

Benemérita Universidad Autónoma de Puebla



Escuela de Artes Plásticas y Audiovisuales

“Diseño, desarrollo y exposición de videojuego Destroylt”

MAYO 2023

Tesina presentada para obtener el título en

Lic. En Arte Digital

Presenta: Verónica González Rivera

Matricula: 201451821

Director y Asesor de Tesina: Abner Quiroz Clemente

RESUMEN

Este documento aborda los procesos utilizados para crear un videojuego de plataformas 2D. Se detalla la documentación necesaria para su desarrollo, los procesos de programación y los procesos artísticos como la animación, diseño de personajes, escenarios e ítems. El objetivo del proyecto era crear un juego divertido y competitivo basado en una única mecánica principal que diera forma al juego, aprovechando todos los conocimientos obtenidos en ilustración, diseño de juegos, programación y animación. El propósito final del proyecto era presentarse en un concurso y permitir que la gente lo probara.

*“CREO QUE LA FANTASÍA EN EL SENTIDO DE QUE LA IMAGINACIÓN ES MUY IMPORTANTE.
NO DEBEMOS CEÑIRNOS DEMASIADO A LA REALIDAD COTIDIANA, TENEMOS QUE DAR
CABIDA A LA REALIDAD DEL CORAZÓN, DE LA MENTE Y DE LA IMAGINACIÓN.”*

CONTENIDO

| | |
|---|----|
| Resumen..... | 2 |
| Agradecimientos | 7 |
| Introducción | 8 |
| Justificación | 8 |
| Planteamiento | 9 |
| Objetivos | 9 |
| Metodología | 10 |
| Antecedentes | 10 |
| Videojuegos de plataformas | 11 |
| Super Smash Bros para Nintendo 3DS y Wii U..... | 11 |
| Ultimate Chicken Horse..... | 11 |
| Marco teórico..... | 12 |
| Diseño de juego..... | 12 |
| Géneros de videojuegos..... | 13 |
| Sandbox | 13 |
| Estrategia en tiempo real (RTS)..... | 14 |
| Shooters | 14 |
| Multiplayer online battle arena (MOBA) | 14 |
| Juegos de Rol..... | 14 |
| Simulación y deportes | 15 |
| Puzzlers y party games | 15 |
| Acción y aventura..... | 15 |
| Supervivencia y horror | 16 |
| Plataformas | 16 |
| Mecánicas..... | 16 |
| Dinámicas | 17 |
| Estética | 17 |
| Diseño de niveles | 17 |
| Requerimientos funcionales | 17 |
| Usabilidad..... | 17 |
| Recompensas | 18 |
| Elementos históricos de arquitectura | 18 |
| Arreglos espaciales..... | 19 |

| | |
|--|----|
| Figure-Ground | 19 |
| Form-Void..... | 19 |
| Arrivals..... | 19 |
| Genius Loci | 20 |
| Estructuras del espacio de juego..... | 20 |
| Tipos de tamaño espacial..... | 21 |
| Planos de nivel..... | 21 |
| Herramientas de desarrollo | 22 |
| Motor de juegos | 22 |
| Software | 22 |
| Arte..... | 22 |
| Pixel art..... | 22 |
| Aliased | 23 |
| Anti-aliased..... | 23 |
| Paleta de colores | 24 |
| Monocromática | 25 |
| Análoga..... | 25 |
| Complementaria..... | 25 |
| Animación | 25 |
| Aplastar y estirar (squash and stretch) | 26 |
| Anticipar (anticipation) | 26 |
| Puesta en escena (staging)..... | 26 |
| Acción paso a paso y acción pose a pose (straight ahead action and pose to pose)..... | 26 |
| Acción de sobreimponer y acción de principio a fin (follow through and overlapping action) | 27 |
| Acción secundaria (secondary action)..... | 27 |
| Exageración (exaggeration)..... | 28 |
| Dibujo sólido (solid drawing)..... | 28 |
| Apariencia (appeal) | 29 |
| Animación en el motor de juegos Unity..... | 29 |
| Animación de partículas para videojuegos. | 29 |
| Desarrollo de proyecto..... | 30 |
| Diseño de juego..... | 30 |
| Mecánicas..... | 31 |
| Dinámicas | 32 |

| | |
|---|----|
| Diseño de nivel | 32 |
| Requerimientos funcionales | 32 |
| Usabilidad..... | 33 |
| Recompensas | 33 |
| Elementos históricos de arquitectura | 33 |
| Arquitectura griega | 33 |
| Arquitectura Romana..... | 34 |
| Arquitectura Medieval | 34 |
| Arreglos espaciales..... | 35 |
| Figure-Ground | 35 |
| Form-Void..... | 36 |
| Arrivals..... | 36 |
| Genius Loci | 36 |
| Estructura del espacio de juego | 37 |
| Maze | 37 |
| Tipos de tamaño espacial..... | 37 |
| Espacio estrecho | 37 |
| Espacio intimo | 38 |
| Espacio de prospección | 38 |
| Planos de nivel y mapas de juego | 38 |
| Programación | 42 |
| Arte..... | 46 |
| Diseño conceptual..... | 46 |
| Diseño de personajes y naves | 49 |
| Diseño de plataformas y elementos complementarios | 53 |
| Diseño de fondo | 56 |
| Animaciones y efectos | 57 |
| Resultados | 58 |
| Conclusión | 67 |
| Anexos | 68 |
| Índice de imágenes..... | 74 |
| Bibliografía | 75 |

AGRADECIMIENTOS

Agradezco a mi familia que siempre ha creído en mí, aunque yo no lo haga, a mis compañeros y profesores quienes me ayudaron a que este proyecto se hiciera realidad con sus comentarios y apoyo.

Agradezco a mi novio y su familia que me apoyaron durante el proceso, siempre estuvieron pendientes y me animaron a seguir adelante.

Quiero agradecerles especialmente a tres personas, mi mamá quien siempre me ha apoyado en cada decisión y ha sido paciente conmigo, aunque no sepa bien por donde estoy caminando. Maguito quien me apoyo a lo largo de la carrera y siempre se vio interesada y orgullosa de todo lo que hacía un agradecimiento al cielo. Mi profesor y amigo Abner que siempre ha estado apoyándome y dándome consejos para mejorar como persona, quien siempre creyó en mí y fue paciente conmigo.

INTRODUCCIÓN

La motivación detrás del desarrollo de este proyecto fue realizar un juego enfocado en mecánicas, dejando de lado una historia compleja debido a la necesidad de probar que los juegos que carecen de un arco argumental complejo son igual de desafiantes y divertidos, tanto para los jugadores como para los desarrolladores.

El juego fue diseñado para ser divertido y casual, para que fuera jugado en sesiones cortas y explotar el lado competitivo de los jugadores. El nivel del juego se diseñó para explorar una mecánica poco usada en juegos de plataformas de la cual se hablará más adelante en el documento.

En este documento, podremos ver el desarrollo y metodología necesaria para el desarrollo de un videojuego de plataformas 2D multijugador, centrado en una mecánica básica: agarrar y aventar, es un juego casual para todo tipo de audiencia. Conoceremos los procesos que se llevaron a cabo para su realización, el diseño de juego, animaciones y programación y cómo este proceso se apoyó en lo aprendido en las diferentes materias impartidas en ARPA.

JUSTIFICACIÓN

El objetivo de este proyecto fue aplicar los conocimientos teóricos adquiridos a través en la carrera de Arte Digital, y traspasarlos a un ámbito práctico, usando los conocimientos de programación, diseño de juegos, creación de props¹, diseño de personajes, animación, diseño de niveles, entre otros. En el proceso de desarrollo de videojuegos, estas habilidades desarrolladas dentro del plan de estudios convergen de forma sinérgica y contribuyen significativamente a mejora y enriquecer la entrega de esta tesina.

La creación de un producto tangible en forma de videojuego se convirtió en la culminación de cinco años de estudio y es una muestra concreta de la adquisición y aplicación efectiva de dicho conjunto de conocimientos. Este proyecto permite demostrar la capacidad de traducir la formación académica en un contexto real, validando así la formación en Arte Digital a través de la creación de una obra interactiva que represente una síntesis de los aprendizajes obtenidos.

La realización de este videojuego, como parte de mi proyecto de tesis, también se fundamenta en una serie de objetivos concretos y metas profesionales. En primer la creación de este juego representa una oportunidad para enriquecer y diversificar mi portafolio profesional. La industria de los videojuegos competitiva, y contar con un videojuego en mi portafolio no solo demuestra mis habilidades técnicas y

¹ En el ámbito de los videojuegos, el término "prop" (derivado de "properties", que se traduce como "objetos" o "aditamentos"). Estos objetos pueden integrarse en la ambientación del juego o desempeñarse como elementos que los personajes pueden recoger, manipular o emplear en el desarrollo de la jugabilidad.

creativas. “Destrylt” me brindó la posibilidad de explorar en profundidad áreas específicas que me interesan dentro de mi formación como artista digital.

La creación del videojuego “Destrylt” como parte de mi tesis no solo es una oportunidad para expandir mi portafolio y consolidar mis conocimientos en áreas las áreas antes mencionadas, sino que también representa un paso crucial en mi desarrollo profesional al alinear mis intereses y habilidades con las demandas del mercado laboral en el campo del arte digital y los videojuegos.

PLANTEAMIENTO

Desde su aparición los juegos han logrado crear experiencias en sus usuarios, alegría, miedo, intriga, diversión, son algunas de ellas. Este proyecto se enfoca en el desarrollo de un juego centrándose en las mecánicas y jugabilidad para el desarrollo de un videojuego, plantea una historia o premisa simple que fue desarrollada para funcionar correctamente con las mecánicas. Los juegos de plataformas han estado en el mercado desde 1980, hay bastantes mecánicas y dinámicas interesantes pueden ser exploradas, sorprender y divertir a los jugadores. “Destry It” es multijugador que intenta explotar el sentimiento competitivo de los jugadores para hacer el juego más divertido. Tiene una mecánica diferente, cuenta dos planos en un ambiente 2D, en los que los jugadores podrán saltar durante el juego, con esto se espera que el juego sea entretenido y retador. Con este proyecto se utilizaron las habilidades de programación y diseño de juegos que se han adquirido a lo largo de la carrera

OBJETIVOS

El objetivo del proyecto fue desarrollar un videojuego multijugador de plataformas, que explorara mecánicas poco utilizadas en otros juegos de su género. Debía ser divertido para los usuarios y se buscó que la experiencia fuera dinámica y cambiara con cada partida.

Entre los objetivos específicos del proyecto se realizaron los siguientes puntos.

- Humor: Se realizó para que sacara risas entre los jugadores y espectadores ya sea por las mecánicas o apariencia visual del juego.
- Re-jugabilidad: Se buscó que la experiencia del juego cambie cada partida. Que los jugadores a pesar de jugar en el mismo nivel tengan una experiencia de juego diferente cada partida.
- Juego juguete: Se logró que el juego sea divertido, aunque el jugador no gane la partida.
- Arte: Se generó un diseño de niveles y personajes acorde al objetivo principal del juego, el cual es hacer un juego divertido y casual.

- Exposición: Uno de los objetivos fue exponer el juego en un evento académico internacional.

METODOLOGÍA

En el desarrollo del juego "DestroyIt", se emplearon varias metodologías. Se combinó la metodología ágil Individual, desarrollo iterativo y el uso de Prototipos. La metodología ágil individual fue necesaria, ya que al ser un proyecto realizado solo por una persona. Asumí la responsabilidad de todas esas fases y procesos, desde la concepción, conceptualización, diseño, hasta la implementación del juego, en lugar de seguir un modelo de desarrollo tradicional con un equipo de varios integrantes, organizados jerárquicamente, donde cada proceso está dividido en diferentes fases.

Esta estrategia de metodología ágil permitió una adaptación continua y mejoras constantes, lo que probablemente contribuyó al éxito del proyecto y facilitó la toma de decisiones basadas en la evolución del juego.

Además, se adoptó un desarrollo iterativo, creando y guardando diferentes versiones del juego, que ayudaron a visualizar y tener un registro del videojuego. Esto ayudo a mejorar el juego en base a la experiencia y retroalimentación obtenida por parte de maestros y jugadores. Se utilizaron prototipos funcionales para probar conceptos y mecánicas de manera eficiente.

Durante el desarrollo de "DestroyIt" se desarrollaron diferentes versiones de prototipos, esto con el objetivo de probar y descartar diferentes mecánicas.

ANTECEDENTES

Dada la naturaleza de este proyecto es importante comenzar definiendo que es el arte digital, "el arte digital se refiere al que se crea digitalmente en una computadora, puede ser en forma de texto, imágenes, sonidos, letras, fotografías, entre otros" (Thomson, 2019) Este proyecto se enfoca en el desarrollo de un videojuego en el contexto mexicano. Según la revista digital Invdes, especializada en dar noticias de desarrollo e investigación, "México ocupa el 14º lugar en cuanto a ingresos en el área de videojuegos" (Contreras, R., 2018).

En Puebla, la industria de juegos es aún joven, son pocas las empresas y/o estudios existentes, entre ellas podemos encontrar a Bromio, Xolótl Estudios Interactive, Man-Eater Games, Yakanda Studios y Ficha Studios. La mayoría de las empresas mexicanas se encuentran en Monterrey, Ciudad de México o Guadalajara, existe La asociación Mexicana de Videojuegos que se encarga de apoyar y difundir los

proyectos de cada empresa de videojuegos en nuestra república, también apoya a la comercialización y atraer inversores.

Videojuegos de plataformas

Los videojuegos de plataformas tradicionalmente son 2D como Mario Bros, Sonic o Hollow Knight, aunque también hay algunos videojuegos de plataformas 3D, como Donkey Kong Country Returns, Crash Bandicoot 4: It's About Time o New Super Lucky's Tale, en estos juegos los jugadores controlan personajes que saltan y escalan entre diferentes plataformas.

Super Smash Bros para Nintendo 3DS y Wii U

Es un videojuego de lucha y plataformas creado por Hal Laboratory en conjunto con Sora Ltd. Y Dandai Namco Gam, fue publicado por Nintendo, es un juego que cuenta con varios personajes en el cual pueden competir hasta 8 jugadores, el objetivo del juego es sacar a los jugadores de las plataformas. Quien quede con vida al final será el ganador.

En este juego hay un mapa basado en el juego de plataformas Donkey Kong Country Returns llamado Jungle Hijinx, este escenario cuenta con dos tipos de escenarios, uno principal y uno secundario, una de las mecánicas del juego hace uso de este tipo de estos escenarios haciendo que se pueda acceder al fondo del escenario a través de barriles.

Esta mecánica ayudó al desarrollo de "Destroy it" ya que en ambos juegos se busca que el jugador tenga acceso a una parte delantera del mapa y una parte trasera, sin embargo, la forma de acceder a ellas es diferente, ya que en Jungle Hijinx se utilizan barriles para llegar de un lado a otro y en "Destroy it" el jugador tiene un botón integrado para pasar de una parte del escenario a la otra.

Ambos juegos son competitivos y también el sistema de vidas es similar. En Super Smash Bros. Ultimate se tiene un total de 3 vidas para cada jugador y el objetivo de los jugadores es acabar con la vida de los demás jugadores antes de que ellos acaben con las suyas. En "Destroy it" es el mismo objetivo, pero las vidas de los jugadores dependen de cuanto daño se le haga a su nave espacial y al igual que en Super Smash Bros. Ultimate, el aventar a un jugador del escenario es conveniente para tener una clara ventaja en el juego.

Ultimate Chicken Horse

Es un juego de plataformas lanzado en 2016 en Steam², desarrollado por Clever Endeavour Games. Es un juego festivo donde los jugadores construyen el nivel mientras juegan colocando trampas y obstáculos para molestar a los otros jugadores.

Es un juego competitivo y gracioso, no solo por sus mecánicas o dinámicas, también su arte es gracioso. Este juego no cuenta con una historia compleja, simplemente es para jugar y divertirse.

“Destroy it” al igual que Ultimate Chicken Horse no busca tener una historia compleja, su objetivo es solo que los jugadores usen sus mecánicas para molestarse unos a los otros, también busca ser llamativo y tener personajes carismáticos y graciosos.

En “Destroy it” los jugadores no pueden construir el nivel, como en Ultimate Chicken Horse, sin embargo, ambos son juegos de plataformas.

MARCO TEÓRICO

Diseño de juego

El diseño de un videojuego consiste en usar la creatividad para desarrollarlo con fines de entretenimiento, educativos o de investigación. El desarrollo de un videojuego involucra la creación de historias, niveles de juego, apartado visual, personajes, metas, reglas y retos que llevan a interacciones con otros personajes, usuarios y objetos dentro del juego.

El diseño de videojuegos es una rama multidisciplinaria, ya que para llevar a cabo esta labor es necesario tener un poco de conocimientos de todo: arte, programación y storytelling³. El diseñador de juego puede ser aquella persona que propuso una premisa para la historia de este, aquel que sentó las mecánicas o quien propuso el arte. Puede ser un equipo o una sola persona la que está encargada de la tarea y sienta las bases del juego.

No se debe subestimar la cantidad de trabajo que se requiere para realizar un videojuego. Hay libros que se dedican al estudio de diferentes tópicos dentro de la rama por ejemplo: An architectural approach to level design de Christopher W. Totten, Level design processes and experiences de Christopher W Totten, The art of game design a book of lenses de Jesse Schell o Level Up the guide to great video game design de Scott Rogers.

² Una plataforma de videojuegos para descargar, activar y jugar contenido digital de videojuegos para computadora.

³ Es el arte de comunicar una idea, mensaje o emoción a través de una historia que se cuenta de forma estructurada y coherente

El diseño de videojuegos abarca desde la concepción de una premisa para el juego basada en una historia compleja o simple, en las mecánicas del juego, o simplemente en el género de juego que se quiere realizar, y se extiende al diseño de niveles, dificultad del juego, diseño de personajes, construcción del mundo, pruebas, interfaz.

Los diseñadores de juegos suelen tener el perfil de artistas técnicos, ya que cuentan con habilidades técnicas y artísticas, no se limitan al diseño de mecánicas, dinámicas o al diseño de niveles, por su perfil pueden encontrar trabajos especializados en una variedad de áreas, que incluyen: Diseñadores de juegos, artistas de juegos, diseñadores de contenidos, programadores o diseñadores de sistemas, diseñadores de mundos o niveles, diseñadores de interfaces de usuario, diseñadores de nivel básico, directores y creativos.

Géneros de videojuegos

En el mercado hoy en día existen sin fin de videojuegos, clasificarlos por género y subgéneros ayuda a que sea más fácil para el jugador encontrar el juego que está buscando. Un juego puede tener varios géneros y categorías.

Se han propuesto varias categorías para clasificar los videojuegos. Sin embargo, aunque los expertos no están de acuerdo sobre cuántas categorías existen, se han identificado diez categorías que cubren los aspectos generales de los videojuegos. La propuesta de estas categorías ha surgido de una revisión completa de la literatura especializada en el campo de los videojuegos, como el libro *An architectural approach to level design* de Christopher W. Totten, así como de un análisis de la taxonomía utilizada tanto en la industria como en la comunidad académica. Esta clasificación se presenta como un marco de referencia que facilita la comprensión y el estudio de los diversos tipos de videojuegos.

Sandbox

Este género de videojuegos es asociado a la elección que hacen los jugadores, un mundo abierto y un *gameplay*⁴ no lineal. Aunque pueden existir objetivos en estos juegos, estos no suelen ser el enfoque principal de los jugadores. Aunque estos puedan llegar a tener una historia los jugadores no suelen seguirla de manera lineal. Todas las tareas para completar tienen diferentes formas de lograrse, y todo dependerá de cómo experimente el juego el jugador.

Muchos juegos Sandbox son muy conceptuales, el jugador no está atado a objetivos específicos, tiene libertad de explorar, interactuar o modificar el ambiente de juego.

Algunos ejemplos de este género son *Minecraft* (2011), *Terraria* (2011), *Don't starve* (2013), *Stardew Valley* (2016).

