

Pans & Dungeons

Documento de Diseño Maestro y Plan de Producción Técnica (GDD)

FICHA TÉCNICA Y ESPECIFICACIONES DEL PROYECTO

Módulo C++: PansDungeons | Motor: Unreal Engine 5.3+ (C++ & Blueprints)
Arquitectura de Red: Listen Server (P2P 3-4 Jugadores) | Herramienta de Agentes: AntigraVity CLI (agy)
Estética: Ragdoll / Muñeco de Trapo Estilizado | Formato: Boss-Rush Cooperativo con Tirada de Dados D&D

1. Visión General y Bucle Jugable (Core Loop)

Pans & Dungeons es un juego de acción RPG cooperativo en 3ª persona diseñado para partidas ágiles y desenfadadas ("chill") de 3 a 4 jugadores. El juego combina el humor absurdo, el tono cómico ("goofy") y la fantasía inspirada en Dungeons & Dragons con mecánicas físicas tipo ragdoll y enfrentamientos directos contra jefes en arenas cerradas.

1.1. Filosofía de Diseño

- Enfoque Casual y Accesible:** Sesiones de 15 a 20 minutos divididas en 4 o 5 salas de jefes sin pasillos ni relleno extenso.
- Cooperación de 3 a 4 Jugadores:** Escala de grupo ideal para mantener visibilidad limpia en pantalla y reducir la sobrecarga de red en arquitectura Listen Server.
- Humor Físico y Absurdo:** Uso intensivo de simulación física ragdoll cuando los personajes reciben impactos masivos, tropiezan o son derrotados.

1.2. Flujo Bucle de Partida

El bucle de juego se estructura en cinco fases consecutivas que garantizan ritmo constante y progresión recompensante:

Fase	Ubicación / Sistema	Descripción Mecánica
------	---------------------	----------------------

1. Lobby	Taberna Central	Selección de clase, arquetipo y equipamiento con cosméticos absurdos antes de la incursión.
2. Combate	Arena de Jefe	Combate cooperativo contra un jefe único con ataques telegrafados y esbirros secundarios.
3. Progresión	Sistema de Datos D&D	Tirada de dados virtuales (d6 atributo + d4 magnitud + dado de riesgo) al superar la sala.
4. Descanso	Tienda del Tabernero	Gasto del oro obtenido en la sala para comprar consumibles, dados trucados o bendiciones.
5. Transición	Level Streaming	Carga sincronizada o teletransporte inmediato hacia la siguiente arena de jefe.

2. Dirección Artística y Pipeline de Creación (Blender → UE5)

El juego adopta una estética amigable de muñecos de trapo (ragdoll estilizado), con formas redondeadas, texturas simples y bordes suaves. Este enfoque simplifica la producción 3D y maximiza la comedia visual.

2.1. Creación de Personajes y Equipamiento Modular

Para acelerar el desarrollo sin sacrificar calidad animación ni rendimiento multijugador, se establece un pipeline estandarizado:

- **Mallas Base en Blender:** Se modela un cuerpo base humanoide de muñeco de tela o saco con bajas poligonizaciones (low-poly estilizado).
- **Compatibilidad de Esqueleto (UE5 Skeleton):** En Blender se adopta exactamente la misma nomenclatura de huesos que el esqueleto predeterminado de Unreal Engine (root, pelvis, spine_01, hand_r, etc.). Esto permite heredar directamente animaciones como ABP_Manny_Combat.
- **Accesorios y Armas Modulares:** Los elementos absurdos (tapas de olla, sartenes, cubiertos gigantes, varitas de cuchara de madera) se modelan como Static Meshes independientes y se anclan dinámicamente en UE5 a los Sockets del personaje (hand_r_socket, head_socket).

2.2. Construcción y Ensamblado de Escenarios

PRINCIPIO DE ARQUITECTURA DE ESCENARIOS

Regla de Oro de Mapeado: Nunca modelar una arena entera como un único bloque en Blender. El espacio de juego se bloquea en Unreal Engine, las piezas modulares se crean en Blender y el ensamblado final se realiza en el editor con NavMesh habilitado.

