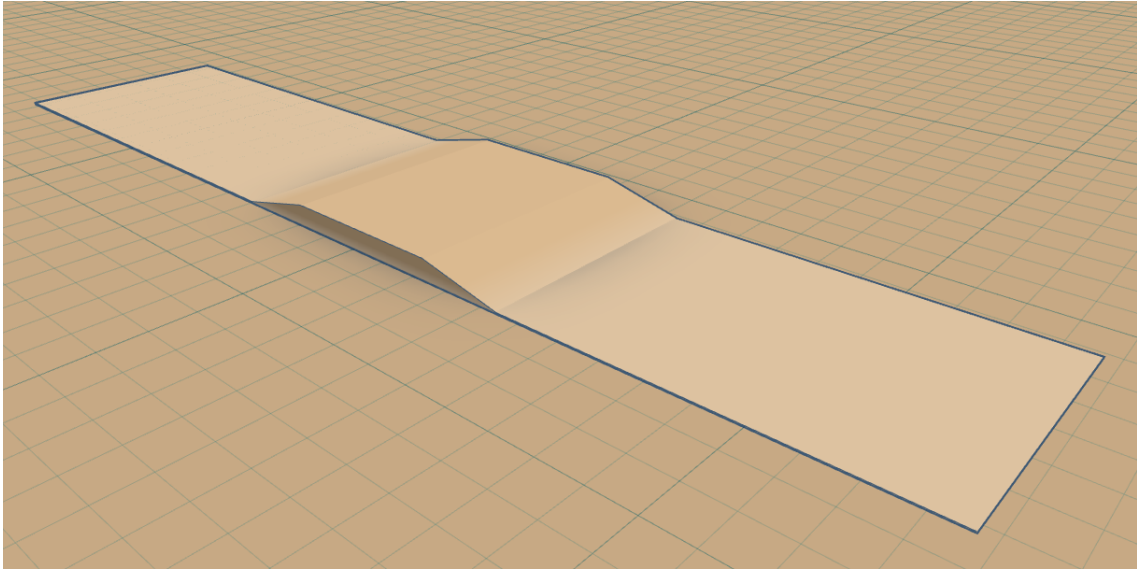


Ilustración 23. Sketch de una meseta

- Las **depresiones** son justo lo opuesto a los baches, es decir, hundimientos en el terreno de mayor o menor tamaño y longitud, a elección del diseñador de niveles. Las depresiones tienen dos variantes: **valles** y **hundimientos**.

