

Game Design Document

POR

SOFÍA ALEJANDRA SALAS AQUINO, DAVID SANTIAGO CÁRDENAS RIVERA,
ALEJANDRO RODRÍGUEZ QUINTERO

UNIVERSIDAD PONTIFICIA BOLIVARIANA
SECCIONAL BUCARAMANGA, SANTANDER

2025

Contents

1 Contextualización	1
1.1 Lluvia de ideas	1
1.2 Idea Final	2
2 Investigación y análisis	3
2.1 Juegos similares	3
2.1.1 Super Mario Bros (1985)	3
2.1.2 Crash Bandicoot	5
2.1.3 Cuphead (2017)	6
2.2 Público Objetivo	8
2.3 Viabilidad técnica del Proyecto	8
3 Definición del juego	9
3.1 Título del juego	9
3.2 Objetivo	9
3.3 Argumento principal y género del juego	9
3.3.1 Argumento principal	9
3.3.2 Género del juego	9
3.4 Personajes	9
3.4.1 Principal	9
3.4.2 Secundarios	10
4 Diseño del juego	11
4.1 Estilo visual	11
4.2 Mecánicas del juego y controles	12
4.3 Interfaces de usuario	13
5 Plan técnico del proyecto	16
5.1 Marco tecnológico	16
5.1.1 Motor de Juego	16
5.1.2 Lenguaje de Programación	16
5.1.3 Diseño y Arte	16
5.1.4 Audio	16
5.1.5 Gestión de Datos y Persistencia	16
5.1.6 Control de Versiones y Gestión del Proyecto	17
5.2 Cronograma de desarrollo	17
5.3 Metodología y Roles del equipo de trabajo	17

Chapter 1

Contextualización

1.1 Lluvia de ideas

Inspiración y arte	Mecánicas	Personajes e historia
Película Flow 2D pixel art	PowerUps para aguantar respiración Transformación en otro animal Barcos dentro de la Ballena para el escape Traje acuático para el gato Sistema de logros Si se muere 9 veces, se pierde lo que se guarda Timer de oxígeno cuando no tiene traje acuático En el mapa, el agua sube desde abajo El gato para atacar aruña, pero puede invocar pirañas también con cierta cantidad de cargas. “Ultimate: Vidas pasadas” a medida que pierdes vidas, gana más fuerza. Existiría un contador según la cantidad de vidas perdidas, para así saber cual sería el escalado de daño. Con el traje de buceo tiene 2 vidas. Si le pegan al gato, pierde una “vida” y si le pegan sin el buceo se muere (dentro del nivel) Modo tormenta como modo difícil 3,4 niveles + boss	Personaje principal: Gato Boss final es una Ballena Gato cae por alcantarilla y termina en el mar Cinemática final: el gato volviendo a su casa y se duerme Rana que salta entre arboles y come insectos como power ups

Figure 1.1: Lluvia de ideas realizada el día 03 de febrero

1.2 Idea Final

Para la definición de la idea final, el equipo se reunió y filtró las ideas, obteniendo las características contempladas en la tabla 1.2. La idea final se define como un juego de plataforma 2D pixel art de vista vertical con tres niveles jugables, donde el primero y el segundo consisten en recorrer el nivel de izquierda a derecha exitosamente, y el tercero se basa en derrotar a un jefe. El personaje principal es un gato, y el jefe final es una ballena.

Inspiración y arte	Mecánicas	Personajes e historia
Película Flow 2D pixel art	<p>PowerUps para aguantar respiración (con timer) Traje acuático para el gato. de traje acuatico después de caerse a la alcantarilla que da al mar.</p> <p>PRIMER NIVEL: alcantarilla. No hay traje, el traje lo encuentra al final del primer nivel. Al final del primer nivel, accede al mundo acuatico. Ratas</p> <p>SEGUNDO NIVEL: acuático.</p> <p>Tercer nivel: boss. Pez linterna</p> <p>AMBIENTACION DEL BOSS FINAL luces y oscuridad. Océano abisal</p> <p>En los niveles, hay oxígeno para recargar el tanque del traje (Intenta meterse sin el pero como es gato no le gusta y se pone el traje y entra).</p> <p>). Este tanque tiene un timer. Si no renueva el timer consiguiendo oxígeno, se muere.</p> <p>Barcos dentro de la Ballena para el escape</p> <p>“Ultimate: Vidas pasadas” a medida que pierdes vidas, gana más fuerza. Existiría un contador según la cantidad de vidas perdidas, para así saber cual sería el escalado de daño. Si se muere 7 veces, se pierde lo que se guarda.</p> <p>El gato para atacar aruña, y por cada aruñado carga ulti.</p> <p>Con el traje de buceo tiene 2 vidas. Si le pegan al gato, pierde una “vida” y si le pegan sin el buceo se muere (dentro del nivel)</p>	<p>Personaje principal: Gato</p> <p>Boss final es una Ballena</p> <p>Gato cae por alcantarilla y termina en el mar</p> <p>Cinemática final: el gato volviendo a su casa y se duerme.</p>

