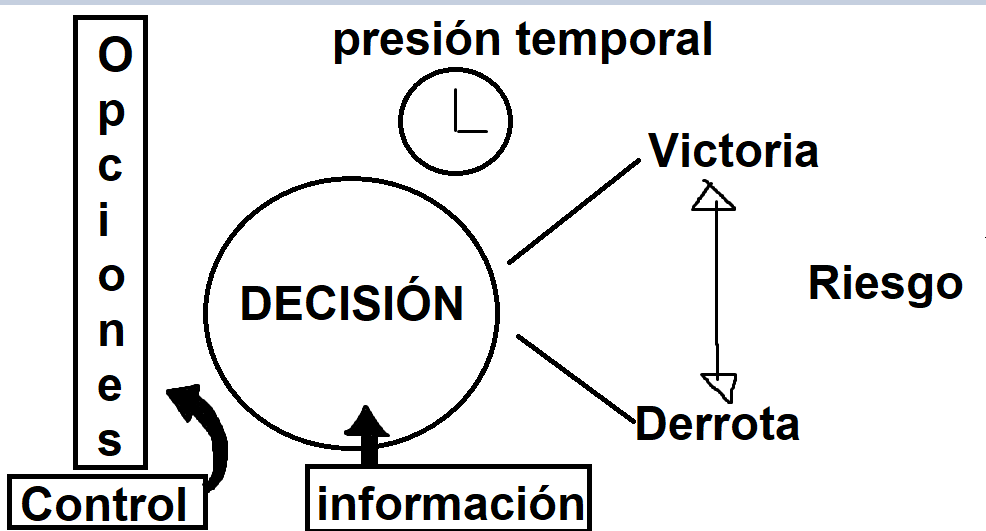
# Gameplay de corto plazo

Mecánica = gameplay de corto plazo

La más divertida de hacer, pero la más complicada.

El punto de partida es nuestro átomo de la Gameplay (lo tomamos como punto de partida).



Esto realmente apela a la gameplay de corto plazo

Ejemplo: piedra-papel-tijera. La gracia es que cada uno gana al otro. Aunque no tenga sentido que el papel mate a la piedra voy a hacerlo así para generar riesgo. Y así nos funciona el diagrama de arriba.

El control es lo de 1,2,3 y en ese momento muestras (es como el penalti del portero). Información ¿la hay? Parece un juego de azar pero no lo es. Estadísticamente cuando ganas tienes tendencia a repetir lo que has ganado y al revés.

Es frágil por la emergencia. Cuando la piedra mata a los dos otros, no hay emergencia. Siempre empataríamos y se dejaría de jugar.

Reglas del tetris:

1. Diseño de las piezas
2. Gravedad y colisión: cuando se deposita una pieza colisiona
3. Rotación (90 grados izq drcha)
4. Desplazamiento (izq y drcha)
5. Borrado de fila (si está completa)
6. Aceleración: a medida que pasa el tiempo va más rápido.

Si toco una regla el juego se va a tomar por culo

Ec de la superficie de una esfera: x2 + y2 + z2 = R2, solo esos puntos pertenecen a la esfera. Los demás no. Nosotros con las mecánicas igual. Debe ser de una única forma.

Mecánicas hay POCAS:

## Twitching (parpadear)

Cualquier mecánica que se basa en pulsar controles muy rápido.

Ej: God Of War: hay que llenar una barra pulsando un botón muchas veces.

Hay una:

* **Stamina** **muscular** (prima, es lo importante)
* Precisión (importante precisión, pero menos)

Te hacen pensar que te estás cansando dándole muchas veces.

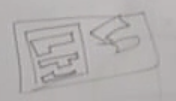
Pero luego te pueden hacer también que poner una parte de precisión. X,X,Y,X

Tienen **fatiga**. Una lucha de Street Fighter dura 45 segundos por eso.

Se intenta modelar un juego de **ESFUERZO**, porq se siente natural. En la vida real requeriría un esfuerzo

## Mecánica de puzzle

Ej: Mass Effect. Hay una mecánica de hackear.



Hay que poner una pieza en su sitio.