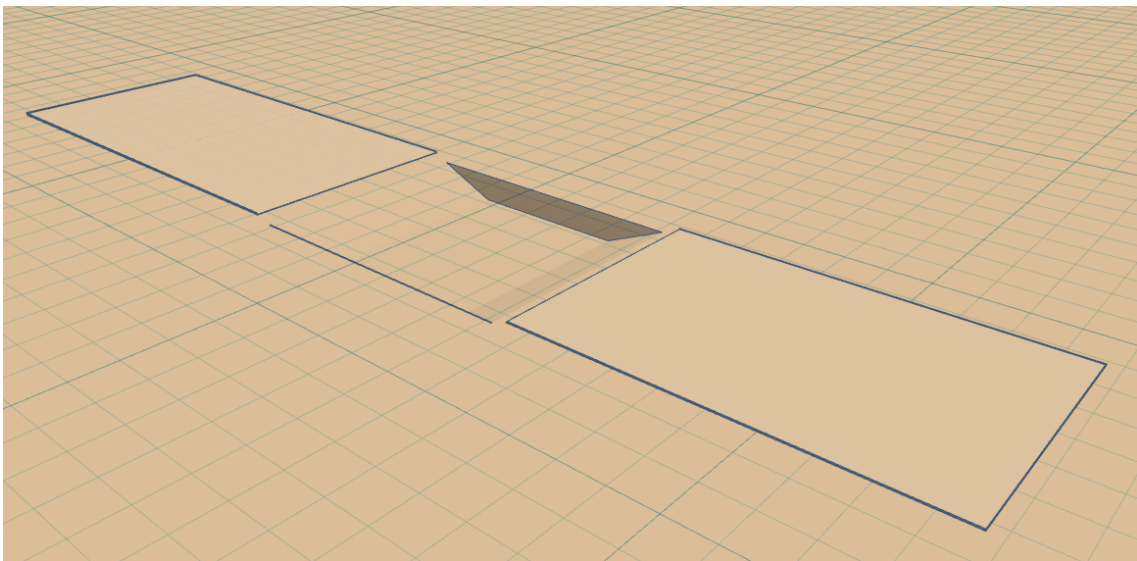


Ilustración 24. Sketch de un valle

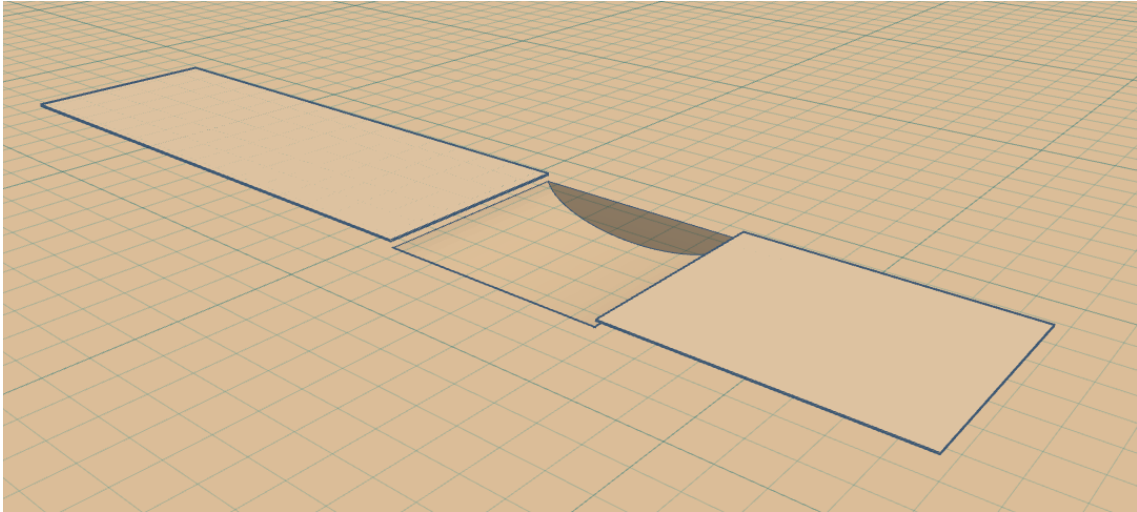


Ilustración 25. Sketch de un hundimiento

- Los **charcos** son zonas de una sección cubiertas por una fina capa de agua que ralentizan el avance del coche o incluso hacerlo resbalar y perder la dirección en curvas. Realmente se trata de valles de muy poca profundidad que están llenos de agua.