Figure 1.2: Lluvia de ideas final

Chapter 2

Investigación y análisis

2.1 Juegos similares

2.1.1 Super Mario Bros (1985)

El *gameplay* de Super Mario Bros consiste en mover a Mario, el personaje que controla el jugador, a lo largo de un nivel 2D, el cual presenta una perspectiva lateral.

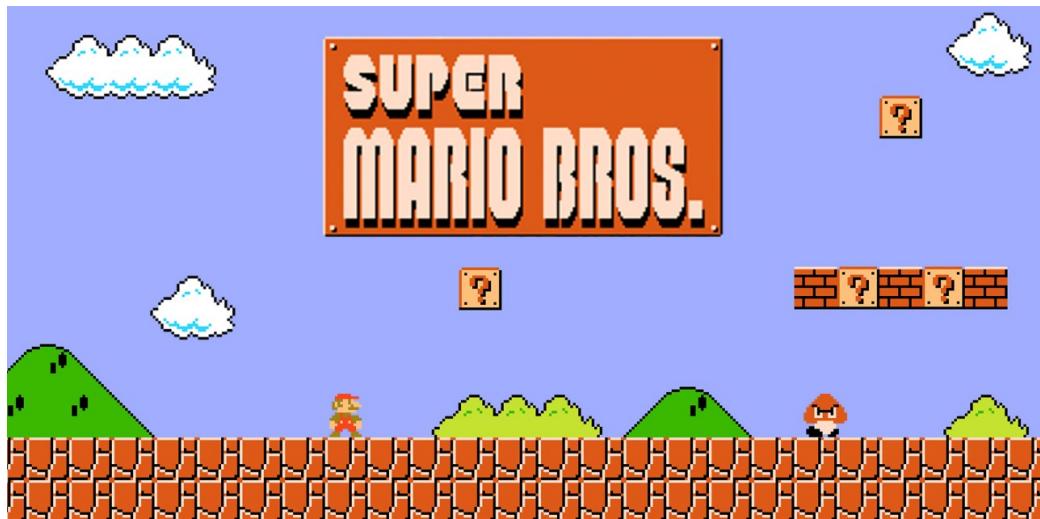


Figure 2.1: Super Mario Bros. Tomado de Nintendo.

Mario puede correr y caminar para la izquierda y derecha, saltar, y dependiendo de las bonificaciones que posea el personaje, puede lanzar bolas de fuego. La gravedad es un aspecto relevante en el juego, volviendo necesario saltar sobre los posibles huecos que se encuentren en el mapa.

Mario se puede encontrar en tres estados: pequeño (al inicio del juego), grande (se obtiene por medio de una bonificación identificada como un "champiñón"), y fuego (puede lanzar bolas de fuego, es obtenido por medio de una bonificación).[1] A continuación, se enseñan los *sprites* de estos tres estados:

2.1. JUEGOS SIMILARES

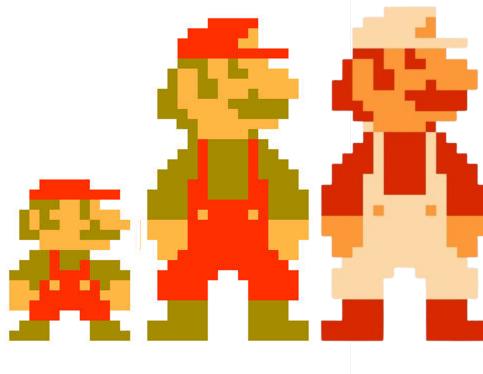


Figure 2.2: Sprites de los 3 estados de Mario

Ahora, el objetivo principal de cada nivel de Mario es completarlo hasta el final, lo que implica recorrerlo de izquierda a derecha en su totalidad. Aparte de esto, existen logros auxiliares:

- Recoger todas las monedas esparcidas a lo largo del mapa.
- Completar el nivel lo más rápido posible.
- Conseguir el puntaje más alto, el cual es calculado a través de la cantidad de enemigos matados y monedas recolectadas.

