



Universidad Nacional Autónoma de México
Escuela Nacional de Artes Plásticas

Proyectos Indie: El papel del diseñador gráfico
en la industria de los videojuegos

Tesis

Que para obtener el Título de:
Licenciatura en Diseño y Comunicación Visual

Presenta: Brenda Esquivel Flores

Director de tesis:
Licenciado Francisco Alarcón González

México, D.F., 2014



Universidad Nacional
Autónoma de México

Dirección General de Bibliotecas de la UNAM

Biblioteca Central



UNAM – Dirección General de Bibliotecas
Tesis Digitales
Restricciones de uso

DERECHOS RESERVADOS ©
PROHIBIDA SU REPRODUCCIÓN TOTAL O PARCIAL

Todo el material contenido en esta tesis esta protegido por la Ley Federal del Derecho de Autor (LFDA) de los Estados Unidos Mexicanos (México).

El uso de imágenes, fragmentos de videos, y demás material que sea objeto de protección de los derechos de autor, será exclusivamente para fines educativos e informativos y deberá citar la fuente donde la obtuvo mencionando el autor o autores. Cualquier uso distinto como el lucro, reproducción, edición o modificación, será perseguido y sancionado por el respectivo titular de los Derechos de Autor.

Agradecimientos

A mi familia; Papá, gracias por darme el amor a lo digital, al diseño, y al mundo 3D desde el demo reel de Autodesk y Beyond the Mind's Eye en los 90. A Mamá, porque, aunque no le gusten los videojuegos, entendió y me apoyó por la felicidad que me brinda el apasionarme con ellos, a Xaydé, por desenredar lo que digo y darle lógica para poder contar esta historia, a Wendy, por darme fuerza y cariño aunque esté un poco lejos. A Yael, por ser de alguna forma como hermano-cuñado postizo que fue un gran apoyo.

A mis maestros Francisco Alarcón por aceptar el reto, a Gerardo García, Joel Pérez, Luis Gabriel Vázquez y Carlos Cardoso por su tiempo y dedicación para este trabajo de investigación. A Heraclio Héctor Ramírez, Ricardo Peláez y José Luis Acevedo Heredia, Guillermo de Gante y Mario Iván Silva por inculcarme mayor cariño por el arte, la gráfica e ilustración.

A mi antiguo jefe Lluís Barceló, quien me ha exigido como nadie a ser mejor y dejar atrás las excusas y miedos, además de haberme apoyado para terminar esta investigación. A Fabián Hernández por apoyarme en este rubro desde que estábamos en Slang y ser de mis primeros maestros en lo que respecta a 3D para videojuegos. A Fabián Roldán por darme las críticas severas que me hacían falta y el apoyo para ser mejor artista 3D. A Víctor Hugo Franco, mi jefe del servicio social en la DGTIC quien fue paciente en sus consejos.

A mis ex-compañeros de trabajo, Gerardo Basurto y Stephanie Prodanovich, por haberme incluido en la primera etapa para crear Celleste. A José “Luperrock” Medina por sus consejos, historias, experiencias, porras y traducciones para esta tesis.

A Daniel Enríquez, a Juan Pablo Reyes Altamirano, Ana Rosa Díaz Hermosillo, Michael Velasco, Roberto Fuchs, Arturo Nereu, Christopher Camacho, John Lindemuth, Giuliana Funkhouser, David Herrera, Jacobo Ríos, Enrique Sandoval, Rodrigo Savage y Phill.

A mis amigos, amigas y más colegas de trabajo, Grissel, Diana, Glenda y Gaby quien hizo el diseño editorial. A Cabañas, Damián, Tole, Karla, Jeshé, Yoshua, Erick, Pablo, Héctor, Pedro y Canaya.

A Human Head Studios, Surreal software, Chuck Sommerville, Scott McCarthy, Larian Studios, Turbine, Inc., Might&Delight, Cockroach Inc., Ubisoft, State of Play, Team Meat, Black Pants, Valve corporation, Sony, Microsoft y Nintendo y muchos estudios y publishers más por crear todos los maravillosos juegos que han hecho felices a muchos de nosotros, los jugadores.

A la Universidad Nacional Autónoma de México y a la Escuela Nacional de Artes Plásticas, pues no me imagino haber estudiado en otro lado.

“If this is all a dream, don’t wake me up”.

Cloud Strife

ÍNDICE

Introducción	IX
Capítulo 1: Sobre videojuegos, estudios, tecnología y México	I
1.1 Algunas definiciones sobre el término “videojuego”	1
1.2 ¿Cómo se estructura un estudio?	2
1.3 Desarrollar a gran y pequeña escala	10
1.4 Elementos generales de un videojuego e importancia del diseño gráfico	12
1.5 Tecnología y desarrollo en México	25
1.6 ¿Gameplay vs. Gráficos?	30
1.7 Papel del Diseñador y Comunicador visual (ENAP-UNAM)	33
Capítulo 2: Técnicas y procesos para la creación de gráficos	40
2.1 Las herramientas del diseño gráfico y su aplicación al arte de un videojuego	40
2.1.1 Ilustración	44
2.1.2 Diseño de interfaz	57
2.1.3 Modelado tridimensional	63
2.1.4 Texturizado	71
2.1.5 Iluminación (fotografía)	78
Capítulo 3: Proyecto Celleste	84
3.1 Concepto general	85
3.2 Antecedentes	85

3.3 Organización del equipo de trabajo	86
3.4 Justificación	88
3.5 Creación de Concept art	91
3.5.1 Personajes	96
3.5.2 Escenarios	101
3.5.3 Props	101
3.6 Características técnicas de gráficos	107
3.6.1 Texturas o mapas	110
3.6.2 Restricciones para modelos	111
3.7 Modelados	114
3.8 Texturizado	118
3.9 Animaciones	123
3.10 Iluminación y efectos especiales	127
3.10.1 Sistemas de partículas	127
3.10.1 Iluminación	130
3.11 Carga de gráficos	132
3.12 Marketing	132
Capítulo 4: Conclusiones	138
4.1 Experiencia personal y laboral	139
4.1.1 SODVI	139
4.1.2 DGTIC	140
4.1.3 Slang Studio	143
4.1.4 Elevator Games (Celleste)	144
4.1.5 Ollin VFX	145
4.2 Consideraciones sobre la industria mexicana	146
4.3 Juegos indie: espacio de experimentación y libertad creativa	148
Glosario	149
Bibliografía	157

Introducción

Jugar.

Jugar lo hacemos todos; unos en mayor o menor medida, con distintos juguetes y de diferentes formas. La acción ha sido la misma, pero los instrumentos han cambiado con el tiempo.

Los videojuegos pertenecen a dichas variaciones en las formas de juego que existen hoy día. Pueden ser transporte de historias muy intrincadas o simples; quizás sólo sean compuestos de simples reglas, pero todos deben cumplir una condición básica: ser entretenidos para su jugador o jugadores. Aun los juegos de ronda o patio escolar como “La rueda de San Miguel”, “Las traes”, “Coleadas”, etc., tienen el mínimo diseño compuesto de reglas, aunque no sea visual.

Al comenzar este trabajo de investigación, mi idea sobre la producción de videojuegos era muy pobre y me sentía bastante desorientada. Creía que un “artista de videojuegos” era en realidad un diseñador gráfico con otro nombre y que el arte — como yo lo conocía— en realidad no podía estar presente en los videojuegos. Entonces, debía haber un método, una fórmula con el fin de recabar información para estudios de mercado y lograr que un juego tuviese éxito comercial.

Mi concepción fue modificándose hasta hacerme pensar que cualquiera podía convertirse en un artista de videojuegos y, a pesar de no tener una formación profesional, que el proyecto en el que participara podría lograr una buena cantidad de ventas.

Finalmente —y después de participar en varios proyectos—, concluyo que el artista de videojuegos debe tener conocimientos, tanto de diseño gráfico como de artes visuales, incluyendo aspectos técnicos y sus limitaciones.

Ahora bien, los medios masivos de comunicación en México —impresos o audiovisuales—, en su mayoría, desconocen el contexto e historia de la industria de videojuegos en nuestro país, de manera que suelen asumir y afirmar que no ha habido un solo videojuego de producción completamente mexicana. Por tanto, si reseñan uno, en numerosas ocasiones se refieren a éste como “el primer videojuego hecho en México”.

Creo que este hecho muestra qué tan desarrollada está la industria, pues aún no se ha hecho de suficiente fama. Siendo así, es lógico que haya pocas instituciones dedicadas a formar profesionales del videojuego hasta la fecha. La mayoría de éstas son instituciones privadas y muy jóvenes. En el caso de la UNAM, ha habido algunos acercamientos en la Facultad de Ingeniería, por ser ésta formadora de ingenieros en computación, quienes pueden dedicarse a programación orientada a objetos (usada para programar, entre otras cosas, videojuegos). Alumnos procedentes de la ENAP se han acercado a algunos grupos de dicha Facultad con el deseo de producir un videojuego, pero por iniciativa propia, no porque sea una actividad profesional contemplada en perfil de egreso determinado por el plan de estudio de nuestra carrera, el cual establece:

El egresado de la Licenciatura en Diseño y Comunicación Visual es el profesional con el conocimiento de los elementos, factores, procesos y fundamentos del diseño, capaz de comprender y manejar el lenguaje visual y aplicarlo en la definición y creación de estrategias de transmisiones de mensajes visuales, para satisfacer demandas de comunicación social, cultural, histórica, científica, tecnológica y educativa.

En el aspecto técnico, tendrá el conocimiento, las capacidades y las habilidades necesarias para estructurar, elaborar y controlar los recursos necesarios para la creación de mensajes en planos bidimensionales y tridimensionales dentro de las áreas respectivas de su área de formación profesional. (ENAP, 2013).

En las siguientes páginas, buscaré demostrar que un alumno de Diseño Gráfico de la Escuela Nacional de Artes Plásticas, interesado en videojuegos, necesita una formación interdisciplinaria: no sólo la parte del diseñador y comunicador

visual, sino que también debe reunir conocimientos propios de un artista visual, rudimentos de programación y habilidades de trabajo en equipo.

En el primer capítulo explicaré las diferencias prácticas entre conceptos como “arte” y “diseño” en el ámbito de los videojuegos, desglosaré brevemente las áreas en las que se divide un equipo productor, las particularidades de los estudios grandes y pequeños, los comerciales y los independientes, el desarrollo de la tecnología (en México), así como las funciones que puede desempeñar un diseñador y comunicador visual en este campo.

En el capítulo segundo enlistaré las herramientas y conceptos que maneja un diseñador y que puede aplicar en la producción de un videojuego. También presentaré algunos ejemplos concretos de juegos donde resaltan dichas herramientas.

En el capítulo tercero aterrizaré lo argumentado en la realización de un videojuego para un dispositivo móvil, estableciendo el contexto general de su creación y —en particular— cuál era la visión del diseñador del juego, quien también dirigió el proyecto y al equipo de trabajo, en el cual participé. Dadas las circunstancias, el análisis práctico se enfoca en un estudio independiente, pero sin dejar de mencionar algunos casos en estudios con mayores recursos.

En el capítulo cuarto expondré mis conclusiones, no sólo a partir de la presente investigación, sino también de experiencias personales en la incipiente industria mexicana de juegos de video.

Esta tesis pretende ser parte de un acervo para los futuros diseñadores que se interesen por esta área, tan rica y todavía poco transitada en nuestro país. De hecho, por esta razón, me inclino a usar los tecnicismos en inglés; en principio, porque muchos no han sido traducidos y, por otro lado, así presento las palabras clave originales para consultas subsiguientes. Si bien defino el significado de cada término en su primera aparición, para efectos prácticos también ofrezco un glosario de todos los conceptos indispensables y que, a lo largo del texto, aparecerán en negritas.

Sin más que añadir, le doy la bienvenida al lector; espero que encuentre herramientas e información que sean útiles, ilustrativas e incluso inspiradoras en las siguientes páginas.

Capítulo 1

Sobre videojuegos, estudios, tecnología y México

1.1 Algunas definiciones sobre el término “videojuego”

La definición de ‘videojuego’ cambia de manera constante en tanto la historia avanza, pues las tecnologías evolucionan a pasos agigantados y, con ellas, muchos conceptos.

Pero esto no sólo depende de las empresas y estudios relacionados con el tema. La reacción de un jugador no siempre es la que esperan los productores; por lo que se hacen estudios sobre el estímulo-respuesta del usuario y si algo puede concluirse a partir de la necesidad de dichas prácticas, es que un videojuego no tiene un método fijo de producción. Un proyecto tiene necesidades distintas a los que le precedieron.

Para muchos, el principal objetivo de un juego es que los entretenga y, para ello, un juego debe ser divertido de una forma original, debe sorprender.

1.2 ¿Cómo se estructura un estudio?

Para comprender el método de producción de videojuegos, es necesario conocer —a grandes rasgos, al menos— las áreas especializadas de un estudio:

- **Dirección del proyecto.** El responsable puede provenir del área de artes, diseño o programación.
- **Diseño.** Comprende toda la planeación hecha para la construcción de un juego: mecánicas de juego, diseño de niveles, personajes, etc.
- **Arte.** Es todo el trabajo de creación de material visual.
- **Programación.** En esta área, ingenieros en computación escriben en forma de código todas las partes de un videojuego.
- **Diseño de sonido y música.** Composición de las pistas musicales para ambientación, junto con todo efecto de sonido.
- **Q&A (Quality Assessment o Quality Assurance).** Es el área de control de calidad por medio de personas (testers) que prueban repetidamente el juego para encontrar errores de programación.
- **Producción.** Los productores toman en cuenta los elementos del mercado actual para tomar decisiones, administran los recursos económicos y se encargan de la prensa.

Si un diseñador y comunicador visual quiere participar en la hechura de un juego, debe tener claras las diferencias de conceptos en el ámbito de diseño gráfico y videojuegos, pues las definiciones de un área y otra —en este caso, artes y el diseño gráfico— no siempre coinciden, aun cuando las palabras parezcan ser las mismas. Juan Acha ofrece un ejemplo muy ilustrativo sobre “Arte”:

Para nosotros, las artes corporizan, en general, un sistema cultural, y, en particular y con sus leyes internas, conforman un sistema estético.

[...] También las artes acusan una naturaleza estética y se dirigen a la sensibilidad, pero en lugares

y tiempos excepcionales, y exigen conocimientos artísticos. Es decir, se alejan de la cotidianeidad para buscar la excepcionalidad. (1995: 24-25).

Respecto al diseño, habla sobre su similitud con el arte, pero con un carácter más tecnológico: “Los diseños conjuntan al trabajo estético con el industrial masivo o, si se requiere, lo insertan en la base material de la sociedad. [...] los diseños constituyen otra variante de la cultura estética occidental: la de su fase industrial-masiva y capitalista-monopólica.” (75).

En el ámbito de los juegos de video, un *diseñador* “a secas” es el creador del concepto general, las formas de interacción, incluyendo historia y concepto visual... es decir, funge como director del proyecto de creación. Jim Thompson, —profesional de videojuegos del Reino Unido— explica el rol de los *diseñadores*: “[...] los responsables de la apariencia y de la interacción de un juego: deben generar el concepto, la historia, el mundo y la mecánica.[...] Bajo el término general de ‘diseñador’ se agrupan funciones como las del diseñador de juegos, el guionista y el diseñador de niveles...” (2007: 76). Así, sus responsabilidades son mayores y también debe intervenir, de vez en cuando, en las áreas de Programación, Arte, Sonido y Música, QA y Producción.

Veamos ahora cómo funciona y se divide el trabajo en el Departamento de Arte de un estudio de videojuegos. De acuerdo con Sam R. Kennedy (2013: 13), los puestos de trabajo son:

- **Art director (Director de arte).** Se encarga de que todo el trabajo hecho por el equipo tenga un estilo en común para entregar una experiencia consistente durante todo el juego. Dependiendo del estudio, este puesto puede conocerse como **Lead Artist**.
- **Artistas:**
 - **Concept artist (Artista conceptual).** Elabora las ilustraciones que serán la base para el arte del resto del equipo.
 - **Environment artist (Artista de escenarios).** Modela y texturiza el mundo de un videojuego, excepto los personajes.
 - **Character artist (Artista de personajes).** Modela todos los personajes y demás criaturas.

- **Character animator (Animador de personajes).** Es el encargado de crear todos los movimientos que hace un personaje como correr, caminar, pelear, etc.
- **FX (Special effects) animator (animador de efectos especiales).** Tiene como responsabilidad hacer efectos como explosiones, golpes de balas, etc.
- **User interface (UI) artist (Artista de interfaces).** Estos artistas trabajan conjuntamente con los ingenieros que permiten a los jugadores navegar por las opciones que ofrecen los menús de un juego; además, proveen de información sobre los datos vitales de su personaje.
- **Marketing artista (Artista de mercadeo).** Se encarga de crear o adaptar arte del juego para su publicidad y venta en el público en general. Puede incluirse diseño de empaque o diseño de publicidad en línea.

Por supuesto, son los estudios grandes —de aproximadamente 50 personas a más— los que usan esta estructura. En los estudios más pequeños o independientes, no es necesaria, dado que una persona del equipo puede realizar más de una tarea de varias subáreas de arte (un modelador puede ser el texturizador y animador de un modelo).

Entonces, para empezar, debe idearse el concepto básico del juego sobre qué puede tratarse éste, condensado en una idea simple.

Sin embargo, la capacidad para crear interfaces y otros mecanismos de comunicación entre humano-computadora o humano-humano implica la participación de ingenieros que programen las funciones necesarias. Para poder hacer un trabajo de esta naturaleza, es imperativo que haya un diálogo abierto entre el programador y diseñador gráfico. En un estudio de videojuegos, la comunicación es vital —Joan Costa afirma que es imprescindible en toda empresa desde mediados de siglo XX hasta hoy (2006: 15)—; el equipo de arte debe saber transmitir ideas de manera clara al equipo de programación y viceversa. En este sentido, sí existe un cargo específico que integra las disciplinas de programación (o ingeniería en computación) con diseño y comunicación visual (o diseño gráfico) y cuyo propósito es el de ser un conducto entre ambos equipos e incluso, supervisar que el arte creado cumpla con las características que requiere la tecnología utilizada para armar un videojuego: el **Technical artist**.

La empresa de videojuegos de origen alemán, *Crytek* ha publicado una vacante para dicho cargo en el presente mes (noviembre de 2013) y así lo describe:

Technical Artists are responsible for the correct technical implementation of art content. They should be able to solve any technological issues related to art production. In close cooperation with the Art Production and R&D Groups they should work on improving the existing pipelines and developing new tools for the Art Production Groups.¹

De esta manera, resulta evidente que la tarea del diseñador de proponer soluciones visuales debería abarcar el campo de los juegos de video, el cual ha sido explorado por pocos profesionales en nuestro país hasta ahora.

Por su parte, un *artista* es un profesional en el área de la imagen, experto en manejar significados a nivel visual, de manera que el perfil del diseñador y comunicador visual (tal como se concibe en la ENAP) encaja más con dicha función. Son “[...] os profesionales artísticos y de animación (artistas o grafistas) [que] dan forma visual a las ideas de los diseñadores, desde los dibujos en 2D hasta los modelos y animaciones en 3D completamente realizados. [...]” (76)² (Imagen 1).

A fin de cuentas, un videojuego es un producto y como tal, debe satisfacer las necesidades y gustos de un cliente y/o target demográfico. Por ello, en algunas empresas se hacen grupos de control para obtener datos sobre las reacciones del público al que se destina.

Ahora, en cuanto al aspecto mercadológico, es necesario distinguir dos tipos de juegos:

1) Los comerciales o de calidad AAA, que cuentan con un alto presupuesto para su producción. Suele ocurrir que

-
- 1 “Los artistas técnicos son los responsables por la implementación correcta de contenido artístico. Deben ser capaces de resolver cualquier problema relacionado a producción de arte. En estrecha cooperación con los equipos de Producción de arte e Investigación y Desarrollo, los artistas técnicos deberían trabajar para mejorar las existentes formas de trabajo y para desarrollar nuevas herramientas para los equipos de Producción de Arte.” La traducción es mía.
 - 2 Aun así, la libertad creativa varía mucho, pues depende del proyecto y quién lo dirija; además, este cargo implica conocimientos de modelado tridimensional.

Creación de recursos 3D para un videojuego



Fuente: Worcester Polytechnic Institute

1. Mapa conceptual sobre las tareas necesarias para crear recursos 3D para un videojuego.

un estudio desarrollador o empresa publicadora (*publisher*) logra obtener varios éxitos comerciales y, para mantenerse así, debe invertir más recursos en la producción del siguiente título. Por ello, se recurre a estudios de mercado, para tener claro hacia qué tipo de público apuntar y cómo satisfacer sus necesidades de juego (aunque no todas las compañías lo aplican). Esto puede limitar las posibilidades de crear algo innovador y aumentar el peligro de caer en fórmulas ya hechas, lo cual cansará y aburrirá a los receptores.

- 2) **Los independientes o Indie, que reciben poca inversión monetaria.** Por lo regular, los estudios indie tienen poco tiempo de existir y se conforman tanto por expertos (con experiencia previa en grandes estudios), como por autodidactas (con poca experiencia en el campo). Éstos comenzaron a tener mayor presencia y reconocimiento con la producción de juegos para dispositivos móviles. Pueden darse mayor libertad de experimentar con las mecánicas de juego y el aspecto visual (Look and feel). La calidad y recepción de estos productos puede llegar a igualarse con los éxitos AAA.

En tal contexto, el artista de videojuegos ha de ocuparse en satisfacer las necesidades visuales del jugador potencial, creando un vínculo entre éste y el juego. Por ejemplo, su trabajo puede influir enormemente en la formación de una experiencia



2. Mario, el personaje principal de la serie de juegos de Mario Bros., concebido por Shigeru Miyamoto.

emocional, la cual constituye uno de los grandes atractivos que permiten la vigencia del producto a lo largo de varias generaciones. Para que un título se convierta en franquicia, debe provocar experiencias agradables que el público quiera repetir; así, se establece un pacto de fidelidad y tradición (basado incluso en la nostalgia). Las mecánicas de juego propuestas por el game designer juegan el papel más importante al momento de construir tales experiencias, pero la promesa inicia con el trabajo del artista.

Valga un ejemplo conocidísimo: el personaje de Mario (imagen 2) —de la serie de juegos de Mario Bros—, fue creado a principios de la década de los 80 y aún se venden sus títulos en consola y dispositivos móviles tanto para nuevos jugadores como para las primeras generaciones, gracias a la experiencia que se les brindó y ahora tiene un fuerte valor nostálgico.

En la etapa de Pre-producción, se discute la idea general del videojuego y el diseñador construye las mecánicas con el fin de seleccionar qué herramientas o tecnologías serán las adecuadas para hacer la programación (motores de juego³, dispositivo⁴ y plataforma⁵ en la que funcionará).

3 “Código de programación que controla la estructura y la mecánica de un juego.” (Thompson et al. 2007: 188).

4 “Tipo de *hardware* con el que se juega; por ejemplo, una consola o un PC.” (188)

5 Sistema operativo que hace funcionar el hardware; por ejemplo,

Entonces, el diseño no es —no puede ser— igual al anterior, lo mismo que la manera de producirlo debe adecuarse al nuevo proyecto y al equipo de trabajo. Por otro lado, la tecnología disponible en el momento también condiciona las decisiones de uso o la creación y desarrollo de nueva tecnología.

Una vez que ya se tiene la dinámica de juego y todos los documentos de diseño en donde se incluye las características específicas de cada objeto, puede comenzar la etapa de Producción con la programación y el diseño del concept art: crear un concepto visual, hacer las ilustraciones necesarias para seguir un mismo estilo (personajes principales, utilería, vestuarios, escenarios, etc.)⁶. Matthew Omernick, Lead artist de LucasArts Entertainment Company define el concept art de la siguiente forma: “On most teams, you’ll find at least one or more concept artists whose job is to previsualize and create art for use as reference, color compositions, and just good old-fashioned inspiration. Concept art is vital to a production for numerous reasons.” (2004: 6).⁷

Para la etapa de Post-producción, el estudio entrega una versión llamada alpha del juego, en la que básicamente todos los elementos visuales están listos e implementados y sólo deben refinarse errores que pueda haber en la programación. Después, en la versión beta, se corrigen defectos menores y más específicos, incluyendo las necesidades que tenga la plataforma para la que irá dirigido el juego. Finalmente, en la versión gold, éste podrá empezar a ser distribuido en los puntos de venta designados.

Por otra parte, un juego de video, en cuanto a gráficos, puede ser en 2D (dos dimensiones) o 3D (tres dimensiones). El libro *Videojuegos. Manual para diseñadores gráficos* explica a grandes rasgos ambos tipos:

una *Tablet* puede funcionar con Android o iOS.

- 6 En este sentido, los conocimientos de un diseñador y comunicador visual son cruciales para cumplir con este trabajo. Como artista de videojuegos, es necesario saber sobre teoría del color, texturas, tipos de materiales, composición, reticulación (aplicable a diseño de *GUIs*), comunicación, proporción humana e ilustración digital y tradicional. Con todo, para complementar semejante lista, también hacen falta conocimientos de escultura (por ejemplo, para modelado 3D); las herramientas y materiales serán distintas, pero los principios son los mismos.
- 7 “En la mayoría de los equipos de trabajo, encontrarás al menos uno o más artistas conceptuales cuyo trabajo es previsualizar y crear arte para usar como referencia, composiciones de color, e inspiración tradicional. El arte conceptual es vital para una producción por numerosas razones.” La traducción es mía.



3. Screenshot del juego *Wolfenstein 3D*.

Un juego en 2D tiene acciones y actividades que funcionan en dos dimensiones [...] Quizás la mejor manera de identificar un juego es considerar si podría jugarse utilizando gráficos en 2D además de un sencillo movimiento de cámara en 2D con pocos o ningún cambio en la mecánica de juego.

[...] Un juego 3D se desarrolla en tres dimensiones. [...] [N]ecesitan movimiento y visión verdaderamente tridimensionales: sería imposible representar estos juegos en mundos bidimensionales sin cambios fundamentales en sus reglas y sus mecánicas.” (Thompson et al. 2007: 48-49).

Los primeros videojuegos nacieron con gráficos en dos dimensiones (Space Invaders, Centipede, Asteroids, Pong, etc), pues en los años en los que se crearon, los gráficos en computadora debían restringirse a ello. Sin embargo, una de las

razones por las cuales los videojuegos tienen tanto éxito es por su capacidad de crear un mundo alterno y la sensación inmersión que ofrecen al jugador. Con esta tendencia, surgieron imágenes cada vez más detalladas, hasta derivar en gráficos 2D que parecían ser de una dimensión superior (como el juego Wolfenstein 3D (imagen 3), desarrollado por id Software en 1992).

Tal desarrollo dio paso a los juegos en tres dimensiones, que le darían un giro a la historia de los videojuegos y los harían más populares. En este punto, Berbank-Green resalta el carácter revolucionario de esta nueva perspectiva:

La producción de juegos en 3D subrayó la importancia de la cámara en lo referente a la jugabilidad (experiencia del jugador). Los diseñadores se enfrentan al problema de situar y controlar una cámara de manera que no interfiera con las acciones del jugador, pero que al mismo tiempo, le permita ver lo que desea. Los primeros juegos en 3D utilizaban la perspectiva de cámara en primera persona, lo que significa que el jugador ve principalmente a través de los ojos del personaje que controla. De esta forma, los controles otorgaban un movimiento natural a la visión y el personaje del jugador no tenía que aparecer en la pantalla, lo que suponía un ahorro en producción y en tiempo de procesamiento. ((Thompson et al. 2007: 49).

Según el tipo de gráficos que se utilizarán, se decide qué tipo de software se manejará para crear los assets o elementos individuales que conforman el juego.

1.3 Desarrollar a gran y pequeña escala

Más arriba ya decía que, sin importar las técnicas y el software, sin importar que un equipo de trabajo tenga muchos o pocos integrantes, la mejor opción es la que se adapte al diseño del juego, porque el objetivo siempre será atraer muchos jugadores. Porque no se necesita pertenecer a un

estudio gigantesco y famoso para ofrecer una experiencia grata y adictiva. Veamos dos ejemplos de estudios con características diferentes: Blizzard Entertainment, un estudio muy grande y con varios equipos numerosos, y Might & Delight, un estudio independiente de apenas once personas.

Blizzard Entertainment.

Surgió en 1991 bajo el nombre de Silicon & Synapse con tres ingenieros de la Universidad de California en Los Ángeles. El estudio se fundó con su propio dinero y comenzaron a contratar gente para su equipo. El primer artista 3D en sus filas fue Joeyray Hall (ahora DVD/Video Production Supervisor) y el artista que ahora es Senior Art Director, Sam Didier.

Para celebrar su vigésimo aniversario, el estudio publicó una serie de videos y material que muestra los procesos, los recuerdos y los logros a lo largo de los años. Uno de los videos (2011) contiene justamente un mensaje de los fundadores y entrevistas con los desarrolladores donde se dice que desde sus inicios y hasta ahora, han logrado muchos avances tanto en tecnología como en arte. Para el juego de Blackthorne que lanzaron en 1994, por ejemplo, aplicaron en sus animaciones por primera vez técnica de rotoscopiado. Hall, en entrevista (2011), comenta: “Nos llevamos a Frank Pearce al callejón e hicimos que saltara montones de madera, que corriera, subiera escaleras y cosas así; lo grabamos en video. Posteriormente regresaríamos a la oficina y dibujaríamos al personaje sobre él.”

En el mismo año, comenzaron a trabajar en Warcraft: Orcs & Humans, juego que tuvo gráficos en 3D. Nick Carpenter —ahora Vicepresidente de Desarrollo de arte y Cinemáticos— relata: “Empecé en Blizzard en el año '94 y rápidamente me pusieron en un cubículo junto con Micky Neilson, Dave Berggren y Chris Metzen. No conocía bien a Metzen en aquel entonces. Recuerdo que estábamos empezando a aprender 3D y era un dolor de muelas porque todos teníamos un trasfondo artístico muy tradicional.” Chris Metzen, Senior Vicepresident en Story and franchise development⁸ lo apoya: “Todos los artistas estaban haciendo la transición a 3D MAX, la industria iba en dirección al 3D. Recuerdo que el programa me intimidó sobremanera.”

Después de eso, desarrollaron muchos juegos más, incluyendo toda la saga de Warcraft, de Diablo, y de Starcraft (todos,

⁸ Historia y desarrollo de franquicias.

juegos de rol para jugar en línea), con millones de jugadores activos. Este estudio cuenta hoy con más de 4000 empleados en todas sus sedes.

Might & Delight

Estudio independiente creado en 2010 en Estocolmo, Suecia. Sin embargo, a pesar de ser joven, sus integrantes principales no lo son tanto, pues han trabajado en títulos reconocidos como Battlefield, Killzone, Mirror's Edge, entre otros (Might & Delight, 2013).

Los integrantes principales son once, pero afirman tener más miembros en otras partes del mundo. Hasta el momento han lanzado dos títulos: Pid (2012) para XBOX 360, Microsoft Windows, Mac OS X y Playstation 3 —que recibió críticas por la dificultad de sus niveles y acertijos y elogios sobre la mecánica de manipulación de gravedad— y Shelter (2013) para Windows y Mac —que igualmente ha recibido críticas mezcladas, sobre el gameplay, no tan divertido para muchos, pero con un arte y valor sentimental alto—.

Ambos estudios son opuestos en muchos aspectos, pero cada uno ha logrado forjar una imagen propia y obtener excelentes ventas, las cuales dependen del tipo de público y el contexto general; después de todo, las propias etiquetas de “AAA” e “independiente” a veces condicionan el trato del público y lo que éste espera del estudio. Cuando inició Blizzard, sus directivos llevaron al estudio en una dirección de tendencias generales. Might & Delight, por su parte, comienza en un periodo de agresiva competencia y la necesidad de ofrecer algo diferente, experimental, más íntimo.

1.4 Elementos generales de un videojuego e importancia del diseño gráfico

Ya tenemos una idea más clara de los elementos que constituyen cualquier juego y quiénes son los encargados de las diferentes áreas. Ahora subrayaré la importancia de nociones específicas al momento de construir una experiencia agradable e intuitiva para un jugador.

Para empezar, es necesario tener conocimientos básicos sobre psicología del color y la sintaxis de la imagen, los cuales resultan decisivos al crear un aspecto visual que atraiga la



4, 5. Screenshots de distintos niveles de *Diablo III*.



6. Screenshot de *Diablo III*.



mayor atención posible. En este sentido, la experiencia de Christian Lichtner —Art director del juego de *Diablo III*⁹— es un ejemplo muy ilustrativo. Al hacer su presentación en la Game Developers Conference (2012), Lichtner remarcó la importancia de aplicar una progresión de color en el juego o tener filtros de distintos colores en la iluminación, (imágenes 4, 5 y 6) para mantener el dinamismo y, por tanto, la atención del jugador.

El juego comienza cuando eliges tu personaje. Hay diferentes clases (mago, monje, bárbaro, cazador de demonios, médico brujo, etc.) y cada uno tiene algún tipo de ventaja. El primer nivel está ambientado en una aldea, en la noche. Los colores son azulados y oscuros. Hay muy poca acción, pues es un nivel de tutorial, para familiarizarse con los controles y mecánicas. En el siguiente nivel, el ambiente sigue siendo nocturno y con tonos azulados, pero es de un color más brillante. Hay más enemigos que enfrentar y ponen a prue-

⁹ Lanzado en 2012 por el estudio Blizzard Entertainment. Se trata de un popular juego de rol y de acción para PC

ba tus habilidades como jugador. El siguiente nivel sucede en interiores y tiene colores más cálidos. El peligro comienza a acechar por más rincones. Sin embargo, la progresión de color no es lineal. De tanto en tanto, el jugador se dirige a niveles y escenarios con colores fríos como se mostraba en los anteriores. Cada uno de los screenshots de más arriba muestra esto precisamente. Algunos escenarios contienen mayor cantidad de combates y tienen ambientaciones de colores más cálidos y agresivos; otros son más contemplativos (aunque no al punto de aburrir al jugador), para descubrir nuevas cosas e historia; son de colores fríos, frescos y relajantes.

Otro aspecto importantísimo es cómo lograr una armonía entre el arte y la mecánica del juego, es decir, que el arte tenga un propósito funcional, más allá de la estética y Robert Briscoe (2012) —Lead artist del juego Dear Esther— da fe de ello.

Este videojuego tiene una estructura basada más en la narrativa interactiva que en el juego propiamente dicho; no obstante, ganó muchos adeptos a nivel mundial casi de inmediato. En Dear Esther, se aplican elementos visuales que ayudan a crear un vínculo emocional entre el jugador y la historia: el mundo virtual tiene varios escenarios con piezas de utilería que, aunque pequeños, tienen una historia detrás; unos no están perfectamente limpios y cuidados como si los hubiesen recién sacados de una fábrica, otros tienen marcas del paso del tiempo, herrumbre, hay huellas de tacto humano, etc. En la imagen 7 se aprecia esto en la costa donde se encuentran restos de otros barcos que naufragaron en la misma isla que el protagonista; Briscoe señala que esto mostraba una mayor profundidad en la historia de la isla.

Asimismo, Briscoe hace uso de los “señalamientos subliminales” como él los llama: elementos que guían al jugador por el camino correcto, indicaciones visuales sencillas como pequeñas luces ubicadas al final de un pasillo (imagen 8).

