

7 DE NOVIEMBRE 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



Universidad
Rey Juan Carlos

TABLA DE CONTENIDO

Índice de figuras.....	4
Índice de ilustraciones.....	4
Índice de tablas.....	7
1 Introducción	8
1.1 Introducción al juego y su enfoque al aprendizaje pensamiento computacional	8
1.2 Contenido del documento	8
2 Referencias y concepto de juego	9
2.1 Principales referencias.....	9
2.2 Concepto de juego y bases de la jugabilidad	13
2.2.1 Rol del jugador	13
2.2.2 Objetivo del jugador, jugabilidad y gestión de recursos.....	13
3 Enseñanza del Pensamiento Computacional a niños pequeños como objetivo.....	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	17
5 Cámara y perspectiva	19
6 Gameplay loop	20
6.1 Fase de reconocimiento.....	21
6.1.1 Puntos de control	21
6.1.2 Colocación de instrucciones en la pista	23
6.2 Fase de ejecución.....	24
6.2.1 Posibles eventos y sus consecuencias	25
6.2.2 Cantado de instrucciones durante la carrera	26
7 Control y comportamiento del vehículo	27
7.1 Relación entre tipos de sección de pista e instrucciones	27
7.1.1 Tipos de sección	28
7.2 Obstáculos	31
7.2.1 Obstáculos de relieve	31
7.2.2 Obstáculos del entorno	34
7.2.3 Concatenación de obstáculos	35
8 Gestión de recursos.....	36

8.1 Consumo de combustible	36
8.2 Sistema de daños	37
8.2.1 Daño por colisión con obstáculos	37
8.2.2 Daño por caída	38
8.2.3 Umbral mínimo para sufrir daños.....	39
8.3 Agotamiento de recursos	40
8.4 Recursos entre pruebas	40
9 Progresión.....	41
9.1 Objetivos del jugador	41
9.2 Estructura del juego.....	41
9.3 Clasificación en línea.....	44
10 Controles y periféricos.....	45
11 Interfaz de usuario	46
11.1 Requisitos de la interfaz.....	46
11.2 Diagramas de flujo de navegación	47
11.3 Diseño esquemático de la UI	49
12 Arte	55
12.1 Estética del juego	55
12.1.1 Referencias artísticas	55
12.1.2 Diseño visual del juego	56
12.2 Concept	64
12.2.1 Concepts del coche	64
12.2.2 Concepts de props.....	65
12.3 Assets 3D.....	70
12.4 Diseño visual de la UI	71
13 Sonido	72
13.1 Efectos de sonido.....	72
13.2 Música	73
14 Producción.....	74
14.1 Modelo de negocio del juego y plan de financiación	74
14.1.1 Información sobre el usuario	74
14.1.2 Mapa de Empatía	75
14.1.3 Caja de Herramientas.....	76
14.1.4 Modelo de Lienzo	77
14.2 Marketing	78
Conclusiones	79
Glosario.....	80

Terminología del juego	80
Acciones dentro del juego.....	82
Historial de versiones.....	83
Versión 0.1 (26/09/23)	83
Versión 0.2 (28/09/23)	83
Versión 0.3 (05/10/23)	83
Versión 0.4 (08/10/23)	83
Versión 0.6 (12/10/23)	84
Versión 0.7 (17/10/23)	84
Versión 0.8 (20/10/23)	84
Versión 1.0 (22/10/23)	85
Versión 1.1 (03/11/23)	85
Versión 1.2 (07/11/23)	85

ÍNDICE DE FIGURAS

ÍNDICE DE ILUSTRACIONES

Ilustración 1. Colin McRae Rally 3 (PS2)	9
Ilustración 2. DiRT Rally (PS4).....	9
Ilustración 3. WRC 10 (PC)	10
Ilustración 4. Virtual Racing (Arcade)	10
Ilustración 5. Ridge Racer (Arcade)	11
Ilustración 6. Daytona USA (Arcade).....	11
Ilustración 7. Art of Rally (PC)	12
Ilustración 8. You Suck at Parking (Xbox One)	12
Ilustración 9. WRC: Audi Quattro de Walter Röhrl en 1985	16
Ilustración 10. WRC: Toyota Celica GT-Four de Carlos Sainz en 1990	17
Ilustración 11. Perspectiva y movimiento de la cámara	19
Ilustración 12. Gameplay loop de un tramo	20
Ilustración 13. Gameplay loop de una etapa completa	20
Ilustración 14. Leyenda para diagramas de flujo sobre el gameplay	21
Ilustración 15. Gameplay loop de la fase de reconocimiento.....	22
Ilustración 16. Gameplay loop de la fase de ejecución.....	25
Ilustración 17. Leyenda para el comportamiento del vehículo	27
Ilustración 18. Iconos de las instrucciones de tipo de trazado	27
Ilustración 19. Comportamiento del coche en recta	28
Ilustración 20. Comportamiento del coche en curva poco cerrada	29
Ilustración 21. Comportamiento del coche en curva cerrada.....	29
Ilustración 22. Comportamiento del vehículo en curva muy cerrada	30
Ilustración 23. Comportamiento del coche en horquilla.....	30
Ilustración 24. Sketch de una rampa	31
Ilustración 25. Sketch de una meseta	32
Ilustración 26. Sketch de un valle	32
Ilustración 27. Sketch de un hundimiento	33
Ilustración 28. Sketch de un charco	33
Ilustración 29. Sketch de un bache seguido de un charco en una curva poco cerrada	35
Ilustración 30. Sketch de las barras de combustible y daño	36
Ilustración 31. Vectores de impacto y dirección.....	37
Ilustración 32. Sketch de las barras recursos: sin combustible	40
Ilustración 33. Sketch de las barras recursos: coche destrozado.....	40

Ilustración 34. Estructura de un tramo cronometrado (días/pruebas 1 a 3).....	42
Ilustración 35. Estructura de una etapa completa cronometrada (último día/prueba)	42
Ilustración 36. Estructura del juego	43
Ilustración 37. Estructura de un evento	43
Ilustración 38. Diagrama de navegación de inicio del juego	47
Ilustración 39. Diagrama de navegación: menús y juego	48
Ilustración 40. Diseño de la Splash Screen con el logo de la empresa.....	49
Ilustración 41. Diseño de la selección de idioma (primera vez que se inicia el juego)	49
Ilustración 42. Diseño de la pantalla de inicio	50
Ilustración 43. Diseño del pop-up de inicio de sesión, si no se ha iniciado previamente en el dispositivo	50
Ilustración 44. Diseño del pop-up de registro	51
Ilustración 45. Diseño del pop-up de inicio de sesión	51
Ilustración 46. Diseño del menú principal	52
Ilustración 47. Diseños de los pop-ups de tienda, garaje y ajustes, respectivamente	52
Ilustración 48. Diseño de la pantalla de selección de etapa	53
Ilustración 49. Diseño de la pantalla de selección de tramo en la etapa	53
Ilustración 50. Representación del flujo de navegación en Figma	54
Ilustración 51. Ridge Racer (Namco, 1993).....	55
Ilustración 52. Art of Rally (Funselektor Labs, 2020)	55
Ilustración 53. Color primario escogido	56
Ilustración 54. Paleta de sombras del color primario	56
Ilustración 55. Color primario y su complementario	56
Ilustración 56. Color secundario escogido	57
Ilustración 57. Paleta de sombras del color secundario	57
Ilustración 58. Color secundario y su complementario	57
Ilustración 59. Negro no puro escogido	58
Ilustración 60. Blanco no puro escogido	58
Ilustración 61. Paleta base de RTT	58
Ilustración 62. Muestra de mubenk regular.....	59
Ilustración 63. Muestra de Thecnic, ThecnicBold y ThecnicLite.....	59
Ilustración 64. Logo de RTT.....	60
Ilustración 65. Ejemplo de estilo del marco de un menú emergente.....	61
Ilustración 66. Ejemplo de estilo de un botón normal	61

Ilustración 67. Ejemplo de estilo de botones especiales superpuestos a un marco de ventana emergente	62
Ilustración 68. Ejemplo de estilo de pantalla de carga.....	62
Ilustración 69. Ejemplo de estilo de elementos del HUD.....	63
Ilustración 70. Concept de coche en perspectiva	64
Ilustración 71. Concept de coche de perfil.....	65
Ilustración 72. Concept de un punto de control	65
Ilustración 73. Concept de la salida	66
Ilustración 74. Concept de diferentes variantes de árbol	66
Ilustración 75. Concept de diferentes rocas.....	67
Ilustración 76. Concept de un barril.....	67
Ilustración 77. Concept de una pila de ruedas.....	68
Ilustración 78. Concept de una valla de obra	68
Ilustración 79. Concept de un cono de tráfico	69
Ilustración 80. Mapa de Empatía del usuario final	75
Ilustración 81. Caja de Herramientas	76
Ilustración 82. Canvas del modelo de negocio	77

ÍNDICE DE TABLAS

Tabla 1. Destrezas del Pensamiento Computacional en el juego	15
Tabla 2. Consumo de combustible en función del comportamiento del coche.....	36
Tabla 3. Glosario de terminología del juego	82
Tabla 4. Glosario de acciones dentro del juego	82

1 INTRODUCCIÓN

Este documento cubre todos los aspectos referentes al diseño del juego *Rally Team Tactics (RTT)*, desarrollado por *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 volver al inicio. 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
- Proceso de diseño del juego en todos sus ámbitos:
 - Trasfondo
 - Mecánicas
 - Progresión
 - Interfaces y experiencia de usuario (UI y UX)
 - Arte
 - Música y sonido
- 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 puzzles 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 instrucciones, que describen el trazado (notas) y aconsejan un temperamento concreto**. 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 instrucciones “cantemos” determinarán el comportamiento de este. Para situar las instrucciones 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**. Durante los tres primeros días de un evento, **el jugador se debe enfrentar a pruebas cronometradas de cada tramo por separado**, refinando su estrategia de cara al día de la prueba de 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)**.

3 ENSEÑANZA DEL PENSAMIENTO COMPUTACIONAL A NIÑOS PEQUEÑOS COMO OBJETIVO

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 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
Abstracción	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 capa superior de planificación y toma de decisiones . La “vista de pájaro” pone énfasis en esta capacidad de ver el problema en su totalidad y reconocer así sus elementos clave .
Pensamiento algorítmico	En RTT se pone a disposición del jugador un conjunto de notas y un <i>slider</i> de temperamento que determinan el comportamiento del coche para cada sección de una prueba. Se debe crear un algoritmo , al fin y al cabo, una secuencia de acciones concretas que da lugar al mejor resultado posible.
Descomposición	En base a la respuesta del vehículo en un determinado tramo con una determinada serie de instrucciones, el jugador puede adquirir

	la capacidad de deducir cómo se va a comportar el coche en el mismo escenario u otro similar en función de sus decisiones en la fase de reconocimiento.
Evaluación	<p>El <i>RTT</i> se presta especial atención a la gestión de recursos, 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 detección de fallos, y toma decisiones ajustadas al objetivo. 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>
Generalización	<p><i>RTT</i> pone el foco en la repetición para encontrar soluciones mejores. Resolver el problema en base a soluciones anteriores es parte fundamental del diseño. Cuando el jugador encuentra patrones y similitudes 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

RTT está ambientado en la actualidad, y el jugador viajará alrededor del mundo compitiendo cara a cara con los mejores pilotos.

El mundo de *Rally Team Tactics* es una realidad alternativa donde la época dorada de los *rallies* nunca cesó y esta competición automovilística mantiene una tremenda popularidad.

Se suele considerar que la época dorada del rally fue la década de los 80, cuando se creó la categoría de Grupo B, que permitía a los fabricantes desarrollar coches extremadamente potentes y veloces, y las restricciones de velocidad y técnicas eran mínimas. Las carreras eran espectaculares y peligrosas: los pilotos se ponían en riesgo su vida en cada etapa y las masas de espectadores se agolpaban al borde de las carreteras para ver a sus ídolos de cerca. Algunos de los modelos más emblemáticos de esta época fueron el Audi Quattro, el Lancia Delta S4, el Peugeot 205 T16 o el Ford RS200.



Ilustración 9. WRC: Audi Quattro de Walter Röhrl en 1985



Ilustración 10. WRC: Toyota Celica GT-Four de Carlos Sainz en 1990

4.2 NARRATIVA

El varias veces campeón del mundo de rally **Erik Ström** estaba en el pico de su carrera. Su destreza y agresividad en la carretera lo hacían casi imparable.

Pese a haber crecido en una familia humilde, el talento de Ström lo había llevado de la pobreza a ser una de las figuras más admiradas y reconocibles del mundo del automovilismo.

No obstante, su escalada al zenit no fue sin consecuencias para su salud mental. Tras una serie de derrotas en etapas donde era el favorito, Ström no era capaz de concentrarse en la pista. Jóvenes pilotos estaban despuntando, y Ström no tenía un coche competitivo y estaba harto de seguir las instrucciones de los ingenieros, a los que consideraba ineptos. Las averías eran demasiado frecuentes, y la actuación cautelosa del equipo estaba acabando con su paciencia. Desesperado, firmó con un constructor dispuesto a dejarle desobedecer las órdenes del equipo, incluido su copiloto, si lo veía necesario. Además, sabía que su nuevo coche no cumplía con la normativa, y que las piezas prohibidas habían sido camufladas y para pasar irregularmente las comprobaciones de seguridad.

La temporada de rally de 2013 comenzó bien para Ström, posicionándose entre los tres primeros justo antes de una de las etapas más temidas del circuito anual: la *Acropolis Rally of Greece*. Terrenos montañosos, acantilados rocosos y calor sofocante sobre un asfalto antiguo y unas secciones campo a través desafiantes. No era la etapa para la audacia ni el riesgo, pero Ström no dudó en jugarse el campeonato en tierras griegas.

El día de la carrera, el piloto sueco estaba codo a codo con los mejores tiempos marcados por sus rivales más cercanos en la clasificación, y decidió acelerar en uno de los tramos más infames de la etapa: la curva del abismo. Fue aquí donde Ström, cegado por la ambición, no frenó a tiempo y cayó por el barranco. Cuando los equipos médicos

le encontraron, Erik, inconsciente, no respondía. Pero la suerte le había sonreído, pues no había sido un accidente fatal.

Sin embargo, esto le supondría su ingreso hospitalario, donde estaría años en rehabilitación, sometiéndose a numerosas operaciones, hasta que pudo recuperar parte de la movilidad. Su tren inferior no era funcional, nunca podría volver a competir. Su tiempo de recuperación le llevó a replantearse su vida y su carrera profesional: sabía que no podría volver a competir, y no encontraba su propósito para seguir adelante. Fueron años oscuros, pero Ström superó la depresión y, tras finalizar su rehabilitación, regresó al mundo de los *rallies*. Su discapacidad le impedía conducir, pero sabía que tenía mucho conocimiento que aportar a las nuevas generaciones.

En 2019, fundó una escuela en su ciudad natal y hizo de mentor para decenas de aspirantes a futuros pilotos. Se aseguraba también de que los jóvenes pusiesen su salud mental y física por encima de cualquier objetivo en la pista.

Ström era una figura respetada en el automovilismo y los rumores de su vuelta a los *rallies* se multiplicaban, pero nadie sabía qué rol tomaría de ser así. A sus 56 años, en 2022, se asoció con algunos de sus amigos y antiguos conocidos de la industria para montar un equipo y firmar un contrato con una reputada constructora. Sus coches llevaban los colores de su país natal en su carrocería.

Erik tenía claro cuál era su sitio ahora: el de copiloto. Quería redimirse por todos sus errores y llevar el talento de su escuela a lo más alto a base de aprendizaje y constancia.

Dado el pistoletazo de salida a la temporada 2023, Ström acaparaba todas las portadas: ¿será este el verdadero regreso del rey de los *rallies*?

5 CÁMARA Y PERSPECTIVA

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

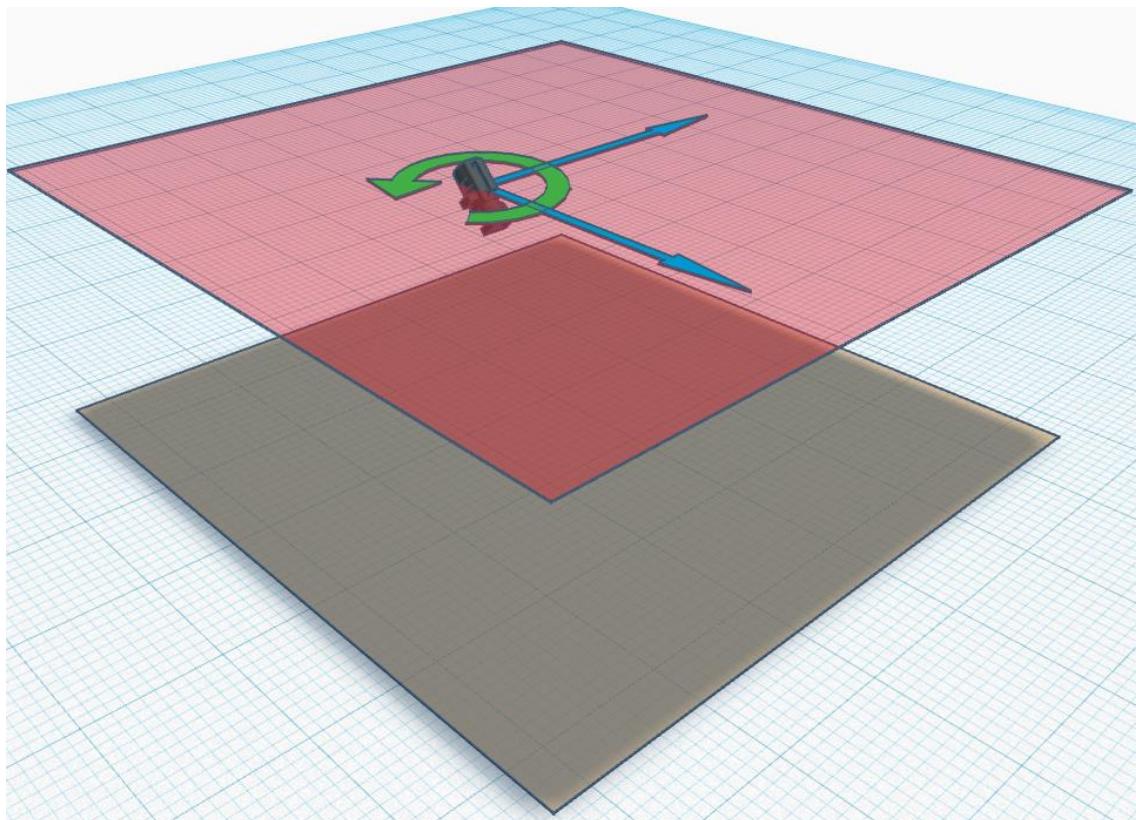


Ilustración 11. Perspectiva y movimiento de la cámara

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 11).