- 1. Bloqueo en UE5 (Greybox):** Uso del Modeling Mode de UE5 para definir alturas, rampas y áreas de combate. Se prueba el NavMesh Bounds Volume (tecla P) para garantizar superficie verde transitable en toda la arena.
- 2. Kit Modular en Blender:** Modelado de baldosas modulares (2m x 2m / 4m x 4m), muros imperfectos, columnas de cobertura y props cómicos (barriles, calderos, sacos).
- 3. Set Dressing e Iluminación en UE5:** Ensamblado con alineación a cuadrícula (Grid Snapping), colisiones simples, herramientas de follaje (Foliage Tool) y Toon Shading estilizado.

3. Catálogo Completo de Clases y Arquetipos Cómicos

El juego incluye 6 clases principales, cada una dividida en variantes tácticas con identidad y roles complementarios:

Clase	Subclase / Variante	Rol Táctico	Mecánica / Identidad Cómica
Guerrero / Espadachín	Paladín de Cocina	Tanque Defensivo	Tapa de olla por escudo. Taunt estruendoso haciendo ruido con cuchara de palo.
Guerrero / Espadachín	Espadachín Ágil	Ofensivo Melee	Espada larga de juguete. Contraataques rápidos sin escudo.
Asesino	Pícaro Dramático	Daño Crítico & Movilidad	Backstab por la espalda con sangrado. Si corre recto demasiado tiempo, tropieza en ragdoll.
Asesino	Ninja Improvisado	Ataque Rápido / Veneno	Intercala ataques melee y lanzamiento de cubiertos con veneno. Dashes aéreos.
Arquero	Arquero Clásico / Blindado	Distancia / Híbrido	Dispara flechas de pimienta (fuego) o hielo. La variante blindada gana defensa pasiva.
Mago Ofensivo	El Mago Impuntual	Glass Cannon / AoE	Hechizos devastadores con retardo de casteo. Requiere detonación por estados alterados.

Mago Soporte & Tanque	Bardo Desafinado / Golem	Soporte / Control Espacial	Bardo otorga Concentración (sin curas directas). Golem se Reconstruye con regeneración pasiva.
-----------------------	--------------------------	----------------------------	--

4. Matriz de Estados Alterados (Buffs & Debuffs)

Todos los efectos de combate se gestionan centralizadamente en C++ mediante el componente **UCombatStatusComponent**. La matriz combina utilidad táctica con representación visual cómica:

Estado	Tipo	Efecto Mecánico	Efecto Visual Cómico
Sangrado	Negativo	Daño DoT continuo por segundo que escala si el objetivo se mueve.	Gotas de salsa de tomate o líneas rojas cómicas.
Veneno	Negativo	Daño prolongado que reduce la eficacia de regeneraciones pasivas.	Burbujas verdes emanando de la cabeza.
Alteración	Negativo	Incrementa el tiempo de recarga (cooldown) de las habilidades activas.	Relojes o signos de interrogación girando.
Lentitud	Negativo	Reduce el MaxWalkSpeed del CharacterMovement en un porcentaje.	Pies pegajosos con miel o barro animado.
Ceguera	Negativo	En jugadores oscurece la pantalla; en la IA reduce el rango de percepción.	Una olla o cubo atascado en la cabeza.
Reducción Atk/Def	Negativo	Disminuye el daño infligido o incrementa el daño recibido.	Escudo roto o espada doblada flotante.
Marcado	Positivo	Incrementa la probabilidad de recibir golpes críticos del grupo.	Una diana de tiro al blanco en la espalda.
Concentración	Positivo	Acelera la recuperación de tiempos de recarga de habilidades.	Una bombilla encendida sobre la cabeza.
Reconstrucción	Positivo	Regeneración pasiva de salud (exclusiva del Golem u objetos especiales).	Rocas volviendo a pegarse al cuerpo.

