



Universidad Nacional Autónoma de México
Facultad de Artes y Diseño

TITULO:
DISEÑO Y DESARROLLO DE VIDEOJUEGO CON CONSTRUCT 2, INTRODUCCIÓN
A LA LÓGICA DE INTERACCIÓN PARA NO PROGRAMADORES

TESIS
Que para optar por el grado de:
LICENCIADO EN DISEÑO Y COMUNICACIÓN VISUAL

Licenciatura en Diseño y Comunicación Visual

PRESENTA:
Francisco Gabriel Ortuño Silva

DIRECTOR DE TESIS:
Licenciado Carlos Emilio Cardoso Olmedo

México CDMX 2021



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.



Universidad Nacional Autónoma de México

Licenciatura en Diseño y Comunicación Visual
Facultad de Artes y Diseño

TITULO:
DISEÑO Y DESARROLLO DE VIDEOJUEGO CON CONSTRUCT 2, INTRODUCCIÓN
A LA LÓGICA DE INTERACCIÓN PARA NO PROGRAMADORES

TESIS
Que para optar por el grado de:
LICENCIADO EN DISEÑO Y COMUNICACIÓN VISUAL

PRESENTA:
Francisco Gabriel Ortuño Silva

DIRECTOR DE TESIS:
Licenciado Carlos Emilio Cardoso Olmedo

México CDMX 2021

Agradecimientos

A mi Familia a mis Amigos y a los Profesores que lo hacen posible.

Indice

Indice

Introducción	6
Capítulo 1. La industria de los Videojuegos y el rol del Diseñador	8
1.1 Contexto Tecnológico	8
1.2 La Industria Global y Nacional de Videojuegos	18
1.3 El diseñador y el desarrollo de videojuegos	30
1.4 Los Gráficos en los videojuegos	34
1.5 Estilos gráficos de videojuegos	36
1.6 Uso de programación por parte del diseñador	47
1.7 Adopción del pensamiento computacional	56
Capítulo 2. Introducción a los Videojuegos	58
2.1 Definición de Videojuegos	58
2.3 Historia de los videojuegos	63
2.4 Fines de los videojuegos	79
2.5 Géneros	81
2.6 Plataformas	85
2.7 Modos de Juego	89
2.8 ESRB - Clasificación de videojuegos por contenidos	91
2.9 Game Engines	93
2.10 Tipos de programación usados en Game Engines	96
2.11 Estructuras de control para programación visual de videojuegos	100
2.12 Programación profesional de videojuegos	102
Capítulo 3. Producción de Videojuegos	106
3.1 Fases de Producción	106
3.2 Roles de desarrollo	117
3.3 Recursos Online para aprender a programar	124
3.4 Introducción a la Programación	127
3.4.1 Componentes básicos de programación	128
3.4.2 Estructura secuencial	131
3.4.3 Estructuras Selectivas	133
3.4.4 Estructuras repetitivas	141
3.5 Software para elaborar arte para videojuegos (Game Art)	146
3.6 Uso de Construct 2 para la creación de un videojuego	151
3.7 Estructuras de control aplicadas a Construct 2	155
3.8 Creación de Videojuego: Cazareliquias	159
3.9 Documento de Diseño de Juego (GDD)	165
4.0 GDD Videojuego Cazareliquias	167
Conclusión	177
Anexo	189
Glosario	322
Bibliografía	336

Introducción

Introducción

La presente investigación se refiere al tema de los videojuegos con un enfoque donde se exponen sus características, historia, las partes necesarias para su producción, así como la creación de prototipos o juegos completos de forma accesible tomando como referencia la industria global y nacional dentro del contexto tecnológico de la Cuarta Revolución Industrial.

Dentro de la industria del Entretenimiento el término Videojuego es definido como un producto en forma de juego electrónico que genera retos y desafíos que el jugador debe superar con el objetivo de obtener la victoria, provocando diversión en el proceso de uso.

Aunque de manera profesional la interacción que genera el *Gameplay* de un videojuego es creada por medio de código de programación en un entorno dado por ingeniería de software el tema se aborda desde la perspectiva en donde se facilite la creación de videojuegos por personas sin conocimientos estrictos de programación dando pauta para generar contenidos originales sin conocimiento técnico.

Para facilitar esta tarea se han descrito las características y ventajas en el uso de la programación visual, sistema por el cual está basado *Construct 2*, el *Game Engine* usado para generar y manipular los elementos que producen interacción de forma gráfica e intuitiva.

Un factor que permite la creación de prototipos o juegos más elaborados es el uso de estructuras básicas de control de programación, por lo que se contempla la descripción de conceptos introductorios sobre programación, así como su relación práctica con la programación visual por medio de ejemplos comparativos de código usados dentro de las hojas de eventos del software de *Construct 2*.

También se considera la producción de *Game Art* por medio del uso de software 2D y 3D que complementa la producción interactiva del videojuego, creando un rol de trabajo más completo.

De igual manera se presenta el desarrollo de un videojuego usando el Documento de Diseño de Juego (GDD), documento necesario para la planeación que permite tener un listado sobre los elementos del juego a producir y sus requerimientos, así como también genera un orden y comunica al equipo de desarrollo sus funciones y obligaciones dentro del ciclo de trabajo.

El objetivo principal de la investigación consiste en incentivar a Diseñadores en la

Introducción

creación de videojuegos presentando las bases necesarias para comprender tanto los roles y ciclos productivos como las herramientas para materializar sus propias ideas logrando una motivación especial al ver su trabajo realizado demostrando que la producción de un videojuego puede ser posible y que siembre la semilla que le permita ampliar su percepción productiva para continuar investigando sobre programación sentando las bases para futuras especializaciones ya que su formación académica original generalmente no cubre esa área, además al existir una demanda laboral donde la programación y el diseño se integran en múltiples áreas, resulta de provecho el conocer los puntos fuertes de la mezcla de temas.

Por medio de la investigación se ha conseguido compilar y ordenar temas y conceptos que sufren una actualización constante dentro de la industria del videojuego, permitiendo el registro del contexto y la forma de adopción de *Game Engines* con programación visual por parte de un público mayor.

También se resalta la importancia de aprender programación para aumentar la productividad tras la adquisición de beneficios estructurales para la resolución de problemas dados por el pensamiento computacional, dentro de entornos interactivos o áreas relacionadas bajo los principios tecnológicos fundamentados en la Cuarta Revolución Industrial.

Para complementar la teoría con la práctica se ha incorporado a la investigación la realización de un videojuego con funcionalidad cubierta al cien por ciento donde por medio del Documento de Diseño de Juego (GDD) se ejemplifica la planeación general, los requerimientos y las partes que conforman el videojuego, vinculando la comprensión de conceptos completando así el ciclo de creación de un proyecto real.

El videojuego de nombre Cazareliquias puede ser descargado de la página:

<https://cazareliquias.itch.io/relic-hunter>

o en el repositorio en la nube de google:

<https://drive.google.com/drive/>

[folders/1ZHguL8aONbaHnrEN0Thr30n1Zzn1ZKqz?usp=sharing](https://drive.google.com/drive/folders/1ZHguL8aONbaHnrEN0Thr30n1Zzn1ZKqz?usp=sharing)

De igual manera, la investigación busca sentar precedente y estudio a posterior sobre el rol que el diseñador debe de afrontar ante los retos de crear conceptos sólidos de diseño aplicados a *Game Art* y de que manera su posición a nivel gráfico sirve para innovar en teoría de diseño presentando metodologías y estilos propios que respondan a derechos de autor y copyright todo a través del uso de tecnologías emergentes en un entorno que favorablemente evoluciona constantemente buscando el estar actualizado dentro de la industria, partiendo bajo el sustento de las bases que la escuela a generado a lo largo de su historia en forma de tradición artística.

Capítulo 1. La industria de los Videojuegos y el rol del Diseñador

1.1 Contexto Tecnológico

Capítulo 1. La industria de los Videojuegos y el rol del Diseñador

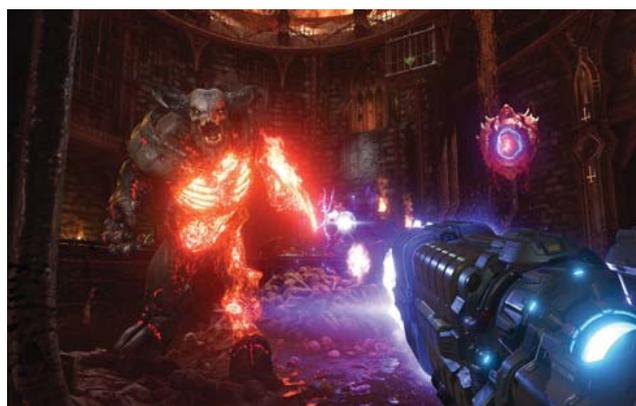
1.1 Contexto Tecnológico

En la actualidad los videojuegos generan experiencias sumamente enriquecedoras al consumidor, a nivel cognitivo con historias fantásticas y mecánicas de juego entretenidas e innovadoras y a nivel sensorial con gráficos y sonidos en alta resolución y mundos sorprendentes desbordantes de creatividad, por lo que forman parte de la cultura y hasta en ocasiones son considerados obras de arte.



1.1 Super Mario Odyssey - Nintendo Switch.

watson.ch. (2020). Super Mario Odyssey - Nintendo Switch [Imagen]. Recuperado 24 Abril 2020, de <https://www.watson.ch/imgdb/cf7b/Qx,E,0,0,1280,720,533,300,213,120/3496040621939683>



1.2 Doom Eternal

Id Software. (2020). Doom Eternal [Imagen]. Recuperado 24 Abril 2020, de <https://www.canardpc.com/403/flammes-des-annees-90-doom-eternal>

La palabra videojuegos usada de forma genérica se refiere a la industria de producción, comercialización y consumo de juegos electrónicos destinados al mercado del entretenimiento interactivo.

Así mismo, estos juegos electrónicos también son denominados videojuegos (*Videogames*) refiriéndose a ellos como el software usado para jugar en consolas, smartphones y computadoras, por ejemplo el juego de *Super Mario Odyssey* es un videojuego (*software*) en formato físico de cartucho que se introduce en la consola Switch de Nintendo (*hardware*) para poder jugar, o el título de *Death Stranding*

Capítulo 1. La industria de los Videojuegos y el rol del Diseñador

1.1 Contexto Tecnológico

(*software*) que es un videojuego en formato físico de disco *Blu-ray* para la plataforma PS4 de Sony (*hardware*).

Los juegos electrónicos (videojuegos) pueden variar en temáticas y estilos gráficos, pero su fundamento principal reside en presentarle al jugador (también llamado usuario), retos interactivos de diferentes dificultades que tendrá que resolver para cumplir con el objetivo final que es conseguir la victoria.

El primer videojuego del mundo fue *Bertie The Brain* un juego Tic-Tac-Toe y tuvo su origen en 1950 por el Dr. Josef Kates para la exhibición de la Universidad de Toronto, desde esa fecha hasta la actual novena generación de consolas conformada por PS5 y Xbox Series X/S es que los juegos han pasado por muchos cambios y transformaciones que van desde la evolución en la complejidad de la jugabilidad, hasta gráficos más detallados y realistas en sus formas y colores, pasando por todo tipo de controles para la manipulación de personajes y elementos de juego; esta evolución de generación en generación se da por el avance en materia de cómputo, por lo que es una industria que va a la par de la innovación y el desarrollo tecnológico de su tiempo.

Las principales características que han marcado a la actual octava generación de consolas que está por cerrar su ciclo productivo, han sido: el lanzamiento de consolas con resoluciones 4k con gráficos superiores HDRI, mundos abiertos gigantescos, partidas en línea masivas, efectos fotorrealistas, el aumento de juegos móviles, Virtual Reality y Augmented Reality, así como la popularidad de e-sports.

A su vez se espera que, en la novena generación de consolas, el avance tecnológico impacte en la industria, de forma que las nuevas consolas probablemente compitan con servicios de Cloud Gaming, como Google Stadia, Project xCloud (en desarrollo) y Apple Arcade, donde los juegos podrán ser jugados en cualquier dispositivo por streaming en internet, exponenciando el mercado móvil gracias a las velocidades 5g de internet.

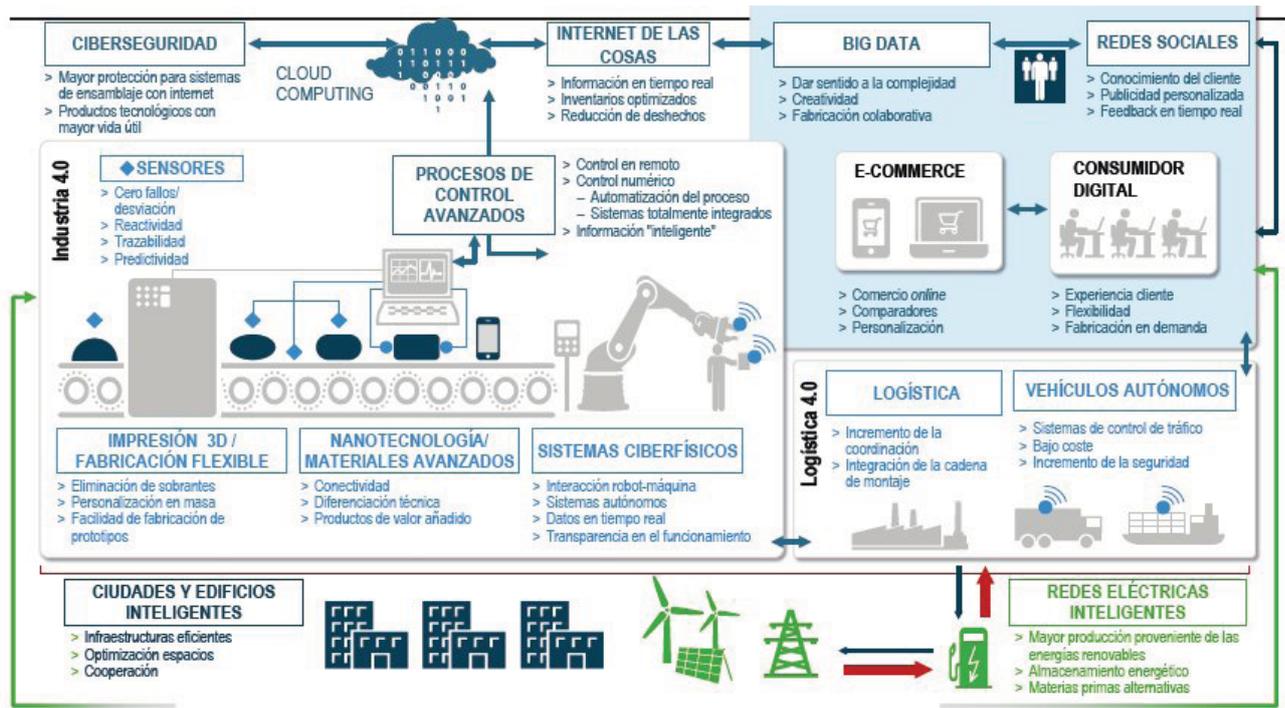
Estos cambios mencionados que se aproximan en los años siguientes en la industria del videojuego se darán en el contexto de lo que se conoce como la Cuarta Revolución Industrial, término presentado en 2016 por Klaus Martin Schwab que describe sobre cómo el mercado está cambiando por la informatización y digitalización de la producción, de forma que se generarán, integrarán y procesarán gran cantidad de datos a lo largo del proceso productivo y ciclo de vida de los productos.

Unirá lo físico, lo digital y lo biológico a través de la conectividad de diversas tecnologías: *Internet of Things (IoT)*, *Realidad Virtual (VR)*, *Realidad Aumentada (AR)*, *Cómputo Cuántico*, *Inteligencia artificial (AI)*, *Vehículos Autónomos*, *Nanotecnología*, *Big data*, *Cloud Computing*, *Ciberseguridad*, *Impresiones en 3d*,

Capítulo 1. La industria de los Videojuegos y el rol del Diseñador

1.1 Contexto Tecnológico

Manipulación Biológica en Bioingeniería Genética, Energías Renovables, Cirugía con Robots y Prótesis.



Fuente: Roland Berger

1.3 Gráfico Sobre Las Interacciones De La Cuarta Revolución Industrial.

periodistadigital.com. (2020). Gráfico Sobre Las Interacciones De La Cuarta Revolución Industrial [Imagen]. Recuperado 25 Abril 2020, de <https://www.periodistadigital.com/exitodigital/wp-content/uploads/sites/11/2017/09/infografia-digitalizacion-estudio.png>

Hay que recordar que la Primera Revolución Industrial se da en Inglaterra en el siglo XVIII con la llegada de la máquina de vapor, produciendo una sociedad que migra del campo a las ciudades para basarse en la industrialización.

La Segunda Revolución Industrial se da 100 años más tarde (1870-1914) con desarrollos técnicos como el uso de nuevas fuentes de energía como gas, petróleo y electricidad y el inicio de la producción en masa y la línea de montaje.

La Tercera Revolución Industrial iniciada en los 70's marca el principio de la era de la información, con innovaciones principalmente en electrónica y computación abarcando computadoras, microprocesadores, celulares e internet, logrando automatizar procesos industriales.

Capítulo 1. La industria de los Videojuegos y el rol del Diseñador

1.1 Contexto Tecnológico

Sin embargo, la velocidad de los avances actuales no tiene precedentes históricos ya que evoluciona de forma exponencial en lugar de lineal.



1.4 La Cuarta Revolución Industrial, Transformación Digital.

gtecnologica.com, 2020. La Cuarta Revolución Industrial, Transformación Digital. [image]. Recuperado 21 Marzo 2020, de <https://gtecnologica.com/wp-content/uploads/2019/07/La-4-revolucion-industrial-2-1024x452.png>



1.5 The Future of Jobs Report 2018

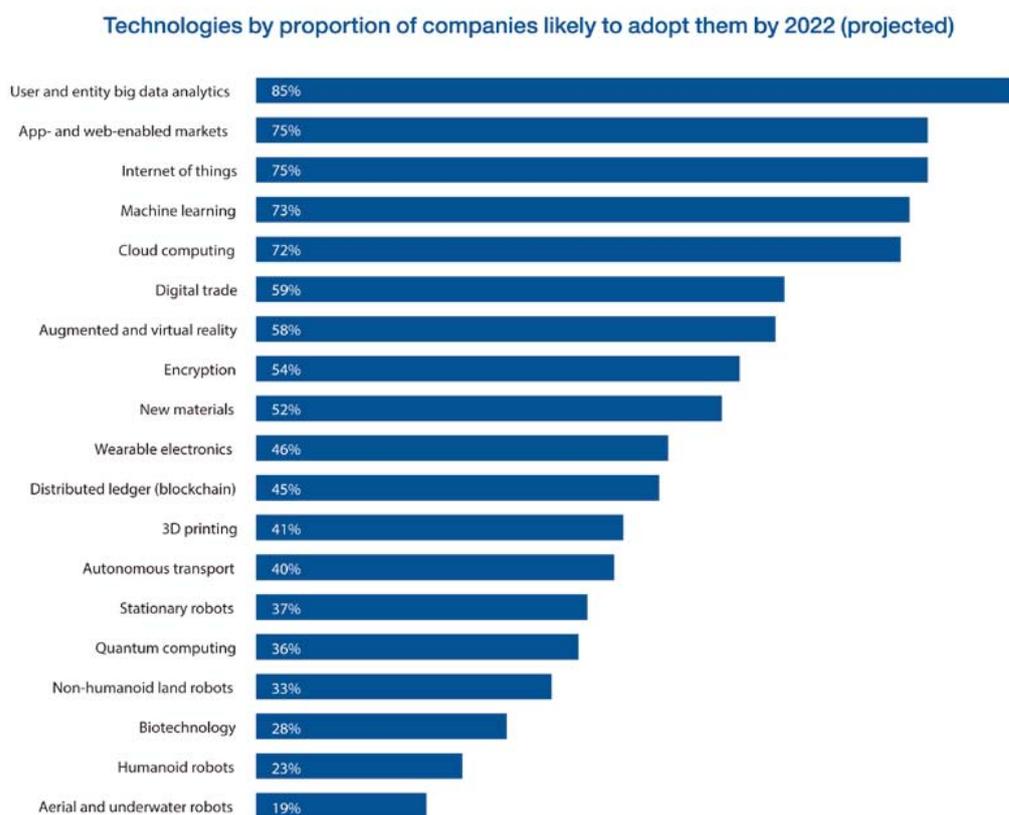
SCHWAB, K., 2018. The Future of Jobs Report 2018. [ebook] Cologny/Geneva Switzerland: World Economic Forum, p.1. Disponible en: <http://www3.weforum.org/docs/WEF_Future_of_Jobs_2018.pdf> [21 Marzo 2020].

Capítulo 1. La industria de los Videojuegos y el rol del Diseñador

1.1 Contexto Tecnológico

En el siguiente apartado se presentan gráficas que son parte del informe *The Future of Jobs Report 2018* del *World Economic Forum* que contienen información sobre las tasas de adopción de nuevas tecnologías (2018-2020), se puntualiza que a medida que los avances tecnológicos cambian rápidamente la frontera entre las tareas de trabajo realizadas por humanos y las realizadas por máquinas y algoritmos, es probable que los mercados laborales globales experimenten grandes transformaciones.

Las gráficas presentan el porcentaje de nuevas tecnologías adoptadas por compañías y por industrias, prediciendo como la Cuarta Revolución Industrial impactará creando e interrumpiendo empleos y teniendo como fin la búsqueda en mejorar la calidad del trabajo, así como la productividad de los trabajadores.



Source: Future of Jobs Survey 2018, World Economic Forum.

1.6 Se muestra la proporción en adopción de nuevas tecnologías en empresas para 2022.

SCHWAB, K., 2018. *The Future of Jobs Report 2018*. [ebook] Cologny/Geneva Switzerland: World Economic Forum, p.7. Disponible en: <http://www3.weforum.org/docs/WEF_Future_of_Jobs_2018.pdf> [Acceso 21 Marzo 2020].

Capítulo 1. La industria de los Videojuegos y el rol del Diseñador

1.1 Contexto Tecnológico

Para 2022, de acuerdo con las intenciones de inversión declaradas de las empresas encuestadas para este informe, es probable o muy probable que el 85% de los encuestados hayan ampliado su adopción en el análisis de Big data de usuarios y entidades. Del mismo modo, es probable o muy probable que grandes proporciones de compañías hayan ampliado su adopción de tecnologías como Internet de las cosas y mercados habilitados para aplicaciones y web, y para hacer un uso extensivo de la computación en la nube. Así también, el aprendizaje automático y la realidad virtual y aumentada están a punto de recibir una inversión comercial considerable.

Technology adoption by industry and share of companies surveyed, 2018–2022 (%)

	Overall	Automotive, Aerospace, Supply Chain & Transport	Aviation, Travel & Tourism	Chemistry, Advanced Materials & Biotechnology	Consumer	Energy Utilities & Technologies	Financial Services & Investors	Global Health & Healthcare	Information & Communication Technologies	Infrastructure	Mining & Metals	Oil & Gas	Professional Services
User and entity big data analytics	85	84	89	79	85	85	86	87	93	65	62	87	85
App- and web-enabled markets	75	76	95	71	88	65	89	80	93	53	50	61	74
Internet of things	75	82	95	58	73	85	65	67	86	76	50	83	74
Machine learning	73	87	79	58	82	77	73	80	91	53	69	70	74
Cloud computing	72	76	79	67	67	73	65	73	91	71	62	78	76
Digital trade	59	68	68	62	82	58	70	53	70	47	50	57	59
Augmented and virtual reality	58	71	68	50	48	65	59	67	72	59	62	65	53
Encryption	54	58	53	25	42	38	73	67	67	41	25	57	53
New materials	52	71	32	79	79	65	22	60	30	82	62	83	41
Wearable electronics	46	61	53	46	45	42	49	73	49	24	25	70	35
Distributed ledger (blockchain)	45	32	37	29	39	54	73	67	67	18	38	48	50
3D printing	41	61	21	58	42	54	19	53	35	41	50	57	29
Autonomous transport	40	74	58	54	39	46	16	20	44	41	50	30	41
Stationary robots	37	53	37	50	42	35	27	47	35	35	38	52	29
Quantum computing	36	29	32	25	33	46	43	33	44	24	19	43	41
Non-humanoid land robots	33	42	26	21	36	27	32	40	37	29	25	30	24
Biotechnology	28	18	0	42	52	42	11	87	23	12	44	39	24
Humanoid robots	23	29	26	17	18	8	35	13	33	12	25	13	24
Aerial and underwater robots	19	18	16	17	12	35	5	0	19	29	25	52	21

Source: Future of Jobs Survey 2018, World Economic Forum.

1.7 Adopción de Tecnología por industria.

SCHWAB, K., 2018. The Future of Jobs Report 2018. [ebook] Coligny/Geneva Switzerland: World Economic Forum, p.16. Disponible en: <http://www3.weforum.org/docs/WEF_Future_of_Jobs_2018.pdf> [Acceso 21 Marzo 2020].

Capítulo 1. La industria de los Videojuegos y el rol del Diseñador

1.1 Contexto Tecnológico

La tabla muestra el rango de demanda para la adopción de tecnologías específicas. La tecnología robótica se adoptará entre el 37% y el 23% de las empresas encuestadas para este informe, dependiendo la industria.

Es más probable que las empresas de todos los sectores adopten el uso de robots estacionarios, en contraste con los robots humanoides, aéreos o submarinos.

Sin embargo, los líderes de la industria del petróleo y el gas informan el mismo nivel de demanda de robots estacionarios, aéreos y subacuáticos, mientras que los empleadores en la industria de servicios financieros e inversores tienen más probabilidades de señalar la adopción planificada de robots humanoides en el período hasta 2022. Las tecnologías de contabilidad distribuida tienen un impacto particular en la industria de servicios financieros, que promete ser una de las primeras en adoptar la tecnología. De hecho, el 73% de los encuestados espera que su empresa adopte su uso. Otra industria preparada para escalar su adopción de tecnologías de contabilidad distribuida será la industria de la Salud y Salud Global. Se espera que el aprendizaje automático se adopte en una variedad de industrias, incluidas la banca y los seguros, donde puede alterar la predicción del riesgo; en el campo de la medicina, donde puede usarse para diagnóstico avanzado; en todo el sector energético, donde puede conducir a un mantenimiento predictivo; y en el sector de consumo, donde puede mejorar la capacidad de la industria para modelar la demanda.

Ante esta realidad, es necesario que México se actualice y prepare talento para poder competir globalmente en el futuro. Por un lado, se debe promover el pensamiento innovador, empezando por generar conocimiento a través del fortalecimiento de la educación y preparación en áreas relacionadas a *STEM (Science, Technology, Engineering and Mathematics)* (Ciencia, Tecnología, Ingeniería y Matemáticas) además se debe generar e incentivar todo el ecosistema de innovación y emprendimiento en el país. Se deben crear estímulos y programas para fortalecer la vinculación académica con el sector empresarial, así como aumentar la inversión en investigación, ciencia y tecnología tanto pública como privada.

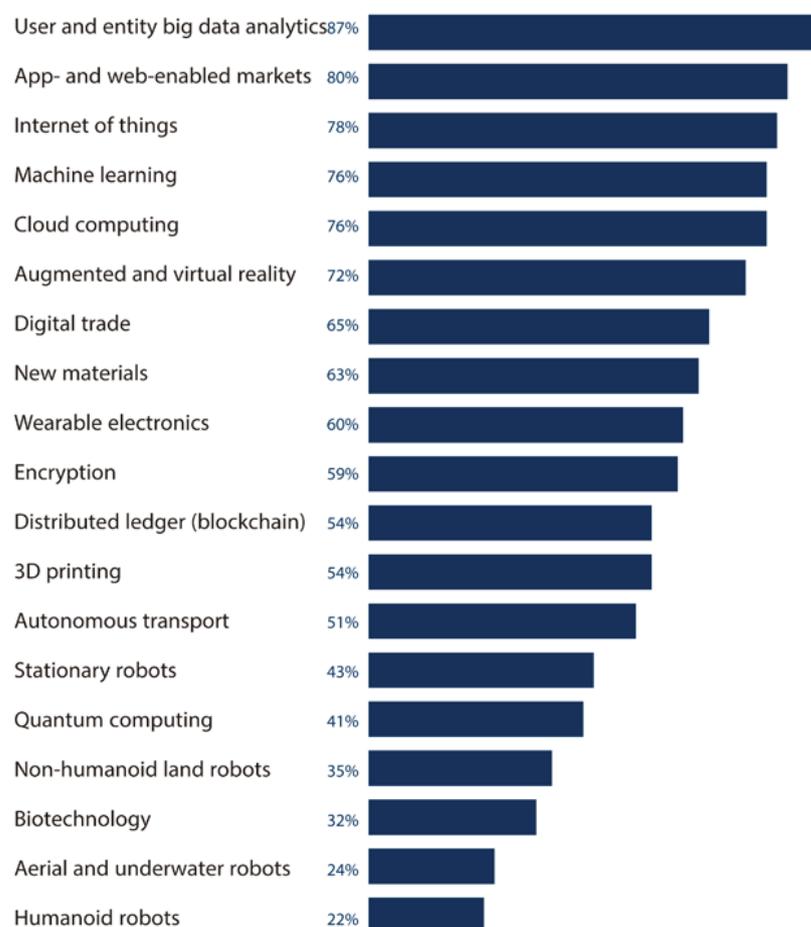
Capítulo 1. La industria de los Videojuegos y el rol del Diseñador

1.1 Contexto Tecnológico

Country Profile

México

Technology adoption (share of companies surveyed)



Source: Future of Jobs Survey 2018, World Economic Forum.