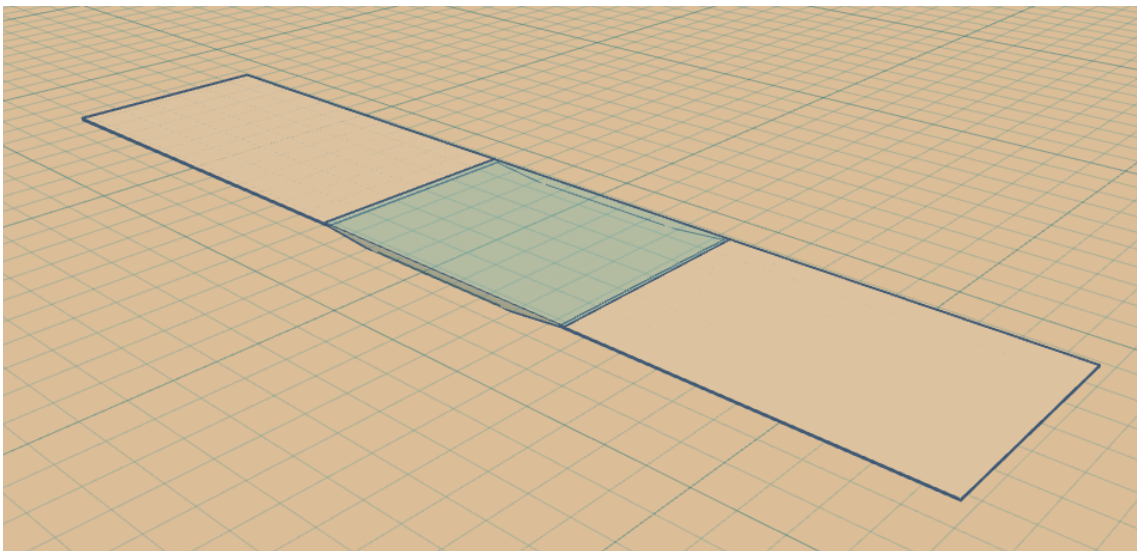


Ilustración 26. Sketch de un charco

---

## CONCATENANDO OBSTÁCULOS

Los obstáculos se pueden concatenar.

Pongamos un ejemplo y cómo este supondría una toma de decisión para el jugador: en el *sketch* mostrado a continuación se observa un bache seguido de un charco en una curva poco cerrada:

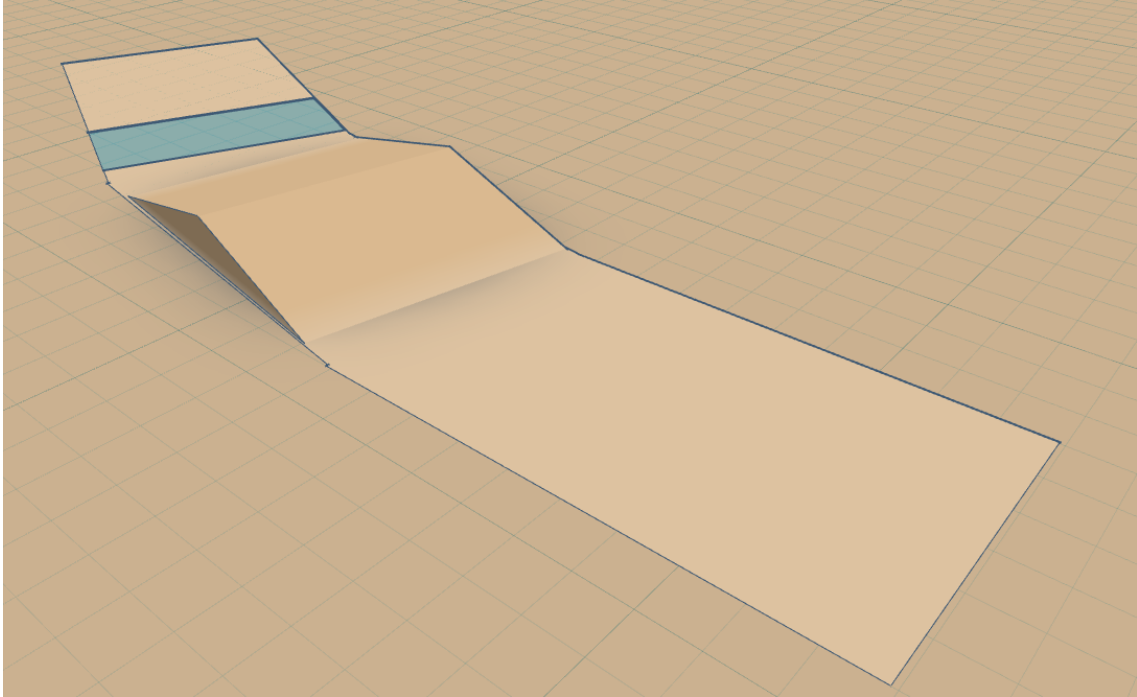


Ilustración 27. Sketch de un bache seguido de un charco en una curva poco cerrada