Entonces, tener nociones de composición y comunicación también permiten construir un aspecto visual que sirve como verdadero apoyo a la experiencia del jugador para que éste cumpla sus metas y termine el juego de forma satisfactoria.

Un ejemplo más es Journey (imagen 9), desarrollado por That Game Company y cuyo Creative director y co-fundador es Jenova Chen¹⁰.

¹⁰ Nacido en China, Jenova Chen hizo una licenciatura en Ciencias de la Computación y una licenciatura menor en Arte Digital y Diseño. En Los

7. Impresión de pantalla del nivel inicial de *Dear Esther*.



8. Robert Briscoe hace hincapié en el simbolismo oculto en el aspecto visual. En esta captura de pantalla, dos luces se proyectan en el suelo un corazón, en el punto más íntimo de la historia relatada en *Dear Esther*.



Según un artículo del sitio PC Mag, Journey ganó el premio de “Juego del Año” en la premiación de los GDC Awards (Premios de la Conferencia de Desarrolladores de Videojuegos, por sus siglas en inglés). Journey compitió con títulos como Dishonored, The Walking Dead, Mass Effect 3 y XCOM: Enemy Unknown:

“When video game outlets post their game of the year lists, the selected titles are typically AAA affairs with bloated development budgets, incredible marketing budgets, and lots of visceral action to instantly pull gamers into their digital worlds,” PCMag software analyst Jeffrey L. Wilson, wrote in his review of the game. “Journey... is worthy of game-of-the-year consideration despite the absence of such elements.”¹¹ (Moscaritolo, 2013).

Y en una entrevista hecha por Jamin Warren para el sitio Kill Screen, Jenova Chen dice respecto a su trabajo en Journey:

I think of my games as something that’ll communicate a strong emotion to make people’s lives better. It’s not quite set into a particular tone just yet since there’s so much open space. But each of my games is pointing it to here or back there. We want games to help people rather than take advantage of people. [...]

I’m not American. I grew up in Asia and China is my home. I always want to make sure that my

Ángeles, obtuvo su maestría de la Universidad del sur de California, en la División de Medios Interactivos. También cabe señalar que el *Art director* del estudio es Ke Jiang, quien estudió en el Colegio de Artes y Diseño de Minneapolis. Después, obtuvo una maestría en Bellas Artes en Animación Experimental en el Instituto de Artes de California.

- 11 “Cuando instituciones del videojuego publican sus listas de “juego del año”, los títulos seleccionados son típicamente de categoría AAA con presupuestos desorbitados para su desarrollo, además de presupuestos increíbles para su mercadeo, además de mucha acción visceral para atraer instantáneamente a los jugadores hacia sus mundo digitales,’ según palabras del analista de software de PCMag, Jeffrey L. Wilson, como lo escribió en su reseña del juego. ‘Journey ... es digno de consideración para juego del año a pesar de la ausencia de dichos elementos.’”. La traducción es mía.

9. Screenshot de Journey.



games can be played here and in my home. Because of that, language is a barrier. You can say “Why not translate?” Well I guess, I’m a bad English language writer. I don’t write good dialogue. That’s my limit.¹² (Warren, 2013).

Un hecho que podría haber constituido una desventaja, Chen logró convertirlo en una cualidad única, pues al no haber diálogos, el lenguaje se volvió “universal” y por tanto, llegó a un público más amplio.

En este sentido, es necesario puntualizar que el Art director decide la forma de trabajo más óptima para crear gráficos de la

12 “Pienso en mis juegos como algo que comunicarán una fuerte emoción para hacer mejor las vidas de las personas. No está dispuesto en un tono en particular aún, ya que hay mucho espacio abierto en él. Pero cada uno de mis juegos lo señala aquí o allá. Queremos juegos para ayudar a las personas, en vez de aprovecharnos de ellas.

[...]

No soy estadounidense. Crecí en Asia y China es mi hogar. Siempre quiero asegurarme de que mis juegos pueden ser jugados aquí y en mi hogar. Por ello, el lenguaje es una barrera. Puedes pensar ‘¿por qué no traducirlo?’, Y bueno, supongo que es en parte porque soy un mal escritor del lenguaje inglés y no escribo buenos diálogos. Ése es mi límite.” La traducción es mía.

mejor calidad posible y que sean adecuados para cargarlos al juego. En el caso de los gráficos en 3D, el Art director suele trabajar codo a codo con el Game designer y el guionista para saber cuáles son sus preferencias y su visión, de manera que tenga una guía para crear la propuesta gráfica. A partir de este trabajo interdisciplinario, nace la llamada Graphic Bible o Art Bible (Biblia gráfica o de arte), la cual establece el estilo general del juego, pues abarca todo el alfabeto visual necesario (utilería, bocetos de interfaz, character sheets, paletas de color, escenarios, storyboard de cinemáticos, etc.).

De acuerdo con el sitio web de desarrollo de videojuegos gamedev.net, una Graphic Bible debe contener:

- Style sheets and color schemes (i.e., pantone) for your characters and monsters. These may specify 4-10 different views of each character, which will guide animators and 3D modelers in maintaining consistency throughout the product; they allow you to let more than one artist work on the same character at the same time if necessary.
- Style sheets for the major objects, vehicles and props.
- Maps of the levels and environments. This may include the graphical part of level design.
- Background drawings.
- A storyboard for the intro and for any other FMV/ animation sequences in the product.

A designer who happens to be an artist can produce the graphic bible, alone or in collaboration with others. (It is a *lot* of work.) Designers without an artistic background should still collaborate and supervise this effort, so as to ensure that the look of the game will be consistent with their vision. (Laramee, 1999)¹³

13 “Hojas de estilo y esquemas de color (por ejemplo, en colores hexadecimales) para tus personajes y monstruos. Éstos pueden especificar de 4 a 10 distintas vistas de cada personaje, que guiará a los animadores y modeladores 3D para mantener la consistencia en todo el producto; éstos permiten que más de un artista trabaje en el mismo personaje al mismo tiempo si es necesario.

- Hojas de estilo para objetos grandes, vehículos y utilería.

El **Art director** o el *Lead artist*, al hacer este compendio, necesitan basarse en referencias de objetos ya existentes y saber reinterpretarlos visualmente de forma que sean atractivos.

Ya que el material principal está listo, los modeladores comienzan a trabajar con los dibujos de referencia de cada personaje, objeto o edificio y los modelan en low poly (pocos polígonos)¹⁴. Después de modelar un personaje, se le da color y profundidad con mapas de textura, luego se construye el Rig (esqueleto) y se incorpora al modelo para que los animadores —al recibir el personaje ya completo— empiecen su trabajo correspondiente.

En muchos casos de videojuegos de carácter realista, es usual recurrir al Motion Capture (o mocap: captura de movimiento) en vez de hacer una animación desde cero. Claro que, lo anterior depende de la inversión y los recursos disponibles; las sesiones de captura son costosas, pero suelen ahorrar tiempo y dar mayor calidad por el detalle que pueden ofrecer. De cualquier manera, se necesitan siempre animadores para “limpiar” las capturas, ya que puede haber errores que las cámaras por sí mismas no son capaces de resolver (por ejemplo, al grabar la sesión de movimiento, una cámara podría no detectar un hueso que parece estar sobre otro). Este tipo de animaciones son usadas para juegos como First Person Shooters (FPS: disparadores en primera persona) o juegos de baile.

Josh Scherr —Cinematics Animation Lead de Uncharted (del estudio Naughty Dog en 2007)—, fue entrevistado por el sitio web “Mocap Club” y recordó que, en un principio, el estudio había desarrollado los juegos de Crash Bandicoot y Jak

-
- Mapas de niveles y ambientes. Esto puede incluir la parte gráfica del diseño de niveles.
 - Ilustraciones de fondos ambientales.
 - Un storyboard para el intro y para cualquier otro FMV (*Full Motion Video*) o secuencias de animación en el producto.

Un diseñador que resulta ser artista puede producir la biblia gráfica, solo o en colaboración con otros. (Esto es mucho trabajo). Los diseñadores aún sin un antecedente artístico deberían colaborar y supervisar este esfuerzo, de tal forma que se asegure que el aspecto del juego sea consistente con su visión.” La traducción es mía.

- 14 Modelar una pieza de utilería o un personaje con poca complejidad como método de optimización para que el juego corra más rápido. Normalmente, el *framerate* o velocidad de cuadros por segundo de un juego debe estar entre los 40 y 60 f/s (*frames per second*).

and Dexter de una forma tradicional, para después continuar con la exitosa franquicia de Uncharted (con tres partes hasta ahora). En este punto, se le preguntó qué había motivado a Naughty Dog para utilizar mocap en Uncharted (imagen 10), después de haber usado animación creada desde cero, a lo que Scherr contestó:

Obviously the biggest change between the Uncharted series and the Jak series was that we were moving from doing cartoony characters with stylized performances to much more realistic humans. We did plenty of keyframe animation tests early in the production of the first Uncharted; what we came to realize is that doing realistic human animation via keyframing is a very difficult and time-consuming process, and something that simply wasn't feasible given our schedules and budgets. For example, during the production of Uncharted 2, the cinematic animators were expected to do 15 seconds of finished animation per week, including facial animation. A very doable goal when working on the stylized characters of the Jak series – in fact, our quotas on those games ranged from 20-30 seconds per animator per week! – but when you're dealing with realistic humans, if you don't get the animation right, it's going to take the player completely out of the experience. We all know what people look like when they move, and we all know when it looks wrong or weird. Motion capture helps us get a good starting base, especially for all those subtleties that are really difficult to keyframe, and then we build on the performance from there.¹⁵ (Scherr, 2013).

15 “Obviamente, el mayor cambio que hubo entre la serie de Uncharted y la de Jak, es que estábamos moviéndonos de hacer personajes caricaturescos con movimientos estilizados a humanos mucho más realistas. Hicimos muchas pruebas de animaciones con cuadros clave en el principio de la producción del primer Uncharted; de lo que nos dimos cuenta fue que hacer animación realista de humanos con cuadros clave es un proceso muy difícil y que conlleva demasiado tiempo, además de que era algo poco posible con nuestros itinerarios y presupuestos. Por ejemplo, durante la producción de Uncharted 2, los animadores de cinemáticos tendrían que haber hecho 15 segundos de animación terminada por semana, incluyendo animación facial. Esto era más posible trabajando

10. Comparación de la escena de **screenshot** contra el cinemático de uno de los juegos de la serie de *Uncharted*.



11. Screenshot del juego de *Battlefield 3*, creado con el motor *Cryengine*, desarrollado por *Crytek*, distinguido por ser uno de los motores comerciales con mayor calidad gráfica.





- II. Escenario virtual en el cual se muestra el **wireframe**, o estructura geométrica y las texturas de los árboles con transparencia o **canal alpha**. En realidad, todos son planos individuales o compuestos de planos separados, con la finalidad de optimizar el rendimiento del juego.

Desde comienzos de 2000, los videojuegos han dependido, en mayor o menor medida, de la narrativa cinematográfica (con ángulos de cámara propios de películas) (imagen 11). Por ello, otro aspecto que mejora la experiencia del jugador —además del mocap— es la iluminación. En tal caso, las nociones de fotografía son vitales.

Por ejemplo, en un juego puede haber muchas sombras de acuerdo con la posición de una fuente de luz inmóvil, pero no todas son iguales: algunas de ellas no son “verdaderas” sombras, sino texturas (para optimizar), el motor del juego debe calcular sombras en tiempo real de las fuentes de luz que sí se mueven. En el caso del PC (Player Character, Personaje que usa el jugador), las sombras que proyecte serán calculadas en tiempo real también, aunque la fuente de luz sea inmóvil.

con personajes estilizados de la serie de Jak (¡De hecho, nuestras cifras en esos juegos oscilaban entre los 20-30 segundos de animación por animador, por semana!) pero cuando estás tratando con humanos realistas, si no tienes la animación correcta, sacarás al jugador de la experiencia completamente. Todos sabemos cómo se ve la gente cuando se mueve, y sabemos cuando se ve incorrecto o raro. La captura de movimiento nos ayuda a tener una buena base para comenzar, especialmente para esos detalles que son muy difíciles de animar en cuadros clave, y sobre eso, nosotros continuamos trabajando.” La traducción es mía.

Por otra parte, están los efectos. Existen diferentes aplicaciones —según lo requiera el juego—, pero en general incluyen elementos como humo (simulado por partículas precalculadas) o Billboards (imagen 11). Omernick dice en este apartado: “[B]illboards are similar to particle effects in that they are sprites that can face the camera. Billboards typically use larger textures (256x256 and up) and are usually static, aside from the rotation to face a camera.”¹⁶ (2004: 212).

Finalmente, está el diseño de Graphic User Interfaces (GUI o Interfaces Gráficas para el Usuario). Para esto, es imperativo tener nociones, no sólo de diseño gráfico, sino de interactividad y usabilidad.

En el ámbito del mercado, un factor muy importante es si el juego podrá venderse en otros países y, de ser así, si será necesario cambiar el idioma en que estén las interfaces y varios gráficos del juego. Mark Peasley —artista veterano con 20 años de experiencia en videojuegos y con más de 25 juegos en su currículum— escribe un artículo donde señala que traducir un título a otro idioma se denomina “Localización” y recomienda:

First, do not embed text directly into the artwork for the game or menu. If possible, text should be handled via the programming code as either a TrueType font or a bitmap font. However if it is embedded in the art (such as a sign or a logo), it is a good idea to separate the text onto a unique layer in the base art file. This will allow the artist easy access to only the artwork that is affected by the translation. (Rabin, 2010: 651).¹⁷

La mayoría de estos conceptos se redefinen y actualizan a partir de los nuevos avances tecnológicos. Por tal razón, el

16 “Los *Billboards* son parecidos a los efectos de partículas en el aspecto de que son *sprites* que siempre apuntan hacia la cámara. Los *billboards* típicamente usan texturas más grandes (de 256x256 y para arriba) y por lo general son estáticos, fuera de su rotación a la cámara”. La traducción es mía.

17 “Primero, no incrustes el texto directamente al arte para el juego o menú. De ser posible, el texto debería ser manejado vía código de programación ya sea como fuente TrueType o de mapa de bits. Sin embargo, si está incrustado el texto en el arte (como un símbolo o logo), es buena idea separar el texto en una capa única en el archivo base de arte. Esto permitirá al artista un acceso más fácil sólo al arte que sea afectado por la traducción.” La traducción es mía.

profesional de arte en videojuegos debe estar a la par en el manejo de herramientas y teorías nuevas.

1.5 Tecnología y desarrollo en México

En la actualidad, la industria de los videojuegos es, sin duda, menos oscura al interior del país y cada vez más jóvenes se interesan en su desarrollo. Sin embargo, los hechos que rodean a la situación nacional son evidentes para muchos. La crisis económica se ha agravado y los inversionistas cuidan con mayor celo en quién y en qué proyectos confían sus recursos. Por ello, resulta un tanto trágico decir que varios de los primeros estudios productores de videojuegos en México, no lograron sobrevivir hasta el día de hoy.

La revista *Game Master* presentó un número especial (noviembre, 2011) sobre esta industria en el país, en el cual expuso el panorama general y los factores involucrados, algunos porqués del lento desarrollo de un mercado de videojuegos mexicanos, así como la opinión de algunos desarrolladores de diversos estudios nacionales.

Dentro de este último punto, Ricardo Gómez —CEO de Kaxan Media Group (autores de *Taco Master*, juego para iOS)— asevera: “Tenemos la misma creatividad y tecnología que tienen los mejores estudios. En este país hay tanta gente talentosa que las desarrolladoras internacionales nos los ‘piratean’. Enfocarse en vender en México ya no es suficiente, tenemos que ver el mercado global. Nosotros mismos nos estamos limitando”. (*Game Master*, 2011: 43)

Por su parte, Miguel Olvera —líder ejecutivo de *Alebrije Studios*— comenta: “La mayoría de los programas de estudios no están al día y pecan de ingenuos en el uso de la tecnología”. (*Game Master*, 2011: 46)

Pero estos no son los únicos problemas. El hecho es que varios recursos de las nuevas tecnologías no están al alcance de todos; además, el conocimiento sobre administrar un negocio de esta naturaleza, alcance y complejidad es algo que no se obtiene con facilidad. Para hacer un producto (cuya historia se conoce poco), es necesario observar la producción de las grandes potencias, recopilar la información necesaria, armar una nueva propuesta y entonces crear un videojuego que logre competir a nivel internacional.

Al respecto, Paulina Sánchez —reportera de Game Master— cita al antiguo productor de Slang, Andro Miralrío:

Un problema que no dejaron de mencionar varios de los entrevistados es que no hay suficiente gente con las herramientas y conocimiento necesario para ingresar a un estudio. Y en las universidades mexicanas, la mayoría de los profesores que imparten las materias ‘especializadas’ en videojuegos, no están preparados. “Las universidades abren, pero no tienen quién dé las materias. ¿De dónde los van a sacar? (Sánchez, 2011: 46).

Aquí hay varios hechos que es imperativo señalar:

- 1) Hace falta una educación especializada, no sólo para los jóvenes que quieren entrar a la industria, sino para los mismos desarrolladores que ya laboran en un estudio.
- 2) Tal educación debe partir de experiencias de campo reales (en el país y en el extranjero).
- 3) Es necesario contar con una infraestructura que sostenga y promueva el trabajo en la industria de los videojuegos en México.

Claro que no se trata de una visión utópica; en términos prácticos y crudos, un plan así supone la prosperidad de una empresa, así como el fortalecimiento de relaciones entre instituciones. Partiendo de esta condición, los productores ejecutivos pueden invertir en la capacitación constante para ingenieros, artistas y productores

Con base en lo dicho anteriormente, está claro que el problema no sólo atañe a los diseñadores, programadores, artistas o instituciones educativas. La inversión en tecnología en el país ha sido poca en comparación con el resto de los líderes de la industria de los videojuegos (Estados Unidos, Japón, Inglaterra, Francia, Canadá), porque, de manera general, el empresario mexicano tiene la particularidad de pensar a corto plazo.

Daniel Enríquez —ingeniero en computación y programador de gráficos con una larga trayectoria¹⁸— señala que,

18 Laboró en *Firetoken* (empresa especializada en videojuegos para casino)

además de recursos monetarios, la inversión en la industria de los videojuegos también es de tiempo. “Hacen falta profesionales y eso lleva tiempo” asegura¹⁹. Una gran parte de la capacitación de profesionales se ha llevado a cabo en empresas de Baja California Norte; sin embargo, la fuga de cerebros es un peligro latente, dado que la industria aún se encuentra en plena formación en el país y, por tanto, es difícil encontrar ofertas de trabajo. En otras palabras, nos encontramos ante un círculo vicioso.

Dadas las circunstancias, pocos ingenieros en computación mexicanos han incursionado en investigación y desarrollo; de los que lo hacen, muchos han sido financiados por instituciones educativas o privadas. En cualquiera de los dos casos, una de las principales metas es la creación de motores de juego (o motores gráficos) propios, nacionales.

Para tener una idea más clara de qué es un motor de juegos, Noel Llopis, en su artículo “Teams and processes” explica y describe lo siguiente:

Programming the game engine involves all the code that goes into the final game that is not game specific. You can think of it as all the support code necessary to have the game run on top of it. Sometimes, people erroneously think of a game engine as being what draws the pretty 3D graphics, but that is only the graphics renderer (also called the “graphics engine”, which adds to the confusion), which is one of the many parts of the game engine.

One of the goals of a game engine is to isolate the game from the hardware on which is running. It does that by creating an abstract layer between the game and the hardware. That way, the game doesn't have to worry about the details of the platform and can concentrate exclusively on the game logic itself. Examples of this type of abstraction are gathering input from the controller, putting graphics on the screen, or playing back sounds.²⁰ (Rabin, 2010: 168).

(imágenes 19, 20 y 21) de 2009 a 2011 y, actualmente, en el estudio de videojuegos de origen francés *Gameloft*, división de Mexicali,

19 Mencionado en una plática informal conmigo.

20 “La programación de un motor de juegos involucra todo el código

Con todo, en México, sí ha habido ingenieros que han construido motores gráficos. Entre ellos, pueden destacarse el motor producido por Gamecoder (2013) —una empresa dedicada al desarrollo de videojuegos y tecnología en gráficos de computadora, cuyos creadores son brillantes estudiantes de ingeniería en computación y residen en Guanajuato—, el motor creado por el mismo Daniel Enríquez para la empresa Firetoken y los estudios de Snake & Eagle (2013)—que habían mostrado actividad hasta el 2010— aseguran en su sitio web que crean motores gráficos y de juegos para sus proyectos.

Paulina Sánchez, en el artículo citado más arriba, afirma que los últimos diez años han sido cruciales en la demostración de la capacidad del país para producir videojuegos:

A partir del 2000 surgieron los primeros estudios mexicanos dedicados a hacer juegos, entre los que resaltan Snake&Eagle, AztecTech Games, Radical y Evoga Entertainment, quienes colaboraron con Noise Factory para lanzar *Rage of the Dragons* para arcades y Neo Geo. Estos estudios poco a poco fueron desapareciendo del radar o se declararon en bancarrota. (Sánchez, 2011: 45).

El interés y la iniciativa están presentes, eso es innegable; la tragedia radica en cómo la falta de recursos y de inversión acaba con lo que pudieron ser proyectos brillantes.

Ahora bien, tampoco puede negarse que la tecnología tiene efectos directos sobre la calidad de los recursos gráficos. Existen motores comerciales desarrollados por estudios extranje-

incluido en el juego final, que no es el juego en sí. Se puede pensar en el motor como toda la ayuda necesaria para que el juego se ejecute sobre él. A veces, la gente piensa de manera errónea sobre los motores de juego como lo que hace posible los impresionantes gráficos en 3D, pero eso sólo es gracias al motor de renderizado (también llamado “motor de gráficos”, que añade algo de confusión), que es una de las muchas partes del motor del juego.

Uno de los objetivos del motor de juegos es aislar el juego del hardware sobre el cual se ejecuta. Esto se logra al crear una capa abstracta entre el juego y el hardware. De esa forma, el juego no tiene que ocuparse en detalles de la plataforma y se dedica exclusivamente a la lógica de sí mismo. Ejemplos de este tipo de abstracción son el acceso al estado del control, dibujar gráficos sobre la pantalla, o reproduciendo sonidos de fondo.”. La traducción es mía.



12. Screenshot del juego *Renegade X*. Ventajas gráficas de UDK: detalle y calidad.



13. Screenshot de *Shadowgun*, juego para iPhone e iPad, utilizando el motor de Unity.

ros como el motor CryEngine 2 (desarrollado por Crytek), el Unreal Engine 3 (de Epic Games), Torque (de GarageGames) o Unity (de Unity Technologies).

Por la calidad que puede brindar el motor de Unreal o UDK (Unreal Development Kit o Equipo de Desarrollo de Unreal) (imagen 12 y 13), es uno de los más usados a nivel profesional.

Unity es un motor que tiene ventajas especialmente para estudiantes, por ser fácil e intuitivo de usar, con muchos recursos gratuitos en línea además de que sus licencias llegan a ser muy económicas, a diferencia del Unreal engine 3.

Una vez elegido el motor, se identifican cuáles son los perfiles que debe cumplir cada tipo de recurso gráfico (por ejemplo, para UDK, deben importarse los modelos en formato de FBX). Es importante tomar en cuenta este tipo de restricciones desde un inicio, pues de no hacerlo, se perdería tiempo al tratar de hacer la conversión necesaria de todos los recursos gráficos a los perfiles que exige el motor.

El artista de videojuegos puede verse limitado por las herramientas utilizadas para crear el juego; no siempre el artista puede elegir el material con el que trabajará y debe adaptarse a las limitaciones que pueden tener los recursos de que se ve provisto. Para tener éxito de las desventajas en este punto, el artista debe contar con capacidad creativa para poder crear una propuesta visual que aproveche las ventajas del motor utilizado.

Uno de los puntos necesarios a tomar en cuenta es la fidelidad con la que se presenta un gráfico de un juego para promocionarlo a sus potenciales clientes. Es posible que si se presenta una ilustración de un personaje del juego, y no el modelo del personaje dentro del juego, pueda haber mucha disparidad entre las características gráficas entre uno y otro, así que por esta razón, es imperativo tener una dirección de arte constante a lo largo de todo el juego.

1.6 ¿Gameplay vs. gráficos?

Ahora bien, uno de los problemas que enfrentan los nuevos juegos es la aparente contienda entre el gameplay (las mecánicas o reglas) y sus gráficos.

En la actualidad, hay una tendencia a imprimir mayor realismo en las expresiones de los personajes, las texturas, la

iluminación, etc., lo cual promete una experiencia muy atractiva; no obstante, en algunos casos parece más importante que el juego sea “visual” y se olvida la parte del entretenimiento. La sola simulación de realidad no basta para convencer a un usuario de que vale la pena permanecer en el mundo que se ha construido para él; si el público no percibe un desafío, si no crea un vínculo o no se identifica con un personaje, el juego no obtendrá el éxito esperado.

La “inmersión” es un concepto central en muchas discusiones sobre videojuegos. Marianne Krawczyk y Jeannie Novak tratan el tema desde una perspectiva narrativa, igualmente aplicable en términos visuales:

The objective behind writing an engaging story, building interesting worlds, and giving the player control leads to one conclusion: immersion. In order for the player to get the most out of the game experience, it is necessary to make the player feel as much a part of the world as possible -thereby giving the player a presence within the game world and the game story. To achieve this, Imitation of the real world is necessary. This is not to say that players shouldn't suspend disbelief or that the stories can't be fictionalized. Fantasy, sci-fi, and horror can and should be used, but there are basic real-world rules that help add to the immersive experience of a game.²¹ (Krawczyk et al., 2007: 93).

Uno de los grandes placeres humanos consiste en tener el control sobre algo (una situación, un objeto o incluso sobre una persona); se disfruta saber cómo se desarrollará un determinado evento en el futuro (Dille et al. 2007: 2). En

21 “El objetivo detrás de escribir una historia atractiva, construir mundo interesantes y darle el control al jugador conlleva a una conclusión: inmersión. Para que el jugador saque la mayor ventaja de la experiencia de juego, es necesario hacer al jugador sentir como si fuese parte de ese mundo, tanto como sea posible; dándole así al jugador una presencia dentro del mundo del juego y la historia del juego. Para lograr esto, la imitación del mundo real es necesaria. Esto no significa que los jugadores no deban suspender incredulidad o que no se puedan hacer historias ficticias. Fantasía, ciencia ficción y horror pueden y deben ser utilizados, pero hay reglas básicas del mundo real que ayudan a hacer inmersiva una experiencia de juego.” La traducción es mía.

14. Screenshot²² de *Too Human* de *Silicon Knights* (2008).



este aspecto radica la importancia del gameplay: no sólo se le ofrece al jugador la oportunidad de participar activamente en una historia, sino que se le da la oportunidad de elaborar estrategias funcionales en aras de completar el juego con éxito. Laitinen explica esta dinámica con gran elocuencia: “El juego debería transmitir la sensación de que el jugador tiene el control. Si demasiadas sorpresas ocurren dentro del juego, existe el riesgo de que el jugador comience a sentir que el juego es más una lotería que un juego. Esto puede afectar de forma negativa la experiencia de juego”. (Isbister et al., 2008: 106)

Muchos juegos han fracasado por darle prioridad al atractivo visual, no importa que pertenezcan a exitosas franquicias. Secuelas de juegos muy esperadas por los fanáticos han sido motivo de avivada crítica, por darle prioridad al aspecto visual del juego sobre el gameplay. La recepción negativa se traduce, invariablemente, en pérdidas monetarias. Un caso es el título *Too Human* (2008) de *Silicon Knights* (imagen 14). Duró diez años en desarrollo, con muchos atrasos y que finalmente obtu-

vo críticas negativas por su gameplay mediocre, problemas con controles, cinemáticos demasiado largos, etc. No todo en el juego fue malo, pero es un juego no muy popular.

A fin de cuentas, es el público general quien tiene la última palabra sobre la calidad de un juego y en cuanto a esta “pugna” de elementos, queda claro que ambas partes deben trabajarse en igualdad de condiciones si se quiere ofrecer un producto armónico, equilibrado y atractivo.

1.7 El papel del diseñador y comunicador visual (ENAP-UNAM)

El constante flujo de información en medios de comunicación ha provocado, entre otras cosas, incertidumbre en los receptores. Muchos de los mensajes presentados al público —aunque parezcan tener fundamento en hechos reales o bases científicas— siguen poniéndose en duda, pues los emisores claman tener la versión “correcta” de las cosas (y esto levanta serias sospechas).

El trabajo que se le encomienda al diseñador y comunicador visual es crear soluciones gráficas que reduzcan al mínimo esa incertidumbre. La meta consiste en convencer visualmente al espectador de la validez de un mensaje por encima de otros. De esta manera, para poder ofrecer las mejores propuestas con la mayor calidad, es imperativo contar con la tecnología más reciente.

No obstante, en la actualidad se presenta un problema doble: por una parte, existe la creencia (no generalizada, para nuestra fortuna) de que los diseñadores no necesitan un título o una formación universitaria para llevar a cabo su trabajo si éste “se ve bien” y funciona para el cliente; por la otra, muchas empresas no exigen un título universitario a sus empleados porque esto les permite deslindarse de su obligación de ofrecerles garantías laborales²³; en otras palabras, el tipo de con-

22 http://static.giantbomb.com/uploads/original/0/26/11800-shot_0031376.jpg

23 En mi experiencia personal, la primera empresa en la que trabajé como pasante o *junior artist* fue en Slang Studio, donde nunca se me pidió tener título, pero se me apoyó para continuar con la redacción de la tesis. El sueldo que recibía era más bien simbólico y como ayuda económica para transporte y comida.

tratación en medida creciente es por outsourcing. Dado que la competencia es cada vez más feroz, algunos de los que quieren avocarse al Diseño y la Comunicación visual optan por tomar cursos o diplomados de corta duración para incorporarse pronto al mercado laboral.

Es verdad que depende mucho de cada persona y de cómo se maneje en el campo profesional. En lo que respecta al arte para videojuegos, puede haber profesionales con una formación autodidacta y que tengan un desempeño excepcional. También es cierto que un título no garantiza conocimientos, habilidades o dominio de la materia en ninguna profesión. Sin embargo, hablar de las condiciones reales en lo laboral, debería permitirnos pensar, reflexionar y proponer cuáles serían las condiciones ideales para un profesionista.

Entonces, primero hay que tener claros los elementos y factores que rodean al diseñador que ha sido formado específicamente en la Escuela Nacional de Artes Plásticas y que busca relacionarse con el campo de los videojuegos:

a) Pertenece a la carrera de Diseño y Comunicación Visual, en la orientación de Audiovisual y Multimedia. Las materias de tronco común (primeros cuatro semestres) que se le ofrecen al alumno son: Diseño, Dibujo, Geometría, Fotografía, Metodología para la Investigación, Teoría e historia del arte, Técnicas de representación gráfica, entre otras. En Audiovisual y Multimedia (especialización de los últimos cuatro semestres), se imparten materias de Producción Audiovisual, Multimedia, Animación (tradicional y digital) y Dirección de arte. A pesar de la enorme utilidad y pertinencia de dichas asignaturas, no hay un enfoque particular en el área de los videojuegos, tampoco se ofrecen teorías ni prácticas especializadas que preparen al alumno dentro del plantel. Todos estos conocimientos deben sintetizarse, aplicarse y ampliarse en otro lado.

b) Hay muy pocas instituciones educativas relacionadas con los videojuegos en el país y, las que hay, son todavía muy jóvenes. Las escuelas particulares se han introducido desde hace algunos años en el campo del diseño gráfico aplicado a videojuegos, como el SAE Institute México (School of Audio Engineering and the SAETechnology College), que abrió sus puertas al público en 2010; la Universidad de Artes Digitales establecida en Guadalajara, Jalisco en 2007. De vuelta a la UNAM, sí se ha mostrado un fuerte interés y las acciones más importantes se realizan en el área de programación para



15. Sitio web de SODVI (Sociedad de Desarrollo en Videojuegos de la Facultad de Ingeniería, UNAM).

videojuegos de la Facultad de Ingeniería. No obstante, uno de los grandes problemas es que se promueve muy poco el trabajo interdisciplinario, es decir, que los involucrados en un proyecto sean de diversas carreras. Para el desarrollo de un videojuego, se necesitaría la participación de diseñadores y comunicadores visuales, ingenieros en computación, ingenieros de sonido, músicos y productores.

c) **Las posibilidades de hacer trabajo interdisciplinario son escasas y no son promovidas por las instituciones, sino por los alumnos.** Existe un grupo de alumnos llamada Sociedad de Desarrollo en Videojuegos (SODVI) fundada en la Facultad de Ingeniería que busca formar alumnos de distintas áreas en la producción de un videojuego y enseñarles a partir de conceptos y nociones propias de cada carrera (Imagen 15).

De acuerdo con la visión de SODVI: "Ser una Sociedad que contribuya a la formación de profesionistas especializados en

la creación y desarrollo de videojuegos, capaces de impulsar ésta industria en el país”. Mientras que la misión es: “Formar profesionales con los conocimientos elementales acerca del desarrollo de un videojuego y poder facilitar su vínculo con la industria en México”.

Uno de los proyectos que han realizado es un juego llamado Pimo que presentaron en un concurso ofrecido por la empresa Square Enix (de origen japonés, división México), el “Latin America Game Contest 2012”. Este proyecto lo desarrollaron alumnos de la Facultad de Ingeniería junto con alumnos de la ENAP, usando el motor de juego Unity (imagen 2).

Rodrigo Medina, actual responsable de SODVI, me comenta algunos aspectos del juego, después de contactar con él. Hubo cuatro diseñadores gráficos, todos provenientes de la ENAP, cinco programadores y un músico. Los diseñadores fueron:

- Jorge Álvarez: **Technical artist** (animador y encargado de comunicación con programación).
- Axur Eneas: **Concept artist** a cargo de diseño de personajes.
- Juan Pablo Riebeling: También concept artist y creador de escenarios.
- Gustavo Marmolejo: Diseñador de interfaces.

Rodrigo Medina me concedió acceso a la versión de prueba del juego que reseñaré brevemente. La historia inicial relata la vida tranquila y feliz de un circo y sus trabajadores, excepto el de uno, quien es un encargado de limpieza y odia su trabajo. Al ver la felicidad de un mago que dirige el circo, el intendente decide robar el sombrero de copa con el que el mago hace sus trucos. Así, hipnotiza a los inconformes trabajadores del circo para que sigan todas sus órdenes. Pero un personaje singular y aparentemente mágico llamado Pimo escapa de este embrollo y parece ser la salvación de este grupo de artistas embaucados. Al finalizar el cinemático que narra esta historia, comienza el juego. Pimo está en un escenario de circo con muchas plataformas sobre las cuales debe saltar para avanzar. Este personaje puede generar de la nada paredes frente a él que empuja sobre las plataformas para bloquear el ataque de enemigos que se le cruzan en el camino (imagen 16).