Si Mario es tocado por un enemigo y está en estado pequeño, pierde una vida. En cambio, si está en estado grande y un enemigo lo toca, pasa a estado pequeño, y si está en estado de fuego y es tocado, pasará a estado grande.

Por otro lado, si logra saltar encima de la cabeza de un enemigo, podrán ocurrir varios casos de uso: la mayoría de los enemigos (por ejemplo, goombas, bolas de fuego) mueren ante este ataque; otros (por ejemplo, piraña) no son vulnerables a esto y proceden a atacar a Mario; finalmente, las tortugas se retraen en sus caparazones si se les salta sobre éste, y estos caparazones pueden ser recogidos por Mario y lanzados a otros enemigos para matarlos.

Ciertos objetos se encuentran por los niveles, ya sea al aire libre o escondidos dentro de bloques de ladrillo y sólo aparecen cuando Mario salta contra estos bloques desde abajo de modo que golpea su cabeza contra ellos. Los objetos disponibles incluyen monedas, las cuales pueden ser recolectadas para puntaje y para vidas extra (cada 100 monedas), hongos que hacen que Mario crezca a Grande si actualmente es pequeño, y flores que hacen que Mario se convierta en el estado de Fuego si ya es Grande.

Super Mario Bros causó impacto a nivel mundial, ya que a mediados de los años 80, los videojuegos para consolas estaban en declive. Se creía que las consolas estaban a punto de extinguirse por completo, hasta que en 1985 se lanzó *Super Mario Bros*. Este juego logró cambiar el curso del ?crash de los videojuegos? y se convirtió en uno de los videojuegos más vendidos de la historia, dando inicio a la edad de oro de los juegos de plataformas[2] [3].

2.1.2 Crash Bandicoot

Una franquicia de videojuegos sacada inicialmente en el año 1996, desarrollado por la empresa Naughty Dog. Se trata de Crash, el personaje principal, quien es un bandicut y cuya historia se centra en luchar contra su enemigo, y creador, el malvado Neo Cortex en el escenario ficticio de las Islas Wumpa [4].



Figure 2.3: Crash Bandicoot de 1996. Imagen tomada de [5]

Es un juego de plataformas en 3D en el que, según Zach Meston [6], el jugador simplemente corre hacia la pantalla, a diferencia de otros juegos como Mario, cuyo movimiento es de izquierda a derecha.

Tiene una mecánica basada en niveles, en donde el objetivo de cada uno de estos es guiar a Crash desde el inicio hasta el final del mapa, en el cual este requerirá hacer uso de elementos especiales como motocicletas, submarinos, animales salvajes, entre otros.

Para el apartado de sus movimientos, en el Crash Bandicoot de 1996, este podía realizar acciones como saltar, dar vueltas y correr. Más adelante, en nuevas entregas como la de Coretex Strikes Back, es introducido al jugador nuevos movimientos tales como un salto alto, salto alto con voltereta, ataques laterales y golpe de cuerpo. Uno de estos movimientos se puede observar en la figura 2.6.



Figure 2.4: Crash Bandicoot realizando un golpe de cuerpo o *body slam*. Imagen tomada de [7]

Dentro de otras mecánicas cabe destacar que cuenta con objetos para colecciónar. Entre algunos de estos está las Frutas Wumpa, gemas, cristales, reliquias y máscaras Aku Aku. Todos estos ítems se

2.1. JUEGOS SIMILARES

pueden conseguir dentro de cajas o *crates*, siendo las frutas el colecciónable más común. A continuación, se describirá la especialidad de cada elemento [6].