6 GAMEPLAY LOOP

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

La jugabilidad de RTT se divide en dos fases o turnos: reconocimiento y ejecución. Esto es igual en los días de pruebas preparación (los distintos tramos) y en el día de prueba final (la etapa completa). Los diagramas de alto nivel para tramos y etapas completas se presentan a continuación:

Gameplay Loop de un tramo

En un tramo los jugadores solo pueden elegir instrucciones para diez secciones de pista, que determinan el tramo. Pueden buscar mejorar su tiempo en este tramo para prepararse para la etapa final

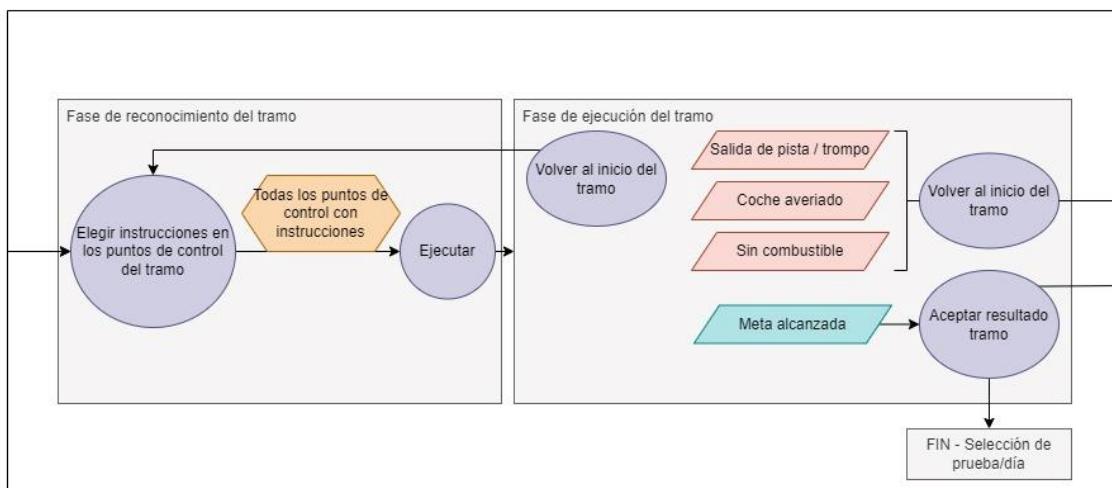


Ilustración 12. Gameplay loop de un tramo

Gameplay Loop de una etapa completa

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

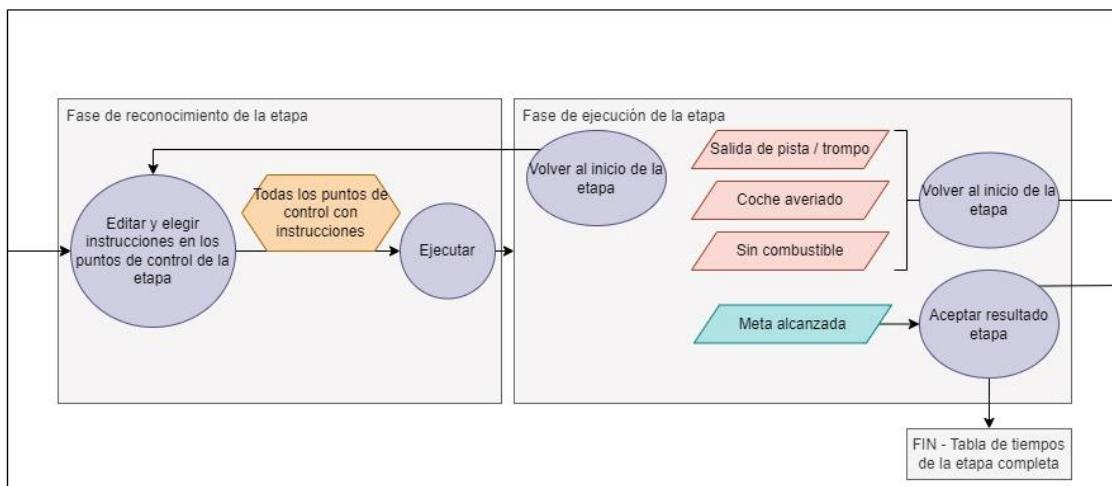


Ilustración 13. Gameplay loop de una etapa completa

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



Ilustración 14. Leyenda para diagramas de flujo sobre el gameplay

6.1 FASE DE RECONOCIMIENTO

Como se plasma en el diagrama Ilustración 15, el jugador puede elegir instrucciones durante la **fase de reconocimiento**. Una vez se hayan seleccionado las instrucciones deseadas para todos los puntos de control, se puede ejecutar.

Durante el reconocimiento, el jugador puede navegar con libertad tal como se ha descrito en [5 Cámara y perspectiva](#), y, además:



6.1.1 PUNTOS DE CONTROL

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

- Seleccionar la nota de código nemotécnico** que ve más conveniente en función de el trazado inmediatamente posterior al punto de control. Entre estas notas encontramos, por ejemplo, “curva muy cerrada” o “recta”.
- Elegir el temperamento** con el que realizar la acción. Esto es posible gracias a un *slider continuo* que permite determinar si se quiere un acercamiento **más cauteloso o más agresivo**.
 - Por defecto, cuando se elige una nota por primera vez en un punto, el *slider* se encuentra en la posición intermedia (temperamento completamente moderado).
 - 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 (ver [7 Control y comportamiento del vehículo](#)).

C. Deseleccionar un punto de control

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 elegido instrucciones en todos los puntos de control.

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:

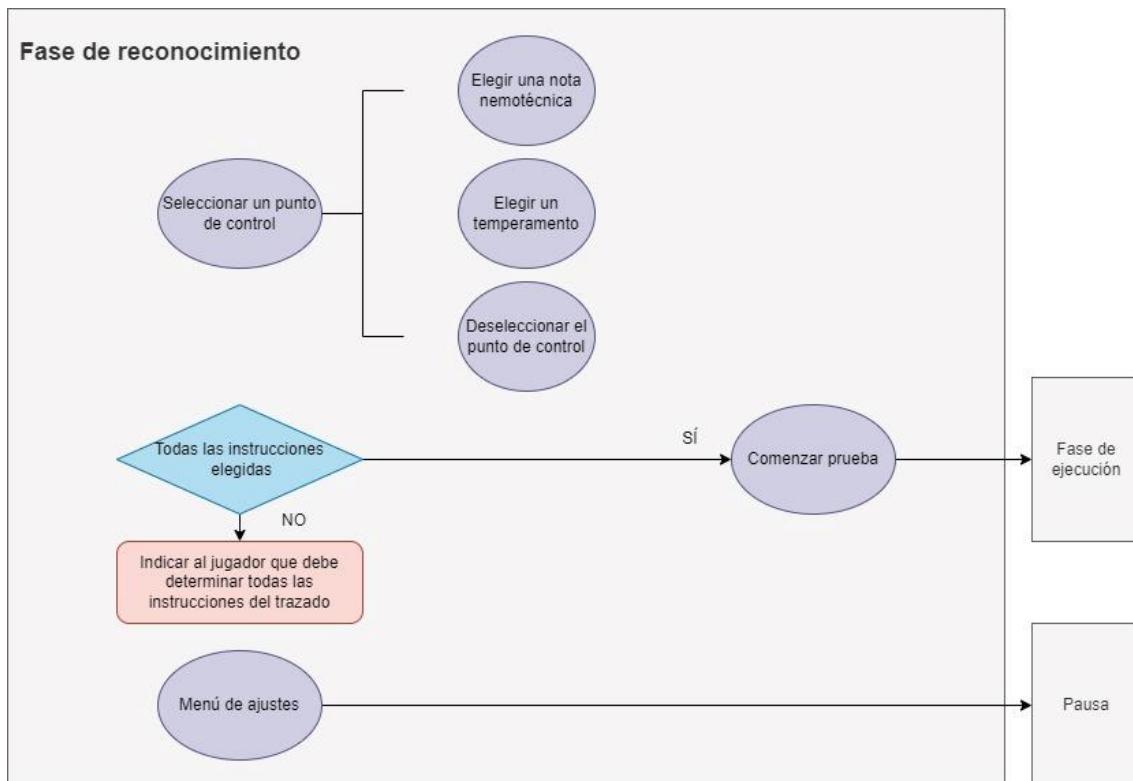


Ilustración 15. Gameplay loop de la fase de reconocimiento

Ver la leyenda de la Ilustración 14 para el significado de las diferentes formas.

6.1.2 COLOCACIÓN DE INSTRUCCIONES EN LA PISTA

En este apartado se hace énfasis en el sistema de selección de puntos de control. Los puntos de control deben quedar claramente señalizados en la pista, y deben proveerse opciones de configuración a los jugadores para que los puedan distinguir con facilidad. Se explican a continuación cómo se plantea esta cuestión en función de la fase de juego en la que se encuentre el jugador:

- El uso de banderas a los laterales de la pista es un marcador diegético de la posición de estos puntos de control, que recordemos se encuentran entre secciones de pista cuya separación el jugador no puede discernir. **Esta señalización se mantiene entre fases de reconocimiento y ejecución**, ya que es parte del mundo de juego.
- De forma extradiegética, se puede mostrar un muro luminoso perpendicular al trazado en esos puntos para hacerlos más notables. Además, el color puede variar si este ha sido elegido ya o si está vacío aún. **Este muro solo es visible en fase de reconocimiento**.
- También de forma extradiegética, aquellos puntos de control donde se ha seleccionado instrucción (nota y temperamento) pueden mostrar la elección de manera flotante o sobre la pista. El temperamento recordemos está controlado por un *slider*, pero el ícono puede variar únicamente al superar un determinado valor en dicho *slider*. Por ejemplo, la barra se dividiría en tres partes, y el ícono solo cambiaría cuando se pasase a un tercio distinto. Como el muro luminoso, **estos iconos son únicamente visibles en fase de reconocimiento**.

6.2 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 con las instrucciones situadas en reconocimiento que “canta” el piloto. Durante la ejecución seguimos teniendo el control de la cámara y podemos reiniciar en cualquier momento, volviendo a la fase de reconocimiento.

Durante la fase de ejecución, se simula la prueba de *rally* de acuerdo con 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 instrucciones 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 instrucciones elegidas previamente para cada punto de control

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:

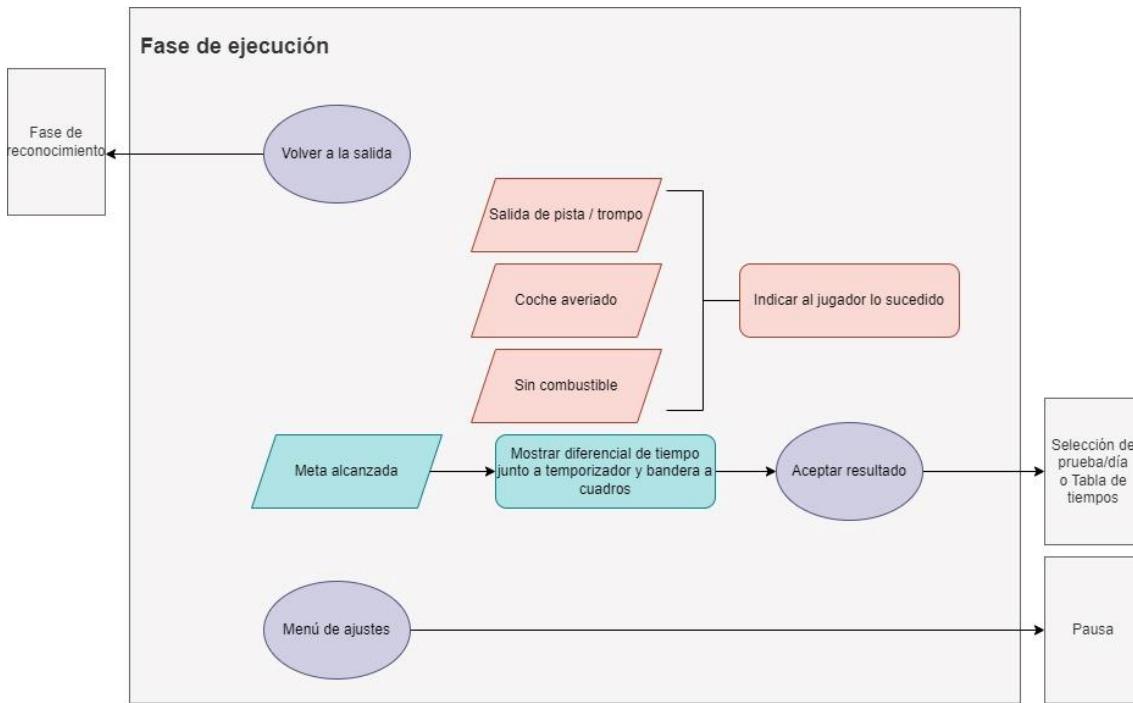


Ilustración 16. Gameplay loop de la fase de ejecución

Ver la leyenda de la Ilustración 14 para el significado de las diferentes formas.

6.2.1 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 final del tramo 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 prueba del evento que se está disputando.

6.2.2 CANTADO DE INSTRUCCIONES DURANTE LA CARRERA

Al igual que para el reconocimiento, este apartado trata cómo se hacen visibles los puntos de control al jugador, pero esta vez durante la fase de ejecución.

- Las banderas físicas que marcan los puntos de control siguen presentes en esta etapa, son elementos diegéticos.
- Los muros luminosos ya no están presentes, ni tampoco los iconos de nota y temperamento, es decir, ningún elemento extradiegético flotante se ve.
- Para que el jugador tenga un refuerzo sobre las decisiones tomadas en la fase de reconocimiento, cada vez que el vehículo pase por un punto de control, **el copiloto (voiceover) “canta” las instrucciones**, no solo diciendo el nombre de esta, sino también el grado de agresividad determinado por el temperamento. Al ser el temperamento continuo y no discreto, como en la fase de reconocimiento, se tiene que discretizar en varios estados. Estos estados corresponden con los tercios del *slider*, siendo estos equivalentes a acompañar la nota en el “cantado” de la instrucción de expresiones, por ejemplo: “curva cerrada, tomar con cuidado”, “recta, recorrer con agresividad” o “curva muy cerrada, acometer con moderación”.

Otra forma de recordar al jugador durante la fase de ejecución las instrucciones elegidas sería **mostrarlas además en el HUD como iconos** al tiempo que son “cantadas” (esto podría ser configurable).

7 CONTROL Y COMPORTAMIENTO DEL VEHÍCULO

Se van a explicar a continuación el control y el comportamiento del vehículo ante las diferentes curvas y obstáculos si se elige su nota de código nemotécnico 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. También se elige instrucción en el punto de inicio para la primera sección, naturalmente.

7.1 RELACIÓN ENTRE TIPOS DE SECCIÓN DE PISTA E INSTRUCCIONES

A continuación, se muestran bocetos de todos los tipos de sección y se explica cómo se comportaría el vehículo de ser 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:

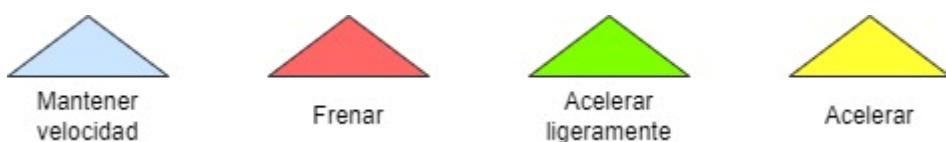


Ilustración 17. Leyenda para el comportamiento del vehículo

A diferencia de la aceleración, que puede ser más suave o más repentina, no existen variantes de frenado, solo un tipo que supone una reducción gradual de la velocidad.

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

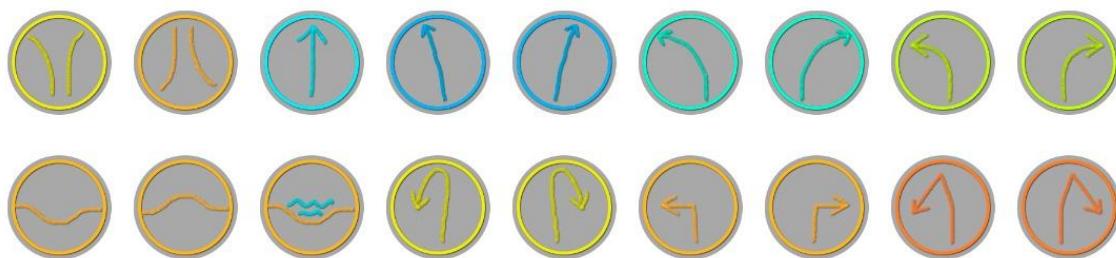


Ilustración 18. Iconos de las instrucciones de tipo de trazado

7.1.1 TIPOS DE SECCIÓN

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.

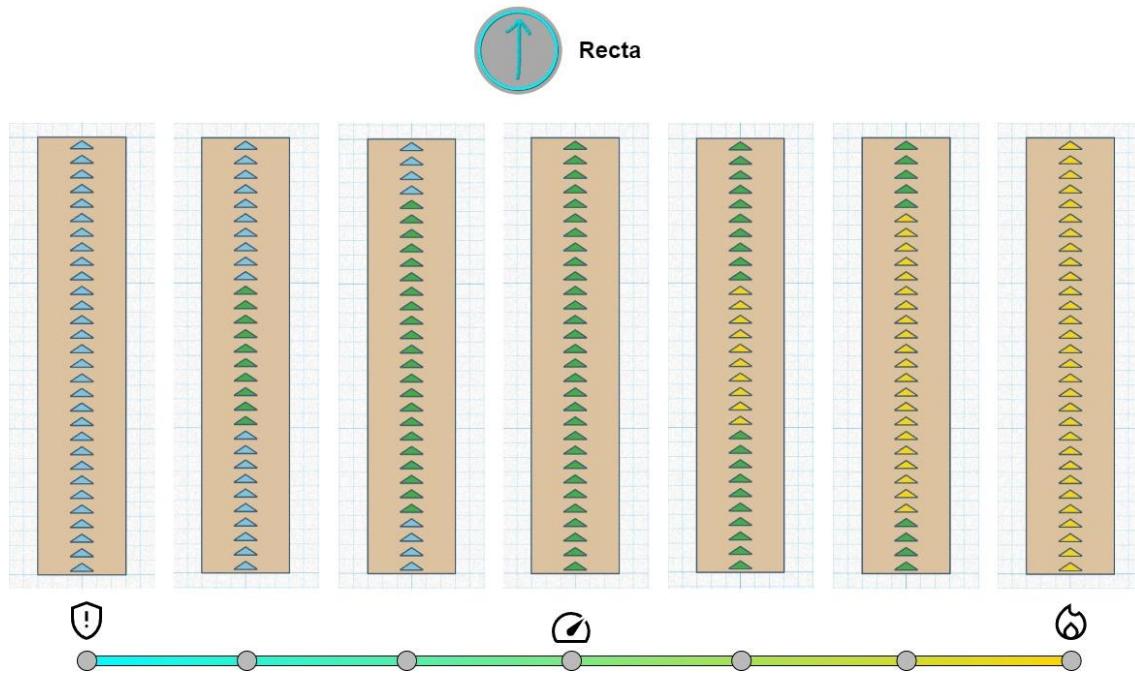


Ilustración 19. Comportamiento del coche en recta

Como se observa, a medida que se toma un perfil más agresivo, la aceleración es más fuerte y prolongada.

CURVA POCO CERRADA

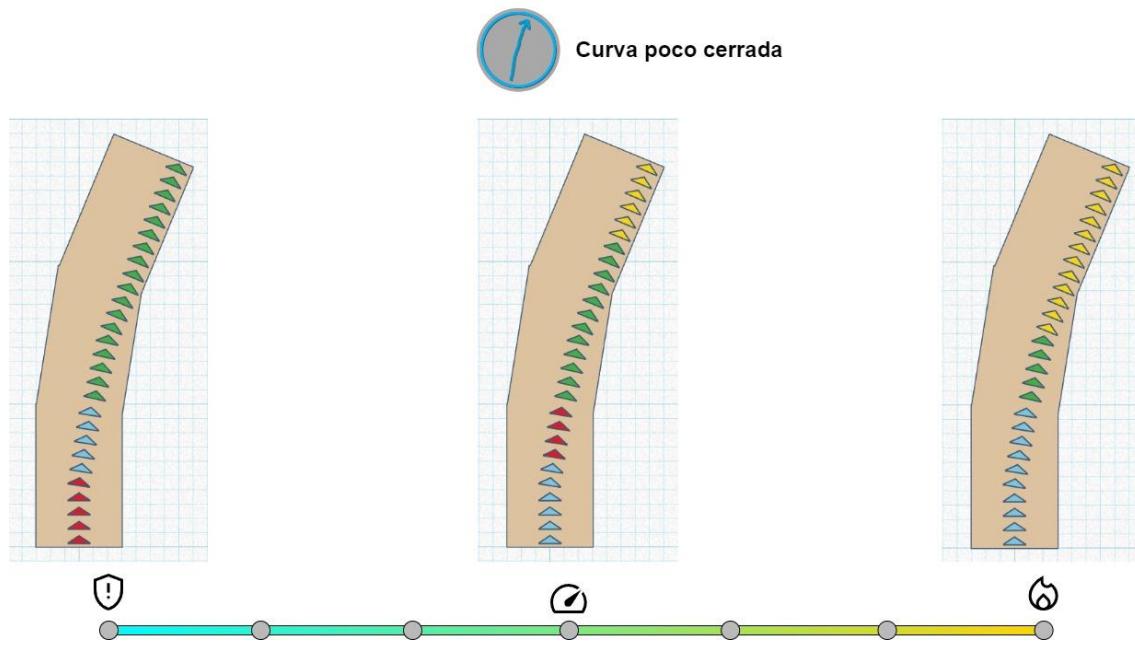


Ilustración 20. Comportamiento del coche en curva poco cerrada

Aquí se observa por primera vez la presencia de tramos de frenado.