Es distinto a la mecánica anterior:

* Interpretación (lectura): el puzzle hay que hacerlo rápido 🡪 presión temporal
* Decidir

En este caso hay que pensar rápido.

Ejemplo: VR Big Saber. La mecánica es igual a la de antes. Vienen muros y tienes que esquivarlas. Tengo unas piezas y tengo que hacer el matching muy rápido y tengo que decidir.

## Mecánica de sincronización

Tengo que hacer algo en un momento concreto. Tengo que hacer algo en un momento concreto y si hago eso gano y si no pierdo

* Mario: al saltar de plataformas
* Coches: voy a la curva, la curva viene y digo 3,2,1,… y giro
* Combate batman: tradicionalmente era twitching (aquel que tiene el dedo mas rápido gana), hasta que llega batman y assassin’s creed. Pero no se puede hacer una aventura con twitching porq acabo muerto. Entonces lo derivo hacia aquí. Si el ataca, yo tengo un determinado tiempo para esquivar. Y yo aprovecho sus ventanas de oportunidad para atacar yo y ganar.

Con esto, el

* Souls: hay mucho castigo cuando no aprovechas tus ventanas.

Hay es muy importante la **precisión**.

*Vayas a donde vayas, el arroz con cosas está bueno. Y la sopa también. Es juegos igual.*

## Ahora ¿Cómo nos inventamos mecánicas nuevas?

Por ejemplo el escalar de Shadow Of The Colosus.

Cuento del borracho: borracho pierde las llaves de casa, de noche. Seguro que busca debajo de la farola pq hay luz. Primero donde se siente seguro.

### Prototipaje – playtest

1. **Identificar un referente**: ¿Cuál es el juego más conocido de lo que intento hacer?

Por ejemplo el Rocket League.

1. **Levantamos un Core** (prototipo rápido de las mecanicas): por ejemplo un 3C. Un coche, miro la cámara veo tal. No se trata de encontrar, se trata de innovar.

**Nada inspira mas la creatividad que un prototipo.** Jugando a juegos de coches si que te puede inspirar. El prototipo debe

* + - **Tener features básicas**
    - **0 innovación**
    - **Coste bajo.**

Esto es para tirar.

1. **Variamos el CP (Core Prototype) añadiendo features:** en el unreal el translocalizador, es como una granada pero te trasladas a los 3 segundos donde la has tirado.
   * **Auto Impacto,**
   * **Bajo coste:** unas dos semanas**.**
   * **Relacionadas con la gameplay.**

Si coges el Gris es un juego innovador (innova en la estetica). Aquí estamos para innovar en gameplay. Debemos ser inspirados por el prototipo.

1. Prototipo super cutre (cubos, prints): y hay que abstraerse y comprobar que todo va bien
2. Playtest: partida en la que todo el mundo piensa en eso. Partida en la que todo el mundo mira las granadas. Hay que evitar:
   1. El todo lo nuevo mola. Que guay es original. Pero, es sostenible en el tiempo?
   2. Las features lentas: hay algunas features que pueden requerir tiempo para aprender a usarlas. Por ejemplo las granadas. Ojo, porq esto normalmente es mejor indicio, es mas prometedor.
3. Discard (descartamos lo que no).
4. Volver al punto 3.

* 3c’S:
* Tema:
* Metáfora:
  + Larry: juego de flirt, ligar. El objetivo es ligar. Querían hacer un juego de sexo pero es difícil. Vamos a hacer como un juego de naves. Entonces eras un espermatozoide y tenias que esquivar mierdas para llegar a dentro.



* + Con la granada q teletransporta igual, es otra metáfora.
  + Valdus / Diablo: metáfora del inventario. Tenemos al héroe donde arrastras cosas al lado



**Innovar no es inventar es encontrar una relación que a otro no se le ha ocurrido antes. Coger un coche y meterle una pelota (rocket league).**

Skull & bones:

El riesgo no es el coste. Si he hecho 4 uncharter y hago el 5, y cuesta 200k. Eso es caro, pero NO arriesgado. Que sea arriesgado es probabilidad de fracaso.

Los de Ubisoft si es muy innovador, y tiene riesgo, le han añadido un modo historia porq ya tenían experiencia con modo historia que funciona.