Al tener presentes ejemplos de trabajos como este, los diseñadores de la ENAP podrían tener mejores herramientas



16. Captura de pantalla del juego *Pimo*, creado por miembros de SODVI.

y conocimientos actualizados sobre tecnología y sus tendencias, así como una mayor claridad de hacia qué direcciones podría apuntar la línea del diseño en videojuegos. Lo anterior permitiría adquirir un mayor compromiso profesional y la expansión de horizontes en medios interactivos como los videojuegos y/o la realidad virtual.

- d) Existe el prejuicio de que los juegos de video sólo son “nuevas formas de perder el tiempo y enajenar a los jóvenes”, cuando la realidad es muy diferente. En la Dirección General de Cómputo y Tecnologías de la Información y la Comunicación (DGTIC) de la UNAM —dentro del Departamento de Realidad Virtual—, el Departamento de Visualización y la Sala Ixtli (imagen 17) ha usado y aplicado nociones de diseño en el área de interacción en Realidad Virtual, principalmente en recorridos virtuales a lugares lejanos o de difícil acceso y en esquemas interactivos utilizados como material didáctico para clases extraordinarias de Medicina, Psicología, Ingeniería, Química, etc.

Además, desde hace tres años, el Departamento de Realidad Virtual se ha embarcado en la tarea de ofrecer recursos didácticos para clases a nivel bachillerato. Se tiene hasta la fecha y

17. Presentación de avances de un proyecto 3D para la enseñanza de Anatomía en la sala Ixtli (2007).



activo un proyecto de videojuego que entra en la categoría de Serious Games²⁴.

Según palabras del Dr. en Informática de la Universidad Pontificia de Salamanca en Madrid, España, Jimmy Rosario, a estas tecnologías se les define de esta forma:

Se denominan Tecnologías de la Información y de Comunicación al conjunto de tecnologías que permiten la adquisición, producción, almacenamiento, tratamiento, comunicación, registro y presentación de informaciones, en forma de voz, imágenes y datos contenidos en señales de naturaleza acústica, óptica o electromagnética. Las TICs incluyen la electrónica como tecnología base que soporta el desarrollo de las telecomunicaciones, la informática y el audiovisual. (2005).

Rosario hace referencia a características principales como la intangibilidad e instantaneidad, pero la más importante para el diseñador es la de sus aplicaciones multimedia:

²⁴ Género de videojuegos destinado a enseñar y apoyar la docencia o capacitar personal, usando el recurso del videojuego como material didáctico.

Las aplicaciones o programas multimedia han sido desarrollados como una interfaz amigable y sencilla de comunicación, para facilitar el acceso a las TICs de todos los usuarios. Una de las características más importantes de estos entornos es “La interactividad”. Es posiblemente la característica más significativa. A diferencia de las tecnologías más clásicas (TV, radio) que permiten una interacción unidireccional, de un emisor a una masa de espectadores pasivos, el uso del ordenador interconectado mediante las redes digitales de comunicación, proporciona una comunicación bidireccional (sincrónica y asincrónica), persona-persona y persona- grupo. Se está produciendo, por tanto, un cambio hacia la comunicación entre personas y grupos que interactúan según sus intereses, conformando lo que se denomina “comunidades virtuales”. El usuario de las TICs es por tanto, un sujeto activo, que envía sus propios mensajes y, lo más importante, toma las decisiones sobre el proceso a seguir: secuencia, ritmo, código, etc.

Otra de las características más relevantes de las aplicaciones multimedia, y que mayor incidencia tienen sobre el sistema educativo, es la posibilidad de transmitir información a partir de diferentes medios (texto, imagen, sonido, animaciones, etc.). Por primera vez, en un mismo documento se pueden transmitir informaciones multi-sensoriales, desde un modelo interactivo. (2005).

Las Tecnologías de la Información permiten a las personas acercarse a este campo desde otra perspectiva, lo cual han impuesto un nuevo reto para los diseñadores gráficos y programadores. Por ello, es necesario que su formación actual comprenda nuevos temas, pues invariablemente, una interacción entre computadora y humano implica la existencia de una interfaz, un medio que ayude a presentar la información necesaria para el usuario, diseñada de manera clara, legible y atractiva.

¿No es evidente la pertinencia y utilidad de nuestros conocimientos como diseñadores y comunicadores visuales en el campo de los videojuegos?

Capítulo 2

Técnicas y procesos para diseño de gráficos

2.1 Las herramientas del diseño gráfico y su aplicación en el arte de un videojuego

Los procesos del diseño —tal cual los conocemos en la carrera— son aplicables en la producción del arte de un juego. “No es posible cambiar una sola unidad del sistema sin modificar el conjunto”. (Dondis, 1973: 53). Todo aquel elemento que pueda extraerse de la composición general de los elementos visuales del juego sin afectar a éste en lo más mínimo, deberá eliminarse por completo, dado que no trabaja con la armonía de la composición.

De acuerdo con la línea de pensamiento de Bruno Munari sobre metodología del diseño (Vilchis, 1998), el proyecto empieza en cuanto el **game designer** —después de haber concebido el concepto general junto con el resto del personal

de su área— provee al resto de los integrantes la información indispensable de su idea general, lo mismo que un cliente pide un cierto producto y pide un rendimiento de cuentas.

En la planeación de un juego es necesario ubicarse en el contexto general de desarrollo. En un estudio con altos recursos económicos (productores de juegos AAA), es común hacer una investigación de mercado que permita conocer las necesidades del **target** o las nuevas tendencias en videojuegos. Esta es una de las herramientas que un diseñador debe tomar siempre en cuenta para construir una propuesta eficaz y favorable para su cliente. En este sentido, es vital que un juego logre captar la atención del jugador y para ello, éste debe ser capaz de identificarse con alguno de los personajes; “[e]l desarrollo de los personajes es una tarea tanto visual como conceptual. Además de los numerosos bocetos que definen la apariencia de un personaje, también debe diseñarse su personalidad, ya que tendrá un efecto directo en la disposición y en la reacción del personaje ante las situaciones en las que se encuentre.” (Thompson et al., 2007: 94).

Sobre el tema, Jon Burgerman (2013), ilustrador inglés, aconseja una serie de ideas para lograr una mayor eficacia al crear un personaje¹:

01. Reseach and evaluate (Investigar y evaluar). Es parte de hacer una investigación previa de otros personajes que ya han sido diseñados e incluso tengan una popularidad consolidada (ya sean en TV, web, o medios impresos).

02. Design and plan (Diseña y planea). Tomar en cuenta en qué medio aparecerá el personaje, pues eso puede limitar o aumentar el número de características que lo distinguen. Un personaje de videojuego debe tener una forma distintiva de caminar y, por tanto, una forma específica de ser animado; debe tener una indumentaria que indique de dónde viene, si pertenece a algún grupo, de qué clase social es (si aplica), si tiene marcas en el cuerpo, si es pulcro o desaseado... es decir, todo aquello que ofrece indicaciones sólidas sobre cómo crear el personaje.

04. Visual impact (Impacto visual). Conviene analizar qué elementos distinguen a un personaje ya popular entre una determinada población. Burgerman también aconseja pen-

1 De todos los que enlista, sólo se citarán algunos de los ejemplos más relevantes e ilustrativos.

sar el tipo de público al que está dirigido, pues esto fijará un estilo visual guía que cumpla con lo básico para captar la atención del público y, al mismo tiempo, ofrezca una propuesta novedosa (debe haber algo fuera de lo común que evite confusiones y semejanzas). Si no hay suficientes elementos que convenzan al público de la particularidad de un personaje, éste pasará desapercibido entre la enorme gama de los ya existentes.

10. Conveying personality (Transmitir personalidad). Algo tan sencillo como el tipo de trazo² dice mucho acerca de un personaje, pero también es necesario construir su personalidad a partir de emociones específicas y tipos de expresiones; debemos preguntarnos cómo reaccionaría ante diversas situaciones, qué tipo de ropa prefiere usar y por qué, si tiene cicatrices (y cuál es su historia, pues podrían indicar eventos trágicos o heroicos de su pasado), su forma de caminar, de moverse (indica su edad, si es sano o si está enfermo, etc.). En este aspecto, Flint Dille y John Zuur Platten (2007: 65-68) señalan que hay varios tipos de personaje, independientemente del género del juego: el **PC** (*Player character*, personaje del usuario), **NPCs** (*Non player characters*, personajes no jugables que son parte del mundo e interactúan con el personaje asignado o elegido por el usuario). Dentro de la categoría de los *NPC*, encontramos los aliados (que serán de ayuda en las misiones), los neutrales (que pueden tener trascendencia en el juego, sin importar las decisiones del jugador), y los enemigos (que tienen la misión clara de atacar o infligir daño al personaje jugador). Un tipo especial de enemigo es el **Boss** o “Jefe” de nivel, pues es el personaje más poderoso y más difícil de eliminar; generalmente, al derrotarlo, se le permite al jugador seguir avanzando al siguiente nivel. Por último, existen los **Player directed characters**. Éstos son múltiples personajes que puede controlar el jugador (por ejemplo, un ejército, en un juego de estrategia).

18. Release the beast (Libera a la bestia). Una vez construido el personaje, debe afinarse y mostrarse al público en general (no sólo al target) para recibir críticas y confirmar si el personaje comunica lo planeado en un principio.

² Es decir, la calidad de línea; por ejemplo, si es de vector, gruesa, con muchas curvas y formas básicas, sugiere una orientación general hacia un público infantil, por ser más llamativo.

Flint Dille y John Zuur Platten (2007: 77-82) ofrecen una plantilla que ayuda a puntualizar las características del personaje. Aquí, las más relevantes:

Nombre	Incluye, a veces, el apellido y/o apodo.
Propósito en la historia	Breve descripción de la travesía que tendrá el jugador con este personaje.
Propósito en la mecánica de juego	¿Qué tipo de personaje es? ¿PC, NPC, Boss...?
Alineación	¿De qué lado está? ¿Cuáles son sus límites éticos?
Personalidad básica	Personalidad condensada en un adjetivo.
Disposición general	Su actitud personal y con respecto a su ambiente.
Habilidades especiales	Las habilidades que utiliza para cumplir su objetivo.
Educación/Inteligencia	Nivel de educación. Destreza mental.
Familia	
Aspiraciones	
Adicciones	
Ocupación y actitud hacia ésta (sea buena o mala)	
¿Qué desea este personaje?	
¿Qué es lo que ama?	
¿A qué le tiene miedo?, etc.	

Así, también se resuelve una parte de la planeación de la historia, tarea a cargo del guionista o del **game designer**³.

Ahora, los personajes creados deben interactuar con un entorno virtual y es por ello que la construcción de los escenarios también debe tener fundamentos: “Una vez que se ha decidido el concepto general y la localización del juego, se puede comenzar a desarrollar su aspecto y su ambientación con más detalle. Al igual que sucede en el cine y en la televisión, el entorno es indispensable para determinar la atmósfera general del producto final”. (Thompson, et al., 2007: 98). Asimismo, las referencias y el estilo visual de los escenarios deben facilitar la inmersión del usuario, debe permitir que se establezca un estrecho vínculo con la historia y la mecánica del juego. La experiencia incluso promueve la lealtad del consumidor hacia el estudio creador y sus futuros productos.

Por supuesto, lo dicho en el párrafo anterior sirve como introducción para quien busca enfocarse en la industria de juegos AAA. En el caso de los estudios independientes, se generan metodologías de trabajo en función de cada proyecto, ya que existe la libertad de un desarrollo afín con las capacidades del equipo.

2.1.1 Ilustración

Todas las paletas de colores, bocetos, propuestas de personajes, ambientes, utilería, etc. que se hacen en videojuegos son **concept art**, responsabilidad del Departamento de Arte. El concepto de arte en el ámbito de los videojuegos, recordemos, se refiere a toda aquella ilustración o bosquejo (alfabeto visual) que sirven de referencia para los iluminadores, animadores, modeladores y texturizadores. Es muy importante que todos en el equipo comprendan cuál será el **Look and feel**⁴ del juego para trabajar de manera coordinada y Dondis lo reafirma desde su propio ámbito: “El diseño y la proyectación del material visual tridimensional, orientado hacia la comprensión

3 Como ya he señalado antes, el *game designer* suele tener la mayor carga de trabajo; es posible que también funja como guionista y puede proponer y corregir el arte de un juego.

4 Literalmente “aspecto y sensación”, es decir, el cómo se percibe a través de los sentidos.

afortunada de un problema, exige muchos pasos para idear y proyectar las posibles soluciones”. (1973: 77)

Si hemos establecido que no todos los juegos deben ser iguales y —lo que es más importante— deben atraer la atención de nuevos jugadores es imperativo tener presente que el juego debe ser distinto, único, lo cual demanda nuevas soluciones gráficas que se adapten a un motor en particular y sus mecánicas.

A continuación, muestro cuatro casos recientes muy ilustrativos sobre este apartado:

THE UNFINISHED SWAN (2012, Giant Sparrow)

En la página de los estudios Giant Sparrow, se explica así la trama de *The Unfinished Swan*: “The player is a young boy chasing after a swan who has wandered off into a surreal, unfinished kingdom. The game begins in a completely white space where players can throw paint to splatter their surroundings and reveal the world around them.”(2013)⁵.

El juego empieza como un cuento para dormir, con el clásico libro que se abre para mostrar el texto y donde se cuenta la historia de Monroe, un niño cuya madre adoraba pintar, pero que nunca terminaba lo que empezaba. Al morir ella, Monroe es enviado a un orfanato en el cual sólo le permiten tener una pintura de su madre; él escoge el favorito de su mamá: el “Cisne sin terminar”. Una noche, descubre que el cisne se ha escapado de la pintura y sólo encuentra el rastro de sus huellas que conducen a una pequeña puerta. Con un cubo de pintura, el niño emprende la búsqueda. Sin embargo, al atravesar el umbral encuentra que todo es absolutamente blanco, y pronto se revela que es a causa de un rey que deseaba tener un bello reino y había concluido que ningún color sería lo bastante digno, así que todo se quedó en blanco. Poco a poco, más partes de la historia se revelan para el jugador conforme éste avanza y los siguientes niveles incluyen algunos objetos más que ya tienen otros colores; el primero en aparecer —además del negro de la pintura— es el amarillo de las huellas del cisne y de los objetos dorados en el reino (imagen 18). Después, se

5 El jugador es un pequeño niño que persigue un cisne que ha ido a vagar hacia un reino sin terminar y surreal. El juego comienza en un espacio completamente blanco donde los jugadores pueden arrojar pintura para manchar a sus alrededores y revelar el mundo que hay detrás.” La traducción es mía.

18. Screenshot de uno de los primeros niveles de *The Unfinished Swan*.



relata que la gente se quejaba de estar sumergida en blanco, pues no podían encontrar sus cajas y se tropezaban a cada rato. Así, el rey comienza a pintar sombras. Es por ello que en los siguientes niveles pueden distinguirse mejor los edificios y otros objetos. Más adelante, el color de la pintura cambia a un azul pastel claro, pero ésta se desvanece a unos segundos después de haber pintado algo (imágenes 19 y 20).

Aun cuando todo empieza como una hoja nueva, se revelan figuras interesantes y que cumplen su función principal: ubicar al jugador en el mundo. La solución gráfica cumplió con la premisa de ser original, ser óptima para el motor (pues no supone mucha información respecto a la gama de colores y no es necesario calcular muchas sombras), tener un propósito en las mecánicas de juego y ofrecer un reto atractivo para el jugador.

THE BINDING OF ISAAC (2011, Edmund McMillen & Florian Himsl)

Un gran ejemplo de arte original es el de *The Binding of Isaac*. Edmund McMillen —co-**designer** del juego— también elaboró el estilo visual y sus gráficos son reconocibles en todos los



19 y 20. Screenshots de los niveles iniciales, en los que empieza a aparecer algo de color.

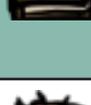


21. Screenshot de *The Binding of Isaac* con todos los bosses.



proyectos en los que ha participado. La historia se basa en la que se describe en la Biblia sobre Abraham y su hijo Isaac, pero reinterpretada como la travesía de un niño (Isaac) que debe escapar de su madre (una fanática religiosa, quien cree oír la voz de Dios que le ordena matar a su retoño como prueba de su fe) y termina por esconderse en el sótano. En él, pelea contra toda clase de monstruos con variadas formas de ataques —los cuales se vuelven más fuertes y complejos en cada nivel—, para finalmente enfrentarse a su madre y batirse en un duelo a muerte por su libertad. Dependiendo del tipo y número de logros, se revelan distintos finales en la historia.

Con una trama así, lo normal, tal vez, sería esperar personajes con calidad de línea delgada, en trazos angulosos y paletas de colores amenazantes. Y sí, pero sólo a medias, pues McMillen crea un mundo con personajes de proporciones pequeñas y redondas —como las de un bebé—, con calidad de línea gruesa y curva, trazos de puntaje alto en vectores y sombras planas en pastas de colores en los que predominan rojos, carmines, negros y verdes oscuros (imagen 21). Lo visual es coherente con la historia, no sólo por los elementos ya mencionados, sino porque hay muchas referencias y símbolos de la religión cristiana-católica. Algunos de los **power ups** u objetos que proveen de habilidades especiales al personaje del jugador son los siguientes:

Nombre del objeto	Descripción	Ícono
Lord of the Pit (El señor de la fosa)	Este objeto le da a Isaac la habilidad de flotar sobre todo obstáculo que se le atraviese.	
Prayer card (Tarjeta de plegaria)	Isaac tiene una media vida “eterna”.	
Book of Revelations (El libro de las revelaciones)	Añade una “vida de alma” y provoca que el próximo boss sea un jinete (del Apocalipsis)	
Book of Shadows (El libro de las sombras)	Otorga invulnerabilidad por poco tiempo.	
Dead Sea scrolls (Rollos del mar muerto)	Usa un objeto activado al azar.	
Rosary (Rosario)	Añade tres “vidas de alma” y aumenta la posibilidad de que aparezca una Biblia en los niveles subsecuentes del juego.	
The Bible (La Biblia)	Transforma a Isaac en un ángel, permitiéndole volar sobre obstáculos sólo en el nivel en el que está. Por otro lado, matará al jugador si se usa contra Satán, Isaac o “???”.	
Blood of the Martyr (Sangre del mártir)	Aumenta el daño contra el enemigo por 2 puntos (x2) al convertir las lágrimas de Isaac en lágrimas de sangre.	
Guardian Angel (Ángel guardián)	Engendra un ángel guardián que orbita alrededor de Isaac. Provoca daño a los enemigos si lo tocan, bloquea proyectiles y duplica la velocidad de otros objetos orbitando alrededor de Isaac.	
Holy Grail (Santo Grial)	Añade un contenedor de vida vacío y le da a Isaac la habilidad de volar.	

Los monstruos con los que Isaac debe de enfrentarse, a pesar de verse amenazadores, entran en el mismo alfabeto visual: con líneas curvas y calidad de línea gruesa, e incluso con los mismos ojos negros y brillantes como Isaac, pero con un toque perverso y monstruoso:

Nombre	Descripción	
Satan (Satán)	Este es uno de los bosses más peligrosos por su complicado modo de ataque. Se divide en tres partes, pues el personaje evoluciona el mismo número de veces.	
Famine (Hambruna)	Flota sobre el suelo y a veces persigue a Isaac. Al hacerlo, Hambruna lo embestirá, saliéndose del cuadro de juego por un lado y regresando por el otro extremo. Dispara moscas que arrojan sangre. Si se debilita lo suficiente, sólo quedará la cabeza que inflige daño con sangre.	
War (Guerra)	(Guerra)Convoca múltiples bombas que estallan cerca de Isaac y dispara en muchas direcciones. Embiste igual que Hambruna. Si se debilita bastante, Guerra queda a pie para perseguir a Isaac. En ocasiones parece hacer gestos de rabieta como un bebé.	
Pestilence (Pestilencia)	Igual que Hambruna, Pestilencia flota; dispara contra Isaac bombas de veneno y éstas marcan un rastro que desvanece poco a poco, pero mientras está activo, inflige daño a Isaac si él camina sobre él. Engendra varios tipos de gusanos.	
Death (Muerte)	Dispara múltiples guadañas que giran en línea recta. Muerte se baja de su caballo para que ambos ataquen desde lados distintos a Isaac. Asimismo, engendra "caballeros", personajes grises invulnerables en todo el cuerpo excepto en la nuca, pues su cerebro está descubierto.	
Duke of Flies (El Duque de las Moscas)	Flota y escupe moscas de diferentes tamaños que dañan a Isaac si lo tocan. Igualmente, si el Duque de las Moscas lo toca, le causará daño.	
Gemini (Géminis)	Persigue a pie a Isaac. Al bajar suficientes puntos a este enemigo, el feto conectado por un cordón umbilical se liberará y perseguirá más rápidamente a Isaac, escupiendo proyectiles de sangre hacia él.	

Puede sorprender al jugador una *aparente* discrepancia entre el carácter visual y la temática, lo cual también sirve para llamar la atención. El hecho de que McMillen haya roto con lo convencional —manteniendo su objetivo de funcionalidad dentro del juego y sus mecánicas—, hace destacar a *The binding of Isaac* de entre el resto de los juegos de temática “oscura”.

TINY&BIG: GRANDPA’S LEFTOVERS (2012, Black Pants Studio)

Set in land where common physics apply, but everything else looks different. You are Tiny, a techophile guy with a ray cutter, a gripping-device and a fine attitude towards the world. But now your nemesis Big again stole the only heritage your grandpa left you: A nice pair of white, fine rib underpants! And he’s surely up to no good, why else would he take them to that forsaken desert... Where noone goes! So load up that selfmade raygun and rope device (snatched from the local hardware store) and take your annoying and jabbering backpack to accompany you. It’s always been good use to your gramps. Besides, the robot taxi to the desert is waiting! Run, jump, drag, shove and laser your way through the ultimately ancient scenery, gain back those underpants!⁶ (2013)

Uno de los aspectos más notorios de este juego es su distintivo estilo visual de cómic (imagen 22). En cada golpe con láser, cada caída de un objeto o algún daño violento contra un personaje, un letrero onomatopéyico en 3D aparece en escena, haciéndolo más cómico-dramático (imagen 23).

6 Toma lugar en una tierra donde la física común reina, pero todo lo demás se ve diferente. Tú eres Tiny, un chico tecnófilo con un cortador de rayo, una agarradera y una buena actitud hacia el mundo. Pero ahora, tu némesis Big volvió a robar la única herencia que tu abuelo te dejó: ¡un bonito par de calzones blancos! Y seguramente no trama nada bueno, ¿por qué otra razón los llevaría al desierto... ¡donde nadie va!? Así que carga tu arma de rayos casera y tu aparato de cuerdas (robado de una ferretería local) y lleva a tu molesta y dicharachera mochila contigo. Siempre le fue útil a tu abue. Además, ¡el taxi robot que va al desierto te espera! Corre, brinca, arrastra, empuja y usa tu láser para abrirte paso a través del máximo escenario antiguo, ¡recupera esos calzones!“. La traducción es mía.

Otra característica propia de cómic incluida en el **arte**, es que, cuando hay un diálogo entre personajes, éste aparece en un globo negro. Las texturas de los personajes también pertenecen a un tipo de ilustración con entintado tradicional y algunos detalles con **ashurado**⁷. Los colores tienen poca profundidad y aparecen en plasta (imagen 24). Los personajes lucen como si estuvieran envueltos por sus texturas, sin sombras precalculadas ni recibiendo luz directa o ambiental como suele apreciarse en otros juegos.

El responsable de este singular estilo es Sebastian Stamm, ilustrador y artista de cómics. Es uno de los **CEO**⁸ de Black Pants, además de **Creative Director**⁹. Estudió Ilustración y Animación en la Escuela de Arte y Diseño de Kassel, graduado con excelencia en Comunicación Visual (Ilustración y Cómic).



22. Una de las portadas del juego.

7 Técnica de sombreado con tinta.

8 *Chief Executive Officer*, director principal de una empresa u organización.

9 Director Creativo. Función similar a la del *Art Director*, es decir, la de definir los parámetros visuales de un proyecto y así, poder administrar al equipo en esa dirección.



23. Una de las portadas el juego.



24. Character sheet de Tiny.

LUME (2012, State of Play)

With a set built entirely out of paper and cardboard, and sumptuously filmed, Lume is a game with a style and tactility unlike any other.

As Lumi, an inquisitive young girl, you discover that power to your grandad's house has failed. What's more, he's nowhere to be seen, but appears to have left you some intriguing clues.

Immerse yourself in Lume's photoreal world, solve perplexing paper puzzles to help restore the power through ingenious eco-technology, and uncover a deeper mystery behind the blackout. Beautifully animated with handcrafted care, this illuminating puzzle adventure game will put visual, aural and memory skills to the test.¹⁰ (2013)



25. Imagen comercial del juego de *Lume*.

10 "Con un escenario construido enteramente con papel y cartón y filmado suntuosamente, *Lume* es un juego con un estilo y tactilidad únicos. Como Lumi, una niña muy inquisitiva, descubres que la electricidad en la casa de tu abuelo ha fallado. Lo que es más, el abuelo no se ve por ninguna parte, pero parece que te ha dejado pistas intrigantes. Sumérgete en el mundo fotoreal de *Lume*, resuelve complicados acertijos y rompecabezas de papel para ayudar a restaurar la luz a través de ingeniosa eco-tecnología y descubre el profundo misterio detrás de apagón. Bellamente animado, con cuidado artesanal, este brillante juego de acertijos y aventuras pondrá a prueba tus habilidades visuales, auditivas y mnemónicas." La traducción es mía.



26. Screenshot del principio del juego, con la protagonista, Lumi, en casa de su abuelo.



27. Uno de los escenarios de Lume.

Este juego podría parecer 2D en algunos momentos; sin embargo, los desarrolladores relatan que su forma de crear los objetos del mundo de Lume fue por completo tradicional. State of Play mantuvo un blog de desarrollo¹¹ y en él muestran las maquetas que hicieron para el pueblo y casas de *Lume* (imagen 28).

28. Luke Whittaker —uno de los desarrolladores— trabajando en un teclado que aparece al principio del juego, en el cual el jugador debe reproducir, exactamente igual, una melodía en un fonógrafo.



Los objetos que conforman el mundo de Lume tienen la característica de partir de una base plana (el papel y cartón), con algunos detalles con un pocas rugosidades cerca de los bordes. La calidad de línea de algunos *props* es delgada, negra, pero notoria, de carácter caricaturesco. Su gama de colores contrasta con la iluminación utilizada sobre el set.

11 Algunos estudios acostumbran hacerlo para mantener interés del público, tanto general como posibles colaboradores.

Otro elemento básico —además de la apariencia— es el movimiento, aunque no entendido en el sentido tradicional, como afirma Dondis: “[...] observemos que incluso en esta forma [cine], no existe movimiento auténtico tal como lo conocemos; este movimiento no es achacable al medio sino al ojo del observador en el que se da el fenómeno fisiológico de la ‘persistencia de la visión’”. (1973: 79). El videojuego está emparentado con el cine, pues gracias a la persistencia de la imagen, también podemos ver el video; sin embargo, éste corre a una velocidad distinta (en promedio, de 30 a 60 cuadros por segundo). Entonces, apariencia y movimiento se juntan para invitar al jugador a participar y permanecer en el mundo ficticio.

Pero sólo eso no basta; para que el usuario sepa identificar y mantener el control dentro de las mecánicas del juego, es necesario armar una buena interfaz.

2.1.2 Diseño de interfaz

Conocer la lógica de una estructura visual es imperativo en este caso, pues constituye la forma de comunicación entre el código programado y el usuario. Mark Peasley —veterano en arte aplicado a videojuegos— afirma que una interfaz debe ser predecible, consistente e informativa para ser exitosa, al mismo tiempo que aboga por el estilo minimalista, ya que, según comenta, ha resultado ser el más atractivo y funcional. (Rabin, 2010: 648).

Por otro lado, también es importante conocer a fondo el concepto de **interactividad**, el cual se refiere a la relación dialógica entre usuario y sistema. Si no se es claro en la estructura interactiva, el público puede frustrarse y perder interés o incluso desarrollar aversión.

Los primeros juegos, en sus inicios de 8 bits, mostraban la información necesaria en una sola **interfaz**, lo cual significaba perder visibilidad, pues las interfaces tenían que ser grandes para ser claras, ocupando al menos 10% de la pantalla. Sin embargo, a medida que la tecnología se desarrollaba, los juegos se hicieron más complejos y surgió la exigencia de construir y desplegar más datos en las interfaces. Al retomar las palabras de Zuur y Dille, acerca de la necesidad del jugador de tener el control sobre una situación, hay que concluir que, mientras

más constantes pueda dominar, menores serán las probabilidades de que se harte.

Las nuevas interfaces creadas en los últimos juegos tienden a reducir su tamaño al máximo y suelen ocultarse cuando no se las necesita, de manera que no afecte la **jugabilidad** de forma negativa. Por ejemplo, Thompson (2007:118) señala que para mostrar información de texto (que no depende del jugador y es parte de la narrativa del juego), debe situarse en la parte central inferior de la pantalla, pues esto indica diálogo interno —según estudios de programación neurolingüística—; en la parte superior o esquinas de la pantalla, por lo general se encuentran datos que necesitan revisarse de manera asidua como los puntos de vida, el número de municiones o los mapas. Es decir, la posición ocular revela ejercicio de memoria cuando mira hacia arriba, y ejercicio de introspección cuando se mira hacia abajo.

Lo cierto es que la forma de presentar información ha cambiado para optimizar las formas de juego. Los acercamientos de los nuevos diseñadores a esta parte muestran una intención de desplegar la información mínima indispensable, tanto para ahorrar espacio como para no interrumpir el ritmo de juego. Al respecto, Omernick comenta: “El trabajo de un diseñador de interfaces es uno que requiere no sólo una fuerte noción del diseño, sino también la habilidad de pensar lógicamente y entender la psicología de la interacción del jugador/juego”. (2004: 247). Asimismo, describe la importancia de los menús de inicio, considerando que son las primeras imágenes antes de comenzar el verdadero juego y también el primer umbral interactivo para captar aún más su atención. A estas primeras imágenes que ayudan a configurar y controlar el juego son llamadas **shell** (coraza) y es de vital importancia darles un estilo coherente con el resto del juego.

Entonces, ya que lo importante es poner orden, Peasley recomienda hacer diagramas de flujo para tener claro cómo será la estructura básica de las interfaces. (Rabin, 2010: 647-649). A continuación, haré una breve muestra de tres juegos contemporáneos que ejemplificarán muy bien estas nociones:



29. Menú inicial de *The Binding of Isaac*.

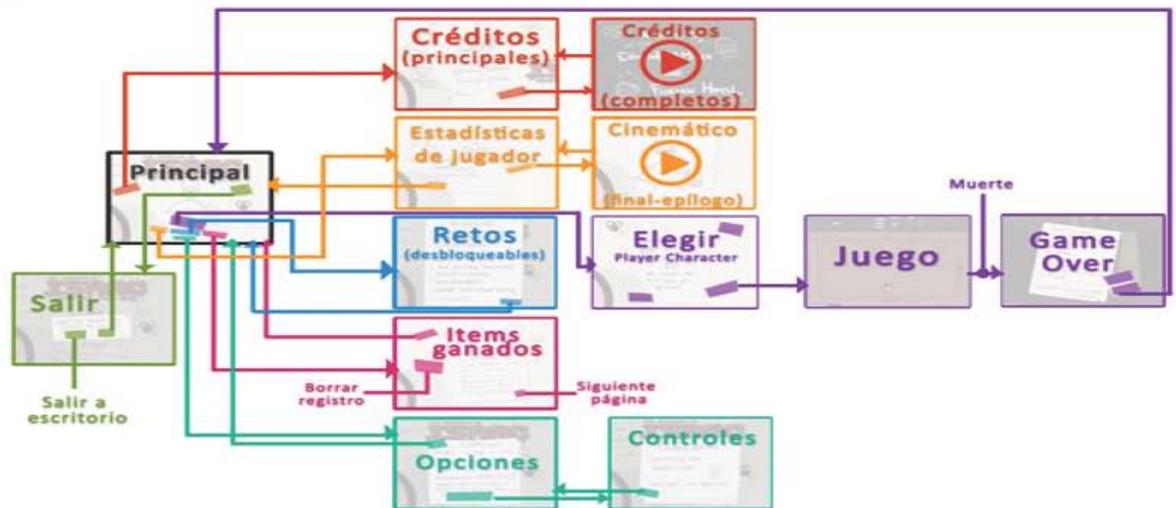
THE BINDING OF ISAAC (2011, Edmund McMillen & Florian Himsl)

Ya conocimos la historia y apariencia general de *The binding of Isaac*; ahora, veamos cómo funciona a nivel de interfaces. A grandes rasgos, se trata de algo muy simple, pero justamente por esa razón es funcional y consistente, porque no hay manera de perderse:

Como diseñadores y al momento de repartir funciones entre interfaces, una herramienta tan sencilla como un código de colores resulta muy útil para entender el circuito y buscar nuevas formas de hacerlo más “intuitivo”.



30. Todos los menús y submenús.



31. Diagrama de flujo en donde se analiza el funcionamiento de las interfaces.

TINY&BIG: GRANDPA'S LEFTOVERS (Black Pants Studio)

Ya conocemos también la historia y apariencia de Tiny&Big: Grandpa's Leftovers y aquí podemos apreciar que el botón de "Salir" en el menú principal y "Atrás" en el submenú de "Seleccionar nivel" tienen la misma configuración y una ubicación similar (imágenes 32 y 33).

Un elemento clave de una interfaz funcional es su estructura. Omernick destaca el hecho de que, como diseñador, se deben tomar en cuenta las necesidades del jugador en plena acción de juego (2004: 249). En el caso de elementos como barras de vida, puntaje, inventario, etc. o **HUD** (*Heads Up Display*) que no han de permanecer en la pantalla, no sólo deben ser claros, sino constantes. Por ello, es tan valioso que la información de texto aparezca en un módulo que no cambiará de posición, pues una vez aprendida la estructura, el modo de juego se optimiza.

DEAD SPACE (Electronic Arts / Redwood Shores)

Una interfaz muy ventajosa es la del juego *Dead Space*. Los datos indispensables sobre la información de Isaac, el **PC**, pueden verse con claridad sobre su espalda. Las barras verdes luminosas en la columna de su traje indican los puntos vitales del personaje, mientras que el medio círculo indica el "poder" para "ralentizar" el movimiento de objetos aislados se mueven continuamente e impiden su paso por ciertas áreas (imagen 34). La disposición de datos indispensables es lo bastante práctica como para no se bloquee una parte del juego de manera permanente. Para saber sobre los datos de armas e inventario (imagen 35), éstos deben activarse a nivel de **output**¹² y se verán sus respectivas interfaces que no ocupan toda la pantalla. Descomponer la información necesaria sobre mapas, inventario, misiones, mejoras en equipo e información vital del **PC** como lo expone este juego, es una forma muy práctica de mostrarla al jugador sin que interrumpa el **game-play**.