⁴ Se refiere a la jugabilidad o experiencia de juego que ofrece un videojuego al usuario mientras lo juega

Estrategia en tiempo real (RTS)

En este tipo de juegos los jugadores humanos y los NPCs⁵ controlan diferentes facciones y competir unos contra otros simultáneamente en tiempo real, los jugadores se centran en conseguir recursos para crear unidades y derrotar a su oponente, estos juegos suelen incluir la gestión de recursos y mapas.

Algunos ejemplos de este género son Age of Empires IV (2021), Warcraft III: Reign of Chaos (2002), StarCraft II: Wings of Liberty (2007), Halo Wars (2009).

Shooters

Son juegos donde la mecánica predominante es disparar, se ramifican en dos subgéneros principales: el shooter de primera persona (por sus siglas en inglés FPS) como Halo Infinite (2021), Left 4 Dead 2 (2009), Borderlands 2 (2012), entre otros, y el shooter en tercera persona (TPS), como Fortnite (2017), Gears of War 4 (2016), Resident Evil 4 (2005), entre otros. Aunque estos se pueden superponer ya que hay muchos videojuegos en los cuales se puede alternar entre puntos de vista.

La diferencia clave es la perspectiva ya que un FPS simula un punto de vista humano, mostrando especialmente lo que ve tu personaje y un TPS (Third Person Shooter) retrocede el punto de vista y muestra el personaje y lo que hay a su alrededor.

Multiplayer online battle arena (MOBA)

Los multiplayer online battle arena comparten características con los juegos de estrategia en tiempo real. Estos cuentan con una vista cenital que enfatiza el mapa y el manejo de este. En estos juegos los jugadores eligen un rol y en cada partida pueden irlos cambiando.

Los juegos MOBA (Multiplayer Online Battle Arena) priorizan el juego en equipo y competitivo. Se juegan contra otros jugadores en tiempo real.

Algunos juegos de este género son Dota 2 (2013), League of Legends (2009), Heroes of the Storm (2015), Vainglory (2014).

Juegos de Rol

En estos juegos el jugador toma el control de un personaje que puede subir de nivel obteniendo puntos de experiencia. Estos juegos tienen muchos subgéneros.

Role-Playing Game o juegos de rol con la sigla en inglés RPG: Estos juegos tienen sus orígenes en juegos de mesa como Dungeons and Dragons, así que suelen tener mecánicas similares a los juegos de mesa antes mencionados. Incluyen una

⁵ NPC (Non-Player Character) es un término en inglés que se refiere a los personajes no controlados por el jugador en un videojuego.

historia desarrollada, elementos narrativos, crecimiento de personaje, re-jugabilidad e inmersión.

Action Role Playing Game o videojuegos de rol de acción con la sigla en inglés ARPG: Son similares a los RPGs, pero con un énfasis en el combate como The Witcher 3.

Massively multiplayer online role-playing game o videojuego de rol multijugador masivo en línea con la sigla en inglés MMORPG: Combina el multiplayer masivo online con el juego RPG.

Roguelike: Son juegos que cuentan con gran re-jugabilidad ya que cada partida es diferente. Los niveles están creados por algoritmos inteligentes. La posición de niveles, los enemigos, armas y mejoras cambian cada vez que el jugador muere y se reinicia la partida.

Simulación y deportes

La variedad de los juegos de deportes se ha ido expandiendo con los años, es común ver un videojuego de este género asociado con alguna organización deportiva. Estos juegos pueden ir desde pistas de carreras, hasta un campo, rio o cancha.

Los juegos de simulación suelen combinarse con otros géneros como sandbox, juegos de construcción o de realidad virtual o aumentada. Estos juegos buscan simular la realidad o experiencias.

Algunos ejemplos de juegos de simulación y deportes son cities: Skylines (2015), Los Sims (2000), FIFA 2000 (1999)

Puzzlers y party games

Estos juegos se centran en las mecánicas y experiencia de juego. Los party games usan elementos de juegos multijugador, mientras que los puzzlers, pueden ser simples o muy complejos e inmersivos, agregando ambientes complejos o un gameplay híbrido. Estos juegos se centran en resolver acertijos, pueden resolverse o ser descubiertos mientras el juego avanza.

Algunos ejemplos de party games y puzzlers son Ultimate Chicken Horse (2016), Super Mario Party (2018).

Acción y aventura

Los juegos de acción y aventura se centran en tener combates, una historia inmersiva con algún giro y mecánicas de juego muy específicas. Por ello gran variedad de juegos pueden encajar en esta categoría. Estos juegos suelen brillar por sus historias y mundos envolventes.

Hay muchos ejemplos en esta categoría algunos de ellos son Ori and the Blind Forest (2015), God of war (2018), BioShock Infinite (2013), Death Stranding (2019).

Supervivencia y horror

Los juegos de supervivencia se centran en el manejo de recursos, incluyen la recolección y elaboración de recursos, además de incluir sistemas para salvar al personaje de ser necesario.

Los juegos de terror pueden abarcar títulos de supervivencia ya que ambos cuentan con características similares, juegos que incluyan zombis, apocalipsis, fantasmas, cualquier elemento que incluya sustos se considera terror. Estos juegos también hacen uso de la tensión para mantener al jugador inmerso.

No es extraño encontrar juegos que combinen ambas categorías.

Algunos ejemplos de juegos de supervivencia u horror son Don't starve (2013), five nights at Freddy's (2014), Resident Evil Village (2021).

Plataformas

Los juegos de plataformas surgieron a principios de los años 1980 y es uno de los géneros más estudiados por los desarrolladores de videojuegos, y está en constante crecimiento. Dos de los pioneros de este género fueron Space Panic de 1980 y Donkey Kong de 1981.

En estos juegos los jugadores se mueven de un punto a otro saltando, corriendo y trepando. Tienen una vista horizontal y controles simples. Puede ser 3D, 2D, isométrico o 2,5D. Se caracterizan por plataformas de diferentes tamaños por las que los jugadores deben navegar haciendo uso de sus habilidades. Estos juegos tienden a volverse más complejos con cada nivel. Son muy dinámicos y ponen a prueba los reflejos y la habilidad del jugador. Este género de plataformas es uno de los favoritos de los estudios independientes. Algunos ejemplos incluyen Cuphead (2020), Super Mario Bros (1983) y Gris (2018).

Mecánicas

Las mecánicas son la base del juego, éstas son la pieza fundamental para su funcionamiento, son reglas, acciones, sistemas y objetos que definen cómo funciona y se juega el juego. Incluyen todos los elementos interactivos que permiten a los jugadores participar en el juego y tomar decisiones.

“Las mecánicas describen los componentes particulares del juego, a nivel de representación de datos y algoritmos” (Hunicke, LeBlanc, & Zubek, 2004), por ejemplo, que el jugador pueda saltar, hacer un deslizamiento, correr, flotar, y en qué momentos estas acciones son permitidas son ejemplos de mecánicas, al igual que las monedas, misiones o logros.

Dinámicas

“Las dinámicas describen como se comportan las mecánicas en tiempo de ejecución, como actúan entre sí con las entradas y salidas del jugador” (Hunicke, LeBlanc, & Zubek, 2004). Son como el jugador interactúa con las reglas del juego y las decisiones que toma conforme este avanza.

Por ejemplo, que un jugador pueda sostener y aventar a otro es una mecánica de juego, cuando el jugador decide sostener a un oponente y aventarlo al precipicio o a una trampa es una dinámica de juego.

Hay que tener presente que las dinámicas ayudan a crear y fortalecer la estética del juego. Cuando las dinámicas se alinean con la estética y la narrativa del juego, se crea una experiencia más coherente y envolvente para el jugador. Por ejemplo, en un juego de terror, las dinámicas que fomentan la exploración cuidadosa y el sigilo pueden contribuir a la estética de tensión y misterio.

Estética

Cada juego tiene como objetivo evocar una respuesta emocional, a eso se le conoce como estética de juego. Tiene que ver con elementos como sensación, descubrimiento, desafíos, retos, fantasía, entre otros, estética “describe las respuestas emocionales deseables que se quiere evocar en el jugador, cuando este interactúa con el sistema de juego (Hunicke, LeBlanc, & Zubek, 2004).

Según el artículo MDA: A Formal Approach to Game Design and Game Research cuando nos referimos a estética nos acercamos a la taxonomía enlistada a continuación: Sensación (juego como placer sensorial), fantasía (el juego como fantasía), narrativa (juego como drama), reto (el juego como carrera de obstáculos), compañerismo (juego como marco social), descubrimiento (juego como territorio inexplorado), expresión (juego como autodescubrimiento) y sumisión (juego como pasatiempo). (Hunicke, LeBlanc, & Zubek, 2004)

Diseño de niveles

Requerimientos funcionales

Los requerimientos funcionales se refieren a que el juego funcione, que el diseño del juego cumpla con lo esperado, que el diseño de nivel sea apto para ser explorado por el jugador, sin importar el motor de juegos utilizado.

Usabilidad

La usabilidad se refiere a que los escenarios deben poder ser navegados por el usuario, en como este va de un punto a otro, sin que interfiera la cámara del juego. El diseño de nivel es un puente de comunicación entre el diseñador de juego y el jugador, esta comunicación debe ser clara. “De este modo, deberíamos concentrarnos en cómo los jugadores ven el espacio de juego a través de puntos de vista y cámaras de juego y cómo se navega por los niveles” (Totten, 2019, pág. 7)

Recompensas

Los niveles de juegos deben tener una recompensa y debe ser satisfactorio atravesar por ellos, por eso los diseñadores hacen uso de elementos psicológicos en el diseño de nivel para que el jugador tenga una respuesta emocional durante la navegación de este. “Al retar y recompensar a los jugadores, les ofrecemos la oportunidad de controlar sobre los elementos culminantes de los juegos como ningún otro medio de comunicación puede ofrecer.” (Totten, 2019, pág. 7)

Elementos históricos de arquitectura

Los elementos de arquitectura siempre están presentes en los diseños de juego, así el diseñador sea consiente de estos o no. Tenerlos presentes ayuda a dar mejor estructura al diseño de nivel y tener una idea más clara de cómo se quiere que el jugador avance por los niveles. Esto será con ayuda de plataformas, pilares, piedras, casas, entre otros elementos, todo dependerá del tipo de arquitectura que se requiera para el juego.

Arquitectura griega

La arquitectura griega se caracterizaba por como enfatizaba formas a través de ejes y líneas de visión. También solían meter elementos importantes al final de estas líneas de visión.

Arquitectura Romana

“Los romanos diseñaban experiencias lineales” (Totten, 2019, pág. 18), los foros romanos construidos en el primer siglo d.C. utilizaban arquitectura y lenguaje similar a las construcciones griegas, utilizaban espacios públicos y privados, también puntos acomodados axialmente que guiaba a los visitantes a través de los espacios públicos.

Los templos romanos se centraban en resaltar una zona importante, ofreciendo interiores gratificantes a los visitantes, y los espacios interiores de la arquitectura romana eran amplios por los ascos, bóvedas y cupulas.

Arquitectura medieval

La arquitectura cristiana se caracterizó por sus edificios llenos de narrativa y simulación. Los cristianos hicieron uso de elementos arquitectónicos que también fueron utilizados por los romanos, como elementos modulares y arreglos axiales, los utilizaron para representar la idea del reino de los cielos y la jerarquía entre los sacerdotes y fieles.

El uso de la iluminación en los edificios cristianos, con el uso de las ventanas elevadas que dejaban entrar la luz desde arriba de los espectadores creó la sensación de estar en un espacio sagrado al estar al interior del edificio.

Para adornar los edificios los cristianos utilizaron elementos simbólicos de la narrativa bíblica en diseños de mosaicos y esculturas con relieve.

Arreglos espaciales

La arquitectura dentro de los juegos difiere de la arquitectura que se usa en el mundo real ya que esta puede prescindir de espacios interiores o exteriores y no necesariamente puede influir uno en el otro.

En el diseño de niveles de un juego promedio los espacios interiores y exteriores están libres de requisitos como el clima, geología, entre otros, el diseño de juegos normalmente es el resultado de la distribución espacial basada en los patrones de movimiento que tienen los jugadores al explorar el nivel, esto da libertad al diseñador de niveles.

Figure-Ground

“Se deriva de las nociones artísticas del espacio positivo y negativo de una composición, donde el espacio positivo describe el área habitada por el sujeto de una pieza y el espacio negativo describe el espacio fuera o entre los sujetos.” (Totten, 2019, pág. 105) En el caso de la arquitectura los espacios positivos son aquellos donde se encuentran edificios y el espacio negativo es el terreno, las zonas donde solo hay suelo.

En diseño urbano los espacios positivos son lugares donde la gente puede habitar, como plazas, patios parques, casas, tiendas o cualquier zona de encuentro y los espacios negativos son las zonas donde la gente se mueve por ejemplo las calles o carreteras.

En diseño de niveles los espacios negativos están basados en como los jugadores recorren el nivel, y los espacios positivos son los lugares a donde viaja el jugador o donde debe llegar.

Form-Void

Este tipo de arreglo espacial es similar a Figure-Ground, pero en un espacio tridimensional, se crean espacios con elementos masivos, agregando objetos y o sustrayendo zonas de ellos. “En los juegos, estas adiciones y sustracciones pueden usarse para escondrijos, pasadizos secretos, puntos de francotirador o incluso objetivos de nivel resaltados.” (Totten, 2019, pág. 109)

Arrivals

Este arreglo espacial se centra en la experiencia que tiene un jugador al llegar a un espacio de juego por primera vez, “gran parte de la comunicación que tiene un desarrollador de niveles con el jugador es a través de las llegadas a un espacio dentro del juego” (Totten, 2019, pág. 109)

La primera impresión que experimenta el jugador al adentrarse en un escenario de juego se halla estrechamente vinculada con los lugares previos que ha recorrido. Además, el ángulo de cámara utilizado a la llegada del nivel resulta de gran importancia, ya que contribuye de manera significativa a la comprensión del entorno recién descubierto y a la generación de una experiencia sensorial particular.

Genius Loci

Este arreglo espacial se centra en las cualidades identificativas o sensación que genera un espacio, se centra en las experiencias que tendrá un jugador en el espacio diseñado, éstas van a depender del tipo de juego que se esté construyendo, y estas sensaciones se pueden generar a través de movimientos de cámara, la iluminación, organización, tamaño y objetos que se encuentren en el espacio, la experiencia de un jugador con Resident Evil no va a ser la misma de un jugador que juegue Mario Kart.

Estructuras del espacio de juego

Labyrinth

Esta estructura de juego se refiere a los laberintos clásicos que se representan en obras de arte de la época griega, se representaron como pasadizos en espiral que dan vueltas sobre sí mismos, cuentan solo con un camino y llegan a un punto final.

Los laberintos demuestran que, aunque los espacios de juego sean lineales, agregar giros, vueltas y desafíos, hacen que el camino que recorre el jugador sea más enriquecedor y entretenido.

Los laberintos orillan al jugador a seguir una serie de acontecimientos, por eso son útiles para un juego que tiene narrativa.

Maze

“Son rompecabezas espaciales en los que los ocupantes y jugadores deben encontrar el camino a través de una elaborada estructura de muros y caminos con múltiples callejones sin salida para encontrar un punto de salida.” (Totten, 2019, pág. 114)

En los recorridos que puede hacer un jugador dentro de un espacio de juego tipo maze puede encontrar enemigos, vidas, ítems, o el recorrido puede ir acompañado de un límite de tiempo, los callejones sin salida pueden llegar a ser positivos, porque el diseñador de nivel puede recompensar al jugador por explorar.

Rizoma

“El rizoma es un término que se utiliza en botánica, se refiere a las raíces formadas por tallos subterráneos.” (Totten, 2019, pág. 116). En esta estructura de juego un rizoma se refiere a un checkpoint⁶ que ayuda a viajar de un lado a otro

⁶ Un checkpoint en videojuegos es un punto específico en un nivel o sección del juego donde el progreso del jugador se guarda automáticamente. Si el jugador muere o falla después de pasar un checkpoint, puede reiniciar desde ese punto en lugar del principio del nivel o sección, lo que facilita la jugabilidad y evita la frustración del jugador.

instantáneamente. Se utiliza con mayor medida en juegos con mapas grandes, como es el caso de los MMORPGs como Guild Wars 2 o World of Warcraft.

Tipos de tamaño espacial

Espacio estrecho

Los espacios estrechos no son mucho más grandes que el tamaño del jugador, estos espacios crean tensión en los jugadores, porque no pueden moverse o actuar que en otras condiciones sí podría, según Totten, C. (2019) "hace que el jugador se sienta encerrado e incapaz de moverse" (pág. 118)

Juegos de sigilo, shooters o juegos de terror ocupan constantemente este tipo de espacio.

Espacio íntimo

Los espacios íntimos tienen una medida que es cómoda para el jugador, soporta las dimensiones del jugador y el movimiento de los personajes de juego. En estos espacios, superficies, ítems están al alcance de las capacidades del personaje. En juegos como shooters y arenas multijugador las zonas donde no hay ventaja para ninguno de los jugadores son espacios íntimos. En este tipo de espacios cuando se encuentran múltiples jugadores ninguno de ellos tiene ventaja sobre otro a pesar de las habilidades de los jugadores.

Los espacios íntimos en juegos multijugador se configuran estratégicamente para brindar comodidad y oportunidades equitativas a múltiples jugadores, fomentando la interacción y la competencia, mientras que en un videojuego de un solo jugador están hechos para dar ventaja sobre sus enemigos.

Espacios de prospección

Los espacios de prospección son descritos como espacios donde se tienen que buscar recursos para supervivencia expuestos a la naturaleza y depredadores.

En el caso de los juegos multiplayer, son espacios donde los jugadores tienen ventaja espacial sobre otros, un ejemplo es cuando se encuentran en la cima de un edificio.

En general los espacios de prospección dan una sensación de vulnerabilidad al jugador, el diseñador de nivel puede incrementar esta sensación con ciertos elementos, como niebla, oscuridad, sombras, música.

Se usan mucho en escenarios donde se encuentra un jefe final, en donde los jugadores no pueden usar sus habilidades para contar con ventaja espacial.

Planos de nivel

Los planos de nivel ayudan a un mejor entendimiento del espacio de juego y el acomodo de elementos, en este caso se utilizaron cinco tipos de plano de nivel para mejor entendimiento del juego.

El plano del nivel de juego es un dibujo de arriba abajo del nivel de juego, este se dibuja sin la perspectiva ayudan a visualizar muros, habitaciones o acomodo de las plataformas y dimensiones del espacio de juego.

El plano de sección es similar al plano sin embargo este está cortado verticalmente, estoy ayudan al diseñador de juego a planificar niveles que tienen un desplazamiento lateral o también a visualizar las plataformas en los juegos de batalla 2D, estos planos también ayudan a planificar el ritmo del nivel, y a visualizar formas alternativas de transición de nivel de un espacio a otro.

Los planos axométricos se hacen tomando un plano hecho y girándolo entre 30 o 45 grados y proyectándolo desde arriba para crear formas tridimensionales, ayudan a visualizar las plataformas desde un espacio tridimensional

Al igual que el plano axométrico los de perspectiva ayudan a visualizar el espacio 3D, sin embargo, estos planos se deforman conforme el punto de vista del espectador. Son útiles para organizar como va a estar acomodado el espacio del juego y reacomodar elementos.

Herramientas de desarrollo

Motor de juegos

Un motor de juegos es un entorno de desarrollo de software, con ajustes y configuraciones que optimizan y simplifican el desarrollo de juegos en diferentes lenguajes de programación. Usualmente incluyen motores gráficos 2D y 3D como Unity o Unreal, también cuentan con diferentes formatos de exportación por ejemplo para pc o para móvil.

Software

En el mercado hay una amplia gama de software especializado para el desarrollo de videojuegos. Entre los que podemos encontrar en la industria hay algunos que requieren más habilidades de hacer código que otros.

Es importante recalcar que para desarrollar juegos de manera profesional la mejor opción es usar un software que este en el estándar de la industria de juegos como Unity o Unreal, software gratuito y con una buena curva de aprendizaje.