En este caso, salvo que el coche esté altamente dañado, quizás lo más interesante sería ejecutar la curva de manera agresiva, aprovechando el salto tras la rampa inicial del bache para evitar el charco y así no perder velocidad. Ahora bien, esto puede aplicar solo en caso de que realmente estemos dispuestos a intercambiar daño por un mejor tiempo en esta sección: quizás más adelante hay otro obstáculo que va a deteriorar el coche significativamente y se debe reconsiderar por tanto la estrategia aquí.

---

### 5.3.3 GESTIÓN DE RECURSOS: COMBUSTIBLE Y DAÑO SUFRIDO

Uno de los pilares fundamentales del *gameplay* de *Rally Team Tactics* es el uso adecuado de los recursos provistos al inicio de cada prueba al jugador. Estos son el **combustible en el tanque** y los **daños sufridos**.

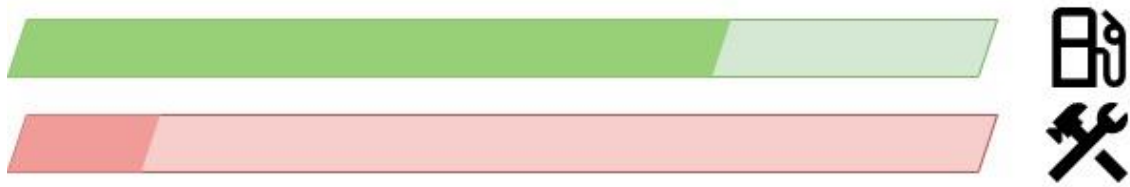


Ilustración 28. Sketch de las barras de combustible y daño

Son **representados en el HUD como barras horizontales o verticales**, según la preferencia del diseñador de UI y UX.

Al comenzar o reiniciar una prueba diaria (es decir, una fase -un tramo concreto o la etapa final-), el combustible del vehículo está siempre al 100%, mientras que el daño comienza al 0%.

---

## CONSUMO DE RECURSOS

### COMBUSTIBLE

---

Recordemos que comenzamos con 100 unidades de combustible (100%). El combustible se gasta de manera constante si se mantiene la velocidad, a un ritmo determinado por los programadores que llamaremos  $c$  y que se mide en unidades por segundo (uds/s).

- Durante el frenado, se consume  $1,5c$ .
- Durante la aceleración ligera, se consume  $1,5c$ .
- Durante la aceleración, se consume  $2c$ .
- Si el coche supera una determinada velocidad, se consume  $1,5c$ .

Estos ritmos están sujetos a cambios cuando se lleven a la implementación en el motor de juegos, deben estar balanceados y variar de manera consistente.

### DAÑOS

---

El daño inicia en 0 unidades de daño (0%). El daño se produce siempre que el coche colisione con el terreno, ya sea horizontal o verticalmente. Por ejemplo, tras caer por un salto, según la altura desde la que se caiga, se sufriría un determinado número de unidades de daño. No se definen ahora valores exactos, pues se requerirá de pruebas en el motor de juegos para decidir qué valores son más apropiados.



---

## QUEDARSE SIN RECURSOS

El consumo de todo el combustible o sufrir un daño irreparable supondrían (alcanzar el 0% o el 100%, respectivamente) la detención del coche y forzarían a reiniciar la prueba o salir de esta.