1.8 Adopción de Tecnología por compañías encuestadas para México

SCHWAB, K., 2018. The Future of Jobs Report 2018. [ebook] Cologny/Geneva Switzerland: World Economic Forum, p.88. Disponible en: <http://www3.weforum.org/docs/WEF_Future_of_Jobs_2018.pdf> [Acceso 21 Marzo 2020].

Capítulo 1. La industria de los Videojuegos y el rol del Diseñador

1.1 Contexto Tecnológico

El 87% de las empresas encuestadas realizarán la adopción en el análisis de Big data de usuarios y entidades. 80% adoptará mercados habilitados para aplicaciones y web 78% ampliará su adopción al Internet de las cosas. ¹

A medida que se desarrolla la Cuarta Revolución Industrial, las compañías buscan aprovechar las tecnologías nuevas y emergentes para alcanzar niveles más altos de eficiencia en producción y consumo, para así expandirse a nuevos mercados y competir con nuevos productos destinados a consumidores globales compuestos en su mayoría por nativos digitales. Esto significa que los empleadores también buscan trabajadores con nuevas habilidades de otros campos laborales para mantener la competitividad en sus empresas y expandir la productividad de su fuerza laboral.

La tabla presentada a continuación proporciona una descripción general de las funciones laborales que se espera experimenten un aumento en la demanda en todo el país durante el período 2018-2022. Los roles de trabajo individuales enumerados son para fines ilustrativos e informan los roles de trabajo más frecuentemente citados por los encuestados de las compañías que operan en México

1- Directores generales y directores ejecutivos	6- Representantes de ventas, venta al por mayor y fabricación, productos técnicos y científicos
2- Desarrolladores y analistas de software y aplicaciones	7- Especialistas en recursos humanos
3- Analistas de datos y científicos	8- Asesores financieros y de inversiones
4- Profesionales de ventas y marketing	9- Trabajadores de ensamblajes y fábricas
5- Gerentes Generales y de Operaciones	10- Analistas financieros

Existen opiniones que fundamentan que la automatización tendrá un impacto negativo en labores antes dedicadas a personas quitando trabajos, y este punto va más allá mencionando que existirá una división en labores por baja habilidad/baja remuneración y alta habilidad/alta remuneración.

¹ El término Internet de las Cosas se refiere a escenarios en los que la conectividad de red y la capacidad de cómputo se extienden a objetos, sensores y artículos de uso diario que habitualmente no se consideran computadoras, permitiendo que estos dispositivos generen, intercambien y consuman datos con una mínima intervención humana.

Capítulo 1. La industria de los Videojuegos y el rol del Diseñador

1.1 Contexto Tecnológico

Según Schwab, “los cambios son tan profundos que, desde la perspectiva de la historia humana, nunca ha habido un momento de mayor promesa o peligro potencial”²

Sin embargo, también es cierto que habrá una apertura en muchas más áreas de trabajo que en un principio no estaban relacionadas, impulsando el desarrollo del conocimiento, además la tecnología está en un punto que facilita procesos, y en el caso de los videojuegos se ve reflejado en la posibilidad de producirlos de forma más económica por medio de herramientas fáciles de usar y en ocasiones gratuitas.

Es importante mencionar que el principal fundamento que todas estas áreas tecnológicas presentan es el aprovechamiento de recursos por medio del desarrollo de software a través de la programación.

² Schwab, K. (2016). La cuarta revolución industrial (p. 9). DEBATE.

Capítulo 1. La industria de los Videojuegos y el rol del Diseñador

1.2 La Industria Global y Nacional de Videojuegos

1.2 La Industria Global y Nacional de Videojuegos

La industria de los videojuegos es un sector económico moderno muy rentable que crece exponencialmente y se basa en la inversión, producción, promoción y monetización de videojuegos y mercancías relacionadas a estos.

En la actualidad es el área de entretenimiento con mayores ventas globales superando a las industrias del cine, música y streaming siendo el conglomerado chino Tencent, la empresa de videojuegos con mayor facturación del mundo, por delante de gigantes como Sony, Microsoft o Nintendo.



1.9 Este gráfico muestra los ingresos en bruto de la industria del entretenimiento en los Estados Unidos en 2018, por segmento.

Richter, F. (2020). Infographic: Are You Not Entertained? Recuperado 3 Enero 2020, de <https://www.statista.com/chart/17394/entertainment-industry-revenue/>

Capítulo 1. La industria de los Videojuegos y el rol del Diseñador

1.2 La Industria Global y Nacional de Videojuegos

Este crecimiento por parte de los videojuegos se ha dado por muchos factores entre ellos, a experiencias interactivas de gran calidad con historias inmersivas, gráficos sorprendentes, mecánicas de juego tanto accesibles como divertidas, experiencias *multiplayer online* y portabilidad móvil, que compiten de forma favorable contra medios de comunicación estáticos como el cine o la tv, ya que un videojuego es la unión de imágenes, audio, texto y lo más importante interactividad por medio de juegos, presentando realidades alternas donde el jugador puede ser inmerso en contraste con la realidad diaria.

La industria de los videojuegos empezó como tal en los años 70's con los primeros videojuegos Arcades, tras su éxito se dio paso a la producción de consolas para el hogar siendo la *Magnavox Odyssey (1972)* la primera en ese ámbito. sin embargo también es cierto que el mismo mercado ha provocado crisis que por momentos se pensó que acabarían con la producción de videojuegos y la misma industria como lo constatan el *videogame crash* de 1977 donde la manufactura de consolas viejas y obsoletas y la piratería por medio de clones, inundó el mercado creando un exceso de consolas, provocando el cierre de compañías; y la crisis del videojuego de 1983 mucho mayor que la de 77 donde el mercado se sobresaturó de juegos de muy mala calidad al borde de la bancarrota y el colapso corporativo, a pesar de esto ante todo, la industria ha sabido reponerse adaptarse y salir adelante como lo constata en 1983 el sello de calidad en la producción de juegos para *NES* que protegía al comprador con juegos excelentes y que dio una inyección de credibilidad a la industria de videojuegos y qué decir de la consola *Playstation 2* con 155 millones de unidades vendidas siendo la mayor consola en tener tantas ventas en la historia mostrando lo consagrado que es este mercado de consumo global.

El triunfo en ventas se ha dado tras comprobar el éxito de algunos juegos sobresalientes generación tras generación por lo que las compañías han sabido aprovechar este factor generado franquicias muy poderosas en ventas donde presentan a personajes y entornos que el público adopta como propios y los hace sus favoritos volviéndolos iconos culturales, llegando al grado de asociar a estos personajes con la marca productora, como es el caso de *Mario Bros* de *Nintendo*, *Sonic The Hedgehog* de *SEGA*, *Master Chief (HALO)* para *Xbox* y *Crash Bandicoot* para *Playstation*.

Otro exitoso modelo de la industria es el *Free-to-Play* donde el jugador descarga un juego gratuitamente y puede jugarlo así, aunque el negocio radica en la compra de mejoras al juego por microtransacciones, un ejemplo de ello en la actualidad es el popular título de acción multijugador *Fortnite*, que obtuvo unos ingresos récord de 2.200 millones de euros en el 2018. Ningún otro juego ha generado tanto en un solo año.

También por otra parte la industria actual ha incorporado otro modelo de producción

Capítulo 1. La industria de los Videojuegos y el rol del Diseñador

1.2 La Industria Global y Nacional de Videojuegos

que le ha resultado redituable llamado producciones Indie, denominado así porque son videojuegos producidos por estudios independientes conformados por unos cuantos desarrolladores que cubren varios puestos de trabajo en comparación con cientos de trabajadores de las grandes compañías, y que en la práctica aún al tener mucho menor presupuesto que un gran corporativo, pueden lograr producciones de calidad notable que al combinarse con promoción adecuada logran ventas de millones volviéndose éxitos globales como lo demuestran títulos como *Minecraft*, *Fez*, *Super Meat Boy*, *Journey* y *Cuphead*.

Por tanto, la industria además de innovar tecnológicamente de forma constante está al acecho de nuevos talentos que les permitan exponenciar sus ventas.

A continuación, se muestran gráficos descriptivos del mercado global de Videojuegos³

Por primera vez desde 2015, Estados Unidos será el mercado de juegos más grande por ingresos a nivel mundial con \$ 36.9 mil millones este año. Impulsado por el crecimiento en los ingresos de los juegos de consola, superará a China para la posición # 1.

Para 2022, el mercado mundial de juegos crecerá a \$ 196.0 mil millones con un TCAC - Tasa de crecimiento anual compuesto (2018-2022) de + 9.0%. Debido a la congelación de licencias que afectó fuertemente a China, Asia-Pacífico ya no es la región de más rápido crecimiento. Impulsado por la mejora de la infraestructura y un mayor apetito por los juegos y los deportes electrónicos, América Latina es ahora el mercado de juegos de más rápido crecimiento en el mundo (basado en las cuatro regiones ilustradas a la derecha), con una TCAC de + 10.4%.

Los juegos móviles (teléfonos inteligentes y tabletas) siguen siendo el segmento más grande en 2019, produciendo ingresos de \$ 68.5 mil millones, el 45% del mercado mundial de juegos. De todos los ingresos de juegos móviles, el 80%, o \$ 54.9 mil millones, provendrán de juegos de para smartphone, y los juegos para Tablet representan los \$ 13.6 mil millones restantes.

Los juegos móviles (Smartphones y Tablet) generarán ingresos de \$ 95.4 mil millones en 2022, creciendo con un TCAC de + 11.3% para representar casi la mitad (49%) de todo el mercado de juegos. Los ingresos y el crecimiento serán impulsados principalmente por los teléfonos inteligentes, con ingresos de \$ 79.7 mil millones para 2022 (TCAC + 12.8%). Las Tablets representan los \$ 15.7 mil millones restantes. Los mercados emergentes contribuirán más al crecimiento del segmento. Sin embargo, también contribuirá una variedad de otros factores, incluidos más títulos multiplataforma, más usuarios de teléfonos inteligentes y mejoras en hardware e infraestructura.

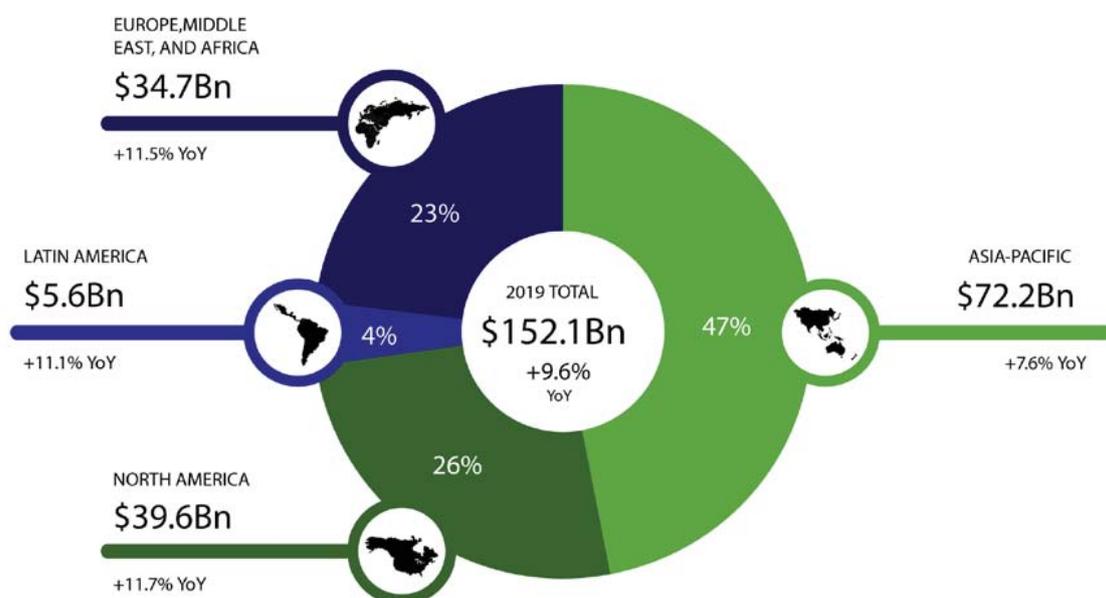
3 Newzoo Global Games Market Report 2019 | Light Version | Newzoo. (2020). Recuperado 3 Enero 2020, de <https://newzoo.com/insights/trend-reports/newzoo-global-games-market-report-2019-light-version/>

Capítulo 1. La industria de los Videojuegos y el rol del Diseñador

1.2 La Industria Global y Nacional de Videojuegos

Los ingresos generados por el segmento de la consola alcanzarán los \$ 61.1 mil millones en 2022, aumentando con una TCAC saludable de + 9.7%. Los ingresos se verán impulsados tanto por el lanzamiento inminente de la próxima generación de consolas Xbox y PlayStation, los próximos modelos nuevos de Switch, como por la base de la generación de consolas anterior (ahora actual).

2019 GLOBAL GAMES MARKET PER REGION



1.10 Mercado Global de Videojuegos por Región

Peter Warman. (2019). Global Games Market Report 2019 [Ebook] (1st ed., p. 13). Recuperado 30 Abril 2020, de <https://newzoo.com/insights/trend-reports/newzoo-global-games-market-report-2019-light-version/>

Capítulo 1. La industria de los Videojuegos y el rol del Diseñador

1.2 La Industria Global y Nacional de Videojuegos

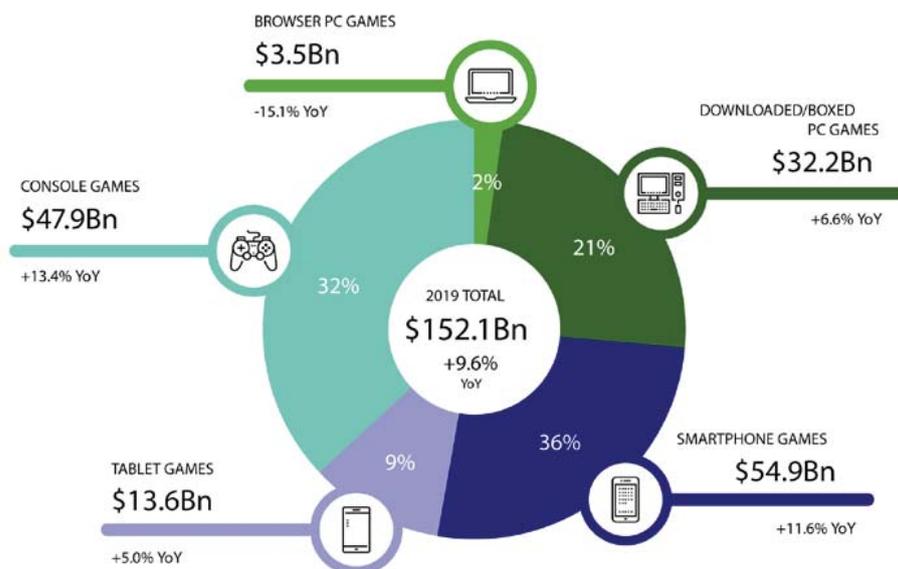
REGIONAL BREAKDOWN
OF GLOBAL GAME REVENUES
TOWARD 2022



1.11 Desglose regional de ganancia global

Peter Warman. (2019). Global Games Market Report 2019 [Ebook] (1st ed., p. 14). Recuperado 30 Abril 2020, from <https://newzoo.com/insights/trend-reports/newzoo-global-games-market-report-2019-light-version/>

2019 GLOBAL GAMES MARKET
PER SEGMENT



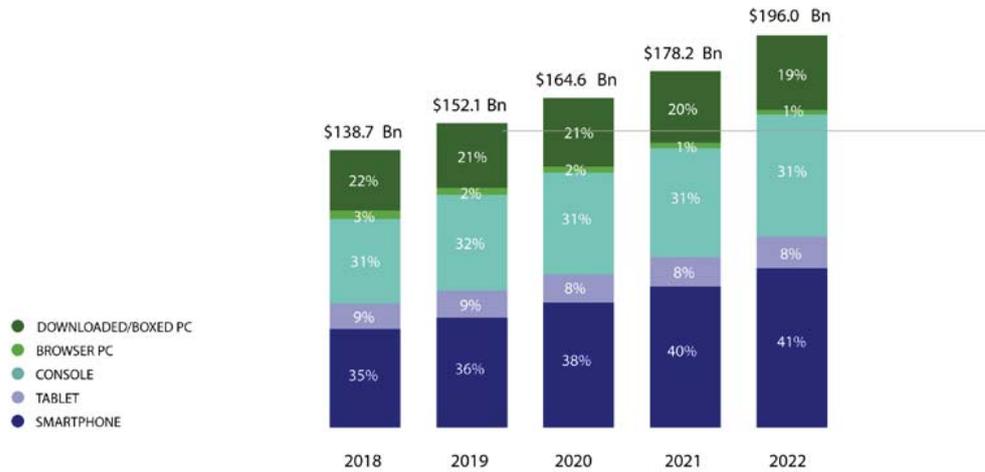
1.12 Mercado Global de Videojuegos por Segmento

Peter Warman. (2019). Global Games Market Report 2019 [Ebook] (1st ed., p. 14). Recuperado 30 April 2020, from <https://newzoo.com/insights/trend-reports/newzoo-global-games-market-report-2019-light-version/>

Capítulo 1. La industria de los Videojuegos y el rol del Diseñador

1.2 La Industria Global y Nacional de Videojuegos

SEGMENT BREAKDOWN
OF GLOBAL GAME REVENUES
TOWARD 2022



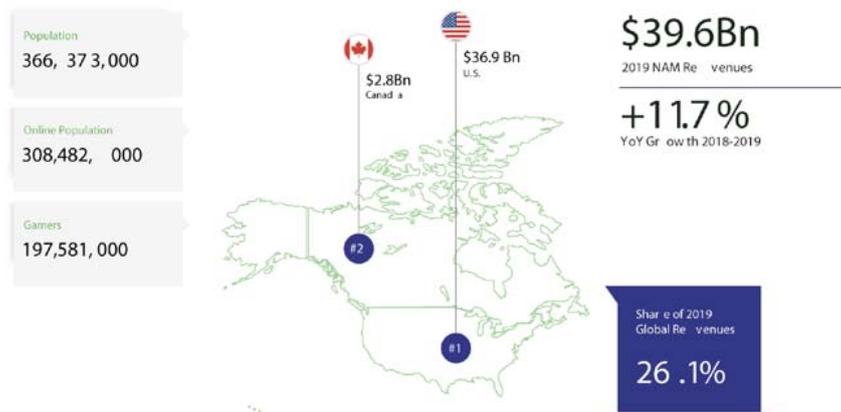
1.13 Desglose Global de Ganancias hacia 2022

Peter Warman. (2019). Global Games Market Report 2019 [Ebook] (1st ed., p. 15). Recuperado 30 Abril 2020, de <https://newzoo.com/insights/trend-reports/newzoo-global-games-market-report-2019-light-version/>

NORTH AMERICA

2019 GAME REVENUES

TOP COUNTRIES BASED ON GAME REVENUES



1.14 Ganancias de la Industria del Videojuego en Norte América 2019

Peter Warman. (2019). Global Games Market Report 2019 [Ebook] (1st ed., p. 29). Recuperado 30 Abril 2020, de <https://newzoo.com/insights/trend-reports/newzoo-global-games-market-report-2019-light-version/>

Capítulo 1. La industria de los Videojuegos y el rol del Diseñador

1.2 La Industria Global y Nacional de Videojuegos

LATIN AMERICA

2019 GAME REVENUES

TOP COUNTRIES BASED ON GAME REVENUES



1.15 Ganancias de la Industria del Videojuego en América Latina 2019

Peter Warman. (2019). Global Games Market Report 2019 [Ebook] (1st ed., p. 29). Retrieved 30 April 2020, from <https://newzoo.com/insights/trend-reports/newzoo-global-games-market-report-2019-light-version/>

Entendiendo este contexto y siendo México el principal consumidor de videojuegos en América Latina, es que se abre una oportunidad para generar en primer lugar contenidos regionales donde existe un gran campo posible de desarrollo (el juego Mulaka es un ejemplo de ello) y en segundo lugar tener un grado estandarizado de producción donde los videojuegos alcancen mercados globales y se puedan producir juegos de grandes compañías en México a menor costo sin sacrificar calidad, empezando a ser un factor económico activo dentro de la industria del videojuego internacional.

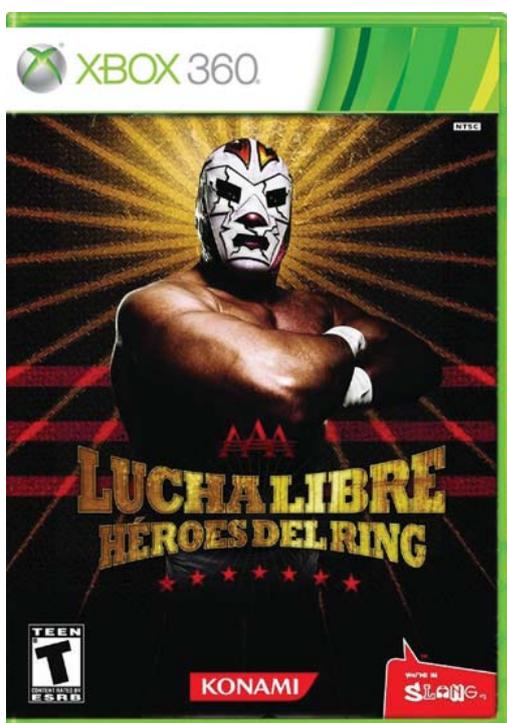
Es de esta manera que desde hace unos años el país ha empezado a producir títulos para diferentes plataformas; conocimiento del sector, emprendimiento y accesibilidad tecnología han provocado el surgimiento de estudios desarrolladores de videojuegos mexicanos, logrando iniciar una industria que poco a poco va cobrando más fuerza y que va ganándose a los inversores gracias a la estandarización de calidad con la que cumplen los juegos publicados en las principales consolas (Switch, ps4 y Xbox One) smartphones y PC.

Capítulo 1. La industria de los Videojuegos y el rol del Diseñador

1.2 La Industria Global y Nacional de Videojuegos

El inicio del desarrollo formal de videojuegos en México se da con la publicación al mercado del juego Lucha Libre AAA: Héroes del Ring, un juego divertido con jugabilidad sencilla y con gráficos estándares gracias al motor *Unreal Engine* siendo bien recibido por los fans a falta de productos con temáticas nacionales.

A continuación, se citan los juegos realizados por estudios mexicanos desde 2010.



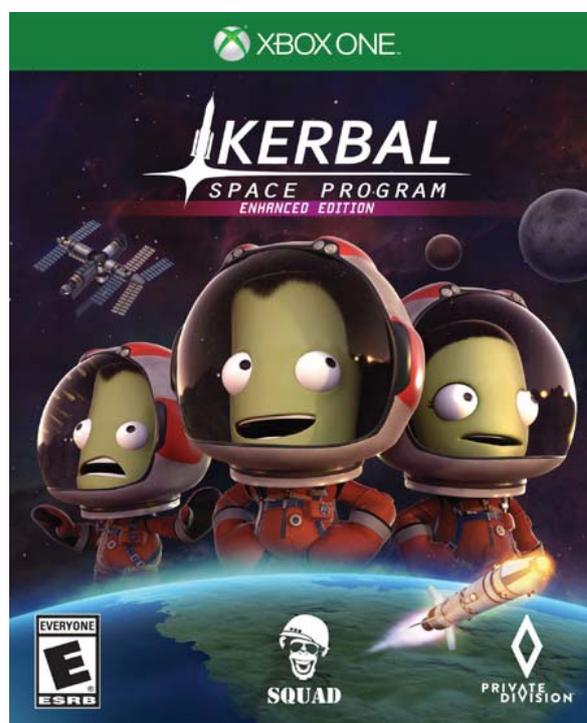
1.16 Lucha Libre AAA: Héroes del Ring

Game Planet, (2020). Lucha Libre AAA: Héroes Del Ring / Xbox 360. [imagen] Recuperado de https://gameplanet-53f8.kxcdn.com/media/catalog/product/cache/4/image/9df78eab33525d08d6e5fb8d27136e95/a/a/aaa-heroes-del-ring-360_3.jpg

Estudio: Larva Game Studios

Nombre del juego: Lucha Libre AAA: Héroes del Ring (2010)

Plataformas: Ps3, Xbox 360, Wii, PSP Portable, Nintendo DS



1.17 Kerbal Space Program

amazon.com, (2020). Kerbal Space Program. [imagen] Recuperado de <https://images-na.ssl-images-amazon.com/images/I/71Org4IrHwL.jpg>

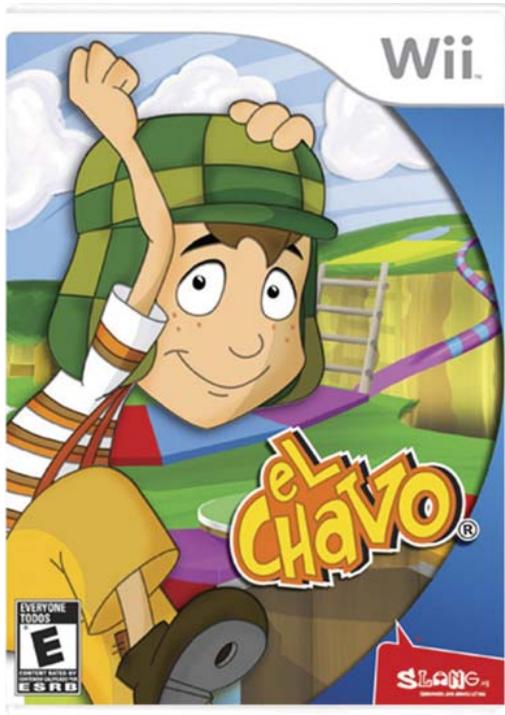
Estudio: Squad

Nombre del juego: Kerbal Space Program (2011)

Plataformas: PC, OSX, Linux, Ps4, Xbox One

Capítulo 1. La industria de los Videojuegos y el rol del Diseñador

1.2 La Industria Global y Nacional de Videojuegos



1.18 El Chavo

gameplanet.com. (2020). El Chavo / Wii [imagen]. Recuperado de <https://gameplanet-53f8.kxcdn.com/media/catalog/product/cache/4/imagen/9df78eab33525d08d6e5fb8d27136e95/e/l/el-chavo.jpg>

Estudio: Kaxan Media Group
Nombre del juego: El Chavo (2012)
Plataformas: Wii, Android, iOS, Windows Phone



1.19 Elliot Quest

Twitter | @hardcopygamers. (2020). Elliot Quest [imagen]. Recuperado de <https://pbs.twimg.com/media/D6XPr8oU0AEDFop.jpg>

Estudio: Ansimuz Games
Nombre del juego: Elliot Quest (2014)
Plataformas: Ps4, Switch, Wii U, Xbox One, PC, Linux, OSX, Ouya