ARTE

Pixel art

El píxel art es un estilo de arte digital dónde las imágenes se crean a partir de pixeles. “En pixel art, tienes control total y puedes manipular cada pixel por tu

cuenta" (Azzi, 2019, pág. 6) Es una técnica usada comúnmente en videojuegos. Se ha mantenido popular en la industria de los videojuegos debido a su simplicidad y eficiencia en términos de uso de recursos.

El pixel art suele caracterizarse por una paleta limitada. Esto se debió originalmente a las restricciones de los ordenadores utilizados, y más tarde fue influenciado en gran medida por las limitadas paletas de colores de las consolas para las que se creaban pixel art. (Hervieux, 2015, pág. 8)

Hoy en día, la creación de pixel art se ha vuelto accesible mediante la utilización de programas especializados disponibles en computadoras, iPads y dispositivos móviles. Entre las aplicaciones especializadas más destacadas se encuentran Aseprite, graphicsgale, promotion, 8bit Pintor, Pixel Studio, Pixel Art Paint Pro y Pixel Station. No obstante, es importante señalar que el pixel art también puede ser elaborado en programas de diseño ampliamente conocidos como Photoshop, Procreate, Clip Paint Studio, GIMP, Paint tool sai, entre otros. La distinción principal radica en que estos últimos requieren una configuración y la creación de pinceles personalizados para lograr resultados óptimos.

Aliased

El aliased es un efecto visual no deseado que se produce cuando una imagen digital, que debería ser suave y continua, muestra bordes dentados o escalonados debido a la limitación de la resolución. Esto ocurre cuando una imagen contiene detalles más finos que la resolución de la pantalla o la cuadrícula de píxeles en la que se muestra.

En el pixel art, el aliasing se manifiesta como bordes dentados y escalonados en las líneas y contornos de los objetos y personajes. A menudo, los artistas de pixel art evitan el aliasing para mantener un aspecto nítido y coherente en sus creaciones.

Según Azzi, M. (2019) los pros que tiene el aliased son que hace que las imágenes pequeñas sean leídas de mejor manera, limita la paleta de colores y es más rápido de hacer, sin embargo, una de sus desventajas es que la línea se vuelve filosa y en bloques. (pág. 43)

Anti-aliased

El anti-aliased es una técnica utilizada en gráficos por computadora para reducir o eliminar el efecto de aliasing. Consiste en suavizar los bordes dentados o escalonados alrededor de los objetos y líneas de una imagen, lo que da como resultado una apariencia más suave. En píxel art el anti-aliased es menos común porque, a menudo, los artistas desean mantener una estética "retro" y nítida. Según

Azzi, M (2019) los pros del anti-aliased son que en imágenes pequeñas las curvas se suavizan, es necesario para hacer imágenes con mayores resoluciones, los contras son que es tedioso y se ocupa demasiado y que hace un efecto de desenfoque en imágenes pequeñas. (pág. 43)

Paleta de colores

En un contexto artístico, las paletas de colores juegan un papel muy importante y diverso. Principalmente, proporciona coherencia visual al establecer límites en la elección de colores, lo que resulta en una presentación más armoniosa y fácil de asimilar. Además, el color juega un papel fundamental como medio para expresar emociones y contar una historia, ya que la paleta de colores puede enriquecer una narrativa y transmitir un estado de ánimo deseado.

Las paletas de colores son herramientas que optimizan la eficiencia del proceso creativo y aceleran la producción artística al eliminar la necesidad de tomar decisiones de color constantemente.

Las limitaciones impuestas por la paleta estimulan la creatividad y alientan a los artistas a encontrar soluciones innovadoras y originales dentro de las limitaciones de la selección de colores. De manera similar, una paleta equilibrada actúa como una barrera contra la saturación visual, permitiendo al espectador centrarse en los elementos importantes de la obra.

Para hacer una paleta de color comúnmente se utiliza el círculo cromático, una herramienta gráfica circular que muestra todos los colores visibles organizados de manera sistemática según su relación y armonía en el espectro cromático. Esta rueda se compone generalmente de los colores primarios (rojo, azul y amarillo), los colores secundarios (verde, naranja y morado) que se obtienen al mezclar dos colores primarios, y los colores terciarios, que surgen de combinar un color primario con un color secundario adyacente.

Un aspecto muy importante del círculo cromático como herramienta es que demuestra que el color se percibe con una temperatura asociada: cálido o frío, en distintos grados. Esta característica es esencial a la hora de expresar un determinado mensaje cromático. La relación de un color con sus adyacentes suele determinar su grado de calidez o frialdad (Eiseman, 2018, págs. 16-17)

En el contexto de pixel art, cada píxel es importante. Una paleta de colores limitada evita la saturación visual y garantiza que los detalles y contornos sean nítidos y legibles, lo que es esencial en juegos o ilustraciones donde la claridad es importante. Existen “cuatro combinaciones de colores principales” (Adobe, 2022)

Monocromática

Una paleta monocromática se compone de colores derivados de una tonalidad base, estos colores pueden ser más oscuros o claros. Dentro de esta paleta, se encuentran diversos tonos, matices y saturaciones. “La desventaja de las paletas monótono es que pueden resultar monótonas “ (Eiseman, 2018, pág. 18)

Análoga

Una paleta análoga se conforma de colores que se ubican próximos entre sí en el círculo cromático. “Las combinaciones de colores análogos se obtienen eligiendo un grupo de 3 a 5 colores que estén próximos entre sí en la rueda cromática.” (Adobe, 2022). Estos colores comparten una tonalidad base idéntica y se distinguen principalmente por su nivel de saturación y luminosidad. Normalmente, una paleta análoga está compuesta por colores que son vecinos en la rueda cromática, lo que conduce a la creación de armonías visuales suaves y agradables.

Complementaria

Una paleta de colores complementaria se compone de colores que se encuentra en la parte opuesta del círculo cromático con respecto a otro color en la misma paleta. Esta elección de colores complementarios se caracteriza por generar un contraste visual fuerte. Como se lee a continuación “cuando los complementarios se sitúan inmediatamente uno al lado del otro, parecen más intensos. Este fenómeno se llama contraste simultáneo- cada complemento intensifica el brillo de su opuesto y parece vibrar visualmente en la zona donde ambos se unen. “ (Eiseman, 2018, pág. 20)

Triádica

Una paleta de color triádica es un conjunto de tres colores que se eligen de manera que estén igualmente espaciados en el círculo cromático. Esto significa que los tres colores seleccionados tienen una separación de 120 grados entre sí, creando un triángulo. “Este tipo de combinaciones pueden crear resultados elegantes interesantes y originales” (Eiseman, 2018)

Animación

La animación cuadro a cuadro, llamada animación tradicional, consiste en crear cada fotograma de la animación manualmente. Para ello, se dibuja cada uno de los cuadros en secuencia, creando la ilusión de movimiento cuando se reproducen a una velocidad determinada.

En la animación hay 12 principios según el libro "The illusion of life" de Thomas, F. y Johnson O. a continuación se resumen cada uno de los 12 principios de animación.

Aplastar y estirar (squash and stretch)

Este principio se refiere a la deformación temporal de un objeto o personaje para dar la sensación de movimiento realista. "La posición aplastada (squashed) puede describir una excesiva presión o una deformación exagerada. La posición estirada (stretched) siempre muestra la misma forma en una condición muy "extendida" (Thomas & Johnston, 1995, pág. 49)

Anticipar (anticipation)

La anticipación en animación se realiza antes de que un objeto o personajes realice la acción principal, este se mueve en la dirección opuesta preparando al espectador, mientras mayor sea la acción principal, mayor va a ser la anticipación. Según Thomas, F. y Johnson, O. (1995) "la anticipación podía ser tan pequeña como un cambio de expresión o tan notoria como un movimiento físico" (pág. 53)

Puesta en escena (staging)

Este principio se refiere a que la animación debe reflejar las emociones y personalidades de los personajes, lo que hace que la narración sea más convincente. Según Thomas, F. y Johnson, O. (1995) "una acción es puesta en escena o actuada para que se entienda y reconozca una personalidad, se vea expresión, o para que un comportamiento conmueva a la audiencia." (pág. 54)

Acción paso a paso y acción pose a pose (straight ahead action and pose to pose)

La animación paso a paso implica crear cada fotograma mientras se va avanzando la animación, lo que da como resultado una animación espontánea, pero dificulta mantener consistencia en la masa del objeto y en el tiempo de la animación.

La animación pose a pose se centra en primero dibujar las poses clave para cada acción y luego completar con poses intermedias. Esto reduce los errores, en proporciones, pérdida de masa y cambios de timing, pero puede resultar en movimientos que parecen más genéricos y predecibles en comparación con la animación paso a paso.

Como afirma el libro "The illusion of life" "ambos métodos están todavía en uso, debido a que cada uno ofrece ciertas ventajas para diferentes tipos de acción. Usualmente son combinados, para mantener la acción continua al alcance de la mano" (Thomas & Johnston, 1995, pág. 58)

Acción de sobreimponer y acción de principio a fin (follow through and overlapping action)

El overlapping en animación se refiere a la manera en que diferentes partes del cuerpo o un objeto se mueven a velocidades distintas y se detienen en momentos diferentes durante la acción.

Por otro lado, el follow through se refiere a cómo se comportan las partes de un objeto o personaje una vez que se ha completado su acción principal. Se refiere a que las partes de un personaje u objeto no se detienen al mismo tiempo al final del movimiento, sino que siguen moviéndose momentáneamente en la dirección en la que se dirigían antes de detenerse por completo.

El cuerpo en sí mismo no se mueve todo a la vez, sino que se comprime, se aplasta, gira, se dobla y se contrae mientras las demás partes interactúan unas con las otras. Mientras algunas partes se detienen en seco, otras partes deben estar aún en movimiento; un brazo debe seguir en movimiento, aún si el cuerpo ya no se mueve. (Thomas & Johnston, 1995, pág. 60).

Salida lenta y llegada lenta (slow in and slow out)

En animación, los principios de "slow in" y "slow out" se refieren a la manera en que los objetos o personajes comienzan y terminan un movimiento. Estos principios hacen que la animación sea más natural y realista.

El término "slow in" se refiere a un movimiento que comienza lentamente. El movimiento de objetos y personajes no comienza a máxima velocidad, sino que comienza lentamente y se acelera con el tiempo. Esto muestra que los objetos reales necesitan tiempo para superar la inercia antes de poder moverse rápidamente.

El "slow out" se refiere a la gradual conclusión de un movimiento. Un objeto o personaje no se detiene de inmediato; disminuye gradualmente su velocidad hasta que finalmente se detiene por completo. Esto se asemeja a la inercia en la vida real, donde los objetos no pueden detenerse de inmediato.

Arcos (arcs)

El movimiento en la naturaleza con frecuencia sigue un patrón curvo. El principio de arcos utiliza el movimiento de objetos y personajes en lugar de un movimiento corporal recto o rígido. Esto da a la animación una apariencia natural.

Acción secundaria (secondary action)

En los principios de la animación, un tipo de movimiento o acción adicional que se superpone y complementa la acción principal de un personaje u objeto en una escena de animación se denomina acción secundaria, al imitar el movimiento

natural del mundo real, esta acción secundaria suele añadir profundidad y realismo a las animaciones. “Siempre está subordinada a la acción central. Cuando entran en conflicto o se vuelven dominantes o más interesantes, suponemos que están mal implementadas o se escogió la acción equivocada.” (Thomas & Johnston, 1995, pág. 65).

Las acciones secundarias sirven esencialmente para dar vida a escenas y personajes agregando detalles sutiles que enriquecen la historia y expresan las personalidades de los personajes. Podría ser un personaje que mueve las manos mientras habla, saluda mientras camina o un objeto que reacciona a movimientos clave en la escena.

Sincronía (timing)

Como menciona Thomas “El número de dibujos usados en cualquier movimiento determina la cantidad de tiempo que cada acción tendrá en pantalla” (Thomas & Johnston, 1995, pág. 65). El principio de animación conocido como "timing" se refiere a la manipulación y control específico de la velocidad y duración del movimiento en una animación. En otras palabras, determina cuánto tiempo lleva realizar una acción dentro de una animación.

El timing afecta la percepción del espectador sobre la velocidad, el peso, el poder y la intención del objeto o personaje.

Exageración (exaggeration)

El principio de animación de exageración se refiere a cuando un animador exagera movimientos, características físicas o emocionales, o comportamiento de un personaje u objeto con el objetivo de hacer que la animación sea más llamativa. Sirve para transmitir una emoción, acción o situación de manera más efectiva y clara.

Dibujo sólido (solid drawing)

El principio de animación llamado "solid drawing" trata sobre la habilidad de los animadores para crear personajes y objetos tridimensionales que tienen dimensión y volumen en un espacio bidimensional. Este principio se basa en la comprensión de la anatomía, estructura y forma de los elementos animados.

Los animadores deben poder representar objetos y personajes como si estuvieran en un espacio 3D, incluso si sólo están dibujados sobre una superficie plana. Comprender la anatomía, la perspectiva, las sombras y la iluminación es clave para hacer que los elementos animados sean consistentes y creíbles.

Apariencia (appeal)

"Appeal" a que un personaje u objeto sea atractivo, interesante y memorable para el espectador. Existen varios factores que contribuyen al atractivo de la animación, como el diseño del personaje, personalidad, expresiones faciales, movimientos y la conexión emocional con la audiencia.

Los personajes con appeal tienen características visuales que los hacen fácilmente reconocibles y únicos, para los espectadores. También tiene una personalidad bien desarrollada, que se puede ver en sus acciones y comportamientos. Los personajes encantadores atraen a la audiencia y la hacen sentir empatía. Como resultado, la audiencia tiende a interesarse y sentir empatía con la historia y el personaje.

Animación en el motor de juegos Unity

En el contexto de videojuegos, el Sprite se refiere a una imagen 2D que se utiliza para representar un personaje, objeto, elemento o fondo dentro de un juego. Es comúnmente utilizado en juegos retro y en plataformas con limitaciones de hardware. El sprite puede ser animado o estático y se mueve en la pantalla en respuesta a las interacciones del jugador o a eventos programados.

El animador se ha convertido en una herramienta esencial para la creación y gestión de animaciones, lo que permite una experiencia de juego más inmersiva y dinámica. Es un componente y sistema completo de Unity diseñado para controlar y administrar las animaciones en videojuegos. Este elemento es fundamental para crear personajes y objetos animados, así como para desarrollar mecánicas de juego más complejas.

Dentro del entorno Unity, las animaciones son fundamentales. Pueden ser importadas desde fuentes externas o creadas internamente. El control de estas animaciones se lleva a cabo a través de los "Animator Controllers", archivos que definen estados, transiciones y condiciones para la reproducción de animaciones.

Los Animator Controllers permiten el uso de parámetros para controlar las transiciones entre estados de animación. Estos parámetros pueden ser de distintos tipos, como booleanos, enteros, flotantes o triggeradores. Además, se pueden emplear los "Blend Trees" para mezclar animaciones en función de los valores de los parámetros. Esto facilita la creación de animaciones complejas y fluidas.

Las transiciones entre estados de animación definen cómo y cuándo se cambia de una animación a otra. Estas transiciones pueden activarse mediante condiciones, temporizadores o eventos específicos en el juego, lo que permite un control dinámico y preciso de las animaciones.

Animación de partículas para videojuegos.

Las partículas y los efectos visuales son importantes para los juegos porque pueden mejorar la experiencia del jugador al agregar detalles visuales que hacen que el juego sea más interesante y atractivo. Las partículas pueden crear cosas como

humo, fuego, polvo, chispas, explosiones y otros efectos. También se pueden utilizar partículas y efectos visuales para enfatizar elementos importantes del juego como: recoger objetos, derrotar enemigos o completar objetivos.

DESARROLLO DE PROYECTO

Diseño de juego

Para este proyecto se decidió que las mecánicas y modo de juego darían la base al aspecto visual, por lo tanto, fue en lo que más se trabajó y pasó por diferentes conceptos antes de llegar a la idea final del juego.

La primera idea fue un juego para un solo jugador donde a través de portales se le permite pasar entre un plano delantero a un plano trasero con una vista en espacio 2D. Al plantear el proyecto, la idea de usar portales se desechó dejando solo el concepto de un juego en el que el jugador pudiera saltar entre dos planos en un ambiente 2D.

La segunda idea planteada fue un multijugador 2D a contra tiempo que respetara el concepto de los dos planos interactivos. En esta propuesta los jugadores portarían prendas, lo que daría paso a la mecánica principal que consiste en que debían quitarles las prendas a otros jugadores. El ganador sería quien al terminar el tiempo tuviera puestas más prendas, sin embargo, la idea también fue desechada ya que el concepto de despojar de su ropa a otros jugadores no resultó tan divertido y atractivo al momento de probar el prototipo.

La tercera y última propuesta fue un multijugador 2D donde cada jugador cuenta con un objeto, que podía ser un tótem, una nave o un peluche. El objetivo de los jugadores es destruir los objetos de otros competidores.

El resultado final es una propuesta que involucra las tres ideas descritas y a continuación se presenta el desarrollo de esta.

Concepto

Seis marcianos han aterrizado accidentalmente en la tierra, pero solo uno puede salir de ella, cada jugador puede elegir entre seis especies de marcianos, el objetivo de cada jugador es destruir la nave de los otros marcianos, y proteger la suya.

Género: Plataforma, batalla, casual.

Público: Todas las edades

Modo de juego: Multijugador

Resumen

Cuatro jugadores se enfrentan cara a cara en diferentes arenas, todos tienen el mismo objetivo, proteger su nave espacial. El jugador con la última nave en pie será el ganador.

Jugabilidad

Es un juego de plataformas y batalla en el cual los jugadores controlan un marciano y deben proteger su nave usando dos acciones principales: agarrar y aventar. Con ellas deberán enfrentarse a los otros jugadores y proteger su nave. Además, cada nivel contara con plataformas colocadas en planos diferentes.

Mindset⁷

Favorecer un ambiente competitivo entre los jugadores que los lleve a idear estrategias, y retarlos con la navegación del nivel.

Estilo visual: Píxel art.

Desarrollo técnico

En esta sección aparecen enumerados los elementos que conforman el videojuego "DestroyIt"

1. Menú de inicio: Es la primera pantalla que podrá ver el jugador, en el menú de inicio está conformado por un nivel simple y las naves espaciales colocadas en una de las plataformas, el jugador puede sacar su cursor presionando A.

1.1 Selección de jugadores: Cuando el jugador tenga su cursor activo en el menú le será posible seleccionar una de las naves y en cuanto lo haga aparecerá el marciano correspondiente, esto se decidió así para que sea claro para el jugador que nave le pertenece a su marciano.

1.2 Selección de arena: La arena también se selecciona en el menú de inicio, este se selecciona cuando un jugador va hacia el icono de la arena.

5. Juego: El juego está conformado por plataformas, fondo, props, personajes y objetos de juego.

5.1 Pausa: Durante la ventana de juego los jugadores pueden acceder a un menú de pausa.

5.2 Reiniciar: En caso de ser necesario los jugadores pueden acceder a un menú donde pueden reiniciar el juego.

5.3 Regresar al menú de inicio: En caso de ser necesario los jugadores pueden acceder a un menú que les permite regresar al menú de inicio donde se pueden seleccionar los personajes y nivel de juego.

Mecánicas

1. El jugador puede moverse de derecha a izquierda y saltar, para moverse en el nivel de juego.
2. El jugador puede agarrar y lanzar jugadores y objetos, esto con el objetivo de hacerle daño a otros jugadores o las naves espaciales enemigas.

⁷ Se refiere al conjunto de actitudes, creencias y mentalidad que un jugador adopta al enfrentar un juego.

- 2.1. El jugador tiene la opción de elegir que tan lejos va a aventar un objeto, mientras más tiempo presione el botón, más lejos lanzará.
3. En el nivel puede moverse por las plataformas solo en un plano o saltar de un plano a otro, dependiendo de donde quiera ir.
4. Si un jugador está cargando algún objeto, o ya está siendo cargado, no puede cargar otra cosa.
5. Cuando un jugador que está cargando algo y se cae o es golpeado por algo, suelta lo que carga.
6. El jugador que logre que su nave espacial no sea destruida gana la partida.