## Ejercicio de innovar a corto plazo

Estamos en una empresa y nos dicen de innovar en diálogos. No puede ser que en 30 años solo tengan 4 opciones para cambiar los diálogos. Vamos a innovar para que los diálogos sean divertidos, sean gameplays.

Por ejemplo, Uncharter.

**¿Por qué el sistema clásico de diálogos no tienen gameplay? ¿Qué información tengo? ¿Hay riesgo?**

Los diálogos no son divertidos porque las decisiones que se toman , no se puede elegir entre varias opciones, tampoco hay ningún tipo de riesgo que te haga ganar o perder. Ni presión temporal, puedes estar leyéndolo todo el tiempo que quieras.

* Muchas veces no hay riesgo y eligo al azar
* No hay presión temporal (tiempo infinito para tomar la decisión)
* Demasiada información (mucho texto).
* Opciones no relevantes
* El control es trivial

¿Un dialogo tiene riesgo? Depende. Normalmente en el dialogo siempre hay un objetivo. (policía interrogando, pareja flirteando…).

* Entonces ya tengo una condición de victoria. (A través de n frases llego a mi objetivo)
* Tiempo: sí pero no forzado.
* Demasiada información: en mi frase en concreto no se sabe exactamente si me estoy acertando al objetivo. En Mario sí, si salto una plataforma me acerco a mi objetivo (cada paso me acerca al final es HILL-CLIMBING), El problema del dialogo es que no es así. El cubo de Rubik tampoco, pq yo resuelvo una cara pero no me lleva a la solución final.

Y esto es importante porque ¿QUÉ INFORMACIÓN le doy al player para que se pueda acercar a la solución final? Si la información es perfecta, la decisión es trivial.

* + Un ejemplo es: Frio frio caliente caliente. Pero la decisión al final será trivial también.

Una posibilidad es que las soluciones vayan con círculos que se van yendo y desapareciendo. Hay que hacerlo rápido

Seguimos, control simple.

Juego: Farengeit

Complicar el control mostrando el texto en los círculos pero luego que se vayan moviendo por la pantalla

Si te estoy entrevistando y antes he visto una cinemática, pues saco info de actos previos. O si estoy en su casa y he visto una foto en mi casa ya sé que eso es así.

Pero sin información previa, ¿Cómo se haria? Barra de nervios. Y cada frase me dice cuánto sube los nervios.

Pongo objetos a la izq y tienes que poner los círculos en tu personaje o algo así, arrastrarlo. (psicomotriz).

Y ponemos varias barras.

Vienen frases muy rapido y tenemos que hacer que confiese antes de que acaba el tiempo. Y además hay también un sistema de puntos. Cuanto antes mejor te saque la verdad.

Otro ejercicio: problema que es algo de vida real. Montar muebles de Ikea. No existe ese juego. Tengo que empezar desde cero. Vamos a hacer un juego de dentistas.

**1ERA FASE**

Hay que empezar desde abajo, lo cojo y pienso en un videojuego pensar en las 3 c’s.

**Character**: yo soy el dentista

**Camera**: no hay respuesta incorrecta, saldrán juegos diferentes.

* + Camara a la boca en 1er plano
  + First person y puedo mover al dentista (en VR)
  + Camara como thesims, saldrá un juego de gestión de la consulta.

Si quiero hacer un juego de dentista, no podemos traicionar las expectativas. Yo no hago una consulta de dentistas, yo lo hago para ser el dentista, no el consultor.

**Control**:

* + Aim en VR
  + Coger herramienta
  + Dejar

**Segunda fase:** Analizo el diagrama según la realidad

Riesgo:

Presión temporal: si (aunque si hay anestesia no)

Información:

Y la propuesta la acabo con un mock-up, un sketch de como se ve esto. Con boli, Photoshop, o lo que sea. Hay que verlo.