5. Progresión D&D: Tirada de Dados y Tienda del Tabernero

Al derrotar a un jefe de sala, se activa la fase de progresión inspirada en los juegos de rol de mesa (D&D):

5.1. Mecánica de Tirada de Dados

- **Dado de Atributo (d6):** Selecciona la estadística a mejorar: Fuerza, Defensa, Velocidad, Cooldowns, Maná o Salud Máxima.
- **Dado de Potencia / Magnitud (d4):** Otorga el valor de la mejora (+1, +2, +3 o +4).
- **Mecánica del "Dado del Caos / Riesgo":** El jugador puede renunciar a la tirada segura para lanzar un dado especial con oportunidad de obtener +5 en atributo a cambio del riesgo de sufrir penalizaciones (-1 o -2).

5.2. Tienda del Tabernero y Economía

El oro acumulado durante el combate se utiliza entre salas en la Tienda del Tabernero para adquirir consumibles de efecto temporal, amuletos pasivos o "dados trucados" para asegurar mejores atributos en la siguiente tirada.

6. Arquitectura Técnica y Código C++ Base (UE5 & Antigravity CLI)

La base de software del proyecto sigue las reglas de red Listen Server de Unreal Engine 5. A continuación se resumen las clases C++ principales implementadas:

6.1. Resumen de Clases C++ del Proyecto

- **APansDungeonsCharacter:** Personaje base del jugador. Incluye salud replicada (Health, MaxHealth), métodos de movimiento (DoMove, DoLook, DoJumpStart, DoJumpEnd) y registro en red mediante DOREPLIFETIME.
- **APansDungeonsBoss:** Clase base del jefe. Implementa salud replicada, sobrescritura de TakeDamage, eventos RPC NetMulticast (Multicast_OnAttackTelegraphed, Multicast_OnAttackImpact) y el método StartTelegraphedAttackAtActor.
- **APansDungeonsAIController & BT_Boss:** Controlador de IA que ejecuta el Behavior Tree (BT_Boss) y Blackboard (BB_Boss). Utiliza el servicio BTS_FindPlayer y la tarea BTT_BossAttack.
- **WBP_PlayerHUD:** Interfaz gráfica del jugador con barra de salud del jugador y barra superior del jefe (BossProgressBar) vinculada dinámicamente mediante GetHealthPercent e IsDead.

REGLA DE RED MULTIJUGADOR

Aislamiento Multijugador: La creación e instanciación de WBP_PlayerHUD en BP_ThirdPersonCharacter está filtrada por el nodo Is Locally Controlled. Esto evita que los clientes dupliquen o visualicen las barras

de interfaz de otros jugadores.

7. Roadmap Priorizado de Producción (Plan de Trabajo Paso a Paso)

Plan de trabajo estructurado por fases técnicas para el desarrollo continuo con AntigraVity CLI:

Fase 1: Núcleo e Infraestructura Base (Estado: COMPLETADO)

Módulo C++ PansDungeons, sincronización Listen Server, personaje APansDungeonsCharacter, IA de persecución en NavMesh, ataques telegrafados, HUD multijugador y físicas Ragdoll con verificación IsValid.

Fase 2: Combate Real del Jugador y Clases (Estado: SIGUIENTE PRIORIDAD)

- **Ataque Cuerpo a Cuerpo Real del Jugador:** Sustituir la tecla provisional F por Enhanced Input (IA_Attack) e implementar trazas de colisión (SphereTraceSingle) en C++ validadas en el servidor.
- **Sistema de Movilidad Táctico:** Implementar Dash / Esquive rodando con invulnerabilidad temporal (I-Frames).
- **Estructura de Clases C++:** Crear UPROPERTY replicadas de Maná/Energía y derivadas de clases para Paladín, Pícaro, Mago y Golem.

Fase 3: Componente de Estados Alterados

- **UCombatStatusComponent en C++:** Gestión de la matriz de efectos (Sangrado, Veneno, Lentitud, Alteración) con daño periódico (DoT) en el servidor.

Fase 4: Progresión de Datos D&D y Tienda

- **Sistema de Datos Virtuales:** Estructuras C++ / Blueprints para la tirada de dados D&D, dados de riesgo y menú de la Tienda del Tabernero.

Fase 5: Mallas en Blender, Audio y Pulido

- **Arte 3D y Sonido:** Modelado de mallas base en Blender, kit modular de escenarios, accesorios de cocina, efectos de sonido de caricatura y animación de impacto.