Dinámicas

- Las plataformas de abajo se encuentran en el primer plano, allí se instancian los ítems aleatoriamente a lo largo de la plataforma, el jugador que baje a esa plataforma no sabe en qué momento instanciará un ítem, ni en que parte de la plataforma aparecerá, no tiene la certeza de ser el quien lo tome o si otro jugador lo tomará primero y sacará ventaja de ello atacándolo, esto hace que los jugadores se sientan nerviosos y hace que su sentimiento competitivo se vuelva más fuerte.
- En la parte alta se encuentran las naves, hay dos plataformas, una ubicada en el primer plano del nivel y otra en el segundo plano, el jugador que se encuentre en la cima contará con ventaja, ya que, al ser la zona de naves, podrá golpearlas y bajarles vida o destruir alguna, pero se sentirá amenazado si otro jugador está en la cima y su nave cuenta con poca vida.
- Se puede llegar a la parte alta saltando entre dos columnas, ya sea del lado derecho o izquierdo, si un jugador se encuentra saltando hacia arriba otro podría negarle el paso, golpearlo y aventarlo, esta situación resulta divertida la mayoría de las veces, ya que el objetivo al hacerlo es solo molestar.
- Si un jugador está cargando a otro, el jugador cagado puede zafarse dependiendo, presionando rápidamente el botón de agarrar/aventar.
- En el transcurso del juego hay ítems que aparecen aleatoriamente en ciertas plataformas, los jugadores que van a esas plataformas tienen la posibilidad de conseguirlos, lo cual implica una ventaja sobre otros jugadores. Si un jugador logra golpear a otro jugador que esté cargando un ítem, este objeto será soltado, lo cual le da oportunidad a los demás de tomarlo.

Diseño de nivel

Requerimientos funcionales

El nivel cuenta con assets⁸ 2D dibujados con perspectiva, esto con el objetivo de que el jugador no se pierda entre planos, los assets que se encuentren en la parte

⁸ Un asset, en el contexto de los videojuegos, se refiere a cualquier elemento o recurso utilizado en el desarrollo de un juego, como modelos 3D, animaciones, efectos de sonido, música, imágenes, texturas y otros elementos visuales y auditivos.

delantera son más saturados, para crear la ilusión de profundidad, ya que en la naturaleza y el arte los objetos más cercanos se encuentran con mayor saturación mientras que los objetos que se encuentran más lejanos se van desaturando hasta desaparecer en el horizonte.

Las animaciones ayudan a que el jugador entienda qué está pasando en el juego, por ejemplo, cuando camina o cambia de un plano a otro, también nos ayudan a identificar cuando se daña una nave. La animación de explosión en las naves ayuda a crear presión entre los jugadores.

Las naves tienen una animación de explosión cada que reciben daño esto ayuda a los jugadores a saber que su nave ha sufrido daño y a hacer un cálculo de cuántas veces podrán seguir regenerándose, genera presión y la necesidad de crear estrategias para no ser el primero en salir del juego.

Para montar el nivel y hacerlo funcional se utilizó el motor de juegos Unity, el cual tiene una gran facilidad para hacer el blocking del nivel⁹, además de agregar partículas y otros efectos visuales que ayudan a crear una mejor experiencia de juego.

Usabilidad

El nivel está armado para que el jugador encuentre el flujo y ritmo entre plataformas y diferentes planos. Los jugadores serán guiados a través de la saturación en los colores.

Las partículas son retroalimentación visual que ayuda a que el jugador identifique con mayor facilidad si ha recibido algún tipo de daño por parte de otro jugador o ha caído por un precipicio.

Recompensas

Los niveles son retadores, lo cual hace que los jugadores se sientan satisfechos una vez que logran dominarlo. El jugador que lo domine primero se siente más satisfecho, ya que, al dominar el nivel, tiene ventaja sobre los otros jugadores, lo cual le permite avanzar con mayor rapidez. Los niveles no tienen un piso continuo por lo que al mínimo error un jugador puede caer al precipicio y reaparecer, esto hace la experiencia del juego más emocionante ya que, si un jugador logra dominar el nivel la altura se vuelve una aliada para acabar con otros jugadores arrojándolos de las plataformas.

Elementos históricos de arquitectura

Arquitectura griega

Relaciones espaciales y simbólicas

⁹ El blocking de nivel es el proceso de creación de la estructura básica de un nivel de juego utilizando formas geométricas simples, con el fin de definir el flujo de la jugabilidad, la escala y las proporciones del nivel antes de agregar detalles y elementos visuales más complejos.

En el diseño del nivel hay líneas de visión que ayudan al jugador a identificar lugares importantes, por ejemplo, los espacios dónde se encuentran las naves o los ítems, además las líneas de visión ayudan a guiar a los jugadores por el mapa. Durante el recorrido del nivel se aprecian jerarquías y arreglos espaciales. A continuación, se presentan dos ejemplos:

- Apreciamos las jerarquías con los ítems (objetos necesarios, pero no tan importantes) aparecen en la parte baja del nivel, sin embargo, las naves que son vitales para seguir en el juego aparecen en la parte superior.
- Los arreglos espaciales los apreciamos en la composición y ritmo del nivel el cual guía al jugador por diferentes caminos a tomar para llegar de un punto A a un punto B, y qué espacios tienen mayor importancia. En este caso, las columnas por las que se tiene que saltar para llegar a la cima, donde están las naves, son las protagonistas en los arreglos espaciales, ya que nos muestran que camino debe tomar el jugador.

Arquitectura Romana Experiencias lineales

Los jugadores aprecian el nivel jerárquicamente ya que al inicio del juego aparecen en la parte inferior del nivel, y conforme avanza el juego son guiados sutilmente a la parte más importante del nivel a través de las columnas que indican como se puede llegar al área de naves, a su vez las columnas ayudan al jugador a descubrir dos mecánicas claves, la mecánica de salto por las paredes y la de deslizamiento por las paredes, la cual los ayudara a permanecer dentro del nivel el mayor tiempo posible.

Arquitectura Medieval Narrativa incrustada

La narrativa del juego es simple, unos marcianos se han estrellado en la tierra, solo uno puede salir de ella. Las plataformas y fondo del nivel están ideados para que sean lo más parecido al mundo que recordamos: nubes, cielo azul, rocas, plantas, entre otros elementos. Sin embargo, para enfatizar la idea de que los jugadores vienen de otro planeta los marcianos tienen un diseño de personaje muy alejado a lo que podríamos vincular al planeta tierra; sus formas son variadas y sus colores son más saturados y llamativos que los colores que se pueden ver en plataformas y fondos.

Las naves, para enfatizar a quien pertenecen, sus diseños se basan en los personajes, en forma y en color. Así los jugadores al verla sabrán qué nave pertenece a cada marciano, lo que facilitará la jugabilidad.

La narrativa también se enfoca en mostrar a los jugadores los puntos más importantes del nivel, por ejemplo, dónde están las naves, ítems y dónde reaparece si muere. El nivel es claro al entender que las naves son el punto clave de la

jugabilidad, ya que están en la parte alta, donde solo se puede llegar subiendo columnas verticales.



Ilustración 1 Muestra una captura del juego con el recorrido del nivel, los espacios, tipo de narrativa y relaciones espaciales.

Arreglos espaciales

Los arreglos espaciales dan al diseño de nivel más libertad ya que se basan en las mecánicas del juego. Los patrones de movimiento que tiene el jugador y los eventos narrativos, además de que los arreglos espaciales no se preocupan por el estilo visualmente el juego, si no en cómo se navega el nivel.

Figure-Ground

En el nivel los espacios positivos los encontramos en las plataformas, lugares que se convierten en puntos de encuentro entre los jugadores, sin embargo, que una plataforma sea un espacio positivo va a depender del plano en que se encuentre el jugador, esto quiere decir que sí un jugador se encuentra navegando en el primer plano del nivel los espacios positivos están conformados por las plataformas que se encuentran también en el primer plano, y las plataformas que se encuentran en el segundo plano forman parte de los espacios negativos que solo se utilizan como una zona de tránsito y no como zona de encuentros, sin embargo, si el jugador se encuentra en el segundo plano del nivel la situación se invierte, esto ayuda a crear más dinamismo y diferentes formas de navegar el nivel

La mecánica de que el jugador pueda elegir en que plano va a transitar cambia la percepción que este tiene sobre los límites de la navegación del nivel y las posibilidades que este tiene. Los espacios positivos pueden convertirse en espacios negativos y viceversa dependiendo de donde se encuentre el jugador.

Form-Void

El nivel está conformado por nueve plataformas, que son diseñadas para ser solo una plataforma en perspectiva falsa, no están conformadas por tiles¹⁰. En el diseño de nivel no hay espacios ocultos al jugador. El juego al ser completamente 2D puede prescindir del Form-Void. Aunque tener en cuenta el nivel como un espacio tridimensional ayuda a entender mejor los espacios dentro del juego y como estos se transitan, tener en cuenta una vista isométrica del nivel ayuda a comprender los saltos del plano delantero al plano de atrás y viceversa. También ayuda a considerar la altura con la que contará el nivel.

Arrivals

Al cargar el nivel los jugadores aparecen en la plataforma inferior, la cámara les da un plano abierto del nivel para visualizarlo por completo. Lo primero que ven es el ambiente que los rodea, el cual se conforma por varias plataformas y columnas flotantes y en la parte superior instanciadas en posiciones aleatorias las naves espaciales de cada jugador. La altura se usa para generar una sensación de peligro por una probable caída.

Al ser el primer nivel no se pretende ocultar peligros al jugador, se busca mostrarle mecánicas y cuáles son sus limitaciones, por ello la paleta de color del nivel es cálida y los jugadores notarán como algunas plataformas menos saturadas que otras, lo que les indica cuáles plataformas están en el primer plano y en el segundo plano. También se muestran columnas que los guiarán a la parte de arriba del nivel.

Los jugadores tendrán diferentes opciones para recorrer el escenario dependiendo de donde se haya instanciado su nave espacial y que nave quieran destruir primero. También su recorrido dependerá de la habilidad para cambiar de plano.

Genius Loci

La forma en que está construida el nivel obliga a los jugadores a mantenerse en movimiento, ya que el nivel es un espacio abierto y no tiene callejones sin salida, además de que si los jugadores se quedan en la parte de abajo del nivel esperando

¹⁰ Tiles es una palabra en inglés que en español significa azulejo o mosaico, los tiles en el contexto de videojuegos se refieren a imágenes de tamaño uniforme, que representan elementos individuales del juego como suelos paredes, obstáculos, objetos, etc. Los diseñadores de juego pueden colocarlos de manera individual para construir niveles o escenarios. Son muy comunes en juegos 2D de plataformas o juegos de rol entre otros.

únicamente los ítems no podrán cumplir el objetivo de destruir las naves que se encuentran en la parte superior del mismo.

La experiencia de juego del nivel pretende ser divertida y competitiva. Para lograrlo hay tres plataformas principales donde ocurren la mayoría de los encuentros y batallas, las primeras dos plataformas son aquellas en que se instancian las naves, ya que allí suben los jugadores con la intención de destruirlas y proteger la suya, la otra plataforma es donde se instancian los ítems, lugar de constantes encuentros ya que en ese lugar se obtienen objetos para dañar las naves enemigas.

Las formas de las plataformas tienen la intención de atraer a los jugadores con objetos que les darán ventaja y las plataformas de arriba atraen a los jugadores porque allí se encuentran las naves que deben destruir, pero las situaciones que se pueden generar en ambas plataformas generan sentimientos competitivos y de nerviosismo, además de situaciones divertidas por la expectativa de lo que puede suceder al enfrentarse con otros jugadores.

La música del juego tiene un ritmo rápido para invitar a los jugadores a moverse en todo momento, también tiene un estilo sci-fi para ayudar a reforzar la idea de que los personajes son de otros planetas.

Estructura del espacio de juego

Maze

La estructura del juego se asemeja a un espacio tipo maze ya que diferencia del espacio de juego tipo rizoma, se cuenta con una entrada y una salida preestablecida. La entrada corresponde a la plataforma inferior del nivel, donde los jugadores aparecen al inicio de la partida. La salida está en la zona superior, que es el objetivo al que deben llegar los jugadores. A diferencia del espacio de juego tipo rizoma, no cuenta con callejones sin salida como en el espacio de juego labyrinth.

El jugador puede encontrar distintas formas de llegar a las naves. Podrá escoger el lado derecho o izquierdo de las plataformas, dependiendo de su habilidad y las decisiones que tome. Existen plataformas que funcionan como vía para llegar al objetivo. Por su parte, la plataforma inferior se presenta como un espacio de recompensa, aunque se encuentra alejada del objetivo. Es recomendable dirigirse a ella, ya que allí se encuentran ítems útiles para completar el juego.

Tipos de tamaño espacial

Espacio estrecho

En el diseño del nivel, se contemplaron los espacios estrechos como una herramienta para generar tensión y estrategia en el juego. Los espacios se corresponden con las dos columnas que acceden a las naves, similares a pequeños pasillos verticales. Durante el ascenso por las columnas, el jugador se ve limitado a saltar, lo que hace que aventar un objeto sea un desperdicio, ya que este chocaría

con la pared y caería automáticamente, lo que impediría que el jugador llegue a la zona de naves con algún arma para hacer daño a sus adversarios.

Al alcanzar la cima de las columnas, el jugador queda expuesto a ataques de otros jugadores, lo que aumenta la tensión en el juego y requiere que el jugador adopte una estrategia para defenderse o avanzar con éxito. De esta manera, se buscó diseñar un nivel que no solo resultara emocionante, sino que también incentivara la estrategia y la planificación por parte de los jugadores.

Espacio íntimo

En relación con la planificación del diseño de niveles, se consideraron las plataformas horizontales como espacios íntimos, zonas que comparten los jugadores. Estas plataformas se caracterizan por ser terrenos amplios y parejos, donde no existen puntos de ventaja importantes. En este tipo de plataformas, los jugadores pueden sentirse con las mismas posibilidades, ya que, aunque se encuentren en la zona de ítems, cualquier jugador puede obtener uno, no por cuestiones de ventaja, sino por habilidad.

Asimismo, se emplean espacios íntimos en combinación con espacios estrechos en los niveles, lo que genera situaciones diversas y divertidas. Este enfoque incentiva el espíritu competitivo de los jugadores y los alienta a crear estrategias. Así, se logra un diseño de niveles que ofrece una experiencia de juego atractiva y desafiante, y que permite a los jugadores disfrutar de un ambiente equitativo y justo.

Espacio de prospección

La plataforma donde reaparecen los jugadores y donde se instancian los ítems, se concibió como un espacio de prospección, ya que en ella los jugadores podrían encontrarse con jugadores armados que solo esperan atacar a sus adversarios, para evitar que suban a destruir las naves, lo que podría intimidar a aquellos que lleguen sin armas.

De esta manera, se planteó un diseño de niveles que, colocara a los jugadores en situaciones desafiantes. Se procuró que los jugadores pudieran disfrutar de una experiencia equilibrada y, al mismo tiempo, emocionante y competitiva.

Planos de nivel y mapas de juego

Para lograr una mejor visualización y comprensión del espacio de juego, se utilizaron tres planos de nivel. Cada uno se usó para planificar dónde se acomodarían las plataformas y columnas, y para determinar dónde se colocarían elementos importantes como naves y ítems, y dónde aparecerían por primera vez los personajes del juego.

Además, se usó un mapa molecular, estos son mapas utilizados para poder planear cual será el recorrido del jugador en un nivel, que puntos del nivel están conectados

y cuáles no. Este mapa ayudó para visualizar cómo sería el recorrido del nivel entre plataformas. De esta manera, se logró una planificación detallada del espacio de juego, lo que permitió optimizar el diseño de los niveles y garantizar una experiencia de juego más fluida y agradable para los jugadores.

A continuación, en la ilustración 2 se muestra el plano del juego que nos da una vista del nivel desde arriba, se dibujó sin perspectiva para ayudar a visualizar la profundidad, y dimensiones de las plataformas dentro del espacio de juego.



Ilustración 2 Plano de nivel del juego “DestroyIt”, donde se muestra el plano del nivel desde arriba.

La ilustración 3 corresponde al plano de sección que muestra el nivel verticalmente, en él se visualizaron las plataformas horizontales y verticales, también ayudó a planificar como se movería el jugador a través del nivel, estos planos también ayudan a planificar el ritmo del nivel, y a visualizar formas alternativas de transición de nivel de un espacio a otro.

El plano en perspectiva (ilustración 4) y axonométrico (ilustración 6) ayudaron a visualizar las plataformas en un espacio tridimensional, para tener más claro cómo iba a ser el salto entre el primer plano de juego y el segundo plano, también fueron útiles para organizar como estaría acomodado el espacio del juego y reacomodar los elementos necesarios para un correcto recorrido de nivel.



Ilustración 3 Plano de sección de juego “DestroyIt”, donde se muestra el plano de nivel de forma vertical.

Igualmente se utilizó un mapa molecular (ilustración 5) para tener claras que zonas del juego correspondían al primer plano y que correspondían al segundo plano. En el mapa molecular se pueden identificar dos tipos de líneas, la continua corresponde a los espacios del juego por donde se pueden pasar directamente sin saltar entre planos y la línea punteada corresponde a las zonas del nivel donde hay que cambiar de planos para acceder a esa zona.

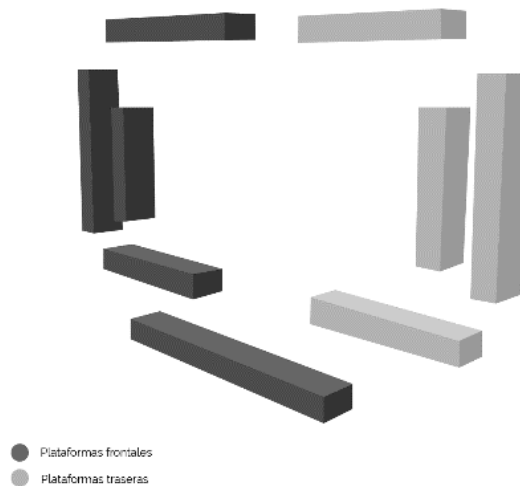


Ilustración 4 Plano de perspectiva del juego “DestroyIt”, donde se muestra el nivel del juego en perspectiva.

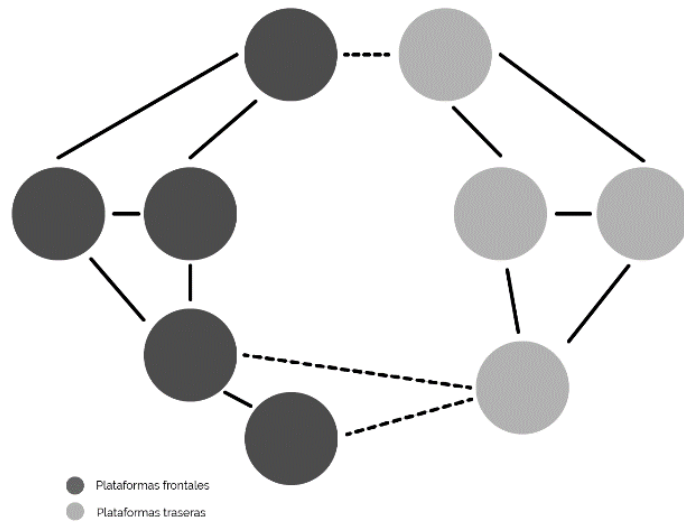


Ilustración 5 Plano molecular del juego donde ponemos ver como se navegan los espacios, la línea continua son los espacios donde el jugador puede transitar directamente y la línea punteada las zonas que requieren un cambio de plano para transitar.