En diálogos estaba innovando pq ya existía los diálogos, en los peluqueros no innovo, estoy creando una mecánica. Y lo que interesa (2/3h para pensar y 1h para escribir documento). Las reglas resumidas en modo guiones:

* Pistola de taladro con barra de sobrecalentado

PELUQUERIA. Cortar el pelo, no teñimos. En femenino, solo hay una mujer en la peluqueria. No es sandbox (no queremos hacer lo que queramos), sino que la mujer quiere un corte de pelo especial. Lo quiere largo, corto, …

2/3 paginas contando el mock-up

FLUJO (FLOW)

Buscar en Wikipedia Flow theory

Mide cual es el rendimiento de la gente usando algo. Lo hacemidiendo dos ejes (el reto y performance)

* Reto: no es lomismo saber usar un coche que otra cosa
* Rendimiento/Performance: lo bien que lo uso

Un pizarrón blanco

Descripción generada automáticamente con confianza media

Por en medio está el Flow State. Pasa lo mismo con el móvil. Conforme vamos aprendiendo el eje va hacia la derecha y hay dopamina.

El estado de flujo es lo normal y lo de la izq es la bici.

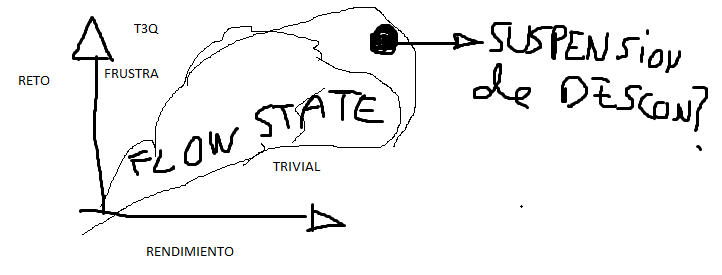
Un dibujo de una persona

Descripción generada automáticamente con confianza baja

Hay 3 zonas:

* Lo de abajo drcha es **trivial** (skylanders). Mi rendimiento supera al reto
* Arriba izq es **frustrante** , el reto es superior a mi rendimiento, por más que lo intento no puedo.

Tanto arriba como abajo se produce abandono.



Suspension de desconfianza (suspensión of disbelief): tu cuando llevas 10/15 min ahí en el videojuego inmerso y te dejas llevar. Lo puedes experimentar en el cine y estas ahí dentro y de repente te dan un codazo y vuelves.

* Se le llama Flow porq hay un Flow de dopamina continuo.
* Hay un Disfrute
* Hay una perdida de sensación del tiempo

Esto es completamente positivo. Queremos que el usuario abandone el mundo rea.

EJ: Cogemos a 3 personas que juegan coches. Hay 3 personas y dado un mismo reto, que es igual de difícil. El rendimiento es distinto (uno muybien, otro regular y el otro fatal). Ahora hay un problema. Tenemos que producir el estado de flujo. Solución: el Flow variable. Hay 3

* NADA: si no hacemos nada estamos echando del juego al que juega muy bien y al que juega muy mal.
* Niveles de dificultad: (fácil, mediano, dificil). Hacer eso es muy fácil. Se coge la métrica que queremos jugar y se le mete un multiplicador (al daño que meten las armas por ejemplo). Se puede también cambiar el numero de enemigos o algo así.

Problema: Ojo esto solo se puede hacer en SINGLEPLAYER

* Niveles de dificultad autoajustados: esmala idea. Luchas contra un boss, te crujen, luego te crujen. Y trackean cuantos intentos llevas y le preguntas si quieres bajar la dificultad. ESO ES MALA IDEA. Acabas de llamar TONTO al player. Esta bien hacerlo pero nunca le podemos decir eso.
* DDA: Dynamic difficulty adjustment: meto una pequeña IA que analiza que hace el player. Y esa IA tiene “el mando de volumen” que regula la dificultad. Ejemplo: en el uncharted que como una línea recta de distintas secciones, en cada sección monitoriza cuantos intentos llevas. Y el uncharted hace eso, baja el daño de las armas de los enemigos, aunque tu por dentro pienses que eres bueno, y te da dopamina. Y aunque estés en modo difícil, dentro del difícil pues te lo pueden bajar un poco.

Una forma de implementación son los Behaviour Trees. Entra solo a algunos nodos en caso de que el multiplicador este por encima de x, de manera que si estoy en modo muyyy difícil entro en todos los nodos y el enemigo puede defenderse, recargar, hacer muchos mas comportamientos. SI no, solo entra a atacar por ejemplo.