12 Información que genera el programa o juego para el usuario o jugador.

32. Screenshot del menú principal donde se hace evidente un proceso de localización¹³

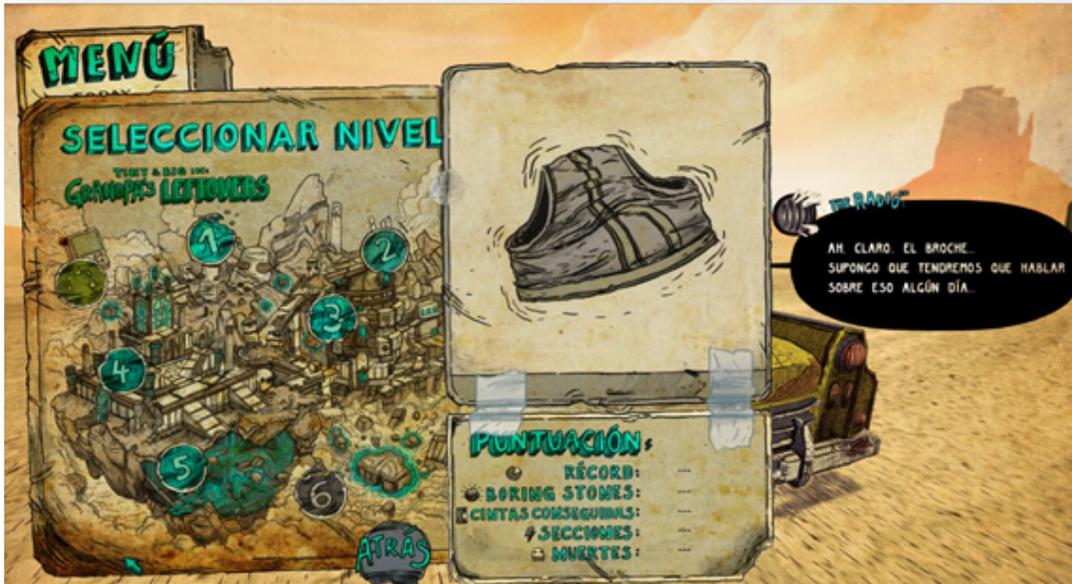


Mark Peasley (2010: 647) comenta que los aspectos cruciales en el diseño de interfaces son:

- Simplicidad
- Consistencia
- Conocer al usuario **target**
- Uso de color
- Mecanismos de retroalimentación (mecanismos que confirmen al usuario que realizó una acción o que está en proceso).

Ahora, procedamos a conocer un poco sobre los procesos de modelado, base de la animación tridimensional.

13 El texto editable es traducido en su totalidad a otro idioma, para que el jugador no tenga dificultades y se sienta más cómodo al manejar controles en su lengua materna.



33. Screenshot de los menús de Tiny&Big: Grandpa's Leftovers.

2.1.3 Modelado tridimensional

No es de sorprender que encontremos, entre las herramientas de un software de modelado en 3D, algunas de las composiciones formales del diseño: traslación, rotación y escala (o dilatación). Éstas sirven como introducción al modelador 3D para comprender la lógica de modelado y para saber qué es lo más óptimo, rápido y útil, ya que es posible modelar con diferentes procesos para llegar a un mismo resultado.

Pero también tenemos otros elementos básicos de la comunicación visual en los componentes de una figura geométrica. Por ejemplo, al partir de un cubo al modelar un objeto, tenemos el punto (vértice), la línea (el borde o arista) y el plano (cara o polígono). Estos nombres pueden variar de un software a otro, pero la base es la misma.

Hay distintas técnicas para poder crear un modelo en tres dimensiones y, en efecto, hay una amplia gama de softwares que podemos utilizar como herramienta para llevar a cabo esta tarea, incluyendo software libre (Blender) y comercial (Maya y 3ds Max de Autodesk, entre muchos otros).

34. Screenshot de Dead Space donde se aprecia la barra de “vida” y el “poder” en la espalda del personaje principal.



Los llamados “primitivos” en un software de modelado en 3D son las formas básicas sobre las cuales se trabajará un modelo (esfera, cubo, cilindro, plano, cono, etc.). Éstas no son elaboradas por el modelador, aparecen por defecto en el programa y el usuario sólo debe indicar dónde van a crearse las figuras, de qué altura, grosor y profundidad serán.

Omernick (2004: 18-19) hace la analogía de los “primitivos” con las figuras básicas en un dibujo; menciona el ejemplo en que un maestro de dibujo indica a sus alumnos, en una clase con modelo, que se concentren en las formas básicas que componen un cuerpo humano.

Los elementos de un primitivo, recordemos, son: vértices, aristas, triángulos y caras. Las herramientas de transformación de un polígono son: escala, traslación y rotación.

Una de las técnicas más utilizadas es el box modeling o “modelado de caja”, el cual consiste en partir desde la figura de un cubo y transformarlo, esculpirlo: agregar nuevas aristas, extruir caras, rotarlo, trasladarlo o escalar elementos, etc.



35. Screenshot de Dead Space donde se aprecia el despliegue del inventario.

Como afirma Dondis: “El conocimiento en profundidad de la construcción elemental de las formas visuales permite al visualizador una mayor libertad y un mayor número de opciones en la composición; esas opciones son esenciales para el comunicador visual” (1973: 54). En este caso, sólo debemos sustituir “visualizador” por “modelador” o “artista 3D”.

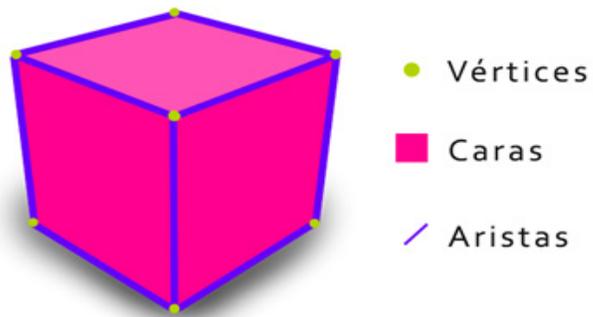
En este punto, el equipo de artistas 3D se ha de apoyar en el concept art, en las ilustraciones y planos que los artistas realizaron para la creación de modelos. Con el fin de facilitar la tarea, suelen utilizarse las vistas frontal, lateral (y, en ocasiones, la superior) de un objeto o personaje a manera de guía.

Antes de comenzar, siempre es necesario establecer los parámetros técnicos que un modelo debe cumplir para ser insertado en el juego. Por lo general, los tipos utilizados hoy en día son los modelados en low-poly (con pocos polígonos) para ahorrar memoria y evitar que el juego cargue demasiada información innecesaria (imagen 39).

14 <http://shemdawson.blogspot.mx/2012/09/hand-painted-low-poly->



36 y 37. Logos respectivos de los software mencionados.



38. Elementos básicos de un primitivo.



39. Arma modelada en Low-Poly por el artista 3D Shem Arma modelada en Low-Poly por el artista 3D Shem Dawson de su portafolio personal.¹⁴

Un término indispensable en esta área es el de poly-count o el conteo de polígonos. Dependiendo de la plataforma hacia la que se orienta el juego (PC, consola, celular, web, etc.), se decidirá cuántos polígonos puede tener un personaje, que por lo general, son los objetos más complejos en un juego. También, dicho conteo está sujeto al motor de juego elegido, al número de personajes en escena, el nivel de detalle y más. Esta decisión corresponde a los directivos del departamento —el Lead Programmer, el Art Director y el productor—, quienes toman en cuenta el número de polígonos del resto de objetos y personajes en la misma escena.

Ya que el ahorro de memoria es tan importante, un modelado de objetos inmóviles, como los de utilería, pueden prescindir de las caras que no serán vistas desde ningún ángulo por el jugador (como las caras que dan al suelo).

En este sentido, veamos algunos ejemplos de trabajos extraordinarios:

Pid (2012, Might & Delight)

Kurt, el niño protagonista — al igual que el personaje principal de McMillen—, tiene proporciones pequeñas y redondead-

40. Screenshot de una de las escenas iniciales de Pid.



as, lo que contrasta con el tamaño del resto de los personajes. Los modelos creados para este juego aprovechan de forma singular la restricción de ser *low-poly* al mostrar un mundo que parece recortado de cartón, pero con mayor profundidad que *Lume* y una profundidad de campo más intensa, haciendo lucir algunos ambientes como si fueran parte de un sueño (imagen 40). Así se describe en el sitio oficial del juego (2013):

Pid is a platform game that will turn everything you know about 2D gameplay upside down. You play as a young boy that gets stranded on an old remote planet. He must fight off a variety of malicious robots bent on stopping him and befriend unlikely allies to shed light on a huge conspiracy that keeps the planet mesmerized and prevents him from ever reaching home.¹⁵

15 “Pid es un juego de plataformas que pondrá de cabeza todo lo que sabes sobre reglas de juego en 2D. Juegas siendo un niño pequeño que se pierde en un remoto y viejo planeta. Debe luchar contra una variedad de robots maliciosos decididos a detenerlo y hacerse amigo de inesperados aliados para arrojar luz sobre una enorme conspiración que mantiene al planeta asombrado y lo detiene de llegar a casa.” La



41. Screenshot de Unmechanical.

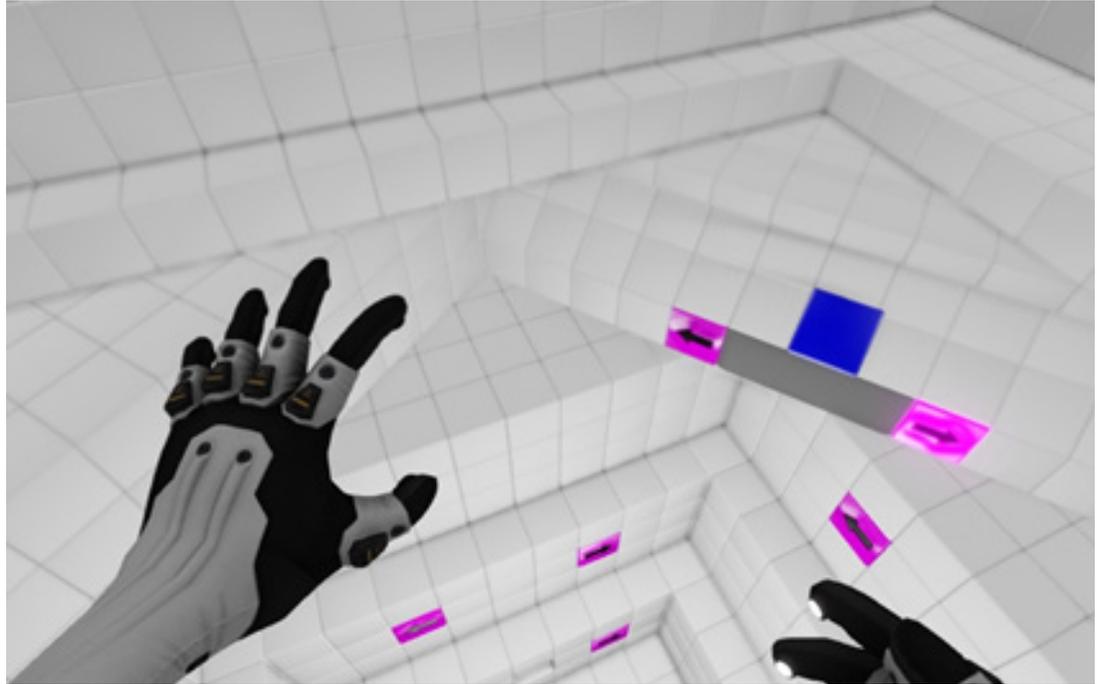
Unmechanical (2012, Teotl)

Este juego es catalogado por los desarrolladores (2013) como un juego *side-scroller* —es decir, donde hay libertad de movimiento hacia los lados, arriba y abajo— en 2.5D (hecho en 3D, pero sin poder ver los ángulos que no capta la cámara desde la que el jugador aprecia el juego), lleno de acertijos y misiones para recorrer niveles laberínticos con un robot pequeño que tiene la habilidad de flotar, además de atraer objetos para poder cargarlos, jalarlos o empujarlos.

Este juego fue desarrollado con **UDK** y aprovecha los detalles que ofrece su calidad gráfica en texturas sin tener que aumentar la complejidad de la geometría de los modelos, especialmente porque el juego fue hecho para funcionar en

traducción es mía.

42. Screenshot de uno de los niveles de Q.U.B.E.



dispositivos móviles, y eso requiere mayor nivel de optimización.

Q.U.B.E. (Quick Understanding of Block Extrusion. 2012, Toxic Games)

No tiene una trama como tal, sólo se centra en el ingenio de sus mecánicas de juego. El **PC** se encuentra en un cuarto blanco hecho, en apariencia, de cubos blancos. Conforme el jugador avanza por el nivel, descubrirá que algunos cubos — de distinto color— poseen propiedades interesantes que le ayudarán a sortear obstáculos.

Parte de las características visuales del juego es que la única parte del **PC** que puede verse en cámara son sus antebrazos y manos, se ven muy cerca de la cámara y se ven casi siempre. Es por ello que Harry Corr, el artista encargado, esculpió con el mayor detalle posible estas partes del personaje.

16 <http://harrycorr.co.uk/qube/qube-glove-high-rez-sculpt.jpg>



43. Modelos de antebrazos del personaje principal creados por Harry Corr, artista subcontratado para Q.U.B.E.¹⁶

El siguiente paso en la producción de un objeto en 3D es su texturizado, ya que puede contener información que indique al jugador elementos vitales de la historia.

2.1.4 Texturizado

La aplicación de texturas es esencial para profundizar la inmersión del jugador, pues éstas son el segundo elemento —después de los modelos— que definen el grado de realismo o caricaturización del juego. Y algo muy importante: cuando el grado de detalle en polígonos de un modelo es bajo, las texturas pueden compensarlo.

En este caso, Omernick (2004:45) subraya lo importante que es distinguir textura y shader. Un shader es un pequeño programa que define las características de un material: qué tan brillante es, qué tanto puede reflejar objetos en su superficie, su opacidad, etc. Una textura es sólo un archivo de imagen conectado al *shader* que define algún aspecto de la imagen —brillo, color, rugosidad— que queramos que aparezca en el modelo.

Omernick también establece que, parte de ser un buen artista de texturas es pensar en la historia detrás del objeto y sugiere que el artista se haga preguntas como: “What mate-

rial is it? Wood? If so, what type? How old is the door? Who uses it? [...] What else can we add or exaggerate in order to make it more interesting?”¹⁷ (2004: 47).

El tipo de **render** (porque hay varios) que aplica en videojuegos es llamado “renderizado en tiempo real”, esto significa que el dispositivo (consola, computadora, teléfono, etc) en que funciona el juego debe mostrar las imágenes como si fuese video, pero con la diferencia de que el jugador pueda interactuar en tiempo real. Es una de las ramas de la computación gráfica.

El papel de las texturas es tan importante porque —como se menciona más arriba— debe compensar los detalles que el modelo tridimensional no puede mostrar por su baja cantidad de caras.

Ahora bien, para texturizar un modelo 3D, hay que mencionar que todas las **geometrías** o modelos cuentan con información de coordenadas **UV**. Omernick ilustra el concepto: “When you apply a texture to geometry, you are wrapping a 2D image around a 3D model. UVs are the coordinates used to tell the computer where to place the texture in 3D space by taking each point of the model (vertex) and assigning it to a point on the map.”¹⁸ (115).

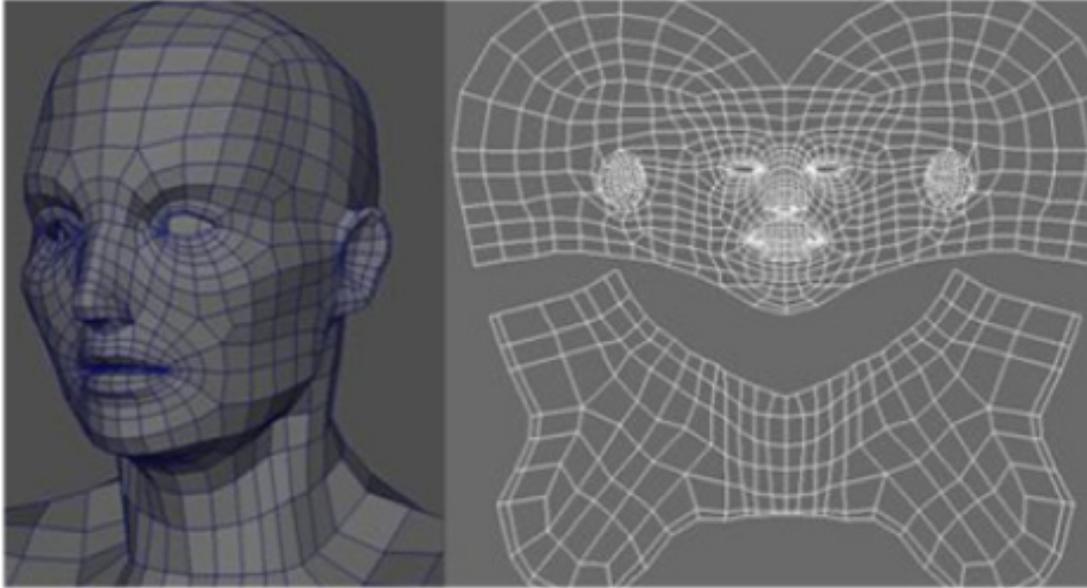
Si estos datos aún suenan un poco vagos, veámoslo aplicado en la imagen 44:

Entre los materiales o texturas, es necesario destacar tres principales: **diffuse map**, **normal map** y **specular map**. Para definir cada uno, utilizaré la información que ofrece el sitio oficial de Recursos Educativos de UDK:

The diffuse color of a material represents how much of the incoming light reflects equally in all directions. A value of (1,1,1) means that 100% of the incoming light reflects in all directions. Diffuse is not view dependent so it will look the same from

17 “¿De qué material está hecho? ¿Madera?, Si es así, ¿de qué tipo? ¿Qué antigüedad tiene? ¿Quién lo utiliza? [...] ¿Qué más podemos agregar o exagerar para hacerla más interesante?”. La traducción es mía.

18 “Cuando aplicas una textura a una geometría, estás envolviendo con una imagen en dos dimensiones a un modelo en tres dimensiones. Las UVs son las coordenadas usadas para decirle a la computadora dónde ubicar la textura en el espacio 3D al tomar cada punto del modelo (o vértice) y asignándolo a un punto del mapa.” La traducción es mía.



44. Modelo de busto humano en 3D en la parte izquierda de la imagen. A la derecha, vemos el mapa de coordenadas UVs, o sólo UV map del mismo modelo. Luce curioso porque es como si el rostro estuviera “aplastado”; sin embargo, esto es usual al “desenvolver” las coordenadas UVs (una imagen plana) de un modelo en 3D. Hay que tener cuidado de elegir en qué aristas o bordes se hará el corte de UVs, para que queden planas con la menor deformación.

any angle, but it does depend on the normal at each pixel. Diffuse color will only show up when affected by some un-shadowed lighting, because it scales the incoming lighting¹⁹ (2013)

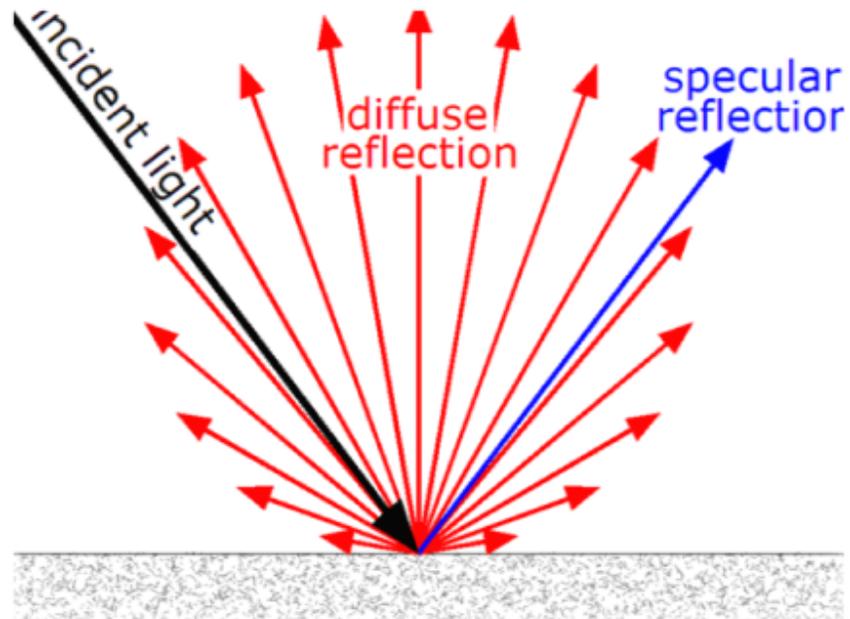
Esto significa que el *diffuse map* representa el color de un objeto; por ejemplo, si una piedra es café y tiene manchas. La rugosidad, brillo u opacidad serán tareas de otros materiales.

El material de **specular** se define así:

Specular color represents how much of the incoming light reflects in a directional manner. The specular response is brightest when your eye lines up with the direction of the reflected incoming light, so

19 “El color difuso de un material representa cuánto de la luz que se recibe se refleja equitativamente en todas direcciones. Un valor de (1,1,1 -en 3 ejes, de X, Y y Z-) significa que 100% de la luz se refleja en todas direcciones. Lo difuso se verá igual desde cualquier ángulo, pero sí depende de la normal en cada pixel. El color difuso sólo se mostrará cuando se vea afectado por iluminación sin rastros de sombra, porque aquél escala la luz que recibe.” La traducción es mía.

45. En este ejemplo²¹, podemos ver cuál es el efecto de cada caso, en el que el color difuso luce de igual forma, sin importar la dirección desde la que se vea la superficie, mientras que el color de especular es más intenso en sólo una dirección en específico, obedeciendo el ángulo que forme la luz al incidir con el objeto.



specular is view dependent. The normal also affects specular because it affects the reflected light direction. Specular power controls how shiny or glossy the surface is. A very high specular power represents a mirror-like surface while a lower power represents a rougher surface²⁰ (2013)

Estos valores altos son representados por el blanco y escalas de grises cercanos al blanco. Los valores bajos corresponden al negro como el valor más opaco y escalas de grises cercanas al negro (imagen 45).

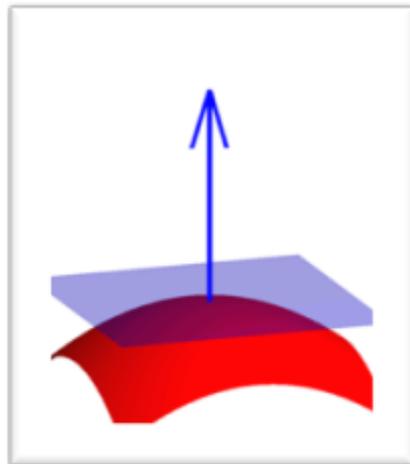
Finalmente, el material de **normal** es definido por Omer-nick de la siguiente manera:

20 “El color especular representa cuánto de la luz recibida se refleja en una forma direccional. La respuesta especular es más brillante cuando el ojo se alinea con la dirección de la luz dirigida hacia el objeto. Las normales también afectan el valor del material de especular, porque afecta la dirección de la luz reflejada. La potencia del especular controla cuán brillante o lustrosa es la superficie. Un valor muy alto de especular representaría una superficie similar a la de un espejo, mientras que un valor muy bajo, representaría una superficie más opaca.” La traducción es mía.

21 http://en.wikipedia.org/wiki/Diffuse_reflection

Normal map generation tracks the surface normals of a high-poly model, and encodes that high-detail normal information to every pixel in the normal map for use on a low-poly mesh. So when light falls across the object, it appears to be extremely high in detail. Think of it as an invisible, high-detail shell around the low-poly object used to provide lighting information.

[...] Normal maps use RGB values to provide X, Y, and Z information to calculate lighting and surface properties, giving a more realistic and believable illusion.²² (2004: 195)



46. Representación gráfica de un vector normal. La flecha azul indica la dirección del vector normal que corresponde a ese solo punto²³

22 "Al generar mapas de normales se ubican los vectores normales de un modelo high-poly o de una gran cantidad de polígonos, y se codifica la información de vectores normales de todo ese detalle a cada pixel en la textura del mapa de normales para su uso en una geometría de pocos polígonos. Así, cuando la luz cae sobre el objeto, da la impresión de tener mucho detalle. Piensen en él como una coraza invisible, con mucho detalle que envuelve al objeto de pocos polígonos, y se usa para proveer de información para iluminación.

[...] Los mapas de normales usan valores de RGB para proveer de información a los ejes de X, Y y Z para calcular iluminación y propiedades en la superficie, dándole una ilusión mucho más realista y creíble." La traducción es mía.

23 Imagen obtenida de: [http://en.wikipedia.org/wiki/Normal_\(geometry\)](http://en.wikipedia.org/wiki/Normal_(geometry))

Para aterrizar la teoría, veamos los tres mapas funcionando en el modelo de un revólver en 3D:

46 y **47**. Revólver Desert Eagle modelado por Irakli Khakhviashvili. En la imagen de arriba, está el modelo renderizado en tiempo real, mientras que en la de abajo, se ve la “estructura de alambre” o wireframe de los polígonos de que está compuesto el modelo, con 3800 tris. En el recuadro de la derecha, vemos los mapas de diffuse —los colores del arma—, de normals —el mapa que provee de detalle para iluminación— y el specular —donde, en los valores blancos, está la información que indica mayor brillo en el arma, y en los grises, superficies más opacas.



El tamaño o resolución de una textura siempre debe ser en múltiplos de dos, con el fin primario de ahorrar memoria, tomando en cuenta que las computadoras funcionan a partir de un código binario. Los tamaños de las texturas pueden ser:

Tamaños de texturas normalmente aceptadas en videojuegos (Omernick, 2004: 51)
32x32
64x32
64x64
128x64
128x128
256x128
256x256
512x256
512x512
1024x1024
2048x2048

Los formatos de imagen para este rubro son: **.TGA** (Targa), **.TIFF** (Tagged Image File Format), **.PNG** (Portable Network Graphics), **.BMP** (Windows Bitmap)... todo depende del motor de juego.

No hace falta hacer una ilustración para crear texturas; es frecuente partir de fotografías que serán proyectadas sobre los modelos²⁴ (aunque, desde luego, deben ser originales o libres de derechos de autor). También es común aplicar imágenes con transparencia para evitar hacer un modelo más complejo y, por tanto, con más polígonos. Por ejemplo, para hacer pasto, se puede crear un plano con dos o tres divisiones en total (o ninguna, dependiendo del largo del pasto), y se aplica una imagen con un formato que acepte el cuarto canal para la transparencia. Además de los colores luz normales de

24 En ocasiones, una textura deberá repetirse varias veces en un mismo objeto para ahorrar memoria y no hacerla de tamaño completo. Para este fin, hay que cuidar que los bordes sean iguales entre sí para evitar divisiones muy notorias entre cada "mosaico" de textura.

R (*Red=rojo*), G (*Green=verde*), y B (*Blue=azul*), tendremos 8 bits extra para agregar el canal alpha (utilizado como mapa de transparencia), sumando así 32 bits. Los formatos de imagen que lo aceptan son **.TGA**, **.PNG** y **.TIFF**.

Una imagen con sólo el canal alpha habilitado se verá en blanco, negro y escala de grises. El blanco muestra la parte más opaca de la imagen y el negro, la transparente al 100%.

La aplicación de texturas depende de los UVs. Hagamos esta analogía: tenemos una caja de regalo en forma de cubo, pero es blanca y queremos darle color. Entonces tomaremos papel estampado para envolver y lo doblaremos y pegaremos para que, a pesar de ser un plano, se adapte a la forma de la caja. Esto mismo sucede con la aplicación de texturas.

Por último queda abordar el tema de la iluminación.

2.1.5 Iluminación (fotografía)

Un diseñador gráfico que tenga nociones de fotografía, ilustración e iluminación, tiene todos los elementos necesarios para aplicarlos a una escena de un videojuego. Matthew Omernick (2004: 158) recomienda que se piense en la escena como una composición total, donde deberán priorizarse las siluetas. Además, señala la importancia de conocer al público, pues el uso de colores en un juego dependerá de la cultura que el jugador potencial tenga, pues la percepción no es siempre la misma. En Occidente se tiene un color bajo un concepto y en Oriente, en uno diametralmente opuesto (no siempre, pero llegan a haber casos así). Además, el autor indica que, con los conocimientos necesarios sobre teoría del color, es posible transmitir al jugador el preciso sentimiento que el **game designer** pretende. Éste es un factor crucial para obtener una mayor inmersión; sin el aspecto emocional que da el color y las luces y sombras en una escena, la experiencia pierde intensidad.

En el capítulo “Lighting Principles”, Omernick (166) señala los tipos distintos de luces que pueden encontrarse en una escena genérica de fotografía (además del trabajo hecho con luces 3D): luces direccionales, ambientales, reflectores, puntuales u *omni* y la clásica iluminación global.

Las luces direccionales son utilizadas frecuentemente en escenas 3D para simular la luz solar o de luna. Se usa como

iluminación base por ser intensa y constante. También es frecuente oír el término “luz paralela” para referirse a este tipo, ya que es parecida a la forma en que el sol ilumina la Tierra²⁵.

Ahora, veamos algunos ejemplos donde la luz juega un papel fundamental en el desarrollo y comprensión de un videojuego.

The Dream Machine (2012, Cockroach Inc.)

Victor Neff es el protagonista en *The Dream Machine*. Él y su esposa embarazada se mudan a un nuevo departamento y se encuentran con un casero y vecinos extraños. Se trata de un juego de tipo policíaco y de misterio, no sólo por los escenarios, sino por sus personajes, los cuales parecen ser sombríos y con mucho qué esconder. La mecánica del juego es simple: investigar objetos, dialogar con los otros personajes y recoger pistas.

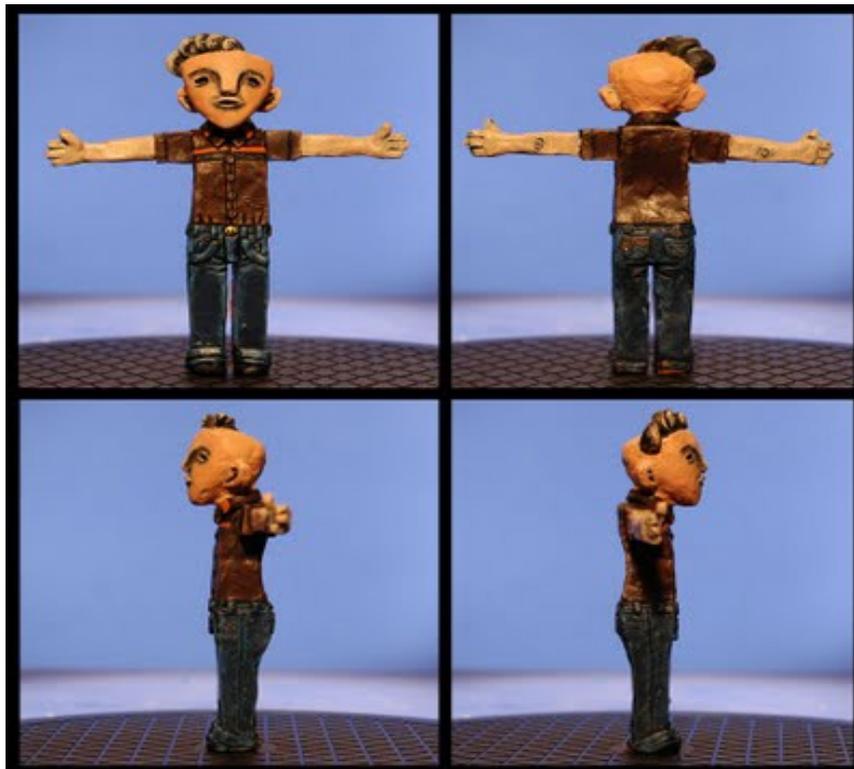
Comparte el mismo carácter de técnica tradicional con *Lume*, pero en vez de tener elementos hechos de cartón y papel, tiene masilla pintada como componente adicional. Anders Gustafsson —uno de los desarrolladores— relata el proceso de creación de personajes (2009). Comenta al respecto que después de que Erik Zaring —el encargado del arte— terminara de hacer los personajes en masilla, se le tomaban fotografías al modelo en diferentes vistas. Así, estas fotografías eran proyectadas sobre un modelo 3D digital como textura, para animarse después.

Se eligió un tipo de iluminación constante sobre el modelo, para que funcionara bien en los diferentes escenarios. Primero, se necesitaba obtener las imágenes del personaje en los diferentes ángulos de las numerosas acciones que iba a efectuar; luego, eran descompuestas en “pases” —aspectos aislados en imágenes diferentes de una escena digital iluminada.

Las imágenes de cada pose eran post-procesadas en programas de manipulación de imagen como **Photoshop** y **After Effects**. Después podían ser incorporadas en el juego, que funciona como un archivo interactivo de Adobe Flash, un programa muy popular para animaciones en vectores y aplicaciones virtuales interactivas.

²⁵ Dado que ambos están tan lejos una del otro, los rayos de luz que emite el sol se hacen más paralelos entre sí, y la luz es más homogénea a la superficie que toque. (166-168).

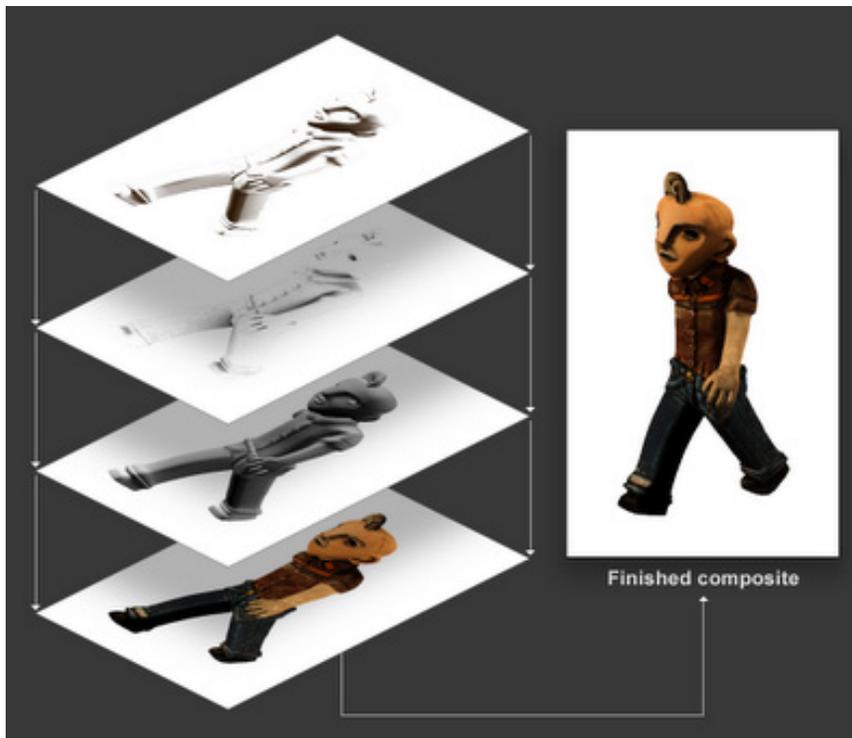
48. Vistas frontal, trasera y laterales de Victor Neff²⁶



Esta imagen muestra los pases que componen la imagen final. La de la parte superior corresponde a una iluminación “dura” producida por luz direccional, creando sombras más oscuras y contrastadas. La siguiente hacia abajo es lograda por **ambient occlusion**, así que, en este caso, las zonas sombreadas son las que están menos expuestas a la luz por objetos aledaños que bloquean el paso de iluminación general (por ejemplo, el espacio entre las piernas y debajo de los brazos). La siguiente pertenece a la iluminación de tipo **environment** o ambiental. Ésta se logra con un domo que ilumina desde todas direcciones al modelo en cuestión. En ocasiones el domo está tapizado con una imagen panorámica, lo que ofrece una iluminación menos plana. La última del fondo sólo contiene la información de la textura de **diffuse**, sin ninguna iluminació

26 <http://dreammachinegame.blogspot.mx/2009/11/character-generation.html>

27 <http://dreammachinegame.blogspot.mx/2009/11/character-generation.html>



49. Pases de la imagen final de Victor Neff.²⁷

La imagen final del objeto, si tiene cada uno de estos componentes por separado, es mucho más fácil modificar y corregir.