- **Frutas Wumpa:** Al recolectar 100 de estas, se le otorga una vida extra al jugador.
- **Máscaras Aku Aku:** Protegen a Crash del daño proveniente de enemigos y obstáculos. Pueden recolectarse hasta un máximo de dos, con una tercera otorgándole invencibilidad temporal.
- **Gemas:** Se pueden obtener si el jugador rompe todas las cajas en un nivel, así como se pueden obtener otras adicionales como secretos en lugares escondidos o realizando ciertas tareas.
- **Cristales:** Estos son requeridos para poder progresar dentro de los juegos de plataforma de la saga.
- **Reliquias:** Estas se otorgan en modos de tiempo a prueba.

El juego parte con un mapa que contiene distintos niveles y algunas zonas secretas. Para progresar, el jugador, dentro de cada uno de los niveles, debe conseguir un cristal para cada una de las etapas donde se juega en un orden cualquiera y contienen un jefe al que derrotar.

Asimismo, la música de este juego fue desarrollada por Josh Mancell, Steve Duckworth, Andy Blathe, entre otros autores que han venido variando a medida que salió cada entrega de la serie [8].

2.1.3 Cuphead (2017)

Cuphead es un videojuego de disparos y plataformas desarrollado por Studio MDHR y lanzado en 2017. Se distingue por su estética inspirada en las caricaturas de los años 30 y su desafiante jugabilidad basada en patrones de ataque y memoria muscular [9].



Figure 2.5: *Cuphead* y *Mugman* deberán luchar a través de múltiples niveles que culminan en una pelea con un jefe de cada zona para pagar su deuda con el diablo. [10]

2.1. JUEGOS SIMILARES

En Cuphead, el jugador debe enfrentarse a diversos jefes, cada uno con patrones de ataque claramente telegráficos. A medida que el combate avanza, los jefes cambian de fase, modificando sus ataques y dificultando la pelea. La dificultad del juego radica en aprender y reaccionar rápidamente a estos patrones mientras se mantiene la ofensiva.

Cada jefe cuenta con varias fases que alteran su apariencia y patrones de ataque. Un ejemplo es el Capitán Brineybeard, quien empieza en su barco atacando con proyectiles simples, pero conforme avanza la pelea, invoca aliados y finalmente su barco cobra vida, exigiendo nuevas estrategias de esquiva y ataque.

Cuphead ofrece dos tipos de niveles principales:

- **Run and Gun:** El jugador avanza de un punto A a un punto B mientras enfrenta enemigos y recolecta monedas para comprar mejoras.
- **Boss Battles:** Niveles enfocados en la lucha contra jefes, identificados en el mapa por ligeras animaciones en el entorno.

El juego también incluye un mundo hub u overworld similar a *Super Mario World*, donde los jugadores pueden moverse libremente, hablar con NPCs, acceder a niveles y comprar mejoras.

Un aspecto clave del combate es la mecánica de **parry**, que permite bloquear ataques de color rosa para cargar la barra de habilidad especial. Esto da acceso a **EX moves**, ataques mejorados de las armas equipadas, y **Super moves**, poderosas habilidades desbloqueadas en los niveles de mausoleo.

Cuphead es un juego desafiante pero justo, donde la destreza del jugador define el progreso. Su estética, combinada con una jugabilidad exigente y bien equilibrada, lo convierten en una experiencia única dentro del género de acción y plataformas.



Figure 2.6: Cuphead fue inspirado por el estilo de animación *rubber hose* usada en las caricaturas de los años 30. [11]

2.2. PÚBLICO OBJETIVO

2.2.1. Público Objetivo

El juego va enfocado a niños de 6 a 13 años, contando con controles intuitivos y una historia fácil de procesar. Los jugadores son casuales, ya que no se requiere experiencia en anteriores juegos de plataforma para poder superar los niveles.

2.3. Viabilidad técnica del Proyecto

Para la realización del proyecto, se requiere equipos con 16 GB de RAM como mínimo para el correcto funcionamiento de Unity. Los miembros del equipo cuentan con este hardware disponible. Unity es líder en el desarrollo de videojuegos [12]. Dos miembros del equipo de trabajo tienen previa experiencia en el desarrollo de videojuegos 2D pixelart con la herramienta GameMaker Studio. Sin embargo, se planteó un proyecto de corto alcance, contando con tres niveles y un solo personaje principal.