Capítulo 1. La industria de los Videojuegos y el rol del Diseñador

1.2 La Industria Global y Nacional de Videojuegos



1.20 Viva Sancho Villa

2Dnutz. (2020). Viva Sancho Villa / ios, Android [imagen]. Recuperado de https://scontent.fmex7-2.fna.fbcdn.net/v/t31.0-8/p960x960/21728929_1646468102093879_7166558691277988110_o.jpg?_nc_cat=107&_nc_sid=730e14&_nc_ohc=LY-6yEylWeC8AX_e2_0H&_nc_ht=scontent.fmex7-2.fna&_nc_tp=6&oh=72408b3b19624518067a4efa00dba2cd&oe=5E9BBB58

Estudio: 2Dnutz

Nombre del juego: Viva Sancho Villa (2016)

Plataforma: Android



1.21 MilitAnt

Xibalba Studios. (2020). MilitAnt [imagen]. Recuperado de <https://i.ytimg.com/vi/ojJpXCtYvqw/maxresdefault.jpg>

Estudio: Xibalba Studios

Nombre del juego: MilitAnt (2016)

Plataformas: Ps4, PC



1.22 Attractio

GameCoder Studios. (2020). Attractio [imagen]. Recuperado de https://media.vandal.net/m/36622/attractio-2016217113427_10.jpg

Estudio: GameCoder Studios

Nombre del juego: Attractio (2016)

Plataformas: Ps4, Ps Vita, PC, OSX, Linux

Capítulo 1. La industria de los Videojuegos y el rol del Diseñador

1.2 La Industria Global y Nacional de Videojuegos



1.23 Pato Box

play-asia.com. (2020). <https://s.pacn.ws/1500/wk/pato-box-586477.19.jpg?pnkjgg> [imagen]. Recuperado de <http://play-asia.com>

Estudio: Bromio

Nombre del juego: Pato Box (2018)

Plataformas: Ps4, Switch, PC, OSX, PS Vita



1.24 Mulaka

nintendosoup.com. (2020). Mulaka / Switch [imagen]. Recuperado de <https://i.imgur.com/rg9Lv7v.png>

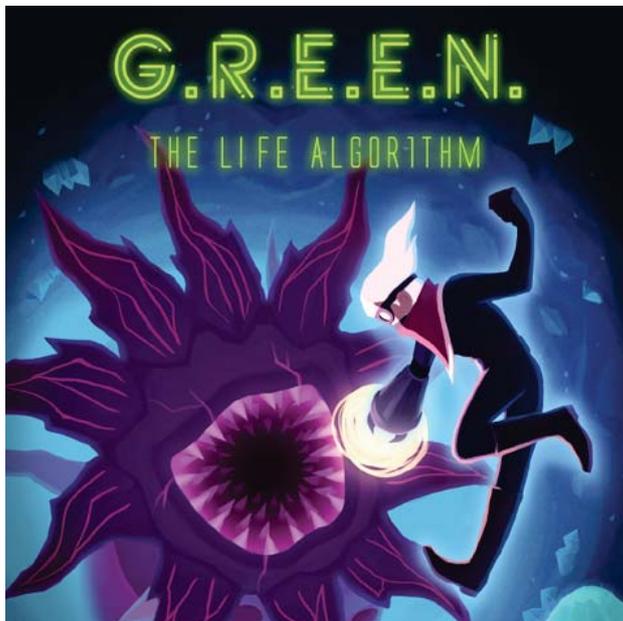
Estudio: Lienzo

Nombre del juego: Mulaka (2018)

Plataformas: Ps4, Switch, Xbox One, PC, OSX

Capítulo 1. La industria de los Videojuegos y el rol del Diseñador

1.2 La Industria Global y Nacional de Videojuegos



1.25 GREEN The Life Algorithm

Nintendo. (2020). GREEN The Life Algorithm [imagen]. Recuperado de https://cdn02.nintendo-europe.com/media/imagenes/11_square_imagens/games_18/nintendo_switch_download_software/SQ_NSwitchDS_GreenVideoGame.jpg

Estudio: Estación PI

Nombre del juego: GREEN The Life Algorithm (2019)

Plataformas: Switch, Ps4, Xbox One, PC



1.26 KleptoCorns

hyperbeard.com. (2020). KleptoCorns [imagen]. Recuperado de https://hyperbeardgamessite.s3.amazonaws.com/media/imagenes/Screen_Shot_2019-10-17_at_14.13.4720191017193517.png

Estudio: Hyperbeard

Nombre del juego: KleptoCorns (2019)

Plataformas: iOS, Android

Capítulo 1. La industria de los Videojuegos y el rol del Diseñador

1.3 El diseñador y el desarrollo de videojuegos

1.3 El diseñador y el desarrollo de videojuegos

El proceso de Diseño puede ser visto como la estrategia a seguir donde la creatividad y habilidad utilizan arte y ciencia para generar soluciones a problemas.

La labor del diseñador visual se basa en resolver problemas de comunicación visual de forma efectiva en tiempo y presupuesto, este proceso de comunicación involucra un emisor que tiene la intención de comunicar una idea con ciertos fines, para esto recurre a un diseñador visual que toma en cuenta el contexto y al usuario final al que va dirigido el mensaje y basándose en ellos, diseña mensajes de forma visual, utilizando como soporte texto, hipermedia, imagen fija e imagen en movimiento.

Para lograr impacto, atención y efectividad en la comunicación visual del mensaje, el diseñador utiliza de forma práctica elementos formales de diseño como línea, punto, forma, figura-fondo, color, espacio, tamaño, por mencionar algunos y sobre estos elementos aplica principios de diseño como énfasis, balance, ritmo, simetría, contraste, repetición, gradación, y abstracción.

La relación del diseñador con los videojuegos ha variado con el tiempo.

En los inicios de los videojuegos Arcade en los 70's, hasta la generación de *Atari* en los primeros años de los 80's tanto la programación como los gráficos que daban vida a los juegos eran realizados por programadores, ya que eran procesos sumamente complejos y técnicos así que los diseñadores gráficos, publicistas o lustradores eran los encargados de crear las imágenes destinadas a dar soporte y presentación al juego para su identificación tanto en cartucho como en promoción de medios impresos y televisión.

De estas primeras etapas surgieron iconos que marcaron la memoria de la industria como el arcade *Pac - Man*, *Space Invaders* para *Atari*, *Super Mario Bros* para *NES*.



1.27 Atari 2600 Cartridges

guevoonadas.com. (2020). Atari 2600 Cartridges [imagen]. Recuperado 4 Mayo 2020, de <https://www.guevoonadas.com/wp-content/uploads/2019/04/cartuchos-de-atari-2600.png>.

Capítulo 1. La industria de los Videojuegos y el rol del Diseñador

1.3 El diseñador y el desarrollo de videojuegos



1.28 Impreso Promocional de Atari
Lapetino, T. (2016). Art of Atari (p. 16). Dynamite Entertainment.

Conforme evolucionó la industria y el poder de procesamiento de las consolas aumentó, los sistemas pudieron soportar imágenes más detalladas con mejor resolución y mayor profundidad de bits en colores y fue entonces que los creativos de diferentes áreas gráficas pudieron aportar dentro de los juegos sus ideas, esto se dio a principios de los 80's con la creación de gráficos básicos píxel a píxel para los juegos 2d en consolas de 8 bit, para posteriormente expandirse a desarrollos gráficos más avanzados logrado por pequeños grupos de artistas, dados por medio del incremento en un mayor número de colores desplegados en pantalla gracias a los 16 bit de la nueva generación de consolas, de ahí se llegó a las plataformas de 3d de 32 bit y 64 bit, que involucraban una nueva manera de especialización para la creación de gráficos por la compleja labor en el uso del nuevo software para generar gráficos 3d y hasta nuestro días donde los gráficos son sorprendentes fotorrealistas y cada área de arte está compuesta por grandes equipos de producción.

Actualmente la solvencia del diseñador con los videojuegos está presente de forma estandarizada en la creación de elementos como diseño de Interfaz para menús, barras de vidas y puntuación, branding del juego en diferentes formatos, identidad y manual corporativo como logotipos del estudio o del nombre del juego, diseño de sitio web para referencia del juego, Animación promocional del juego en 2d o 3d, diseño de empaque para la distribución física del juego, ilustración para concept art de personajes o fondos y Storyboards destinados a cinemáticas dentro del juego, diseño editorial para publicaciones impresas, por mencionar algunas áreas.

Capítulo 1. La industria de los Videojuegos y el rol del Diseñador

1.3 El diseñador y el desarrollo de videojuegos

Sin embargo el esquema dentro de la industria de los videojuegos donde el diseñador solo es generador de gráficos está cambiando poco a poco para dar paso a una forma donde el diseñador pueda crear por sí mismo prototipos funcionales o hasta videojuegos completos, siendo más independiente y teniendo más control sobre el proceso creativo, además surge la oportunidad en donde su desarrollo pueda ser monetizable de una forma real, por medio de publicidad dentro del juego, *crowdfunding-patreon* o como propuesta de prototipo destinada a estudios profesionales.

Estas nuevas facilidades en las formas de creación de juegos se dan por el uso de *Game Engines* que integran programación visual para generar interacción de forma fácil y amigable, hay que mencionar que la interactividad es la base fundamental de los videojuegos, y en entornos profesionales, un ingeniero programador en software es el encargado de darle interactividad al videojuego por medio de código de computadora.

Es necesario mencionar que los *Game Engines* de programación visual son fáciles de entender y ejecutar porque tienen acciones preestablecidas que el desarrollador puede ejercer sobre los elementos del juego, por ejemplo en el caso de un personaje podrá saltar, correr, moverse de un lado a otro, reiniciar el nivel, disparar proyectiles, llevar su puntuación entre otros, sin escribir una sola línea de código, por lo que estas acciones facilitan al desarrollador la creación de interacción en los mundos virtuales.

La programación de videojuegos involucra conocimientos en las áreas de *STEM*, término en inglés para referirse a *Science, Technology, Engineering and Mathematics* y son conocimientos que requieren una comprensión y facilidad de asimilación que sólo con dedicación y estudio se logra, sin embargo existen individuos que no tienen ese conocimiento o que no comparten esa facilidad de asimilación, por lo que recurrir a programación visual en un Game Engine es una alternativa real muy práctica para quitar barreras técnicas y empezar a crear videojuegos.

Teniendo en cuenta estas herramientas que facilitan la producción de videojuegos aunado al *background* de conocimientos en producción de gráficos por parte del diseñador, se plantea la generación de videojuegos con una perspectiva integradora, de esta forma se materializan ideas llenas de enfoques diferentes y aprovechando de una manera más efectiva y eficiente un presupuesto al conformar la interacción del producto en el rol del diseñador, reduciendo o redistribuyendo los costos para su uso en otras áreas de producción, en un mercado que está atento al consumo de productos novedosos.

También, es importante mencionar que, aunque existe un alza en el desarrollo de

Capítulo 1. La industria de los Videojuegos y el rol del Diseñador

1.3 El diseñador y el desarrollo de videojuegos

videojuegos en México, el mercado laboral es aún reducido en comparación con los grandes productores como US, Japón y UK, por esta razón los talentos mexicanos salen a otros países en búsqueda de oportunidades, además, es común encontrar en México ofertas de trabajo donde se busquen perfiles de diseñador - programador para abaratar costos.

Bajo este contexto es que se generará para fines prácticos de la investigación, la documentación y elaboración de un videojuego con programación visual, logrando también así la comprensión de conceptos básicos de programación; se tomará como base la formación que un diseñador y comunicador visual en orientación audiovisual e hipermedia tiene, donde domina con facilidad software de diseño, pero carece de educación formal en STEM.

Con esto el diseñador podrá concretar y materializar ideas sobre desarrollo creativo de videojuegos en un entorno accesible y por otra parte podrá conocer y comprender las bases de una rama que está cada vez más presente en la sociedad, es decir la programación de software, de esta forma tendrá un abanico de posibilidades para expandirse y especializarse en medios interactivos.

t

Los beneficios que obtendría la industria con el desarrollo de un entorno productivo en México serían varios y significaría desde la creación de comunidades de estudiantes hasta conferencias, eventos, festivales y ferias dirigidos a profesionales como el E3 de los Ángeles así como también la adopción de normas estandarizadas de desarrollo ISO logrando las bases para producción y generando conocimiento propio siendo el área de la investigación en medios interactivos y videojuegos un factor esencial para la evolución del mercado y de la teoría de los mismos; de esta manera el público objetivo que al hacer uso de un videojuego no solo obtendría beneficios en materia de coordinación motriz, relación ojo–mano y amplitud en percepción espacial, ganaría también en contenidos con estilos propios, generados de acuerdo al entorno regional que se encuentre, significando mayor diversión, entretenimiento más amplio y oportunidades de segmentación de mercado con rangos de error menores.

Capítulo 1. La industria de los Videojuegos y el rol del Diseñador

1.4 Los Gráficos en los videojuegos

1.4 Los Gráficos en los videojuegos

Los gráficos en los videojuegos son representaciones electrónicas en forma de imágenes, que son proyectadas en una pantalla y se dan por el cálculo de operaciones avanzadas por parte del *hardware*, es decir de la plataforma de juego.

Los gráficos son la forma en la cual el juego llega al jugador por primera vez, es la referencia principal del juego, el revestimiento, a través de los gráficos se crea una conexión donde el jugador puede identificarse y crear una afinidad, además es en esa rápida conexión que interpreta si es parecido a alguno conocido antes o es completamente nuevo y mediante ello decide si continúa prestando atención.

En la actualidad los gráficos y la jugabilidad son elementos que van de la mano siendo los más importantes para que un videojuego pueda ser considerado como bueno, divertido y exitoso.

La Historia de los gráficos de videojuegos, ha estado supeditada desde sus inicios por las limitaciones tecnológicas de cada época por lo que los artistas han tenido que adaptar sus ideas para crear experiencias que toquen los límites del *hardware* donde desarrollan, es un hecho que gráficos y tecnología van de la mano.

En la actualidad gracias al avance tecnológico de *hardware* más potente a menores costos, ha permitido que los desarrolladores utilicen software que les da mayor libertad y flexibilidad para materializar sus creaciones artísticas tomando rumbos nuevos.

Profundizando más en la producción de elementos de un videojuego, un término que está relacionado con el entorno de trabajo de un diseñador visual es el de *Game Art*, término vigente usado en la industria de los videojuegos para definir las áreas encargadas de la producción digital de elementos visuales dentro del juego, combinando habilidades artísticas y técnicas.

La diferencia con los diseñadores se basa en que el *Game Artist* tiene una formación encaminada al desarrollo artístico digital, mientras que el Diseñador resuelve procesos de comunicación.

Capítulo 1. La industria de los Videojuegos y el rol del Diseñador

1.4 Los Gráficos en los videojuegos

Los principales roles del departamento de *Game Art* en el desarrollo de un videojuego son:

Director de Arte / <i>Art Director</i> Jefe de Arte / <i>Lead Artist</i> Artista Conceptual / <i>Concept Artist</i> Artista 3D / <i>3D Artist</i> Artista 2D / <i>2D Artist</i> Artista de Entornos / <i>Environment Artist</i> Animador 3D / <i>3D Animator</i> Animador 2D / <i>2D Animator</i> <i>Storyboard Artist</i>	Artista de Efectos / <i>FX Artist</i> Artista Técnico / <i>Technical Artist</i> Artista de Interfaz / <i>UI Artist</i> Artista de Texturas / <i>Texture Artist</i> Iluminador / <i>Lighting Artist</i> <i>Rigger Artist</i> <i>Video Editor & Packaging Artist</i> <i>Motion Graphics Artist</i>
--	---

Esta segmentación del trabajo se da en los grandes estudios, pero es común que, en estudios pequeños o Indies, una persona pueda asumir varios de estos roles; por lo tanto, el diseñador tiene la capacidad de entrar en el área de *Game art* y a su vez utilizar programación visual creando un binomio ganador donde los costos son reducidos y con esto tener la posibilidad de producir y monetizar un videojuego.

Teniendo producido el Arte y la Interacción (*Gameplay*) en un videojuego se tendrá cubierta la parte esencial para que un juego funcione, sin embargo, existen más áreas que son importantes como las mecánicas de juego y la historia entre otras que ameritan atención e investigación por parte del desarrollador.

Capítulo 1. La industria de los Videojuegos y el rol del Diseñador

1.5 Estilos gráficos de videojuegos

1.5 Estilos gráficos de videojuegos

Cuando los elementos de juego dentro de un videojuego obtienen logros visuales sobresalientes y destacan por su novedad o por el impacto de su grado estético dentro de la industria y además es reconocido por el público consumidor y queda en la memoria colectiva, tomándolo como referencia para futuros juegos, se dice que ha aportado un estilo gráfico.

Además, los estilos gráficos en los videojuegos son creados para ser utilizados en soportes 2D y 3D.

Los gráficos en 2 dimensiones (2D) son rasterizados, que son la unión de píxeles para formar una imagen que responde a los planos X y Y.

Los gráficos tridimensionales (3D) poligonales, son el conjunto de vértices en el espacio unidos por bordes formando polígonos, y al unir varios polígonos se crea un objeto geométrico, con capacidad de desplazamiento en los ejes X, Y y Z, teniendo volumen y profundidad.

A través de la historia de los videojuegos han existido diferentes métodos de visualizar los contenidos del juego dependiendo de la etapa tecnológica en que se encuentren, pero en síntesis los principales estilos gráficos se categorizan en Abstractos, Estilizados y Realistas.

- **Abstracto**

Este estilo es representado principalmente por el uso de formas y elementos geométricos en lugar de presentar descripciones literales de objetos reales.

- **Estilizado**

Siendo el estilo más flexible al estar entre abstracto y realista, toma elementos propios del objeto y puede exagerar, endurecer, deformar sus proporciones.

- **Realista**

El estilo realista emula los elementos del juego asemejándolos lo más posible a la realidad, creando una gran experiencia inmersiva, es el estilo más usado en la actualidad.

Existen un sinnúmero de estilos dentro de estas 3 categorías, pero a continuación se citan los principales estilos gráficos en videojuegos.

Se consideran estilos gráficos porque representan conceptos por medio de estrategias visuales que ubican al jugador en espacio tiempo, describen características de

Capítulo 1. La industria de los Videojuegos y el rol del Diseñador

1.5 Estilos gráficos de videojuegos

jugabilidad presentando información sobre qué tipo de juego es y de que se trata, los gráficos generan sensaciones y emociones en el jugador proporcionando un placer estético, generan también una empatía con el jugador, además los gráficos forman parte de la imagen del producto en publicidad y medios de comunicación.

Pixel Art

Los gráficos del juego se construyen como mosaicos. Este estilo es la herencia de las limitaciones gráficas que tenían los primeros ordenadores y videoconsolas, se usa con la intención de dar un aspecto retro.



1.29 Shovel Knight

as.com, (2020). Shovel Knight. [imagen] Recuperado de https://as01.epimg.net/meristation/imagenes/2014/07/01/album/1404202140_202140_000002_album_normal.png.



1.30 FEZ

pcgamer.com, (2020). FEZ. [imagen] Recuperado de <https://cdn.mos.cms.futurecdn.net/GztKgubjffM3dishEVyXfj.jpg>

Voxels

Realmente una subcategoría de gráficos 3D, donde la unidad mínima de representación pasa a ser una caja 3D.



1.31 Minecraft: Dungeons

media.vandal.net, (2020). Minecraft: Dungeons. [imagen] Recuperado de https://media.vandal.net/m/74356/minecraft-dungeons-201962511505948_1.jpg

Capítulo 1. La industria de los Videojuegos y el rol del Diseñador

1.5 Estilos gráficos de videojuegos

2D Illustration

Tiende a parecerse más a una ilustración, dando un resultado similar a lo que podemos ver en los dibujos animados.



1.32 Rayman Legends

ubisoft.com. (2020). Rayman Legends [imagen]. Recuperado 24 Marzo 2020, de https://ubistatic19-a.akamaihd.net/ubicomstatic/es-MX/global/media/raymanlegends_screenshot_lushjungle_gc_160462.jpg.



1.33 Cuphead

vandal.espanol.com. (2020). Cuphead [imagen]. Recuperado 24 Marzo 2020, de https://media.vandal.net/i/1280x720/24774/cuphead-201782194553_1.jpg.

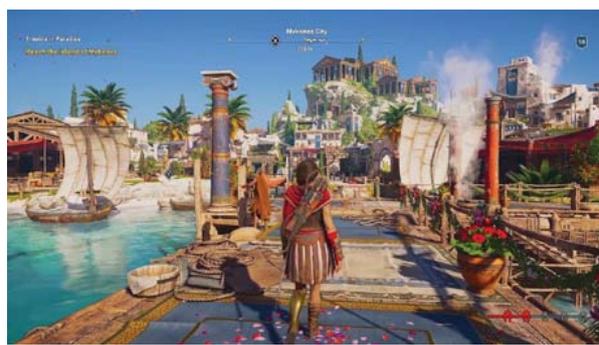
Photorealistic

Estilo que simula de manera fiel la realidad



1.34 Mortal Kombat 11

wccfttech.com. (2020). Mortal Kombat 11 [imagen]. Recuperado 24 Marzo 2020, de <https://cdn.wccfttech.com/wp-content/uploads/2019/04/WCCFmortalkombat1123.jpg>.



1.35 Assassin's Creed: Odyssey

dagersystem.com. (2020). Assassin's Creed: Odyssey [imagen]. Recuperado 24 Marzo 2020, de <https://mdhtpos2i4-flywheel.netdna-ssl.com/wp-content/uploads/2019/01/assassins-creed-odyssey-gameplay.jpg>.

Capítulo 1. La industria de los Videojuegos y el rol del Diseñador

1.5 Estilos gráficos de videojuegos

Gráficos 2.5D, Pseudo 3D o Cámara Fija

Se presentan como sprites 2D aplicados a planos 3D o modelos 3D completos presentados a través de una cámara con una vista fija en 2D, dando la sensación de juego 3D.



1.36 New Super Mario Bros U Deluxe Nintendo Switch. lavozdeg Galicia.es, (2020). New Super Mario Bros U Deluxe Nintendo Switch. [imagen] Recuperado de <https://cflvdg.avoze/default/2019/01/22/00121548187082973942557/Foto/j22e9077.jpg>



1.37 Kirby Star Allies | Nintendo Switch. nintendo-europe.com, (2020). Kirby Star Allies | Nintendo Switch. [imagen] Recuperado de https://cdn02.nintendo-europe.com/media/imagenes/08_content_imagenes/games_6/nintendo_switch_7/nswitch_kirbystarallies_1/NSwitch_KirbyStarAllies_inhale.jpg

2D Cutout

Estilo gráfico 2D donde las extremidades de los personajes están diseñadas para ser movibles y animadas individualmente. Paper Mario and the Thousand-year door, Odin Sphere, Muramasa: The Demon Blade.



1.38 Paper Mario: Color Splash nacionpix.com, (2020). Paper Mario: Color Splash. [imagen] Recuperado de <https://nacionpix.com/wp-content/uploads/2016/09/Paper-Mario-Color-Splash-story.jpg>



1.39 Odin Sphere: Leifthrasir atomix.vg, (2020). ODIN SPHERE: LEIFTHRASIR Odin Sphere: Leifthrasir. [imagen] Recuperado de <https://cdn.atomix.vg/wp-content/uploads/2015/08/imagenes-imagenes-Odin-Sphere-Leifthrasir-23.jpg>

Capítulo 1. La industria de los Videojuegos y el rol del Diseñador

1.5 Estilos gráficos de videojuegos

Collage

Estilo donde se usan y mezclan diferentes estilos y medios libremente.

Procedurally-generated

Es un método de creación de elementos con algoritmos de computadora en lugar de forma manual. *Euforia, Love, kriegler, Minecraft, Subversion.*



1.40 No Man's Sky

masgamers.com. (2020). No Man's Sky [imagen]. Recuperado 23 Marzo 2020, de <https://www.masgamers.com/wp-content/uploads/2019/11/No-Man-Sky-foto-1.jpg>.



1.41 Minecraft

nintenderos.com. (2020). Minecraft [imagen]. Recuperado 23 Marzo 2020, de <https://www.nintenderos.com/wp-content/uploads/2018/12/DuJth40XgAMojr2.jpg>.

Cel-shaded

Un estilo limpio y fresco, da al juego una sensación atemporal marcando el contorno y la silueta de los objetos de forma simple, puede parecerse al *anime/cartoon*. *The Legend of Zelda: Wind Waker, Giftpia, XIII, Afro Samurai, Viewtiful Joe, Okami, Fire Fall, Killer 7.*



1.42 The Legend of Zelda Wind Waker

articulo.mercadolibre.com.mx. (2020). The Legend Of Zelda Wind Waker [imagen]. Recuperado 23 Marzo 2020, de https://http2.mlstatic.com/the-legend-of-zelda-wind-waker-juego-gamecube-envio-gratis-D_NQ_NP_690451-ML-M26749179797_022018-F.jpg.

Capítulo 1. La industria de los Videojuegos y el rol del Diseñador

1.5 Estilos gráficos de videojuegos

Illustrative

Estilo de meticulosa manufactura, cuidadosamente dibujado a mano artesanalmente; en entornos aplicados a 3d es a menudo mezclado con *cel-shaded*. *Tales of Vesperia*, *Prince of Persia*, *Battle Field Heroes*, *Dark Chronicle*, *Borderlands*, *El Shaddai*, *Ascension of the Metatron*, *Yoshi's Island*



1.43 Borderlands 3

Steam. (2020). Borderlands 3 [imagen]. Recuperado 23 Marzo 2020, de https://steamcdn-a.akamaihd.net/steam/apps/49520/ss_8b1154bc507678f4a21fcc2d9083ef79a5db4e76.1920x1080.jpg?t=1582831290.

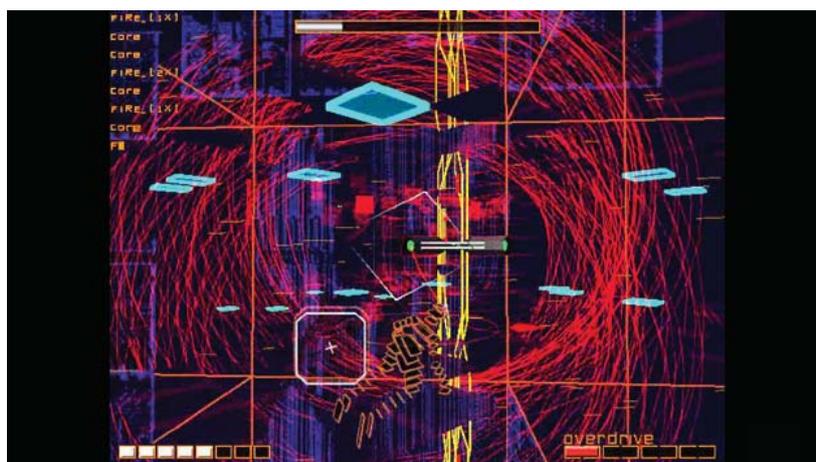


1.44 Yoshi's Island

startvideojuegos.com. (2020). Yoshi's Island [imagen]. Recuperado 23 Marzo 2020, de http://www.startvideojuegos.com/wp-content/uploads/2014/11/analogico_vidas.png.

Wireframe

Estilo en que los gráficos son esquematizados utilizando algoritmos de render que resultan en una imagen semitransparente, de la que sólo se dibujan las aristas de la malla del objeto o polígono. *Cosmic Smash*, *Rez*,



1.45 REZ

youtube.com. (2020). REZ [imagen]. Recuperado 23 Marzo 2020, de <https://i.ytimg.com/vi/YZL4Cbt-kin/kmaxresdefault.jpg>.

Capítulo 1. La industria de los Videojuegos y el rol del Diseñador

1.5 Estilos gráficos de videojuegos

Dark Toon

Estilo deformado y tierno, una versión lúdica de posibles temas oscuros reflejados en los visuales. *Ushiro*, *Devil Survivor*, *Mushroom Men*, *Knights of the Crystal*, *The Nightmare before Christmas*.



1.46 Mushroom Men: The Spore Wars

gamereactor.es. (2020). Mushroom Men: The Spore Wars [imagen]. Recuperado 23 Marzo 2020, de https://www.gamereactor.es/media/19/mushroommen_161924b.jpg.



1.47 The Nightmare Before Christmas: Oogie's Revenge

youtube.com. (2020). The Nightmare Before Christmas: Oogie's Revenge [imagen]. Recuperado 23 Marzo 2020, de <https://i.ytimg.com/vi/7zS-Pt07xAk/maxresdefault.jpg>.