CURVA CERRADA

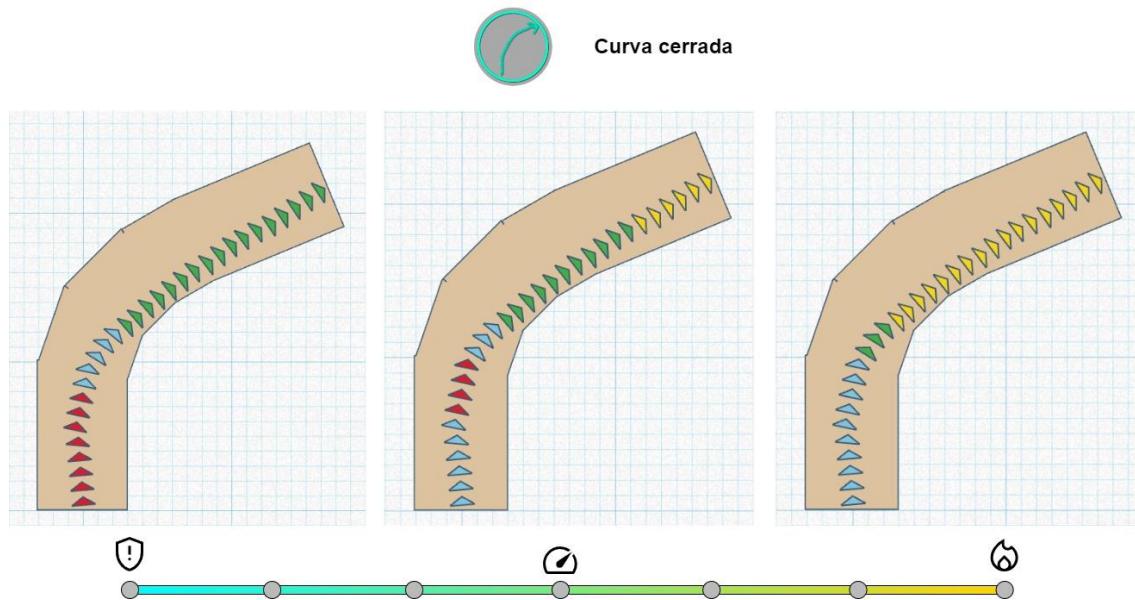


Ilustración 21. Comportamiento del coche en curva cerrada

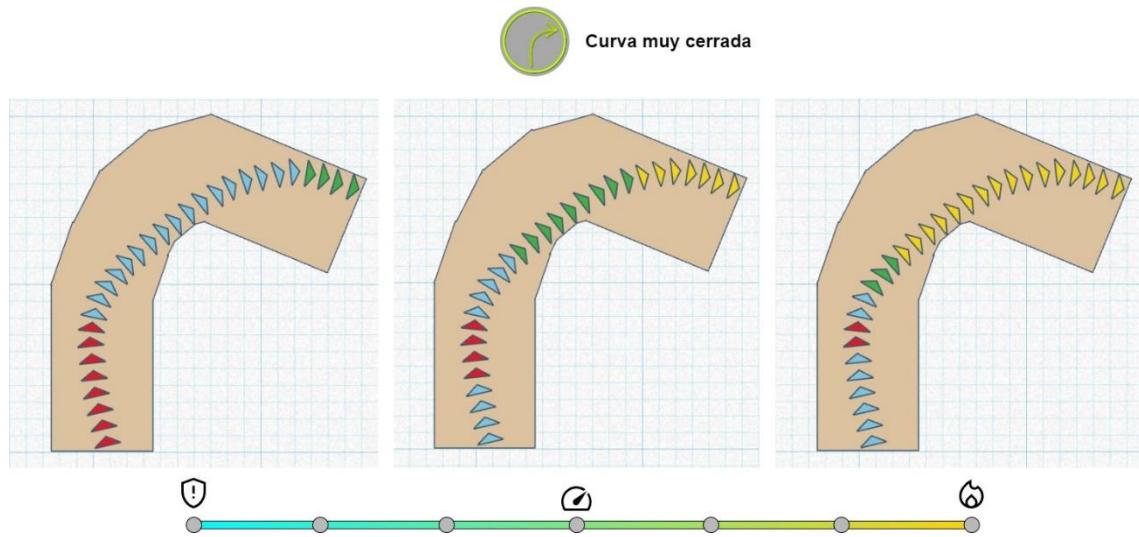
CURVA MUY CERRADA

Ilustración 22. Comportamiento del vehículo en curva muy cerrada

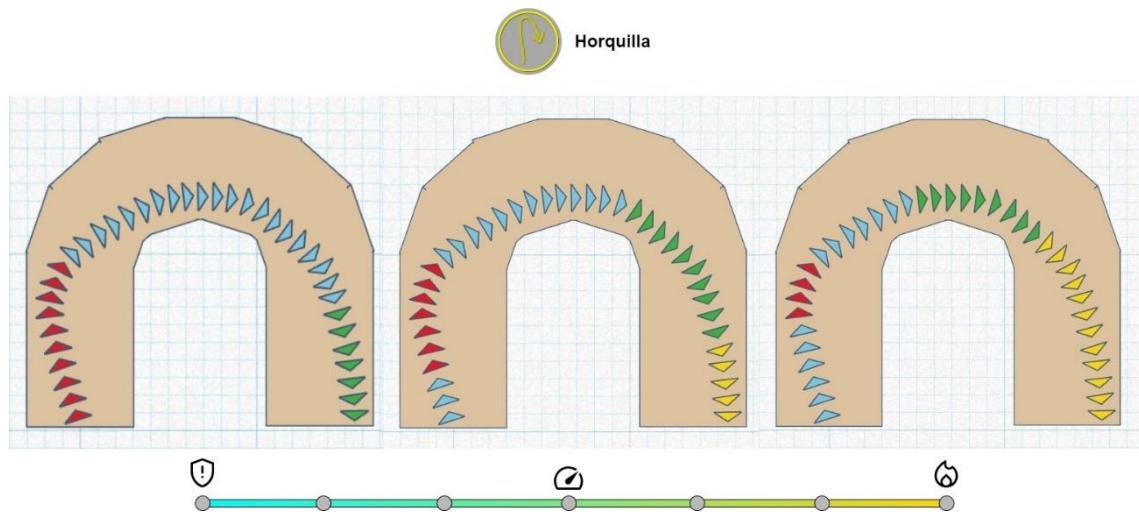
HORQUILLA

Ilustración 23. Comportamiento del coche en horquilla

7.2 OBSTÁCULOS

7.2.1 OBSTÁCULOS DE RELIEVE

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**.

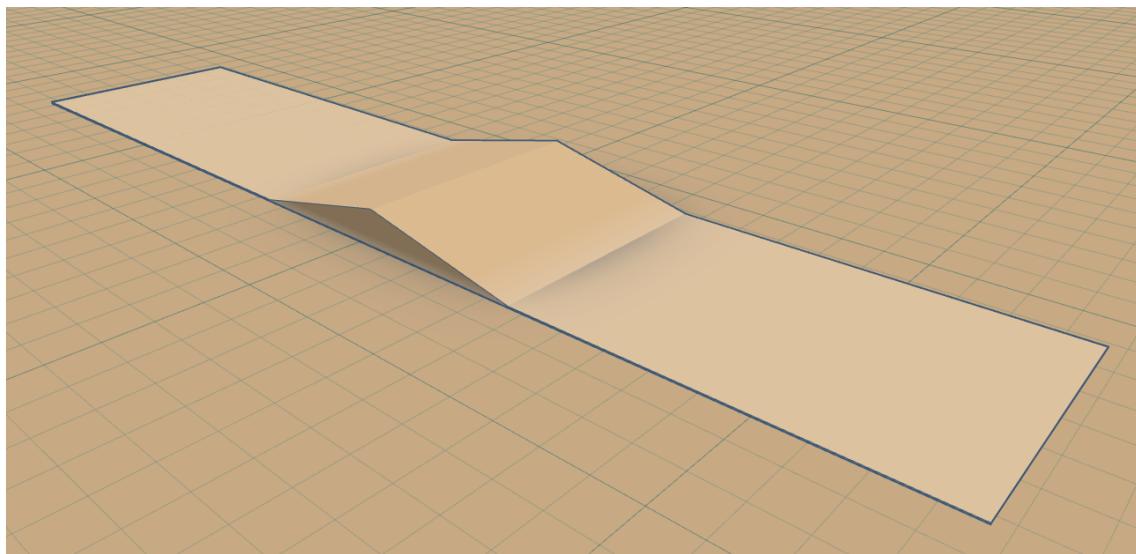


Ilustración 24. Sketch de una rampa

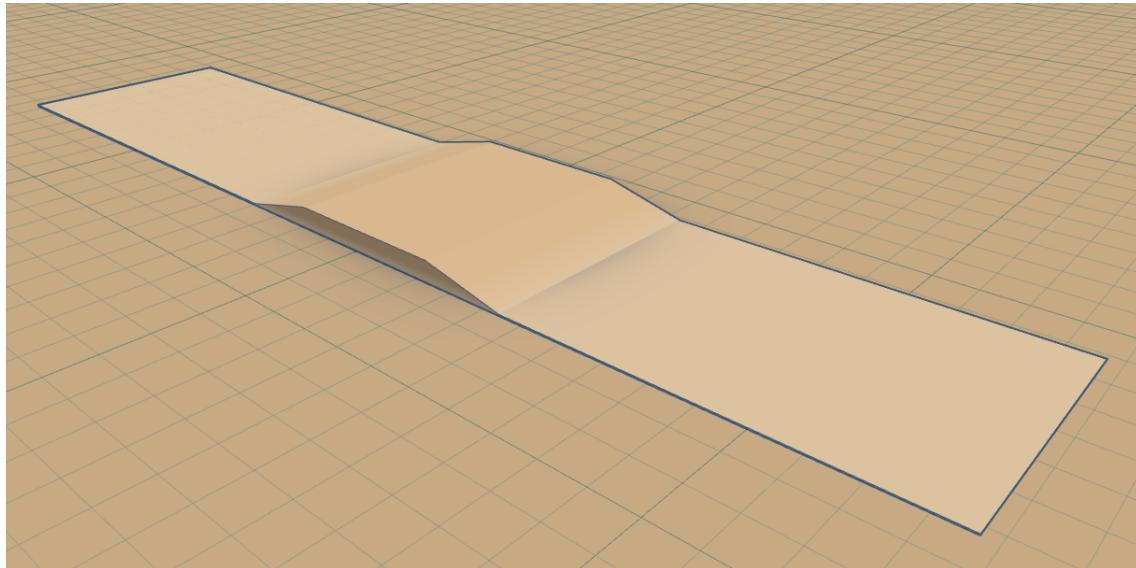


Ilustración 25. Sketch de una meseta

- 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**.

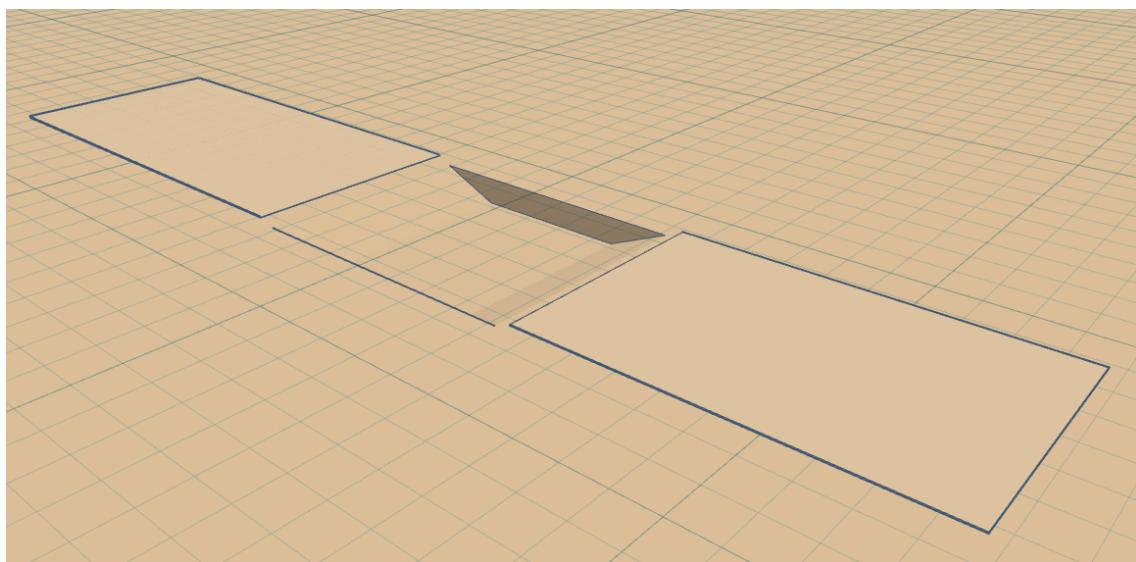


Ilustración 26. Sketch de un valle

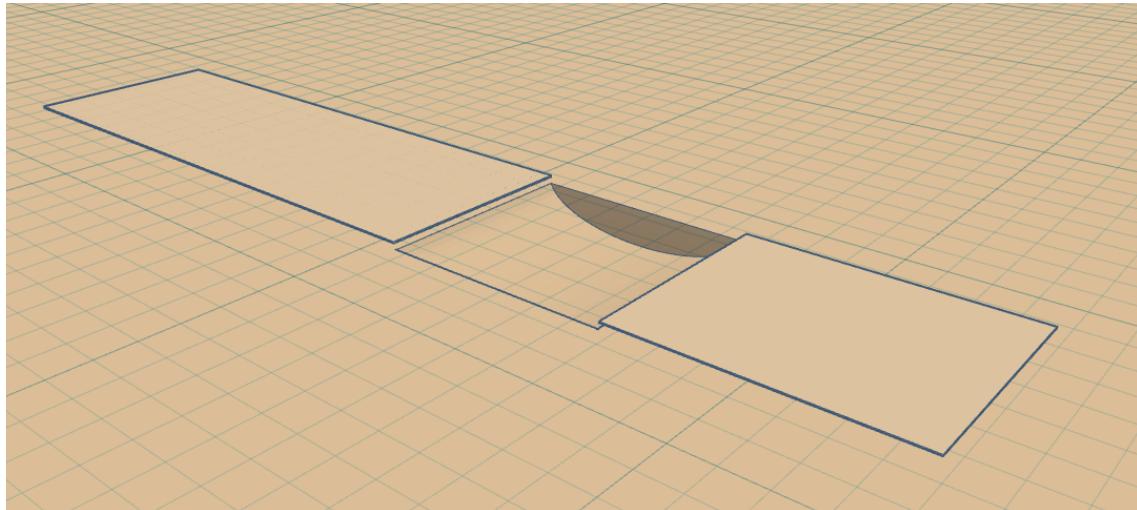


Ilustración 27. Sketch de un hundimiento

- 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.

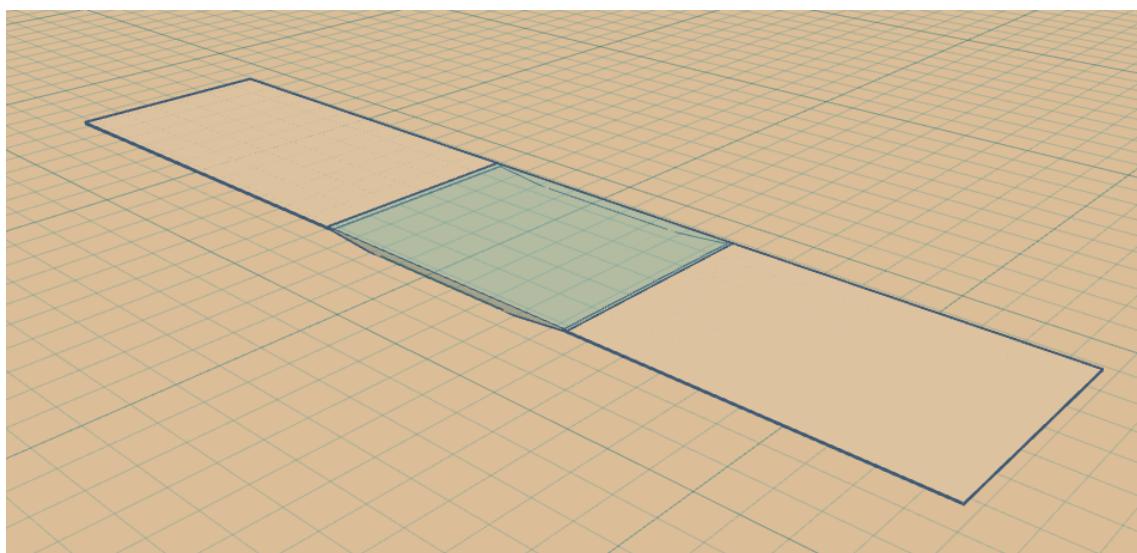


Ilustración 28. Sketch de un charco

7.2.2 OBSTÁCULOS DEL ENTORNO

7.2.3 CONCATENACIÓN DE 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:

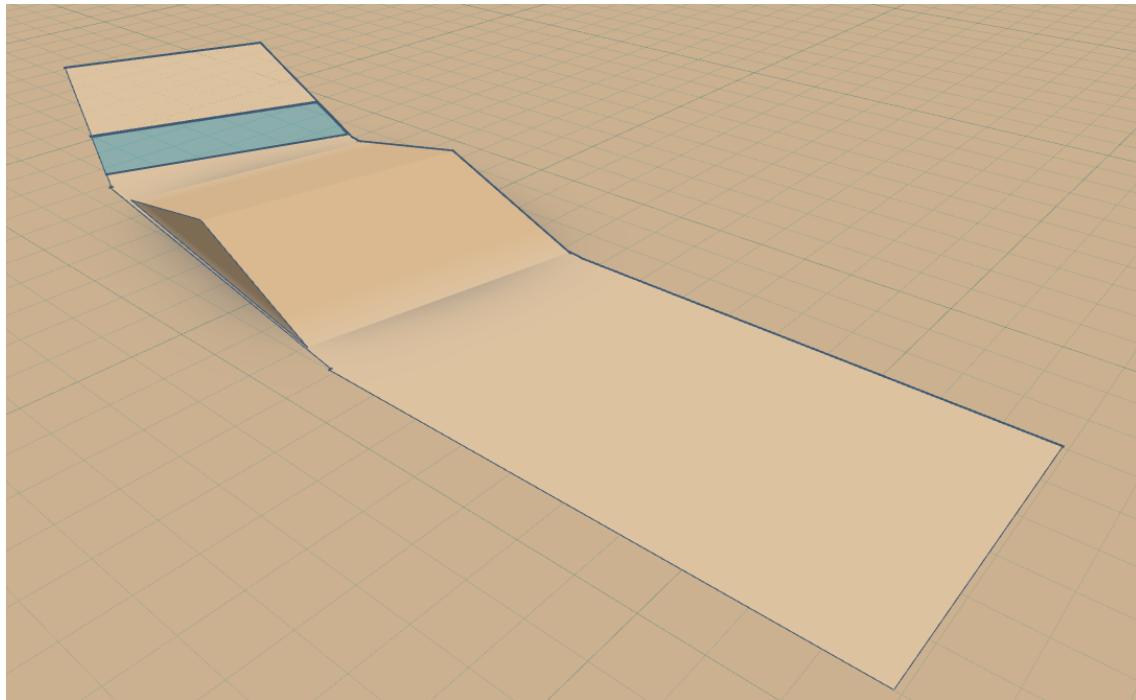


Ilustración 29. Sketch de un bache seguido de un charco en una curva poco cerrada

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.