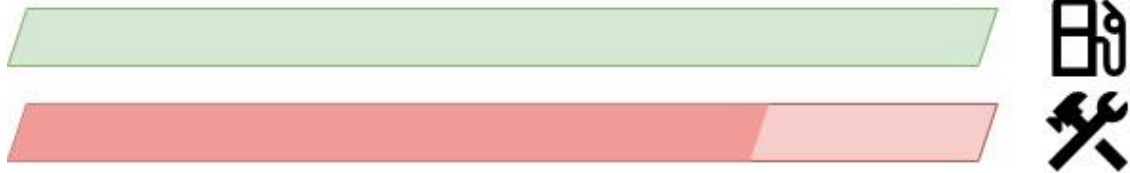


Ilustración 29. Sketch de los recursos: sin combustible

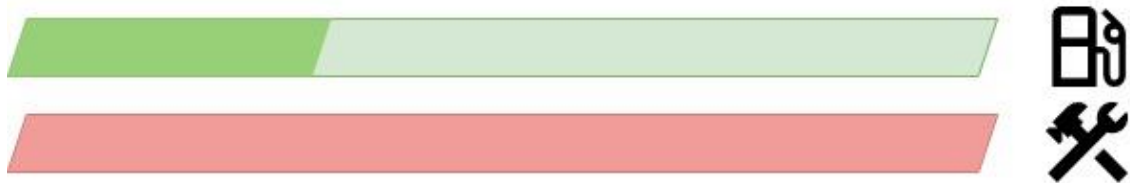


Ilustración 30. Sketch de los recursos: coche destrozado

---

## RECURSOS ENTRE FASES

Como **el coche se repara y recarga gasolina siempre que terminamos una prueba, se debe mostrar al jugador encima de cada día (o fase), cuánto daño sufrió y cuánto combustible consumió en su último intento guardado.**

Esto permite al jugador considerar si su desempeño en dicho tramo es apropiado para enfrentarse a la etapa completa. Recordemos que la etapa completa incluye todos los tramos de los días anteriores, y como se va a disponer de la misma cantidad de combustible y la misma resistencia que en los tramos separados, el jugador debe aprender no solo obtener buenos tiempos en los tramos, sino a hacerlo con un consumo de combustible y daños que sea proporcional a su dificultad y que lleve a una gestión óptima de recursos en la fase final.

## 5.4 CONTROLES Y PERIFÉRICOS



## 6 PROGRESIÓN

### 6.1 OBJETIVOS DEL JUGADOR

El objetivo del jugador es conseguir el mejor tiempo en cada etapa de *rally*, que presenta un trazado distinto y una ambientación única. Como se ha explicado con detenimiento en apartados anteriores, para lograr dicho tiempo, el jugador deberá ser capaz de seleccionar las instrucciones y el temperamento en cada sección que produzcan los mejores resultados.

Gestionar los recursos adecuadamente es crucial para llegar a la meta en primer lugar, por tanto, un manejo inteligente de estos puede dar mejores resultados.

### 6.2 ESTRUCTURA DEL JUEGO

El juego se divide en varios **eventos**, pero el progreso no es lineal, es decir, se pueden completar en el orden que se desee. La etapa de cada evento presenta su propio itinerario y características, no necesariamente suponiendo un incremento o decremento de dificultad. Son simplemente distintas pruebas que se pueden jugar en cualquier orden en función de la preferencia del jugador.

Para preparar al jugador de cara a una **etapa completa**, que se compone de **tres tramos cronometrados**, cada evento se divide **cuatro días ficticios o fases**:

- Los tres primeros días ponen al jugador en cada uno de los tramos en orden de aparición en la etapa. Una vez que se completa un día correspondiente a un tramo, se guarda el resultado en tiempo y recursos y se desbloquea el siguiente día si no ha sido desbloqueado ya. Se pueden intentar estos tramos repetidas veces antes o después de la etapa final, son pruebas preparatorias. Los tramos se componen de **diez secciones de pista**.
- Cuando se comienza la etapa completa el último día, los puntos de instrucción estarán ya por defecto elegidos tal y como quedaron en los tramos independientes (cuando se intentaron la última vez).