Journey (2012, thatgamecompany)

Journey tuvo gran reconocimiento a nivel internacional por su innovación respecto a la experiencia que ofrece al jugador. El objetivo que debe cumplir éste con su PC es avanzar en una travesía —acompañado de otro jugador en línea al azar— para conocer un mundo lleno de arena y momentos de introspección. Los distintos niveles proponen una atmósfera distinta en cada ocasión. Al inicio del juego, los escenarios tienen mucha luz (imagen 50), edificios en ruinas y varios momentos de experimentación con las habilidades del personaje propio (hablar para crear una onda expansiva que activa otros seres y cosas; o flotar, gracias a la interacción del personaje con pedazos de tela que ondean al viento).

Existe la variación de entornos a partir de diferentes pale-

50. Screenshot de uno de los escenarios luminosos de Journey.



tas de colores y grados de iluminación (imagen 51). El material del suelo, en esencia, no cambia: es y sigue siendo arena y hay dunas alrededor; sin embargo, al inicio, la arena es de un color ocre —como se puede ver en la ilustración— y hay luz de medio día. Pero conforme se avanza, la iluminación es más escasa. “Anochece” y los contrastes se tornan más evidentes.

Obedeciendo a una paleta nocturna, el entorno se torna azul-verdoso e incluso la arena se vuelve de tono azulado por la luz y parece agua al deslizarse sobre ella. Sobre el suelo aparecen entes hechos de tela que tienen aspecto de medusas y peces (imagen 52). Los últimos niveles lucen una mayor armonía de colores y la luz se convierte en un elemento casi mágico. Como si toda la aventura hubiese durado tan sólo un día, desde el sol intenso de la tarde hasta la noche, la madrugada y la mañana limpia del día siguiente. Los niveles de noche y madrugada son los más “oscuros” en temática; hay oponentes que causan más daño y hay mayor presión hacia el jugador para seguir avanzando. Como el ambiente debe imponer esta tensión, es coherente que los elementos en color sean fríos (para agua y después nieve). Al final, parece que todos los colores han retornado a la normalidad y hay una mayor gama de éstos, además de que son más saturados, especialmente el color del cielo.

The world of Journey is poetic, with the beginning on the left and the horizon goal on the right. In the north-south plane, the game had only one direction: north. The final game played north-south.



51. Paleta de colores respecto a la progresión de niveles de Journey.



52. Screenshot de una medusa.

Capítulo 3

Proyecto Celleste

Aunque yo como diseñadora no haya trabajado en todos los aspectos de la creación de arte de este videojuego, se explicará brevemente cómo fue trabajado por otros miembros del equipo para tener un desglose más integral del arte del proyecto. Celleste es un juego para iPad, realizado durante más de un año en el cual hubo muchos ensayos y errores que permitieron lograr óptimos resultados. Desde la concepción del proyecto hasta la fecha se pasó por varias propuestas visuales, muchas de las cuales no permanecieron en la que ha sido la última de las versiones hasta el momento, pero sirvieron para trabajar sobre ellas para concebir mejores ideas.

3.1 Concepto general

En un punto recóndito de una galaxia, existe un planeta solitario que rebosa de vida.

El jugador tiene la misión de proteger una especie de ese planeta: inusuales y pequeñas vacas que deben atravesar el planeta hasta llegar a un portal y que necesitan alimentarse de lechugas para subsistir. Los enemigos naturales de estas vacas son insectos que las igualan en tamaño y pretenden comérselas. Pero el jugador puede evitarlo al poner trampas de hielo para congelarlos momentáneamente, lanzarles asteroides o al ubicar agujeros negros que los absorben. El reto es poder controlar la existencia de estos bichos en el planeta, pues constantemente salen de sus guaridas y, por tanto, su número aumenta. Todas estas son reglas del juego que finalmente se tituló *Celleste*.

3.2 Antecedentes

Este proyecto comenzó con una lluvia de ideas sobre mecánicas y conceptos generales de juego y, con cada aportación, todo adquirió una forma definida. Gerardo Basurto —ingeniero en computación del Tecnológico de Monterrey— es el **Game designer** y líder del proyecto, además desde un principio financió la participación de todos los miembros del equipo. El principio no fue fácil, ya que, para la mayoría, se trataba de trabajar sobre un primer juego “desde cero” sin apoyo de una empresa, y por ello, los métodos cambiaron progresivamente de acuerdo con las necesidades del proyecto.

La encargada de Producción, Stephanie Prodanovich —Licenciada en Comunicación Audiovisual de la Universidad del Claustro de Sor Juana— y Basurto establecieron como punto de partida que el juego pertenecería al género de *Arcade Action*, para plataforma *iOS* en dispositivo de *iPad* y usaría *Unreal Engine 3* como motor.

Basurto y Prodanovich —ambos ex-empleados del desaparecido estudio mexicano *Slang Studio*, sucesor de *Sabarasa México*— después me contactaron para desarrollar algunas partes del aspecto visual, especialmente para investigar sobre otros juegos exitosos, para analizar el **arte** y presentar sus ventajas e incluso proponer nuevas ideas con el fin de armar una línea de producción adaptada a las necesidades del pro-

yecto con base en técnicas y teoría aplicada a juegos de video. Pero, poco a poco se hizo evidente que era necesario un equipo más grande. Pronto, se unió Fabián Hernández —quien ya contaba con experiencia en otros estudios¹— como **Technical artist**, por su vasto conocimiento sobre **UDK**. En el área de programación, José Medina² entró para apoyar con la parte de inteligencia artificial.

La primera fase de ilustración corrió por cuenta de Quetzal Cárdenas; sin embargo, por variadas circunstancias, no podía asegurarse su permanencia, lo que provocó un periodo ambiguo y un significativo retraso en la producción. Finalmente, se contactó a un nuevo ilustrador, Juan Pablo Riebeling, quien presentó bocetos más cercanos a la visión del **game designer** y así, estaba disponible el **Art Bible** para proceder con el modelado de algunas **props**. Dicho cargo fue dividido entre varios miembros del equipo; entre ellos, yo. Y para el área de diseño de imagen como logo e interfaces, se subcontrató a más diseñadores.

3.3 Organización del equipo de trabajo

Para nuestra fortuna, no fue necesario contar con un espacio físico, pues el trabajo y su registro se llevaban a cabo en la aplicación **Drive**³ de Google, a la que sólo los miembros principales del equipo tenían acceso. Por otro lado, también se utilizó el sitio **Trello**⁴ para llevar una organización coherente y dividir las tareas en tres categorías: “Por hacer”, “En proceso” y “Realizado” (imagen 53).

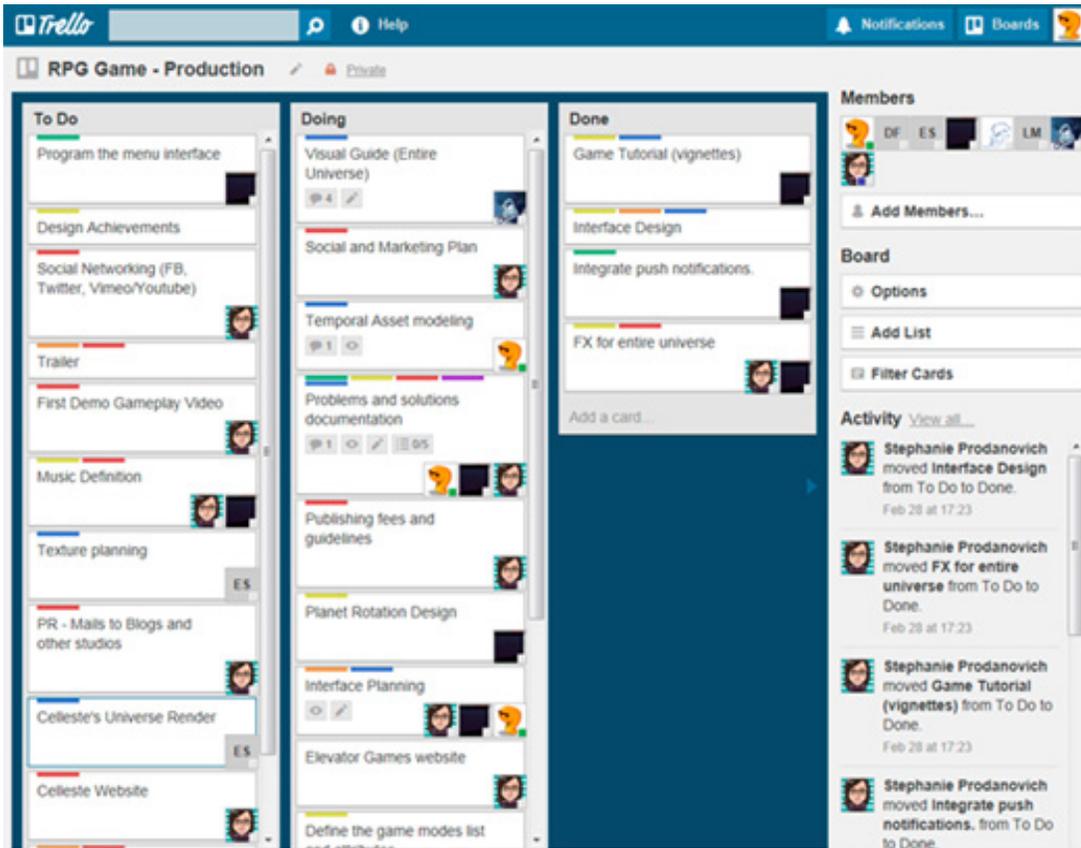
No obstante, se presentó un inconveniente: el trato con otros miembros del equipo era demasiado indirecto, de manera que se optó por enviar avances por correo electrónico o avisos sobre descargas desde el servidor.

1 *Kaxan Games*, de Guadalajara y *Slang Studio* del Distrito Federal, además de trabajos de **freelance** y ser director y co-fundador de *Wasd Studio*.

2 Ingeniero en computación del Instituto Politécnico Nacional, ex-programador en *Slang Studio*, en *Gameloft* de Guadalajara y actual programador en *Larva Studio*.

3 Aplicación en línea que permite armar y modificar documentos (textos, hojas de cálculo, presentaciones, etc.) entre varios miembros en tiempo real.

4 Ofrece el servicio de un sistema de organización en línea y de acceso libre, similar a una tabla de contenidos.



53. Screenshot de una tabla de organización de Trello.

En este punto, vale la pena señalar un hecho importante: aún no había un nombre para el incipiente estudio. Después de algunos meses de trabajo, el líder y **game designer** eligió *Elevator Games*, por hacer referencia a un elevador; a la expectación que siente una persona mientras se desplaza en vertical a través de un edificio desconocido y antes de que las puertas se abran en el piso deseado (imagen 54). De alguna forma, el objetivo del estudio se vería también reflejado en esa analogía al presentar juegos que generen expectativa a un público que entre en una categoría de “jugador casual”. Entre el área de diseño gráfico y producción se creó una imagen del estudio y también del juego por parte de algunos de los diseñadores subcontratados.

54. Logo de Elevator Games.



3.4 Justificación

Gerardo Basurto (2013) me envió las siguientes instrucciones redactadas por él mismo, donde especificó cuál era la meta del aspecto visual:

Arte y Motivaciones:

Algunos de los objetivos del arte es dar la intención de:

- La inmensidad del espacio.
- La naturaleza pasiva del espacio.
- El tiempo parece transcurrir más lento en el espacio.
- Un mundo mágico, imaginativo.
- Un mundo que captura e incita tu imaginación.
- El origen y la evolución de la vida.
- La inocencia e ingenuidad de la vida.
- Peligros naturales, causalidades del mundo y del espacio.

Algunas de las sensaciones que buscamos transmitir al jugador:

- Paz, tranquilidad y equilibrio espacial.
- Inocencia.
- Diversión.
- Peligro, acción.
- Tensión.
- Triunfo.
- Sensación de alcanzar algo inalcanzable.
- Un “pequeño paso para las vacas, pero un gran paso para la vida”.
- Imaginativo, mágico.

La dirección de arte para el planeta, props y personajes refuerza esta idea, al enfatizar curvas suaves y elementos compuestos por figuras sencillas creamos un universo que es “fácil” de leer, “fácil” de distinguir.

La premisa del juego “proteger y salvar a las vacas” también fue una decisión cuidada, las vacas representan la inocencia del universo, su forma, expresiones, animaciones y poses fueron diseñadas para enfatizar la idea de que son seres torpes e indefensos.

Los enemigos por otra parte no son completamente grotescos, puesto que no deseamos romper una ambientación general de inocencia en el universo. Los enemigos no son verdaderamente malvados, son cómplices de su propia naturaleza.

La forma y los componentes del planeta tomaron una fuerte inspiración de “planetas pequeños”, generados a partir de fotografía panorámica con una proyección estereográfica [imagen 44].

La intención detrás de estos planetas es generar la sensación de control. El jugador tiene a la palma de su mano el control sobre este planeta, podría sostenerlo en la palma de su mano y eso le otorga una fuerte sensación de poder y responsabilidad sobre lo que sucede en este planeta.

55. Imágenes de referencia de planetas pequeños.



Otra de las instrucciones de carácter específico fue que los assets se parecieran lo más posible a las ilustraciones que Riebeling había realizado para *Celleste*.

3.5 Creación de Concept Art

En el periodo de incertidumbre sobre el encargo oficial del arte para el juego, se me pidió hacer propuestas visuales con el fin de tener un punto de partida más claro, aun cuando después sólo me dedicaría a trabajos de modelado y texturizado. He aquí algunos ejemplos:

- La paleta de colores (imagen 56.5) fue el primer elemento y lo obtuve a partir del trabajo realizado por otro de los ilustradores subcontratados, Enrique Sandoval (imagen 56).



56. Ilustración hecha por Enrique Sandoval para *Celleste*.

Paleta de colores - Celleste



Paleta de colores general



Paleta de colores de espacio sideral

56.5 Paleta de colores de *Celleste*.

- Según instrucciones de Basurto, se dio prioridad a crear los **assets** principales, entre ellos, asteroides que el jugador lanzaría sobre un insecto y las lechugas de las cuales se alimentarían las vacas (imagen 57).
- Otro elemento de gran importancia eran los módulos que componen la superficie del planeta, con una forma base de hexágono, de acuerdo con las especificaciones del **game designer** sobre la retícula en que se iba a dividir el suelo. Los módulos que hacían falta eran los de pasto, piedra y cráteres hechos por los asteroides (imagen 58).



57. Assets: Asteroides y módulos de lechugas.



58. Módulos de pasto, piedra y cráter.

59. Imagen de referencia externa.

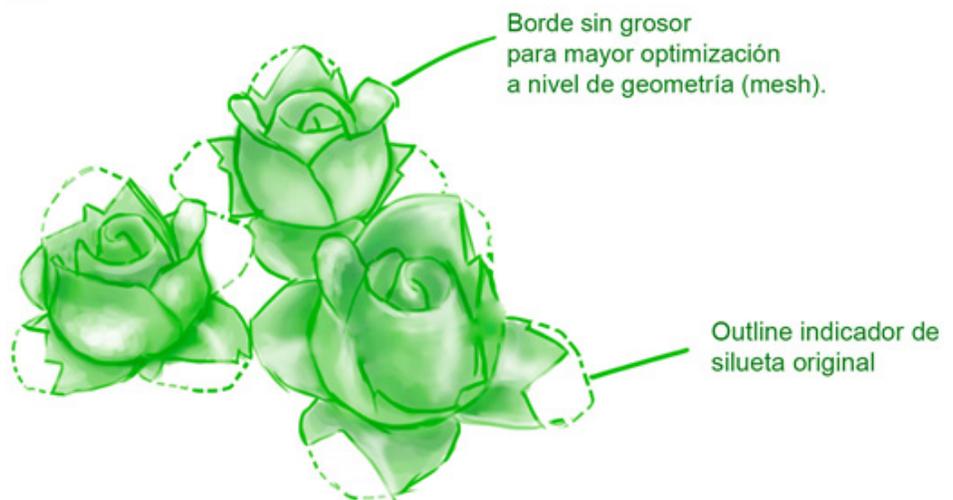


De estos módulos, el que más necesitó trabajo y numerosas versiones, fue el de la lechuga. Después de varios ensayos e interpretaciones, Prodanovich optó por modelar varias figuras en plastilina con la forma requerida por Basurto.

La forma básica es de una rosa de repostería, suave, curvada y con pocos pétalos muy gruesos (Imagen 59).

Este fue el resultado, según las anotaciones que hice sobre esa versión:

Lechuga. Versión 20.



60. Mordidas redondas sobre la lechuga.

La línea punteada indica las mordidas de las vacas. La forma dañada sería el “Estado 2” de la lechuga normal (que correspondería al “Estado 1”), pues era necesario comunicar al usuario que las vacas afectan su entorno y éste cambia de acuerdo con el grado de interacción; en otras palabras, no hay alimento infinito. Sin embargo, más tarde se indicó que las mordidas redondas (imagen 48) no iban a ser fáciles de crear de forma óptima en el modelo en 3D de la lechuga. Por eso que se prefirieron cortes más rectos (imagen 61).

Lechuga. Versión 20.



61. Mordidas con formas angulosas en las lechugas.

En una plática personal, Riebeling comentó que Basurto le había entregado el **arte** que realizamos varios integrantes con anterioridad, le indicó algunas características adicionales y la visión definitiva que tenía sobre el juego. Riebeling entonces procedió a bocetar. Después de unos primeros bocetos, comenzó a enviarle sus avances, los cuales fueron aprobados como estilo final. Incluida en los documentos intercambiados entre **game designer** y **concept artist**, había una presentación de diapositivas con el glosario de **assets** y sus instrucciones para trabajar en ellos desde varias áreas.

62. Asset de bicho rojo normal.



A continuación, presento una muestra únicamente de los objetos, en cuyo trabajo de modelado y texturizado (total o parcial) participé yo⁵:

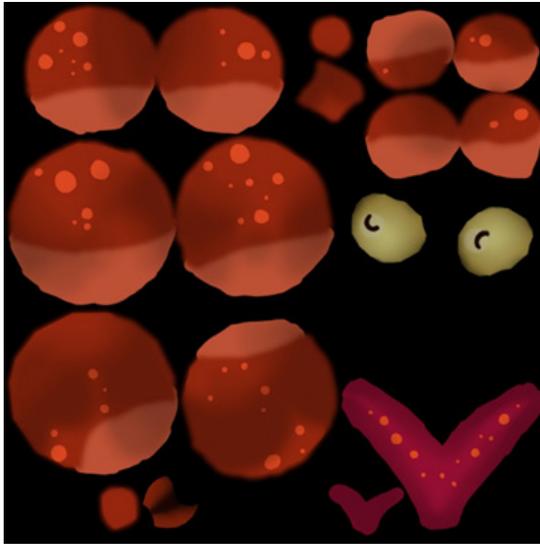
3.5.1 Personajes

El diseño inicial de personajes estuvo a cargo de Basurto y Cárdenas. En las siguientes imágenes, sólo se muestra a uno de los personajes, ya reinterpretado por Riebeling.

- **ASSET – TEXTURA DE BICHO ROJO NORMAL.** Sólo se requería la textura para este enemigo de las vacas. Ésta —con la presente gama de colores— indica que el jugador debe estar alerta; pero al mismo tiempo, no debe parecer demasiado agresivo. En la imagen izquierda, se sacó un **render** del modelo y se pintó sobre él, para hacerlo más preciso en el **concept**. La imagen derecha se usó como referencia para la textura de la parte inferior del bicho o “pancita”.

⁵ Por supuesto, no son los únicos que existen en el juego; además se repartieron las tareas de ilustración y modelado.

Se consultó con el **game designer** sobre si serían necesarias más texturas, además del **diffuse map**, como el **specular map** y/o **normal map**. No fue así. El resultado —a nivel de textura— fue este (imagen 63):



63. Diffuse map de bicho rojo.

Fabián Hernández había trabajado previamente con el modelo 3D (o **geometría**) de este personaje, para revisar que las animaciones funcionaran bien. En el proceso, sacó el mapa de **ambient occlusion**⁶ del bicho (imagen 64) y se trabajó sobre éste como base para aplicar color en el **software** de esculpido y texturizado de **Autodesk Mudbox**⁷. Se importó la textura del **ambient occlusion** en una primera capa en tamaño de 1024x1024 **píxeles**, y en modo de **blending** de **Multiply**; después se añadieron las capas de color. Al final, se combinaron en una sola y se exportó.

6 Literalmente: “oclusión ambiental”. Un mapa de oclusión ambiental crea una textura de sombreado suave, como si el modelo fuera iluminado sin una fuente de luz directa, como en un día nublado (Definición tomada el 3 de noviembre de 2013 del sitio: <http://wiki.polycount.com/AmbientOcclusionMap>)

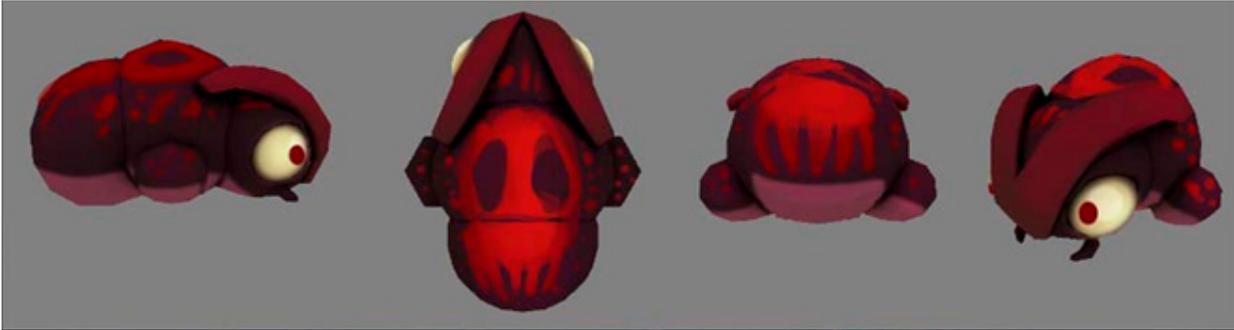
7 Funciona con la misma estructura de capas que **Photoshop**, tanto para texturas pintadas como para capas de esculpido.

64. *Mapa de iluminación pre-cocinada (o pre calculada) de ambient occlusion del bicho.*



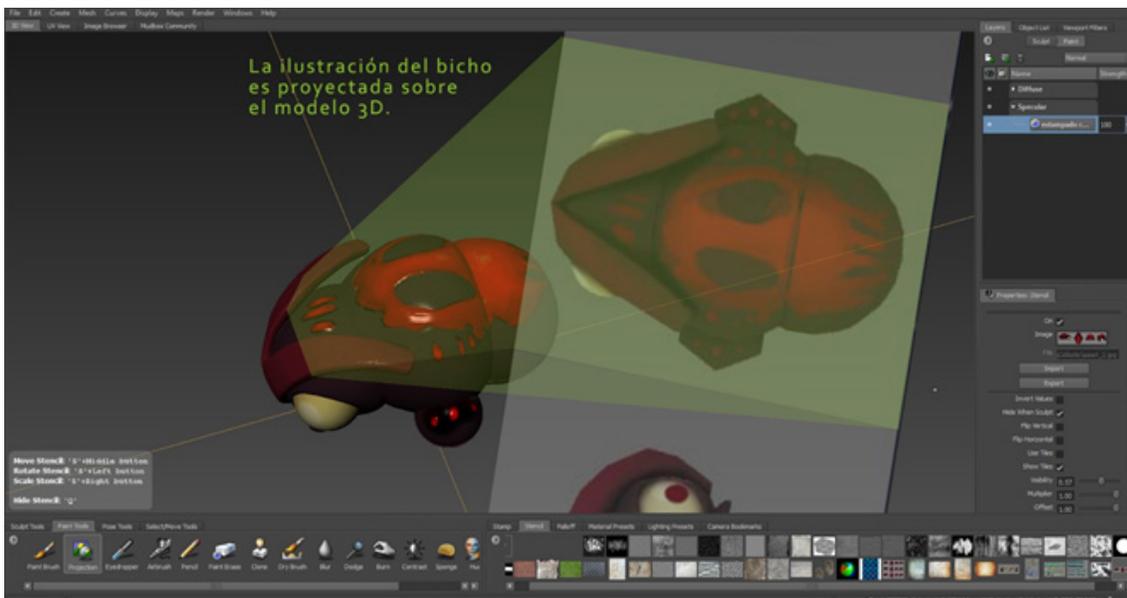
64.5 *Resultado final.*





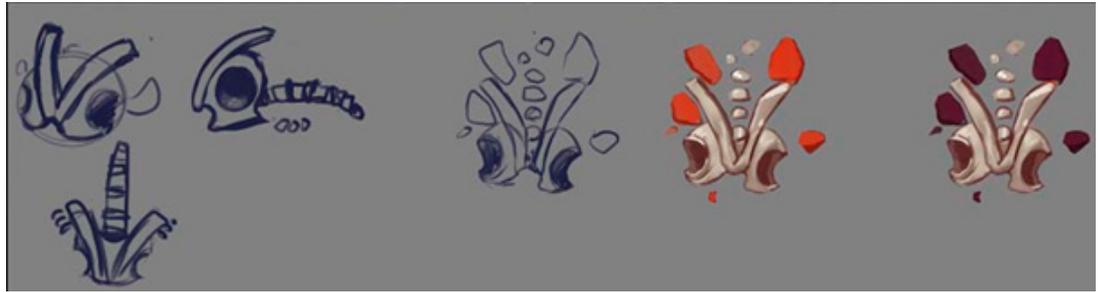
66. Bicho rojo explosivo.

- **ASSET – TEXTURA DE BICHO EXPLOSIVO.** Los colores son más intensos y un diseño ligeramente más amenazador. Para garantizar que el diseño básico



66.5 Proyección de textura sobre modelo.

67. Residuos mortuorios de los bichos (normal y explosivo).



68. Residuos mortuorios de la vaca.



fuera el mismo, se proyectó la imagen del **concept** sobre el modelo en **Autodesk Mudbox** (imagen 66 y 66.5).

- **ASSET – MODELADO Y TEXTURA DE RESIDUOS DE BICHO Y VACA.** Se requería modelado desde cero y con texturas con **normal maps**, **diffuse** y **specular maps**. Estos objetos aparecen como los restos de los personajes principales, bicho y vaca, cuando recibieran daño mortal causado por un meteorito (imágenes 67 y 68).



69. Concept art *del planeta sin habitantes* por Juan Pablo Riebeling.

3.5.2 Escenarios

Basurto y Riebeling también fueron los responsables de los escenarios, tomando en cuenta las reglas del juego y su programación. Para tener un amplio rango de movilidad, el sistema de reticulación del planeta es de base hexagonal; si se elegía una de base cuadrada, los personajes no hubieran podido moverse de forma óptima. (imagen 69).

3.5.3 Props

Riebeling —con base en la justificación del proyecto y en la constante comunicación que mantenía con Basurto y Prodanovich— estuvo a cargo del diseño de **props** o utilería, lo cual se ve reflejado en el **concept art**.

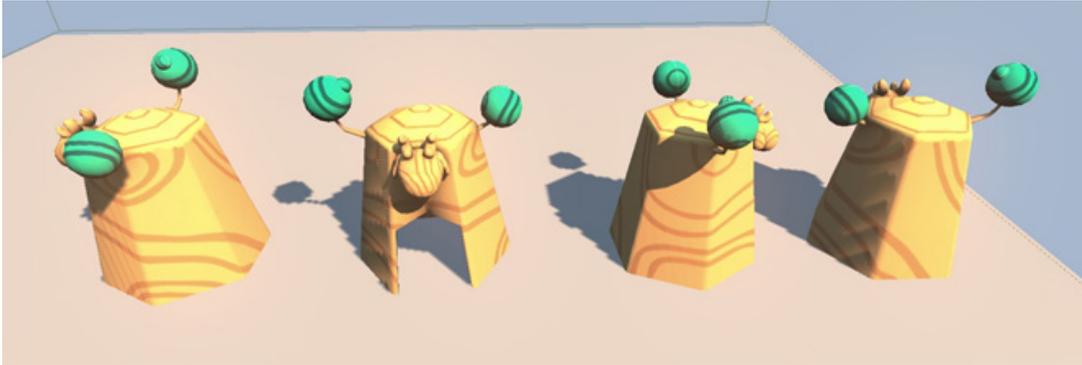


70. Portal de la vaca.



70.5 Diffuse map con ambient occlusion añadido.

- **ASSET – TEXTURA DE CUEVA/BASE.** Punto meta al que deben llegar las vacas (imagen 56). El tercer dibujo a color de la derecha indica a grandes rasgos cómo se requería el sistema de partículas con el que se iba a contar dentro de la cueva-base.



70.7 Modelo con textura funcionando en UDK.

Fue modelado por Hernández y a mí se me pidió hacer la textura de **diffuse map**, que igualmente se trabajó a partir de un mapa de **ambient occlusion** con iluminación precocinada.

➤ **ASSET – MODELADO Y TEXTURAS DE ROCAS.**

Objeto ambiental que le da forma a la atmósfera general, pues orbita alrededor del planeta, como restos de asteroides que colisionaron en algún momento contra él (imágenes 71 y 72).



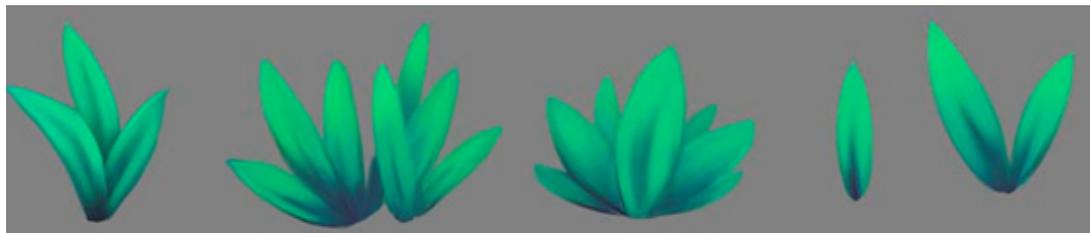
71. Roca que flotaría en la atmósfera o se usaría en la superficie del planeta.

72. Piedra esférica adherida a la superficie del planeta.



- **ASSETS – MODELADO Y TEXTURAS DE VARIADOS CÚMULOS DE PASTO.** Ubicado en todo el planeta, era importante crear cada grupo como un solo objeto (imagen 73), por separado, para hacer variaciones de composición en distintas áreas. Como en el caso del bicho explosivo, se utilizó el mismo **concept** para hacer la textura de **diffuse** al proyectar la ilustración del césped.

73. Cinco assets de pasto.



- **ASSET – MODELADO Y TEXTURAS DE RAMAS.** Surgen del suelo, como complemento del ecosistema (imagen 74). En total, se indicó que se modelaran cuatro objetos, con dos juegos de texturas distintas (con **diffuse**, **specular** y **normal map**).



74. Cuatro assets de ramas adheridas al suelo.

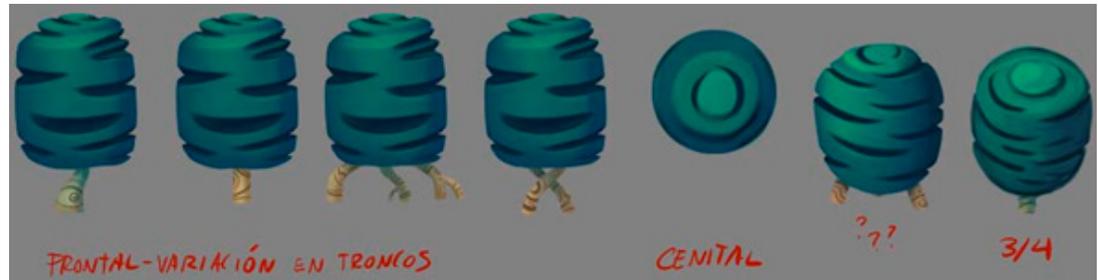
- **ASSET – MODELADO Y TEXTURAS DE ÁRBOLES.** Estos cuatro árboles (imagen 75) comparten una misma pieza en la copa, pero tienen distintos troncos a nivel de geometría y textura.



75. Variación de árboles con patrones en las copas.

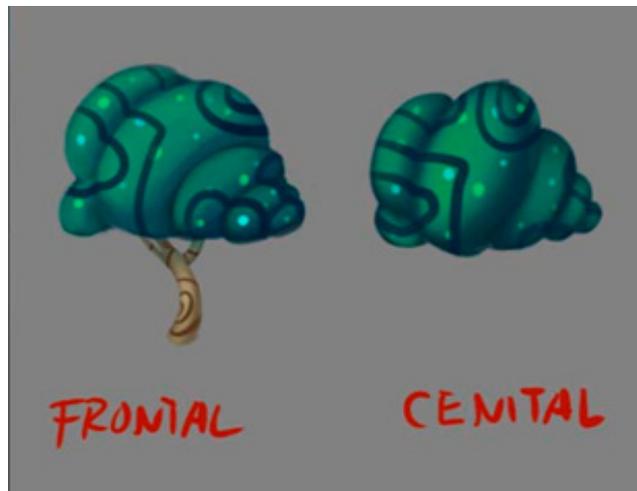
- **ASSET – MODELADO Y TEXTURAS DE ÁRBOLES.** La diferencia con los árboles de la diapositiva anterior es la ausencia de los patrones de diseño de la copa de los árboles.

76. Variación de árboles sin patrones.

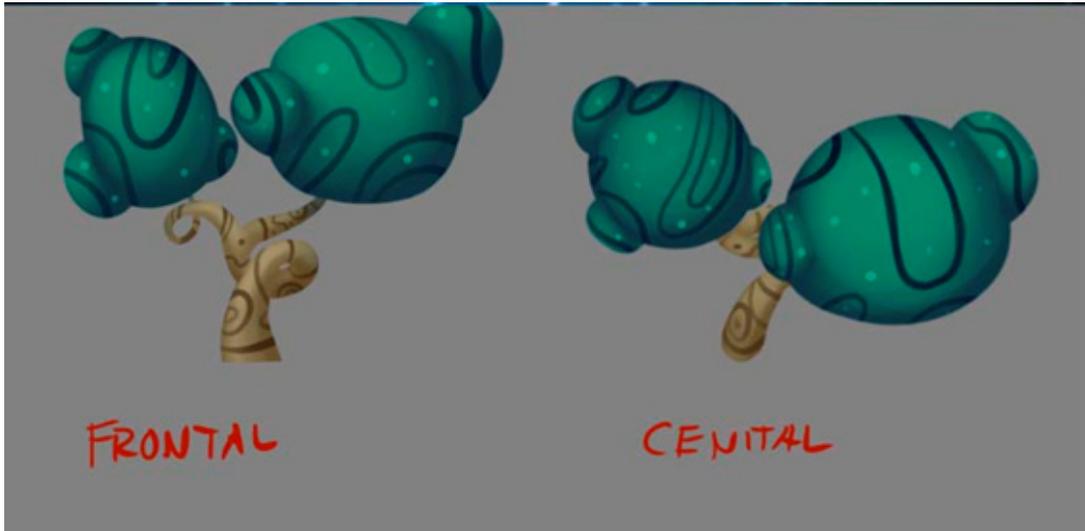


- **ASSET – MODELADO Y TEXTURAS DE ÁRBOLES ALTERNATIVOS 1.** Este árbol tiene un tronco y copa diferentes de los árboles anteriores. Era importante agregar un elemento visual orgánico en ellos, por eso son tan redondos y con patrones garigoleados

77. Árbol con vista frontal y superior.