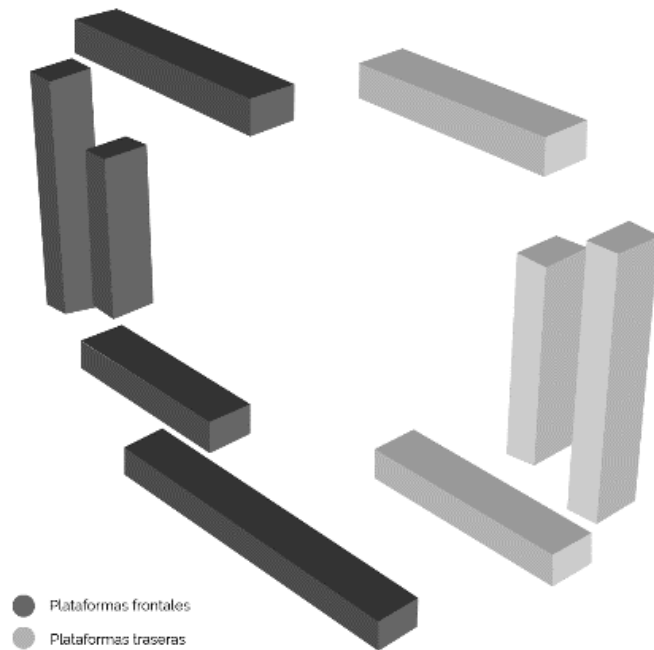


Ilustración 6 Plano axométrico del nivel del juego donde podemos ver con perspectiva y desde arriba el acomodo de las plataformas dentro del nivel del juego.

PROGRAMACIÓN

El juego está desarrollado en el motor de juegos Unity. Los códigos están desarrollados en lenguaje C# usando el entorno de desarrollo Visual Studio. A continuación, se enlistan todas las clases y componentes usadas para el desarrollo del juego.

Clases abstractas/Componentes

1. Física jugador

1.1 Base jugador

2. física objetos

2.1 Base ítems

2.2 Base obstáculos

2.3 Base plataformas

2.4 Base tótems

3. Cámara

4. Revitalizador

Clases derivadas/Composición componentes

1. Base jugador

1.1 Controlador 2D

1.2 Input

1.3 Agarrador de objetos

1.4 Lanzador

1.5 Cambiador de capas

2. Base ítems

2.1 Sumador vida

2.2 Restador de vida

2.3 Aumentador de velocidad

3. Base obstáculos

3.1 Pared

4. Base plataforma

4.1 Plataformas móviles

4.1 Plataforma de hielo.

5. Base tótems

5.1 Tótem jugador

6. Cámara

6.1 Seguimiento de cámara

7. Reaparecer

7.1 Posiciones aleatorias

“DestroyIt” está compuesto de 5 objetos principales que son el jugador, los ítems, las naves, la cámara y el menú.

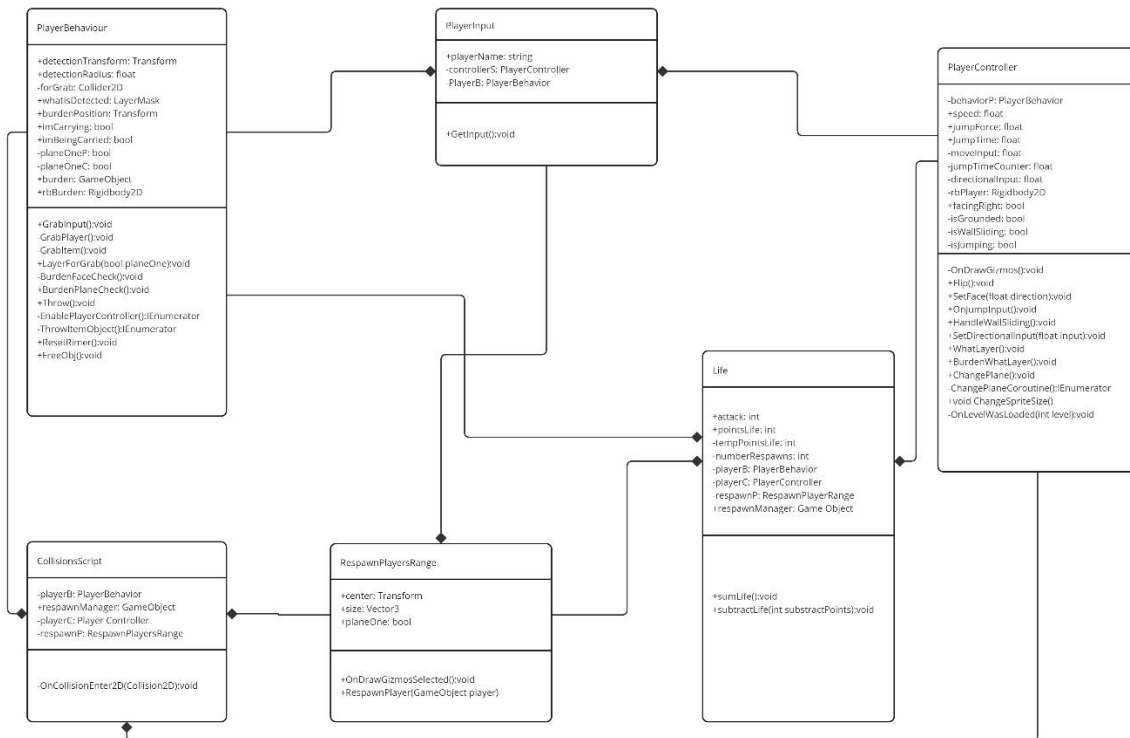


Ilustración 7 Mapa UML del objeto de juego “jugador”

En la ilustración 7 se enlistan las funciones que definen al jugador. Controller que se encarga de acciones básicas del jugador. El Player Behaviour que contiene los comportamientos del jugador durante el juego. Collisions encargado de detectar las diferentes colisiones que experimenta el jugador. Life se encarga de sumar o restar vida al jugador y establecer una vida base además de contabilizar cuantas veces ha

muerto. Respawn Player Range que reaparece a los jugadores de forma aleatoria en un rango de espacio y Player Input que asigna el nombre del player.

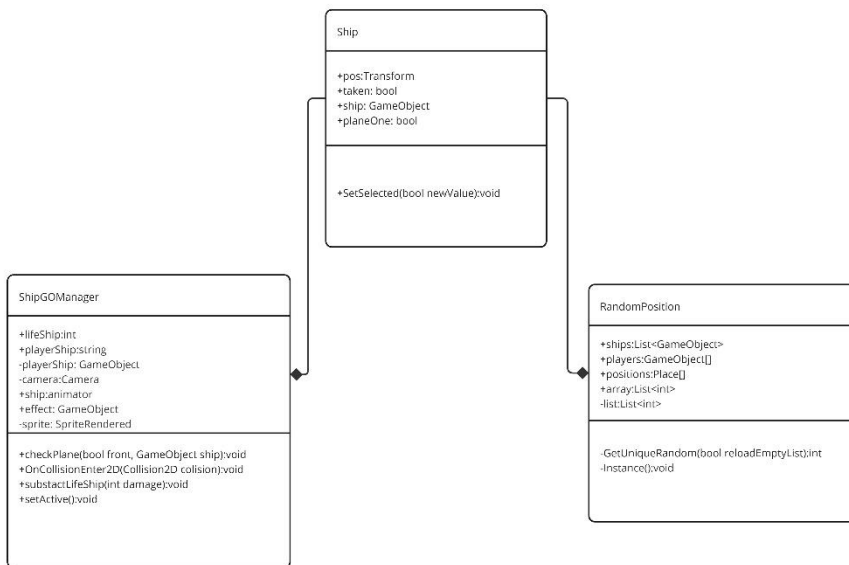


Ilustración 8 Mapa UML de objeto de juego naves.

Las funciones que definen a las naves: Random Position que se encarga de instanciar las naves de los jugadores de forma aleatoria en posiciones predeterminadas. Ship Game Object Manager encargado de monitorear la vida de la nave, el plano en que se encuentra y las colisiones. Ships contiene atributos que ayudan a saber en qué plano se encuentra la nave, en que posición está y si la nave ya pertenece a algún jugador.

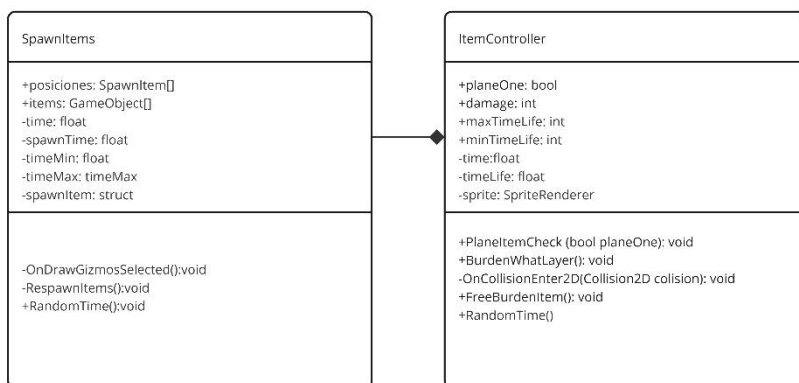


Ilustración 9 Mapa UML de objeto de juego "ítem"

Las funciones que definen los ítems son: ítem Controller que se encarga de manejar la física de los ítems. Spawn ítem que se encarga de instanciar ítems de forma aleatoria en un rango de espacio.

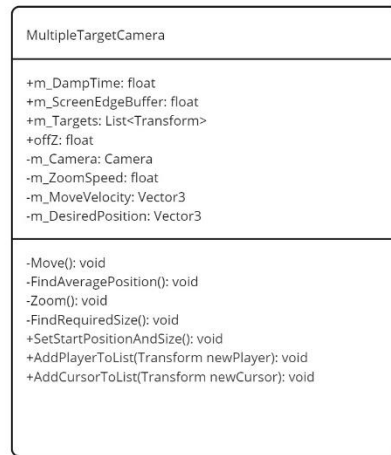


Ilustración 10 Mapa UML de objeto de juego “cámara”

La función que define la cámara es Multiple Target Camera, hace que la cámara siga a los jugadores dependiendo de su estado.

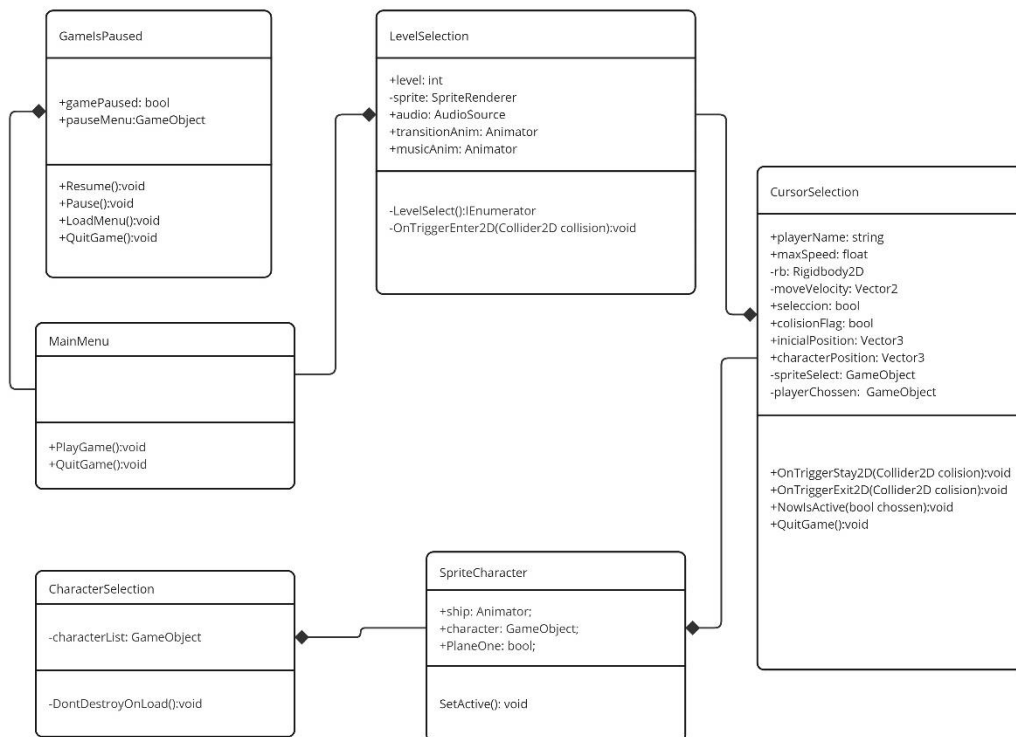


Ilustración 11 Mapa UML de objeto de juego “menú”

Las funciones que definen el menú del juego: Sprite Character que contiene los atributos de cada nave del menú. Character Selection verifica que los jugadores no se destruyan al pasar al nivel seleccionado. Cursor Selection se encarga del movimiento del cursor y selección de personajes. Game is paused que pausa el juego y se encarga de regresar al menú del juego. Level Selection que se encarga de seleccionar el nivel.

ARTE

El diseño del juego “DestroyIt” se planificó con la idea de que una sola persona llevaría a cabo múltiples tareas de diseño, incluyendo la creación de personajes, plataformas, ítems, interfaz de usuario, efectos visuales, animación, entre otras. Se buscó un diseño visualmente atractivo que ahorrara tiempo en su desarrollo, por ello se optó por un diseño general simple y rápido, que no requiriera de múltiples softwares ni de un gran poder computacional para llevarlo a cabo.

Para el diseño general del juego, se utilizaron referencias de otros juegos como Ultimate Chicken Horse, Mario Bros y Hollow Knight, cuyos diseños son agradables a simple vista y no tienen una complejidad alta. La pauta del desarrollo visual del juego fue dada por el diseño de los personajes principales. Al inicio del proyecto, se consideró un estilo 2D tipo cartoon¹¹ con líneas gruesas, colorido con plastas, degradados, sombras y luces duras.

Durante la fase de propuestas de personajes, se realizaron dos propuestas de diseño de personaje, uno en estilo píxel art y otro en estilo cartoon. Se monitoreó el tiempo que llevó hacer un personaje en una pose neutra en cada estilo (línea, plastas, luces y sombras). Durante el monitoreo se concluyó que hacer un personaje estilo cartoon llevaba más tiempo que uno con estilo píxel art.

Dado el número de personajes y animaciones requeridos para el desarrollo del proyecto, se optó por un estilo retro en píxel art, ya que sus procesos de diseño y de color son más simples y rápidos. Algunas de las características del píxel art en “Destroy It” son que los personajes, plataformas, assets y extras no tienen tanto detalle ni texturas complejas. Cuentan con sombras y luces planas y no tienen línea.

Diseño conceptual

El diseño conceptual fue una parte crucial en el proceso de desarrollo de “DestroyIt”, ya que ayudó a plasmar visualmente las ideas y conceptos para el diseño de personajes, escenarios, objetos y todo lo que conformó el mundo del juego, ayudó a establecer un estilo visual coherente y distintivo para el juego, y hacerlo reconocible para el público.

¹¹ El estilo cartoon es un estilo visual en el que los objetos y personajes se representan de una manera simplificada y exagerada, con características como líneas gruesas, colores brillantes y sombras simples o ausentes. Este estilo se asemeja a la estética de los dibujos animados y se utiliza comúnmente en videojuegos y animaciones para crear un ambiente divertido y atractivo.

Con ilustraciones, se pudo visualizar y planificar claramente cómo sería el juego, y se planificó y tomó decisiones sobre el diseño y la implementación de los elementos del juego. Inspiró para nuevas ideas en mecánicas de juego, personajes y escenarios, y ayudó a ahorrar tiempo y recursos.



Ilustración 12 Primer arte conceptual desarrollado para el juego “DestroyIt”.

Durante la etapa de desarrollo del videojuego, se elaboró el arte conceptual tomando en cuenta la mecánica fundamental del juego, la cual se centraba en un salto de un plano de juego delantero a uno trasero. Esta mecánica sirvió de base para idear las demás mecánicas del juego y determinar cuál sería su objetivo, para hacerlo más entretenido y dinámico.

Los primeros conceptos de juego imitaron el estilo de arte que tiene Ultimate Chicken Horse. Este estilo se caracteriza por su apariencia cartoon, sin líneas y con una paleta de colores limitada. En el primer arte conceptual se puede apreciar que la mecánica de cambiar de plano era diferente, ya que el jugador no tenía un botón asignado para realizar esta acción. Se pensó que podría hacerlo a través de cañones o portales, sin embargo, esta idea fue desechada debido a que ralentizaría el ritmo del juego.

En esta primera versión, el objetivo del juego no era destruir naves, el jugador que llegara primero a la bandera era el que ganaría el juego, el nivel del juego es continuo en las orillas de la pantalla. Los diferentes planos se representaban solo con el ancho de las plataformas, siendo las más gruesas las del primer plano y las más delgadas las del segundo. Sin embargo, estas ideas iniciales evolucionaron a medida que se avanzó en el desarrollo del juego.

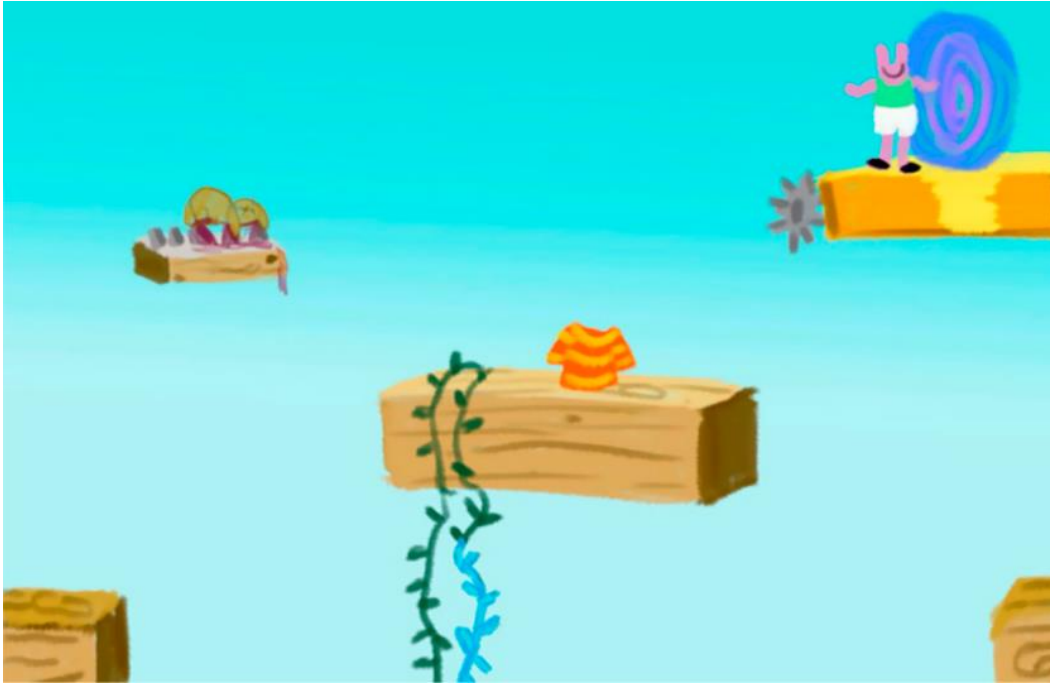


Ilustración 13 Segundo arte conceptual desarrollado para el aspecto visual

Durante el proceso de desarrollo del juego “DestroyIt”, se realizó una revisión del arte conceptual con el objetivo de mejorar la claridad en la distinción entre los planos de juego. En este sentido, se incorporaron plataformas con perspectiva en el segundo arte conceptual, lo cual permitió diferenciar de manera más efectiva el primer plano del segundo plano.