Los siguientes esquemas muestran la estructura de cada día o fase de tramo cronometrado y de un día o fase de etapa completa cronometrada:

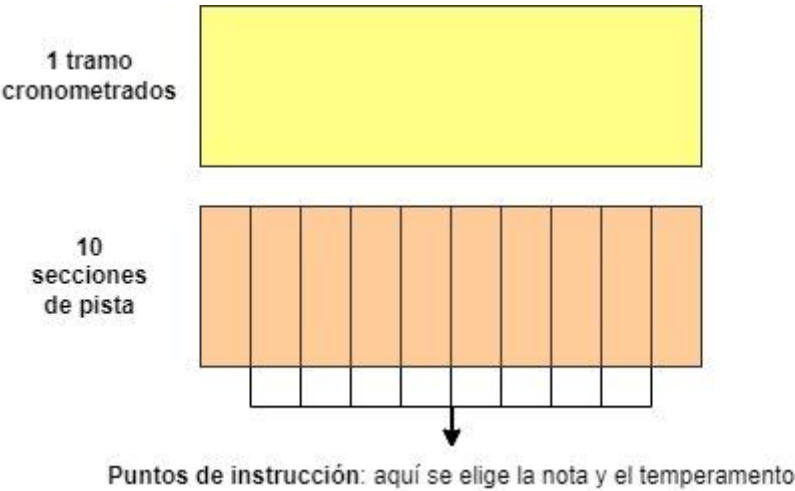


Ilustración 31. Estructura de un tramo cronometrado (días 1 a 3 -fases 1, 2 y 3-)

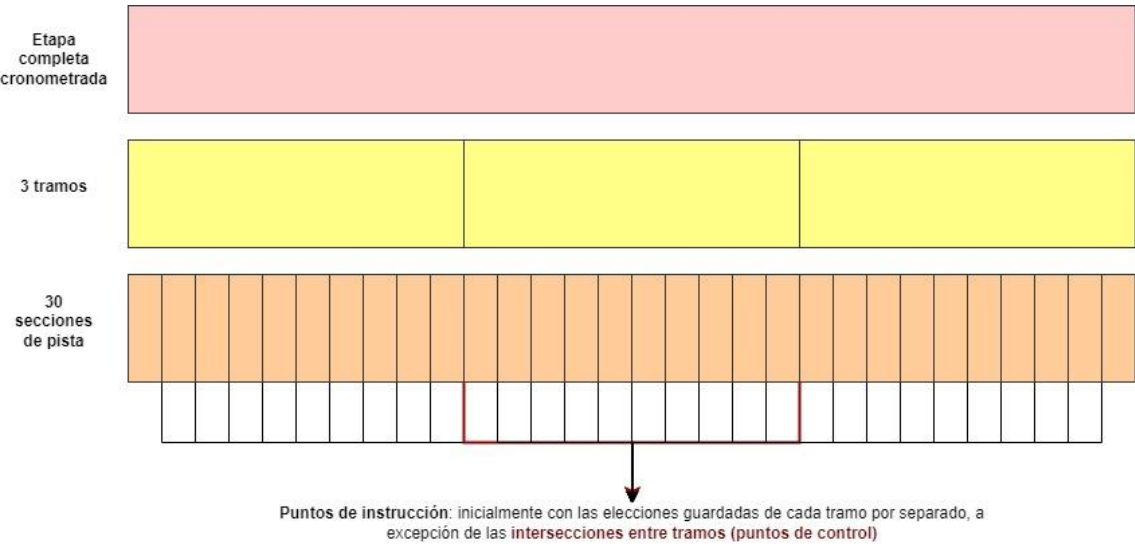


Ilustración 32., Estructura de una etapa completa cronometrada (último día -fase 4-)

A continuación, se presentan varios esquemas que ilustran la estructura del juego y de cada evento:

Estructura del juego

Supongamos que se ha completado C, se debe hacer visible al jugador. En todos los casos, se muestra el porcentaje de completitud, se podría reforzar el número con el borde del selector.