- **ASSET – MODELADO Y TEXTURAS DE ÁRBOLES ALTERNATIVOS 2.** Mismo caso que el anterior.



78. Árbol con vista frontal y superior.

3.6 Características técnicas de gráficos

Al mismo tiempo que se desarrollaba el **concept art**, era necesario estudiar el motor de juego. La experiencia de Fabián Hernández y las investigaciones sobre el tema le permitieron al **game designer** optar por el Unreal Engine 3 (descargable en línea y sin costo alguno), el cual funciona para iOS de iPad.

Muy al principio, se me mandó investigar los aspectos técnicos del motor relacionados con la parte visual. Mis anotaciones fueron las siguientes:

Asset management

3D pipeline:

1.- Concept art (Quetzal Cárdenas como encargado)

Recomendaciones para bocetos

Bocetos a grandes rasgos (yo recomendaría que los bocetos a color, fuesen en técnica digital, no tradicional). Una vez aprobados, se procede a hacer los bocetos más detallados.

Tratamientos diseñados detalladamente para indicar escenarios, personajes, piezas de utilería, iluminación y paletas de colores.

Igualmente, una vez aprobados, se continúa con el modelado.

Respecto a la iluminación, como dato; en Diablo III (y en más juegos, claro está), había visto que la progresión de color es importante; así que si el juego va a contar con diferentes niveles, sería más enriquecedor para la experiencia del jugador el cambiar un poco la paleta de colores conforme se avance en el juego (además de que éste sea cada vez más rápido), para añadir emoción.

Prototipos- Maquetas

Una vez listos los personajes, proceder a bocetos de animación/flipbooks.

Maquetas de modelos-vistas de frente y perfil, de ser posible; si no, sólo vista de tres cuartos.

Maquetas de textura (indicarlas como si fuesen paletas de colores, indicando a qué asset pertenece cada una).

Diseño de arquitectura

Diseño de UI- bocetos a grandes rasgos; cuando sean aprobados, proceder a los definitivos y sus mockups¹.

2.- Creación

Herramientas comerciales:

Photoshop, Gimp.

Modelado con herramientas 3D: 3D studio max, Maya. El modelador deberá entregar la pieza modelada con las UVs listas para el texturizador-respecto a este punto.

Deberán entregarse dos modelos por cada pieza; una con bajos polígonos, y la otra con detalle.

3.- Asset management

Herramientas de superficie/shaders (contiene los mapas de normals, diffuse y specular)

Sistemas de video/renderero

4.- Conversión

Poniendo los recursos en su forma final.

Tipo de archivo de conversión

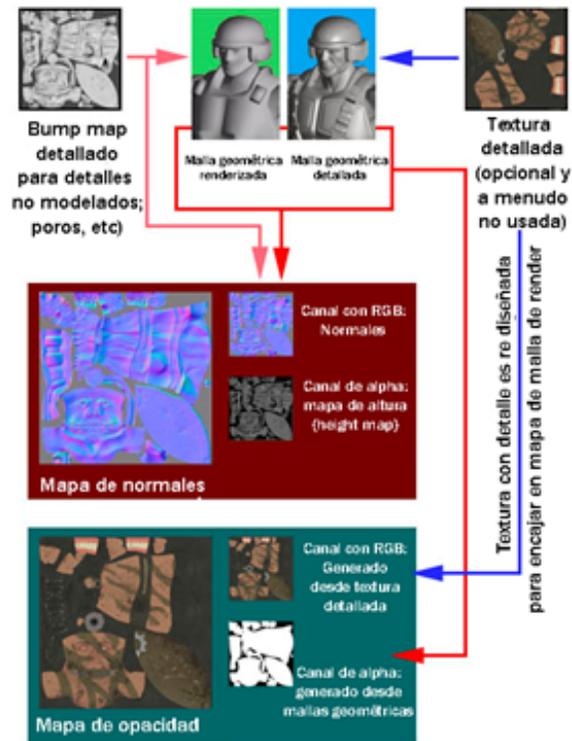
PSD a TGA/JPG, por ejemplo

Compresión

Colección (archivos zip, pak, etc)

Escalamiento según resolución de pantalla

¹ *Maquetas.*



El mapa de normales es creado desde las mallas y el detalle del bump map. El Mapa de opacidad es creado desde las mallas y la textura con detalle.

Componentes de texturas de un material en UDK.

Exportar desde formatos nativos de modelado (dependiendo de las necesidades de Unreal)

Para el punto descrito arriba, esta documentación nos será de ayuda; lo que no recuerdo es qué dispositivos móviles son los que usaremos

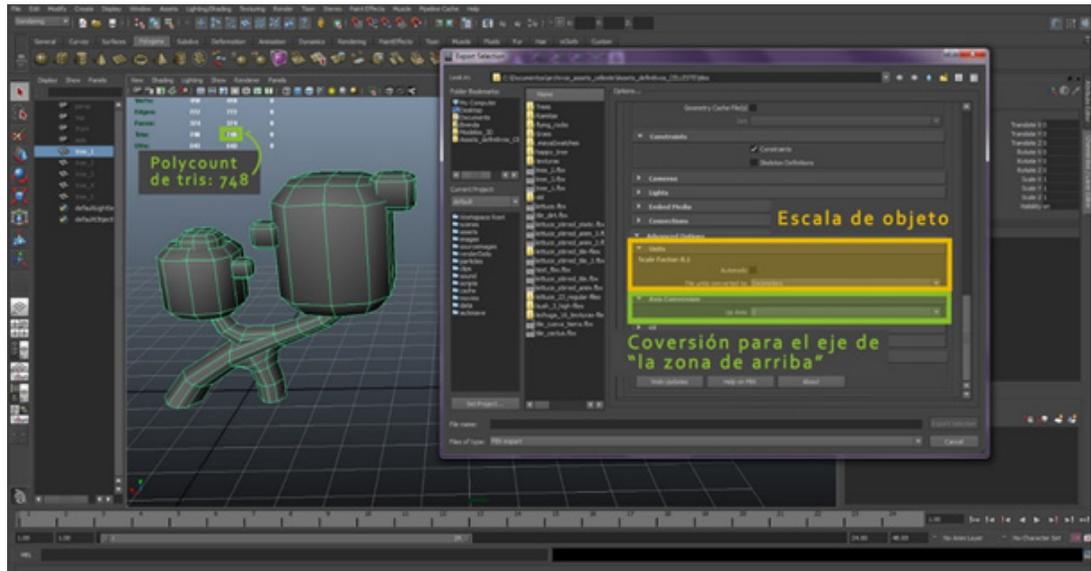
<http://udn.epicgames.com/Three/GettingStartediOSDevelopment.html>

Probar en el demo del juego

Debugging³

- 2 Imagen traducida por mí, tomada de <http://udn.epicgames.com/Three/WebHome.html>
- 3 Proceso metódico de encontrar y reducir el número de bugs o defectos en un programa de computadora o en una pieza de hardware electrónico, haciéndolo entonces funcionar como se espera.

79. Polycount, escala y up axis en Maya de uno de los assets de Celleste.



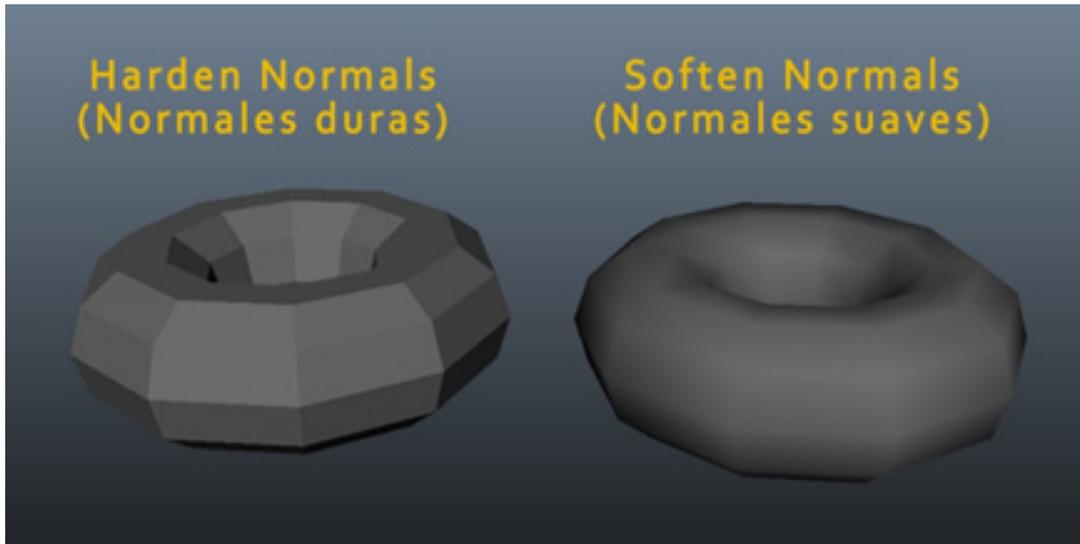
Poco después, se revisó la información y se hicieron algunas pruebas con esqueletos, lo cual se verá más adelante, en el apartado 3.7.

3.6.1 Texturas o mapas

El formato oficial de imagen seleccionado fue el **.tga**, por su nivel de compresión y calidad; además de ser un formato común compatible con UDK. Sólo era necesario usar el **canal alpha** de los mapas de **diffuse** para almacenar los **specular** y ahorrar **draw calls**, es decir, las órdenes del programa de juego para “llamar a escena” un recurso gráfico (tarea que consume memoria).

En el caso de los **normal maps**, el **green channel**⁸ tiene que estar invertido, pues UDK está configurado para interpretar la información y mostrar los relieves en la dirección correcta. Los tamaños de textura variaron de

⁸ Canal de color luz, en este caso, verde, de los tres que existen; R (Red), G (Green) B (Blue).



79.5. Comparación entre normales duras y suaves.

entre 2048x2048px y 512x512px. La importancia del objeto determinó la calidad de la textura, lo que facilitó decidir qué resolución era suficiente para la **geometría** en cuestión.

Como se describió en algunos assets del subcapítulo de concept art, se trabajó en los **diffuse maps** con texturas con iluminación **cocinada** de **ambient occlusion** como base, y en modo **multiply**, uno de los filtros conocidos como *blendings*⁹ para evitar que tuviesen un acabado plano. Para obtener los **occlusion y normal maps**, se utilizaron los *softwares* de Maya y Xnormal.

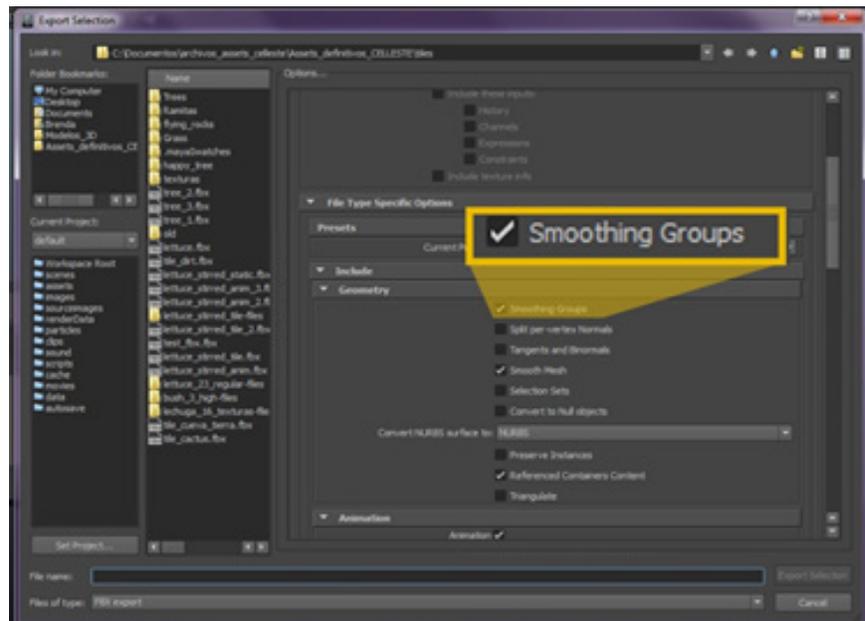
Otra limitante fue el uso de **canal alpha** a modo de **transparencia**, pues éste no es muy estable en el **Unreal engine** para dispositivo móvil. Es por ello que la **geometría** debía ofrecer el detalle necesario para evitar el uso de transparencias en textura, pero manteniendo cierto nivel de simplicidad.

3.6.2 Restricciones para modelos

Para exportar modelos de Maya a UDK fue utilizado el

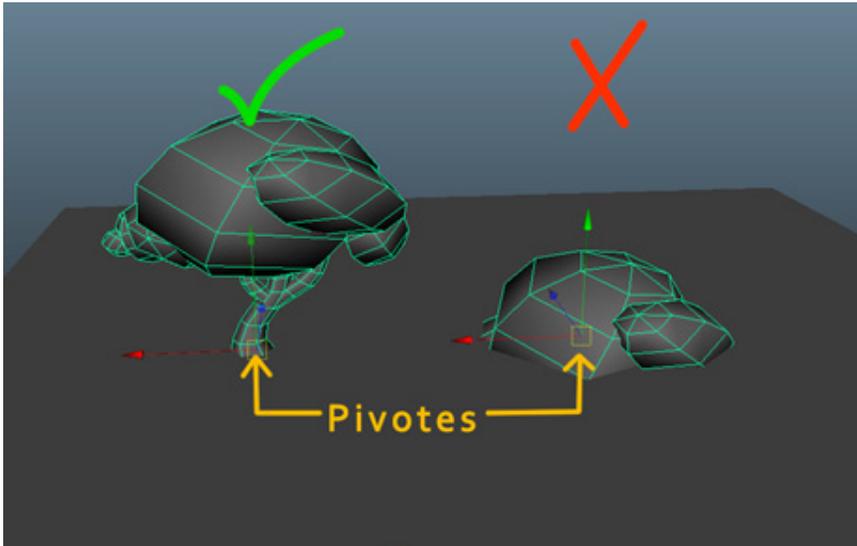
⁹ Existen en Photoshop para intercalar y componer una imagen en distintas capas de contenido.

80. Cuadro de diálogo de exportación en Maya.



81. Asset de roca semi-esférica de Celleste en UDK en versiones con y sin smoothing groups.





82. Un asset de un árbol en dos versiones: con el punto de pivote ubicado correctamente en su punto de origen (izquierda), y en el otro, ubicado en un punto erróneo (derecha).

formato **.fbx** (indicado en la documentación sobre UDK), para geometrías estáticas y geometrías con huesos animados. En tanto, el **poly-count** de los **assets** era en promedio de unos **800 tris**¹⁰; muy pocos podían ser más complejos que eso.

Por otro lado, estaba el eje que indica la “zona de arriba”: Los modelados se hicieron en Maya, pero en UDK, el **up axis** es distinto¹¹. Para resolver este detalle al exportar la geometría en **.fbx**, sólo debe elegirse el eje **Z** como **up axis** en el cuadro de diálogo. Asimismo, se presentó la diferencia entre escalas; la de **3ds Max** es más compatible con UDK que la de Maya (la cual es menor en comparación con las otras herramientas de *software*). Esto se resuelve igualmente en el cuadro de diálogo de exportación, en el apartado de **Units** como se indica en la imagen 79.

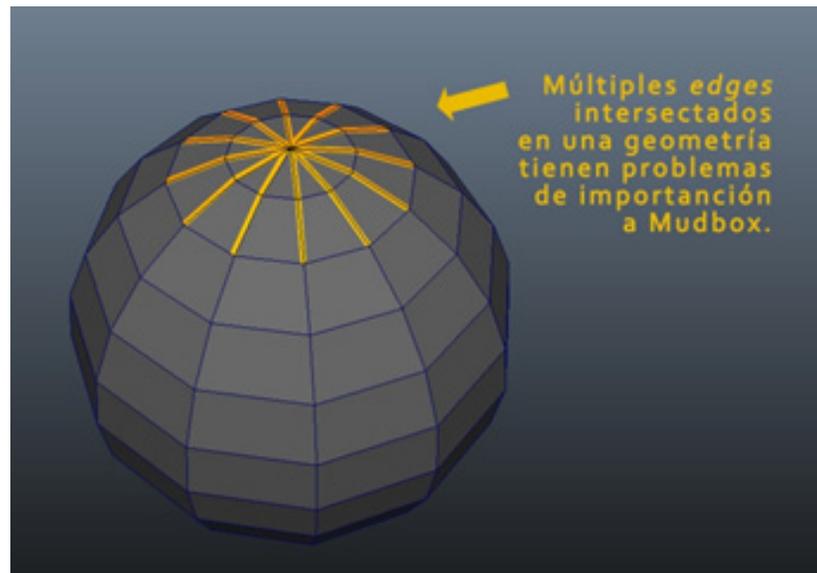
Ahora, para sacar el máximo provecho de las texturas —en especial de los **normal maps**— es necesario suavizar normales de la geometría; pero cuando se aplica el proceso a

¹⁰ Triangles (triángulos). Polígonos o caras que componen un modelo tridimensional. Como su nombre indica, es un plano de tres vértices. El conteo de éstos es importante para tener siempre un modelo optimizado.

¹¹ El up axis o eje que corresponde a la zona de arriba en Maya es el Y, mientras que el de UDK es el Z.

todas ellas, el *shading* o sombreado tenga un aspecto extraño, con sombras donde normalmente no correspondería. Si no se hace este proceso, la geometría lucirá con bordes duros y finos. Pero habiendo suavizado las normales, y borrado la historia de transformaciones del objeto, al exportar en .fbx, debe habilitarse la casilla de *Smoothing groups* para que *UDK* reconozca estas características en la geometría importada.

Por último, pero no menos importante, el **punto de pivote** debe estar colocado correctamente en la geometría. Si un árbol tiene su punto de pivote en la copa y no en la base del tronco, al cargarlo en *UDK*, lo único que se verá del árbol es la copa, no el tronco (suponiendo que el objeto esté en un plano que esconda el tronco y no “flotando”).



83. Ejemplo de edges o bordes para iniciar el modelado.

3.7 Modelados

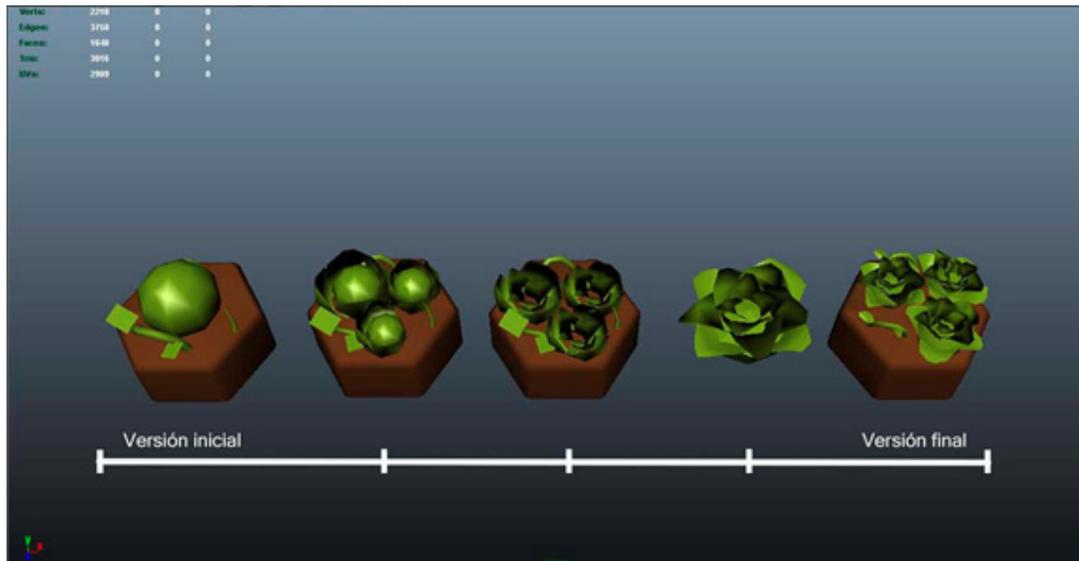
Tomemos el *asset* de la lechuga como ejemplo de evolución de modelado para un elemento visual.

Una de las primeras tareas al analizar una **prop** es ubicar la forma base del objeto. Al principio, hice una serie de ensayos con *props* esféricos o cilíndricos y usé esos **primitivos** en el **software** de modelado. Tiempo después, se hizo evidente que no era una decisión que beneficiara al proyecto, ya que había

problemas de compatibilidad entre estas figuras y el programa de esculpido **Mudbox**; pues éste no acepta geometrías con más de dos bordes intersectados entre sí. Al subdividir la figura, hay errores en la geometría y muchas deformaciones, entregando un resultado poco satisfactorio en las texturas obtenidas.

Para solucionar este problema, se decidió usar mayoritariamente un cubo como base, sin importar la forma del **asset**. Esto evitó que Mudbox continuara teniendo problemas para subdividir y, así, las texturas no mostraron deformaciones inesperadas (o se mantenían al mínimo).

Se conservó la última versión de lechuga (imagen 84), pero después fue modificada, según nuevas necesidades del proyecto.



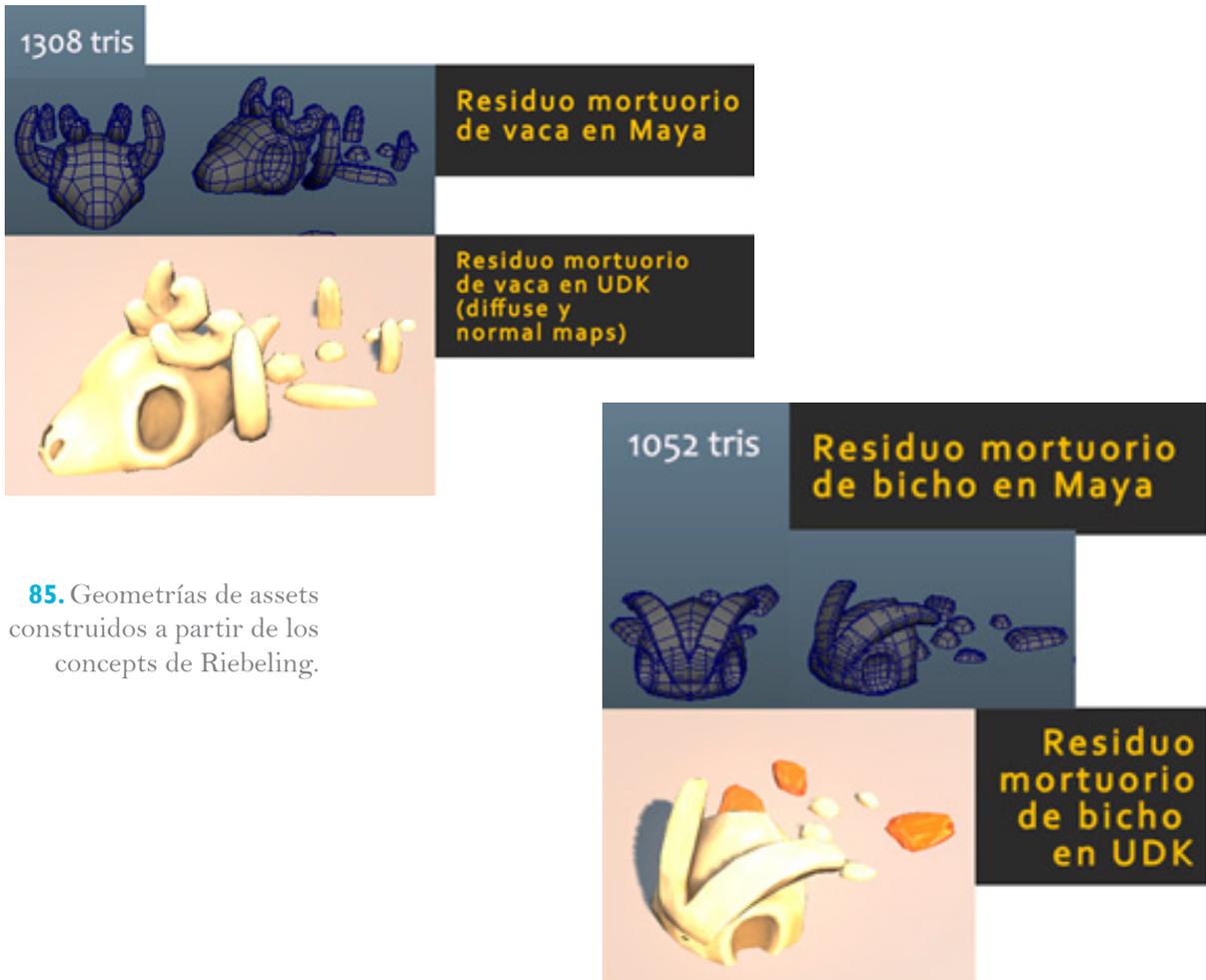
84. La progresión de forma de la lechuga puede verse en estas geometrías.

Ahora bien, en UDK invariablemente se hará una triangulación de caras y por eso no debe haber una cara de más de cuatro vértices, ya que, si tiene más, UDK hará la triangulación automática para obtener el resultado más óptimo al interior de sistema, lo cual no siempre da buenos resultados a nivel visual.

Por otro lado, se trató de evitar, dentro de lo posible, que muchos bordes convergieran en un mismo vértice. Además, había que tomar en cuenta que los esqueletos o residuos mortuorios de los habitantes del planeta eran los que tienen

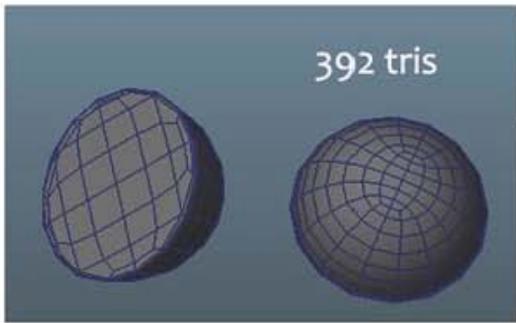
un **poly-count** más alto. Sin embargo, los números aún resultaban tolerables, porque no se espera que haya muchas copias de esas **geometrías** en el juego¹².

Para efectos prácticos y de claridad, presento imágenes con algunos de los **assets** y su **poly-count**:



85. Geometrías de assets construidos a partir de los concepts de Riebeling.

12 Si el juego hubiese sido diseñado para celular, estos objetos (y por los alcances hasta el momento en tecnología) deberían bajar al menos a la mitad su *poly-count*.



392 tris

Rocas semi-esféricas en Maya, sin texturas

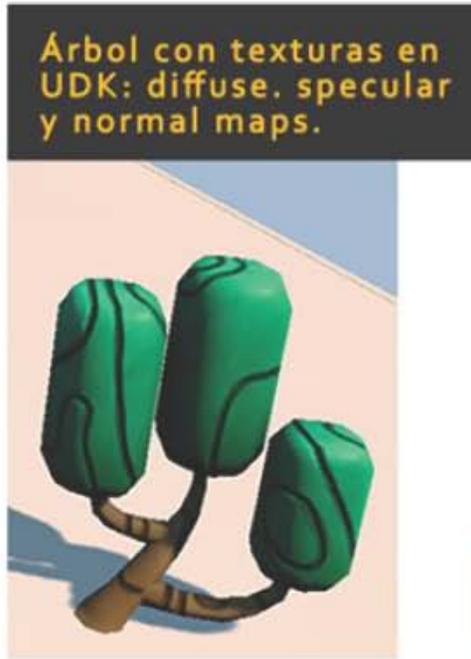


Mismas rocas en UDK, con todas sus texturas



584 tris

Árbol en Maya.



Árbol con texturas en UDK: diffuse, specular y normal maps.



120 tris

Roca larga en Maya



Roca en UDK



Ramita en Maya

128 tris



Ramita en UDK con diffuse y normal maps

86. UV maps de algunos assets, indicando las islas correspondientes a un solo asset, en cada mapa.



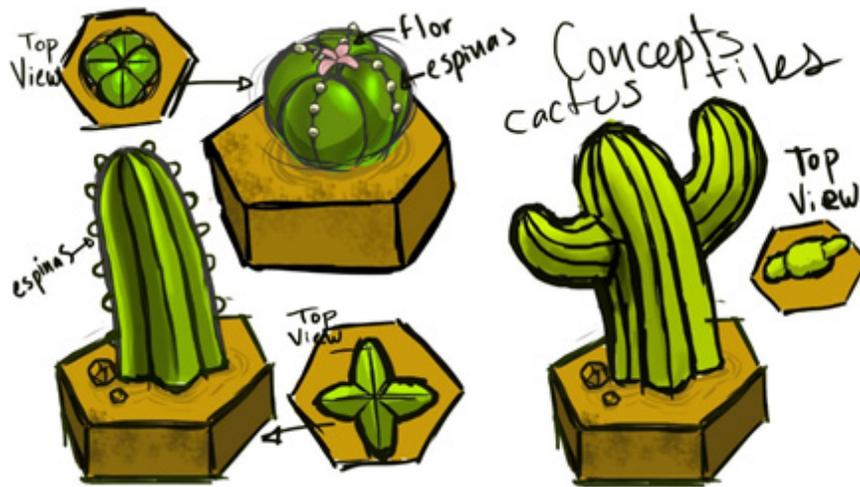
Y aquí, un segundo diagrama (imagen 86), pero mostrando los **UV maps** de algunos **assets**. Como se podrá apreciar, muchas geometrías comparten el mismo *UV map* para ahorrar memoria:

3.8 Texturizado

El siguiente cactus es un ejemplo de pieza de utilería para vegetación realizada como parte de las propuestas de arte, cuando no se contaba con un **art concept** definido.

Se me pidió crearlo desde cero y comencé por el concept, basada en las imágenes de diferentes cactáceas, tratando de ceñirme a la idea del **game designer**, de favorecer las formas redondas y casi inofensivas (imagen 87), por ser parte de la dieta del personaje principal del juego:

De entre estos modelos, Basurto eligió el de la flor rosa en la parte superior. Entonces, procedí con el modelado y el esculpido, para luego hacer el **cocinado** de iluminación en **occlusion map** (imagen 88):



87. Concept inicial de assets de cactus.

El resultado fue aprobado y continué con un ensayo de textura de **diffuse** al pintar sobre este **screenshot** en **Photoshop**:

Después de la aprobación de este ensayo, continué con el texturizado del **diffuse map** y el resultado fue el siguiente:



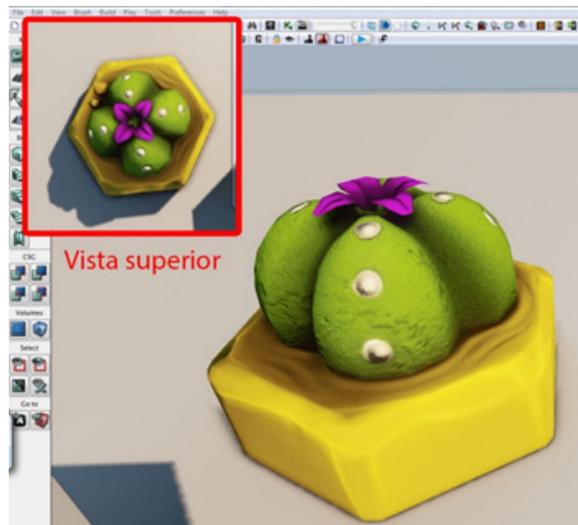
88. Modelo de cactus con mapa de normales y oclusionion como diffuse.

En la versión final del juego, no se incluyó este *asset*.
Ahora, veamos una breve muestra del desglose de texturas de un par de *assets*:



89. Modelo de cactus con ensayo de colores sobre screenshot.

➤ Abajo, se aprecian los dos mapas requeridos



89.5. Modelo con texturas finales, con vista superior en un cuadro.

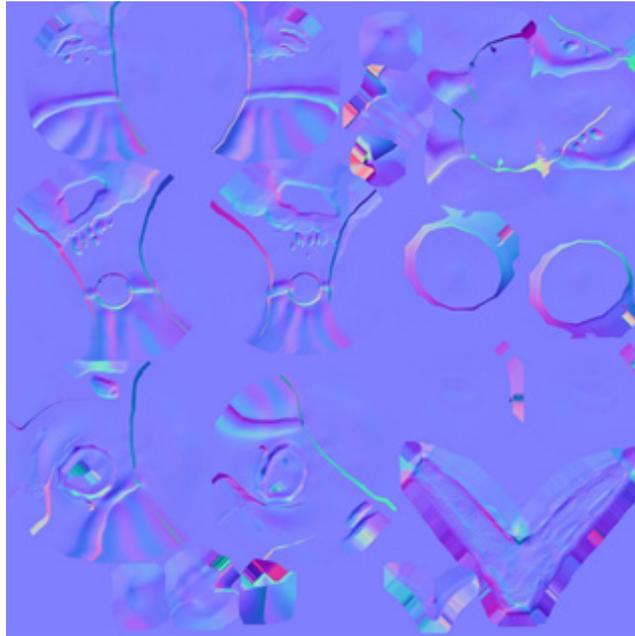
para el *asset* del bicho explosivo, el diffuse map a la izquierda, y el **normal map** a la derecha (imágenes 90 y 91).

- Una muestra de las texturas para los árboles con la misma copa:

Y ahora, procedamos con una muestra de un material construido en UDK:



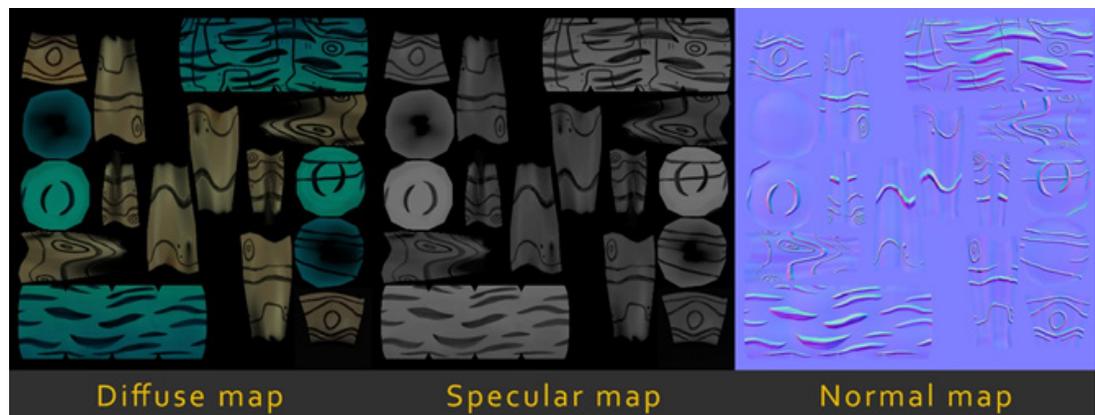
90. Texturas de bicho explosivo.