A pesar de la variedad de juegos de plataforma en el mercado, se busca diferenciar el proyecto por medio de su mecánica de la habilidad Ultimate titulada "Vidas pasadas", la cual será explicada en la sección 4 del documento.

Chapter 3

Definición del juego

3.1 Título del juego

El título del juego es "*Anubis' Adventure*".

3.2 Objetivo

Ayudar a Anubis, un gato común quien es el protagonista principal de la historia, a volver a su hogar, donde tendrá que atravesar por distintos escenarios o niveles cruzando por obstáculos, peleando con enemigos y adquiriendo objetos especiales que le permitirán progresar en el juego.

3.3 Argumento principal y género del juego

3.3.1 Argumento principal

La historia principal se centra en un gato llamado Anubis, quien un día iba trepando por los techos de las casas del barrio donde vive, hasta que al cruzar por un tejado inestable, este se desploma, ocasionando que se caiga por un hueco que conduce a una alcantarilla. Es ahí que Anubis tiene como objetivo retornar a su hogar. Él deberá cruzar por lugares como la, dicha ya, alcantarilla, el mar y hasta busear por las profundidades del océano abisal. Al completar los niveles, Anubis se embarca en un bote que lo conduce hasta la orilla, donde se va caminando a su morada y finaliza la historia echándose una siesta después de un largo día.

3.3.2 Género del juego

Es principalmente un juego de plataformas que contiene elementos de otros géneros como acción, debido a que el personaje principal debe luchar con diversos enemigos y un jefe final.

3.4 Personajes

3.4.1 Principal

El personaje principal es Anubis, un gato común y corriente.

3.4.2 Secundarios

Entre los personajes secundarios se encuentran:

- **Squeakies:** Unas pequeñas ratas, enemigas de Anubis, las cuales se encuentran en la alcantarilla.
- **Bitees:** Pirañas ubicadas en el mar y la zona abisal, también enemigas de Anubis.
- **Cete:** Una ballena situada en la zona abisal del océano, quien es la jefa final de la historia.

Chapter 4

Diseño del juego

4.1 Estilo visual

Se optó por un estilo visual pixelart 2D, tomando como referencia juegos como Super Mario Bros de 1985.

El primer nivel del juego estará ambientado en un nivel de alcantarilla, contando con colores oscuros.

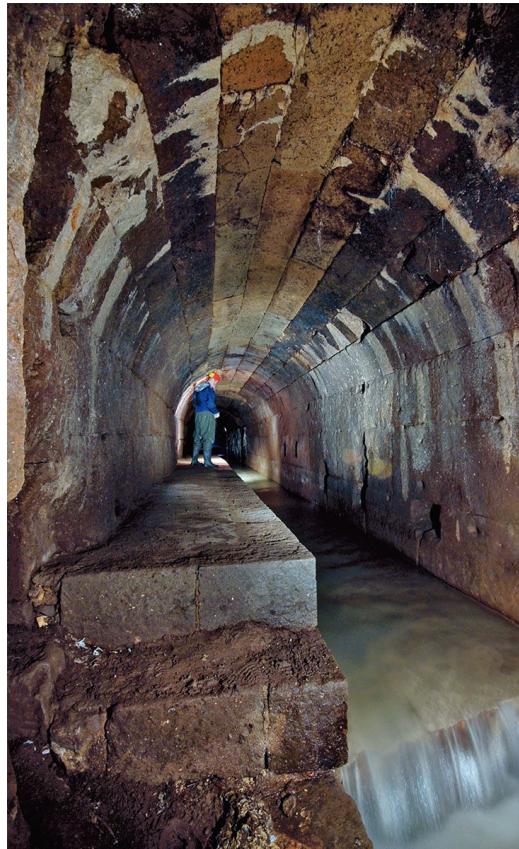


Figure 4.1: Inspiración para el primer nivel, ambientado en alcantarillas.

4.2. MECÁNICAS DEL JUEGO Y CONTROLES

El segundo nivel del juego está ambientado en el océano, específicamente en la zona de sol, o zona epipelágica.



Figure 4.2: Inspiración para el segundo nivel, ambientado en el mar.

Por último, el tercer nivel estará ambientado en el océano abisal.



Figure 4.3: Inspiración para el tercer nivel, ambientado en el océano abisal.