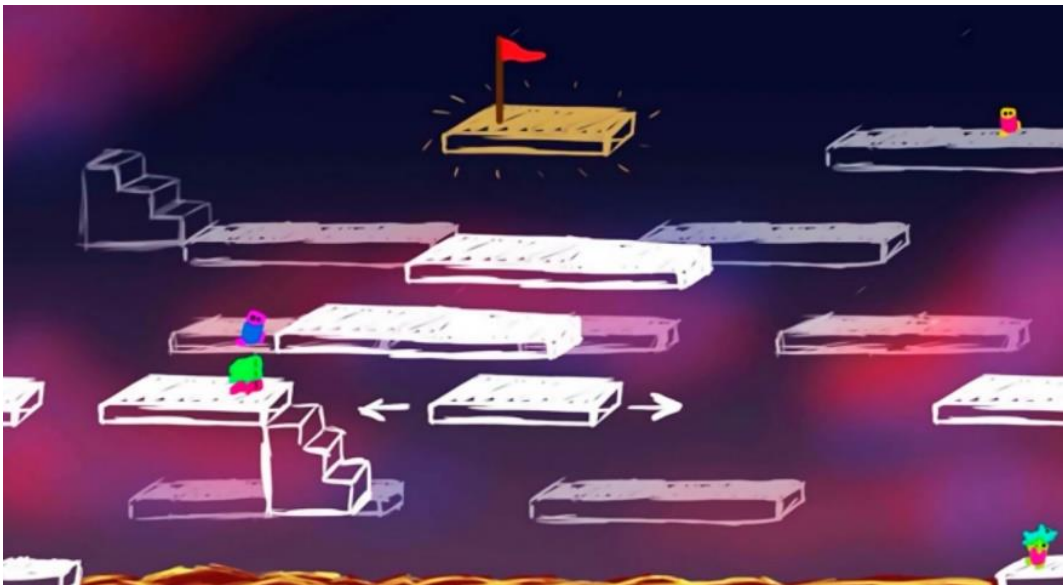


Ilustración 14 Tercer arte conceptual con el desarrollo del concepto de juego y mecánicas.

Cabe mencionar que en este arte conceptual se incluye una prenda de ropa en una de las plataformas, lo que responde a la exploración de una idea de juego donde se buscaba recolectar toda la ropa posible. De acuerdo con esta propuesta, el jugador que llegara a la bandera roja con mayor cantidad de prendas de vestir sería el ganador. Esta idea se descartó en etapas posteriores del desarrollo del juego.

En la imagen 14 podemos apreciar cómo se mantuvo la perspectiva en las plataformas, lo que facilitaría al jugador la distinción entre planos. Además, se agregó desaturación a las plataformas con la finalidad de lograr un mayor contraste entre ambos planos. En este arte conceptual se exploró la posibilidad de que los jugadores pudieran cargar y lanzar a otros como si fueran armas, así como también la idea de utilizarlos como plataformas para acceder a zonas de difícil alcance. Esta idea fue descartada, al igual que la idea del paso de planos con portales o cañones. En esta etapa se tomó en cuenta tener un botón preestablecido para hacer el cambio entre planos. En esta etapa, el objetivo del juego seguía siendo llegar a la bandera roja antes que los demás jugadores.

Diseño de personajes y naves

Durante el proceso de diseño de personajes para “DestroyIt”, se puso un gran énfasis en la generación de siluetas que fueran memorables y fácilmente reconocibles por los jugadores. Esto fue un paso importante para establecer el estilo visual del juego. Se trabajó cuidadosamente para asegurarse de que ninguna silueta se pareciera demasiado a otra, y que cada personaje tuviera un toque único.

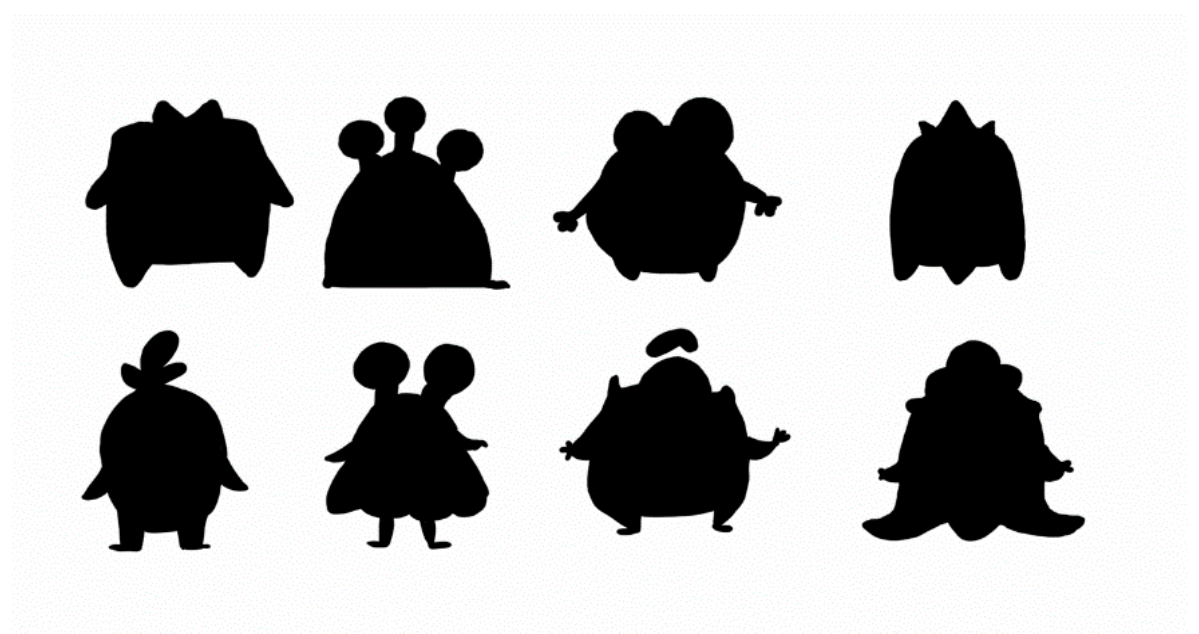


Ilustración 15 Diseño de siluetas de los personajes.

El diseño de personajes fue una parte crucial en el desarrollo de “DestroyIt”, ya que ayudó a establecer el estilo visual y la personalidad del juego, la elección de la paleta de colores y el estilo de dibujo fue importante para mantener una cohesión visual en

el juego. La atención al detalle en la generación de siluetas, elección del estilo de dibujo y paleta de colores fueron factores determinantes en la creación de personajes memorables y únicos, que contribuyeron al desarrollo del juego.

Para la creación de los diseños de personajes, se empleó la herramienta de edición de imágenes Photoshop. Una vez definida esta elección de software, se procedió a realizar los bocetos de los personajes en un lienzo en blanco. Esto se llevó a cabo con la aplicación de trazos negros mediante el uso de un pincel redondo duro, con el fin de lograr siluetas distintivas y memorables para cada uno de los personajes. Los resultados finales de estas siluetas para los personajes del juego pueden apreciarse en la figura 15. Dado que el juego se concibió como un juego multijugador para un rango de 2 a 4 jugadores, se procedió a diseñar un total de 8 siluetas. Sin embargo, finalmente se optó por seleccionar únicamente 6 de estas siluetas para integrarlas como personajes en el juego.

A partir de estas 6 siluetas seleccionadas, se llevaron a cabo bocetos detallados para cada uno de los personajes. Durante esta etapa, se dotó a cada personaje de una personalidad única y se les atribuyeron rasgos distintivos con el objetivo de conferirles singularidad.

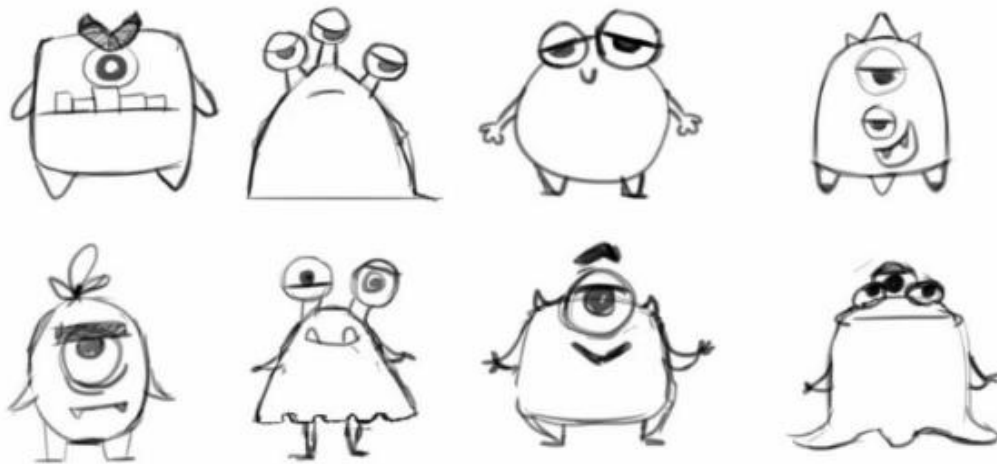


Ilustración 16 Diseño de personajes basado en siluetas.

Se exploraron diferentes opciones para el estilo de dibujo, y se procedió a hacer una prueba de personaje con estilo cartoon parecido al estilo del juego “Hollow Knight”, con colores brillantes y saturados sería el adecuado para el juego. Se hicieron dos propuestas en esta fase: una se hizo con plastita de color, luz y sombra duras y línea del color principal del personaje, y la otra propuesta constó de plastita de color, luz y sombra, pero con una línea negra para delimitar el personaje.



Ilustración 17 Prueba de estilo de personaje para el juego "DestroyIt"

Después de estas pruebas, se evaluó el tiempo que llevaría realizar la limpieza de línea y el color de los personajes, tomando en cuenta que solo una persona se encargaría del arte de los 6 personajes. Se determinó que sería mejor cambiar el estilo artístico del juego a uno que pudiera simplificarse y hacerse en menos tiempo. Por esta razón, se eligió el estilo píxel art, ya que la cantidad de píxeles haría más simple el proceso de animación y color de personajes, plataformas, fondos y objetos.

Una vez tomada la decisión de que el juego tuviera el estilo píxel art, se procedió a hacer pruebas de cual sería la cantidad de píxeles necesarios para el diseño de personajes. La resolución elegida para los personajes y naves fue de 50x50 píxeles. Después se desarrollaron los personajes en escala de grises, para definir los contrastes de color. Una vez que el contraste de color fue adecuado se procedió a desarrollar las paletas de colores para cada uno de ellos. Se tuvo en cuenta que debían ser colores brillantes y saturados para que resaltaran en el nivel y se diferenciaban de los demás elementos del juego.



Ilustración 18 Personajes finales para el juego "DestroyIt"

Para la selección de colores se utilizaron diversas técnicas, como la selección de colores complementarios o la utilización de armonías de color para cada personaje. Se experimentó con diferentes tonalidades y se hicieron pruebas para verificar que los colores se veían correctamente en el juego.

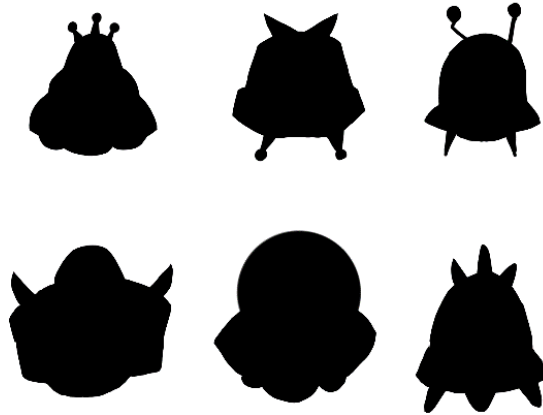


Ilustración 19 Diseño de siluetas de las naves de los personajes del juego.

En el proceso de diseño de “DestroyIt”, la creación de las naves espaciales fue un aspecto fundamental. Para lograr que cada nave fuera fácilmente reconocible y se pudiera asociar rápidamente con el marciano correspondiente, se utilizó como referencia la silueta de cada personaje



Ilustración 20 Diseño final de las naves de los personajes del juego.

Para las naves se siguió un proceso similar al empleado en el diseño de los personajes, comenzando con la creación de las siluetas correspondientes a cada una de estas. Una vez establecidas las siluetas, se procedió a la elaboración de los bocetos y, posteriormente, a la realización de las naves en píxel art. Para asegurar la coherencia visual y la identificación clara de cada nave, se utilizó la misma paleta de colores que la empleada para el marciano al que pertenece. Además, se agregó una textura que ayudó a definir las sombras y mejorar la percepción de profundidad en cada nave.

Diseño de plataformas y elementos complementarios

En diseño de plataformas es fundamental para crear un entorno interactivo en el que el jugador pueda moverse y progresar, permiten al jugador avanzar en el nivel y lograr los objetivos del juego. Además, el diseño de estas puede incluir diversos elementos como obstáculos, enemigos, power ups¹², y otros elementos que pueden influir en la experiencia de movimiento y juego. La creación de un buen diseño de estas también puede afectar el ritmo general del juego y la sensación que produce en los jugadores.



Ilustración 21 Primera propuesta de diseño de plataformas.

En el caso del videojuego en cuestión, se diseñaron dos propuestas de plataformas como tiles: una sin perspectiva con colores vibrantes y fosforescentes, y otra con perspectiva, colores más neutros y similares a los que se podrían encontrar en ladrillos comúnmente. La primera propuesta se pensó para que las plataformas encajaran con el color de los personajes, mientras que la segunda se decidió en base a la premisa del juego, en la que los marcianos llegaron a la tierra y deben salir de ella. Después de realizar pruebas, se determinó que las plataformas con

¹² Un power up, en los videojuegos, es un objeto o elemento que se recolecta para proporcionar beneficios temporales o permanentes al personaje o jugador, como aumentar la salud, la velocidad, el daño, o agregar habilidades especiales para superar desafíos.

perspectiva serían las más útiles para que el jugador entendiera que hay dos planos navegables dentro del juego.

En un inicio se había considerado el acomodo de las plataformas pensando en que el juego sería con pantalla continua como el juego de Snow Bros, tal como se mostró en el arte conceptual del juego. Sin embargo, después de realizar pruebas de nivel, se determinó que lo mejor sería hacer plataformas personalizadas y que fueran por bloques, similar a los niveles en Smash Bros, como se muestra en los planos de nivel. Por ello, se trabajó en el diseño de una nueva propuesta de plataformas, consistente en bloques de tierra con pasto sobre ellos, que también cuentan con perspectiva para dar profundidad al diseño. Para lograr este efecto, se bajó el contraste de las plataformas que irían en el segundo plano, y se diseñaron dos columnas verticales para que los jugadores pudieran subir al nivel superior. Estas columnas también se diseñaron con perspectiva y se bajó el contraste a las del segundo plano.



Ilustración 22 Segunda propuesta de diseño de plataformas

En cuanto al diseño de nivel, se incluyeron algunas plantas que funcionaron como elementos de adorno para enfatizar el ambiente en el que se desarrolla el juego, en el planeta tierra.

Para “DestroyIt” se hicieron diversas pruebas y se diseñaron varias propuestas para un diseño de plataformas efectivo y adecuado a la premisa del juego, y se incluyeron

elementos complementarios que contribuyeron a crear una experiencia atractiva y dinámica.



Ilustración 23 Diseño de plataformas finales para el videojuego "DestroyIt"



Ilustración 24 Diseño de vegetación para videojuego "DestroyIt"

Diseño de fondo

En el proceso de diseño de juegos, la elección del fondo es un elemento importante que debe tomarse con cuidado, ya que puede influir en la experiencia del jugador. Para contextualizar el juego "DestroyIt" en el planeta Tierra, se decidió que el fondo del juego presentaría un paisaje terrestre. El diseño se enfocó en un fondo de montañas, con un cielo azul y luz natural, para hacerlo familiar y agradable para los jugadores. De esta manera, se evitó agregar mecánicas extras que jugaran con un paisaje nocturno. Se eligió una vista de montañas con un cielo azul y luz natural, lo que busca proporcionar una sensación de familiaridad al jugador.

Para la creación del fondo, se utilizó el software de diseño gráfico Photoshop y se buscaron referencias en línea de paisajes reales de montañas y del cielo, así como de píxel art. Se agregaron elementos como montañas en el fondo para enfatizar en donde se localiza el nivel del jugador.

Se utilizaron referencias en línea de paisajes reales de montañas y del cielo, así como algunas referencias de píxel art. Se añadieron elementos como montañas de fondo para enfatizar la ubicación del nivel del jugador y para agregarle reto y un sentimiento de peligro al juego, se decidió que se viera la parte alta de las montañas para sentir peligro de caída. Para asegurarse de que el fondo tuviera los colores correctos, se hicieron revisiones constantes a la escala de grises del fondo, logrando así un contraste correcto entre los elementos más cercanos y lejanos del fondo.



Ilustración 25 Diseño de fondo del videojuego "DestroyIt"

Animaciones y efectos

El proceso de animación puede variar según el tipo de juego y los requisitos específicos del personaje, "DestroyIt" se llevó a cabo mediante la utilización de herramientas específicas como Photoshop, empleando el método de cuadro a cuadro y aplicando los principios de animación para lograr una animación fluida y coherente con el estilo del juego.



Ilustración 26 Animación de ciclo de caminata para un personaje del videojuego.

Para realizar las animaciones se utilizaron las plastas de color de los personajes y se fueron animando los elementos según los requerimientos de cada tipo de personaje. Los ciclos de caminata de los jugadores se hicieron con entre cinco y seis fotogramas, cada uno es personalizado dependiendo del personaje. Los personajes ocuparon algunos principios de animación entre ellos squash y stretch, exageración, arcos, staggging.

Otro recurso para animar fueron las explosiones, estas aparecen en el juego cuando el jugador selecciona su nave espacial y cuando se destruye alguna nave durante el juego explosiones son del color principal de cada nave y personaje.



Ilustración 27 Animación de explosión.

Para que fuera más fácil el proceso de crear el personaje y agregarle componentes como animaciones, partículas y códigos fue necesario crear un prefab¹³, en videojuegos es un objeto o conjunto de objetos que ha sido preconfigurado y se puede reutilizar en múltiples instancias de un juego sin tener que recrearlos desde

¹³ Un prefab es esencialmente una plantilla que contiene uno o varios objetos, como modelos 3D, luces, cámaras, scripts y otros componentes. Puedes crear un prefab a partir de cualquier conjunto de objetos en tu escena y luego reutilizarlo tantas veces como desees en diferentes partes de tu juego.

cero cada vez. Es una forma de optimizar el proceso de desarrollo y reducir el tiempo necesario para crear contenido de juego repetitivo.

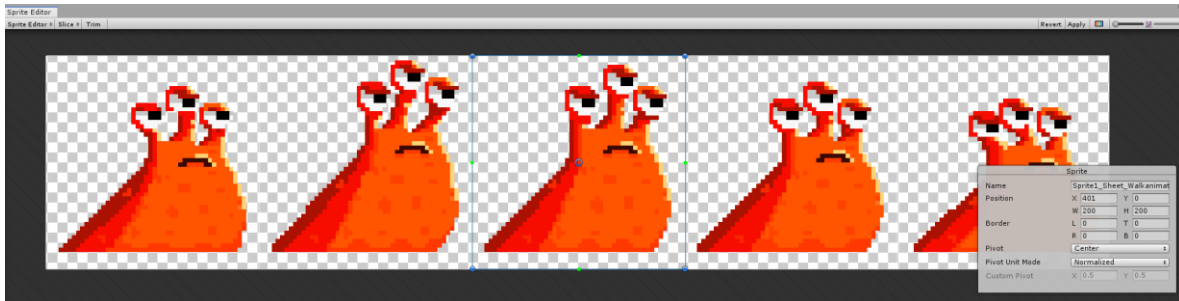


Ilustración 28 Se muestra cómo luce el Sprite sheet editor en Unity.

Luego, se agregó un componente Animator al prefab de cada personaje que se creó, el cual se encargó de controlar todas las animaciones de los modelos.

Para agregar las animaciones a Unity se utilizó la máquina de estados, que es una herramienta útil para controlar el comportamiento de un objeto en el juego. Se puede definir como un sistema que permite a un objeto o personaje cambiar su comportamiento en función de su estado actual y dependiendo de los eventos que ocurren en el juego. Las transiciones de las animaciones se configuraron la máquina de estados. Finalmente, se asignó la máquina de estados al modelo en Unity para que las animaciones se reproduzcan correctamente durante el juego. De esta manera, se pudieron crear animaciones en Unity para los personajes y explosiones del juego.

Unity tiene un sistema de partículas, donde se pueden crear efectos de partículas personalizados utilizando gráficos personalizados y texturas. Una vez que se configuraron las propiedades del sistema de partículas, se agregaron a un objeto en la escena.

Las partículas usadas en "Destroy It" tienen el aspecto de pixeles explotando, esto con el objetivo de que vayan acorde a la estética visual del juego. Cada efecto de partículas es diferente según el personaje, así es más evidente qué jugador ha sufrido daño.