8 GESTIÓN DE RECURSOS

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 depósito** y los **daños sufridos**.

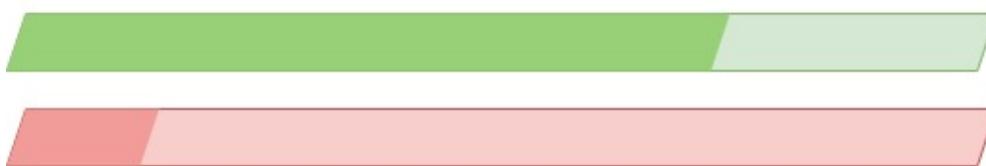


Ilustración 30. Sketch de las barras de combustible y daño

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 (sea tramo o etapa completa) el combustible del vehículo está siempre al 100%, mientras que el daño comienza al 0%.

Se va a realizar seguidamente un análisis del diseño de la consumición de los recursos descritos: combustible y daño.

8.1 CONSUMO DE COMBUSTIBLE

El combustible se gasta de manera constante si se mantiene la velocidad, a un ritmo constante `FUEL_CONSUMPTION_RATE` y que se mide en unidades de combustible por segundo:

$$\text{FUEL_CONSUMPTION_RATE} = 1 \text{ uds/s}$$

Ahora se presentan los diferentes casos de consumo que usan esta constante:

<i>Cuando...</i>	<i>Ritmo de consumo (uds/s)</i>
<i>Se mantiene la velocidad</i>	<code>FUEL_CONSUMPTION_RATE</code>
<i>Se está frenando</i>	<code>1.5*FUEL_CONSUMPTION_RATE</code>
<i>Se está acelerando poco</i>	<code>1.5*FUEL_CONSUMPTION_RATE</code>
<i>Se está acelerando mucho</i>	<code>2*FUEL_CONSUMPTION_RATE</code>

Tabla 2. Consumo de combustible en función del comportamiento del coche

8.2 SISTEMA DE DAÑOS

8.2.1 DAÑO POR COLISIÓN CON OBSTÁCULOS

Este tipo de daño se produce siempre que el coche colisione con obstáculos, como pequeños muros de piedra o rocas.

Para especificar el daño que sufre el cuando el vehículo se estrella, se define de una constante denominada `BASE_COLLISION_DAMAGE` en unidades de daño:

$$\text{BASE_COLLISION_DAMAGE} = 50 \text{ uds}$$

Se va a explicar paso a paso cómo se calcula el daño para una colisión. Cuando el coche impacta con un obstáculo, se recogen dos parámetros, siendo estos `collisionAngle` y `horizontalSpeed`. La velocidad horizontal se obtiene con facilidad, ya que es el componente x del vector de velocidad, pero el proceso de cálculo del ángulo de colisión es algo más enrevesado:

1. Se extrae el vector que va desde el centro de gravedad del coche hasta el centro del obstáculo.
2. Se computa **el ángulo absoluto entre los componentes x de este vector y el vector de dirección del vehículo**. Si el ángulo resultante fuese de 0 grados, el coche estaría impactando frontalmente, mientras que si supera los 90 grados, se trata de un roce. Eso sí, el ángulo nunca podrá ser mayor a 180 grados.

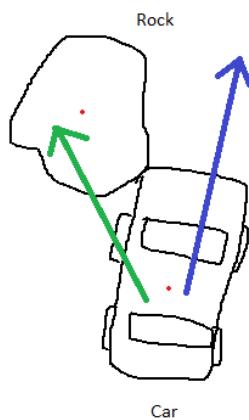


Ilustración 31. Vectores de impacto y dirección

Una vez se cuente con ambas variables, se puede proceder al cómputo del daño recibido. La función debe reflejar que, a mayor sea el ángulo de entrada, el daño debe ser menor. Por ello, debe ser una función inversamente proporcional a este parámetro:

$$\text{damageReceived} = \frac{\text{BASE_COLLISION_DAMAGE}}{\text{collisionAngle}}$$

No obstante, se da ahora el problema de que, si el ángulo de entrada en 0, el daño no se puede calcular (tendería a ∞). La solución escogida consiste en sumar 1 al ángulo de entrada:

$$damageReceived = \frac{BASE_COLLISION_DAMAGE}{collisionAngle + 1}$$

Otro problema que encontrariamos sería que el daño base debería ser muy alto para que el daño recibido fuese considerable. Como se prefiere trabajar con números pequeños, una solución posible es dividir el ángulo de entrada por 22.5:

$$damageReceived = \frac{BASE_COLLISION_DAMAGE}{\frac{collisionAngle}{22.5} + 1}$$

Con esto se consigue que, cuando el coche impacte de frente con un obstáculo, reciba la totalidad del daño base, mientras que cuando lo haga lateralmente, sufra una quinta parte de este.

Por último, es crucial considerar la velocidad en la ecuación. La velocidad debe aumentar o reducir proporcionalmente el daño recibido:

$$damageReceived = \frac{horizontalSpeed}{60} \cdot \frac{BASE_COLLISION_DAMAGE}{\frac{collisionAngle}{22.5} + 1}$$

8.2.2 DAÑO POR CAÍDA

Este tipo de daño se produce cuando el vehículo impacta con el terreno después de un salto al entrar o salir de una variación del relieve de la pista.

Para especificar el daño que sufre el cuando el vehículo choca con el suelo, se define de una constante denominada `BASE_FALL_DAMAGE` en unidades de daño:

$$BASE_FALL_DAMAGE = 30 \text{ uds}$$

Se tiene en cuenta en este caso el ángulo de caída, `fallAngle`, ángulo absoluto entre el vector de dirección del coche y el vector global hacia abajo, $(0, -1, 0)$. La velocidad que se necesita es también distinta, pues se trata del componente y del vector de velocidad del coche, que llamaremos `verticalSpeed`.

La fórmula de cálculo del daño muy diferente. Si cayésemos con un ángulo de 180 grados el coche estaría impactando de manera perpendicular con el terreno, lo cual debería causar significativamente más daño. Debería pasar lo mismo si caemos con la parte frontal, es decir, con un ángulo de 0 grados. La función debe alcanzar su máximo en estos dos casos. Por ello, se utiliza un exponente y se centra la función a 90 grados:

$$damageReceived = BASE_FALL_DAMAGE \cdot \left(\frac{fallAngle - 90}{90} \right)^2$$

Sin embargo, también se debe considerar que el vehículo sufrirá daño incluso cuando caiga de manera paralela al terreno, pues la amortiguación no absorberá todo el impacto:

damageReceived

$$= \text{BASE_FALL_DAMAGE} \cdot \left(\frac{\text{fallAngle} - 90}{90} \right)^2 + \frac{\text{BASE_FALL_DAMAGE}}{6}$$

Finalmente, se debe tener en cuenta la velocidad vertical para escalar el daño:

damageReceived

$$= \frac{\text{verticalSpeed}}{5} \cdot (\text{BASE_FALL_DAMAGE} \cdot \left(\frac{\text{fallAngle} - 90}{90} \right)^2 + \frac{\text{BASE_FALL_DAMAGE}}{6})$$

8.2.3 UMBRAL MÍNIMO PARA SUFRIR DAÑOS

Hasta cierto punto el coche debe ser capaz de absorber el daño. Para evitar que sufra daños en situaciones donde apenas se levanta o simplemente roza un obstáculo, se ha decidido establecer un daño mínimo que *damageReceived* debe superar o igualar para poder sumarse al daño total sufrido. Este daño umbral lo denominaremos *MIN_DAMAGE*:

$$\text{MIN_DAMAGE} = 5$$

IF (damageReceived ≥ 5) THEN (damage += damageReceived)

8.3 AGOTAMIENTO DE RECURSOS

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

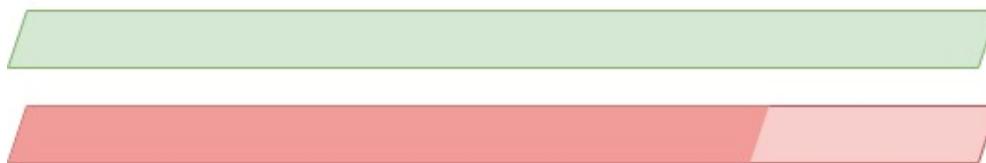


Ilustración 32. Sketch de las barras recursos: sin combustible

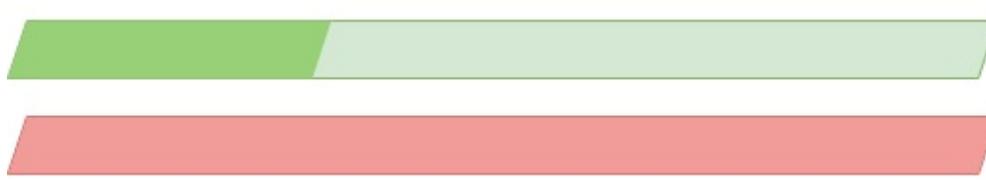


Ilustración 33. Sketch de las barras recursos: coche destrozado

8.4 RECURSOS ENTRE PRUEBAS

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

Esto permite al jugador considerar, por ejemplo, si su desempeño en un tramo es apropiado para enfrentarse a la etapa completa. Recordemos que la etapa completa incluye todos los tramos de los días/pruebas 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 prueba final.

9 PROGRESIÓN

9.1 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.

9.2 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 distintos niveles 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**, cada uno albergando **una prueba**:

- 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 prueba de la etapa completa el último día, los puntos de control estarán ya por defecto elegidos tal y como quedaron en los tramos por separado (cuando se intentaron por última vez).

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

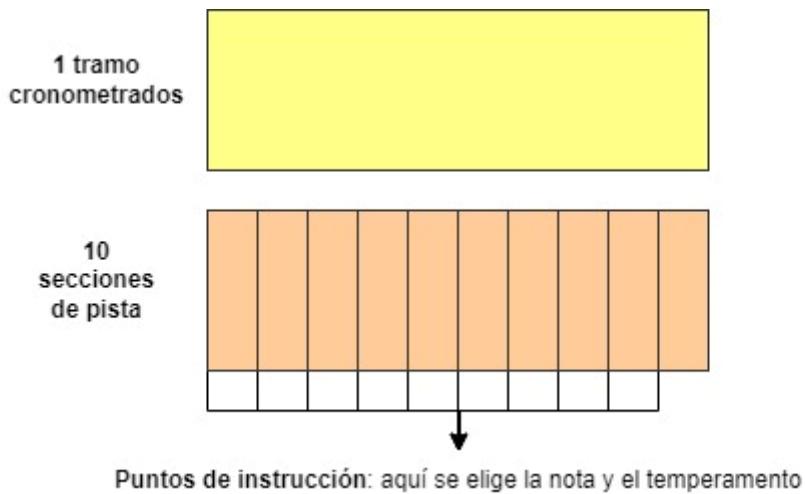


Ilustración 34. Estructura de un tramo cronometrado (días/pruebas 1 a 3)

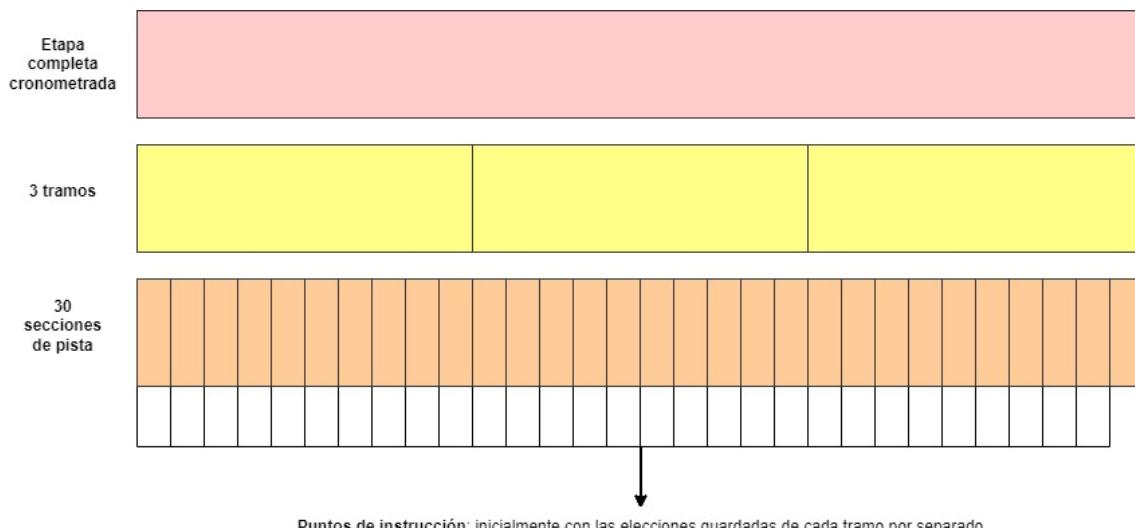


Ilustración 35. Estructura de una etapa completa cronometrada (último día/prueba)

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.

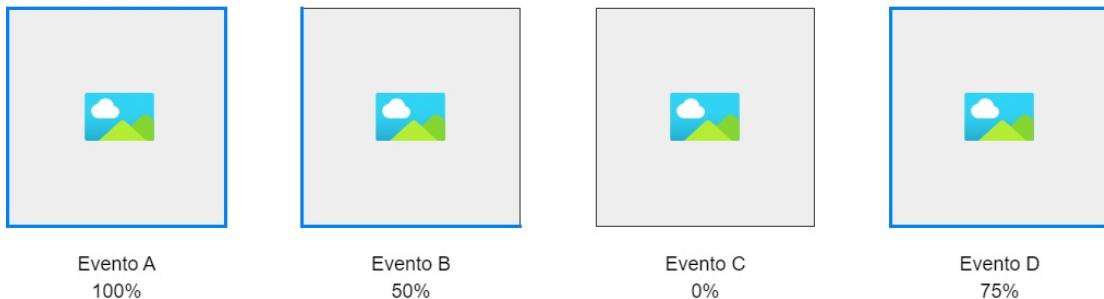


Ilustración 36. Estructura del juego

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

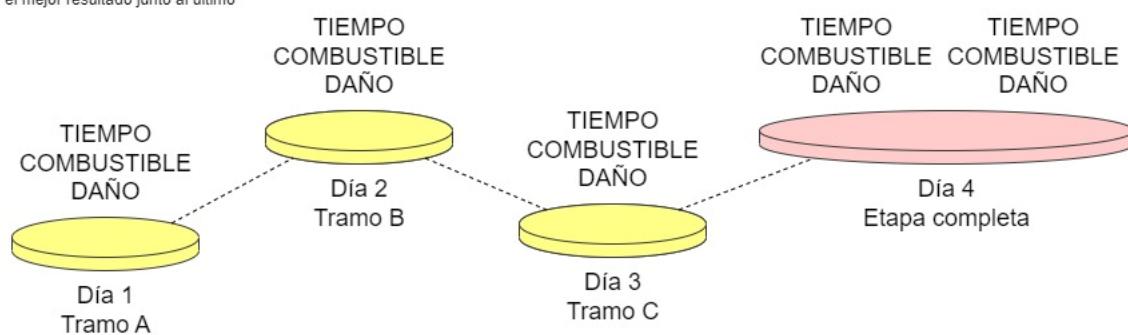


Ilustración 37. Estructura de un evento

9.3 CLASIFICACIÓN EN LÍNEA

10 CONTROLES Y PERIFÉRICOS

11 INTERFAZ DE USUARIO

11.1 REQUISITOS DE LA INTERFAZ

El siguiente esquema muestra los requisitos que se han considerado para el diseño y flujo de la interfaz de usuario:

Interfaz Gráfica Amigable

- La interfaz debe ser colorida, amigable y atractiva para niños, con personajes y elementos visuales que les resulten llamativos

Tutorial Interactivo

- Incluir un tutorial intuitivo que enseñe los conceptos básicos

Feedback Inmediato

- Proporcionar retroalimentación instantánea cuando los niños tomen decisiones o completen tareas, reforzando positivamente el aprendizaje

Indicadores de Progreso

- Mostrar un medidor de progreso o puntuación para que los niños puedan ver su avance en el juego

Incentivos y Recompensas:

- Ofrecer recompensas, medallas o logros por completar tareas o niveles, fomentando la motivación y el compromiso

Sistema de Pistas

- Proporcionar pistas o sugerencias cuando los niños se enfrenten a desafíos difíciles, sin revelar la solución completa

Interfaz Táctil o Controlador Simple

- El juego debe ser fácil de controlar con un toque en pantalla o con controles simples que los niños puedan entender fácilmente

Instrucciones Visuales y de Voz

- Si es posible, incluir instrucciones visuales y de voz para ayudar a los niños en la comprensión de las tareas y conceptos

Registro de Actividad

- Mantener un registro de las acciones y decisiones de los niños para que los padres o maestros puedan revisar el progreso y el aprendizaje

Ajustes de Dificultad

- Permitir a los padres o tutores ajustar la dificultad del juego según la edad y habilidades del niño

Compatibilidad Multiplataforma

- Hacer que el juego esté disponible en varias plataformas (Web, PC, tabletas) para una mayor accesibilidad. Hay que tener en cuenta el aspect ratio y las características de cada una

Música y Sonidos Atractivos

- Incluir música y efectos de sonido que sean agradables y reconocibles, que permitan reconocer patrones

11.2 DIAGRAMAS DE FLUJO DE NAVEGACIÓN

Seguidamente se presentan diagramas de flujo referentes a la navegación por los menús y pantallas:

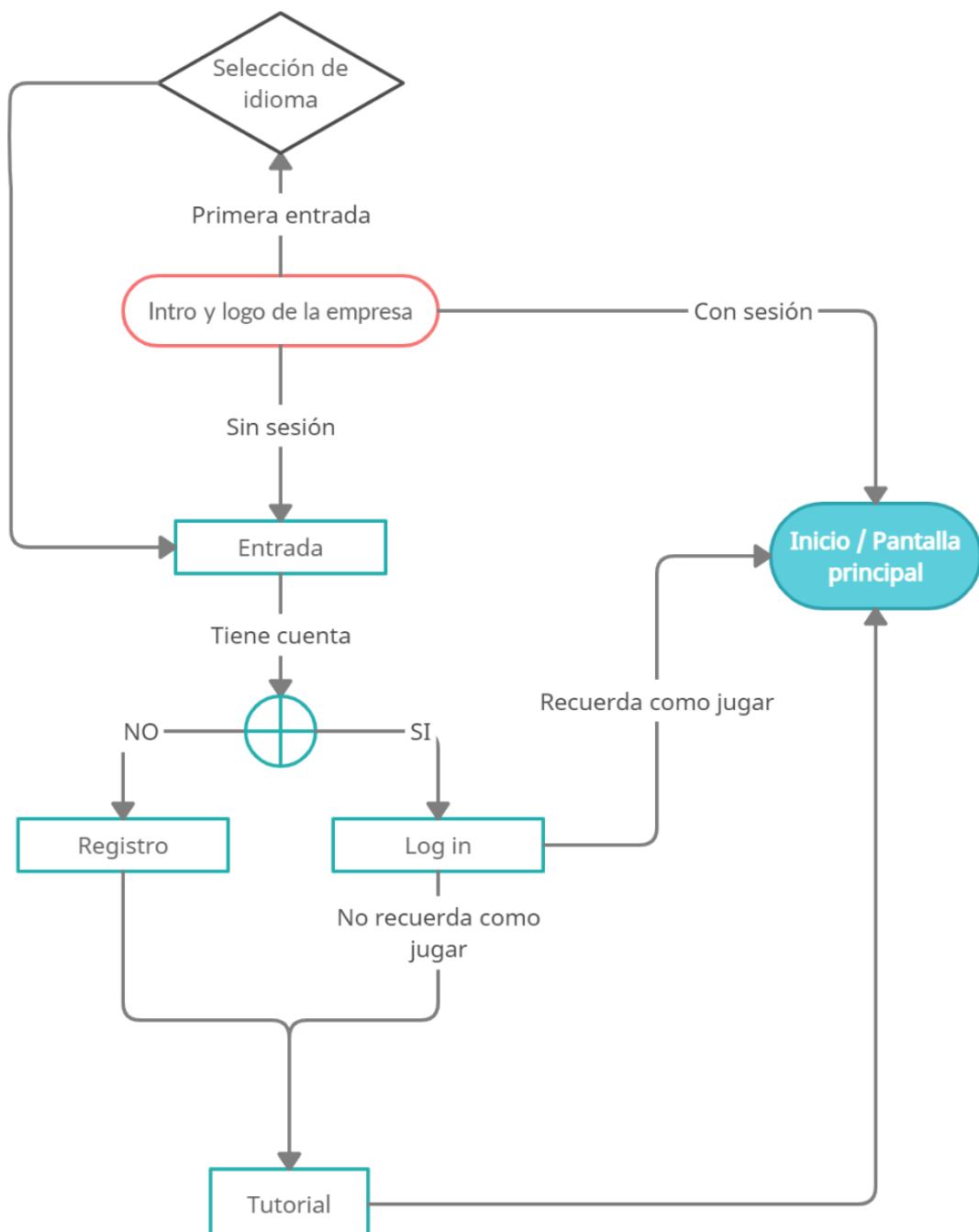


Ilustración 38. Diagrama de navegación de inicio del juego

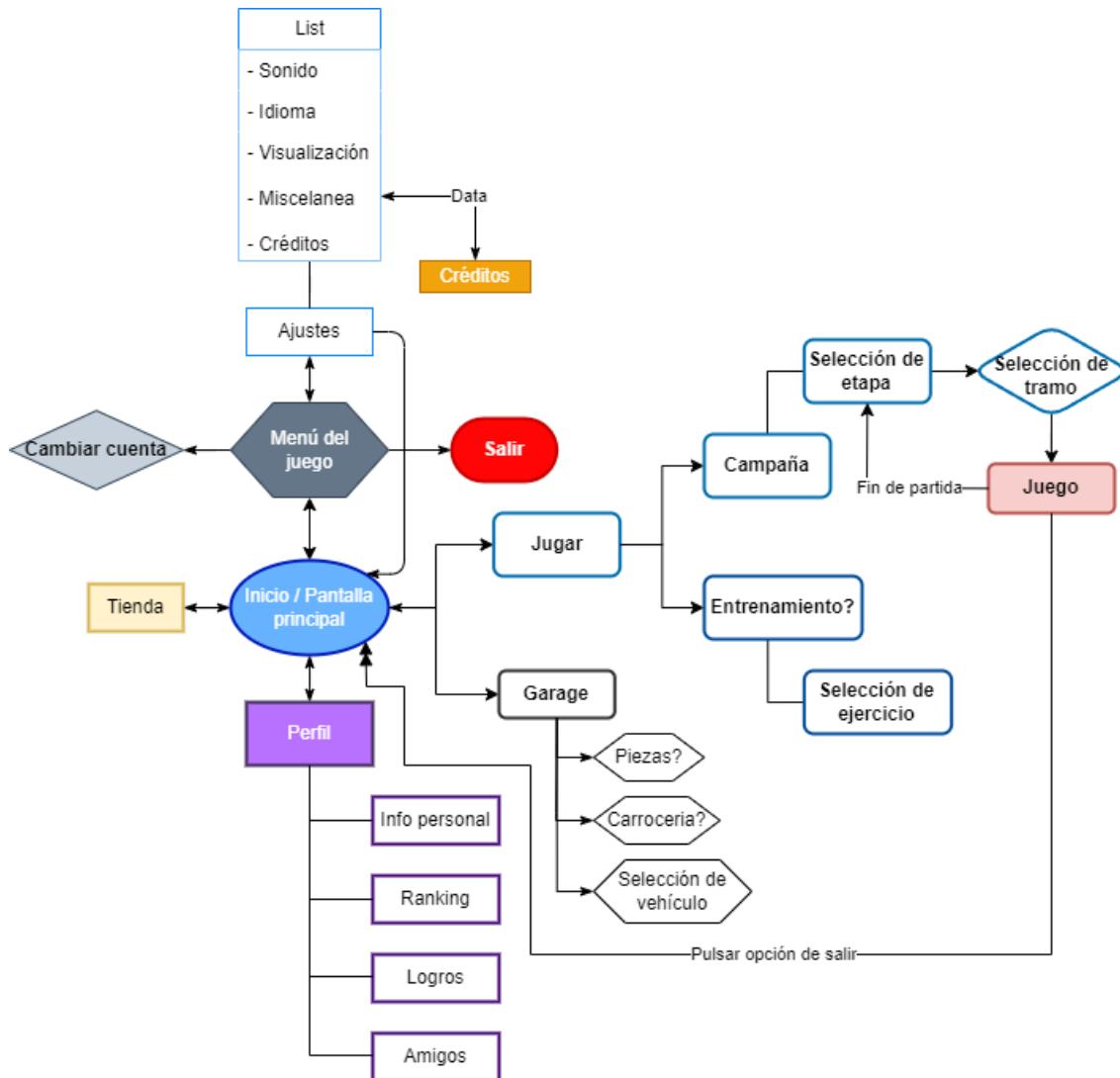


Ilustración 39. Diagrama de navegación: menús y juego

En el diagrama superior se presentan algunas ramas marcadas con interrogación, estas son aquellas que se han planteado como futuras adiciones al juego que en principio no estarán en las versiones preliminares.

11.3 DISEÑO ESQUEMÁTICO DE LA UI

En este apartado se muestran *mockups* de las pantallas, menús y pop-ups que se han diseñado para el prototipo (no incluyen la parte in-game). El formato de estas es para juego en navegador:



Ilustración 40. Diseño de la Splash Screen con el logo de la empresa



Ilustración 41. Diseño de la selección de idioma (primera vez que se inicia el juego)



Ilustración 42. Diseño de la pantalla de inicio

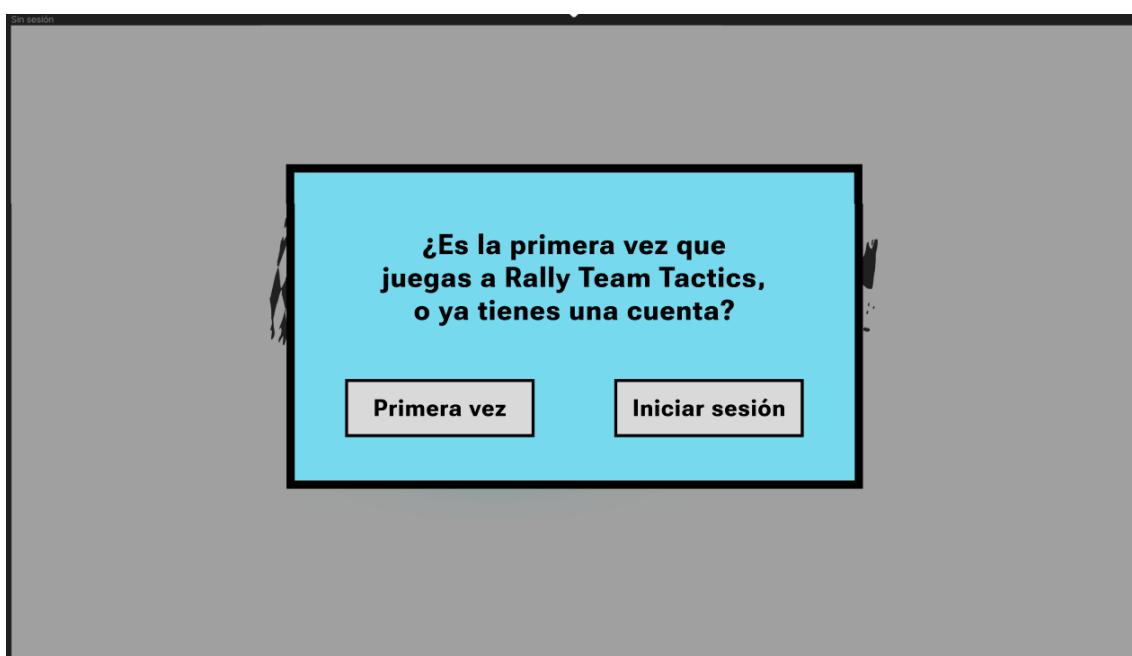


Ilustración 43. Diseño del pop-up de inicio de sesión, si no se ha iniciado previamente en el dispositivo

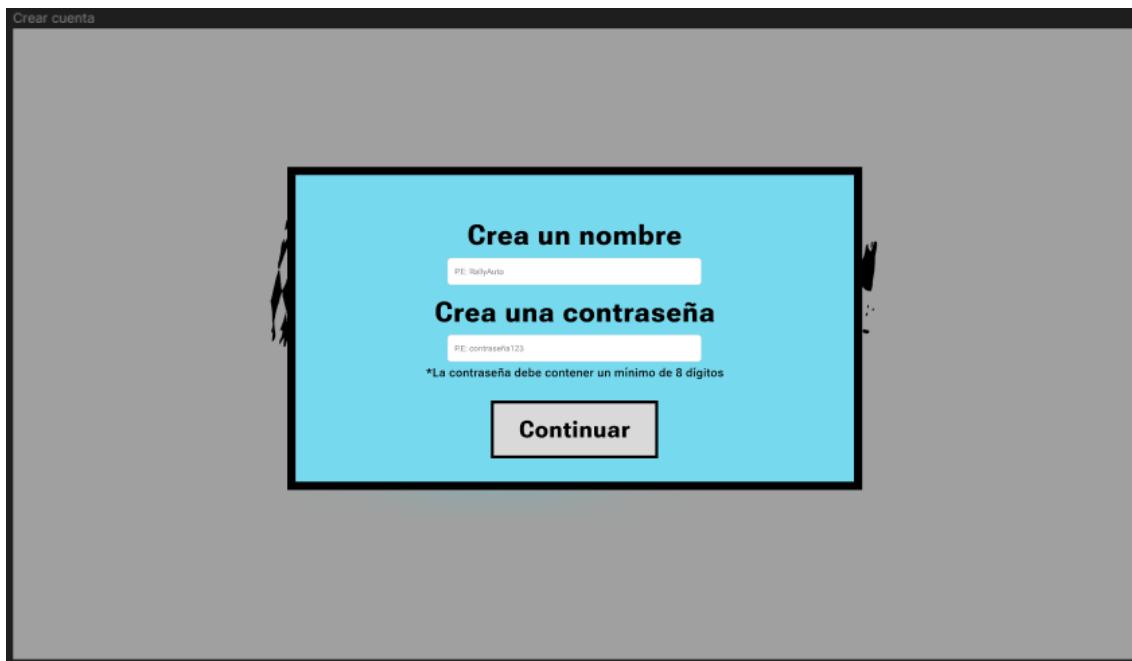


Ilustración 44. Diseño del pop-up de registro



Ilustración 45. Diseño del pop-up de inicio de sesión

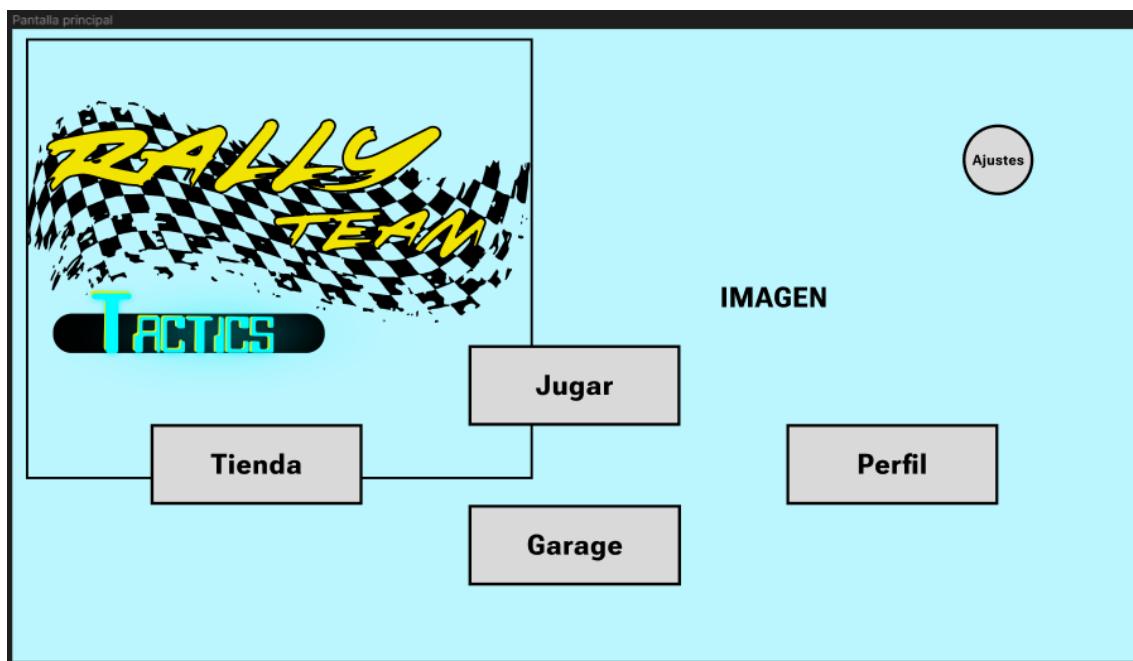


Ilustración 46. Diseño del menú principal

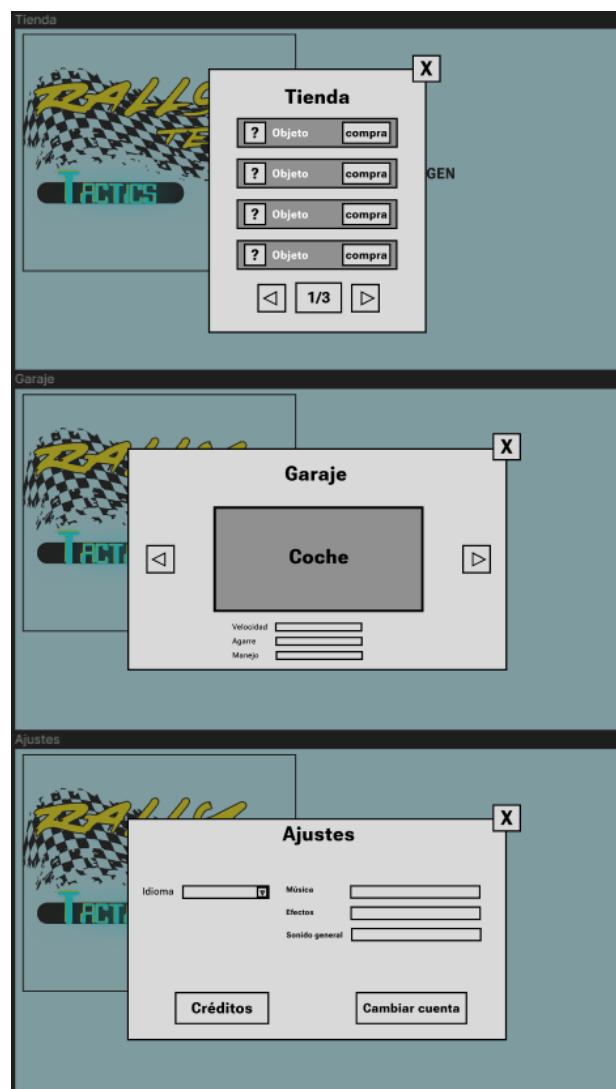


Ilustración 47. Diseños de los pop-ups de tienda, garaje y ajustes, respectivamente