Urban

Un estilo fresco, salvaje y duro, que remite a elementos callejeros, como moda, arte y música. *3 Kingdoms TD*, *S4 League*, *Marc Echo's Getting Up*.



1.48 S4 League

youtube.com. (2020). S4 League [imagen]. Recuperado 23 Marzo 2020, de <https://i.ytimg.com/vi/76eE5SD3SWc/maxresdefault.jpg>.



1.49 Marc Echo's Getting Up: Contents Under Pressure

gamefabrique.com. (2020). Marc Echo's Getting Up: Contents Under Pressure [imagen]. Recuperado 23 Marzo 2020, de <https://gamefabrique.com/storage/screenshots/pc/marc-eckos-getting-up-contents-under-pressure-07.png>.

Capítulo 1. La industria de los Videojuegos y el rol del Diseñador

1.5 Estilos gráficos de videojuegos

Innocent Feelgood

Estilo inocente, vivido y positivo; representa un mundo simulado desde el punto de vista de un niño, reduce el desorden y el ruido para enfocarse en las partes esenciales. *Animal Crossing*, *Ameba Pico*, *Chibi Robo*, *Earthbound*, *HomeLand*, *Parappa The Rapper*



1.50 Animal Crossing New Horizons

vsgamers.es. (2020). Animal Crossing New Horizons [imagen]. Recuperado 23 Marzo 2020, de https://www.vsgamers.es/thumbnails/product_gallery_medium/uploads/products/nintendo/juegos/juego-nintendo-switch-animal-crossing-new-horizons/galeria/juego-nintendo-switch-animal-crossing-new-horizons-galeria-2.jpg.

1.51 Farm Together - Chickpea Pack

store.steampowered.com. (2020). Farm Together - Chickpea Pack [imagen]. Recuperado 23 Marzo 2020, de https://steamcdn-a.akamaihd.net/steam/apps/995640/ss_66d1ef9a6988f94765abb18ea9ec535109ff55dc.1920x1080.jpg?t=1568718561.

Noire

Sombras duras, contraste y colores apagados. *Beneath a Steel Sky*, *Hotel Dusk*, *Gemini Rue*, *Deus Ex Human Evolution*, *Insecticide*.



1.52 Deus Ex: Mankind Divided

articulo.mercadolibre.com.mx/. (2020). Deus Ex: Mankind Divided [imagen]. Recuperado 23 Marzo 2020, de https://http2.mlstatic.com/deus-ex-mankind-divided-ed-de-lujo-xbox-one-codigo-D_NQ_NP_607739-MLM28862307611_122018-F.jpg.



1.53 Cyberpunk 2077

pcmrace.com. (2020). Cyberpunk 2077 [imagen]. Recuperado 23 Marzo 2020, de https://www.pcmrace.com/wp-content/uploads/2019/08/Cyberpunk2077-Should_have_taken_the_elevator_RGB-en.jpg.

Capítulo 1. La industria de los Videojuegos y el rol del Diseñador

1.5 Estilos gráficos de videojuegos

Futuristic

De estética minimalista, pulcra, ordenada, limpia, con tintes futuristas. *Luminez*, *WipeOut*, *Mirrors Edge*, P.N.03.



1.54 Wipeout Omega Collection

store.playstation.com. (2020). Wipeout Omega Collection [imagen]. Recuperado 23 Marzo 2020, de <https://i.ytimg.com/vi/NIQso7dAS3c/maxresdefault.jpg>.



1.55 Mirror's Edge

elblojdeneojin.blogspot.com. (2020). Mirror's Edge [imagen]. Recuperado 23 Marzo 2020, de https://1.bp.blogspot.com/-d4XHSElC4JM/W1gaVaO_cII/AAAAAAAAfns/TH7LScD3hfE8fElvXzCExFyiTPCv3uLagCLcBGAs/s1600/mirror%2527s%2Bedge%2Bgameplay.jpg.

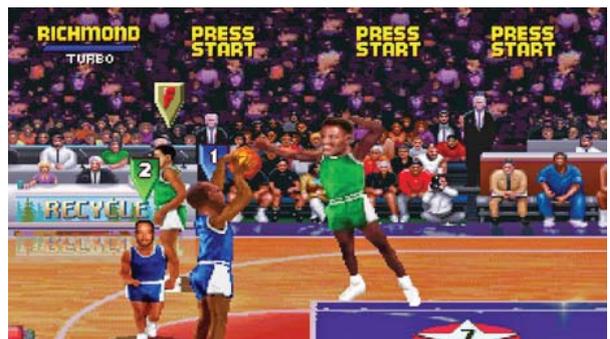
Digitized sprites

Se utilizan fotografías reales de actores o modelos digitalizados para crear al personaje, creando la sensación de gráficos realistas. *Mortal Kombat*



1.56 Mortal Kombat

upsocl.com. (2020). Mortal Kombat [imagen]. Recuperado 23 Marzo 2020, de <http://www.upsocl.com/wp-content/uploads/2016/02/PORTADA-MK.jpg>.



1.57 NBA Jam

nintenderos.com. (2020). NBA Jam [imagen]. Recuperado 23 Marzo 2020, de <https://www.nintenderos.com/wp-content/uploads/2019/10/dims-e1571779062565.jpg>.

Capítulo 1. La industria de los Videojuegos y el rol del Diseñador

1.5 Estilos gráficos de videojuegos

Pre-rendered graphics

Se usan imágenes en 3d prrenderizadas y compiladas como sprites en el juego, dando la apariencia de realismo. *Donkey Kong Country*, *Killer Instinct*, *Yoshi's Story*.



1.58 Donkey Kong Country

www.nintendo.es. (2020). Donkey Kong Country [imagen]. Recuperado 23 Marzo 2020, de https://cdn02.nintendo-europe.com/media/imagenes/06_screenshots/games_5/virtual_console_wii_u_3/wiuvc_donkeykongcountry/WiiUVC_DonkeyKongCountry_02.bmp.

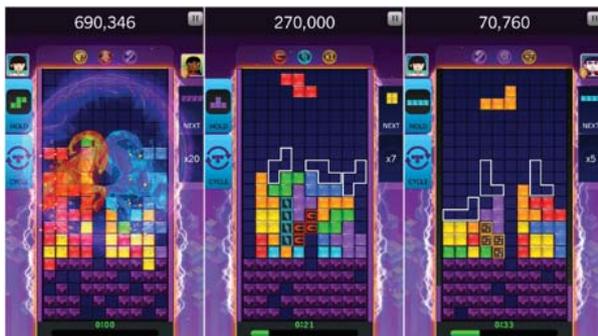


1.59 Killer Instinct

youtube.com. (2020). Killer Instinct [imagen]. Recuperado 23 Marzo 2020, de <https://i.ytimg.com/vi/2McPr2nH2fM/maxres-default.jpg>.

Abstract

Se enfoca en representar el juego por medio de formas geométricas en lugar de personajes, objetos y lugares figurativos. *Tetris*, *Pong*, *Rez*, *Geometry Dash*.



1.60 Tetris

boingboing.net. (2020). Tetris [imagen]. Recuperado 23 Marzo 2020, de https://media.boingboing.net/wp-content/uploads/2020/01/blitz-frame_feature.jpg.



1.61 Geometry Dash

geometrydash.onl. (2020). Geometry Dash [imagen]. Recuperado 23 Marzo 2020, de <https://geometrydash.onl/>.

Capítulo 1. La industria de los Videojuegos y el rol del Diseñador

1.5 Estilos gráficos de videojuegos

Stop motion

Se crean imágenes cuadro por cuadro que después se unirán como sprites animados para formar elementos de videojuego.



1.62 Armikrog

Twitter. (2020). Armikrog [imagen]. Recuperado 23 Marzo 2020, de <https://pbs.twimg.com/media/CQKvb2PWcAAiiZb.jpg>.



1.63 Harold Halibut

haroldhalibut.com. (2020). Harold Halibut [imagen]. Recuperado 23 Marzo 2020, de <http://haroldhalibut.com/wp-content/uploads/2016/07/AgoraArcades2.png>.

Capítulo 1. La industria de los Videojuegos y el rol del Diseñador

1.6 Uso de programación por parte del diseñador

1.6 Uso de programación por parte del diseñador

El mundo de la creación digital evoluciona constantemente, por lo que los creativos en búsqueda de nuevos caminos, formas, y maneras de representar su visión de la realidad, llegan a experimentar y adquirir conocimiento de áreas que no precisamente pertenecen a su formación original, por esta razón la programación enfocada a creativos es una poderosa herramienta que acelera los procesos volviéndolos más prácticos, también, por medio de la programación se pueden presentar experiencias visuales y formas interactivas que de manera tradicional no se podrían obtener.

También es cierto que la industria laboral busca cada vez más la integración de personal con conocimientos estéticos formales y logros técnicos en áreas de programación, es por eso que el diseñador tendrá la opción de emprender especializaciones o estudios de posgrado donde pueda adquirir formación en *STEM*.

Una opción práctica para iniciarse en el mundo de la programación es empezar por programación visual, una práctica que deja en claro los fundamentos básicos de la programación, con una curva de aprendizaje no muy elevada.

Es importante tomar en cuenta las estimaciones del *World Economic Forum* en cuestión de las áreas donde existirán más oportunidades laborales en los próximos años por el desarrollo de nuevas tecnologías, en las cuales el diseñador podrá solventar su trabajo si adquiere conocimientos de programación y que serán fundamentales para el desarrollo de las sociedades, las áreas principales según estas estimaciones son: Realidad Virtual (VR), Realidad Aumentada (AR), desarrollo de Apps (entre ellas videojuegos) y Web.

Realidad Virtual

Se define como una realidad inmersiva simulada por computadora que crea un entorno físico que no existe. Estos entornos digitales podrían basarse en lugares reales o imaginarios.

Con la realidad virtual, se puede mover y mirar en todas las direcciones, como si se estuviera físicamente allí.



1.64 Virtual Reality

arpost.co. (2020). Virtual Reality [imagen]. Recuperado de <https://arpost.co/wp-content/uploads/2019/12/How-to-Start-Career-in-Virtual-Reality.jpg>

Capítulo 1. La industria de los Videojuegos y el rol del Diseñador

1.6 Uso de programación por parte del diseñador

Realidad Aumentada

Se define como una versión mejorada o aumentada de la realidad, que le es agregada información digital en forma de gráficos fijos, audio o video, creada mediante el uso de la tecnología.⁴

AR es diferente de VR en que AR aumenta (se suma a) una escena real o existente en lugar de crear algo nuevo desde cero.⁵



1.65 Augmented Reality

robots.net. (2020). Augmented Reality [imagen]. Recuperado de <https://robots.net/wp-content/uploads/2019/08/Augmented-Reality-1280x720.jpg>

Desarrollo de sitios web Frontend

Frontend son todas las tecnologías de diseño y desarrollo web que corren en el navegador y que desarrollan la interactividad con los usuarios.

Algunos lenguajes usados son: HTML, CSS, Javascript, React, Redux, Angular, Bootstrap, Foundation, LESS, Sass, Stylus y PostCSS.⁶

Cabe mencionar que la principal actividad en la que el diseñador se relaciona con programación es con la web; cada vez las barreras que separan al diseño y desarrollo web son más delgadas, por lo que los empleadores buscan perfiles que puedan cubrir estas áreas tanto para obtener experiencias de diseño mucho más enriquecedoras como por la rapidez en la creación de los productos a un menor costo.

4 Mealy, Paul. (2018). Virtual & Augmented Reality for Dummies (For Dummies (Computer/Tech)). 1a Ed. USA. Wiley.

5 Augmented Reality - The Fourth Industrial Revolution | UK | Waracle. (2020). Retrieved 3 January 2020, from <https://waracle.com/blog/augmented-reality/augmented-reality-the-fourth-industrial-revolution/>

6 Qué es Frontend y Backend. (2020). Retrieved 3 January 2020, from <https://platzi.com/blog/que-es-frontend-y-backend/>

Capítulo 1. La industria de los Videojuegos y el rol del Diseñador

1.6 Uso de programación por parte del diseñador

THE FRONT-END SPECTRUM



1.66 Front-End

Pelletier, J. (2020). Front-End [imagen]. Recuperado de https://miro.medium.com/max/5254/1*bbxDi-plpyPrwg-g7GXC-FA.png

El desarrollador de apps toma conceptos de ingeniería de software para crear, testear y programar apps para computadoras, teléfonos móviles y diversos aparatos electrónicos.

Convive fuertemente con diseñadores web en las áreas de Experiencia de Usuario (UX)⁷ y Diseño de Interfaz de Usuario (UI).⁸



1.67 Lista de apps más descargadas de 2019

geekymint.com. (2020). Lista de apps más descargadas de 2019 [imagen]. Recuperado de <https://geekymint.com/wp-content/uploads/2020/02/top-apps-ww-q1-2019-post-hero-imagen-990x591.jpg>

⁷ Diseño de Experiencia de Usuario o UX Design | Platzi. (2020). Retrieved 3 January 2020, from <https://platzi.com/diseno-ux/>

⁸ Diseño de Interfaces (UI). (2020). Retrieved 3 January 2020, from <https://platzi.com/interfaces-ui/>

Capítulo 1. La industria de los Videojuegos y el rol del Diseñador

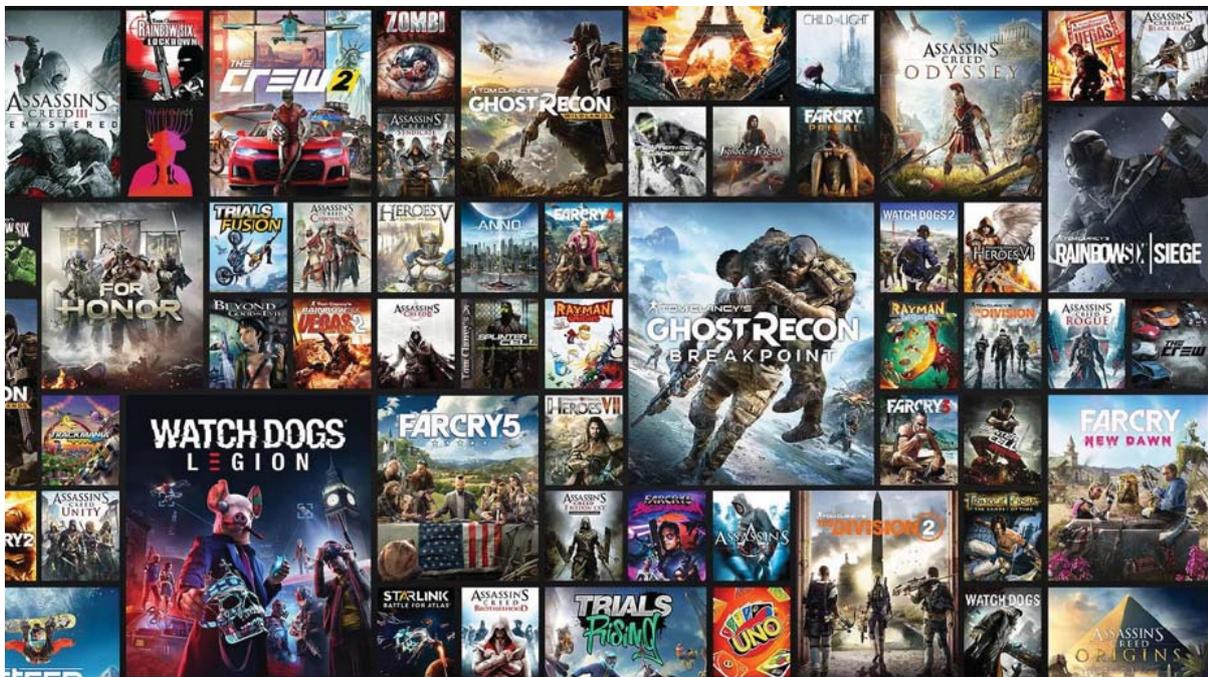
1.6 Uso de programación por parte del diseñador

Videojuegos

El uso de *Game Engines* generan una opción creativa real y accesible para el desarrollo de juegos por parte del Diseñador, puede ser usando programación visual con Engines como *Buildbox* y *Construct 3*, que son 2 de los programas más fáciles y accesibles para hacer videojuegos generando resultados interactivos muy sencillos ya que contienen módulos interactivos preestablecidos; y por otra parte si el diseñador tiene conocimientos previos en programación y *STEM* o posteriormente se especializa en esa área podrá producir videojuegos más complejos con *Engines* como *Unity* o *Unreal Engine*, donde se deberán conocer estructuras de programación y lenguajes de computadora como C # o C++.

El proyecto práctico de esta investigación, el videojuego llamado **Cazareliquias** hecho en *Construct 2*, además de presentar la materialización de un juego, tiene también la función de practicar los conceptos de programación más usados como forma de introducción a la programación.

Existen también otras áreas donde el uso de la programación está relacionado o tiene una afinidad al arte, creatividad y diseño, como son: *Processing*, *Programación en 3d*, *expresiones en After Effects*, *Efectos Visuales VFX* y *Technical Artist*.



1.68 Colección de videojuegos de Ubisoft

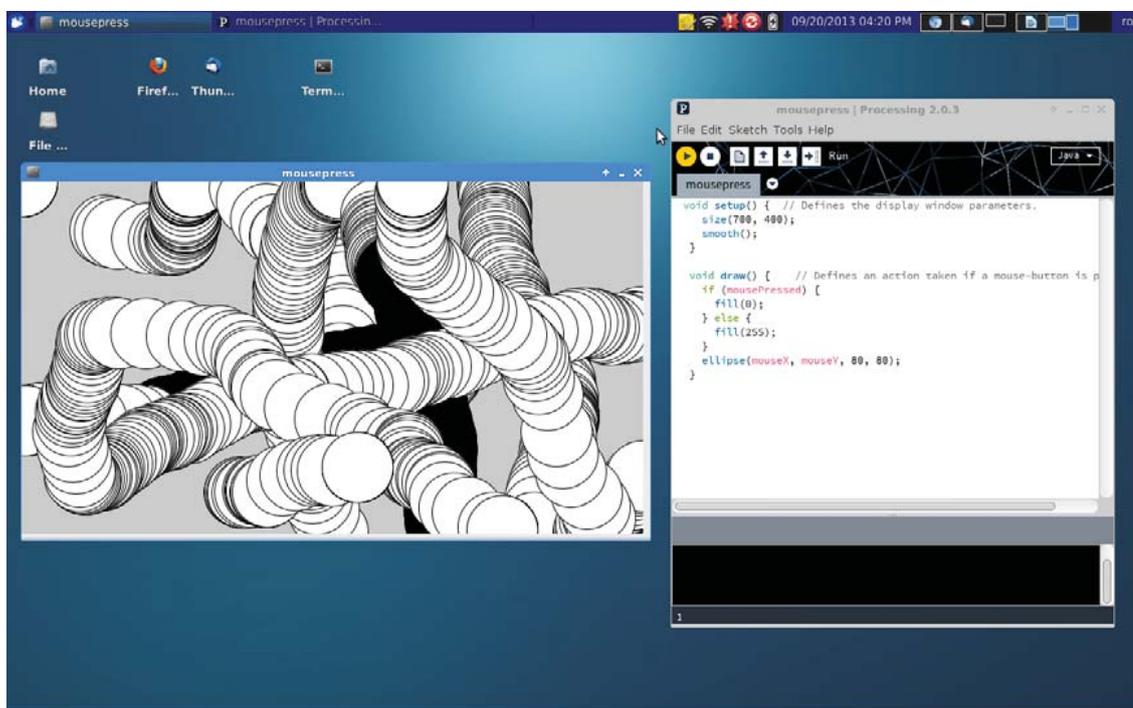
store.ubi.com. (2020). Colección de videojuegos de Ubisoft [imagen]. Recuperado de https://store.ubi.com/on/demandware.static/-/Library-Sites-shared-library-web/default/dw21a2db4d/images/uplay_plus/nca/images/uplayplus_hero.jpg

Capítulo 1. La industria de los Videojuegos y el rol del Diseñador

1.6 Uso de programación por parte del diseñador

Programación interactiva con Processing

Processing es un software inicialmente enfocado a diseñadores y artistas creado para enseñar los fundamentos de programación en un contexto visual por medio de la creación de gráficos; es una herramienta profesional que es usada en artes electrónicas, arte de los nuevos medios y diseño visual e interactivo; algunos ejemplos de uso son: visualización de datos, generación de imágenes complejas con algoritmos matemáticos, desarrollo de instalaciones artísticas y fomento de la interacción en el entorno de diseño digital utilizando video, imagen, sonido y texto, por mencionar algunas.⁹



1.69 Processing

insights.dice.com. (2020). Processing [imagen]. Recuperado de <https://insights.dice.com/wp-content/uploads/2014/03/Processing-Screenshot.png>

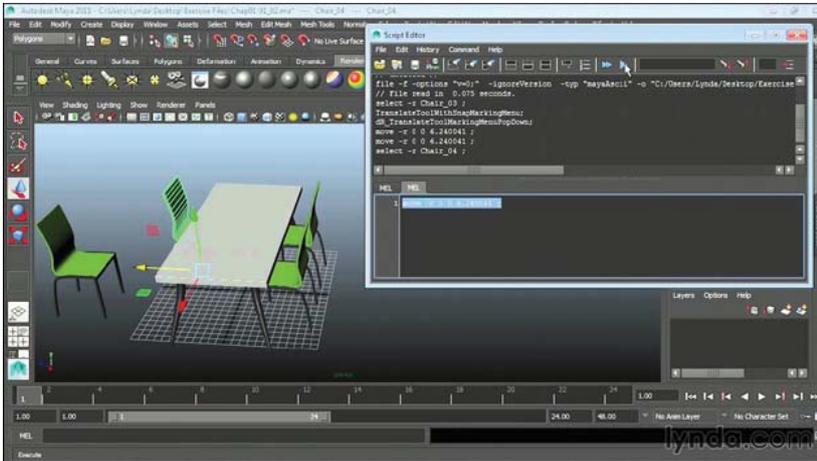
Scripts para software 3d

El uso de scripts con lenguajes de programación como Python, MEL, Maxscript o c++, incrementan la eficiencia en el flujo de trabajo dentro del software de 3d (cinema 4d, 3ds Max, Houdini, Maya, etc.), facilitando tareas que aumentan la productividad.

⁹ Processing (programming language). (2020). Retrieved 3 January 2020, from [https://en.wikipedia.org/wiki/Processing_\(programming_language\)](https://en.wikipedia.org/wiki/Processing_(programming_language))

Capítulo 1. La industria de los Videojuegos y el rol del Diseñador

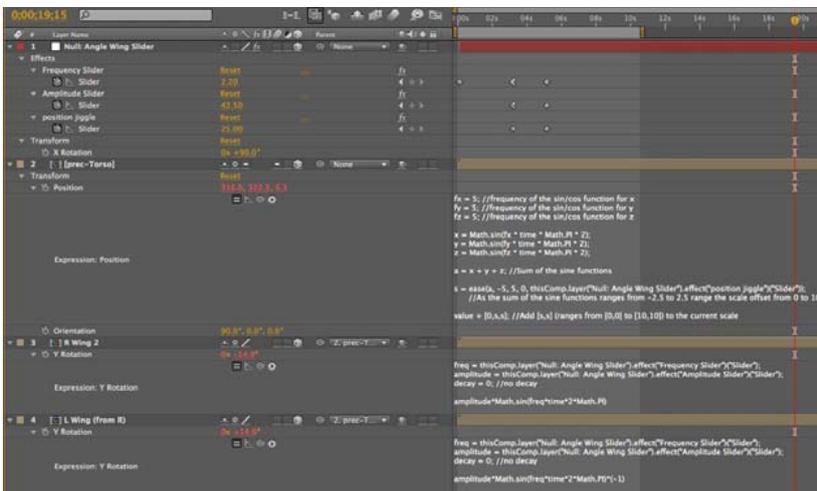
1.6 Uso de programación por parte del diseñador



1.70 MEL Script Maya
lynda.com. (2020). MEL Script Mayo [imagen].
Recuperado de https://i.ytimg.com/vi/Wez-Th_k-5cE/maxresdefault.jpg

After Effects Expressions

Cuando se quiere crear y vincular animaciones complejas en After Effects, pero se desea evitar la creación de decenas o cientos de keyframes manualmente, se usan scripts basados en javascript, llamados expresiones (expressions).



1.71 After Effects expressions
creativecow.net. (2020). After Effects expressions [imagen]. Recuperado de https://i1.creativecow.net/u/821100/bf_expression.jpg

Los gráficos por computadora (CG - Computer Graphics) permiten la representación realista de efectos de agua, fuego, humo, tela, partículas, pelo y viento en el mundo virtual. El cine y los videojuegos son dos industrias representativas para las cuales la tecnología CG es aplicada comercialmente. En estos campos, CG puede llamarse Efectos Visuales.

Capítulo 1. La industria de los Videojuegos y el rol del Diseñador

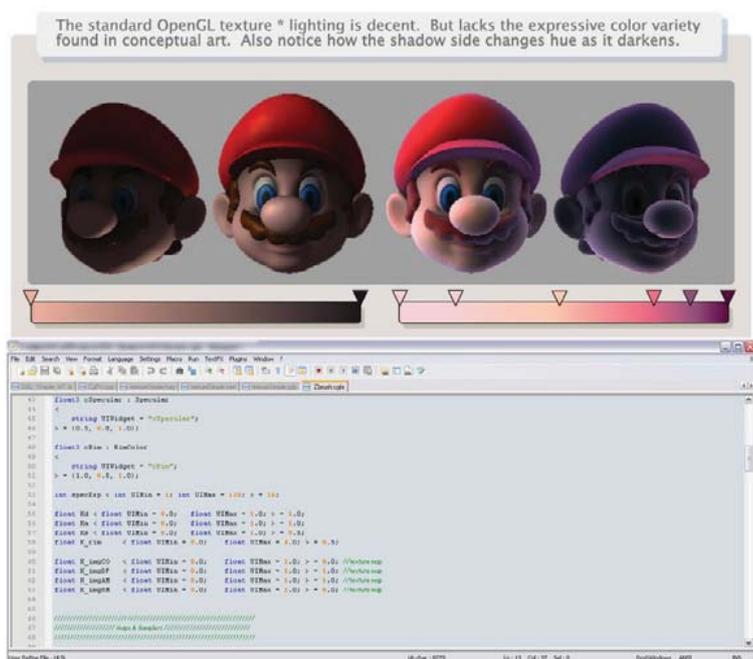
1.6 Uso de programación por parte del diseñador

Visual Effects - VFX

Los VFX para cine se centran en el realismo, por lo que para concebir estos efectos visuales de alta calidad es necesario utilizar grandes cantidades de recursos de cómputo

Real time VFX

Los VFX en los videojuegos se centran en el render de los efectos en tiempo real, bajo limitantes que impone la cantidad de memoria y poder de procesamiento del CPU de la computadora en el que se usan¹⁰



1.72 Textura Open GL

1.72 Ton, M. (2020). Textura Open GL [imagen]. Recuperado de https://sites.google.com/site/miketon/_/rsrc/1468736852843/coding/Mario-Head_shader_2-1.jpg

Technical Artist

Actualmente las tecnologías aplicadas en videojuegos y efectos especiales están convergiendo por lo que se busca especialistas que mantengan un balance entre el rigor técnico y la creatividad; este es el caso del *Technical Artist* ya que cuenta con conocimientos en programación y es el enlace entre el lado artístico y el lado técnico de un proyecto, haciéndolo posible, facilitándole.

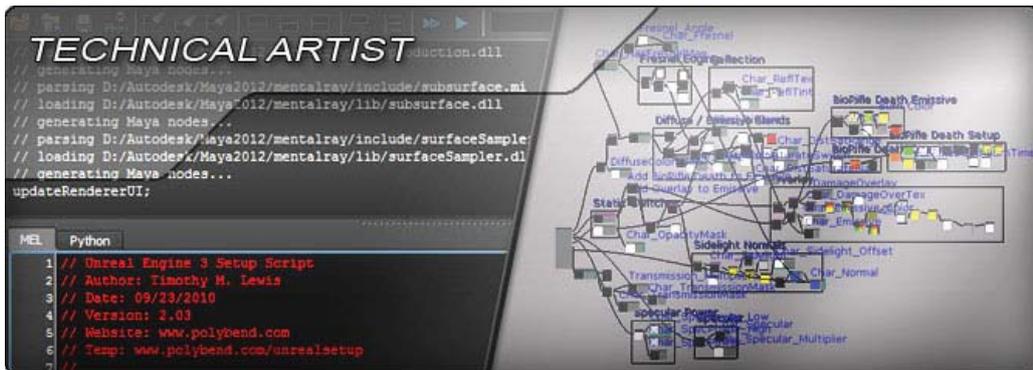
Algunos roles principales en los que interviene son: tener sólidos principios de 3d, ser el responsable de programar los *shaders* para materiales, del real-time *rendering*, de la preparación de *riggs* especiales para animación, creación de colisiones, cuerpos

10 Chang-Hun, Kim. Sun-Jeong, Kim. Soo-Kyun, Kim. Shin-Jin, Kang. (2015). Real-Time Visual Effects for Game Programming. 1a Ed. Singapore. Springer Singapore.