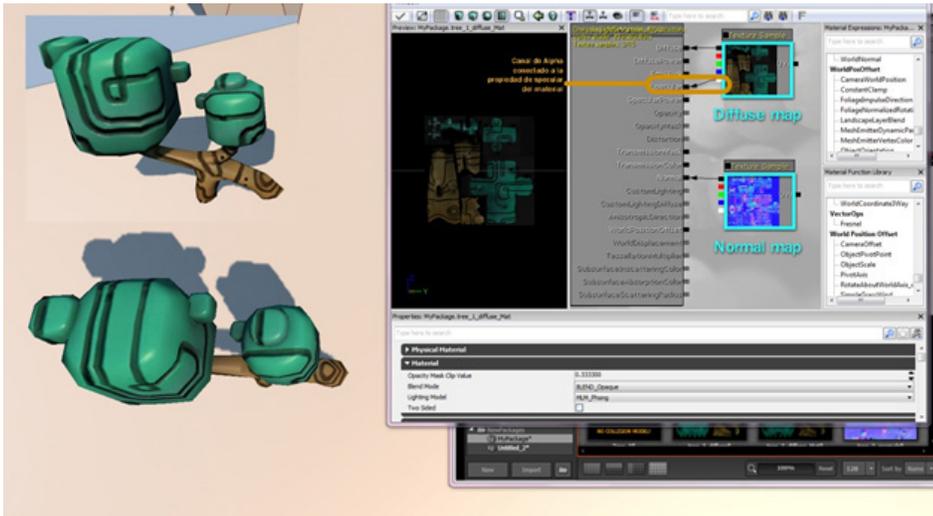


91. Texturas de bicho explosivo.

Una de las principales características de UDK son sus sistemas de nodos para crear materiales, organizar ciclos de animación, crear sistemas de partículas, etc. Al entender esta dinámica, será fácil crear estrategias para construir materiales más completos sin consumir mucha memoria.

91. Cada mapa es de 1024 x 1024 px. El canal alpha del diffuse map contiene la información del specular map por tener colores en escala de grises.





92. Sistema de nodos de UDK para la construcción de un material.

3.9 Animaciones

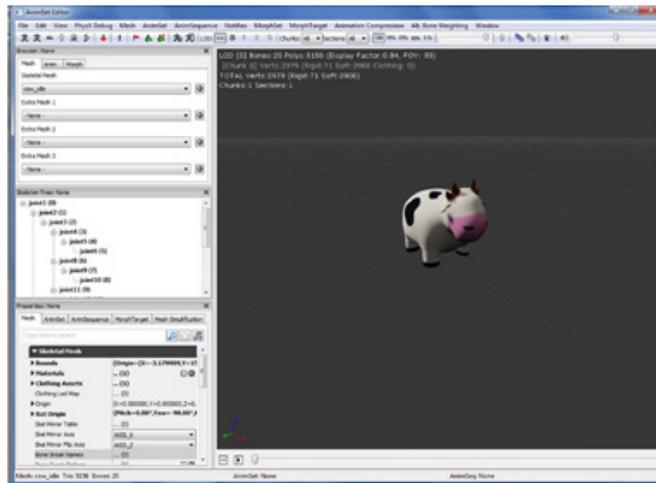
Este trabajo estuvo a cargo de Daniel Farrera como animador y Miguel Ángel Medina como **rigger** o constructor de esqueletos. Cuando comenzaron los experimentos para probar el funcionamiento del *UDK*, hice estas anotaciones para el Departamento de Producción y Diseño sobre animación y esqueletos:

Se hicieron pruebas con la finalidad de asegurar un resultado óptimo en el **build**¹³ del juego, ya que, en algunas ocasiones, *UDK* no aceptaba los esqueletos, y una de las razones fue básicamente por el orden incorrecto entre jerarquías de huesos.

¹³ Versión jugable del videojuego.

Problemas y soluciones para la introducción de recursos gráficos en UDK.

La escala utilizada en 3ds Max y UDK son similares, mientras que la de maya es más pequeña (tomando la retícula base como parámetro). Inicialmente, al cargar la malla, era apenas visible en el punto de origen. Así que una manera de solucionarlo es primero cargar la malla estática de un tamaño considerable y revisarlo antes estando dentro de UDK, y si tiene el tamaño deseado, proceder con el rigging y la animación.



Rig cargado en UDK.

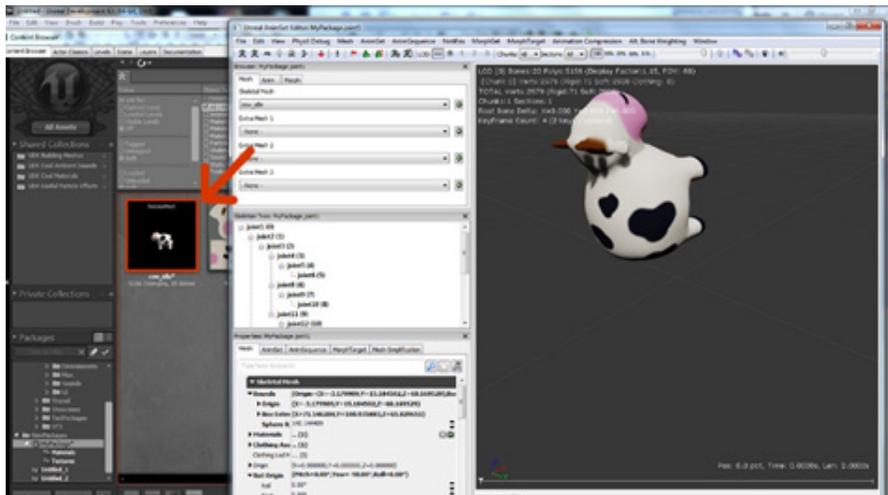
El rigging hecho posteriormente tuvo una serie de errores al ser cargado en UDK. A continuación se describirán los hechos de la forma en que fueron saliendo:

- 1.- Modelado del personaje en Maya.
- 2.- Rigging del personaje.
- 3.- Emparentado del hueso raíz con una curva como controlador.
- 4.- Exportado de la escena en .fbx, sin smoothing groups, ni orientación adecuada para UDK.
- 5.- Cargado de éste en UDK como malla estática

(resultado: tamaño muy reducido).

- 6.- “Des-pegado” de la piel del esqueleto creado inicialmente.
- 7.- Re-escalado del esqueleto usando el controlador de la curva creada previamente.
- 8.- Re-pintado de pesos en la piel.
- 9.- Exportado nuevamente en .fbx
- 10.- Importado en UDK.

Resultado: Modelo de un tamaño normal y visible en UDK, mal orientado, con la colita de la vaca pegada al piso.



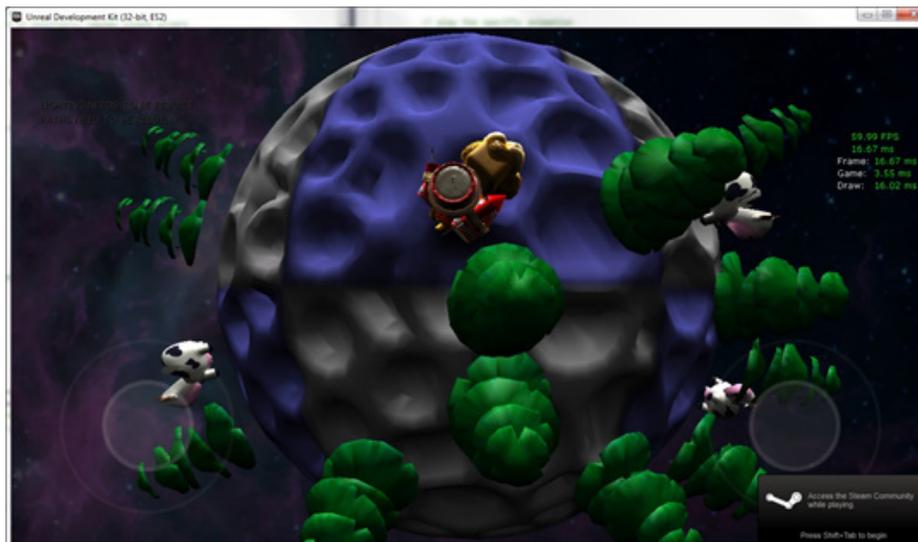
Rig cargado en UDK con errores de orientación.

Al consultar con Jacobo Ríos, antes programador en Larva y Chris Camacho, rigger y technical artist, mencionaron la importancia de no tener curvas emparentadas con algún hueso, y que todos éstos estén en orden, cada uno con un nombre distinto, además de un hueso raíz que esté preferentemente en el origen de la retícula y con los ejes Y y Z intercambiados.

Después de esta revisión, se hicieron los cambios pertinentes: de draw calls en total en el juego) con resultados favorables.

NOTA: UDK requiere idealmente añadir Smoothing groups a las mallas importadas, sin embargo, al probarlo en iOS, los smoothing groups provocaban una deformación en los modelos, así que se decidió siempre exportar cada asset en dos versiones; con y sin smoothing groups.

- 1.- “Des-pegado” de la piel (con la escala correcta) del esqueleto corrupto.
- 2.- Borrado del esqueleto original.
- 3.- Creación de un nuevo esqueleto con nueva escala, y un hueso raíz ubicado en el origen de la retícula, sin curvas como controladores.
- 4.- Pintado de pesos en la piel.
- 5.- Animado de ciclo “Idle” .
- 6.- Exportado con orientación nueva (eje Y como Z) y horneado de animaciones de inicio a final en la línea de tiempo.
- 7.- Importado en UDK, con su material correspondiente (un solo material por asset o assets instanciados, para tener el menor número



Una de las primeras pruebas en simulador para dispositivo móvil de UDK. Nótese las deformaciones en las vacas.

3.10 Iluminación y Efectos especiales

Estos dos aspectos no tomaron tanto tiempo de desarrollo como los anteriores, sin embargo, eso no implica que carezcan de importancia. La iluminación sería responsable de sacar partido de las texturas en los modelos y los efectos utilizados (sistemas de partículas) serían parte crucial para representar acciones como caídas de asteroides, interacción con lechugas, “polvo estelar” en agujeros negros, etc.

3.10.1 Sistemas de partículas

William Reeves, investigador de Lucasfilm Ltd. explica con mucha claridad el concepto: “A particle system is a collection of many many minute particles that together represent a fuzzy object. Over a period of time, particles are generated into a system, move and change from within the system, and die from the system.”¹⁴ (citado por Shiffman, 2012).

Fabián Hernández, debido a su experiencia, fue el encargado de ésta área, aunque al principio yo había intentado hacer pruebas con los sistemas de partículas de UDK. Analicé el juego llamado *Chicken Coup* (un juego para iPad) como modelo a seguir, por recomendación de Basurto, pues mostraba una gran capacidad de optimización por la cantidad de objetos que, en vez de ser construidos como **geometrías**, se concibieron como sistemas de partículas, las cuales utilizaban solamente texturas.

El juego consiste en salvar a las gallinas de una granja de los zorros que las acechan.

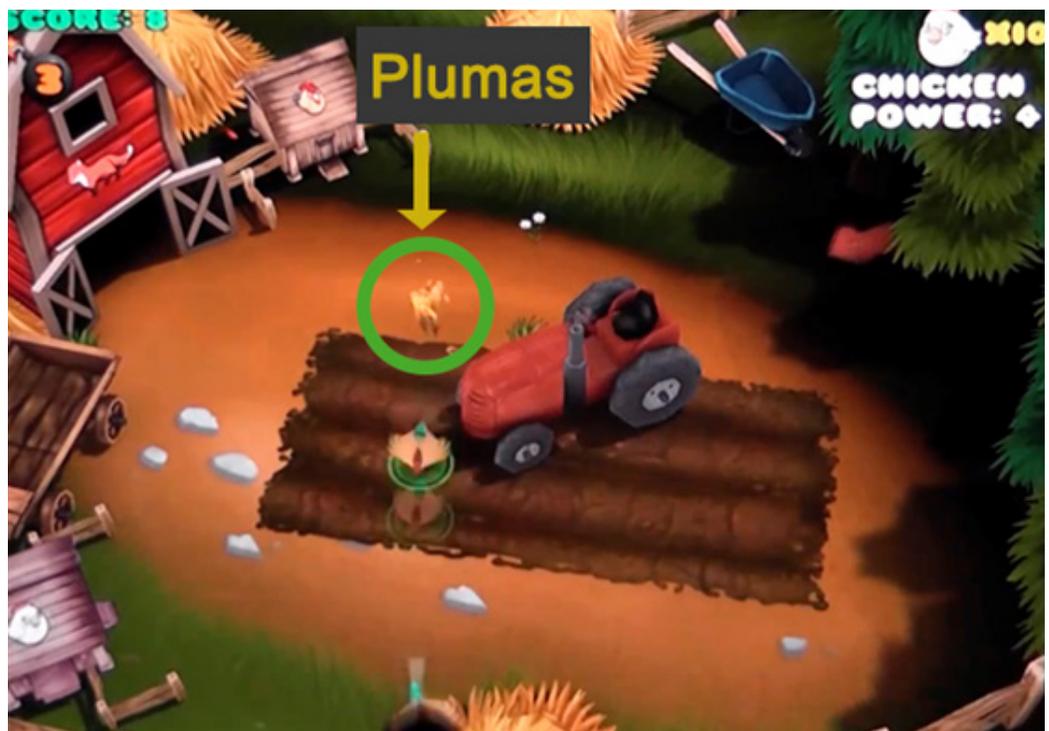
Como puede verse en la imagen 94, las plumas que aparecen en la pantalla salen de la gallina, marcadas con círculos verdes concéntricos debajo de ellas cada vez que es tocada por el jugador para meterla al gallinero, o igualmente

¹⁴ “Un sistema de partículas es una colección de muchas partículas mínimas que en conjunto representan un objeto difuso. Con el (paso del) tiempo, las partículas son generadas dentro de un sistema, se mueven y cambian dentro del sistema y mueren como parte del sistema.” La traducción es del programador José Medina.

93. Portada comercial de Chicken Coup.

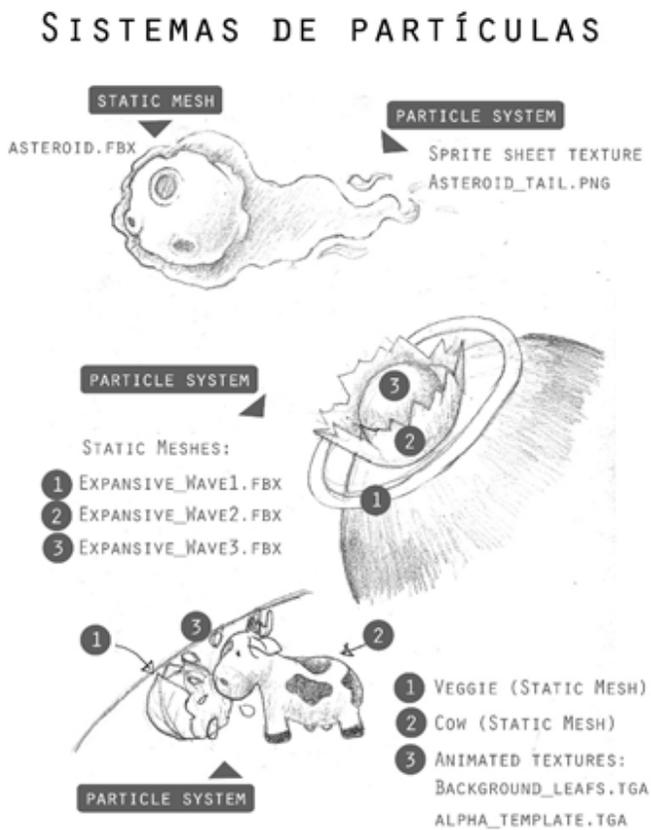


94. Screenshot de Chicken Coup (2011, Puppy Punch Productions).



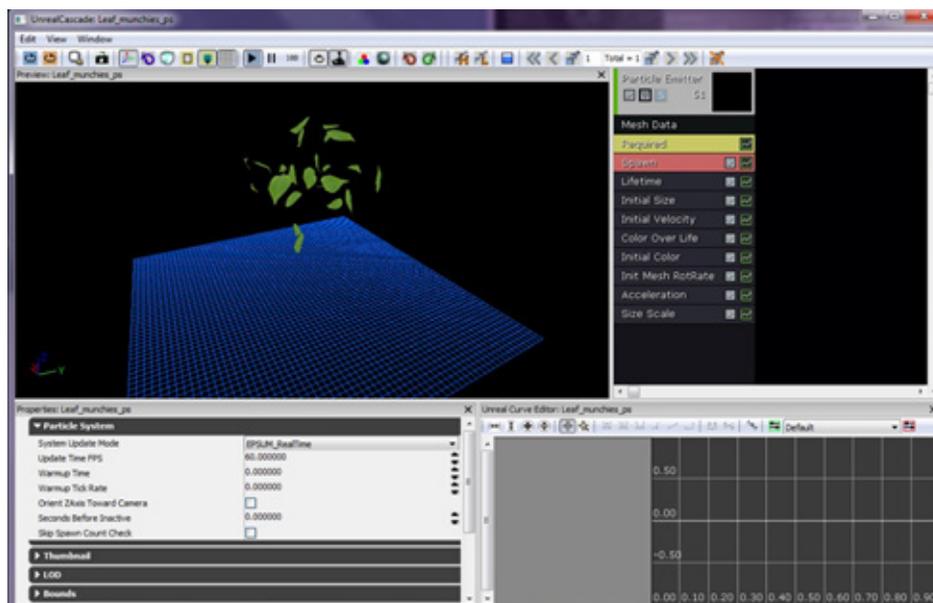
puede sacar plumas cuando un zorro se la lleva a las sombras de los árboles y arbustos para comérsela. Estas plumas son un sistema de partículas creado a partir de una sola partícula que, en este caso, sería un plano geométrico. Se le asigna una textura, que se proyectará en el plano de la partícula.

Como ejemplo, mostraré uno de los sistemas de partículas realizados, en específico, el último en el esquema de las hojas saliendo de la lechuga por las mordidas de la vaca:



95. Diagrama de sistemas de partículas iniciales.

96. Hojas de lechuga mordidas.



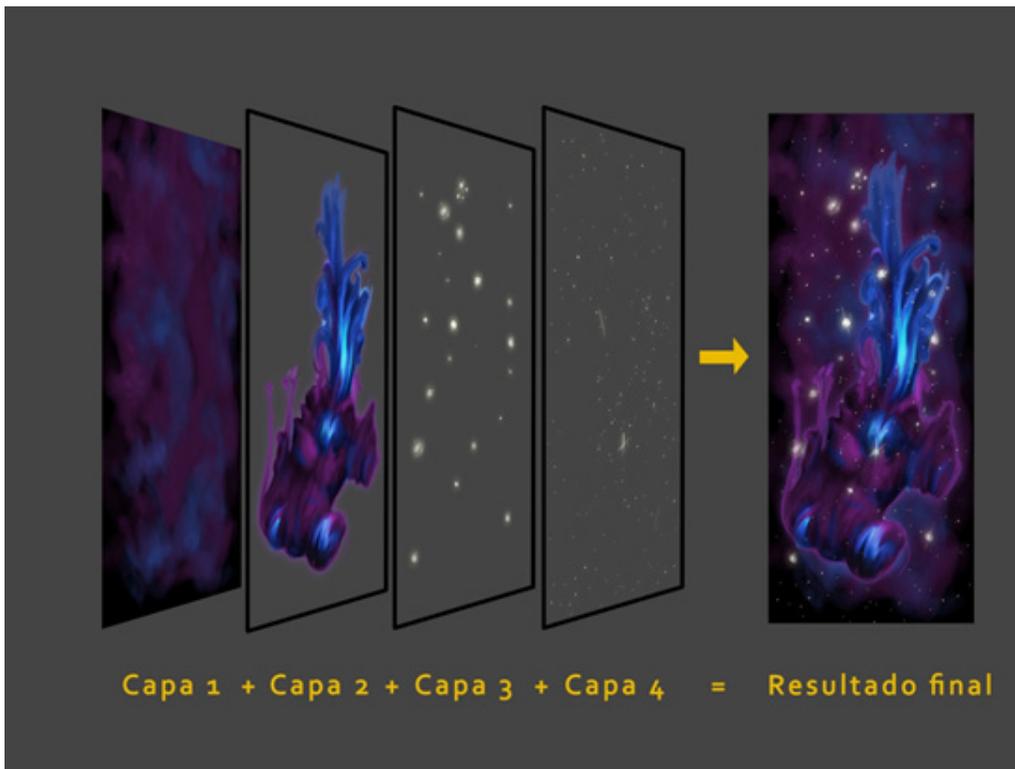
He aquí el editor de sistemas de partículas de UD: Cascade. En él, se especifica el tamaño de una partícula -en este caso, una de las hojas-, la velocidad a la que ésta viajará, en qué dirección, si desaparecerá después de cierto lapso, etc. Las hojas que se ven en la parte superior izquierda de este cuadro de diálogo corresponden a un solo plano geométrico con una textura de una hoja con transparencia proyectada sobre él. Para este efecto en las hojas, fue necesario hacer varios ensayos para que el movimiento de las partículas fuera coherente con la acción que sucedía.

Este elemento fue descartado y los efectos de partículas existentes en el juego fueron logrados por Fabián Hernández.

3.10.2 Iluminación

La iluminación igualmente tuvo constante evolución y al inicio, Fabián Hernández ya se encargaba de este aspecto. El cielo-universo que era el fondo del planeta debía ser constante, coherente, sin llegar a ser aburrido. Él me pidió hacer una serie de imágenes para que funcionaran como distintas capas del universo:

- En la primera capa, existiría un negro total como fondo con



97. Capas individuales que componen el “cielo”.

nebulosas tenues. En ésta no habría ninguna transparencia.

- En la segunda, sí habría transparencia en el fondo y habría nebulosas más opacas.
- En la tercera, habría galaxias lejanas, y por tanto, pequeñas (con fondo de transparencia).
- En la última, estrellas pequeñas más cercanas y con fondo transparente.

Este resultado fue tan sólo un ensayo y también tuvo problemas por ser demasiado complejo y que no era un diseño con simetría, para que lograrse el efecto de que fuera un cielo continuo. El cielo que se tiene actualmente en el juego tiene una composición mucho más simple, sin nebulosas, con estrellas en diferentes planos que circundan el planeta y con una ilusión de atmósfera que lo rodea. Este aspecto final se verá en los *screenshots* del siguiente apartado.

98. Build en una de las primeras versiones del juego, funcionando en iPad.



3.11 Carga de gráficos

En la imagen 81 se presenta el esquema de una de las versiones anteriores del juego, en el cual se explican las mecánicas y sus posibles inconvenientes.

3.12 Marketing

Elevator Games, hoy en día (noviembre de 2013) tiene su sitio web: www.elevatorgames.net (administrado por Stephanie Prodanovich) en el cual se puede leer la misión del estudio y sus integrantes líderes (no incluye a todos los miembros del equipo). Asimismo, incluye una página sólo para *Celleste*, a manera de blog.

Por otro lado, se cuenta con una página de *Facebook* (actualmente con 278 “me gusta”) y una cuenta de *Twitter* (con 237 seguidores), en las que se publican las actualizaciones más importantes del juego.

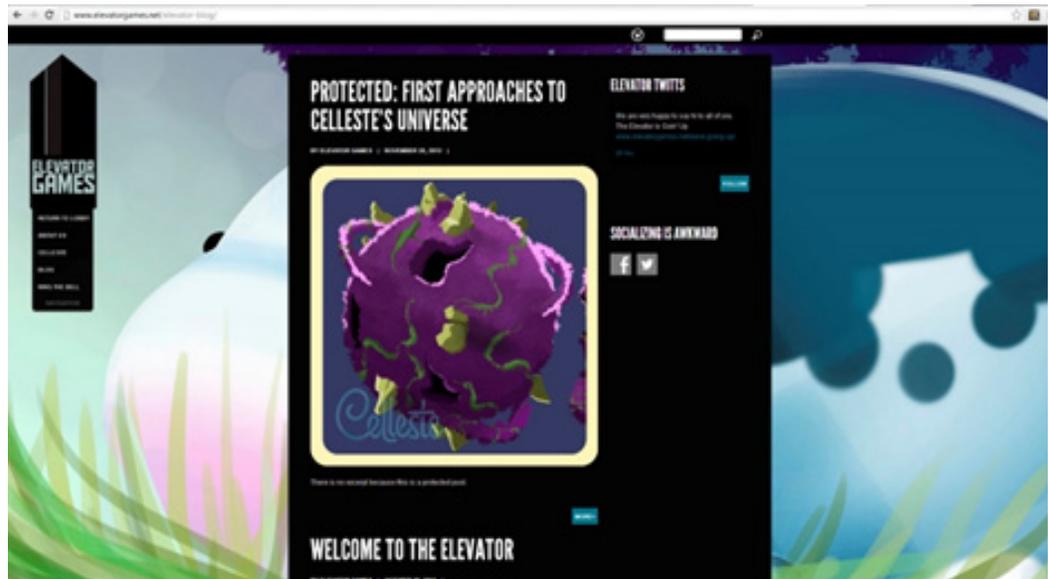
Las huellas más importantes a nivel publicitario que ha tenido *Celleste* (junto con *Elevator Games*) son cuatro hasta esta fecha. La primera es una publicación que tuvo en la



99 y 100. Capturas de pantalla del último build presentado en eventos oficiales en 2013 de Celeste.



101. Vista del sitio oficial de Elevator Games.



revista mexicana de videojuegos de *Game Master* en su edición de marzo de 2013 (imagen 104).

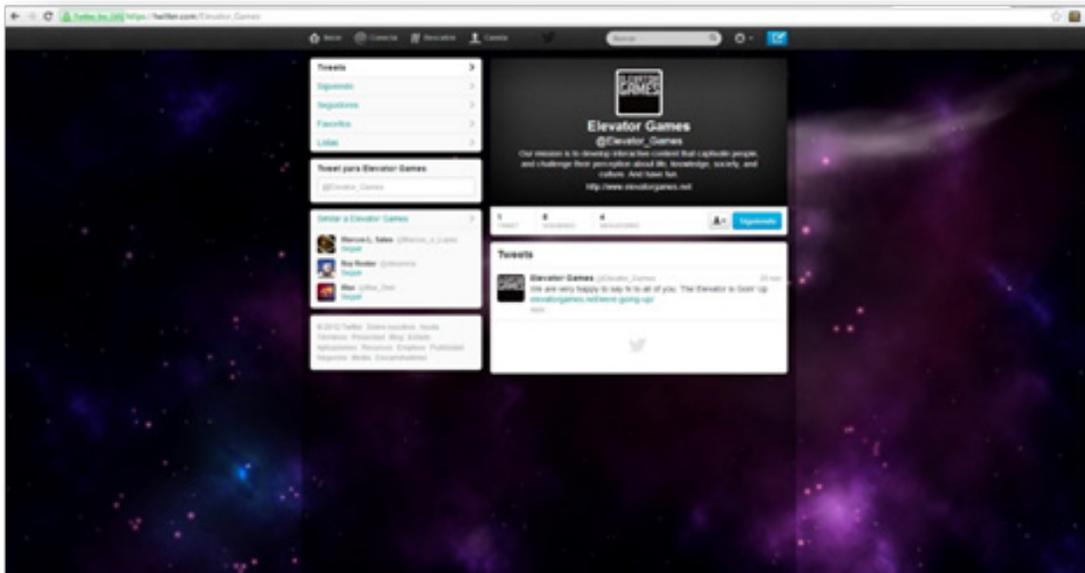
En octubre del 2012 se buscó la difusión del proyecto en eventos relacionados con videojuegos: “Conexión” y “EGS México”, pero por razones logísticas (de los eventos), se replanteó difundirlo en otros medios:

- En marzo de 2013 se organizó en San Francisco, California, uno de los eventos más importantes para desarrolladores de videojuegos en el mundo: la “Game Developers Conference”¹⁵. *Celleste* fue llevado para buscar **publishers** en grandes compañías. Actualmente se está en proceso de negociaciones.
- En julio de 2013 se presentó en la “Campus Party”, un evento de tecnología anual de la Expo Bancomer de Santa Fe. En la conferencia se explicó el proceso creativo y de producción, las ventajas y desventajas para llegar a la versión final. La presentación se grabó y transmitió en vivo para luego subirla a *YouTube* (Intergalaktea, 2013).
- En agosto de 2013 se presentó en la “Gamescom”

15 Conferencia de Desarrolladores de Videojuegos.



102 y 103. Perfil de Twitter y sitio de Facebook de Elevator Games, respectivamente.



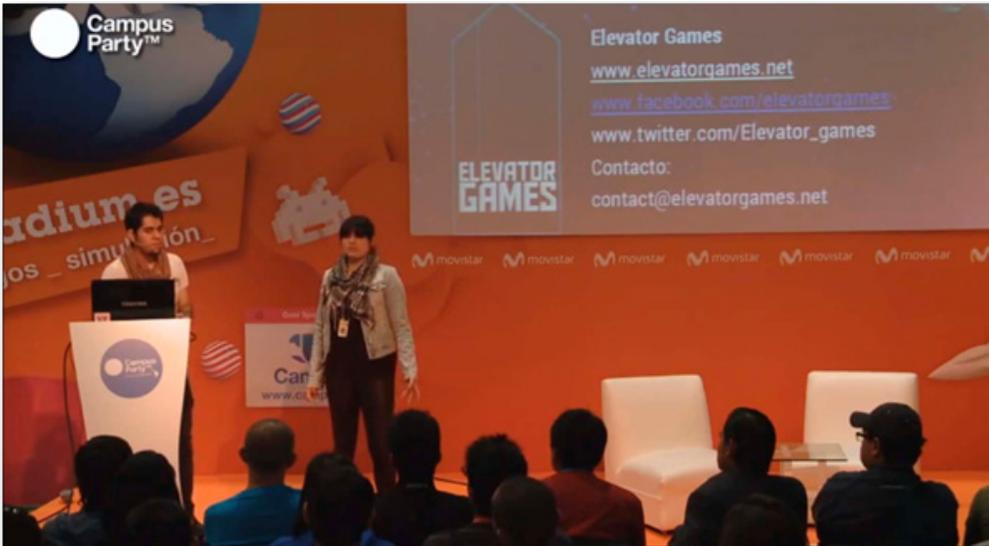
104. Artículo sobre Celleste de la revista de Game Master



en Colonia, Alemania. Sobre esta participación, hubo al menos tres publicaciones en línea: 1) En el sitio *De10* con el artículo “Videojuegos 100% mexicanos que llegarán a Alemania al Gamescom 2013”, escrito por Edna Zúñiga (2013); 2) en el sitio *Game Side Story* con “GC – Celleste” por LeenUyth (2013), describiendo las mecánicas de juego y; 3) en el sitio *Videogames Pockett* con “[GC13 - Preview] Celleste sur iPhone/iPad et Android : oh, la vache !”¹⁶ redactado por Olivier A. (2013). En este artículo se señala la ventaja de que, a falta de texto en el juego, no fue necesario hacer **localización**.

De manera oficial, el juego aún no ha sido lanzado.

16 “GC13 Preview: Celleste en iPhone/iPad y Android: ¡Oh, la vaca!”. La traducción es mía.



105. Screenshot de la conferencia. Gerardo Basurto y Stephanie Prodanovich presentando “El proceso creativo de un desarrollador independiente: Celleste y Elevator Games”.

Capítulo 4

Conclusiones

La necesidad del diseño está siempre presente en medios interactivos. El creciente mercado—engrosado especialmente por los jugadores casuales— indica, al menos en apariencia, que hay muchos temas para nuevos videojuegos sin que impliquen dañar económica, psicológica o moralmente al jugador.

La participación del diseñador gráfico y comunicador visual en proyectos de esta naturaleza es indispensable para la creación de una propuesta visual innovadora, pero es claro que en su formación debe haber algunas características de artista visual. Es por ello que las tareas para artistas de videojuegos tienen diferentes características. Un director creativo o un director de arte puede tener conocimientos avanzados de todas las áreas

de arte, pero a nivel sólo general, y un diseñador de interfaces de juegos masivos en línea ambientados en épocas medievales no hará el mismo trabajo que un diseñador especializado en modelado para juegos de plataforma de aspecto infantil con figuras amigables.

El diseñador y comunicador visual cuenta con las bases necesarias para introducirse en el campo del arte para videojuegos, pero debe estar consciente de qué área es la que le atrae más, y entonces, involucrarse más con los temas pertinentes. Si quiere avocarse a ser artista de ambientes, entonces parte de su antecedente en psicología del color, composiciones formales e informales pueden serle de mucha ayuda al crear un ambiente, pero deberá complementarse con más documentación. Así me lo ha enseñado mi experiencia laboral.

4.1 Experiencia personal y laboral

Hasta antes de iniciar el proyecto, tenía una idea muy rudimentaria sobre lo que significaba hacer un videojuego. En la ENAP pude cultivar un poco de ese interés en un curso intersemestral sobre diseño de videojuegos con el profesor Ulises Rosas, quien tiene experiencia en la Facultad de Ingeniería de la UNAM.

Hice mi primer juego con un programa llamado RPG Maker y fue una gran experiencia; a pesar de que el juego tenía muchos problemas y algunas instrucciones no funcionaban como había esperado, pero sería una prueba muy pequeña y efectiva de lo que en realidad implica hacer un juego desde cero. Como no tenía forma de saber mucho sobre este campo ni tenía algún guía experimentado, era fácil sentirse en un ambiente muy caótico. Pero después de investigar, descubrí que las mayores empresas de videojuegos en países que son potencias económicas (e incluso las independientes) cuentan con artistas de nivel universitario, frecuentemente, a nivel de maestría.

4.1.1 SODVI

Hace aproximadamente tres años me uní a esta sociedad con el objetivo de conocer métodos de producción de videojuegos y su teoría, además de poner los conocimientos

en práctica. Mi jefe de servicio social me había hablado de este proyecto estudiantil y tenía varios integrantes activos con tres proyectos en aquel entonces.

Yo me había integrado a un equipo de al menos ocho personas. Había un concepto inicial e historia creados por un estudiante de comunicación. Había tres programadores y el resto eran artistas, incluyéndome. Poco a poco me di cuenta de los conocimientos que nos hacían falta para poder organizarnos mejor y trabajar con un método ordenado. Finalmente, sugerí hacer cambios para lograrlo, basándome en una somera investigación que había hecho sobre el rol que debíamos desempeñar los artistas. En aquel entonces, yo seguía convencida de que era imperativo ser sólo Diseñador y Comunicador Visual para poder ejecutar las tareas de un artista de videojuegos y por ello había propuesto una idea muy rígida sobre el proceso que debíamos seguir. Algunos de los artistas eran Artistas Visuales y no concordaron con mi propuesta, que había sido apoyada por el líder del proyecto, así que decidieron abandonar el proyecto. Después de eso, poco a poco el resto del equipo se disolvió.

Ahora puedo entender que el factor humano de saber trabajar equipo es crucial para poder llevar a cabo cualquier proyecto. Si lo hubiese tomado en cuenta, tal vez habríamos podido avanzar. Otra de mis conclusiones de esa experiencia fue que el conocimiento que tenía en materia de diseño, geometría, fotografía, técnicas de representación gráfica, etc., eran muy útiles, pero debía encauzarlas en la dirección correcta.