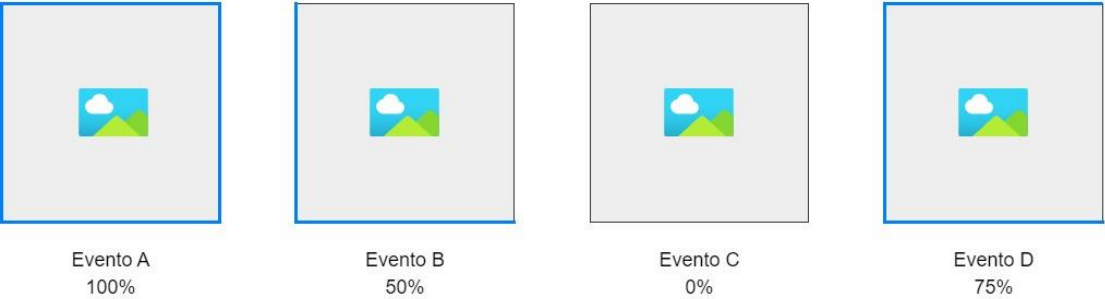


Ilustración 33. Estructura del juego

Estructura de un evento

Solo se puede acceder al día siguiente si se ha completado el anterior. Una vez completado un día, se guarda el tiempo, combustible gastado y daño sufrido en este. Se mostrará entonces, y cambiará cada vez que se vuelva a intentar, con el último resultado. En el caso de la etapa completa, se mostrará siempre el mejor resultado junto al último

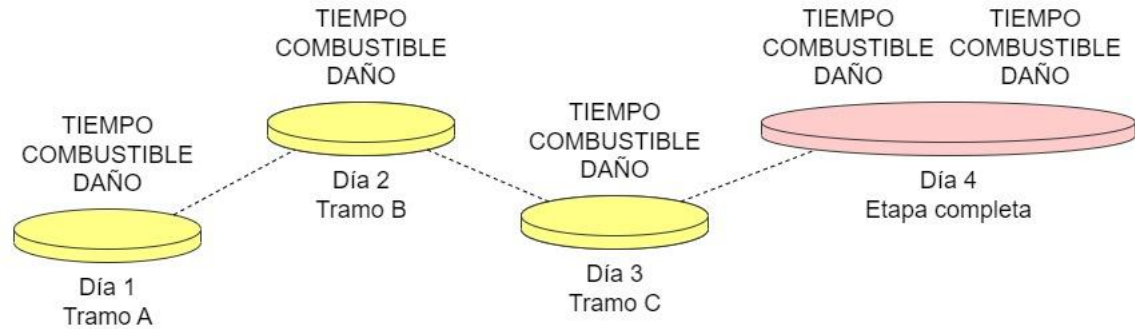


Ilustración 34. Estructura de un evento

## 6.3 CLASIFICACIÓN EN LÍNEA

## 7 ARTE

### 7.1 REFERENCIAS Y ESTÉTICA DEL JUEGO

### 7.2 ARTE FINAL

## 8 INTERFAZ DE USUARIO

### 8.1 REQUISITOS DE LA INTERFAZ

### 8.2 DIAGRAMA DE FLUJO DE NAVEGACIÓN

### 8.3 DISEÑO VISUAL DE LAS PANTALLAS

## 9 SONIDO

### 9.1 EFECTOS DE SONIDO

### 9.2 MÚSICA

## 10 PRODUCCIÓN

### 10.1 MODELO DE NEGOCIO DEL JUEGO Y PLAN DE FINANCIACIÓN

### 10.2 MARKETING

### 10.3 VERSIONES PRELIMINARES DEL PRODUCTO

### 10.4 PRODUCTO FINAL



## 11 CONCLUSIONES

## REFERENCIAS Y BIBLIOGRAFÍA

## ANEXO