Capítulo 1. La industria de los Videojuegos y el rol del Diseñador

1.6 Uso de programación por parte del diseñador

rígidos, deformaciones, fluidos, uso de *Unity* y *Unreal*, así como estar al pendiente técnicamente de que el material sea transferido correctamente entre etapas de desarrollo.



1.73 Technical Artist

docs.unrealengine.com. (2020). Technical Artist [imagen]. Recuperado de <https://docs.unrealengine.com/udk/Three/rsrc/Three/TechnicalArtistHome/tech.jpg>

Para concluir se puede decir que el ritmo de evolución en la actualización tecnológica avanza a pasos agigantados y este avance se ve reflejado en todas las áreas productivas de la sociedad generando nuevas estructuras de conocimiento, de esta forma el rol del diseñador saldrá beneficiado de muchas maneras, empezando por enumerar desde el potencial que significa trabajar con computadoras más poderosas que generan imágenes super definidas con más resolución en tiempos de proceso de render rápidos y visualizaciones en tiempo real más detalladas; para ejemplificar se puede hablar de la salida de la computadora más cara en la historia de Apple, una Mac Pro con procesador Intel de 28 núcleos, 1.5 terabytes de memoria RAM con un precio de \$53.247USD que sin duda presenta un entorno de trabajo sumamente rápido y poderoso; pero dejando a un lado las capacidades técnicas para producción y los beneficios que traen consigo, el mayor avance que está cobrando fuerza es el uso de la programación como sustento de nuevas formas de hacer las cosas, en primer lugar para cubrir expectativas creativas abriendo nuevos horizontes, presentando contenidos innovadores en un entorno cada vez más lleno de nativos digitales ya que por medio de la programación los diseñadores podrán dar un giro en la creación de gráficos y experiencias interactivas de una forma vanguardista adoptando esta medio de producción estandarizadamente en la labor profesional y en segundo lugar para acelerar los procesos generativos volviéndolos más eficientes; el dominar la programación por parte de diseñadores generará una conexión constante con más áreas afines y relacionará su labor en nuevos campos, accediendo a un campo laboral más amplio; en términos prácticos obtendrá mayores oportunidades de trabajo y crecimiento.

Capítulo 1. La industria de los Videojuegos y el rol del Diseñador

1.6 Uso de programación por parte del diseñador

También el rol de diseñador-programador tendrá un beneficio no solo como empleado dentro de una compañía, sino como empresario o emprendedor donde pueda ser él el responsable de un proyecto unificador, ya que la programación le permitirá cubrir y dominar un mayor número de áreas de producción a un menor costo, ofreciendo más servicios a un mercado ávido por nuevas experiencias, logrando solventar el proyecto como inversión real.

Es tanto el potencial que tienen el diseño y la programación en su interacción práctica que es muy probable que la industria profesional empiece a demandar estas áreas cada vez más, esto obligará que de manera académica la programación y el diseño lleguen a converger en planes de estudio en un tiempo no mayor a diez años, modificando los programas académicos actuales, no presentando un área u otra como materias optativas sino introduciéndolas en un tronco común que generaría roles nuevos en beneficio del entorno productivo y al diseñador-programador mismo; mientras esto sucede, el camino y las decisiones académicas que el diseñador escoja son fundamentales ya que podrá tomar cursos, especializarse o estudiar una licenciatura nueva, pero tendrá que tomar en cuenta el grado de conocimiento que quiera adquirir, el tiempo y dinero invertido, además de escoger las áreas que le parezcan atractivas para desenvolverse profesionalmente ya que dependiendo la elección será el tiempo de retorno de inversión en un entorno que evoluciona muy rápido.

Capítulo 1. La industria de los Videojuegos y el rol del Diseñador

1.7 Adopción del pensamiento computacional

1.7 Adopción del pensamiento computacional

El término Pensamiento Computacional fue concretado por Jeannette M. Wing en 2006¹¹ y se basa en la definición sobre la capacidad y aptitud que tiene una persona de pensar, organizar y crear un plan para la solución de problemas de forma eficiente, utilizando métodos resolutivos que tienen fundamento en las metodologías de ciencias de cómputo o programación, aunado a ello la ventaja en la adopción de este pensamiento no se limita a ciencias de cómputo sino que siguiendo este modelo, la resolución de problemas puede aplicarse para diferentes áreas de conocimiento, además el pensamiento computacional puede ser usado también en la resolución de problemas cotidianos no sólo enfocado a cuestiones académicas, de ahí la gran importancia que este tema tiene en el desarrollo de las sociedades del siglo XXI, incluso algunas escuelas de educación pre universitaria en países como Portugal, Dinamarca y Alemania han empezado a enseñar en sus aulas fundamentos de programación visual y temas de *STEM (Science, Technology, Engineering and Mathematics)*.

Para la resolución de un problema de programación se ponen en práctica varios conceptos que forman la base del pensamiento computacional por lo que se presentará una descripción de cada componente.¹²

Abstracción

La abstracción es el proceso de hacer que un artefacto sea más comprensible mediante la reducción de los detalles innecesarios. La habilidad en la abstracción radica en elegir el detalle correcto para ocultar para que el problema se vuelva más fácil, sin perder nada importante.

Pensamiento Algorítmico

El pensamiento algorítmico es una forma de llegar a una solución a través de una definición clara de los pasos.

Automatización

La automatización es un proceso de ahorro de mano de obra en el que una computadora recibe instrucciones de ejecutar un conjunto de tareas repetitivas de manera rápida y eficiente en comparación con el poder de procesamiento de un ser humano.

11 Wing, Jeannette. (2006). Computational Thinking. Communications of the ACM. 49. 33-35. 10.1145/1118178.1118215.

12 Bocconi, Stefania & Chiocciariello, Augusto & Dettori, Giuliana & Ferrari, Anusca & Engelhardt, Katja & Kamylyis, Panagiotis & Punie, Yves. (2016). Developing Computational Thinking in Compulsory Education. Implications for policy and practice. EUR - Scientific and Technical Research Reports. 10.2791/792158.

Capítulo 1. La industria de los Videojuegos y el rol del Diseñador

1.7 Adopción del pensamiento computacional

Descomposición

La descomposición es una forma de pensar acerca de los artefactos en términos de sus componentes. Las partes se pueden entender, resolver, desarrollar y evaluar por separado. Esto hace que los problemas complejos sean más fáciles de resolver, las situaciones novedosas se entiendan mejor y los sistemas grandes sean más fáciles de diseñar.

Depuración (debugging)

La depuración es la aplicación sistemática de análisis y evaluación utilizando habilidades tales como pruebas, rastreo y pensamiento lógico para predecir y verificar resultados.

Generalización

La generalización está asociada con la identificación de patrones, similitudes y conexiones, y la explotación de esas características. Es una forma de resolver rápidamente nuevos problemas basados en soluciones previas a los problemas y construir sobre la experiencia previa. Los algoritmos que resuelven algunos problemas específicos se pueden adaptar para resolver una clase completa de problemas similares.

El pensamiento computacional es aplicado en la codificación y programación y a menudo se usan indistintamente para indicar el proceso de escritura de instrucciones para que una computadora las ejecute. Sin embargo, la programación se refiere a la actividad más amplia de analizar un problema, diseñar una solución e implementarla. La codificación es la etapa de implementación de soluciones en un lenguaje de programación particular.

Por lo tanto en primer término el Pensamiento Computacional solventará la innovación en desarrollos tecnológicos por medio de resolución de problemas computacionales que se presenten en la llamada 4ta Revolución Industrial, creando tecnología que permee el avance de la sociedad de forma sustentable, además a nivel individual sentará las bases para que la persona tenga la aptitud y disposición de forma natural hacia los cambios y actualizaciones tecnológicas que se presenten en todas las esferas de la sociedad de forma eficiente.

En segundo término, generará un beneficio en torno a la calidad de vida de las personas en su ámbito común, conforme estas puedan realizar tomas de decisiones fundamentadas en la razón, teniendo un pensamiento más ordenado, abstracto y fijo en objetivos por medio de la resolución de problemas.

Capítulo 2. Introducción a los Videojuegos

2.1 Definición de Videojuegos

Capítulo 2. Introducción a los Videojuegos

2.1 Definición de Videojuegos

Videojuegos con gráficos sorprendentes, excelente jugabilidad e inteligencia artificial avanzada como los que se juegan en la actualidad se han logrado gracias al desarrollo tecnológico y académico ya que la creación de un videojuego conlleva un proceso complejo y laborioso en donde diversas disciplinas entran en contacto interactuando favorablemente culminando en la producción, mercadeo y venta de un producto que cumple con los requerimientos estandarizados de calidad en una industria que crece exponencialmente; este crecimiento ha afianzado su estatus como la industria principal de entretenimiento, provocando su influencia en la cultura popular, volviendo a los videojuegos un fenómeno global que abarca diferentes medios como TV, cine, Internet y música, además logros como las partidas *multiplayer online*, el aumento de juegos móviles y el afianzamiento de los *e-sports* han provocado que exista un cambio en la interacción social de los *Gamers*, volviéndolos una comunidad dependiente interconectada.

Incluso hay quien afirma que los videojuegos son una forma de arte por lo complejo que pueden llegar a ser las experiencias que brinda visual, narrativa e interactivamente, aunque por supuesto también hay detractores que ven a los videojuegos como un ente netamente funcional donde su objetivo es conseguir la victoria.

Lo que sí es objetivo y real es que los videojuegos son muy atractivos porque presentan simulaciones o realidades ficticias donde cada una de estas tiene su propio orden y lógica, dando la posibilidad al jugador de experimentar situaciones que involucran el desarrollo del pensamiento y la agilidad mental de forma divertida y emocionante, creando empatía hacia estos mundos virtuales, abstrayéndose en una realidad donde él forma parte activa controlando personajes imaginarios explorando sus mundos, lográndose así la inmersión por medio del *Gameplay* y las mecánicas de juego, las partes principales de un videojuego.

De manera genérica un Videojuego es un dispositivo electrónico que tiene como objetivo generar diversión por medio del uso de juegos digitales, controlados mediante mandos específicos, llevados a cabo frente a las pantallas de un televisor, una computadora u otro dispositivo electrónico, ejerciendo una prueba mental al

Capítulo 2. Introducción a los Videojuegos

2.1 Definición de Videojuegos

jugador, de acuerdo con ciertas reglas.¹³

Dentro de un contexto de software, para referirse a los juegos electrónicos dentro del mundo del entretenimiento electrónico, un videojuego se asocia al control y evolución de un personaje por parte del jugador dentro de un juego virtual delimitado por reglas, planteando una serie de retos al usuario con el objetivo de garantizar la diversión y entretenimiento, estas mecánicas suelen tener un componente cognitivo, obligando al jugador a aprender técnicas y a dominar el comportamiento del personaje para cumplir el objetivo final que es la victoria, de éste modo existe una interacción entre el jugador y el videojuego.

Entendiendo estas definiciones se puede decir que un videojuego hace referencia tanto al aparato electrónico donde se usan los juegos, como a los juegos en sí mismos, además en el mercado puede recibir varias denominaciones, como el término ya mencionado Videojuegos (*Videogames*), Juegos Digitales (*Digital Games*), Juegos Electrónicos (*Electronic Games*) y Juegos de Computadora (*Computer Games*), siendo el Diseñador de Juego (*Game Designer*) el encargado de generar el concepto principal que servirá como base para el desarrollo del juego, generando un reto cognitivo que sea entretenido y divertido, además él guiará en las decisiones creativas y técnicas hasta tener el juego concretado.

Técnicamente los videojuegos están compuestos de dos partes principales: archivos de tipo Data y archivos en forma de Código.

Los archivos de tipo *Data* se refiere a todo lo que el Videojugador puede visualizar en pantalla y da forma característica a cada nivel de videojuego como son las imágenes, sonidos, textos, modelos en 3D y luces.

Los archivos en forma de *Código* se refieren a las instrucciones programadas por un desarrollador por medio de las cuales la computadora genera el mundo virtual y sus reglas, además son la base para que el videojuego funcione, siendo la columna vertebral del desarrollo.

A lo largo de la historia de los videojuegos los factores tecnológicos han ido modificándose y evolucionando a favor del jugador brindando experiencias de juego cada vez más enriquecedoras y satisfactorias, sin embargo los componentes fundamentales para que un videojuego pueda ser considerado como tal siguen siendo conformados por gráficos, sonidos, interfaces, *Gameplay*, mecánicas de juego e historia o contexto, siendo el *Gameplay* el factor principal de un videojuego.

13 Aranda, Daniel. (2018). *Game & Play: La cultura del juego digital*. 1a Ed. España. Ediciones Egregius. pp60.

Capítulo 2. Introducción a los Videojuegos

2.1 Definición de Videojuegos

Gráficos

Es cualquier imagen desplegada, así como cualquier tipo de efecto renderizado en el juego.

Incluye objetos 3d, mapas 2d, planos cinematográficos, *Full Motion Video (FMV)*, estadísticas, extractos de información y cualquier elemento que el jugador pueda observar.

Sonido

Cualquier música o efecto de sonido que es reproducido dentro del juego. Esto incluye música, FX, y sonido ambiental.

Interfaz

La interfaz se refiere a cualquier cosa que el jugador tenga que usar o tenga contacto directo para poder jugar el videojuego. Abarca desde el control físico del juego, ratón, teclado, joystick, hasta gráficos en los que el jugador deberá hacer click, sistema de menús donde navegará, sistemas de control de juego, etc.

Gameplay

Es la interacción que ocurre entre un jugador y un juego, engloba la capacidad de diversión que un juego aporta, la facilidad de uso, lo inmersivo que es y qué tanta jugabilidad tiene.

Mecánicas de Juego

Define los tipos de desafíos del juego; determina como el jugador interactúa y controla esos desafíos.

Incluye actividades cognitivas de nivel superior como resolución de problemas, planificación y aprendizaje, así como procesos cognitivos de nivel inferior como codificación y recuperación de memoria, pre-atención y procesamiento de información.

Historia o Contexto

Es la información que se muestra incluso antes de que el juego empiece, la información que el jugador descubre a lo largo del juego o en el final, así como información de los personajes y elementos en el juego.

Un punto notable de los videojuegos, es que, aunque son considerados como software de aplicaciones, tienen características propias que los diferencian del software tradicional ya que los softwares generales pasan por un ciclo de actualización donde la misma funcionalidad cambia o es mejorada con el tiempo, por ejemplo las versiones de Word, Excel, y las suites de adobe como Photoshop e Illustrator,

Capítulo 2. Introducción a los Videojuegos

2.1 Definición de Videojuegos

siendo las versiones antiguas olvidadas con el tiempo, mientras que los videojuegos al ser un producto creativo pueden ser usados independientemente de la época y continúan con su frescura, funcionalidad y objetivos originales con que fueron hechos, aplicando los términos *Retrogaming*, *Classic Gaming* y *Old School Games* a la actividad de jugar y coleccionar los clásicos.

Cada elemento de un videojuego es realizado por medio de las cualidades profesionales que un equipo creativo ofrece, la especialización de cada área es lo que caracteriza a cada videojuego volviéndolo destacable en el mercado, siendo dos elementos los imprescindibles para que un videojuego tenga éxito, definiendo éxito a la concepción de un juego divertido con buena jugabilidad y que consiga muchas ventas; estos son *Gameplay* y mecánicas de juego.

El *Gameplay* es fundamental porque aunque es nombrado generalmente como la experiencia agradable que da un juego al momento de jugar como facilidad y diversión en términos generales, también debe aplicarse a la forma en como el jugador manipula estrictamente los elementos interactivos, es decir la cantidad de control que el jugador tiene hacia el juego por medio de la respuesta en la coordinación mano-ojo, por ejemplo el tipo de salto del personaje principal, el golpe hacia enemigos, el uso de armas para enfocar un enemigo etc. este punto es muy importante porque es la forma de transmitir la vida del personaje del juego hacia el jugador y esa interacción es la que durará durante todo el juego, por lo que se espera que el personaje responda efectivamente a la coordinación motriz del input del usuario creando un fuerte enlace.

Se ejemplifica esto porque existen superproducciones que desarrollan juegos con grandes historias, gráficos excepcionales y gran publicidad y que aún con todas estas enormes cualidades fracasan en ventas, por una razón sencilla, dejan de lado el *Gameplay* con personajes e interacciones complejas de usar, teniendo un juego aburrido, difícil de jugar, relegando a segundo plano lo principal de un juego que es la jugabilidad. El juego *Super Mario 3D World* de Wii U es un ejemplo de muy buena jugabilidad en controles y como juego en general teniendo un gran *Gameplay*.



2.74 Super Mario 3D World

trucoteca.com. (2020). Super Mario 3D World [imagen]. Recuperado 25 Marzo 2020, de <https://f3.trucoteca.com/fotos/14123/super-mario-3d-world-1.jpeg>.

Capítulo 2. Introducción a los Videojuegos

2.1 Definición de Videojuegos

El segundo elemento esencial son las mecánicas de juego ya que se puede cubrir el lado de usabilidad por medio de que el personaje reaccione muy bien a los controles, añadiendo al juego una gran historia y gráficos pero si el jugador no tiene un elemento que lo haga pensar que le provoque un desafío mental, intelectual, que busque la solución a una problemática dentro del juego, el jugador sentirá que tiene en sus manos un producto difícil, no divertido que le produce dolores de cabeza en lugar de placer y diversión. La serie de juegos *The Legend of Zelda* y *Metal Gear Solid* son un ejemplo de grandes mecánicas de juego ejecutadas con maestría e ingenio.



2.75 Metal Gear Solid V

elchapuzasinformatico.com. (2020). Metal Gear Solid V [imagen]. Recuperado 25 Marzo 2020, de <https://elchapuzasinformatico.com/wp-content/uploads/2014/11/Metal-Gear-Solid-V-Ground-Zeroes-PC-1.jpg>.

Estos dos elementos siendo esenciales en videojuegos AAA pueden variar en porcentaje de importancia dependiendo del fin y la industria del producto, ya que por ejemplo un juego educativo se centrará principalmente en las mecánicas creando retos intelectuales por lo que no necesitará tanta cantidad de jugabilidad que un videojuego de acción; y en comparación, un juego de simulación necesitará en primer término de velocidad de respuesta motriz siendo esencial para representar una simulación fehaciente de la realidad, como por ejemplo un Serious Game que simule una operación del cuerpo humano destinado al entrenamiento de cirujanos.

Un punto importante que debe ser tomado en cuenta por entusiastas que empiezan en la creación de videojuegos, es el estudiar más a fondo las partes y las áreas de las que está compuesto un videojuego todo esto con el fin de tener un panorama más amplio que permita generar dentro de lo más posible un producto que sea divertido y usable.

Capítulo 2. Introducción a los Videojuegos

2.3 Historia de los videojuegos

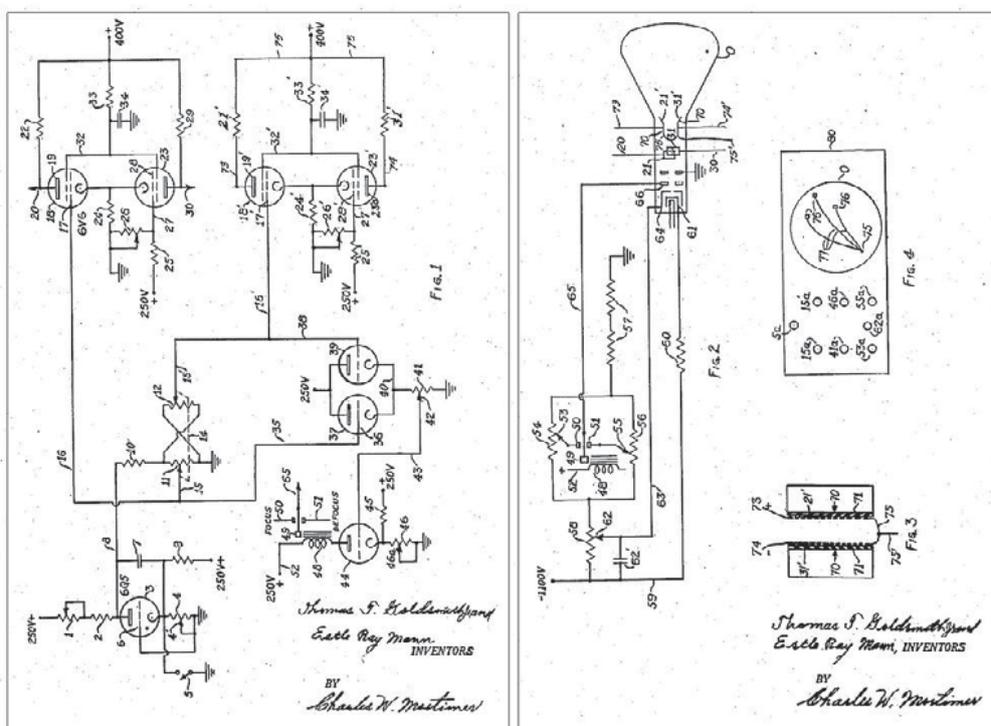
2.3 Historia de los videojuegos

Los videojuegos han existido durante décadas y abarcan toda la gama de plataformas, desde sistemas arcade, computadoras, hasta consolas domésticas, consolas portátiles y dispositivos móviles, por lo tanto, es una industria que suele estar a la vanguardia de la tecnología informática.

Inicios Interactivos - pioneros

En el año de 1947 en una época postguerra Thomas Goldsmith y Estle Ray Mann realizaron una patente en los Estados Unidos por lo que fue el primer juego interactivo llamado Dispositivo de entretenimiento con tubo de rayos catódicos, nunca fue construido.

En 1951 Ralph Baer fue nombrado como padre de los videojuegos tras marcar el concepto de videoconsola para el hogar y que en 1972 se convertiría en Magnavox Odyssey.



2.76 Cathode Ray Tube Amusement Device

wikimedia.org. (2020). Cathode Ray Tube Amusement Device [imagen]. Recuperado 25 Marzo 2020, de https://upload.wikimedia.org/wikipedia/commons/thumb/2/2d/Cathode_ray_tube_amusement_device_-_schematic.jpg/1024px-Cathode_ray_tube_amusement_device_-_schematic.jpg.

Capítulo 2. Introducción a los Videojuegos

2.3 Historia de los videojuegos

En 1951 Ralph Baer fue nombrado como padre de los videojuegos tras marcar el concepto de videoconsola para el hogar y que en 1972 se convertiría en Magnavox Odyssey.



2.77 Magnavox Odyssey

es.wikipedia.org. (2020). Magnavox Odyssey [imagen]. Recuperado 25 Marzo 2020, de <https://upload.wikimedia.org/wikipedia/commons/0/0f/Magnavox-Odyssey-Console-Set.png>.

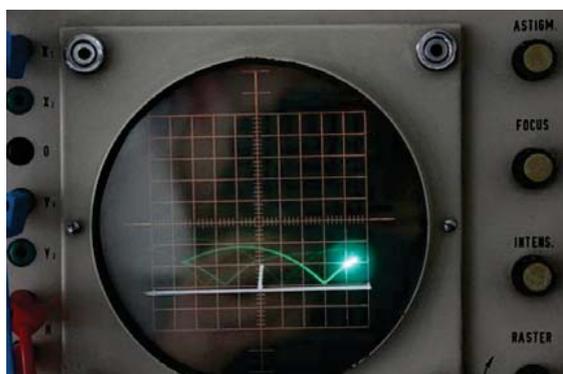
En 1952 Alexander S. Douglas crea “OXO” el juego de gato o tic-tac-toe, como tesis de Doctorado en Cambridge.



2.78 OXO

timetoast.com. (2020). OXO [imagen]. Recuperado 25 Marzo 2020, de <https://s3.amazonaws.com/s3.timetoast.com/public/uploads/photos/5483744/OXO.jpg>.

En 1958 es creado “Tennis for Two” por el físico Nuclear William Higginbotham, un precursor de Pong, que fue creado para instruir sobre los efectos de la gravedad.



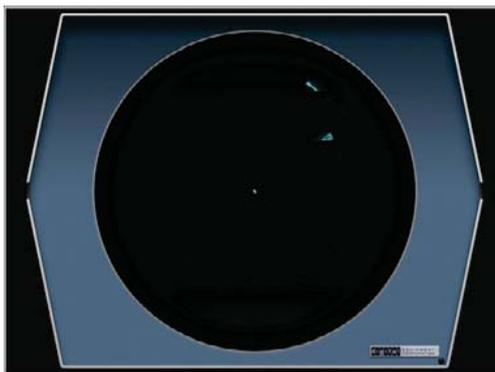
2.79 Tennis for Two

eulixe.com. (2020). Tennis for Two [imagen]. Recuperado 25 Marzo 2020, de https://www.eulixe.com/asset/zoomcrop,770,512,center,center/media/eulixe/imagenes/2018/10/26/tennis_for_two.png.

Capítulo 2. Introducción a los Videojuegos

2.3 Historia de los videojuegos

El gran momento para los videojuegos después de estos primeros desarrollos interactivos, se da en 1961 con el estudiante del MIT Steve Russell que crea Spacewar!, un juego con conceptos como partidas para 2 jugadores, misiles y sistema de puntaje.



2.80 Spacewar!

img.microsiervos.com. (2020). Spacewar! [imagen]. Recuperado 25 Marzo 2020, de <https://img.microsiervos.com/juegos/spacewar.jpg>.

Arcade

El inicio de los Arcade se da en 1971 cuando Nolan Bushnell con la ayuda del programador Ted Dabney producen Computer Space, la primera máquina de videojuegos echa en masa operada por monedas, con una producción inicial de 1500 gabinetes.



2.81 Computer Space

paleotronic.com. (2020). Computer Space [imagen]. Recuperado 25 Marzo 2020, de <https://paleotronic.com/wp-content/uploads/2018/10/computerspacecut-1.png>.

Un año después en 1972 fundan Atari y producen el icónico Pong.



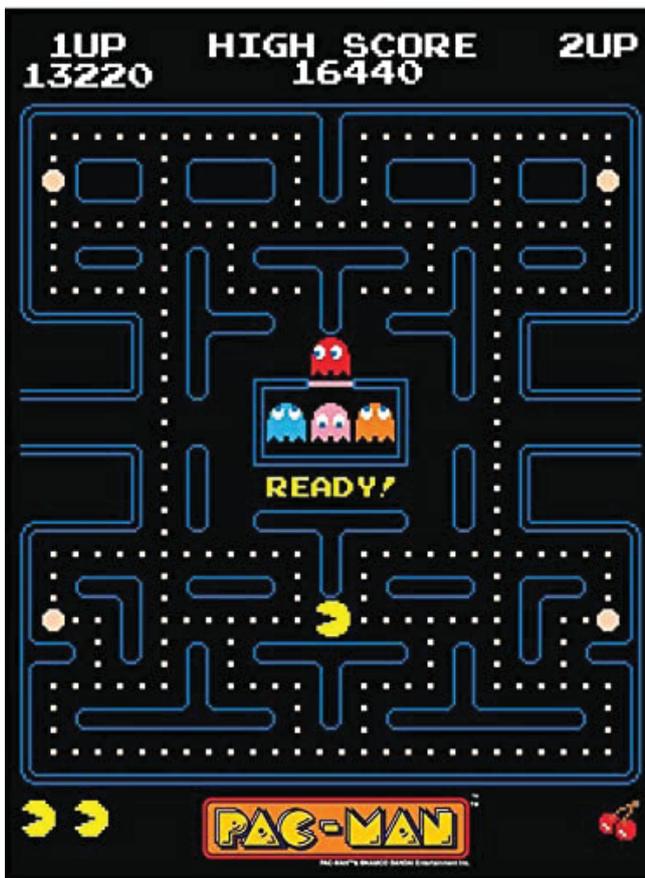
2.82 Pong

wikimedia.org. (2020). Pong [imagen]. Recuperado 25 Marzo 2020, de <https://upload.wikimedia.org/wikipedia/commons/thumb/2/26/Pong.svg/1280px-Pong.svg.png>.