4.2 Mecánicas del juego y controles

Para el movimiento del personaje principal, se usa la combinación "AD", siendo "A" para el movimiento horizontal a la izquierda, "D" para el movimiento horizontal a la derecha, la tecla espacio para el salto, click derecho para ataque y la tecla "e" para activar la ultimate.

El personaje principal cuenta con dos tipos de ataque: básico y la ultimate, llamada "Vidas pasadas". La ultimate tiene una mecánica de "cargado", en donde por cada básico acertado se carga un 25% de la ultimate. La Ultimate consiste en que, dependiendo de la cantidad de muertes que el personaje lleve acumulado a lo largo de los niveles, la Ultimate escalará en daño por cada muerte. Cada muerte cuenta como un "golpe", teniendo un total de 7 vidas acumulables. Es decir, si el personaje ha muerto 9 veces y usa la ultimate, esta dará nueve golpes consecutivos. Una vez el personaje principal supera las 7 muertes, se pierde el progreso del juego.

4.3. INTERFACES DE USUARIO

A partir del segundo nivel del juego, al estar ambientado en el océano, el personaje estará equipado con un traje acuático encontrado en el nivel (1). En los próximos dos niveles, habrá oxígeno como un ítem esparcido por el mapa, el cual el personaje deberá recolectar en un plazo de 30 segundos, para así recargar su tanque. En el caso de no recargar el tanque a tiempo, el gato muere.

El gato cuenta con "20" de HP inicialmente, en donde cada golpe de un enemigo representa "10" de HP. Si el gato es agredido dos veces, morirá.

4.3 Interfaces de usuario

Se plantea el diseño del personaje principal Anubis en la figura 4.4, donde se incluye una versión del personaje con el traje acuático.

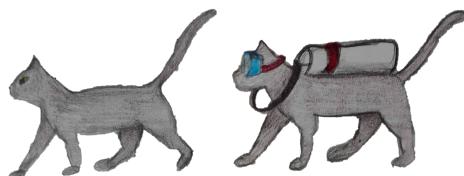


Figure 4.4: Personaje Anubis en sus dos estados.

Por otro lado, se propone para el primer, segundo y tercer nivel los diseños de las figuras 4.5,4.6,4.7, respectivamente



Figure 4.5: Diseño propuesto para el primer nivel del juego.

4.3. INTERFACES DE USUARIO

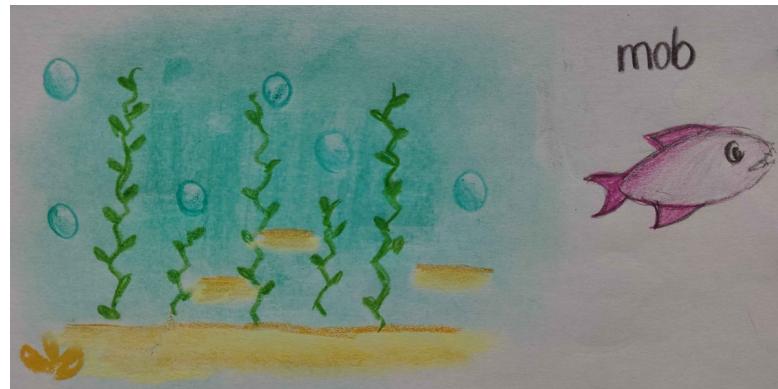


Figure 4.6: Diseño propuesto para el segundo nivel del juego.

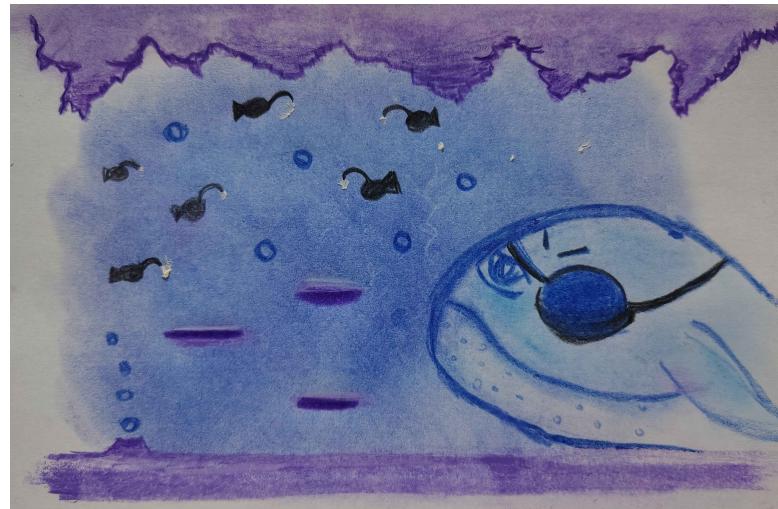


Figure 4.7: Diseño propuesto para el tercer nivel del juego.