4.1.2 DGTIC

Entretanto, mientras continuaba en la escuela y en SODVI, pude elegir mi servicio social en el Departamento de Realidad Virtual —dirigido por la Ing. María del Carmen Ramos Nava— de la DGTIC en la UNAM en 2010¹.

1 Me enteré que ofrecían un puesto de becario por carteles que había pegados en la escuela, pero tal parece, sólo yo solicité entrar. Llegué a pensar que los que llegaron a leer el cartel tuvieron miedo de acercarse a preguntar para más informes.



Imagen a): Diseño de interfaz realizado por mí para videojuego concebido en el Departamento de Realidad Virtual de la DGTIC en la UNAM. La imagen de fondo del barco no es de mi autoría.

Imagen b): Propuesta de diseño de menú del mismo videojuego realizado por mí. La imagen de fondo de ambos barcos fue un render (editado por mí) realizado por el diseñador Victor Hugo Franco.

Imagen c): Render de modelo de barco por Victor Hugo Franco, funcionando en la aplicación del juego

106 .Interfaz y modelado del barco.

Colaboré en un proyecto de videojuego educativo o de género **Serious games** haciendo diseño de interfaces y la construcción de un modelo 3D de un barco, mientras recibía la orientación del Lic. Víctor Hugo Franco (Imagen 106, incisos a), b) y c)).

Ahora bien, entre las bases que utilizó el Departamento de Realidad Virtual para desarrollar proyectos como el mencionado, cabe resaltar el concepto de Tecnologías de la Información como un vehículo del trabajo interdisciplinario.

Uno de los inconvenientes que me comentó Franco durante el desarrollo de este juego, fue que hay una diferencia abismal entre trabajos de carácter escolar y empresarial por razones obvias; en una universidad del tamaño de la UNAM, un proyecto así no tiene la presión constante de entregar resultados para vender o solamente lanzar un juego, como sucede en un estudio desarrollador. Es por ello que los avances logrados en un ámbito escolar pueden llegar a ser bastante escasos.

No supe mucho de este videojuego al terminar mi tiempo en el servicio social, sino hasta hace poco, en noviembre de 2013.

107. Screenshot del juego actual. Imagen facilitada por Víctor Hugo Franco.



Al conversar con Franco, me comentó acerca de la presencia del Departamento de Realidad Virtual en la edición de este año del evento de videojuegos de la “Electronic Game Show” que se lleva a cabo cada año. El proyecto finalmente tomó el nombre de *Ocean Math Wars*. Podo después el sitio de *Facebook* de la Sala Ixtli —parte de la DGTIC — hizo pública esta imagen al respecto del evento:

Fue un gran salto en cuanto a difusión, pero la realidad es que es insuficiente. La buena noticia es que sí hay interés, pero hay que incentivarlo. Proyectos como éste pierden voz entre todas las demás notas sobre el último videojuego lanzado al mercado por una gran compañía desarrolladora con títulos previos galardonados. Si al difundir *Ocean Math Wars*, se presenta como una iniciativa de la UNAM para estudiantes que están interesados en tecnología para videojuegos, el interés sería mayor, y podría destinarse más presupuesto para abastecer la demanda.



108. Imagen publicitaria de Ocean Math Wars.

4.1.3 Slang Studio

Poco después de terminar mi servicio, le comenté de mi inquietud por los videojuegos a Gonzalo Sánchez, **Lead Artist** en Sabarasa, después Slang Studio. Sánchez me contactó para hablarme sobre un puesto de becaria disponible para ser *junior artist*. Fue entonces que entré a dicho estudio y según mi apreciación, fue un buen inicio, pero tampoco fue suficiente.

Mis responsabilidades estaban muy restringidas y no tenía un horario que se extendiera más allá de cuatro horas, tres días a la semana. De quienes más pude aprender sobre estos procesos fue de mis compañeros de trabajo, quienes sí tenían jornadas completas de trabajo y, frecuentemente, cubrían horas extras para completar alguna tarea (más que nadie, los programadores). Esperaba aprender bases de modelado 3D para videojuegos, pero fue una de las tareas que menos me asignaron; fui encargada de diseñar y modificar algunos elementos de interfaces, junto con otros miembros del equipo de arte.

Una de las cosas que aprendí y valoro de la estancia fueron las formas de trabajo que incluían hacer reuniones de SCRUM, una forma de organización de equipo en la que se dividen las tareas a realizar por tareas aún más pequeñas, para completar paulatinamente características del juego en un orden coherente. Las reuniones se hacían a la misma hora, en la mañana, y eso evitaba que hubiese confusión respecto a lo que a cada uno tocaba hacer.

También empecé a conocer qué otros estudios mexicanos existían y habían existido, además de las razones por las que habían caído en bancarrota.

El estudio pasó por diversos problemas de cohesión grupal y varios fueron desertando por razones que mantuvieron para sí mismos. Para entonces, ya me encontraba redactando esta tesis, con la esperanza de poder hacer un videojuego con una idea mía y Gonzalo Sánchez me ofreció asesorías, pero era difícil hacer mi labor en el estudio, además de cumplir con mis materias en la escuela, así que después de ocho meses de trabajo en Slang, renuncié para poder dedicarme de lleno a mi investigación.

Poco después, me enteré de que Slang también terminaba labores. Varios integrantes de Slang Studio migraron a Slang Publishing, la división encargada de publicar juegos de aquél y otros estudios. Fueron estos compañeros de trabajo con quienes mantuve contacto y que después me comentaron sobre la idea que tenían sobre Celleste.

Los que después integramos a Elevator Games teníamos nula experiencia en hacer juegos desde cero, sin alguna empresa respaldándonos. Fue por ello que el camino recorrido fue bastante difícil, pero muy enriquecedor.

4.1.4 Elevator Games (Celleste)

En una experiencia que duró más de un año, hubo muchos tropiezos y victorias. Uno de los retos fue dominar dentro de lo necesario el Unreal Engine, porque queríamos aprender de él, y por ser un motor respetable en muchos países. De lo que investigué sobre esta herramienta, varias cosas fueron beneficiosas para el proyecto y constituyó una excelente base teórica, pues Basurto me había presentado los recursos audiovisuales de GDC Vault, el cual contiene

muchas conferencias de desarrolladores, artistas, músicos y productores de estudios de videojuegos en varios países. Mi papel como “administradora de arte” fue afectado porque tomé un segundo trabajo de tiempo completo en Ollin VFX y, visto en retrospectiva, admito que fue una falla importante. A pesar de haberme comprometido a continuar y de asegurar que mi trabajo en *Celleste* no se vería afectado, empecé a entregar tarde resultados de baja calidad, en parte, por el exceso de trabajo y en parte, porque yo aún no entendía por completo la visión del **game designer**. Así, mi participación fue disminuyendo poco a poco. Si he de ser completamente honesta, di prioridad al hecho de obtener una remuneración económica por sobre terminar *Celleste*. Es uno de los riesgos más peligroso en esta industria, como me di cuenta: la necesidad económica suele nublar y apagar las mejores ideas e intenciones.

4.1.5 Ollin VFX

Tuve la oportunidad de comparar una empresa de efectos especiales con una de videojuegos, codo a codo, pues poco después de iniciado el desarrollo de *Celleste*, conseguí otro puesto de becaria en el estudio de efectos especiales *Ollin VFX* como *CG artist* —donde también se me brindó apoyo para terminar este trabajo. Fue muy interesante saber qué puntos en común tenían estos dos campos.

La importancia de conocimientos de fotografía, por ejemplo, aún es aplicable a ambos campos, de VFX y videojuegos. Tanto en los videojuegos, como en el cine, es importante conocer los elementos básicos de composición, iluminación y diseño de imagen. Además, es una tendencia ya imparable la unión en este aspecto sobre la narrativa en videojuegos y cine. No obstante, las formas de optimizar una escena en videojuegos es muy distinta de las que hay en una escena para *CG* de Efectos visuales; esto debido a que en la escena *CG*, la imagen final será previamente procesada en un motor de **render** (*Renderman*, *Vray*, *Mental ray*, etc., suponiendo que contenga elementos 3D) para que, con esa secuencia de imágenes, un compositor digital añada elementos, corrija color, borre cosas como cables, vallas, etc, y así, complete la escena para que entre en la edición final y el espectador la vea.

En videojuegos, por su parte, el renderizado será en tiempo real, y por tanto, la optimización corresponderá a

las necesidades del motor del juego. Cada *asset* deberá ser exportado para armar un nivel de juego en ese motor con todos esos elementos, además de los efectos especiales del juego (como sistemas de partículas, agua) y los personajes animados, además de sus interfaces.

Este trabajo en Ollin VFX —que continúa hasta la fecha— ha sido el que más esfuerzo me ha exigido por ser una empresa consolidada, con al menos doce años de experiencia.

Mi primer jefe, el artista 3D Lluís Barceló aceptó entrenarme (nunca antes habían aceptado becarios) y no fue fácil. Fue sincero conmigo y me dijo que él no me hubiera contratado con el nivel que tenía cuando llegué al estudio. Estando ahí, yo tenía muy poca disciplina y, con cierta frecuencia, pasaba por múltiples fracasos que me hacían dudar de si quería permanecer ahí; me daba cuenta de que realidad no tenía idea de lo que implicaba un trabajo en una empresa de este tipo.

Lamentablemente, es uno de los principios que se han dejado de lado en México: la profesionalidad, hacerse responsable, tener fuerte ética laboral y enfrentar consecuencias de actos propios en el área de trabajo son aspectos que están ausentes en muchas empresas.

4.2 Consideraciones sobre la industria mexicana

Desde donde yo lo veo, la industria mexicana de videojuegos no puede crecer con una identidad completamente “mexicana”. No sólo por los antecedentes, sino porque la cultura nacional actual dificulta el trabajo en equipo. Hay quienes han podido organizarse por su propia iniciativa, y quizás crear una empresa, pero muy pocos han logrado mantenerse en pie y suelen obtener resultados mediocres.

Por otro lado, el inversionista mexicano de hoy, parece tener la creencia de que el tiempo no es una inversión y subestima el factor humano. Una de las constantes en videojuegos que he podido observar es que algunos productores crean una ilusión de tener recursos para atraer inversionistas, pero no toman en cuenta las implicaciones a largo plazo; al no tener esos recursos, el producto no saldrá a la luz y todos, incluidos los inversionistas, saldrán perdiendo.

Es un hecho que debe invertirse en educación para formar profesionales en área especializadas, pero para dar abasto a sus necesidades, los sueldos deben ser competitivos con las potencias internacionales. Sin embargo, para que esto suceda, la idiosincrasia de este país tendría que ser distinta. Quizás, si la nación retomara elementos de identidad indígena mexicana (pues estas comunidades tienen más presente el trabajo en equipo, de comunidad, y de un concepto de “nosotros”), sería más posible un desarrollo de la industria en el país, que estuviera en manos de mexicanos. Por desgracia, en la mayoría de los contextos industriales impera la individualidad y una falta de profesionalidad a nivel casi general. Es por ello que pocos (en comparación con otros campos de estudio) son los que se han destacado en videojuegos en el país.

En el contexto mexicano, con pocos estudios, lo más probable es que al iniciar como artistas de videojuegos, el diseñador y comunicador visual entre en un proyecto independiente donde tenga que involucrarse con personas de otra carreras o áreas, especialmente de programación, y en algunas ocasiones, más dedicados a la producción, y deba ocuparse de varias tareas; por tanto, es imperativo que sepa comunicarse con sus colegas de trabajo y esté al tanto de las nuevas tecnologías. Pero conforme acumule experiencia, el profesional en el área idealmente deberá elegir una especialidad.

A partir de mis pasadas experiencias, he descubierto que un cargo que puede satisfacer mi necesidad de crear mundos nuevos es el de Environment Artist. Conocer diferentes arquitecturas, diferentes acabados de materiales, investigar lugares inusuales en el planeta que sirvan de inspiración para la creación de un mundo virtual es de mis mayores motivaciones. Pero sería poco realista pensar que en poco tiempo podría hacer una carrera de ello en México. Hay grandes artistas aquí, sin duda, pero creo que el punto en común es que ellos lo hicieron posible (tal vez algunos tuvieron ventaja previa, por contar con mayores recursos). Una carrera en cualquier área de videojuegos es cara; por ello, me parece importante que se incentive desde escuelas públicas la participación de alumnos con la curiosidad y la motivación.

4.3 Juegos Indie: espacio de experimentación y libertad creativa

Muchos estudios independientes en el mundo están surgiendo gracias a la prosperidad de los dispositivos móviles, y la mayoría requieren muchos menos recursos que los estudios AAA. Y ciertamente —como mencioné en apartados anteriores—, hay mucha más libertad. Esto ofrece un reto para los desarrolladores, pero al mismo tiempo una oportunidad de destacar al ser realmente diferentes. Sunni Pavlovic, gerente general de Thatgamecompany lo confirma, después del éxito que tuvo *Journey*: la ventaja de hacer juegos que experimenten con ideas nuevas y únicas es que, por ser los primeros en proponer algo en específico, no pueden “equivocarse”, pues no hay reglas para esas ideas. Otro parámetro que hace diversa la experiencia es la inclusión de público no asiduo a los videojuegos, porque esto también permite concebir ideas diferentes (Reyes, 2013).

Creo que México tiene un lugar potencialmente importante. No por estudios grandes, no por inversionistas millonarios ni acuerdos con el gobierno que legislen en materia de videojuegos, sino por los estudiantes y aficionados que no temen experimentar para ofrecer un juego nuevo. Lo valioso de estos trabajos sería que no tuvieran que ver con vender productos de los monopolios televisivos o alguna otra empresa ajena al medio con tal de obtener beneficios monetarios.

Pero también necesitamos aprender a compartir la experiencia, tanto lo bueno como lo malo, para beneficio de otros. Los estudios mexicanos que desaparecieron no revelan las causas por las que “tiraron la toalla” y eso, invariablemente contribuye a la enfermedad, ya que permite la recurrencia del error al no sentar un precedente que enseñe a los demás cómo evitarlo.

Glosario

#	
.bmp	Formato de imagen Bit map o mapa de bits.
.fbx	Formato de imagen tridimensional propio de programas como 3ds Max, Maya, Blender y motores como Unity y UDK.
.png	De las siglas de Portable Network Graphics. Formato de imagen que soporta transparencias y comprime datos sin perder información. No está destinado para imágenes para impresión.
.tga	.TARGA o Truevision Advanced Raster Graphics Adapter. Formato de imagen que soporta hasta 32 bits de información (8 por canal: RGB y alpha).
.tiff	(Tagged Image File Format) Formato de imagen que, como los anteriores, comprime información.
3ds Max	Programa o software para modelado de objetos y escenarios en tercera dimensión. Pertenece a la empresa de Autodesk.
A	
After Effects	Programa de composición de efectos especiales para secuencias de video.
Ambient occlusion	Tipo de sombreado que se basa en una iluminación ambiental, por lo general, tenue. Así, los bordes que queden escondidos se verán más oscuros.
Arcade action	Género que combina dos: los de tipo “arcade” y los de acción. Implica condensación una experiencia de juego en poco tiempo.
Arte	En contexto de videojuegos, arte se refiere a todo recurso visual creado para un juego y esto incluye desde las ilustraciones, diseño gráfico, modelos en 3D hasta texturas, animaciones, efectos especiales, etc.
Art Director	Director de arte. Responsable de todo el Departamento de Arte, también llamado Lead Artist
Asset	Cualquier elemento individual usado en desarrollo de videojuegos, como un componente del arte, un personaje, una muestra, etc.

B	
(Filtro) Blending	En el programa de manipulación de imágenes Photoshop, existen diferentes formas de editar las capas de una imagen, y para cada una, se le puede aplicar un filtro o blending. Por defecto, éste está configurado en modo “normal”, es decir, no aplica ningún efecto.
Billboards	Es un plano geométrico con una imagen proyectada sobre él. Su particularidad es que siempre está dirigido a la cámara desde donde se observa el videojuego.
Boss	“Jefe”. Enemigo más poderoso en un determinado nivel de un juego.
Box modelling	Técnica de modelado en 3D. Se parte desde un cubo para darle forma al resto del objeto.
Build	Generalmente es una de varias versiones jugables de un juego.
C	
Canal alpha	En una imagen, pueden existir los tres canales de colores luz: rojo, verde y azul. En los formatos donde se soporta, está incluido el canal alpha en el cual hay información generalmente utilizada para indicar transparencia en una imagen. Esta información se indica con escala de grises. El valor de negro indica el punto de total transparencia y el valor blanco, el valor más opaco, mientras que los grises son puntos intermedios.
CEO	De las siglas en inglés Chief Executive Officer. Es el director de una empresa u organización.
CGI	(Computer Generated Imagery) Comprende cualquier imagen o personaje hecho artificialmente usando programación, código o software.
Cinemático	Secuencia de video que puede estar hecho de imágenes en 3D o acción real. Puede facilitar la narrativa de un videojuego al inicio de éste, entre un nivel y otro.
Cocinar (iluminación)	Implica pre-calcular la información de la iluminación emitida sobre un objeto en 3D para registrar la información en una textura. Este proceso es ventajoso para los casos en que el objeto en cuestión no se moverá y por tanto, no tendrá discrepancias con la iluminación dinámica o en tiempo real de un videojuego.
Concept art	Cubre toda ilustración hecha para efectos de definir el estilo general del videojuego. Es posible que en algunos casos, parte del concept art se incluya dentro del juego (como texturas de carteles).

Concept artist	Encargado de hacer el concept art.
Creative director	Director creativo. Define los parámetros visuales de un proyecto y así, poder administrar al equipo en esa dirección.
D	
Debugging	Proceso metódico de encontrar y reducir el número de bugs o defectos en un programa de computadora o en una pieza de hardware electrónico, haciéndolo entonces funcionar como se espera.
Diffuse map	Textura que contiene los datos básicos de color de un objeto sin que necesariamente se incluya información de sombreado.
Draw calls	Es toda ocasión en que se envía información al GPU (Graphics Processor Unit, un chip de procesamiento normalmente destinado para dibujar cada cuadro de una escena 3D), acción que se refleja en lo que se renderiza en tiempo real en pantalla.
E	
Extruir, Extrude	Acción de alzado o elevación de una cara o polígono plano para crear volumen.
F	
Flash	Programa para generar animaciones y/o aplicaciones interactivas; sea con imágenes en vectores o fotografías.
F/S (Frames per Second)	“Cuadros por segundo”. Se utiliza como unidad para establecer la velocidad a la que va un video o animación.
Feedback	Respuestas, reacciones en un proceso de retroalimentación al interior de un diálogo.
FMV (Full Motion Video)	Es una técnica de narración en videojuegos que se apoya de archivos de video pre grabados para mostrar la acción en un videojuego.
FPS (First Person Shooter)	“Tirador en primera persona”. Subgénero de videojuegos.
Freelance	Trabajo por cuenta propia, ocasional, sin plaza ni prestaciones lab
G	
Game art	Ilustraciones, paletas de colores, modelos en 3D, iluminación y otros recursos gráficos destinados a su implementación en un videojuego.

Game designer	Diseñador del juego. Encargado principal del diseño general que puede incluir historia, arte, diseño de niveles y otros aspectos.
Gameplay	Mecánicas y/o reglas de juego.
Geometría	Término usado para referirse a uno o varios modelos 3D.
Graphic Bible, Art Bible	Biblia de arte. Guía de estilo.
H	
High poly	Modelo en 3D con un alto nivel de detalle, y por tanto, de caras o triángulos.
HUD	Siglas del inglés Heads Up Display. Elementos como barras de vida, inventario, puntaje. Estos datos aparecen en plena acción de juego para indicar las estadísticas del personaje.
I	
Idle	“Desocupado”. Es uno de los estados o acciones de un personaje animado.
Interfaz	Es la herramienta gráfica utilizada para mediar entre el usuario y el producto; en el caso de esta tesis, de un videojuego.
J	
Junior artist	Integrante de un equipo de arte de videojuegos con rango menor.
K	
Keyframes	Cuadros clave. Término utilizado en animación para definir los cuadros con las poses principales en una acción
L	
Lead Programmer	Programador líder. Es el jefe del equipo de programación.
Localization	Localización. Éste es el proceso de optimizar el juego para otros mercados. Por lo general, cubre lo siguiente: Lenguajes, texto, diálogo.
Look and feel	Aspecto y sensación. Conceptos muy utilizados en arte para videojuegos.
Low poly	Modelo en 3D con poco nivel de detalle por su mínimo número de caras o tris.

M	
Marketing	Mercadeo. Implica la comercialización y distribución de un producto.
Maya	Programa de la empresa de Autodesk desarrollado para modelado 3D. Es muy popular en la industria de animación digital.
Mip maps	Serie de imágenes precalculadas, cada una de menor calidad que parten de una original. Se utilizan para ahorrar memoria en aplicaciones 3D.
Mocap (Motion Capture)	Captura de movimiento. Es una técnica de grabación de acciones de humanos y animales. Se utiliza para ello sensores de movimiento ubicados en puntos clave del cuerpo y varias cámaras distribuidas en el set captan esta información que los sensores reciben.
Motor de juego (Game engine)	Un motor de juegos es un programa que establece las reglas del universo donde funciona el juego y por ejemplo, calculará la física de éste. Establece las reglas que debe obedecer (velocidad de viento, manejar texturas, hacer funcionar sistemas de partículas, etc.
Multiply	Es uno de los filtros o Blendings de Photoshop.
Mudbox	Software de esculpido en 3D. La diferencia con programas como Maya y 3ds Max, es que los softwares de esculpido tienen la capacidad de soportar renderizado de millones de polígonos en pantalla.
N	
Normal map	Mapa de textura que contiene la información del relieve y detalles de un modelo de alta calidad para que, una vez aplicado sobre una geometría, interactúe con la luz emitida sobre ésta y le otorgue mayor detalle. Se compone de datos en cada canal de color, cada uno, correspondiente a un eje en un espacio 3D: canal de color rojo (R): eje X, canal de color verde (G): Y, canal de color azul (B): Z. NPC (Non Player Character)
NPC (Non Player Character)	Trad. Personaje no jugador. Es todo aquel personaje animado que ejecuta acciones debido a la inteligencia artificial programada para éste.

O	
Outsourcing	Tipo de contratación para realizar un trabajo de forma externa en una empresa, sin contar con prestaciones de ley.
Output	Se refiere a la información que genera el programa o juego para el usuario o jugador.
P	
Partículas (Sistemas de partículas)	Un sistema de partículas es una colección de muchas partículas mínimas que en conjunto representan un objeto difuso. Con el (paso del) tiempo, las partículas son generadas dentro de un sistema, se mueven y cambian dentro del sistema y mueren como parte del sistema.
PC (Player Character)	Personaje del jugador. Como el nombre indica, es el personaje que el jugador controla. Player Directed Character Personaje dirigido por el jugador. Es un personaje que es influenciado por su inteligencia artificial, pero que es influenciado por las órdenes del jugador.
Photoshop	Software de manipulación y creación de imágenes, sean éstas fotografías o ilustraciones.
Pipeline	El camino que deben seguir recursos individuales para facilitar el desarrollo del juego.
Primitivo	Término utilizado para denominar las formas básicas desde las que, por lo general, se parte para hacer un modelo 3D (cubo, esfera, pirámide, cilindro, etc.)
Prop	Significa "propiedad". Cualquier objeto físico usado dentro de una escena a incluir dentro de un escenario.
Publisher	Trad. Empresa publicadora de videojuegos.
Punto de pivote	Punto de un objeto 3D desde el cual se parte para hacer transformaciones como rotaciones, traslaciones y para modificar su escala.
Px	Píxeles. La unidad básica de una imagen computarizada.
R	
Render	Creación o dibujado de una imagen.
Rig	Esqueleto de un objeto o personaje animado.
Rigger	Constructor de esqueletos. Esta persona debe tener conocimientos de animación.
Rigging	La acción de otorgarle un esqueleto a un modelo 3D.

S	
Scale	Escala
Screenshot	Captura de pantalla. Imagen tomada de una secuencia de video.
Serious games	Género de videojuegos de tipo educativo.
Shader	Programa que especifica cualidades de sombreado de un material.
Shading	Proceso de sombreado de figuras tridimensionales.
Shell (coraza)	En términos de UV Maps, las diferentes partes de un mapa que ha sido dividido para evitar deformaciones en la textura.
Side-scroller	Género de videojuegos que ofrece libertad de movimiento a los lados, arriba y debajo.
Smoothing groups	Grupos de suavizado. Contiene la información de un modelo 3D respecto a sus normales “suavizadas”.
Software	Programa.
Specular map	Mapa en escala de grises que indica la brillantez u opacidad de una textura. El valor de mayor ausencia de color indica la máxima opacidad, y el de color blanco, una superficie pulida.
Sprite	Una imagen estática o una animación. Es una de las partes que compone una escena y economiza memoria.
Static mesh	Trad. Malla estática. Se refiere a un modelo 3D sin un “esqueleto” o animación que constituyen los escenarios en un motor de juego.
T	
Target	Objetivo demográfico al que se destina un producto.
Technical artist	Artista técnico. Funge como un puente entre artistas, diseñadores y programadores de un juego, pues cuenta con conocimientos de las áreas de programación y arte.
Tester	Probador. Encargado de jugar varias veces un juego con el fin de detectar fallas, errores y ofrecer una valoración general del producto.
Transparencia	Propiedad de una imagen para mostrar objetos a través de ella.
Tris (Triangles)	Sitio web gratuito que contiene un sistema gráfico de organización de tareas a realizar por una o más personas.
Trello	Triángulos. Polígonos o caras que componen un modelo tridimensional. Como su nombre indica, es un plano de tres vértices. El conteo de éstos es importante para tener siempre un modelo optimizado.

U	
UDK (Unreal Development Kit)	Software para el desarrollo de aplicaciones con el motor Unreal Engine.
Unreal Engine 3	Motor de juego desarrollado por el estudio Epic Games.
Up axis	Eje de “arriba”. Puede ser el eje Y o X.
UV Maps	Mapas de UVs. Son los “planos” de un modelo 3D que indican en coordenadas 2D la ubicación de los vértices de que está compuesto.
V	
VFX (Visual Effects)	Efectos visuales, efectos especiales.
W	
Wireframe	Estructura visible compuesta por aristas de un objeto 3D.
X	
Xnormal	Software que tiene la función de renderizar varios tipos de mapas de textura, como normal maps, height maps, ambient occlusion, etc. a partir de las versiones en high-poly y low-poly de un objeto 3D en formato .obj
X,Y,Z (Vistas Ortogonales)	En un entorno 3D, son las coordenadas o ejes que indican profundidad, ancho y alto. En un software, se observa la información de estas coordenadas desde cámaras: desde vista superior, lateral y frontal.

Bibliografía

- ACHA, Juan (1995). “1. Lo estético y lo artístico diferenciados” y “6. Los diseños. Artes tecnológicas” en *Introducción a la Teoría de los Diseños*, 3a ed., México: Trillas. Pp. 17-25 y 75-94.
- BASURTO, Gerardo (2013). “Arte y motivaciones” en *Documentos privados del Proyecto Celeste*.
- COSTA, Joan (1999). *La comunicación en acción. Informe sobre la nueva cultura de la gestión*, Barcelona: Paidós. (Papeles de comunicación, 26). Pp. 13-16.
- DILLE, Flint y John Zuur Platten (2007). “Level 5. Building characters” en *The ultimate guide to videogame writing and design*, Nueva York: Skip Press. Pp. 65-88.
- DONDIS, Donis A. (1973). “3. Elementos básicos de la comunicación visual” en *La sintaxis de la imagen. Introducción al alfabeto visual*, 8a ed., Barcelona: Gustavo Gili. (GG Diseño). Pp. 53-82.
- ISBISTER, Katherine y Noah Schaffer (2008). “Introduction” y “Chapter seven. Usability and Playability Expert Evaluation” en *Game usability: Advice from the experts for advancing the player experience*, Burlington, MA: Morgan Kauffman / CRC Press Taylor & Francis Group. Pp. 3-5 y 91-111.
- KENNEDY, Sam R. (2013). “Video Game Jobs Defined” en *How to Become a Video Game Artist: The Insider’s Guide to Landing a Job in the Gaming World*, New York: Crown Publishing Group. P 13.
- KRAWCZYK, Marianne y Jeannie Novak (2007). “Chapter 4. Game Storytelling devices: integrating game-specifics elements into a story” en *Game Development Essentials: Game Story & Character Development*, Nueva York: Delmar Cengage Learning. (Game Development Essentials). Pp. 83-104.
- NEWMAN, Rich (2009). *Cinematic game secrets for creative directors and producers*, Estados Unidos: Focal Press.
- OMERNICK, Matthew (2004). *Creating the art of the game*, Berkeley: New Riders. (NRG).
- RABIN, Steven (Ed.) (2010). *Introduction to Game Development*, 2a ed., Boston, MA: Course Technology Cengage Learning.
- SÁNCHEZ, Paulina (2011). “Revelaciones sobre Juegos mexicanos, parte 1”, en *Game Master*, Año 2, Número 25, Noviembre, México: Grupo Medios. Pp. 42-47.
- THOMPSON, Jim, Barnaby Berbank-Green y Nic Cusworth. (2007). “02: El proceso” en *Videojuegos. Manual para diseñadores gráficos*, Barcelona: Gustavo Gili. (GG Diseño).
- VILCHIS, Luz del Carmen (1998). *Metodología del diseño. Fundamentos teóricos*, México: Centro Juan Acha A.C.
- WONG, Wucius (1988). “Primera parte. Los principios del diseño” en *Principios del diseño en*

color, 2a ed., Barcelona: Gustavo Gili. (GG Diseño). Pp. 3-24.
ZIMMERMAN, Yves (1998). *Del Diseño*, Barcelona: Gustavo Gili.

Fuentes electrónicas

- A., Olivier (2013). “[GC13 - Preview] Celleste sur iPhone/iPad et Android : oh, la vache !” en el sitio de Videogames Pockett: http://www.pockett.net/n18858_iPhone_Celleste_sur_iPhone_iPad_et_Android_oh_la_vache (Consultado el día 9 de noviembre de 2013).
- BLIZZARD Entertainment (2011). “Videos” en la página de “20 Aniversario” del sitio oficial del estudio: <http://us.blizzard.com/es-mx/company/about/b20/videos.html> (Consultado el día 9 de noviembre de 2013).
- BRISCOE, Robert (2012). “Game Developers Conference. The art of Dear Esther: Building an environment to tell a story” en el sitio oficial de GDC Vault: <http://gdcvault.com/play/1015893/The-Art-of-Dear-Esther> (Consultado el día 9 de noviembre de 2013).
- ENAP (2013). “Alumnos. Licenciaturas. Diseño y Comunicación visual” en la página oficial del plantel: <http://www.enap.unam.mx/acceso/alumnos.php> (Consultado el día 9 de noviembre de 2013).
- GUSTAFSSON, Anders (2009). “Character generation” en el blog oficial de The Dream Machine: <http://dreammachinegame.blogspot.mx/2009/11/character-generation.html> (Consultado el día 9 de noviembre de 2013).
- INTERGALAKTEA (2013) “CPMX4 - El proceso creativo de un desarrollador independiente: Celleste y Elevator Games” en YouTube: <http://youtu.be/GcVRhlySP64> (Consultado el día 9 de noviembre de 2013).
- LARAMEE, Francois Dominic (1999). “The Game Design Process”. http://www.gamedev.net/page/resources/_/reference/103/general-game-design/222/the-game-design-process-r273 (Consultado el día 9 de noviembre de 2013).
- LEENUYTH (2013). “GC – Celleste” en el sitio Game Side Story: <http://www.gamesidestory.com/2013/08/28/gc13-celleste/> (Consultado el día 9 de noviembre de 2013).
- LICHTNER, Christian. (2012). “Game Developers Conference. The Art of

- Diablo III” en el sitio oficial de GDC Vault: <http://gdcvault.com/play/1015306/The-Art-of-Diablo> (Consultado el día 9 de noviembre de 2013).
- MIGHT & DELIGHT (2013). “Studio” en su sitio oficial: <http://mightanddelight.com/index.php?page=studio> (Consultado el día 9 de noviembre de 2013).
- MOSCARITOLO, Angela (2013) “Journey Takes Game of the Year at GDC Awards” en PC Mag: <http://www.pcmag.com/article2/0,2817,2417198,00.asp> (Consultado el día 9 de noviembre de 2013).
- PID (2013). “Welcome to the website for Pid - A unique, retro-flavored adventure game!” en su sitio oficial: <http://www.pidgame.com/> (Consultado el día 9 de noviembre de 2013).
- Q.U.B.E. (2013). “About” en su sitio oficial: <http://qube-game.com/about/> (Consultado el día 9 de noviembre de 2013).
- REYES, Emilio (2013). “Thatgamecompany: la clave del éxito está en atreverse a experimentar” en el sitio de Atomix. Never grow up: <http://atomix.vg/2013/11/11/thatgamecompany-la-clave-del-exito-esta-en-atreverse-a-experimentar/> (Consultado el día 9 de noviembre de 2013).
- ROSARIO, Jimmy (2005). “La Tecnología de la Información y la Comunicación (TIC). Su uso como Herramienta para el Fortalecimiento y el Desarrollo de la Educación Virtual”: <https://docs.google.com/document/preview?hgd=1&id=1dJbf7Q-ihdrqZZP0v-SPpHw7SF5GH8Dt0dmhhezWLk> (Consultado el día 9 de noviembre de 2013).
- SHIFFMAN, Daniel (2012). “Chapter 4. Particle systems” en The nature of code. Simulating natural systems with processing, Mountain View, CA: Creative Commons. Versión digital: <http://natureofcode.com/book/chapter-4-particle-systems/> (Consultado el día 9 de noviembre de 2013).
- STATE OF PLAY GAMES (2013). “Lume” en su sitio oficial: <http://www.stateofplaygames.com/work/lume/> (Consultado el día 9 de noviembre de 2013).
- THE DREAM MACHINE (2013). “Página principal” en su blog oficial: <http://dreammachinegame.blogspot.mx/search?updated-max=2010-04-22T12%3A37%3A00-07%3A00&max-results=100&start=30&by-date=false> (Consultado el día 9 de noviembre de 2013).

- TINY&BIG: GRANDPA'S LEFTOVERS (2013). "The story" en su sitio oficial: <http://tinyandbig.com/> (Consultado el día 9 de noviembre de 2013).
- UNMECHANICAL (2013). "About Unmechanical" en su sitio oficial: <http://unmechanical.net/what-is-unmechanical/> (Consultado el día 9 de noviembre de 2013).
- WARREN, Jamin (2013). "Is Journey creator Jenova Chen the videogame world's Terrence Malick?" en el sitio Kill Screen. Games. Play. Life.: <http://killscreendaily.com/articles/interviews/journey-creator-jenova-chen-videogame-worlds-terrence-malick/> (Consultado el día 9 de noviembre de 2013).
- ZÚÑIGA, Edna (2013). "Videojuegos 100% mexicanos que llegarán a Alemania al Gamescom 2013" en el sitio De10 de El Universal: <http://de10.com.mx/cultura-digital/2013/9-videojuegos-que-representaran-a-mexico-en-alemania-17195.html> (Consultado el día 9 de noviembre de 2013).