RESULTADOS

En el presente capítulo, se exponen los resultados obtenidos durante el proceso de desarrollo y evaluación del videojuego multijugador de plataformas 2D "DestroyIt", el cual fue creado mediante el empleo del motor de juegos Unity y el lenguaje de programación C#.

Se realizó una encuesta de 12 preguntas a 17 personas que probaron el juego, para verificar si se cumplieron los objetivos en ella podemos ver los siguientes resultados

Se preguntó a los encuestados si habían jugado videojuegos antes, esto para analizar si quienes se interesaron en hacer la prueba del videojuego tenían experiencia en los videojuegos.

Los resultados arrojaron que el 94.1% de los encuestados habían jugado videojuegos antes y solo el 5.9% no habían jugado antes.

¿Has jugado videojuegos antes?

17 respuestas

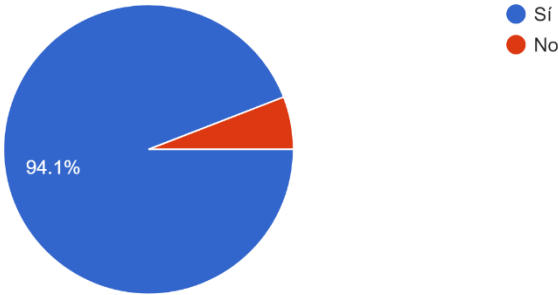


Ilustración 29 Pregunta 1 de encuesta

Se preguntó con qué frecuencia jugaban videojuegos en la semana y los resultados arrojaron que la mayoría de los encuestados juegan frecuentemente con 52.9%, casualmente 29.4%, muy pocas veces solo el 11.8%, sólo el 5.9% no juega, lo que significa que la mayoría de los encuestados tienen experiencia con los videojuegos.

¿Qué tanto juegas videojuegos en la semana?

17 respuestas

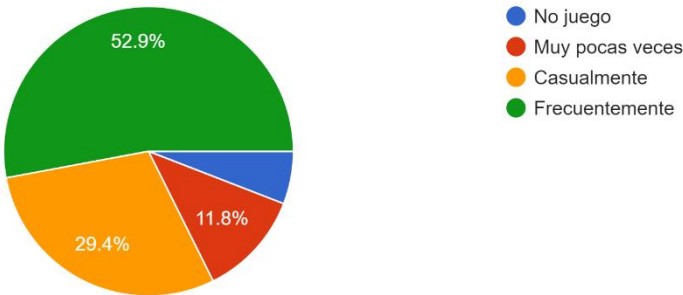


Ilustración 30 Pregunta 2 de encuesta

Se preguntó a los encuestados si les había parecido bueno el movimiento del jugador. Al 100% de los encuestados les pareció bueno, lo que significa que para jugadores novatos, casuales y frecuentes el movimiento fue optimo.

¿Te pareció bueno el movimiento del jugador?

17 respuestas

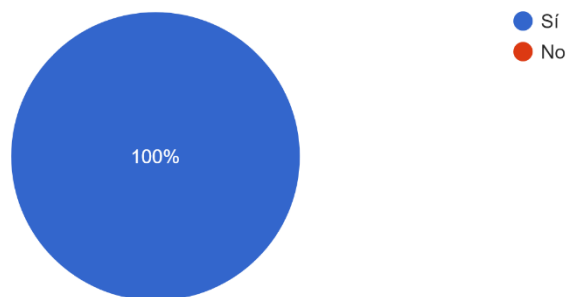


Ilustración 31 Pregunta 3 de encuesta

Se preguntó en la encuesta si los jugadores habían entendido la mecánica de saltar entre planos y los resultados arrojaron que el 82% entendió la mecánica mientras el 17% no logro entenderla. Esto quiere decir que más de la mitad de los jugadores entendieron la mecánica.

¿Entendiste la mecánica de saltar entre planos?

17 respuestas

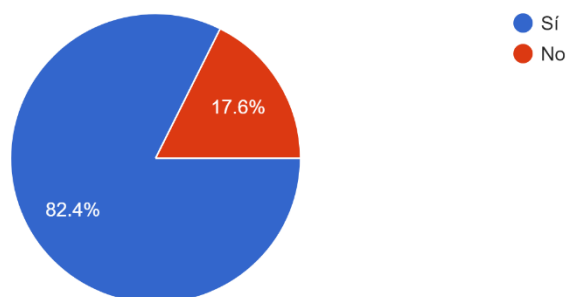


Ilustración 32 Pregunta 4 de encuesta

También se les pregunto si les había gustado la mecánica de salto entre planos, los resultados arrojaron que al 94.1% sí y al 5.9 % no le gustó. Esto quiere decir que a pesar de que algunos de los jugadores no comprendieron la mecánica aun así les pareció buena.

¿Te gustó la mecánica de saltar entre planos?

17 respuestas

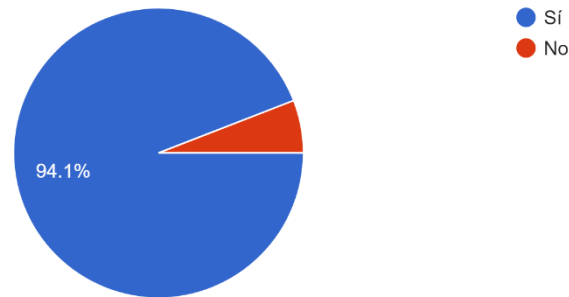


Ilustración 33 Pregunta 5 de encuesta

Se preguntó a los jugadores cuantas veces habían visto la mecánica de salto entre planos en un videojuego. Esto con el objetivo de saber que tan familiarizados estaban con dicha mecánica.

Los resultados arrojaron que el 52.9% la habían visto pocas veces el 35.3% muy pocas veces mientras que solo el 11.8% la habían visto frecuentemente. Con esta información podemos verificar que dentro del grupo de personas que probaron el juego ni siquiera aquellos que juegan con frecuencia videojuegos se han topado con esta mecánica más de 20 veces.

¿Cuantas veces has visto la mecánica de salto entre planos en un videojuego?

17 respuestas

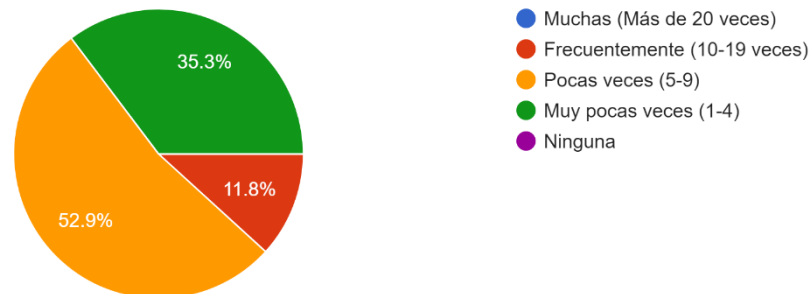


Ilustración 34 Pregunta 6 de encuesta

Uno de los objetivos del proyecto fue hacer un videojuego divertido, los datos de la encuesta arrojaron que el juego fue divertido para el 100% de los jugadores, así que se puede concluir que el juego logro el objetivo de ser divertido.

¿Te pareció divertido el juego?

17 respuestas

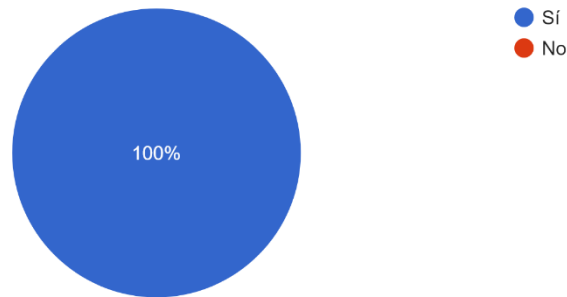


Ilustración 35 Pregunta 7 de encuesta

Se preguntó a los jugadores si hubo risas mientras probaron el juego. Esto con el objetivo de monitorear si el juego es humorístico. Los resultados arrojaron que el 94.1% de los jugadores escucho risas durante sus partidas y el 5.9% no. Con esto podemos asegurar que el juego la mayoría de las veces si cumple su función de divertir y generar risas entre los jugadores.

¿Hubo risas mientras probaste el juego?

17 respuestas

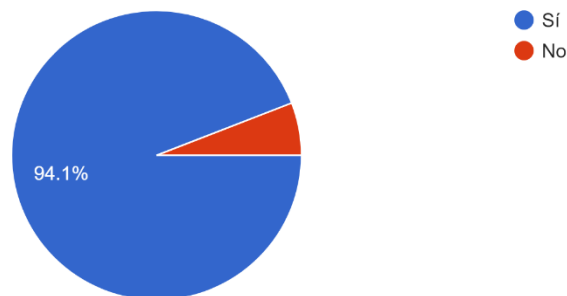


Ilustración 36 pregunta 8 de encuesta

Se les preguntó a los jugadores si sintieron que las partidas fueron repetitivas, esto con el objetivo de saber la re-jugabilidad del videojuego, las estadísticas dieron como resultado que el 52.9% consideraron que el juego no es repetitivo pero el 47.1% consideraron que las partidas si fueron repetitivas, de esto podemos concluir que es necesario trabajar a futuro de mejor manera la re-jugabilidad implementando nuevas mecánicas, ítems y mapas.

¿Sentiste que las partidas fueron repetitivas?

17 respuestas

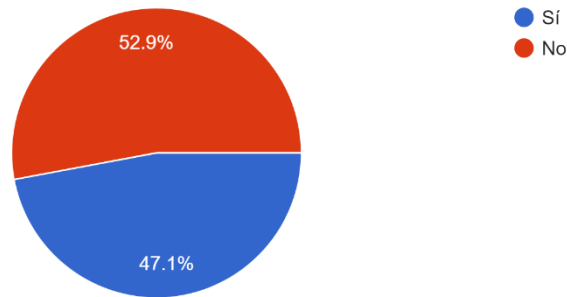


Ilustración 37 Pregunta 9 de encuesta

Uno de los objetivos del juego era generar un arte que fuera de la mano con la temática del juego. Los resultados arrojaron que al 100% de los jugadores les pareció que el arte estuvo de la mano con la temática del juego.

¿El diseño artístico te pareció acorde a la temática de juego?

17 respuestas

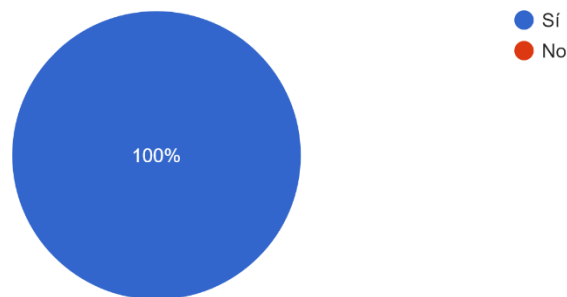


Ilustración 38 pregunta 10 de encuesta

Se agrego una pregunta sobre si el juego fue divertido al ganar y perder. Ya que uno de los objetivos del juego era un juego juguete, donde sin importar si se ganaba o perdía seguiría siendo divertido. El 76.5% consideró divertido jugar “Destrylt” a pesar de perder, así que en base a esos datos podemos afirmar que se logró ese objetivo.

¿ El juego te pareció divertido al ganar y al perder?

17 respuestas

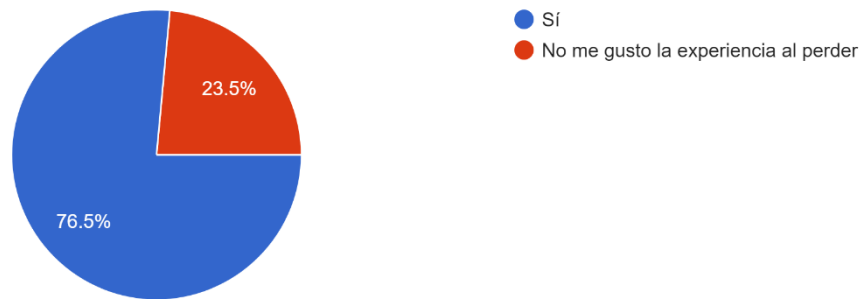


Ilustración 39 Pregunta 11 de Encuesta

Al final de la encuesta se agregaron opciones de adjetivos, para que el jugador pudiera describir el juego con algunas palabras. Esto arroja que el juego es considerado divertido, adictivo, competitivo, difícil y entretenido.

DestroyIt es:

17 respuestas

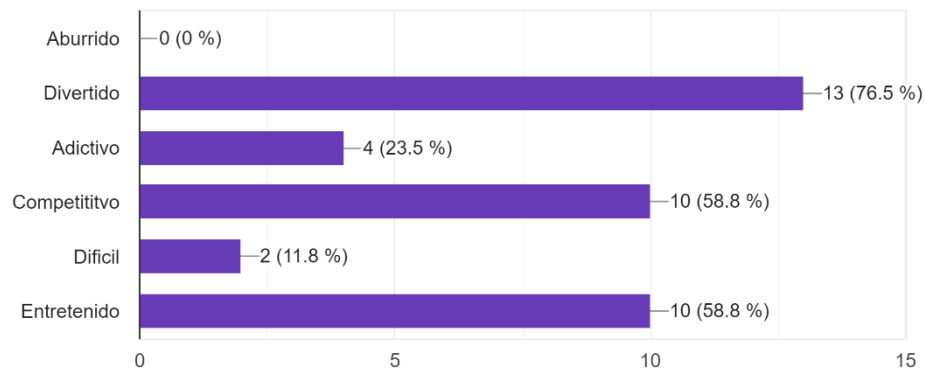


Ilustración 40 Pregunta 12 de encuesta

Durante el proceso de creación, como se puede observar en la imagen 29, se realizaron diversas pruebas del videojuego en la Feria de Proyectos de la BUAP de 2019 (Fepro), un concurso de proyectos de software, se exhibió el juego y los asistentes y compañeros que lo probaron destacaron su enfoque divertido y competitivo, así como su adecuado nivel de desafío. Asimismo, se contó con la

opinión de especialistas en la industria de videojuegos para su evaluación imagen 30.



Ilustración 41 Test del juego con asistentes a la Fepro 2019

En la Fepro de 2019, la competición de proyectos organizada por la facultad de ciencias de la computación, el videojuego fue presentado en la categoría de gráficos por computadora. Durante la exposición, se explicó la mecánica del juego y se detallaron aspectos técnicos y artísticos relevantes. El jurado, compuesto por expertos en el desarrollo de videojuegos, evaluó el videojuego de manera minuciosa y determinó que "DestroyIt" merecía el primer lugar en la feria de proyectos.



Ilustración 42 Ilustración 30 Exposición de videojuego en la Fepro

Se recibieron premios por el primer lugar en la competición, incluyendo un libro de desarrollo en Java, un iPad, un disco duro, una memoria RAM y un año de licencia de Unity.

Adicionalmente José Luis uno de los miembros fundadores de Ficha Studio y juez en la Fepro 2019 dio su opinión sobre el juego: "Visualmente muy bonito, cuando lo empecé a jugar me sorprendió el cambio de planos, fue muy fácil entender, pero me tomó por sorpresa. Jugarlo fue muy divertido, muy competitivo y eso me gusta."

En conclusión, el proceso de desarrollo y evaluación del videojuego multijugador de plataformas 2D "DestroyIt" fue exitoso y permitió cumplir con los objetivos previamente establecidos, obteniendo además el reconocimiento de la comunidad académica y especializada en la industria de videojuegos.

Se logró desarrollar un prototipo del videojuego "DestroyIt", cumplió satisfactoriamente con el objetivo de ser un juego divertido. El juego se basó en una mecánica base para su desarrollo y se implementó a través del uso de Unity y C#.

El arte del videojuego se concibió y diseñó considerando las características y necesidades del juego, incluyendo plataformas, fondos, personajes, animaciones, partículas y VFX, para crear una estética coherente y atractiva.

CONCLUSIÓN

El juego se desarrolló exitosamente, cumpliendo los objetivos propuestos y evidenciándolos con pruebas de juego realizadas con compañeros de escuela, amigos, familiares y visitantes de la feria de proyectos. Desde cero, se ideó una mecánica principal, se creó un prototipo con cajas simples en Unity y se elaboró el arte conceptual para visualizar el aspecto final del juego.

El diseño del juego se llevó a cabo tras la lectura de diferentes libros y documentación sobre diseño de juegos y niveles, lo que permitió crear un juego consistente, competitivo y divertido. Cada etapa del proceso fue debidamente documentada, incluyendo las diferentes versiones del prototipo y del arte del juego.

Tras el game testing realizado en la Fepro con los asistentes, jurado y docentes especializados, se llegó a la conclusión de que el juego es funcional y divertido. Esta afirmación se fundamenta en la reacción positiva de los jugadores al probarlo, demostrando que es posible crear un juego competitivo de plataformas basándose únicamente en una mecánica principal. Se aplicaron todos los conocimientos adquiridos durante la carrera de Arte Digital, incluyendo el diseño de juegos, diseño de niveles, programación, ilustración y animación.

Durante el desarrollo del proyecto, una de las principales limitaciones fue el tamaño del equipo, ya que se trató de un proyecto realizado únicamente por una persona. Esto implicó un proceso más lento, ya que no se podían llevar a cabo varios procesos de manera simultánea, como, por ejemplo, el desarrollo del arte conceptual mientras otro miembro del equipo trabaja en el diseño de niveles. La organización y el cronograma se planificaron cuidadosamente para que cada proceso pudiera realizarse sistemáticamente por una sola persona.

Se espera que “DestroyIt” pueda funcionar como un prototipo base para un juego planeado con más niveles y personajes, así como para ser utilizado como portafolio de diseño, programación y arte de videojuegos en el futuro. El juego representó un reto tanto artístico como técnico, ya que se exploró una mecánica poco común en el desarrollo de juegos, como lo es el cambio de plano en un ambiente 2D. Para lograrlo, se ideó una solución innovadora, utilizando capas y cambios de plano en el motor de juegos Unity, lo cual implicó meses de investigación y trabajo para poder resolver la mecánica principal. Cuando se tuvo el prototipo funcional de cajas, se diseñó el arte de manera eficiente para que lo hiciera una sola persona.

ANEXOS

Anexo A. capturas de pantalla de códigos esenciales para el desarrollo del juego.

```
public void WhatLayer()
{
    if (planeOne)
    {
        gameObject.layer = 10;
        whatIsGround.value = 1 << 8;
        // whatIsGround.value = 1 << 10;
        Physics2D.IgnoreLayerCollision(10, 9);
        Physics2D.IgnoreLayerCollision(10, 8, false);
        Physics2D.IgnoreLayerCollision(10, 11);
        sprite.sortingOrder = 3;
    }

    else
    {
        gameObject.layer = 11;
        whatIsGround.value = 1 << 9;
        //whatIsGround.value = 1 << 11;
        Physics2D.IgnoreLayerCollision(11, 8);
        Physics2D.IgnoreLayerCollision(11, 9, false);
        Physics2D.IgnoreLayerCollision(11, 10);
        sprite.sortingOrder = -3;
    }
}
```

Ilustración 43 Script que detecta en que capa dentro del juego se encuentra el personaje, y la cambia dependiendo de en qué plano se encuentre el jugador.

```

public void ChangePlane()
{
    if ((!changingPlane && isGrounded) || (!changingPlane && !isGrounded && isWallSliding))
    {
        SoundManagerScript.PlaySound("ChangeP");
        StartCoroutine("ChangePlaneCoroutine");
    }
}
private bool changingPlane = false;
IEnumerator ChangePlaneCoroutine()
{
    changingPlane = true; // estoy cambiando de plano

    isJumping = true;
    jumpTimeCounter = jumpTime;
    rbPlayer.velocity = Vector2.up * jumpForce*2.2f;

    yield return new WaitForSeconds(.5f);

    planeOne = !planeOne;

    ChangeSpriteSize();

    //si esta cargando un objeto el objeto tambien tiene que cambiar
    // la capa que pisa y la que agarra
    if (behaviorP.carga)
    {
        if (behaviorP.carga.tag == "Player")
        {
            behaviorP.carga.GetComponent<PlayerController>().planeOne = planeOne;
            behaviorP.carga.GetComponent<PlayerController>().BurdenWhatLayer();
            behaviorP.carga.GetComponent<PlayerBehavior>().LayerForGrab(planeOne);
        }
        else if (behaviorP.carga.tag == "Item")
        {
            behaviorP.carga.GetComponent<ItemController>().planeOne = planeOne;
            behaviorP.carga.GetComponent<ItemController>().BurdenWhatLayer();
        }
    }
}

```

Ilustración 44 Script que cambia el tamaño del personaje y avisa el cambio de plano, y también llama a una función para que cambie de plano lo que el personaje está cargando.