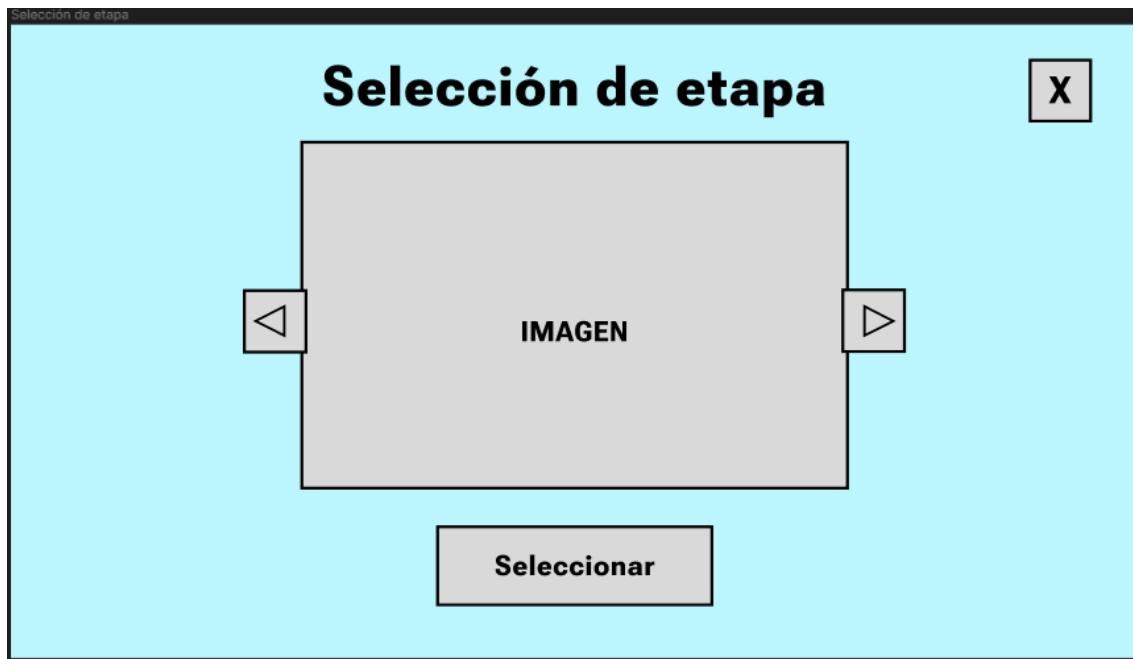


Ilustración 48. Diseño de la pantalla de selección de etapa

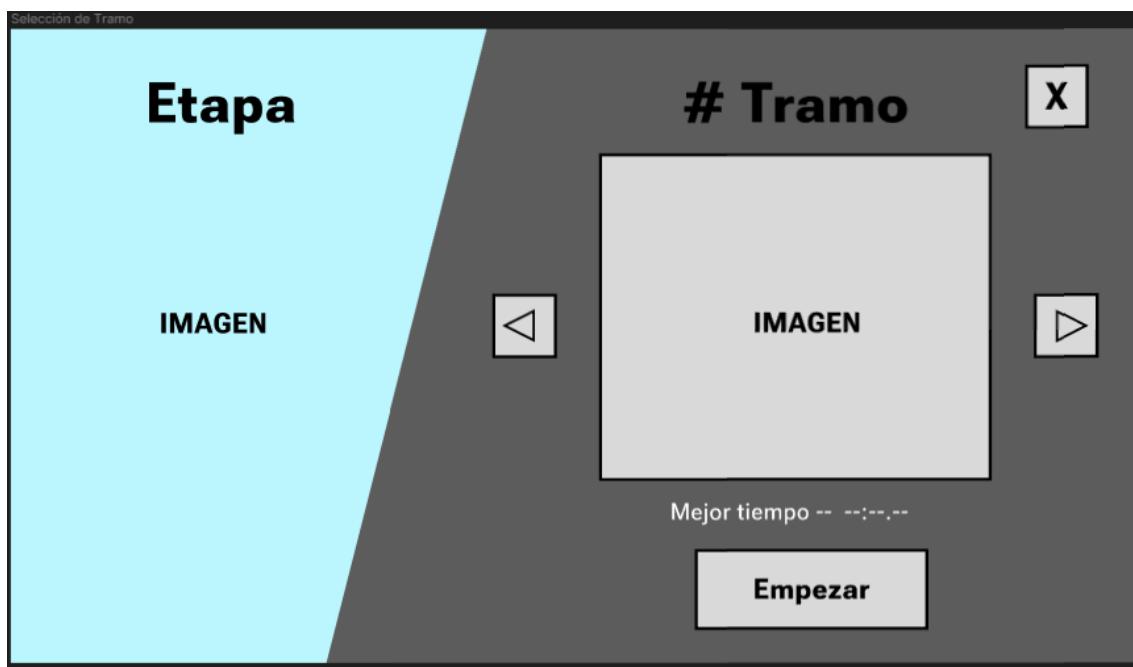


Ilustración 49. Diseño de la pantalla de selección de tramo en la etapa

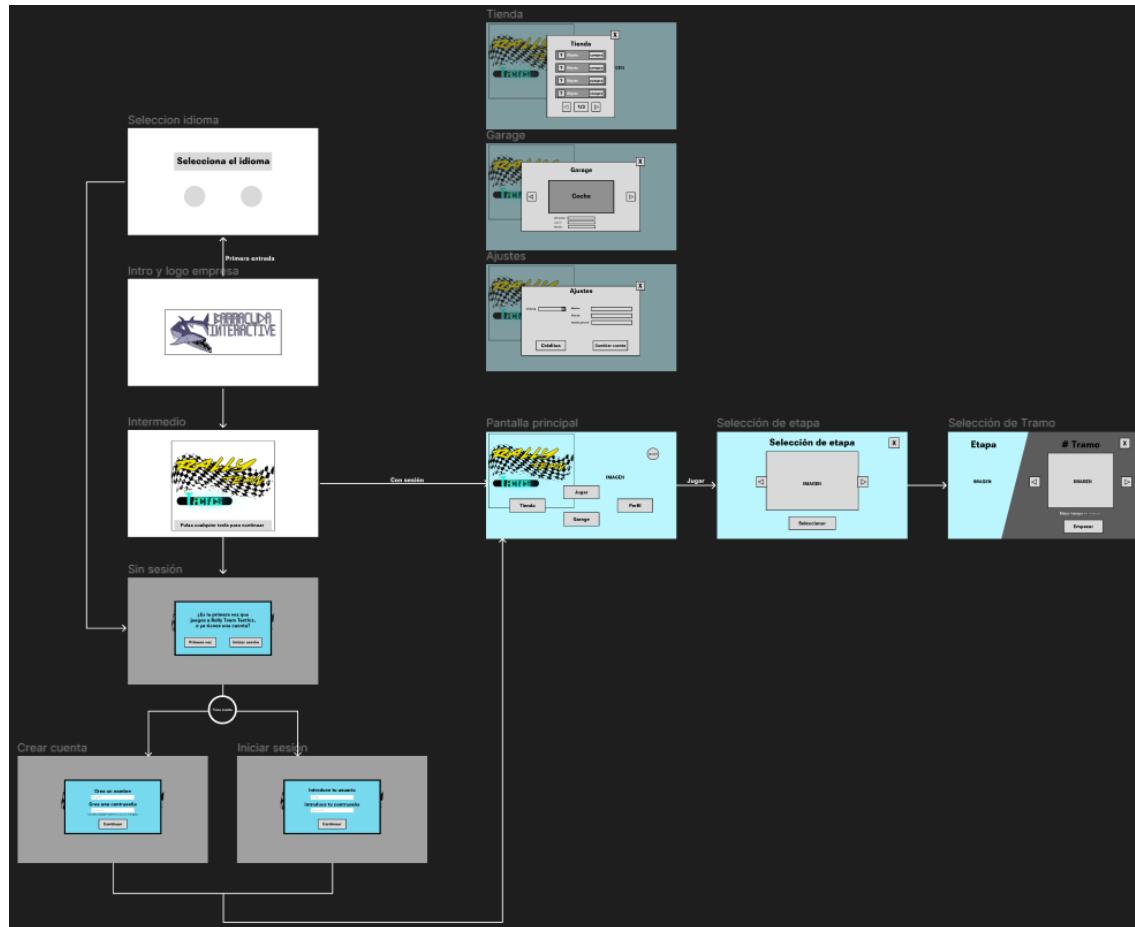


Ilustración 50. Representación del flujo de navegación en Figma

En la imagen superior, se proporciona una representación gráfica del diagrama de flujo de las pantallas con el fin de facilitar la comprensión de la navegación por los menús. Más abajo se incluye un enlace al proyecto en Figma, donde se ha recreado el flujo de navegación para previsualizar su funcionamiento.

<https://www.figma.com/file/LILMOV1jsncNQxOc0rWFNt/RTT?type=design&node-id=0%3A1&mode=design&t=zzw8euf4DUNr2HJW-1>

12 ARTE

12.1 ESTÉTICA DEL JUEGO

12.1.1 REFERENCIAS ARTÍSTICAS

Como se explicó en el apartado [2.1 Principales referencias](#), el juego presentará una estética *low-poly* colorida, inspirada en especial por el *arcade racer Ridge Racer* (Namco, 1993) y el *indie* de *rallies* contemporáneo *Art of Rally* (Funselektor Labs, 2020).



Ilustración 51. Ridge Racer (Namco, 1993)



Ilustración 52. Art of Rally (Funselektor Labs, 2020)

12.1.2 DISEÑO VISUAL DEL JUEGO

Para que todo el contenido relacionado con *RTT*, tanto jugable como promocional (y su documentación), siguiesen el mismo estilo visual, se elaboró un documento de estilo de juego.

Este apartado del GDD recogen los aspectos más importantes del documento.

PALETAS DE COLORES

RTT tiene una paleta base que incluye el color principal, secundario, sus respectivos complementarios, y colores para texto y otros elementos sobre oscuro o claro.

El color principal es un amarillo que se usa principalmente en todos los elementos de la UI:



Ilustración 53. Color primario escogido

Para añadir efectos de relieve a aquellos elementos de la interfaz que lo requieran se ha definido una paleta de sombras. Además, de esta también se extrae el color del fondo del menú:



Ilustración 54. Paleta de sombras del color primario

El color complementario a este amarillo es un violeta intenso que se utiliza para contrastar ciertos detalles en elementos que usen el color primario:

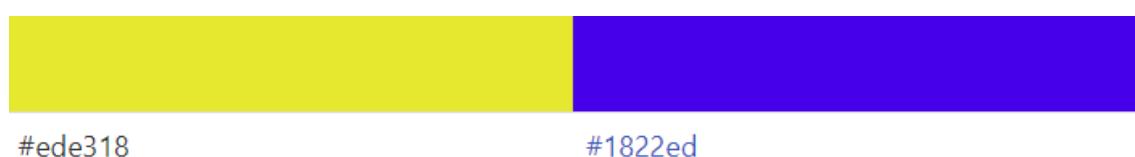


Ilustración 55. Color primario y su complementario

Por otro lado, el color secundario es un cian neón cuyo propósito es destacar elementos al interactuar con la interfaz:



Ilustración 56. Color secundario escogido

Igualmente para este se define una paleta de sombras para regular efectos de iluminación en la UI:



Ilustración 57. Paleta de sombras del color secundario

El color complementario sirve para añadir detalles sobre elementos de la UI del color secundario cian:



Ilustración 58. Color secundario y su complementario

El negro no puro definido para *RTT* se usa para texto y elementos situados sobre fondos claros o para añadir detalles sobre el fondo base:



Ilustración 59. Negro no puro escogido

Por el contrario, el blanco no puro elegido se usa para textos y elementos sobre oscuro:



Ilustración 60. Blanco no puro escogido

La paleta base resultante sería la siguiente:

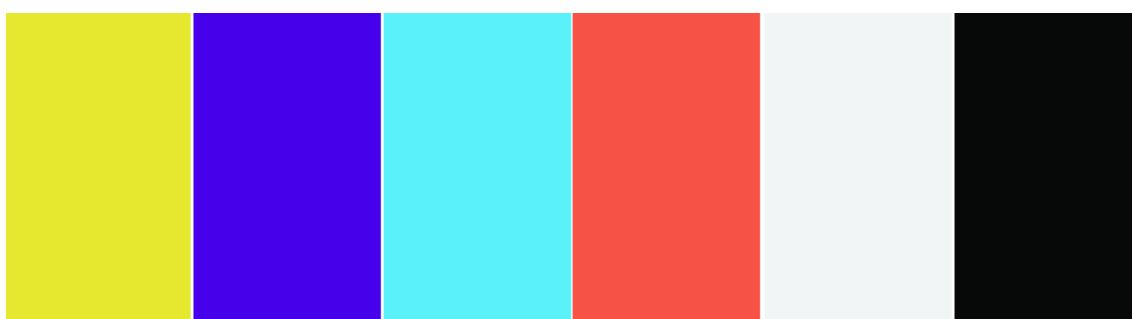


Ilustración 61. Paleta base de RTT

TIPOGRAFÍAS

Los tamaños se adecuarán según necesidades. Se incluyen los enlaces de descarga de las fuentes, que además incluyen detalles sobre estas.

MUBENK REGULAR

Esta tipografía se utiliza exclusivamente para el título del juego y en los créditos, si se incluyen encabezados en ellos:

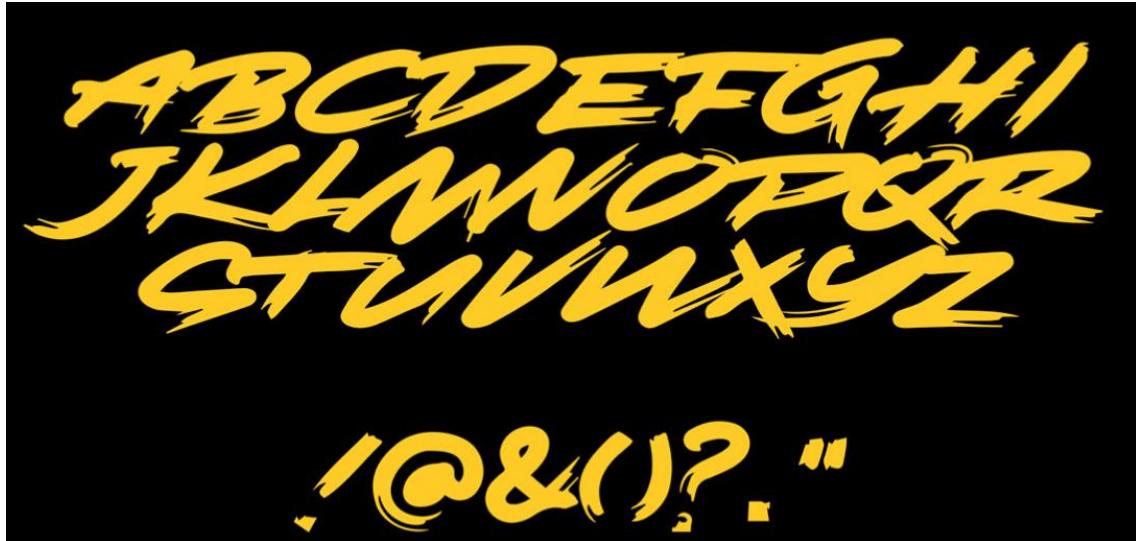


Ilustración 62. Muestra de mubenk regular

<https://www.behance.net/gallery/100102693/Mubenk-Brush-Font-Free-Download>

THECNIC

Esta tipografía con mayor legibilidad a diferentes tamaños sirve como fuente para todo el texto del juego que no sean el título ni los encabezados de los créditos. Se usan las tres versiones de la fuente, Regular, Bold y Lite, incluidas en Windows por defecto.

- Thecnic para texto de corta y media longitud
- ThecnicBold para remarcar texto importante, como el de botones
- ThecnicLite para textos largos, como instrucciones o créditos

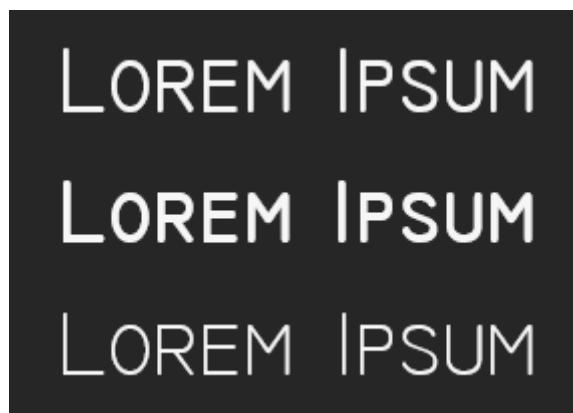


Ilustración 63. Muestra de Thecnic, ThecnicBold y ThecnicLite

ESTILO DE LA UI

Se procede a explicar el estilo de todos los elementos gráficos que componen todas las pantallas menú y la interfaz *in-game*.

Todas las pantallas mostrarán un fondo cuyo color será un degradado entre #80880e y #d5cc15. No obstante, la pantalla de selección de evento tendrá una particularidad, pues su fondo estará dividido entre este color y la imagen del evento.

TÍTULO

La pantalla de título usa la tipografía *mubenk*, pues presenta el logo del juego, caracterizado por los colores primario y el negro escogidos.



Ilustración 64. Logo de RTT

MARCOS DE LOS MENÚS EMERGENTES

Aquellas pantallas que requieren de un marco, como el pop-up de login, utilizan el color negro para este.

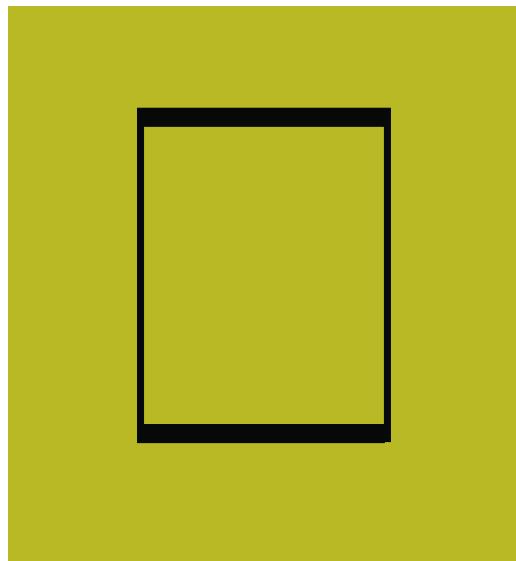


Ilustración 65. Ejemplo de estilo del marco de un menú emergente

BOTONES EN LOS MENÚS

La mayor parte de los botones de la interfaz se presentan en el color primario con detalles de sombreado #d5cc15. El interior puede contener iconos o texto en el negro escogido, #070808. El texto, como en el resto de la UI (a excepción de algunos elementos como el título), está en fuente Thecnic y ThecnicBold. Si el jugador se sitúa sobre un botón, el texto o ícono cambia al color secundario (cian).

Algunos botones especiales que contienen iconos, como los que se superponen al marco de un pop-up, siguen un código de colores distinto. Los botones para “salir” usan el naranja complementario al cian definido como color secundario, mientras que los botones de “avance” o “retroceso” utilizan el violeta complementario al amarillo definido como color primario.

Si el jugador se sitúa sobre un botón, el texto o ícono cambia al color secundario (cian).

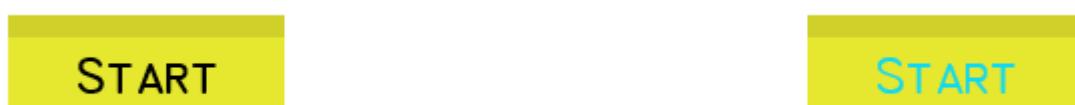


Ilustración 66. Ejemplo de estilo de un botón normal

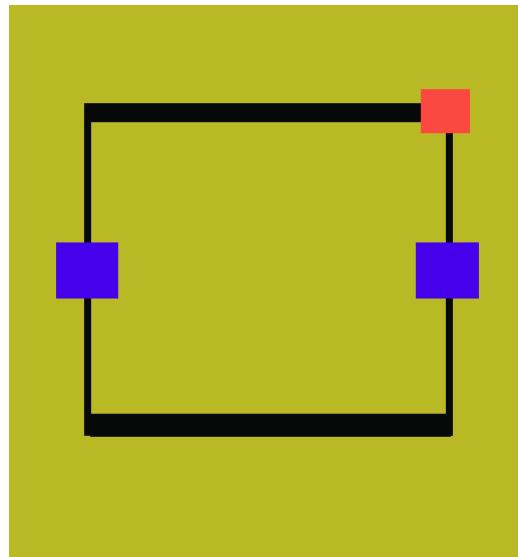


Ilustración 67. Ejemplo de estilo de botones especiales superpuestos a un marco de ventana emergente

PANTALLA DE CARGA

La pantalla de carga sigue la tonalidad del fondo base, pero cuentan con elementos extras en el color negro escogido que están relacionados con el mundo del rally.

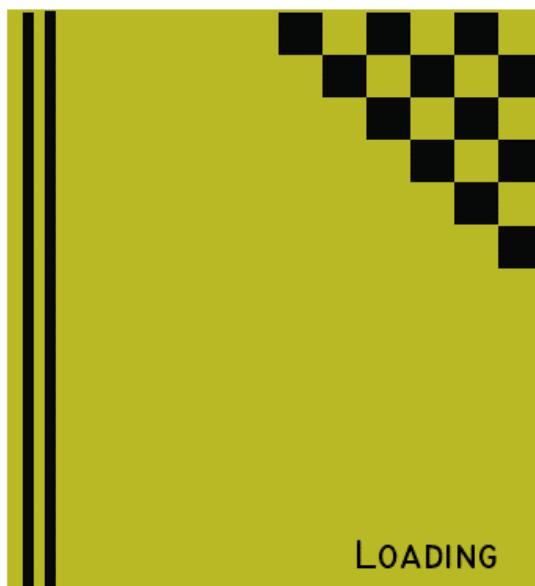


Ilustración 68. Ejemplo de estilo de pantalla de carga

HEADS-UP DISPLAY (INTERFAZ IN-GAME)

Dentro de una partida encontramos seis elementos principales a tratar en el HUD: los medidores de daño y combustible, los botones de acceso a menús o de acción, los selectores de nota, el *slider* de temperamento y los indicadores del estado de cada punto de control.

- Los botones de acceso a menús emergentes o de acción siguen el mismo estilo que los botones estándar fuera de la partida.

- Los selectores de nota son similares en estilo, y son redondeados y contienen el ícono correspondiente a la instrucción. Eso sí, no se iluminan ni resaltan al ponerse sobre ellos.
- El *slider* de temperamento usa principalmente el gris #919191.
- La barra de daño usa el rojo #db0f0f para el relleno y una tonalidad más suave del mismo para la parte sin llenar (este no se muestra en la previsualización inferior). La barra de combustible es igual, pero con el verde #21db21. En ambos casos la parte que contiene el ícono o texto referente al recurso está en el amarillo primario.
- El muro luminoso que se muestra sobre los puntos de control adquiere un color verdoso (#21db21) cuando se ha seleccionado instrucción, negro #070808 en caso contrario.

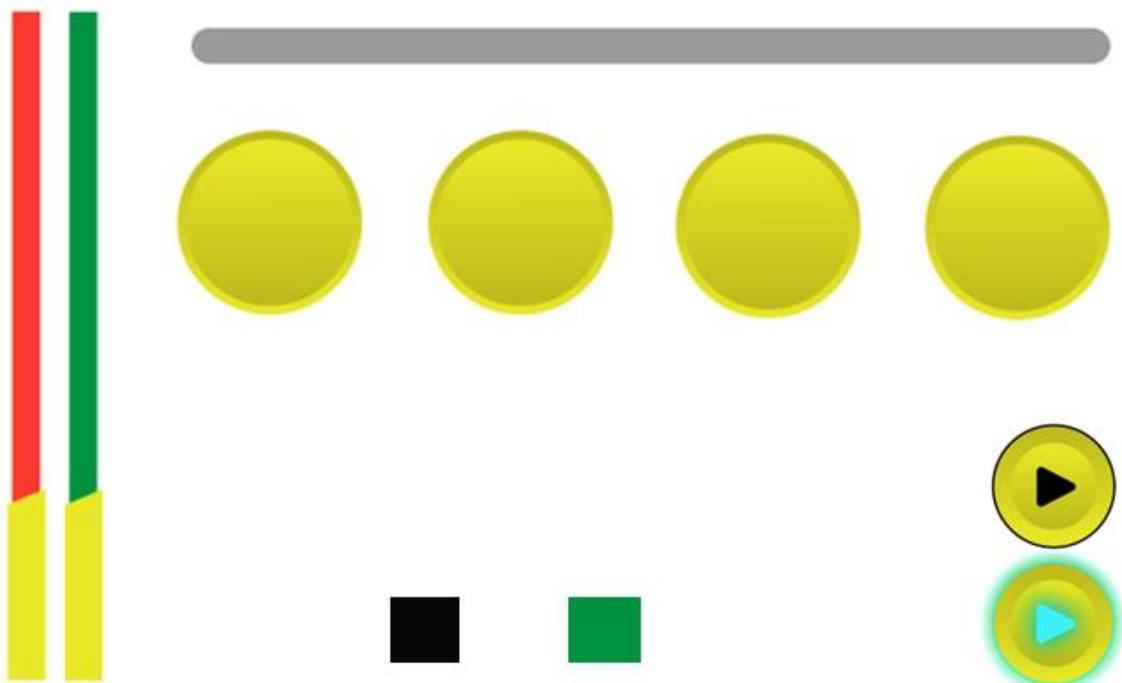


Ilustración 69. Ejemplo de estilo de elementos del HUD

12.2 CONCEPT

Se presentan en este apartado una serie de *concepts* de elementos del juego, como el coche o los *props* de los escenarios.

