Catedral–Laberinto de Axiom

Diseño de Juego (borrador)

Kevin Esguerra Cardona

24 de abril de 2025

Índice

1. Concepto–Mantra

- **Runs cortas, caos épico, poder exponencial**.
- Tres pilares:
  1. El héroe inicia débil y se convierte en leyenda en ≤ 20 min.
  2. Cada partida debe sentirse irrepetible (roguelike procedural).
  3. La emoción prima sobre la narrativa extensa.

2. Trasfondo (microhistoria)

Un programador universitario muere de agotamiento y despierta en la *Catedral–Laberinto de Axiom*, un plano que devora mundos. Una voz le promete regresar a casa si derrota al *Custodio de Nivel 5* antes de que el **reloj del caos** (20 minutos de tiempo real) llegue a cero.

3. Clases jugables y estadísticas base

Clase	HP	MP	ATK_F	ATK_M	**INT**	DEF	RES	VEL	CRIT
Caballero	120	40	18	4	6	14	6	7	5 %
Guerrero	110	20	**22**	0	4	10	4	9	7 %
Paladín	100	60	15	10	12	12	12	6	5 %
Mago Arcano	70	120	6	**22**	**24**	6	14	8	3 %

Cuadro 1: Atributos iniciales. **INT** amplifica el daño/heal de los hechizos; VEL controla frames de invulnerabilidad y cadencia.

Cada 90 s vivos el jugador asciende un *Rango de Caos* y todos los atributos crecen un 5 % de forma multiplicativa.

4. Sistema de magia

Escuela	Ejemplos (Nv 1–5)	Atributo clave	Notas
Elemental	Chispa → Bola de Fuego → Tormenta Eléctrica → Lluvia Meteórica → Nova de Plasma	INT	Alto daño de área.
Mística	Destello → Curar → Sello → Puerta Dimensional → Paradoja	INT	Apoyo / control temporal.
Fuerza	Impulso → Rompe-suelo → Onda de Choque → Lanza Éter → Colapso	STR	Sin coste de MP; entra en CD.

Cuadro 2: Escuelas y progresión de hechizos.

Fórmulas básicas

$$\text{Coste MP} = 5 \times \text{nivel} + \frac{t}{2}, \qquad \text{Efecto} = \text{base}\left(1 + \frac{\text{nivel}}{2}\right)\left(1 + \frac{t}{600}\right)$$

donde t es el tiempo (s) sobrevivido en la run.

5. Objetos y progresión dinámica

Evolución de ítems

$$\text{bonus}(t) = \text{base}\Big(1 + (\rho\, t/300)^{\alpha}\Big)$$

Calidad	base (%)	ρ	α	Ejemplo
1	3–5	0.5	0.8	Espada oxidada
2	6–10	0.8	1.0	Espada templada
3	12–18	1.1	1.2	Mandoble rúnico
4	20–28	1.4	1.3	Bastón del vacío
5	30–40	1.8	1.4	Hoja de singularidad

Tablas de botín por clase (probabilidad)

- **Mago**: bastones 50 %, grimorios 30 %, sellos 20 %.
- **Caballero**: espadas 40 %, lanzas 20 %, escudos 40 %.
- **Guerrero**: hachas 60 %, martillos 25 %, brazales 15 %.
- **Paladín**: mandobles 35 %, mazas 25 %, reliquias 40 %.

6. Enemigos

Lv	Ejemplos	IA	Acompañantes	Magia	HP
1	Goblinoide, Slime	Paseo aleatorio + golpe	—	—	30
2	Chamán, Arquero	Kite + disparo	2–3 Lv1	Elem 1	60
3	Caballero Oscuro	Parry + combo	1–2 Lv2	Elem/Míst 2	120
4	Señor de Guerra	Carga + invocar	8 Lv3	Elem/Fz 3	250
5	Custodio del Caos	Árbol + Q-learning	2 legiones Lv4	Todas 5	1000

Cuadro 3: Jerarquía y capacidades de enemigos.

7. Generador de laberintos

1. Grid 64×64 de celdas de 48 px.
2. *Prim* modificado: un pasillo principal y 30 % de ramificaciones.
3. Sellar 10 % de celdas con trampas (púas, fuego, rayos).
4. Insertar *Salas de Evento* cada 6–8 celdas lineales.
5. Distancia jugador–boss ≥ 40 celdas de Manhattan.