Para las interfaces de usuario, se definió la estructura inicial del menú de inicio y el menú de pausa en las figuras 4.8, 4.9 respectivamente.



Figure 4.8: Estructura de menú inicial propuesto.

4.3. INTERFACES DE USUARIO

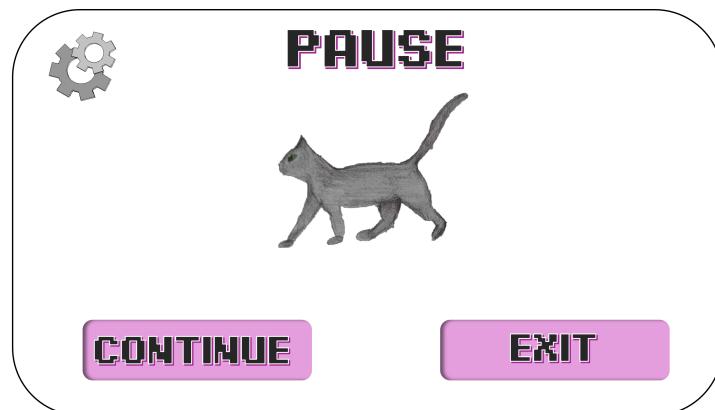


Figure 4.9: Estructura menú de pausa propuesto.

Chapter 5

Plan técnico del proyecto

5.1 Marco tecnológico

5.1.1 Motor de Juego

- **Unity:** Plataforma robusta con herramientas optimizadas para el desarrollo de videojuegos 2D, incluyendo el sistema de físicas, animaciones y herramientas de scripting en C#. Unity cuenta con un sistema de físicas para el manejo de colisiones, gravedad y comportamiento de plataformas y personajes.

5.1.2 Lenguaje de Programación

- **C#:** Lenguaje principal para la lógica del juego en Unity. Permite programación orientada a objetos y optimización eficiente.

5.1.3 Diseño y Arte

- **Pixilart:** Editor de dibujo gratuito y plataforma social para el intercambio de diseño pixel art.
- **Aseprite:** Software especializado en la creación de sprites y animaciones en pixel art.
- **Spine:** Herramienta para animaciones esqueléticas en 2D, permitiendo movimientos fluidos y reutilización de assets.
- **Tiled:** Editor para diseño de niveles con tilesets.

5.1.4 Audio

- **FMOD:** Middleware para la gestión avanzada de sonido, permitiendo efectos dinámicos y adaptación de audio a eventos del juego.
- **Audacity:** Para edición y limpieza de efectos de sonido y música.

5.1.5 Gestión de Datos y Persistencia

- **PlayerPrefs:** Se usa para almacenar configuraciones básicas y preferencias del usuario, es un sistema integrado en Unity.
- **SQLite:** Para la gestión de datos en local, permitiendo guardar progresos del jugador.

5.2. CRONOGRAMA DE DESARROLLO

5.1.6 Control de Versiones y Gestión del Proyecto

- **Git + GitHub:** Sistema de control de versiones para la gestión de cambios en el código y assets.
- **Jira:** Plataforma de gestión de proyectos basada en metodologías ágiles.

5.2 Cronograma de desarrollo

En el documento adjunto *project_crono.xlsx* se encuentra el cronograma de ejecución de proyecto.

5.3 Metodología y Roles del equipo de trabajo

El desarrollo del videojuego seguirá el marco de trabajo **Scrum**, utilizando una planificación iterativa e incremental con sprints definidos en el cronograma. Se realizarán reuniones de control los **lunes y jueves**, en las cuales el equipo presentará avances y resolverá dudas con el docente.