12.2.1 CONCEPTS DEL COCHE

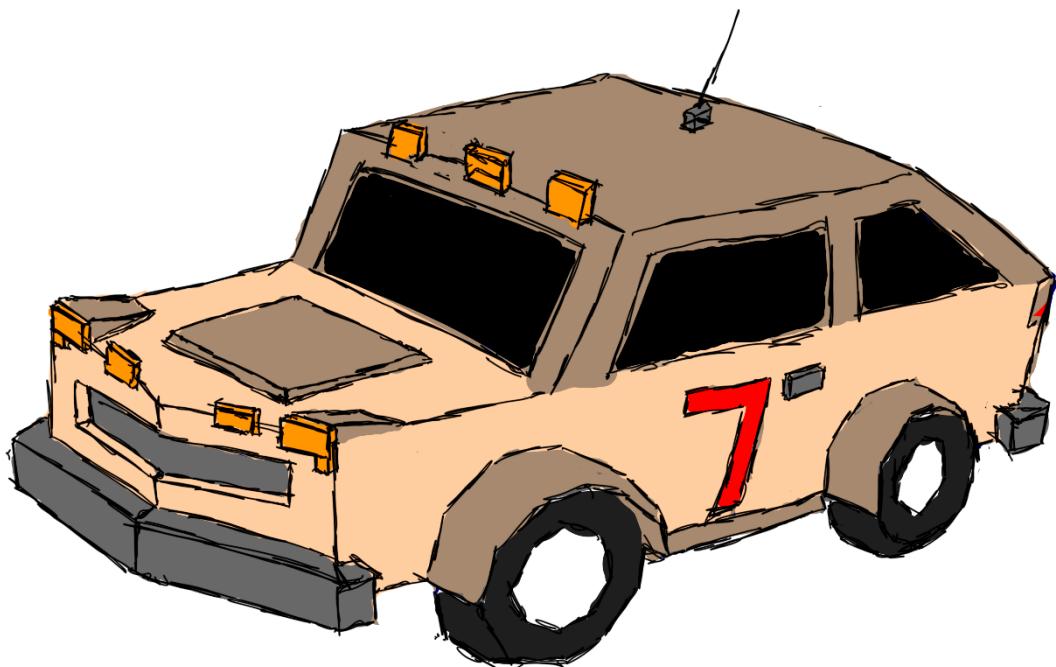


Ilustración 70. Concept de coche en perspectiva



Ilustración 71. Concept de coche de perfil

12.2.2 CONCEPTS DE PROPS



Ilustración 72. Concept de un punto de control

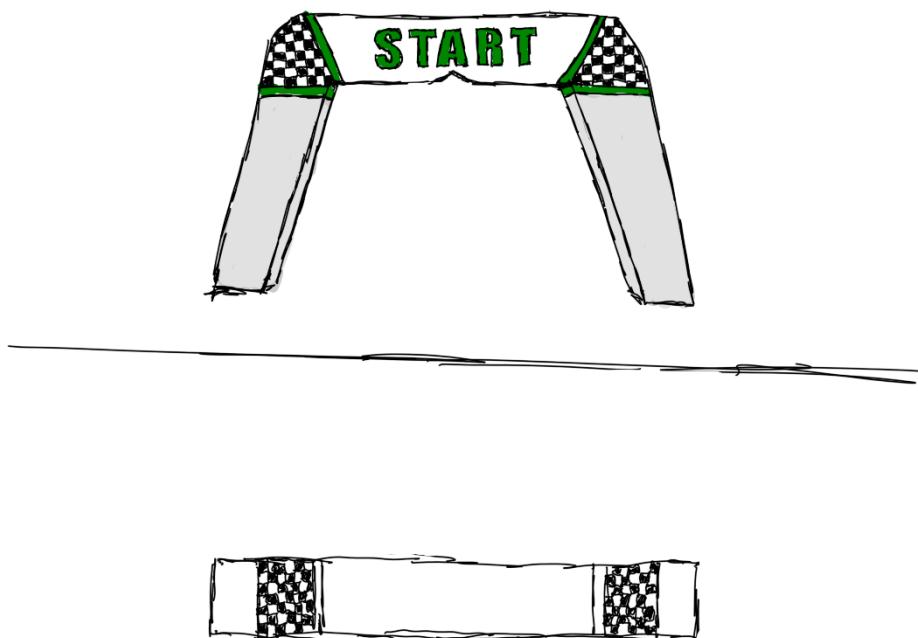


Ilustración 73. Concept de la salida

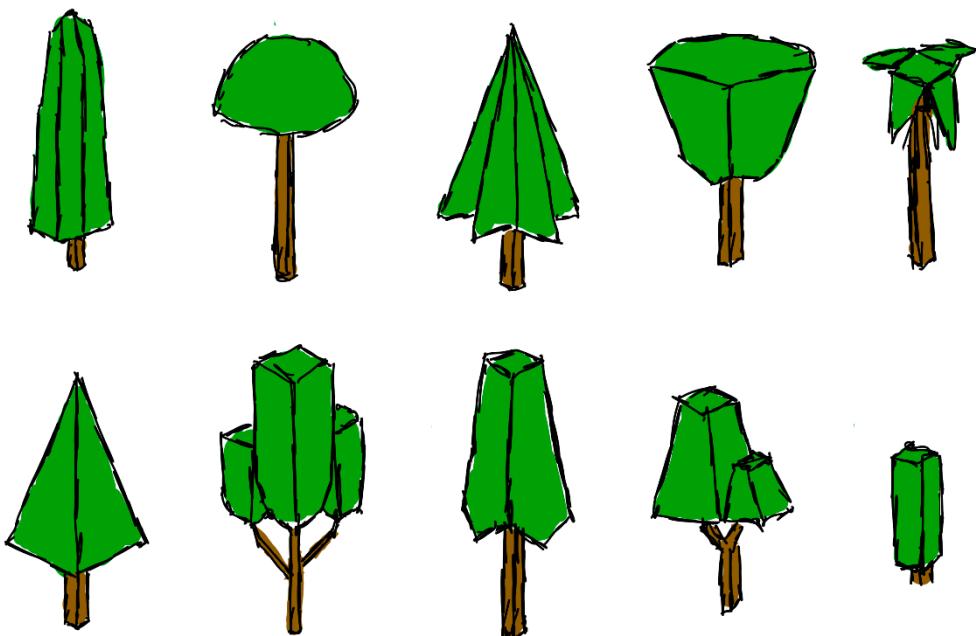


Ilustración 74. Concept de diferentes variantes de árbol

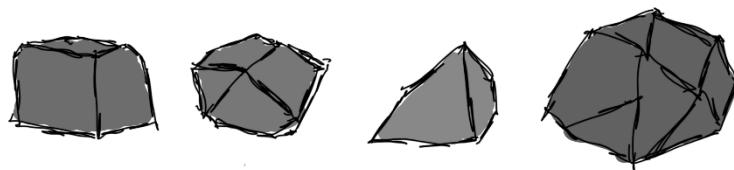
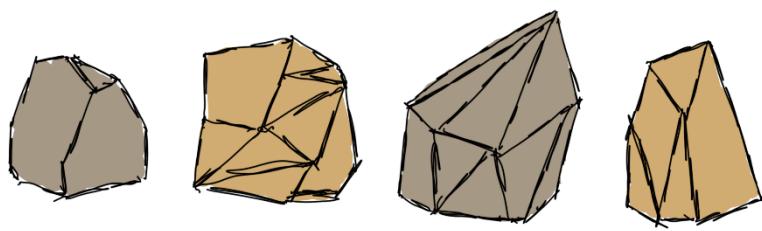


Ilustración 75. Concept de diferentes rocas

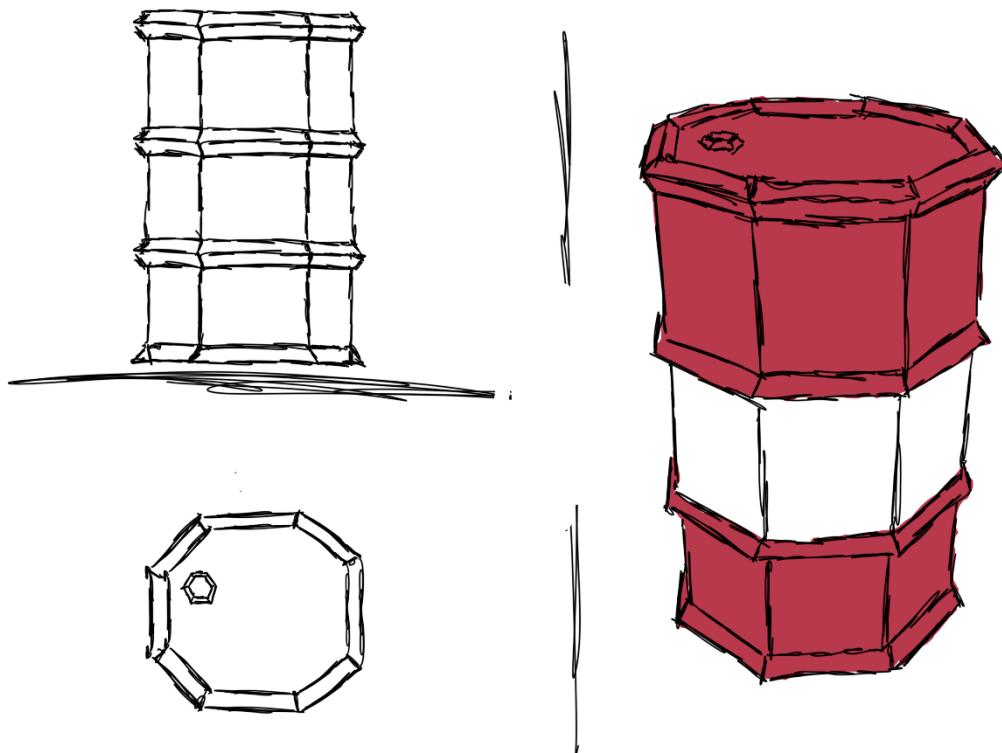


Ilustración 76. Concept de un barril

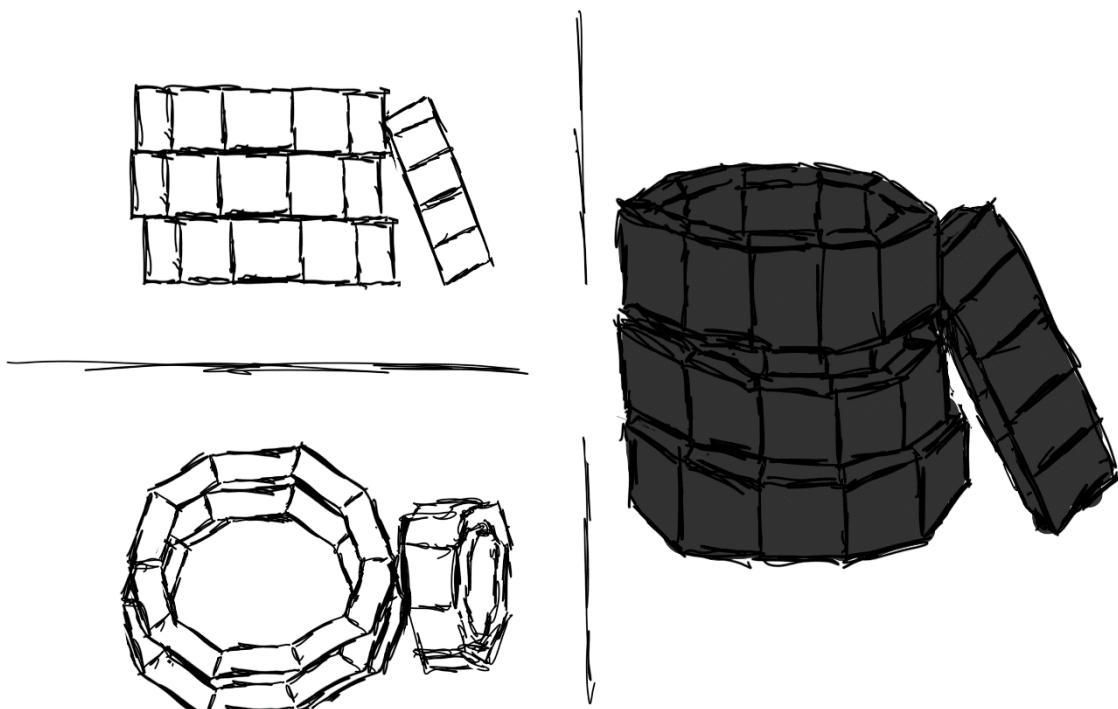


Ilustración 77. Concept de una pila de ruedas

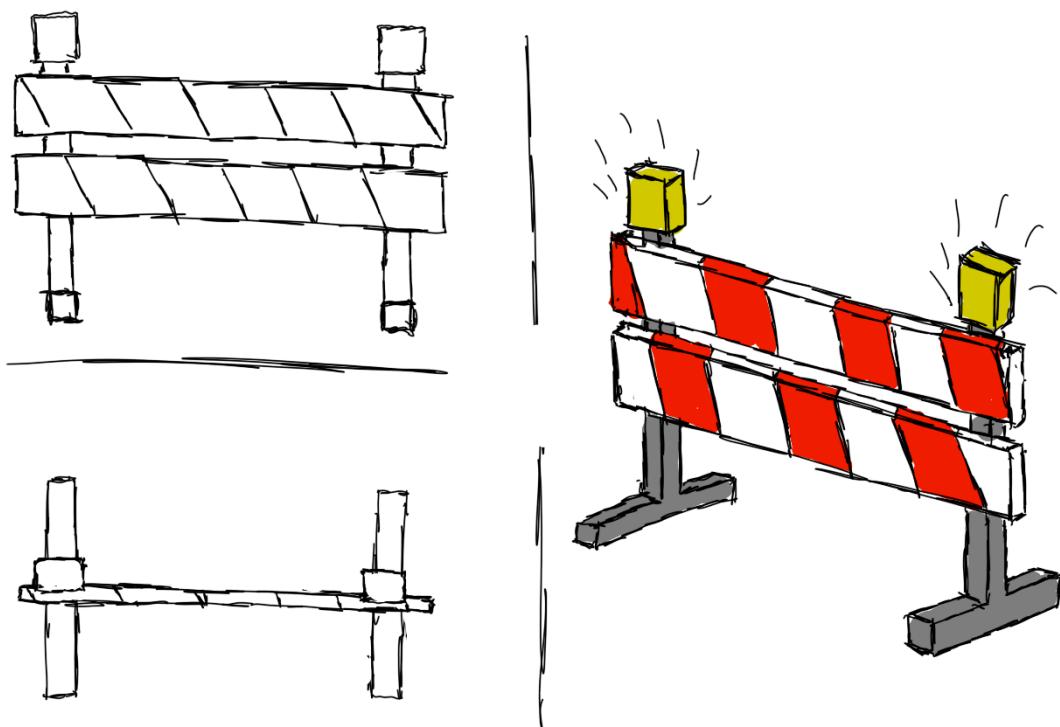


Ilustración 78. Concept de una valla de obra

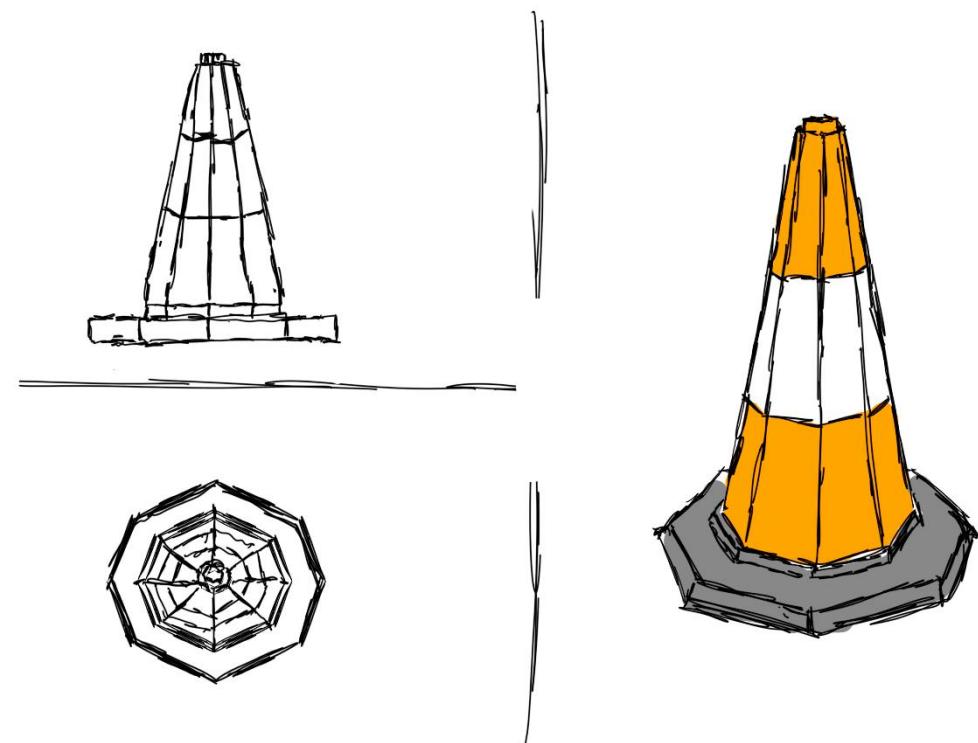


Ilustración 79. Concept de un cono de tráfico

12.3 ASSETS 3D

12.4 DISEÑO VISUAL DE LA UI

13 SONIDO

13.1 EFECTOS DE SONIDO

13.2 MÚSICA

14 PRODUCCIÓN

14.1 MODELO DE NEGOCIO DEL JUEGO Y PLAN DE FINANCIACIÓN

14.1.1 INFORMACIÓN SOBRE EL USUARIO

¿Quién es?	<ul style="list-style-type: none"> Existen dos tipos de usuarios, un primer usuario-cliente que sería una entidad financiera/educativa, que obtiene datos del usuario final mediante analíticas, y un usuario final, que es el niño que juega al videojuego y genera las métricas.
¿Qué quiere?	<ul style="list-style-type: none"> Por parte del cliente-usuario, quiere obtener las métricas y potenciar el pensamiento computacional de los niños. Por parte del usuario final, quiere diversión y entretenimiento sin tener la sensación de estar haciendo una actividad académica.
¿Aficiones?	<ul style="list-style-type: none"> El usuario final es un niño de edad temprana, entre los 8 y 12 años, que frecuenta videojuegos de corte actual, de todos los géneros. Tiene nociones y se asume que comprende el funcionamiento-mecánicas del medio audiovisual.
¿Situación?	<ul style="list-style-type: none"> Por parte del cliente-usuario, su situación es de dependencia con instituciones públicas/privadas para poder financiar proyectos innovadores como el estudio del Pensamiento Computacional en centros educativos. Por parte del usuario final, su situación es la de adquisición habilidades formativas a través de innovaciones que sustituyen la educación tradicional. Esta gamificación camufla la docencia en sí misma.
¿Actividad?	<ul style="list-style-type: none"> Por parte del cliente-usuario, su actividad son labores educativas, administrativas y corporativas. Por parte del usuario final, su actividad es la asistencia y participación en las clases como alumnos del centro educativo.
¿Cómo es?	<ul style="list-style-type: none"> Por parte del cliente-usuario, es una entidad educativa compuesta por expertos en docencia y que busca encontrar métodos más efectivos y vanguardistas para generar recursos y/o enfoques académicos. Por parte del usuario final, es una persona con energía que necesita reposo entre clases, que aprende mejor a través de metodologías lúdicas que tradicionales.
¿Qué necesita?	<ul style="list-style-type: none"> El cliente-usuario necesita conocer las capacidades de sus estudiantes de cara a mejorar su método de enseñanza. Al conocer sus aptitudes, la de los usuarios finales, podrá dirigir mejor sus esfuerzos a la hora de personalizar la educación, plan académico, etcétera. Asimismo, la entidad educativa mejorará su posición de prestigio al ofrecer mejores programas de enseñanza.