8. Pacing de una run

Tiempo	Cambios globales
0–3 min	Solo enemigos Lv1–2, +10 % consumo.
4–6 min	Se habilita Lv3, trampas dobles.
7–12 min	Aparece el primer Lv4; Rango Caos +1 cada 90 s.
13–17 min	Se abre la puerta del Boss Lv5, cuenta atrás 5 min.
18–20 min	Muerte súbita: +10 % stats enemigos / 30 s.

9. Arquitectura y datos

- Motor **PyGame**. Datos en JSON/YAML editables sin recompilar.
- Patrón *component–based*: `GameObject` + mixins (`Health`, `Inventory`, `SpellCaster`...).
- IA Lv1–4: árboles de estados; Boss: wrapper `QLearner`.

10. Hitos de desarrollo (6 semanas)

1. Semana 1 – Prototipo: grid, movimiento, ataque básico.
2. Semana 2 – Generador de laberintos y sistema de clases.
3. Semana 3 – Ítems dinámicos, loot y UI mínima.
4. Semana 4 – IA Lv1–3 y sistema de magia completo.
5. Semana 5 – IA Lv4–5, balance inicial y partículas.
6. Semana 6 – Sonido, pulido fino y empaquetado.

Anexo A
Ajustes y aclaraciones para la Versión 0.2 del GDD

A1 · Cambios de alto nivel

1. **Pilar #1 — Tiempo y progresión**
   Reformular: «Cuanto más tiempo pase, más poderosos se vuelven *tanto* el héroe como sus enemigos. La partida termina cuando uno de los dos extremos rompe el equilibrio.»

2. **Trasfondo**

   - Añadir tono cómico: el protagonista estaba *programando su proyecto final de Computación Gráfica* cuando sucumbe al agotamiento.
   - Eliminar el límite duro de 20 minutos; el *reloj del caos* se redefine como multiplicador de dificultad que escala exponencialmente $(1{+}0{,}05\,t)$.

A2 · Profundización de sistemas

3. **Magia**

   a) Añadir dos escuelas: *Somática* (buffs físicos) y *Entropía* (debuffs y efectos de caos).

   b) Crear ficha por hechizo: {id, escuela, nivel, coste, tiempo_cast, escala, efectos_secundarios}.

4. **Ítems**

   a) Lista exhaustiva (mín. 5 armas, 3 armaduras, 3 accesorios por clase).

   b) Etiqueta `owner = {caballero, enemigo_lv3}` para controlar loot exclusivo.

5. **Sistema de escoltas**
   Definir tres capas jerárquicas:

   Escolta 1–3 mobs que orbitan al líder (`follow_radius`).

   Escuadra 2–3 escoltas; el líder puede `reinforce()` una escolta destruida (CD 15s).

   Legión 2 escuadras; sólo Lv4–5. El Boss usa `spawn_legion()` con CD escalable al reloj del caos.

A3 · Diagrama de flujo principal

- Insertar después de la sección «Pacing de una run».
- Notación BPMN simplificada:

  1. *Inicio* → **Generar laberinto** → **Spawnear jugador** + **mobs**.

2. Bucle `while vivo`: **Actualizar reloj de caos** → **Procesar IA / Eventos** → **Render / Sonido**.
3. Condición de fin: *Jugador muere* **o** *Custodio derrotado* → **Pantalla de resultados**.

A4 · Aclaraciones técnicas incorporadas

Patrón Component-Based se mantiene para las entidades del juego; MVC se reserva exclusivamente para la interfaz (*HUD, menús*).

Algoritmo Prim seguirá siendo la base del laberinto; añadir parámetro `branch_density` en JSON para balancear ramificaciones.

IA Árboles de estados para Lv1–4; Boss Lv5 usará Q-Learning *pre-entrenado* (tabla Q serializada) para minimizar coste en tiempo real.

Estos ajustes servirán de guía para la redacción de la **Versión 0.2** del GDD, que deberá reflejar todos los cambios de diseño y ampliar las secciones descritas.