Capítulo 2. Introducción a los Videojuegos

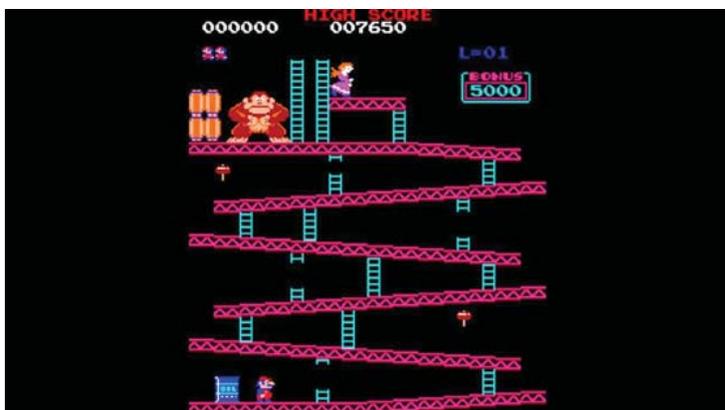
2.3 Historia de los videojuegos

Notables juegos arcades de esa época fueron: Breakout (1976) Space Invaders (1978), Asteroids (1979), Pac-Man (1980) Donkey Kong (1981).



2.83 Pac-Man

amazon.com. (2021). Pac-Man [imagen]. Recuperado 3 September 2021, de https://m.media-amazon.com/imagens/I/61D0gY-JH5OL._AC_UX385_.jpg.



2.84 Donkey Kong

Youtube.com. (2020). Donkey Kong [imagen]. Recuperado 25 Marzo 2020, de <https://i.ytimg.com/vi/KJkcNP4VkiM/maxresdefault.jpg>.

Capítulo 2. Introducción a los Videojuegos

2.3 Historia de los videojuegos

Consolas Caseras

La primera consola casera se le atribuye Ralph Baer en 1972 con la Magnavox Odyssey.

Consolas rivales de la época fueron *Coleco Telstar*, *Channel F* (con primeros cartuchos) *Mattel Intellivision*, hasta que una de las más importantes llegó al mercado; en 1977 *Atari Video Computer System (VCS)* o también llamada Atari 2600, primera consola en tener gran éxito popularizando el uso de cartuchos intercambiables. Luego en 1980 esta consola trajo consigo el primer *port* de videojuego, de arcade a consola con el juego *Space Invaders*.



2.85 Atari 2600

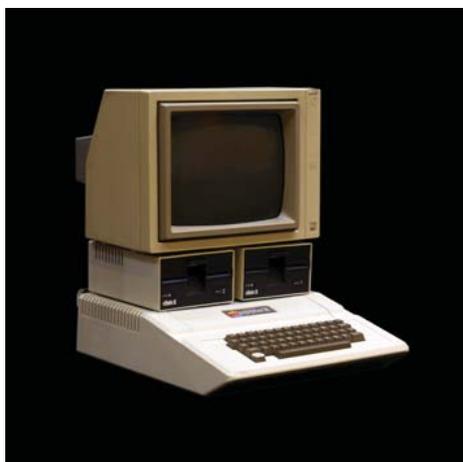
wikimedia.org. (2020). Atari 2600 [imagen]. Recuperado 25 Marzo 2020, de <https://upload.wikimedia.org/wikipedia/commons/b/b9/Atari-2600-Wood-4Sw-Set.jpg>.

La revolución de Computadores personales

Durante la mitad de los 70 otro segmento de la industria empezó su desarrollo; el de las computadoras personales, que trajo tecnología al hogar de uso exclusivo para programadores y hobbistas, siendo así que muchos juegos arcade se adaptaran para estas plataformas. Computadoras notables fueron: La Apple II (con juegos como Zork, Lode Runner, Wizardry, Ultima) Macintosh, Atari 400, 800 y 5200, BBC Microcomputer (1981), Acorn's Electron (1983), ZX80 (1980), ZX81 (1981), ZX Spectrum (1982), Commodore 64 (1982), amiga (1985).

Capítulo 2. Introducción a los Videojuegos

2.3 Historia de los videojuegos



2.86 Apple II

wikimedia.org. (2020). Apple II [imagen]. Recuperado 25 Marzo 2020, de https://upload.wikimedia.org/wikipedia/commons/5/57/Apple_II_IMG_4218-black.jpg.



2.87 Commodore 64

interface-experience.org. (2020). Commodore 64 [imagen]. Recuperado 25 Marzo 2020, de <https://interface-experience.org/site/wp-content/uploads/2015/01/IE-008KKEDIT.jpg>.

Crisis del videojuego

El boom de los videojuegos de los principios de los 80s no duró mucho tiempo ya que la industria pasó por el llamado “The great videogame crash” donde el mal manejo de la compañía de Atari fue el culpable de llevar a la quiebra a varias empresas desarrolladoras de videojuegos, una de las razones principales fue el bajo nivel de calidad de videojuegos que se producían, ejemplos de estos juegos: Custer’s Revenge y E.T. The Extra Terrestrial.

Las ventas bajaron, había sobreoferta llegando a pensar que la industria de los videojuegos era pasajera.



2.88 E.T. The Extra-Terrestrial

Youtube.com. (2020). E.T. the Extra-Terrestrial [imagen]. Recuperado 25 Marzo 2020, de https://i.ytimg.com/vi/8TcbJ4Y9_38/maxresdefault.jpg.

Capítulo 2. Introducción a los Videojuegos

2.3 Historia de los videojuegos

Los videojuegos Japoneses

Se pensó en US que la industria del videojuego estaba completamente muerta después del crash de 1983, pero esta idea no fue compartida en japon o el resto del mundo, que veían con buenos ojos la inversión en videojuegos y es así, que Nintendo saca su primera consola en 1983, la 8 bit *Famicom* (abreviado *Family Computer*) un sistema que cambiaría la industria para siempre.

Para 1984 3 millones de consolas son vendidas, con la característica de que sus juegos solo se pueden jugar exclusivamente en su consola, restringiendo las licencias a otras compañías.

Para 1985 Nintendo continuó expandiéndose en el mercado y lanza en USA el *Nintendo Entertainment System (NES)* manteniendo un estricto control sobre los juegos producidos al público, con cartuchos antipiratería.

Super Mario Bros, uno de los juegos más influyentes de la cultura actual se hizo para esta consola en 1985, otros éxitos son: *Legend of Zelda*, *Metroid*, *Castlevania*.

También en el mismo día del lanzamiento del NES salió al mercado la primera consola de SEGA la SG-1000.

En esta generación el *gamepad* o *joypad* con su diseño de 8 dirección marcó un estándar.

Muchas otras consolas salieron en esa década consolidando el mercado de los videojuegos entre ellas: *Sega Master System (1987)*, *Atari 5200 (1982)*, *Atari 7800 (1986)*.

El *Game Boy* sacado en Japon y USA en 1989 fue el primer portátil en cambiar cartuchos y en dar a luz a uno de los juegos más populares de la historia: Tetris, vendiendo 33 Millones de copias.



2.89 Super Mario Bros

businessinsider.com. (2020). Super Mario Bros [imagen]. Recuperado 25 Marzo 2020, de <https://i.insider.com/560e-bbe7dd0895325c8b458e?width=1100&format=jpeg&auto=webp>.

Capítulo 2. Introducción a los Videojuegos

2.3 Historia de los videojuegos



2.90 FAMILICOM

wikimedia.org. (2020). FAMILICOM [imagen]. Recuperado 25 Marzo 2020, de <https://upload.wikimedia.org/wikipedia/commons/4/4c/Nintendo-Famicom-Console-Set-FL.png>.

La generación de 16 Bits empezó al final de los 80s y diversas consolas salieron al mercado: *TurboGrafx-16* (1989), *Mega Drive/Genesis* (1990), *Super Nes* (1991), *Neo Geo* (1990) etc.

Es en esta época cuando Nintendo empieza a experimentar con tecnología de formatos de medios ópticos para el SNES en conjunto con SONY, pero este proyecto nunca saldría para el SNES y en cambio se transformaría en el PlayStation, proyecto que para SONY sería muy redituable, como en la época del walkman.



2.91 Nintendo PlayStation

Nintendo PlayStation [imagen]. Recuperado 25 Marzo 2020, de https://i.blogs.es/6042ca/nintendo-playstation-2/1366_2000.jpeg.

Empieza la guerra de consolas principalmente por SEGA (*Mega Drive/Genesis*) con *Sonic The Hedgehog* y Nintendo (SNES) con *Mario*.

La Neo-Geo (24bits) fue la consola más cara de su generación, pero es cierto que también ofrecía los mejores gráficos 2D de su época ya que contenía el mismo hardware que usaban los arcades de SNK.

Los arcades tuvieron un pequeño periodo de ascenso nuevamente por el lanzamiento de *Street Fighter II* por parte de *Capcom*, *Mortal Kombat* y *The King of Fighters* entre otros.

Capítulo 2. Introducción a los Videojuegos

2.3 Historia de los videojuegos



2.92 Mega Drive

wikimedia.org. (2020). Mega Drive [imagen]. Recuperado 25 Marzo 2020, de <https://upload.wikimedia.org/wikipedia/commons/a/a1/Sega-Mega-Drive-JP-Mk1-Console-Set.jpg>.



2.93 SNES

wikimedia.org. (2020). SNES [imagen]. Recuperado 25 Marzo 2020, de <https://upload.wikimedia.org/wikipedia/commons/3/36/SNES-Mod1-Console-Set.png>.



2.94 Neo-Geo

wikimedia.org. (2020). Neo-Geo [imagen]. Recuperado 25 Marzo 2020, de <https://upload.wikimedia.org/wikipedia/commons/8/8e/Neo-Geo-AES-Console-Set.png>.

Transición al mundo de 3D (1990)

Fue a la mitad de la década de los noventa que surgieron innovaciones para la industria, los desarrolladores se beneficiaron de más colores y velocidades de procesamiento más rápidas, se pasó de juegos rasterizados a gráficos 3d con polígonos, capaces de producir gráficos en 3d mapeados con textura y qué decir del formato CD que gracias a él se podía incluir *full motion video*, mundos más grandes y sonido más claro.

Los principales lanzamientos fueron Sega Saturn (32-bits - 1995), Playstation (32-Bits - 1994) y Nintendo 64 (64-bits - 1996)

Capítulo 2. Introducción a los Videojuegos

2.3 Historia de los videojuegos



2.95 Playstation

wikimedia.org. (2020). Playstation [imagen]. Recuperado 25 Marzo 2020, de <https://upload.wikimedia.org/wikipedia/commons/9/95/PSX-Console-wController.png>.



2.96 Nintendo 64

wikimedia.org. (2020). Nintendo 64 [imagen]. Recuperado 25 Marzo 2020, de <https://upload.wikimedia.org/wikipedia/commons/0/02/N64-Console-Set.png>.

Se funda la ESRB en 1994 para crear categorías de juego de acuerdo al contenido.

El Game Boy continuó su reinado a lo largo de la década con sus portátiles Game Boy Pocket, Game Boy Color y Game Boy Advance hasta que en 2004 se lanza el Playstation Portable (PSP) para competir con el Nintendo DS.



2.97 Game Boy

es.wikipedia.org. (2020). Game Boy [imagen]. Recuperado 25 Marzo 2020, de <https://upload.wikimedia.org/wikipedia/commons/f/f4/Game-Boy-FL.jpg>.

Capítulo 2. Introducción a los Videojuegos

2.3 Historia de los videojuegos

Capacidades Multimedia

Casi en los finales de los 90 y principios de los 2000 la guerra de consolas era a muerte y en esta batalla se une un nuevo gigante: el *Xbox* de *Microsoft* (2001) mientras tanto *Sony* lanza al mercado su consola *Playstation 2* (2000) que demuestra que no ofrece entretenimiento solo a pequeños, ya que su promesa es vender sistemas de entretenimiento todo en uno, como películas, música e internet.

La *Dreamcast* de *SEGA* (1998) y *Nintendo GameCube* (2001) fueron consolas de esa generación.



2.98 Playstation 2

Peter Warman. (2019). Global Games Market Report 2019 [Ebook] (1st ed., p. 29). Recuperado 30 Abril 2020, de <https://newzoo.com/insights/trend-reports/newzoo-global-games-market-report-2019-light-version/>



2.99 Xbox

wikimedia.org. (2020). Xbox [imagen]. Recuperado 25 Marzo 2020, de <https://upload.wikimedia.org/wikipedia/commons/c/c2/Xbox-Console-Set.png>.

De lleno en la Era Digital

Señaladas como consolas de “*Next-Gen*”, *Xbox 360* (2005), *Ps3* (2006) *Wii* (2006) lograron avances en el almacenamiento digital con una mayor capacidad y discos extraíbles, una calidad de imagen asombrosa en HDTV gracias al Blu-ray, Dolby Digital, acceso a banda ancha de internet por Wi-fi o modem, además de las capacidades multimedia ya estandarizadas para la época y nuevas funciones, ahora podrías hacer llamadas, grabar tv, habitar mundos virtuales online, etc.

Cabe mencionar que el *Wii* aún al ser de menor potencia en comparación a *Xbox 360*

Capítulo 2. Introducción a los Videojuegos

2.3 Historia de los videojuegos

y Ps3 volvió a Nintendo a las ventas, ya que poseía un novedoso sistema de control de movimiento, ya que cualquier persona podría fácilmente manipular y divertirse con su nuevo sistema.



2.100 Xbox 360

backmarket.es. (2020). Xbox 360 [imagen]. Recuperado 25 Marzo 2020, de https://d1eh9yux7w8iql.cloudfront.net/product_images/1520930481.85.jpg.



2.101 Playstation 3

upload.wikimedia.org. (2020). Playstation 3 [imagen]. Recuperado 25 Marzo 2020, de <https://upload.wikimedia.org/wikipedia/commons/d/d3/PS3Versions.png>.



2.102 Wii

eluniversal.com.mx. (2020). Wii [imagen]. Recuperado 25 Marzo 2020, de https://www.eluniversal.com.mx/sites/default/files/2020/01/27/nintendo_consola_wii_marzo_2020.jpg.

Capítulo 2. Introducción a los Videojuegos

2.3 Historia de los videojuegos

Consolas de 8va generación

La 8va generación incluye a: *Nintendo 3DS* (2010), *Playstation Vita* (2011), *WiU* (2012), *Ps4* (2013), *Xbox One* (2013), *Switch* (2017) con servicios de compras digitales y microtransacciones, gráficos nunca antes vistos, acceso a APPS de diversos contenidos, son consolas con gran poder de procesamiento (en el caso de las consolas de sobremesa), con gráficos FULL HD y en versiones especiales 4K además de motion control estándar en controles; se piensa que son competidoras directas contra smartphones, tabletas y Smart TV y sus juegos casuales *Freemium* que llenan el mercado.



2.103 Nintendo 3DS

dexerto.es. (2020). Nintendo 3DS [imagen]. Recuperado 25 Marzo 2020, de https://imagenes.dexerto.es/uploads/thumbnails/_thumbnailLarge/nintendo-3ds.jpg.



2.104 Xbox One

sanborns.com.mx. (2020). Xbox One [imagen]. Recuperado 25 Marzo 2020, de https://www.sanborns.com.mx/imagenes-sanborns-ii/1200/889842529036_2.jpg.

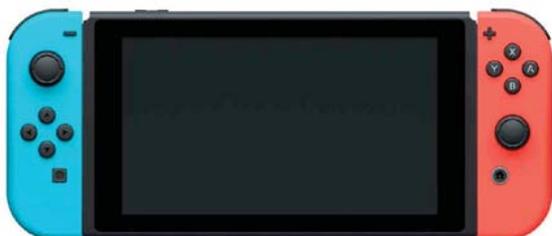


2.105 Playstation 4

wikimedia.org. (2020). Playstation 4 [imagen]. Recuperado 25 Marzo 2020, de <https://upload.wikimedia.org/wikipedia/commons/7/7e/PS4-Console-wDS4.jpg>.

Capítulo 2. Introducción a los Videojuegos

2.3 Historia de los videojuegos



2.106 Switch

resources.claroshop.com. (2020). Switch [imagen]. Recuperado 25 Marzo 2020, de https://resources.claroshop.com/medios-plazavip/fotos/productos_sears1/original/2977765.jpg.

Consolas de 9na generación

Esta generación actual incluye a *PlayStation 5* de Sony y *Xbox Series X|S* de Microsoft.



2.107 Xbox Series X

mlstatic.com. (2021). Xbox Series X [imagen]. Recuperado 3 September 2021, de https://http2.mlstatic.com/D_NQ_NP_850214-MLA45046473101_032021-O.webp.



2.108 Playstation 5

amazon.com. (2021). Playstation 5 [imagen]. Recuperado 3 September 2021, de https://m.media-amazon.com/images/I/41xe-0HXlgML_AC_SL1000_.jpg.

No se sabe a ciencia cierta si esta generación será la última generación de consolas de formato físico ante lo que viene a futuro con el Cloud Gaming, con servicios como Stadia, Playstation Now, Apple Arcade, donde se puede jugar desde cualquier dispositivo conectado a internet.

Capítulo 2. Introducción a los Videojuegos

2.3 Historia de los videojuegos



2.109 Apple Arcade

tuinformaticapersonal.es. (2020). Apple Arcade [imagen]. Recuperado 25 Marzo 2020, de <https://www.tuinformaticapersonal.es/wp-content/uploads/2020/01/los-cinco-mejores-juegos-de-apple-arcade-7.jpg>.



2.110 Playstation Now

xataka.com.mx. (2020). Playstation Now [imagen]. Recuperado 25 Marzo 2020, de https://i.blogs.es/f6502e/psnow/840_560.jpg.



2.111 Stadia

expansion.mx. (2020). Stadia [imagen]. Recuperado 25 Marzo 2020, de <https://cdn-3.expansion.mx/dims4/default/bfc8562/2147483647/strip/true/crop/716x472+0+0/resize/1800x1187!/quality/90/?url=https%3A%2F%2Fcdn-3.expansion.mx%2F%2F8c%2F6d5fdd5149ea90b9413ff76e0460%2Fstadia.PNG>.

Capítulo 2. Introducción a los Videojuegos

2.3 Historia de los videojuegos

Como se ha visto los videojuegos son una industria tecnológica en la que se invierten millones de dólares y que en ocasiones el éxito económico por medio de retorno de inversión no está asegurado, esto hace que el éxito se deba a muchos factores como competencia en el mercado con sistemas y especificaciones técnicas parecidas, posicionamiento de la marca en los jugadores, lo fácil o difícil que puede ser desarrollar para determinada plataforma, la calidad de los videojuegos, la piratería que afecta determinadamente los ingresos de las empresas, el *marketing* de promoción de un juego, así como procesos económicos globales que afecten la producción de un país incrementando el precio de venta e impuestos para consolas y juegos.

Sin embargo, la industria siempre se ha mantenido en la mente de los consumidores gracias a que ha sabido adaptarse al mercado de *Gamers*, creando nuevas estrategias y formas de consumo - como juegos *Freemium*- y buscando en primer lugar la innovación interactiva que las experiencias de sus productos generen generación tras generación.

También, un punto remarcable dentro del sistema productivo de los videojuegos es la preservación y conservación de videojuegos a lo largo del tiempo, un tema sumamente importante ya que los videojuegos al ser productos de software que son lanzados al público por miles cada generación, corren el riesgo de llegar a perderse u olvidarse en unos cuantos años ya que los soportes magnéticos como *Cassettes*, *disquetes*, HDD o soportes ópticos como CD-ROM, DVD-Rom entre otros, se deterioran con el paso del tiempo y también en ocasiones si no se tiene la consola específica donde corre determinado juego al quedar obsoleta, es imposible jugarlo; en este sentido debe destinarse tiempo y esfuerzo a la preservación de los videojuegos ya que se trata de un bien cultural humano donde se plasma el grado de desarrollo tecnológico e intelectual de la época.

Una de las formas de preservación es el uso de emuladores que son programas que una vez instalados en la computadora pueden correr los videojuegos en formato digital, ripeados del juego original emulando la consola original, llegando a encontrarse en internet el full Rom set de cada consola, es decir todos los juegos de determinada consola en un archivo comprimido rar.

Es cierto también que la revisión documental de la historia de los videojuegos permitirá a los nuevos creadores tener un panorama efectivo de comparación y reflexión para el momento de la producción.

Capítulo 2. Introducción a los Videojuegos

2.4 Fines de los videojuegos

2.4 Fines de los videojuegos

Los videojuegos tienen en común varios aspectos como son el presentar un mundo virtual donde el usuario pueda sentirse inmerso y por medio de la interacción dentro de este mundo ficticio cambiar su percepción de la realidad, sin embargo serán los objetivos que el producto interactivo tenga como fin, los que propiciarán la forma en cómo se diseña y presenta al usuario final.

Aunque los videojuegos en su formato de entretenimiento son el sector más desarrollado en la industria por la cantidad de retorno de inversión, el surgimiento de los *Game Engines* ha posibilitado que otros sectores como el sector educativo, el sector de simulaciones y el sector artístico, salgan beneficiados al tener la posibilidad de crear sus propios productos interactivos con presupuestos reducidos.

Se presentan a continuación los principales fines que los videojuegos cumplen donde lo espectacular y detallado de un videojuego -o también llamado producto interactivo- puede variar de acuerdo a los objetivos y presupuestos de producción.

Entretenimiento

Así como los contenidos musicales, televisivos, cinematográficos, editoriales los videojuegos cumplen también el objetivo de entretener con historias diversas donde el jugador puede formar parte, llegar a identificarse y hacer empatía con el concepto del producto.

Interacción social

El poder jugar un videojuego con características LAN u Online teniendo interacción social con otros jugadores.

Un ejemplo dominante son los juegos ocasionales sencillos llamados Casual Games de la plataforma social Facebook en donde se puede compartir avances, así como jugar contra otros amigos.

Educacional

Los juegos educativos son pensados principalmente para un público infantil como una alternativa a la enseñanza tradicional, utilizando las formas vistosas, herramientas interactivas y estímulos propios de las experiencias nuevas que proponen los videojuegos, desarrollando capacidades intelectuales y de aprendizaje del jugador.

Reclutamiento y capacitación (Serious Games)

Los videojuegos son usados en estas áreas como una forma de transmitir conocimiento por medio de la interacción formulada por situaciones reales.

También llamados juegos de simulación, en estos juegos se replican las situaciones

Capítulo 2. Introducción a los Videojuegos

2.4 Fines de los videojuegos

que se vivirían en un ambiente común; el usuario toma decisiones simulando su ambiente referenciado o también tomando en cuenta y razonando situaciones que aún no pasan o son difíciles de replicar en la realidad.

Salud y entrenamiento físico

Estos juegos desarrollan un balance entre mente y cuerpo usándose en rehabilitación, terapia psicológica y fortalecimiento y mejora de la salud por ejercicio físico funcionando también para personas del sector salud que simulan los procesos quirúrgicos aumentando la tasa de éxito.

Los juegos destinados a personas con deficiencias físicas y cognitivas entran en esta categoría por medio del término juegos accesibles; un producto destinado a la accesibilidad de los videojuegos es el *Xbox Adaptive Controller*.

Conciencia social

Los juegos dentro de esta categoría promueven por medio de la interacción una acción de cambio en base a promover temáticas de derechos humanos, pobreza, conflictos globales, cambio climático etc.

Estética y creatividad

Son juegos que contienen herramientas artísticas o que muestran una carga estética y que promueven creatividad y emociones.

Marketing y Advertising

Estos juegos son pensados para promover productos o servicios a un público por medio de marketing.

En ocasiones estos juegos llamados Advergames forman parte dentro de un gran rompecabezas donde intervienen muchos formatos de publicidad y la marca o lo anunciado tiene que ser posicionado a través de experiencias interactivas novedosas, siempre manteniendo en primer lugar la imagen de la marca.

Capítulo 2. Introducción a los Videojuegos

2.5 Géneros

2.5 Géneros

Los géneros son las categorías específicas en las cuales la industria clasifica a los videojuegos producidos de acuerdo al *Gameplay* y al tipo de retos que presenta un juego, esta clasificación no necesariamente tiene que ver con el tipo de gráficos, historia o la temática del juego, por ejemplo en el juego *Battlefield V* (2018) se toma el rol de un soldado ambientado en la segunda guerra mundial, en *Doom* (2016) está ambientado en el planeta Marte tomando el rol de un marine espacial y *Overwatch* (2016) es ambientado en el futuro de la tierra en un entorno robótico, todos ellos pertenecen al género First Person Shooter donde tienen en común el tipo de interacción que se basa en el uso de armas con capacidades de disparar proyectiles para la eliminación de enemigos, todo generado por medio de la vista en primera persona del personaje que controla el jugador.

La industria de los videojuegos tiene muy bien estudiados y segmentados a sus consumidores, de esta forma cada género tiene destinado un público muy definido al que le crean experiencias únicas ya que existen Gamers que no soportan juegos violentos como el género de peleas o disparos y son Videojugadores de deportes o puzzles, así como hay jugadores que tienen preferencias de juegos de acción - aventura sobre *puzzles*.

En el momento de desarrollar un juego el género es el principal elemento a tener en cuenta, ya que se tiene que pensar si el mercado al que va dirigido el juego tendría buena recepción de él, si el mercado no está saturado con juegos iguales o parecidos y cuáles serían las cualidades por las que destacaría este nuevo desarrollo sobre sus competidores.

Además, se tiene que contemplar los gastos tecnológicos y profesionales de desarrollo y lo complejo que el juego vaya a ser, porque un juego de *puzzles* como *Candy Crush* con un costo de desarrollo de \$12 000 dólares es mucho más barato de realizar en comparación a un juego de acción - aventura de mundo abierto como *GTA V* con un costo de desarrollo de \$256 millones de dólares.

Al ser una industria que está siempre a la vanguardia, los desarrolladores continuamente crean o mezclan géneros para presentar mejores experiencias interactivas de juego.

A continuación, se muestran los principales géneros que la industria produce.

Acción (Action)

Uno de los géneros más antiguos; es relacionado con los juegos Arcade, son juegos a menudo simples ya que su mecánica consiste en que el jugador se mantenga vivo el mayor tiempo posible tomando ventaja a enemigos y destruyéndolos, son juegos

Capítulo 2. Introducción a los Videojuegos

2.5 Géneros

donde la coordinación motriz es muy necesaria ya que el tiempo de reacción debe ser corto.

Los subgéneros de Acción son:

Plataformas (Platformers)

El jugador controla un personaje que camina, salta y colecciona ítems, mata enemigos y se mueve en un entorno 2d o 3d lleno de obstáculos

Shooters

Este género se basa en el combate entre el jugador y enemigos humanos o NPC, principalmente con diversas armas y disparos.

Existen el **First Person Shooter (FPS)** donde la perspectiva es desde el jugador en primera persona y sólo salen sus brazos y manos, y el **Third Person Shooter**, donde sale por completo en pantalla.

Carreras

Utilizando vehículos, el fin es llegar en primer lugar o realizar el menor tiempo posible en pistas de carreras.

Peleas

Son juegos donde se controla un jugador que pelea en contra de uno o varios rivales, utilizando golpes específicos.

Aventura

Son juegos que se basan en la exploración de entornos, las libertades de movimiento y decisión son sus principales características, aunque también están presentes el coleccionar ítems y resolver acertijos.

Acción aventura

Es un género híbrido, con velocidad y diversión que generan los juegos de acción, en conjunto con elementos de puzzles y acertijos.

Gambling (casino)

Versiónes electrónicas de los populares juegos: ruleta, poker, black jack, craps, máquinas tragamonedas, pueden ser online o en lugares físicos.

Puzzle

Los juegos *Puzzle* consisten en resolver una serie de rompecabezas sin controlar un personaje, con diversas dificultades en contrarreloj, así como contra personas o NPC.

Capítulo 2. Introducción a los Videojuegos

2.5 Géneros

Juegos de rol (RPG)

Los *Role Playing Games*, nacieron en los 70's con *Dungeons & Dragons* y desde ese momento se ha hecho uno de los géneros favoritos de los jugadores; la mecánica principal es tomar el rol de personajes como guerreros, magos, sacerdotes y elfos y luchar contra monstruos y dragones entre otros, en calabozos.