Roles del equipo

- **Product Owner (PO):** David Santiago Cárdenas Rivera
- **Scrum Master:** Sofía Alejandra Salas Aquino
- **Equipo de Desarrollo:** Alejandro Rodríguez Quintero, Sofía Alejandra Salas Aquino, David Santiago Cárdenas Rivera

Flujo de trabajo

1. Sprint Planning (Inicio de cada sprint)

- Se seleccionan las tareas prioritarias del Jira para el sprint en curso.
- Se definen las metas específicas a cumplir.

2. Desarrollo del Sprint (Iteraciones semanales)

- El equipo trabaja en las funcionalidades acordadas.
- Se realizan reuniones de control los lunes y jueves para revisar avances y comentar problemas.

3. Sprint Review (Fin de cada sprint)

- Se presenta el estado del videojuego con las funcionalidades implementadas.
- Se obtiene feedback para iteraciones futuras.

4. Sprint Retrospective

- Se identifican mejoras en el proceso de trabajo.
- Se registran aprendizajes y ajustes para el siguiente sprint.

5.3. METODOLOGÍA Y ROLES DEL EQUIPO DE TRABAJO

Ciclo de Sprints

- **Sprint 1 (Semana 7):** Prototipo inicial con menú principal, pantalla de pausa y primer nivel jugable.
- **Sprint 2 (Semana 11):** Agregación de interfaz de usuario, HUD del jugador y segundo nivel jugable.
- **Sprint 3 (Semana 15):** Implementación de jefe final, música, efectos de sonido y refinamiento final.

Bibliography

- [1] J. Togelius, S. Karakovskiy, J. Koutnik **and** J. Schmidhuber, ?Super mario evolution,? in *2009 IEEE Symposium on Computational Intelligence and Games* 2009, pages 156–161. DOI: 10.1109/CIG.2009.5286481.
- [2] T. Minkkinen, ?Basics of Platform Games,? 2016.
- [3] S. L. Kent, *The Ultimate History of Video Games: From Pong to Pokémon and Beyond*. Three Rivers Press, 2001.
- [4] H. Goldberg, *All Your Base Are Belong to Us: How Fifty Years of Videogames Conquered Pop Culture*. Three Rivers Press, 2003, Incluye análisis histórico y contextual de grandes franquicias, entre ellas Crash Bandicoot.
- [5] Press Start, *Remembering Crash Bandicoot (1996) – The Original PlayStation Mascot*, <https://press-start.com/features/2016/05/10/remembering-crash-bandicoot-1996-original-playstation-mascot/>, Accedido: 14 de febrero de 2025, 2016.
- [6] Z. Meston, *Crash Bandicoot: The Official Strategy Guide*. Dimension Publishing, 1996, Guía oficial que detalla mecánicas, niveles y aspectos jugables del juego original.
- [7] Crash Bandicoot Wiki contributors, *Body Slam*, https://crashbandicoot.fandom.com/wiki/Body_Slam, Accessed: 14 February 2025, n.d.
- [8] C. Moriarty, ?Rising to Greatness: The History of Naughty Dog,? *IGN*, 2013, Artículo de referencia sobre la evolución del estudio y la creación de franquicias como Crash Bandicoot. url: <https://www.ign.com/articles/2013/10/04/rising-to-greatness-the-history-of-naughty-dog>.
- [9] C. Wiki. ?Cuphead (video game).? Consultado el 14 de febrero de 2025. (2025), url: [https://cuphead.fandom.com/wiki/Cuphead_\(video_game\)](https://cuphead.fandom.com/wiki/Cuphead_(video_game)).
- [10] Nerfeados. ?Cuphead - Reseña (PC) - Nerfeados.? Consultado el 14 de febrero de 2025. (2025), url: <https://nerfeados.com/cuphead-resena-pc-nerfeados/>.
- [11] Pinterest. ?Rubber Hose - Tablero de Ohdad en Pinterest.? Consultado el 14 de febrero de 2025. (2025), url: <https://co.pinterest.com/ohdad/rubber-hose/>.
- [12] E. Castaneda. ?Impulsando la Magia del Entretenimiento: Qué es un Motor de Videojuego y los Principales Exponentes en el Mercado.? Consultado el 13 de febrero de 2025. (2024), url: <https://www.argosoftgroup.com/2024/01/04/impulsando-la-magia-del-entretenimiento-que-es-un-motor-de-videojuego-y-los-principales-exponentes-en-el-mercado/>.