14.1.2 MAPA DE EMPATÍA

Se muestra a continuación el Mapa de Empatía del usuario final, niños y niñas entre 8 y 12 años:

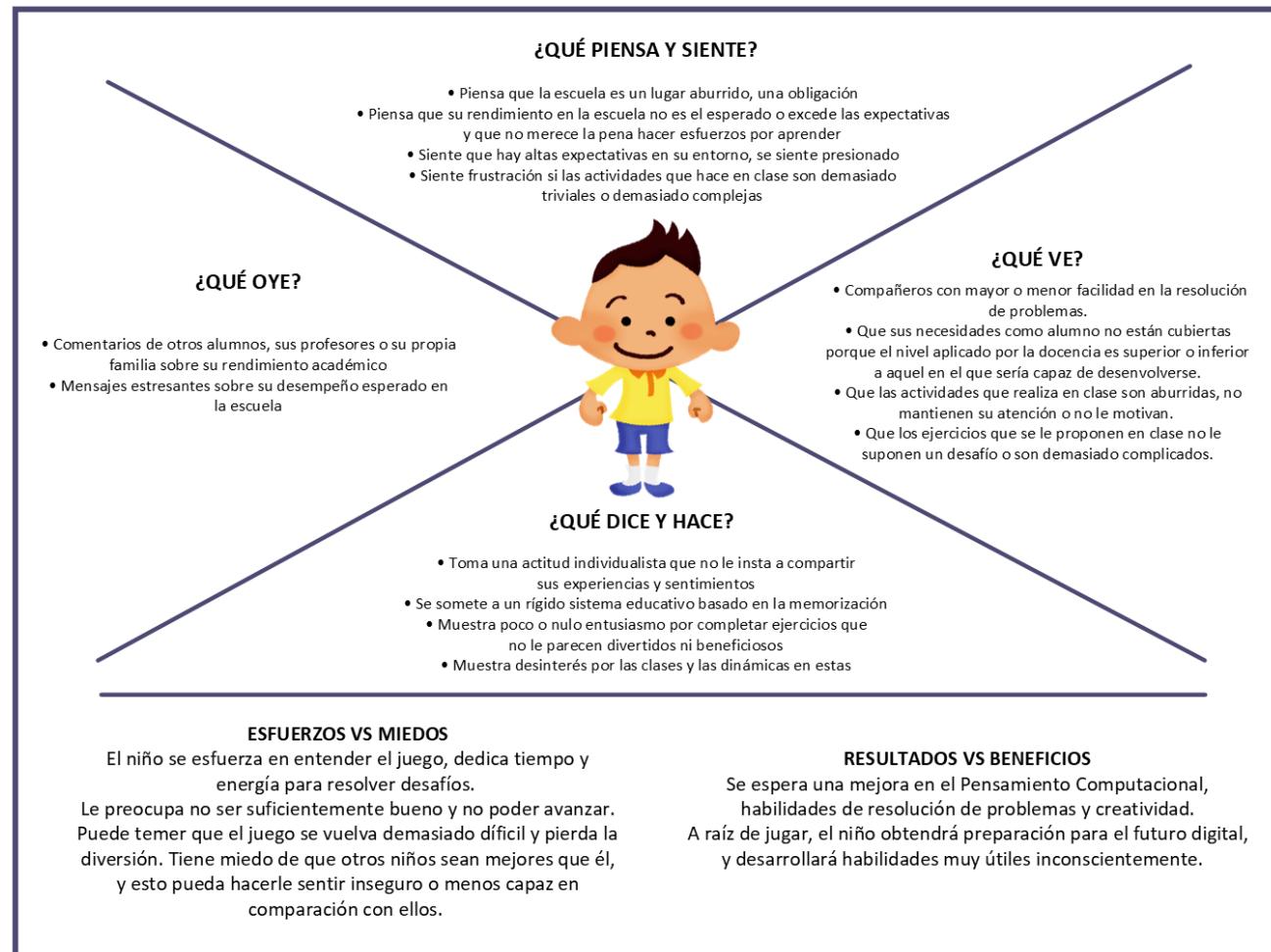


Ilustración 80. Mapa de Empatía del usuario final

14.1.3 CAJA DE HERRAMIENTAS

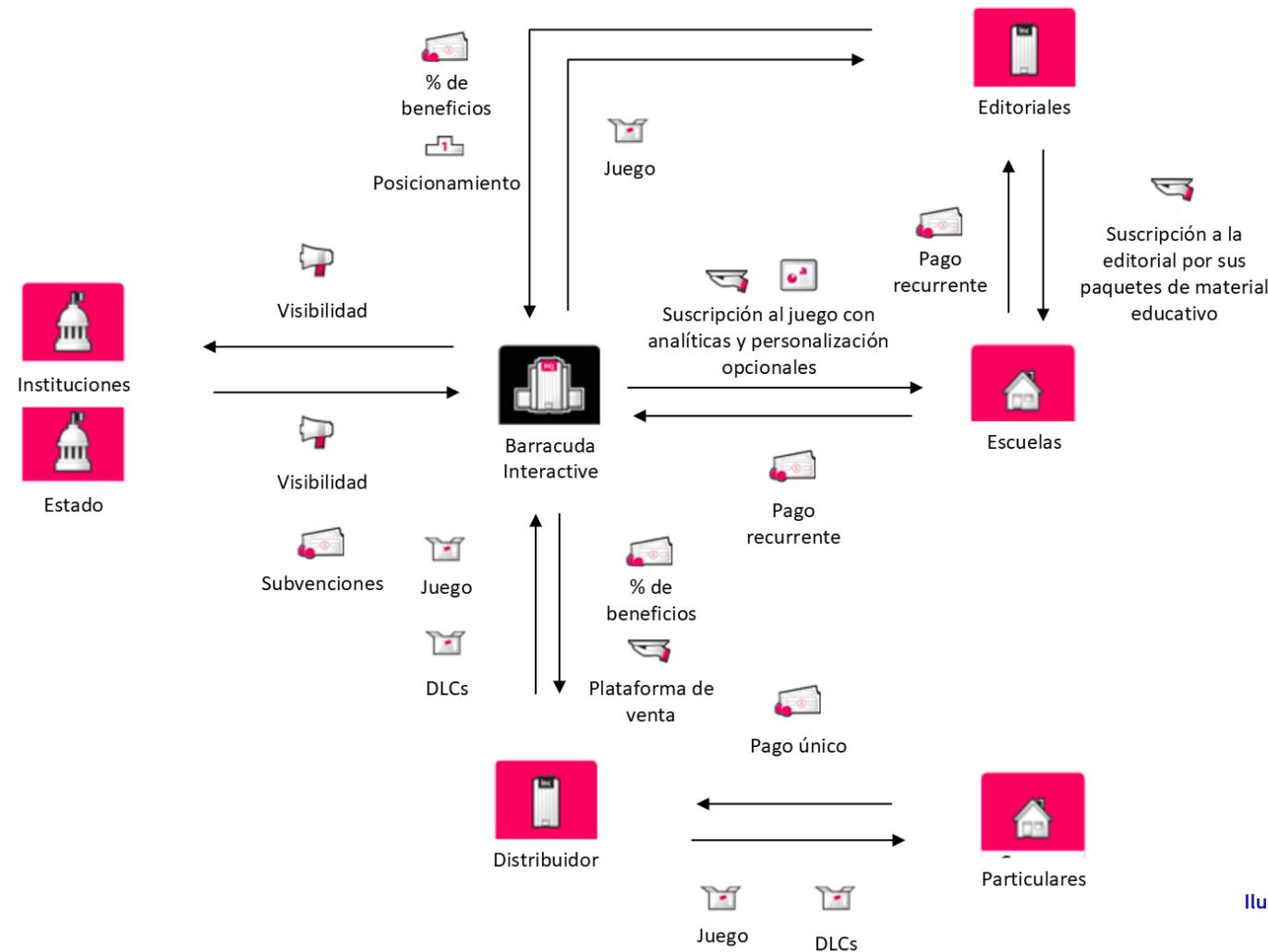


Ilustración 81. Caja de Herramientas

14.1.4 MODELO DE LIENZO

Se presenta seguidamente el Canvas elaborado para *Rally Team Tactics*:



*

Acuerdos institucionales: mantener conversaciones y llegar a acuerdos con instituciones nacionales e internacionales y el Estado para financiar nuestro videojuego y promover su uso en las aulas.

Integración con el modelo educativo: asegurarse de que el producto sea adecuado a los currículos escolares actuales.

Promoción y marketing: promoción del videojuego en ferias del sector educativo, en colaboración con las editoriales, en las plataformas de distribución y en redes sociales.

Mantenimiento y desarrollo de nuevo contenido: supervisión de la recepción al juego tras el lanzamiento para el despliegue de parches y el potencial desarrollo de contenido adicional.

Ilustración 82. Canvas del modelo de negocio

14.2 MARKETING

CONCLUSIONES

GLOSARIO

TERMINOLOGÍA DEL JUEGO

Término en español	Término en inglés	Significado
Evento	Event	Conjunto de pruebas de cuatro días de duración: los tres primeros para pruebas cronometradas en los tramos de la etapa, el último para la etapa al completo.
Prueba	Time Trial	Competición contrarreloj celebrada cada día de un evento, puede ser un tramo cronometrado o la etapa completa cronometrada.
Día	Day	Uno de los cuatro días durante los cuales se celebra un evento. A cada día le corresponde una prueba, ya sea un tramo o la etapa completa cronometrada.
Etapa	Leg	Circuito punto a punto cronometrado completo correspondiente a un determinado evento. Está compuesto por tres tramos que se han probado los días anteriores.
Tramo	Stage	Una tercera parte de la etapa del evento en progreso. Se dedica un día para realizar una prueba punto a punto exclusiva de tramo, para luego conformar una de las tres partes de la etapa completa. Está compuesto por diez secciones de pista.
Sección	Section	Trozo de pista de una longitud y curvatura concretas. Se conectan para formar tramos o etapas.
Punto de Control	Checkpoint	Puntos de la prueba que el jugador puede y debe seleccionar para elegir instrucciones antes de ejecutar. Estos se sitúan al inicio de la prueba y en todas las intersecciones entre secciones. El punto de control final no es seleccionable, pues es la meta del tramo o etapa y no tiene sentido asignarle instrucción, únicamente sirve para monitorear el tiempo del jugador.
Línea de Meta	Finish Line	Último punto de control de un tramo o de una etapa (es decir, de una prueba). No se puede seleccionar para elegir instrucción, solo tiene como propósito marcar el final y pedir la muestra del tiempo y del diferencial.
Instrucción	Instruction	Conjunto de nota y temperamento asignado a un determinado punto de control por el jugador y que va a determinar el

		comportamiento del vehículo en la sección siguiente.
Nota	Pacenote	Código nemotécnico representado en forma de ícono en el juego que indica al piloto las características de la sección siguiente. Determina el trayecto que va a intentar seguir el coche en la sección.
Temperamento	Temper	Valor continuo que indica al piloto con qué nivel de cautela o agresividad debe acometer la sección siguiente. Determina cuando se frena, acelera o se mantiene la velocidad en la sección.
Rampa	Ramp	Obstáculo sólido de relieve que implica una inclinación de terreno en subida seguida de una bajada de la inclinación opuesta.
Meseta	Plateau	Obstáculo sólido de relieve que implica una inclinación de terreno en subida seguida de un tramo liso en altura que termina en una bajada de la inclinación opuesta a la de entrada.
Depresión	Dip	Obstáculo sólido de relieve que implica una inclinación de terreno en bajada seguida de un tramo liso en profundidad que termina en una subida de la inclinación opuesta a la de entrada.
Charco	Pool	Depresión que contiene agua y que reduce significativamente la velocidad del coche.
Reconocimiento	Reconnaissance	Fase de <i>gameplay</i> donde el jugador debe seleccionar todos los puntos de control de la prueba y asignarles instrucciones. Entonces podrá ejecutar, pasando a la fase de ejecución.
Ejecución	Execution	Fase de <i>gameplay</i> donde el coche recorre el trazado de la prueba de acuerdo a las instrucciones seleccionadas por el jugador durante el reconocimiento. En cualquier momento el jugador puede reiniciar la prueba, volviendo a ejecución. Es obligatorio reiniciar si el coche sale por completo de la pista o se queda sin recursos. En caso de que se complete la ejecución exitosamente (el coche cruce la meta), se da la opción de aceptar el resultado también.
Combustible	Fuel	Recurso de juego que se consume progresivamente de 100 a 0 en función de la velocidad, aceleración y frenado. El agotamiento de este supone que el coche quedaría varado, lo que fuerza al reinicio.
Daño	Damage	Recurso del juego que consume progresivamente la integridad del vehículo

cuando se producen colisiones con obstáculos del entorno. Va de 0 a 100, llegar a 100 supone daños irreparables en el coche y fuerza al reinicio.

Tabla 3. Glosario de terminología del juego

ACCIONES DENTRO DEL JUEGO

Acción en español	Acción en inglés	Significado
Seleccionar Punto de Control	Select Checkpoint	Selección de un punto de control de la prueba para poder elegir instrucción para este. Acción solo disponible durante el reconocimiento.
Deseleccionar Punto de Control	Deselect Checkpoint	Deselección de un punto de control de la prueba para guardar la instrucción elegida en este, si es que se ha elegido alguna. Acción solo disponible durante el reconocimiento.
Elegir Instrucción	Choose Instruction	Elección de una nota de código nemotécnico y un temperamento (que por defecto se pondrá en 0.5 al elegir nota) para el punto de control que se tiene seleccionado. Acción solo disponible durante el reconocimiento cuando se ha seleccionado un punto de control.
Limpiar El Circuito	Clear Track	Eliminación de todas las instrucciones asignadas a cada punto de control de la prueba para empezar de cero. Acción solo disponible durante el reconocimiento.
Ejecutar	Execute	Supone el comienzo de la ejecución, es decir, el coche arranca y sigue las instrucciones en cada punto de control. Acción solo disponible durante el reconocimiento cuando se han seleccionado instrucciones para todos los puntos de control de la prueba.
Reiniciar	Restart	Supone el regreso del coche, con todos los recursos reestablecidos, al inicio de la prueba. No elimina ninguna instrucción seleccionada. Acción solo disponible durante la ejecución.
Aceptar Resultado	Accept Result	Aceptación del tiempo obtenido en una prueba junto con el diferencial que lo compara con el histórico. La otra opción sería reiniciar. Tras la aceptación, el tiempo quedaría registrado y se daría la opción de pasar a la siguiente prueba si es que la hay (si no, mostrar la clasificación, pues se acaba de completar la etapa) o de volver al menú de selección de prueba. Acción solo disponible durante la ejecución si se ha cruzado la línea de meta.

Tabla 4. Glosario de acciones dentro del juego

HISTORIAL DE VERSIONES

VERSIÓN 0.1 (26/09/23)

Borrador del GDD, versión de presentación de la idea inicial de acuerdo con el tema:

- Escritura de *Introducción*.
- Escritura de *Referencias y concepto de juego*. Se incluyen diagramas de flujo para el *gameplay loop*.
- Escritura de *Objetivo del proyecto: enseñanza del Pensamiento Computacional a niños pequeños*. Se incluye una tabla sobre las destrezas.

VERSIÓN 0.2 (28/09/23)

Inclusión de la plantilla con todos los capítulos y apartados que se consideran necesarios en este momento.

VERSIÓN 0.3 (05/10/23)

Modificación del concepto de juego y explicación de mecánicas inicial:

- Cambio de *Concepto de juego y bases de la jugabilidad* para adaptarse mejor a la temática y simplificar el gameplay. Se incluyen nuevas ilustraciones y diagramas.
- Escritura de los dos apartados *Cámara y perspectiva* y *Gameplay loop*. Se incluyen ilustraciones y diagramas.

VERSIÓN 0.4 (08/10/23)

Ampliación de las mecánicas, ahora describiendo obstáculos y gestión de recursos:

- Escritura del apartado *Mecánicas de juego en detalle*. Se incluyen ilustraciones y diagramas explicativos.

Explicación de la progresión en el juego:

- Escritura del apartado *Objetivos del jugador* y *Estructura del juego*. Se incluyen ilustraciones y diagramas explicativos.

VERSIÓN 0.6 (12/10/23)

Cambios en la elección de vocabulario para los diferentes elementos de juego, de tal manera que este sea consistente en todo el documento.

Modificación del *gameplay loop* general y adición de secciones para la explicación detallada de los puntos de control en las fases de juego:

- Retoque en los diagramas de flujo de *Gameplay loop* para hacerlos consistentes en vocabulario con el resto del documento y reflejar cambios en el diseño del juego.
- Escritura de los apartados *Colocación de notas en la pista* y *Cantado de notas durante la carrera*.

VERSIÓN 0.7 (17/10/23)

Reorganización de los capítulos referentes a UI/UX y arte:

- Ahora el diseño de UI va antes del capítulo de arte y se cambia el último apartado a *Diseño esquemático de la UI*, para incluir en una versión futura los bocetos o esquemas de las pantallas para todas las plataformas.
- El apartado de arte ahora se desglosa en más apartados, aún vacíos. Se incluye uno para el diseño visual de la UI y otro para *concept art*.

Integración de parte del contenido del apartado de UI/UX que ya está listo:

- Integración de los apartados *Requisitos de la interfaz* y *Diagrama de flujo de navegación*. Se incluyen diagramas.

VERSIÓN 0.8 (20/10/23)

Escritura de *Trasfondo del juego*, con una ambientación con imágenes de referencia y una narrativa donde se cuenta la historia del copiloto al que el jugador encarna en el juego.

Integración del modelo de negocio, revisado, en el GDD, concretamente en el apartado *Modelo de negocio del juego y plan de financiación*.

Paleta de colores del documento de diseño visual del juego, logo del juego, del estudio y Splash Art añadidos.

VERSIÓN 1.0 (22/10/23)

Versión del GDD para la entrega del prototipo del videojuego. En esta iteración se incorpora contenido a:

- Apartado *Diseño esquemático de la UI*, con mockups de las pantallas diseñados hasta la fecha (integración del documento de diseño de interfaces revisado).
- Apartados *Estética del juego* (integración del documento de diseño visual del juego revisado) y *Concept*, con imágenes y bocetos.

VERSIÓN 1.1 (03/11/23)

Reorganización de los capítulos del documento, en especial de la parte que cubre las mecánicas y la jugabilidad, para poder expandir en ellas sin sobrecargar ningún capítulo en particular.

Cambio de vocabulario y terminología de acuerdo con el glosario que se incluye ahora al final del documento en *Glosario*. Este se ha elaborado con el propósito de mantener un lenguaje común respecto a los elementos y acciones más importantes del juego, tanto en la documentación como en el diseño y desarrollo de este.

Integración de nuevos documentos de diseño en el GDD:

- Adición y revisión del *Sistema de daños*
- Adición del *Glosario*

Cambio del logo y de elementos de la portada y cambio de estilo del documento.

VERSIÓN 1.2 (07/11/23)

Modificación del vocabulario usado a lo largo de todo el documento referente a los elementos del juego de acuerdo con el glosario. También se han modificado esquemas y diagramas que contenían vocabulario inconsistente en los capítulos de *Gameplay loop* y *Progresión*.

Reorganización de apartados y corrección de la numeración.

Integración y corrección de documentos:

- Documento de Diseño Visual final para el apartado *Diseño visual del juego*