Este género sobresale porque sus juegos contienen grandes historias fantásticas donde se detalla la experiencia del personaje que se ha elegido, sus emociones, sus logros, etc.

Sus partidas pueden ser por turnos.

Simulaciones

En las simulaciones se trata de reproducir sistemas, máquinas y experiencias fieles a la realidad.

Simulación de vehículos

Se trata de operar meticulosamente el control de vehículos complejos.

Deportes

El jugador forma parte de la experiencia que presencia en su deporte favorito, pudiendo ser entrenador o atleta.

Procesos

También llamados *Toy* o *God Games* consisten en que el jugador cree, controle y mantenga todos los procesos de un determinado sistema, grupo, construcción, ciudad, etc.

Estrategia

Basándose en juegos de mesa, retoma el funcionamiento de tomar decisiones de acuerdo a los recursos limitados con los que se cuenta.

Turn Based Strategy

Se trata de juegos por turnos donde se da tiempo a que el jugador piense la acción a realizar.

Real Time Strategy

Llenos de acción frenética este género se ha convertido en uno de los favoritos por los jugadores de la década ya que los enfrentamientos dentro del juego son de un jugador o *multiplayer* y tienen el fin de administrar los recursos de pelea y estrategias para lograr la victoria, todo en tiempo real.

Massively Multiplayer Online Games (MMOG)

Este tipo de juegos incluyen miles de jugadores simultáneamente por medio de

Capítulo 2. Introducción a los Videojuegos

2.5 Géneros

conexiones a internet y pueden ser massively multiplayer online role-playing games (MMORPGs), massively multiplayer first-person shooters (MMOFPS), massively multiplayer real-time strategy games (MMORTS), multiplayer online battle arena (MOBA) y Battle royale game.

La acción es constante y son sumamente adictivos ya que puedes socializar con otros jugadores lo que produce una gran inmersión y diversión.

Capítulo 2. Introducción a los Videojuegos

2.6 Plataformas

2.6 Plataformas

Las plataformas son los dispositivos electrónicos en donde se pueden jugar los videojuegos, además las plataformas hacen uso de diferentes dispositivos que facilitan el intercambio de información creando interacción entre la computadora y el usuario, estos son:

el input por medio de mandos o controles de juego por los que el jugador toma control del personaje u objetos del videojuego dentro del entorno virtual

la consola principal que es el hardware destinado a leer los juegos físicos o digitales y es la encargada de realizar los cálculos de cómputo para generar el juego mismo.

La unidad de *output* dividida en *display* y audio, donde el *display* utiliza una pantalla que visualiza lo que pasa en el videojuego y es a través de esta que el jugador recibe información sobre el videojuego en forma de gráficos y audio donde las experiencias sonoras del juego son percibidas por el jugador en forma de ondas sonoras por medio de altavoces.

Las diversas plataformas son concebidas para diferentes fines y funcionalidades, como por ejemplo la portabilidad que dan los juegos casuales para smartphones, las experiencias realistas en simulación que un *PC Gaming* de alta gama brinda o las experiencias de entretenimiento en el centro multimedia que brinda el PS4, por esta razón los juegos también cambian dependiendo de la plataforma.

También existe el tema de la exclusividad de juegos, donde el desarrollador sólo publica su juego para determinada plataforma creando la necesidad en el público de comprar la consola para jugar su exclusiva, esta forma de negocios ha dado mucho éxito en la industria con la explotación de grandes franquicias conocidas, como *Mario Bros* por parte de *Nintendo*, *Gears of War* por parte de *Xbox* y *God of War* por parte de *Playstation*.

Se tiene que mencionar que para que una plataforma sea exitosa en el mercado, debe de tener las prestaciones en cuanto a accesibilidad, facilidad y soporte para que las comunidades de desarrolladores de videojuegos puedan crear en ellas de forma positiva y no tengan dificultades o restricciones que podrían generar un detrimento en la calidad de juego o en la poca producción de juegos para cierta plataforma.

En la actualidad existen varias plataformas para jugar, las más populares son Computadoras Personales (PC), Consolas de videojuegos y dispositivos móviles, sin embargo parece que existirá un gran cambio en la forma de cómo se juega ya que en 2019 Google lanzó *Google Stadia* lo que parece el futuro de los videojuegos,

Capítulo 2. Introducción a los Videojuegos

2.6 Plataformas

con su infraestructura en la nube, esto significa que se puede jugar cierto título por medio de internet en cualquier dispositivo compatible, ya sea un smartphone, un pc, una Tablet o una consola, ofreciendo la misma calidad y experiencia de juego en cualquiera de estos dispositivos; cabe mencionar que aunque ya está en el mercado sólo ciertos países pueden acceder a este servicio y los requerimientos de velocidad de internet son muy altos.

Es importante mencionar que en la actualidad los *Game Engines* cuentan con capacidades multiplataforma, esto quiere decir que en ellos se desarrolla el juego y el mismo se puede exportar para diversas plataformas como Windows, Android, iOS, PS4, Switch entre otras, con pequeños ajustes de compatibilidad, pero conservando el mismo juego en varios sistemas.

A continuación, se describirán las principales plataformas de videojuegos.

Consola

Son sistemas de cómputo adaptados para poder utilizar su propio software licenciado en forma de juegos en formato físico o descarga digital, están enfocados para un consumo en el hogar y soportan varios controles y accesorios que han sido configurados exclusivamente para su respectiva consola.

De la novena generación de consolas forman parte: PlayStation 5 de Sony y Xbox Series X|S de Microsoft.

Los juegos de consola tienen una ligera variación de sistema en sistema esto dependiendo de la capacidad gráfica y de procesamiento de cada una, y estas diferencias se notan en gráficos con mejor resolución (4k) texturas más definidas e interacción más fluida, también influye la forma en como fueron programados por los desarrolladores.

Algunos juegos son multilicencia es decir que pueden aparecer para varias consolas, así como hay exclusivas para una consola. Actualmente todas las consolas tienen capacidades de navegación en internet.

Computadora PC

La computadora tiene la particularidad de no ser un *hardware* propietario para los videojuegos, este le da varias diferencias con consolas, la primera característica es que existe una gran variación de *settings*, así también se pueden generar resultados mucho más satisfactorios a nivel gráfico y de procesamiento ya que un computador puede ser varias veces más poderoso y escalar el videojuego llevándolo a otros niveles de calidad.

Capítulo 2. Introducción a los Videojuegos

2.6 Plataformas

Los juegos en computadora generalmente se juegan con teclado y mouse, aunque también existen una gran variedad de controles, así como accesorios y adaptadores. Otra ventaja es que muchos juegos antiguos pueden correr en un computador no poderoso siendo una alternativa divertida de bajo costo.

Los principales sistemas operativos donde corren los videojuegos para Computadora son Linux, Mac OS X y Windows.

Handled

La portabilidad es el principio que gobierna a estas consolas, que han sabido perdurar con el paso del tiempo, desde la época de las primeras portátiles de los 80 como *Atari Lynx* y *TurboExpress*, pasando por los 90 con *Game Boy* y *Neo Geo Pocket* y manteniéndose en los 2000 con *Nintendo DS* y *Playstation Portable*, ya en los 2010 con *Nintendo 3ds* y *Playstation Vita* hasta el nuevo híbrido que Nintendo empezó a producir en 2017 llamado Switch, que logra mezclar gráficos avanzados en Full HD para TV y también tener acceso a su modo portátil donde se puede jugar en cualquier lugar.

Smartphones

Otro mercado en auge que lleva años creciendo gracias al desarrollo de apps es el de los móviles, que aparte de ser un canal principal de comunicación, se pueden usar contenidos interactivos en sus sistemas, además conforme pasa el tiempo y se mejora el hardware las capacidades multimedia de estos van aumentando y es así que en celulares de gama media se pueden correr juegos en 3d con gráficas semejantes a la generación de ps2-ps3, y qué decir de lo último en tecnología donde están los teléfonos dedicados al *Gaming* con juegos de última generación como *PUGB mobile*, *Fortnite Battle Royale*, *Pokemon GO*, *Monument Valley*, etc.

Ante el conjunto de plataformas existentes en la actualidad, el creador de videojuegos tiene una gran gama de opciones para desarrollar su idea, el punto que siempre debe de tener en cuenta es la experiencia de usuario que el videojuego pueda dar al jugador, es decir de qué manera se puede diseñar un videojuego para que aproveche al máximo las características únicas de la plataforma que lo soporta creando una relación satisfactoria entre el juego y el usuario, un ejemplo claro es el de la consola Switch donde Nintendo ha aprovechado los beneficios de los nuevos desarrollos tecnológicos en cuanto a miniaturización de hardware alcanzados unos años antes por el iPad y iPhone dando portabilidad y al mismo tiempo potencia de cómputo y gráfica creando una consola híbrida que puede jugarse en tv o en un entorno portable.

En este sentido las empresas también deben de aprender de sus errores y que mejor ejemplo que la marca Nintendo nuevamente ya que aunque se ha tocado anteriormente el tema de innovación, también esa misma compañía ha cometido

Capítulo 2. Introducción a los Videojuegos

2.6 Plataformas

errores que le ha perjudicado en ventas y en su estatus como puntera en el mercado, para ejemplificar esta situación se puede mencionar la salida a la venta de la consola *Virtual Boy* en 1995 donde Nintendo intentaba recrear la experiencia de realidad virtual, pero en la práctica fue una consola que simulaba el 3d real, con una pantalla monocromática en rojo y negro que causaba irritación en los ojos, además para jugarse se tenía que ejecutar una postura incómoda ya que era como casco de VR pero con unos soportes para usarse sobre una mesa causando dolor de cuello; en síntesis no era ni una consola portátil, ni una consola de sobremesa, ni un simulador de realidad virtual por lo que con 22 títulos en el mercado y con tan solo 800.000 unidades vendidas, fue retirada del mercado a menos de un año de su lanzamiento.

Es importante recalcar que para el creador de videojuegos novato el *smartphone* es la mejor elección de desarrollo inicialmente ya que esta plataforma permite crear juegos que no tienen tantos requerimientos técnicos de desarrollo, por ejemplo se usan modelos 3d low poly que pueden ser creados en una computadora promedio, provocando que los tiempos de desarrollo sean bajos, también hay una multitud de Engines a buen precio que pueden funcionar de forma estandarizada para armar el juego logrando un ahorro en el presupuesto inicial y además lo más interesante es la monetización del juego a través de anuncios que se presentan dentro del videojuego en forma de *banners* o *rich media ads* por medio de compañías como *AdMob*, *Unity Ads*, *WebEye* y *mopub* ya que cada vez que el jugador da click en ellos el desarrollador se lleva una cantidad de ganancia solventando los esfuerzos iniciales del desarrollador.

Otro punto importante que puede cambiar el panorama de consumo de videojuegos es el *Cloud Gaming* ya que si se logra establecer de forma estandarizada puede llegar a disminuir la piratería y los mercados no formales alrededor del mundo consolidando las ganancias reales para los estudios y la industria ya que se tendrá acceso a un gran número de juegos a precios competitivos.

Capítulo 2. Introducción a los Videojuegos

2.7 Modos de Juego

2.7 Modos de Juego

El modo de juego se define por la cantidad de jugadores y la forma en que los usuarios juegan un determinado videojuego, existen juegos donde es un jugador exclusivamente el que interactúa, así como juegos donde pueden ser varios jugadores al mismo tiempo ya sea en pantallas divididas, por conexiones LAN o por Internet.

Con la llegada de velocidades más rápidas de internet y el uso masivo de smartphones con capacidades de juego online, los videojuegos multijugador multiplataforma están cobrando gran fuerza siendo el género Battle Royale con juegos como *PUGB Mobile*, *Fortnite Battle Royale*, *Aphex Legends*, así como el género *Multiplayer Online Battle Arena* con juegos como *League of Legends*, *SMITE* y *Clash Royale* ambos géneros en su modalidad *Free to Play* de los preferidos por los jugadores por su capacidad de poner a prueba las habilidades a nivel global jugando en contra de varios usuarios al mismo tiempo; incluso *League of Legends* es reconocido como uno de los juegos más populares de los e-sports.

Los principales modos de juegos son:

Un jugador

En esta modalidad únicamente es un jugador el que interactúa con la Inteligencia Artificial (también llamada Non Player Character (NPC))

Dos jugadores

Dos Jugadores son los que hacen uso de la lógica del juego, en sus inicios los 2 jugadores competían entre ellos (*Tennis for Two* y *Pong*) después se popularizó el modo turno por turno, es decir si el jugador 1 perdía en el nivel recorrido, era el turno del jugador 2 y empezaba desde el principio, otra modalidad que con el juego *Double Dragon* se popularizó, fue la de 2 jugadores contra el CPU.

Multijugador (multiplayer)

Los jugadores están conectados simultáneamente usando el mismo hardware, así como la misma pantalla, con controles separados.

LAN multiplayer

Numerosos jugadores pueden compartir la experiencia de juego a través de su computadora en un lugar específico (LAN parties) por medio de una conexión LAN (Local Area Network).

Online multiplayer

Esta categoría es definida por la gran cantidad de usuarios que pueden estar conectados dentro de un juego interactuando entre ellos en tiempo real, por medio

Capítulo 2. Introducción a los Videojuegos

2.7 Modos de Juego

de Internet.

Es la forma más novedosa y atractiva de jugar en la actualidad ya que los servidores donde se conectan los jugadores para estar en línea están a disposición las 24 horas y no es necesario esperar muchos minutos para tener una partida.

Las experiencias de juego creadas por los desarrolladores son diseñadas para que contengan *Gameplay* que pueda variar de acuerdo al modo de juego y que pueda cubrir las expectativas de los jugadores ya que existen muchos nichos de *Gamers*, dándoles opciones que mejor se adapten a sus gustos y requerimientos de interacción social.

Por ejemplo los juegos de un jugador en términos generales generan una experiencia más personalizada, introspectiva, detallada, donde el jugador puede meterse en el personaje, además si involucra una historia, ser partícipe de ella, se puede decir que en el modo un jugador el reto a vencer es con uno mismo gracias a los diferentes grados de dificultad que puedan existir; por otra parte en los juegos multijugador online, el jugador tiene que enfrentarse a un oponente humano por lo que los niveles de dificultad se segmentan por el grado de destreza de los jugadores de forma exponencial ya que la principal característica de este modo es encontrar oponentes con diferentes cualidades y niveles de juego haciendo que las partidas online se vuelvan más ricas e interesantes para los jugadores con patrones de juego únicos, siendo estas partidas mucho más complejas que la interacción con una Inteligencia Artificial, además el jugador tiene la satisfacción de saber que tiene como oponente a una persona de carne y hueso que utilizará todo sus recursos a la mano y hará todo lo posible por ganar.

En este sentido hay que resaltar las experiencias de juego que proveen los videojuegos multijugador online que se han convertido en un fenómeno mundial generando ganancias millonarias a las empresas productoras, tal es el caso de Fortnite Battle Royale que al ser desarrollado en PC, consolas y smartphones cubrió ampliamente un nicho de jugadores ávidos de experiencias de juego en contra de oponentes reales.

Otro caso interesante es el de League of Legends un juego MOBA que desde su lanzamiento en 2009 ha ganado una cantidad gigantesca de jugadores, registrando para 2019 80 millones de usuarios activos en un mes, ganando año con año nuevos usuarios convirtiéndose en uno de los juegos más jugados del mundo, manteniéndose vigente en la actualidad, además gracias a juegos como este es que el fenómeno de los e-sports se ha exponenciado en la década pasada y una muestra de ello a nivel regional, es la creación del primer estadio de e-sports en México que lleva por nombre ARENA Esports Stadium en 2020 en la Ciudad de México, esto nos habla de las posibilidades que tiene esta actividad y su favorable adopción en el mercado latinoamericano potenciado por marcas que inyectan gran cantidad de inversión.

Capítulo 2. Introducción a los Videojuegos

2.8 ESRB - Clasificación de videojuegos por contenidos

2.8 ESRB - Clasificación de videojuegos por contenidos

El ESRB (*Entertainment Software Rating Board*) es una organización autorregulada estadounidense surgida en el año de 1994 que tiene la función de clasificar los videojuegos de acuerdo a su contenido.

Fue pensada en respuesta a títulos que contenían una excesiva carga de contenido violento, así como temas de sexualidad.

Un ejemplo de juegos que generaron controversia en esa época y por lo cual se fundó la ESRB fueron: *Night Trap*, *Lethal Enforcers*, *Doom* y *Mortal Kombat*.

La regulación de contenidos en los videojuegos principalmente por parte de instituciones gubernamentales ha sido tema controversial a lo largo de la vida de los videojuegos, ya que estas dependencias argumentan que los contenidos violentos pueden influir en la conducta de los jugadores en su mayor caso menores de edad, volviéndolos agresivos y tolerantes a la violencia haciendo de ésta un ente normal al que el jugador se acostumbra en su vida diaria provocando cambios violentos en su comportamiento. Ante estos ataques la industria argumenta que existen muchos más factores que influyen en la conducta violenta de un jugador, como su entorno familiar, su entorno social, su capacidad de distinguir lo real de lo ficticio y hasta el contacto con otros contenidos como películas y series de TV, ya que el contexto de cada jugador es muy diferente.

Sin embargo es verdad que existe un mercado de videojuegos sumamente violentos destinados al público adulto que ha tenido gran éxito con un grado de fotorrealismo gráfico muy grande que presentan situaciones y contenidos tan parecidos a la realidad que para niños pequeños puede ser difícil distinguir lo que es real y lo que es ficticio, por eso mismo las clasificaciones por edad y la atención que los padres ponen en los juegos que consumen los hijos son una medida efectiva y real de delimitar los contenidos adecuados para cierto público. De otra forma limitando los contenidos por medio de censura sólo causaría molestias en los consumidores alejándolos y sería injusto para un medio que tiene alcances tan amplios y ricos donde el límite es la imaginación, además una de las características principales de los videojuegos es presentar realidades o mundos ficticios y poder asumirlos como una experiencia propia.

También es curioso cómo en la actualidad con consolas de última generación como *PlayStation 5* de Sony, *Xbox Series X|S* de Microsoft y *Switch* de Nintendo con gráficos hiperrealistas, los gráficos de los juegos violentos de generaciones anteriores como *Mortal Kombat* o *Quake*, que en su momento causaron furor por lo violentos, ahora dan nostalgia y hasta parecen graciosos.

Estas clasificaciones seguirán modificándose conforme aparezcan nuevas formas de juego y los gráficos sean más realistas e inmersivos.

Capítulo 2. Introducción a los Videojuegos

2.8 ESRB - Clasificación de videojuegos por contenidos

Las Clasificaciones ESRB son las siguientes:

	<p>Everyone: Contenido para todas las edades</p>
	<p>Everyone 10+: Contenido para usuarios de 10 años en adelante. Contiene un mínimo de temas sugestivos</p>
	<p>Teen: Contenido para mayores de 13 en adelante. Contiene violencia, humor crudo, infrecuente uso de lenguaje fuerte.</p>
	<p>Mature 17+: Contenido para mayores de 17 años. Escenas intensas de sexo, violencia y lenguaje fuerte.</p>
	<p>Adults only 18+: Contenido para mayores de 18 años. Escenas prolongadas de violencia, sexo y juego de apuestas.</p>
	<p>Rating Pending: Rating Pending: Sin categoría asignada (para promocionales).</p> <p>2.112 Guía de Clasificaciones ESRB ESRB. (2021). Guía de Clasificaciones ESRB [imagen]. Recuperado 3 September 2021, de https://www.esrb.org/ratings-guide/es/#rating_cat.</p>

Capítulo 2. Introducción a los Videojuegos

2.9 Game Engines

2.9 Game Engines

Las oportunidades que el avance tecnológico brinda para el desarrollo de videojuego se materializaron principalmente por la comercialización y adopción masiva de los *Game Engines*, usados inicialmente a mitades de los 90's.

Los Motores de Juego (*Game Engines*) son programas que sirven para crear videojuegos o aplicaciones interactivas ya que contienen módulos preprogramados en su sistema -como son los módulos de Física, Audio, Inteligencia Artificial, Renderizado Gráfico, Animación, Efectos especiales entre otros-, que ahorran tiempo al desarrollador facilitando la producción ya que no se parte de cero evitando programar cada módulo individualmente, lo que trae consigo muchos beneficios económicos, prácticos, y funcionales en el proceso de desarrollo, además los *Game Engines* son pensados para exportar el juego creado en varios soportes multiplataforma como son consolas, móviles y PC, además se puede modificar el código fuente original del *Game Engine* en dado caso que se tuvieran que hacer cambios para aumentar la funcionalidad de acuerdo los requerimientos de cada proyecto.

Múltiples factores han permitido el desarrollo y la adopción masiva de *Game Engines*, como los son el avance tecnológico reflejado en la estandarización de los requerimientos de cómputo haciendo el desarrollo de juegos más accesible a menor costo, la creación de *Engines* gratis u *Open Source*, la generación y publicación de documentación por medio de manuales, libros, tutoriales, cursos, certificaciones y hasta estudios universitarios y la estandarización en el uso de gráficos sorprendentes por los principales *Game Engines* como *Unreal Engine* y *Unity*, logrando la proliferación de estudios independientes llamados indies, así también ha ido a la alza la producción de contenidos interactivos diferentes a videojuegos, en industrias de arquitectura, ingeniería, cine, publicidad, sector automotriz, del transporte, salud entre otros; también es de considerar que no sólo los estudios indies escogen *Game Engines*, sino que también los grandes estudios los incluyen para su desarrollo, en ocasiones creados específicamente para su producción, aunque también es común encontrar el uso de *Game Engines* comerciales como *Unity 2019.3*, *Unreal Engine 4* o *Cryengine 5.7* algunos de los principales motores de juego de 2020.

Otro factor muy importante para que exista un mayor desarrollo de videojuegos relacionado con los *Game Engines*, es que a la par del avance tecnológico de los motores de juego, se mantiene la evolución y desarrollo de software para la creación de elementos artísticos (*assets*) 2d y 3d como *3ds Max*, *Maya* o *Cinema 4D* para generar personajes, ambientes y objetos; texturas procedurales con *Substance Painter* y *Substance Designer* donde la computadora utiliza mecanismos matemáticos para generar texturas similares a las que se encuentran en la naturaleza y efectos visuales en tiempo real con *Embergen*, por citar algunos, estas herramientas permiten generar

Capítulo 2. Introducción a los Videojuegos

2.9 Game Engines

productos más elaborados de forma más rápida y sin tan complejidad, dando una facilidad y adaptabilidad sin precedentes al uso de software de creación de assets (2D y 3D) para videojuego siendo una actividad laboriosa pero flexible que poco a poco es generadora de arte más asombroso y complejo.

Los Game Engines tienen varias licencias de uso donde se especifica la propiedad intelectual del software, siendo Licencia de software libre (*GNU General Public License*) donde se puede reproducir, modificar o usar de forma libre ese software y licencias Propietarias (*Property*) donde el código fuente no es libre y forma parte dentro de los derechos de autor de una empresa y su uso tiene costo.

En un principio el software listado fueron los iniciadores de los *Game Engines*:

Ultima Underworld (1989)	Renderware (1996)
Doom Engine (1993-ID software)	GoldSRC (1998)
Voxel Engine (NovaLogic -1992)	Unreal Engine (1998)
Build Engine (1993)	Serious Engine (2001)
XnGine (1995)	Source Engine (2005)
Quake Engines (1996)	Rockstar Advanced Game Engine(RA-GE)(2006)

Para 2020 algunos de los Engines más exitosos comercialmente hablando son:

Frostbite, de licencia propietaria, uno de los *Engines* más usados, algunos de sus títulos: *Anthem (2019)*, *Battlefield V (2018)*, *FIFA 19 (2019)*, *Star Wars Battlefront II (2017)*.

Unreal Engine 4 con juegos como *Fornite (2017)*, *Gears 5(2019)* *Street Fighter V (2016)* *Daemon X Machina (2019)* *Minecraft Dungeons (2019)* es gratis hasta que se obtengan ganancias de \$ 3000 USD, entonces se paga un porcentaje de 5% de la ganancia total.

Unity (2019.3), plataforma de desarrollo en tiempo real, viene en la versión gratis y de pago, algunos juegos memorables son: *Cuphead (2017)*, *Monument Valley 2 (2017)*, *Inside (2016)*, *Super Mario Run (2016)*.

Se presenta también una lista con algunos otros *Engines* muy útiles para personas que van iniciando en la creación de juegos por su sencillez y bajo costo o distribución gratuita.

Capítulo 2. Introducción a los Videojuegos

2.9 Game Engines

Gdevelop	Defold
CopperCube	RPG Maker
Corona	BuildBox
Play Canvas	Gamesalad
Torque 3d	Shiva Engine
Stencyl	Panda 3d
Amazon Lumberyard	Construct 3
Cryengine V	Game Maker
Godot	Cocos 2d

Para escoger un *Game Engine* adecuado se deben tomar en cuenta factores como el precio, las licencias de uso, qué tan complejo es de aprender, si el lenguaje de programación usado en el Engine es conocido y que tan difícil es, si ese mismo lenguaje durará por mucho tiempo y que tipo de soporte tiene, si existe una gran compatibilidad con programas de creación de *assets* como *Maya*, *3ds Max* o *Zbrush*, si el *Engine* tiene suficiente documentación o existan lugares destinados a la capacitación; además se debe tomar en cuenta el tipo de género y el fin del juego a desarrollar para constatar la compatibilidad en el desarrollo especificando los alcances gráficos e interactivos del producto, además si el *Engine* tiene tienda de *assets* preconfigurados o para uso como plantillas; en cuanto al entorno tecnológico sobre qué sistemas operativos correrá, si es un *Engine* conocido utilizar desarrolladores con conocimiento en el área o en caso contrario donde se necesite capacitación así como revisar la versión del Engine que más favorezca a la inversión por las actualizaciones constantes del software teniendo en cuenta el factor profesional y tecnológico.

Capítulo 2. Introducción a los Videojuegos

2.10 Tipos de programación usados en Game Engines

2.10 Tipos de programación usados en Game Engines

La función principal de un *Game Engine* es crear elementos interactivos en un mundo virtual donde el jugador pueda utilizar cada uno de estos elementos a su antojo, por ejemplo controlar un personaje por un bosque, o manejar un auto de carreras en una pista, para lograr este objetivo el desarrollador del juego llamado en ocasiones programador, crea las instrucciones que controlarán el mundo virtual por medio de código de programación, para ello usa lenguajes de programación que contienen códigos de computadora, de esa forma la computadora interpreta las instrucciones y las traduce en acciones dentro del *Engine*.

Las dos formas más usadas para la programación de videojuegos son la programación basada en texto y la programación visual y su diferencia radica en la facilidad de uso y la complejidad de los resultados que se quieran obtener.