Anexo B. exposición de juego y premiación en la Fepra 2019.



Ilustración 45 Fotografía de la exposición del juego ante del jugado de la Fepra 2019



Ilustración 46 Entrega de premio al juego “DestroyIt”

Anexo C. capturas de pantalla de gameplay de “DestroyIt”

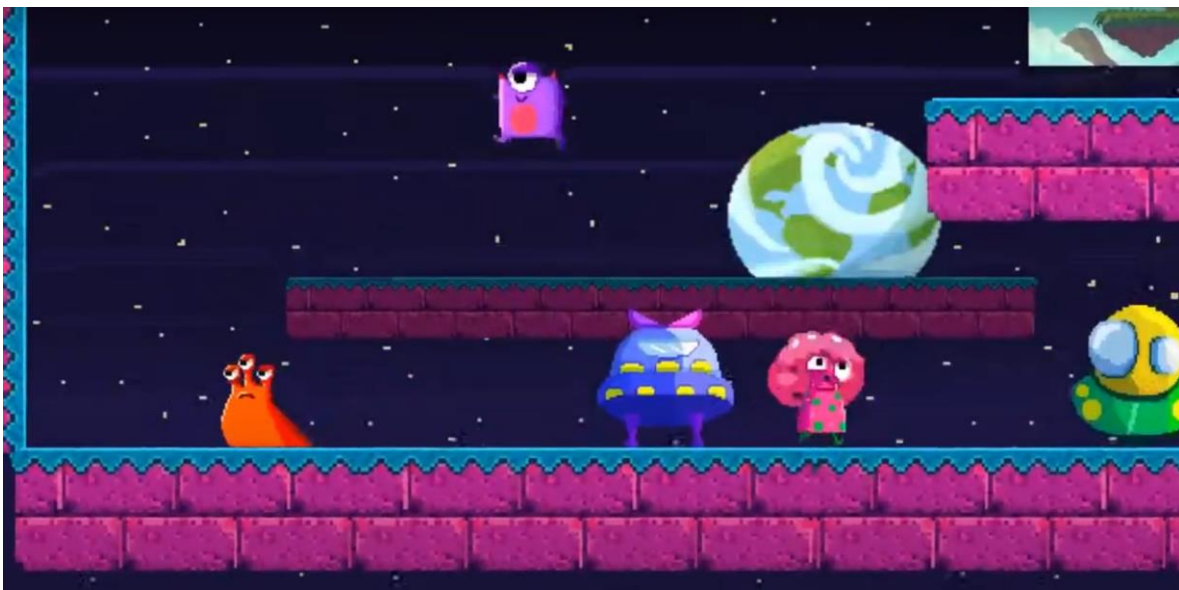


Ilustración 47 Menú de juego de “DestroyIt”



Ilustración 48 Juego "Destroyit"

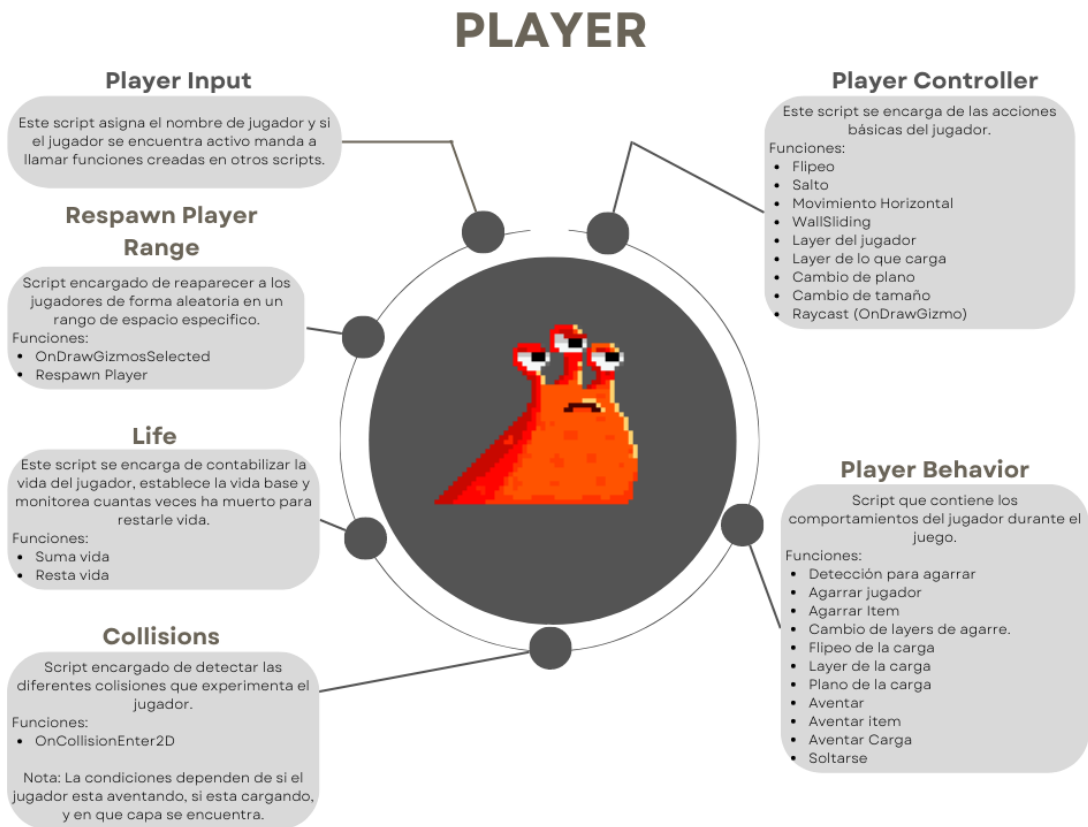


Ilustración 49 Mapa mental del objeto de juego "player"

ITEMS



Ilustración 50 Mapa mental de objeto de juego "items"

NAVES

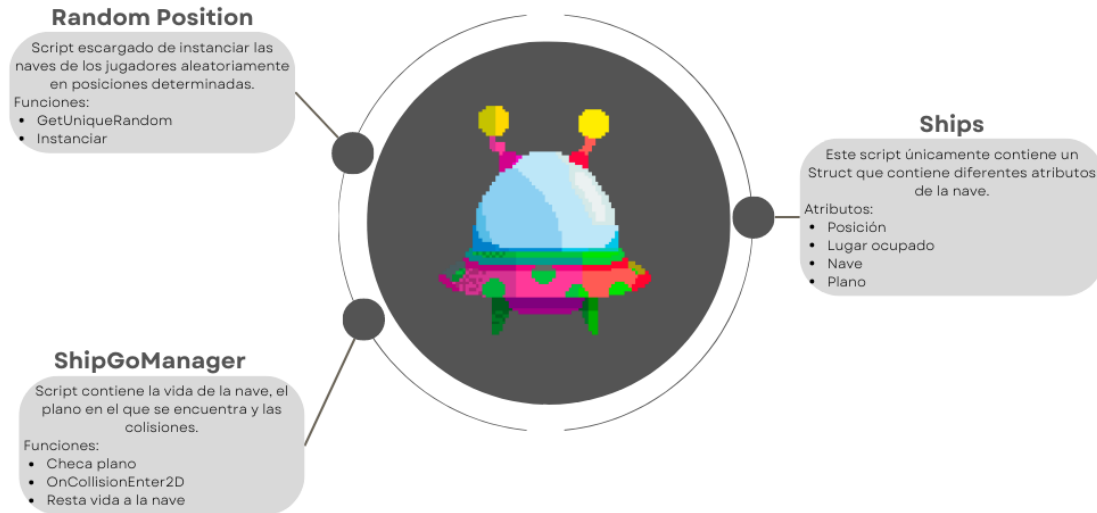


Ilustración 51 Mapa mental de objeto de juego "naves"

MENÚ

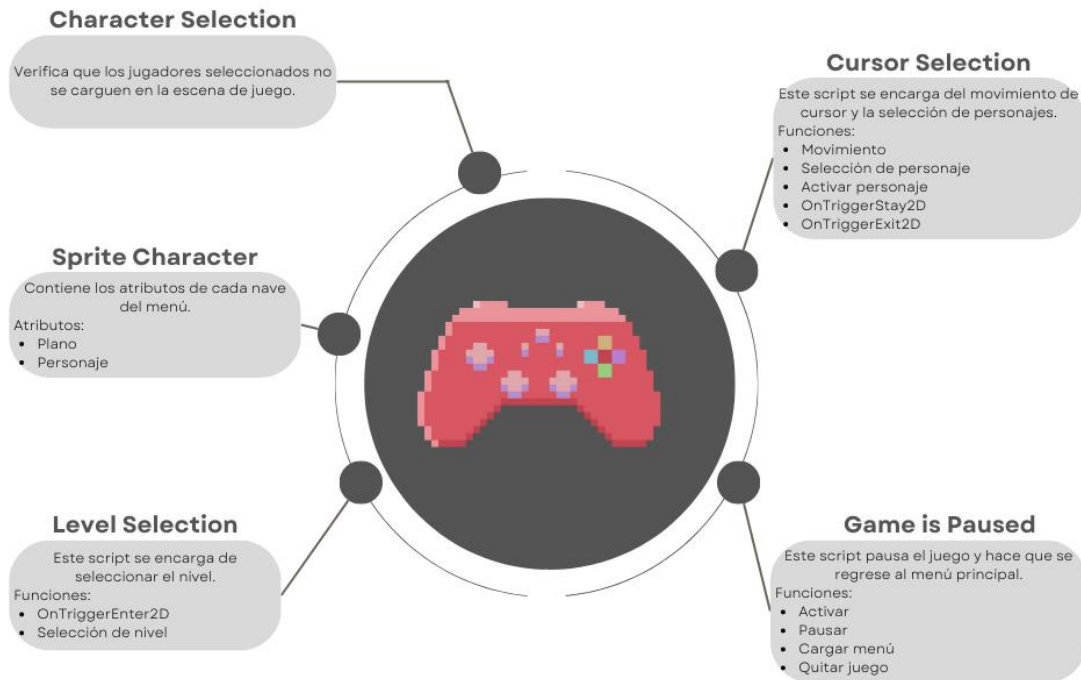


Ilustración 52 Mapa mental de objeto de juego "menú"

CÁMARA

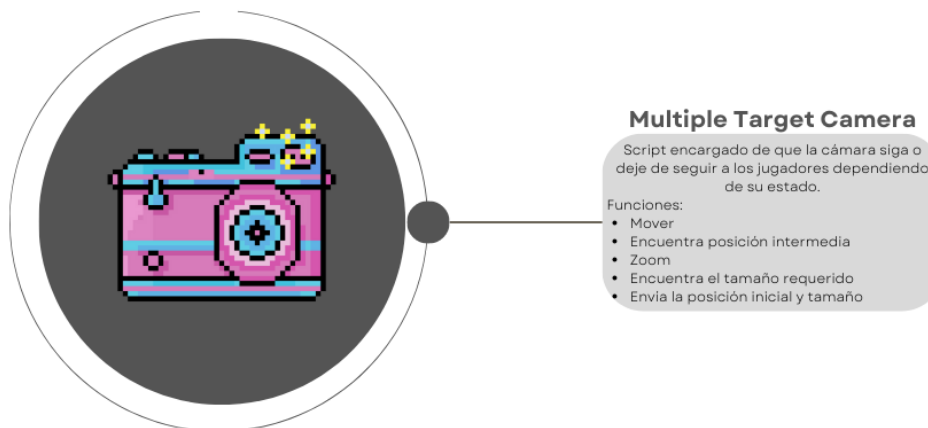


Ilustración 53 Mapa mental de objeto de juego "cámara"

ÍNDICE DE IMÁGENES

| | |
|--|----|
| Ilustración 1 Muestra una captura del juego con el recorrido del nivel, los espacios, tipo de narrativa y relaciones espaciales. | 35 |
| Ilustración 2 Plano de nivel del juego "DestroyIt", donde se muestra el plano del nivel desde arriba. | 39 |
| Ilustración 3 Plano de sección de juego "DestroyIt", donde se muestra el plano de nivel de forma vertical. | 40 |
| Ilustración 4 Plano de perspectiva del juego "Destroy it", donde se muestra el nivel del juego en perspectiva. | 40 |
| Ilustración 5 Plano molecular del juego donde ponemos ver como se navegan los espacios, la línea continua son los espacios donde el jugador puede transitar directamente y la línea punteada las zonas que requieren un cambio de plano para transitar. | 41 |
| Ilustración 6 Plano axométrico del nivel del juego donde podemos ver con perspectiva y desde arriba el acomodo de las plataformas dentro del nivel del juego. | 41 |
| Ilustración 7 Mapa UML del objeto de juego "jugador" | 43 |
| Ilustración 8 Mapa UML de objeto de juego naves. | 44 |
| Ilustración 9 Mapa UML de objeto de juego "ítem" | 44 |
| Ilustración 10 Mapa UML de objeto de juego "cámara" | 45 |
| Ilustración 11 Mapa UML de objeto de juego "menú" | 45 |
| Ilustración 12 Primer arte conceptual desarrollado para el juego "Destroy It". | 47 |
| Ilustración 13 Segundo arte conceptual desarrollado para el aspecto visual. | 48 |
| Ilustración 14 Tercer arte conceptual con el desarrollo del concepto de juego y mecánicas. | 48 |
| Ilustración 15 Diseño de siluetas de los personajes. | 49 |
| Ilustración 16 Diseño de personajes basado en siluetas. | 50 |
| Ilustración 17 Prueba de estilo de personaje para el juego "DestroyIt" | 51 |
| Ilustración 18 Personajes finales para el juego "DestroyIt" | 51 |
| Ilustración 19 Diseño de siluetas de las naves de los personajes del juego. | 52 |
| Ilustración 20 Diseño final de las naves de los personajes del juego. | 52 |
| Ilustración 21 Primera propuesta de diseño de plataformas. | 53 |
| Ilustración 22 Segunda propuesta de diseño de plataformas | 54 |
| Ilustración 23 Diseño de plataformas finales para el videojuego "DestroyIt" | 55 |
| Ilustración 24 Diseño de vegetación para videojuego "DestroyIt" | 55 |
| Ilustración 25 Diseño de fondo del videojuego "DestroyIt" | 56 |
| Ilustración 26 Animación de ciclo de caminata para un personaje del videojuego. | 57 |
| Ilustración 27 Animación de explosión. | 57 |
| Ilustración 28 Se muestra cómo luce el Sprite sheet editor en Unity. | 58 |
| Ilustración 29 Pregunta 1 de encuesta. | 59 |
| Ilustración 30 Pregunta 2 de encuesta. | 59 |
| Ilustración 31 Pregunta 3 de encuesta. | 60 |
| Ilustración 32 Pregunta 4 de encuesta. | 60 |
| Ilustración 33 Pregunta 5 de encuesta. | 61 |
| Ilustración 34 Pregunta 6 de encuesta. | 61 |
| Ilustración 35 Pregunta 7 de encuesta. | 62 |
| Ilustración 36 pregunta 8 de encuesta. | 62 |

| | |
|---|----|
| Ilustración 37 Pregunta 9 de encuesta..... | 63 |
| Ilustración 38 pregunta 10 de encuesta..... | 63 |
| Ilustración 39 Pregunta 11 de Encuesta..... | 64 |
| Ilustración 40 Pregunta 12 de encuesta..... | 64 |
| Ilustración 41 Test del juego con asistentes a la Fepro 2019..... | 65 |
| Ilustración 42 Ilustración 30 Exposición de videojuego en la Fepro | 66 |
| Ilustración 43 Script que detecta en que capa dentro del juego se encuentra el personaje, y la cambia dependiendo de en qué plano se encuentre el jugador. | 68 |
| Ilustración 44 Script que cambia el tamaño del personaje y avisa el cambio de plano, y también llama a una función para que cambie de plano lo que el personaje está cargando. | 69 |
| Ilustración 45 Fotografía de la exposición del juego ante del jugado de la Fepro 2019..... | 69 |
| Ilustración 46 Entrega de premio al juego "DestroyIt" | 70 |
| Ilustración 47 Menú de juego de "DestroyIt" | 70 |
| Ilustración 48 Juego "DestroyIt" | 71 |
| Ilustración 49 Mapa mental del objeto de juego "player" | 71 |
| Ilustración 50 Mapa mental de objeto de juego "ítems" | 72 |
| Ilustración 51 Mapa mental de objeto de juego "naves" | 72 |
| Ilustración 52 Mapa mental de objeto de juego "menú" | 73 |
| Ilustración 53 Mapa mental de objeto de juego "cámara" | 73 |

BIBLIOGRAFIA

- Adobe. (Noviembre de 2022). *101 combinaciones de colores atractivas*. Obtenido de Adobe: <https://www.adobe.com/es/express/learn/blog/101-brilliant-color-combinations>
- Alto nivel. (Mayo de 2011). *¿Qué empresas desarrollan videojuegos en México?* Obtenido de Alto nivel: <https://www.altonivel.com.mx/tecnologia/10521-que-empresas-desarrollan-videojuegos--en-mexico/>
- Asociación Mexicana de Desarrolladores de Videojuegos. (s.f.). Obtenido de Asociación Mexicana de Videojuegos: <http://mexicogames.org/>
- Azzi, M. (2019). *Pixel Logic: A guide to pixel art*.
- Bergström, K. B. (2010). Exploring aesthetical gameplay design patterns. *14th International Academic MindTrek Conference: Envisioning Future Media Environments*, 17-24.
- Contreras, R. (Enero de 2018). *La Industria del videojuego en*. Obtenido de Invdes: <https://invdes.com.mx/los-investigadores/la-industria-del-videojuego-mexico/>
- Despain, W. (2009). *Writing for Video Game Genres: From Fps to RPG*. A K PETERS.
- Eiseman, L. (2018). *Armonía cromática: Edición Pantone*. Blumer.
- Hervieux, M. (2015). *Learning Pixel Art*.
- Hunicke, R., LeBlanc, M., & Zubek, R. (2004). MDA: A Formal Approach to Game Design and Game Research. *Proceedings of the AAAI Workshop on Challenges in Game AI*, 4.
- Rogers, S. (2010). *Level Up!: The Guide to Great Video Game Design*. John Wiley & Sons Inc.

- Schell, J. (2014). *The Art of Game Design: A Book of Lenses, Second Edition*. A K PETERS.
- Stefyn, N. (27 de octubre de 2022). *What is Game Design and How to Become a Game Designer*. Obtenido de CG Spectrum: <https://www.cgspectrum.com/blog/what-is-game-design>
- Thomas, F., & Johnston, O. (1995). *The Illusion of life, Disney Animation*. Hyperion.
- Thomson, K. (Agosto de 2019). *The Philosophy of Digital*. Obtenido de Stanford Encyclopedia of Philosophy: <https://plato.stanford.edu/entries/digital-art/#WhaDigArt>
- Totten, C. (2017). *Level Design: Processes and Experiences*. A K PETERS.
- Totten, C. (2019). *An Architectural Approach to Level Design: Second Edition*. A K PETERS.
- Ultimate Chicken Horse on Steam*. (s.f.). Obtenido de STEAM: https://store.steampowered.com/app/386940/Ultimate_Chicken_Horse/
- Williams, R. (2012). *The animator's survival kit: A manual of methods, principles and formulas for classical, computer, games, stop motion and internet animators*. Farrar Straus & Giroux.
- Winn, B. (2008). The Design, Play, and Experience Framework. *IGI Global eBooks*, 1010-1024.