Programación basada en texto

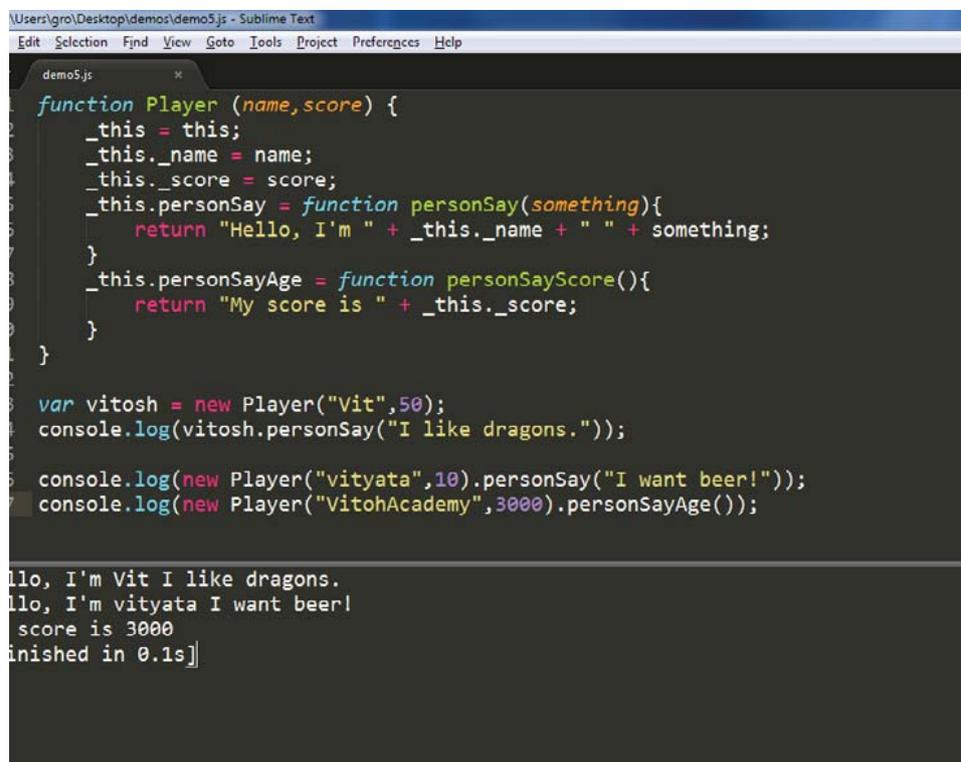
La programación basada en texto es la más usual en la industria y consiste en escribir líneas de código para crear un programa que se ejecute en una computadora, para su correcta operación se debe de conocer la sintaxis del lenguaje en el que se programa, es decir se deben conocer las reglas que dan forma a las instrucciones que la computadora recibe, cabe mencionar que al ser el estándar de producción de los estudios desarrolladores de videojuegos se necesitan más conocimientos previos para poder desenvolverse en el área, como lo es tener conocimientos avanzados en matemáticas, física, ingeniería de software y áreas afines, por lo que puede llegar a ser complejo entender cómo desarrollar al principio bajo esta forma, teniendo una curva de adaptación más larga.

Cabe mencionar que con este tipo de programación se pueden lograr resultados más avanzados ya que se tiene más control en cada una de las partes que conforman el producto interactivo, ganando en rendimiento y personalizándolas al grado que requiera las exigencias de producción.

Para darse una idea, el *Engine Unity* utiliza el lenguaje de programación C# (C Sharp), mientras que el *Unreal Engine* utiliza el lenguaje de programación C++. (C plus plus) y existen una gran cantidad de lenguajes como: C, Python, Java, Javascript, Lua por mencionar algunos.

Capítulo 2. Introducción a los Videojuegos

2.10 Tipos de programación usados en Game Engines



```
function Player (name, score) {
  _this = this;
  _this._name = name;
  _this._score = score;
  _this.personSay = function personSay(something){
    return "Hello, I'm " + _this._name + " " + something;
  }
  _this.personSayAge = function personSayScore(){
    return "My score is " + _this._score;
  }
}

var vitosh = new Player("Vit",50);
console.log(vitosh.personSay("I like dragons."));

console.log(new Player("vityata",10).personSay("I want beer!"));
console.log(new Player("VitoAcademy",3000).personSayAge());

Hello, I'm Vit I like dragons.
Hello, I'm vityata I want beer!
score is 3000
finished in 0.1s]
```

2.113 Código Javascript

vikingcodeschool.com. (2020). Código Javascript [imagen]. Recuperado 25 Marzo 2020, de https://s3.amazonaws.com/viking_education/web_development/javascript/running_code_in_sublime_text.png.

Programación Visual

La programación visual es un término acuñado para definir el uso de software para la manipulación de instrucciones de computadora previamente codificadas por ingenieros con el fin de generar interacción, por medio de un entorno gráfico, sencillo y amigable donde el usuario que crea la interacción no necesita conocer programación y aun así pueda crear programas elaborados y atractivos.

En este entorno visual, no se escriben líneas de código en programas de edición de texto como es la forma tradicional, sino que en cambio se crea la lógica conectando nodos o bloques predefinidos que contienen acciones, eventos y propiedades.

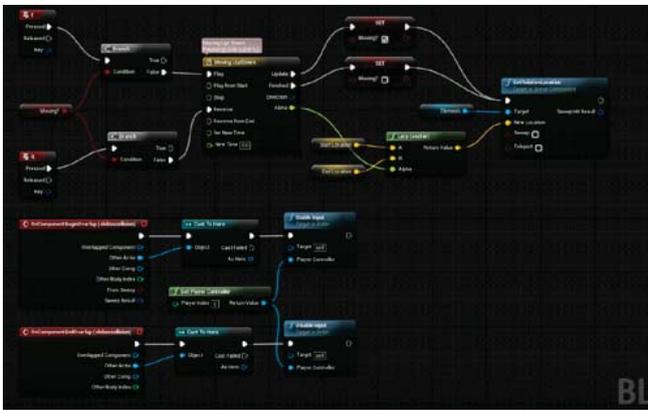
La programación visual en los *Game Engines* surge como alternativa para todas aquellas personas que deseen crear videojuegos de forma fácil y accesible, además también funcionan para prototipar ideas de forma rápida y comprobar si funcionan o no para producirse posteriormente con herramientas más poderosas; sin embargo, hay que destacar que generalmente este tipo de *Engines* con programación visual maneja acciones preestablecidas por lo que es difícil personalizar el contenido.

Capítulo 2. Introducción a los Videojuegos

2.10 Tipos de programación usados en Game Engines

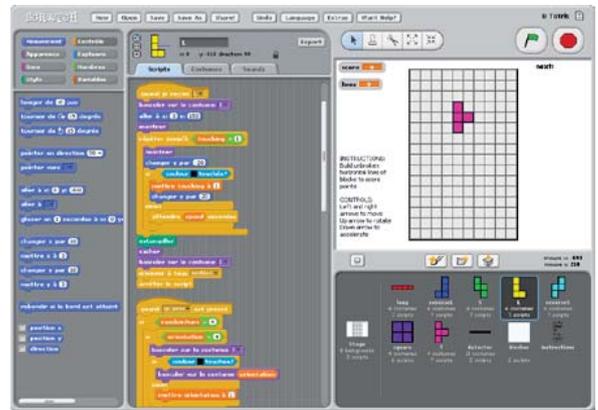
Una ventaja al usar programación visual es que el usuario comienza por razonar de forma lógica para la solución de un problema, empezando por resolver una pequeña parte, abarcar más del problema y finalmente relacionar las partes faltantes para lograr su cometido.

Otras características que adquirirá con el uso de un lenguaje de programación visual es que podrá conocer, practicar, concretar y adoptar conceptos sobre pensamiento computacional, iniciándose en el área de los contenidos interactivos, mejorando el pensamiento lógico matemático.



2.114 Blueprints / Unreal Engine

fiverr.com. (2020). Blueprints / Unreal Engine [imagen]. Recuperado 25 Marzo 2020, de https://fiverr-res.cloudinary.com/imgens/q_auto,f_auto/gigs/121705582/original/cc2e1b0c5d8fd35685887982e737ee6736b1f311/create-your-game-mechanic-on-unreal-engine-4-in-blueprint-d911.png.



2.115 Scratch

wikimedia.org. (2020). Scratch [imagen]. Recuperado 25 Marzo 2020, de https://upload.wikimedia.org/wikipedia/commons/b/b6/Scratch_Fr_D%C3%A9mo.png.

Los *Game Engines* más fáciles que usan programación visual son *Construct 3* con su sistema de eventos preestablecido y *Buildbox 3.0* incorporando el uso de elementos 3d para crear las interacciones.

La estrategia a seguir para un desarrollador novato que quiera crear un videojuego por primera vez dependerá de su formación educativa y las referencias tecnológicas que tenga sobre videojuegos, pero sobre todo dependerá del tipo de videojuego y de la funcionalidad principal que se quiera llegar a producir, así como del presupuesto que se tenga contemplado para invertir en conocimiento y habilidades adquiridas enfocadas al desarrollo.

De esta forma el probable camino a seguir en un principio es empezar a prototipar

Capítulo 2. Introducción a los Videojuegos

2.10 Tipos de programación usados en Game Engines

en *Engines* con programación visual básica tratando de empaparse lo más que se pueda del entorno de producción ya que sólo con la práctica se pueden asimilar conceptos concretos de *Gameplay* que después podrán ser modificados y adaptados con más recursos de forma tradicional por programación basada en un lenguaje de código. Además, cada año surgen nuevas herramientas que mejoran los *Engines* y facilitan la creación de entornos interactivos.

Ya sea que se use programación visual o programación basada en texto se deben conocer puntualmente las estructuras lógicas antes de pensar en la idea del juego.

Tal vez una práctica destacada es la unión de los dos tipos de programación ya que a fin de cuentas son instrucciones que la máquina lee de la misma forma, el único cambio visible es hacia la interfaz que se da con el desarrollador del juego. *Unreal Engine* es el claro ejemplo de cómo los dos tipos de programación conviven en armonía en un desarrollo de videojuego ya que se pueden crear interacciones rápidas gracias a su estructura de *Blueprints* basados en programación visual complementados con clases de programación logradas en el lenguaje de programación C++ que extienden la funcionalidad del juego.

En síntesis, el desarrollador novato tendrá que buscar el *Game Engine* y el tipo de programación que más se adapte a sus necesidades y recursos ya que no existe un desarrollo universal por el que deba guiarse.

Capítulo 2. Introducción a los Videojuegos

2.11 Estructuras de control para programación visual de videojuegos

2.11 Estructuras de control para programación visual de videojuegos

La programación consiste en darle instrucciones a la computadora en un lenguaje de programación que la máquina pueda entender con el fin de que ejecute un programa. Generalmente las instrucciones codificadas en un programa hecho con algún lenguaje de programación son dadas unas tras otras linealmente, si se quiere romper esa linealidad se usan estructuras de control.

Las estructuras de control son instrucciones que permiten romper la secuencialidad variando el orden de la ejecución de un programa; esto significa que una estructura de control permite que se realicen unas instrucciones y omitir otras, de acuerdo a la evaluación de una condición.

La mayoría de los lenguajes de programación utilizan la misma estructura básica y bloques de construcción, por lo que una vez aprendiendo las estructuras de control básicas será más fácil el aprendizaje de otros lenguajes, de esta forma lo que varía entre las estructuras de control de los diferentes lenguajes es su sintaxis; cada lenguaje tiene una sintaxis propia para expresar la estructura.

Aunque es posible crear un entorno interactivo desde cero intuitivamente en un *Game Engine* con programación visual por medio del uso y ejecución de acciones predefinidas, es aconsejable conocer los fundamentos que las estructuras de control ofrecen, ya que de esta manera se podrán crear interacciones más complejas, además de entender la documentación que el *Engine* proporciona sobre su uso, así como bibliografía y tutoriales relacionados aprovechando al máximo los recursos que el *Engine* ofrezca.

Así también es importante remarcar que el empezar a reconocer y aplicar estructuras de control en el Engine genera una introducción práctica al tema facilitando su comprensión, para posteriormente si es necesario, profundizar en el tema con estudios más avanzados.

Se mencionan a continuación de forma simple las estructuras de control (*control Structures*).

Secuencial (Sequential):

Es el modo predeterminado de programación. Se realiza la ejecución secuencial de sentencias de código (una línea tras otra), es como seguir por orden los pasos de una receta.

Capítulo 2. Introducción a los Videojuegos

2.11 Estructuras de control para programación visual de videojuegos

Selección (Selection):

Se utiliza para tomar decisiones, bifurcando elegir entre 2 o más rutas alternativas. incluyen a if, if/else, y switch.

Repetición (Repetition):

Se utiliza para hacer un bucle (loop), es decir, repetir un fragmento de código varias veces seguidas.

Hay 3 tipos de **loop**: **while**, **do/while** y **for**.¹⁴

Si es requerido generar una instrucción más detallada o compleja dentro del programa, se deberán de usar operaciones matemáticas (suma, resta, multiplicación y división) aunque un desarrollo sencillo puede funcionar sin este tipo de operaciones.

Realizando una analogía sencilla se puede decir que las estructuras de control son como recetas de cocina, donde cada estructura se asemeja a la forma y el orden en la preparación de alimentos, son instrucciones que sirven para proveer acciones.

El mostrar las estructuras de control de forma simplificada por medio de las estructuras secuencial, de selección y repetición tiene la finalidad de acercar introductoriamente al lector hacia las partes principales que conforman el proceso de codificación usado en la mayoría de los lenguajes de programación, de esta forma tras entender el funcionamiento de cada una de estas estructuras podrá tener una idea mental sobre el tema, buscando que en investigaciones posteriores se acerque a la temática que más le agrade o que requiera teniendo un background general, ya que cada estructura tiene sus ramificaciones además de la diferencia de sintaxis del lenguaje en que serán escritas.

Sin embargo hay que aclarar que no por tener el conocimiento general de las estructuras de control significa que la persona sepa programar, ya que el área de ingeniería de software es muy vasta e incluye muchos más temas como son los paradigmas, modelos, metodologías, *frameworks*, prácticas, estándares, herramientas, procesos, por mencionar algunos, por lo que el aspirante a programador deberá de memorizar y comprender todo estos recursos y temáticas, siendo una labor que lleva tiempo y esfuerzo pero que es necesaria para realizar efectivamente su labor dentro de la ingeniería de software.

Para tener un acercamiento más fácil al área de las estructuras de control básicas se puede buscar el código por medio de ejemplos en pseudocódigo, ya que el pseudocódigo es la forma de escritura de código en un lenguaje para humanos no para máquina donde se explica el funcionamiento de cada estructura de forma sencilla omitiendo detalles que no son importantes para la comprensión.

14 (2020). Retrieved 3 January 2020, from <https://www.cs.fsu.edu/~myers/c++/notes/control1.html>

Capítulo 2. Introducción a los Videojuegos

2.12 Programación profesional de videojuegos

2.12 Programación profesional de videojuegos

Si de forma introductoria la programación visual resultó atractiva el siguiente paso lógico es empezar a elaborar juegos más avanzados, con mejoras notables, la forma de lograrlo es por medio de la generación de código fuente de forma profesional por medio de ingeniería de software, dando la oportunidad a que el videojuego pueda ser desarrollado de forma diferente al resto, personalizado, que tenga elementos y funciones nuevas mejorando sus características gráficas, efectos, o interacciones que se verán reflejadas en la jugabilidad.

Por lo tanto para aspirar a ser un desarrollador profesional como programador de videojuegos dentro de un estudio donde se produzcan juegos AAA con franquicias como *Mario Bros* o *Halo* o también dentro de un estudio Indie, un lenguaje de programación visual no bastará para cubrir esa exigencia de trabajo, ya que para desarrollar juegos más complejos un programador de videojuegos puede ser requerido para cubrir áreas como simulación, gráficos por computadora, inteligencia artificial, físicas, programación de audio, redes y bases de datos, por mencionar algunas, y cada una de éstas áreas está solventada por sólidos conocimientos en ciencias de cómputo, afirmando que los conocimientos que sostienen estas estructuras de trabajo se basan en trigonometría, álgebra vectorial y lineal, cálculo diferencial, física, arquitectura e ingeniería de software, probabilidad y estadística entre otros, conocimiento que en sí mismo toma años de estudio asimilar y dominar.

Existen varias áreas de programación en el desarrollo de juegos; en general aquí se presentan las principales.¹⁵

Programador de Game Engine: crea o customiza un Game Engine para el videojuego	Programador de jugabilidad: Implementa las mecánicas y lógica del juego
Programador de Físicas: desarrolla las físicas necesarias usadas en un juego	Programador de Interfaz de usuario: Crea el sistema de interfaz, con efectos especiales
Programador del Engine gráfico: se especializa en desarrollar los renders en tiempo real en 2D y 3D	Programador de red: escribe código que permite al jugador conectarse con otros usuarios por internet

¹⁵ Game programming. (2020). Retrieved 3 January 2020, from https://en.wikipedia.org/wiki/Game_programming

Capítulo 2. Introducción a los Videojuegos

2.12 Programación profesional de videojuegos

Programador de Inteligencia Artificial: desarrolla la lógica para simular inteligencia en los enemigos y oponentes	Programador de input: programa los dispositivos de entrada como controles
Programador de Audio: Crean herramientas que asocian sonidos a momentos dentro del juego	Programador lead: Es el responsable encargado del área de programación
Programador de Tecnología: Es el encargado de la investigación y desarrollo tecnológico	

La enseñanza en lo relativo a la programación y desarrollo de videojuegos es muy amplia siendo a nivel internacional en países como USA, Canadá, Singapur, Uk, donde se encuentran las principales entidades educativas ofreciendo lo último en conocimiento en preparación académica Universitaria, con cuotas elevadas pero teniendo la certeza que se obtiene capacitación de punta donde su principal ventaja es la vinculación directa con la industria por medio de prácticas y bolsa laboral para recién egresados.

A continuación, se mencionan algunas escuelas importantes donde se imparte enseñanza a nivel Universitario:

Full Sail University	DigiPen Institute of Technology
Massachusetts Institute of Technology	Vancouver Film School
University of Pennsylvania	SAE Institute
University Center for Technology and Digital Art (U-TAD)	Media Design School

También existen programas más reducidos de enseñanza donde se imparten Masters y Especializaciones con una duración promedio de 1 a 2 años.

Breda University of Applied Sciences	University of Malta
Tampere University	UCLan
ITMO University	UCLan
George Mason University	University of Portsmouth

En último término están los cursos y recursos online donde en ocasiones son contenidos tanto introductorios como demasiados técnicos, la ventaja es que son de precio accesible, pero se corre el riesgo de que el conocimiento se encuentre disperso, no estructurado y que no abarque el total de la temática.

Capítulo 2. Introducción a los Videojuegos

2.12 Programación profesional de videojuegos

UDEMY	Skill share
COURSERA	LYNDA
Game Institute	Pluralsight
CG Spectrum	Game art institute
	Bit Degree

En México las principales escuelas que brindan educación en videojuegos son:

COCO School	Universidad de Guadalajara
Instituto Universitario Amerike	Universidad Marista de Guadalajara
SAE México	Universidad de Morelia
UNIAT	Universidad Panamericana
University of Advanced Technologies	Universidad de Artes Digitales

Cabe mencionar que la literatura sobre videojuegos crece más cada año y se pueden encontrar libros de sus diversas áreas muy detallados y con una enseñanza más clara, mejor desarrollada.

A manera de cierre se puede decir que el ser un programador profesional implica tener varios puntos cubiertos que lograrán hacer que se destaque en la industria de forma exitosa; en primer lugar un gusto o afición por los videojuegos ya que conociendo el entorno de los juegos y teniendo la experiencia de haber jugado una serie de juegos favoritos se podrá generar un producto con características positivas ya que el creador podrá identificar puntos efectivos que darán forma a un juego notable y funcional; en segundo lugar una serie de capacidades y habilidades mentales e intelectuales que el programador deberá aplicar y en dado caso adquirir para el uso y procesamiento de información y operaciones, por ejemplo el entender conceptos abstractos que vienen relacionados de áreas como matemáticas, ingeniería y ciencia desarrollando varias cualidades, como la capacidad analítica para la resolución de problemas, la eficiente planificación del trabajo hecho por fases, la flexibilidad a cambios en los entornos y tareas de trabajo, el tener la facilidad de relacionar conceptos, y sobre todo tener disciplina de aprendizaje constante siendo autodidacta ya que la industria de los videojuegos es un área que está en constante cambio y actualización donde surgen tecnologías nuevas en tan solo un par de años, siendo la innovación la base que sostiene a la industria.

En tercer lugar, el estar completamente seguro al momento de escoger el área de programación de videojuegos ya que las cuotas educativas son altas por lo que se debe definir qué tipo de inversión económica se hará.

Capítulo 2. Introducción a los Videojuegos

2.12 Programación profesional de videojuegos

Algunas opciones son matricularse en una escuela reconocida en el extranjero o alguna de las nuevas escuelas dentro del país buscando becas o apoyos económicos donde el plan de estudios sea enfocado cien por ciento a videojuegos, esto si existe la posibilidad de cubrir las cuotas; aunque también está la opción de estudiar una carrera en ingeniería de software que abarca temáticas del desarrollo de software de forma general para después especializarse en alguna escuela online o presencial durante uno o dos años, de esta forma será más económico que matricularse en una escuela donde se cubre el tronco común de temáticas relacionadas con videojuegos.

Para esto existen muchos sitios web que comparan las escuelas y los planes de estudio, además presentan información sobre las ramas de trabajo y los salarios promedio que se pueden alcanzar en la práctica profesional.

Capítulo 3. Producción de Videojuegos

3.1 Fases de Producción

Capítulo 3. Producción de Videojuegos

3.1 Fases de Producción

Las fases de producción son las etapas en las que se divide la creación de un videojuego siendo necesarias para llevar un orden y método en la realización, tomando en cuenta factores muy importantes como la calidad, el tiempo y el presupuesto destinado. Cada una de estas fases trae consigo la generación de documentos especializados que, si bien son creados durante una etapa específica de desarrollo, su uso puede abarcar varias de estas. De esta forma la concepción de un videojuego pasa por las fases de concepto, preproducción, prototipo, producción, testing y postproducción; cada una puede extenderse por volúmenes enteros, sin embargo, con el fin de que el diseñador pueda entender por completo el ciclo productivo, en los siguientes apartados se describirán estas fases de forma sencilla y concreta.

Concepto

Esta fase empieza con la concepción de la idea del juego y termina con la planeación del juego. En esta etapa, los desarrolladores, diseñadores, artistas, productores y otros miembros del equipo generan el documento de concepto del juego donde hablan sobre descripciones de la motivación del jugador, el *Gameplay*, historia, audiencia, genero, plataforma, análisis de competencia, objetivos del juego, premisa, descripción del juego, sus puntos de venta entre otros, teniendo como finalidad dejarlo claro y por escrito, siendo la fase más importante en la producción.



3.116 Fallout 3 Game of The Year Edition - front / back cover.box3.net. (2020). Fallout 3 Game Of The Year Edition - front / back [imagen]. Recuperado 27 Marzo 2020, de <https://cover.box3.net/newsimg/dvdmov/max1267441795-front-cover.jpg>.

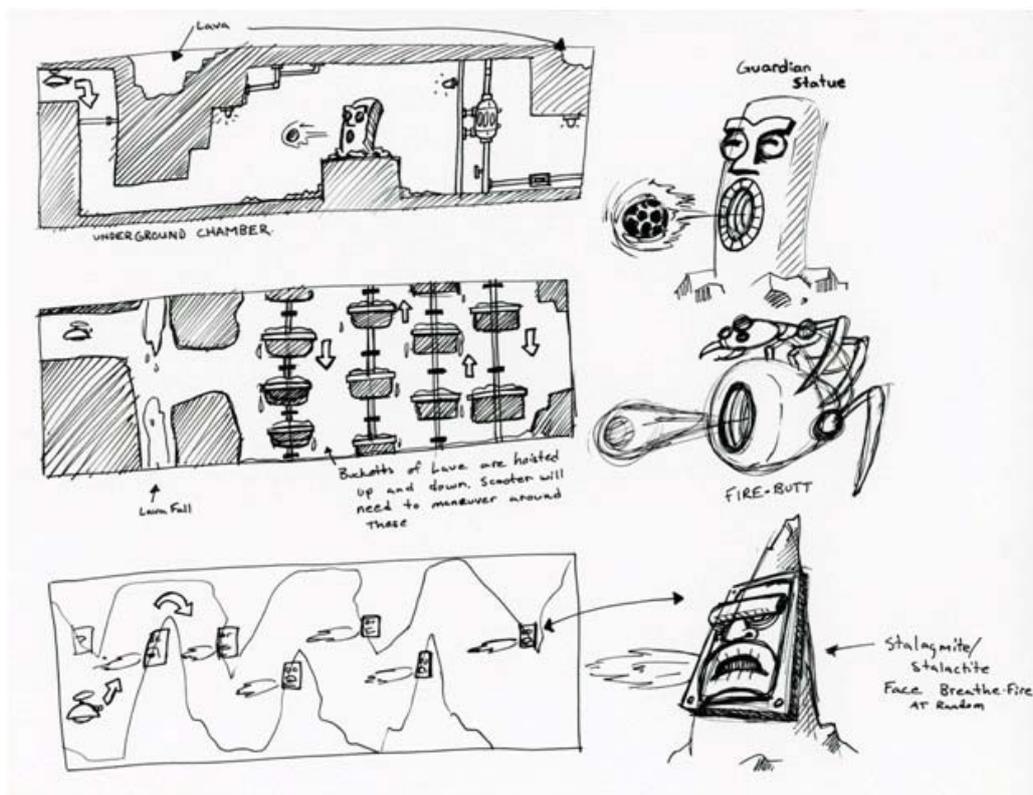
Capítulo 3. Producción de Videojuegos

3.1 Fases de Producción



3.117 Ganancias de la Industria del Videojuego en Norte América 2019

thecoverproject.net. (2020). Diablo III [imagen]. Recuperado 27 Marzo 2020, de http://www.thecoverproject.net/view.php?cover_id=16486.



3.118 Game concept

hub.packtpub.com. (2020). Game concept [imagen]. Recuperado 27 Marzo 2020, de https://hub.packtpub.com/sites/default/files/new_blog_imagens/552/552_01.png.

Capítulo 3. Producción de Videojuegos

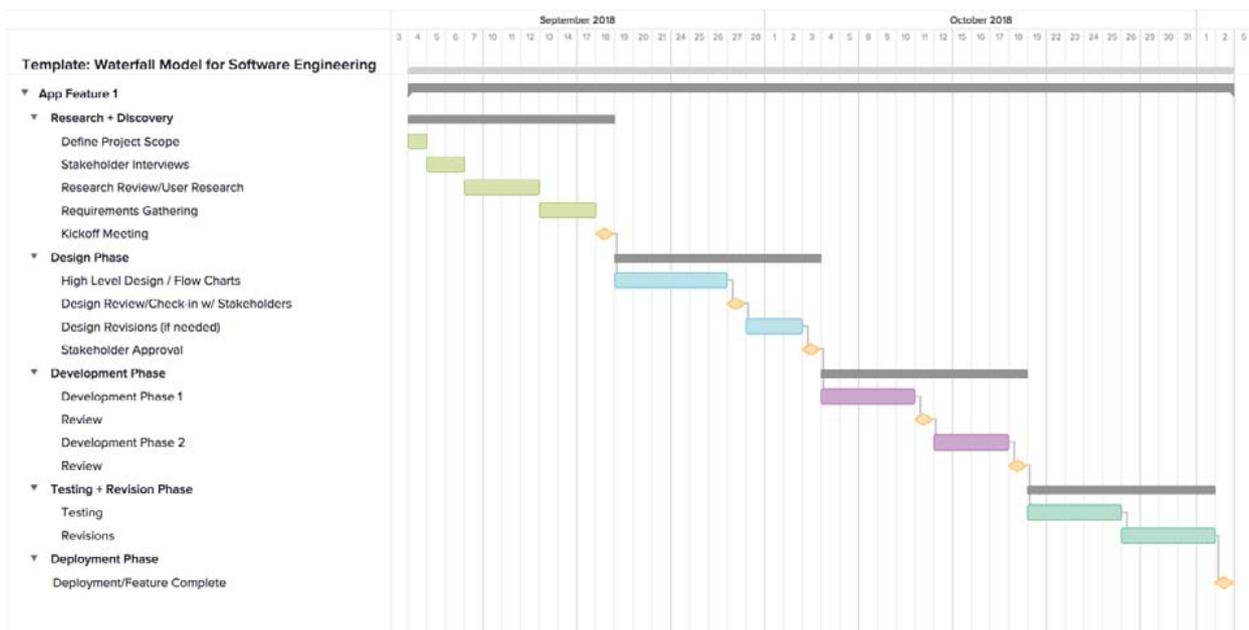
3.1 Fases de Producción

Preproducción

Cuando se tiene concretado un concepto sólido se pasa a la etapa de preproducción donde se desarrolla la propuesta detalladamente y se empieza la planeación del proyecto, además los documentos elaborados en esta fase son usados como guía en los que se basarán los integrantes del equipo para realizar el juego.

La fase termina con el Plan de Producción, la Biblia de Arte, el Documento de Diseño de Juego (*Game Design Document - GDD*) y el Documento Técnico de Diseño (*Technical Design Document - TDD*)

Por medio del Plan de Producción se describen todas las tareas, cronogramas y estimaciones que el equipo de desarrollo tendrá que ejecutar.



3.119 Waterfall model example for software development

teamgantt.com. (2020). Waterfall model example for software development [imagen]. Recuperado 27 Marzo 2020, de https://assets-global.website-files.com/5a5399a10a77cc0001b18774/5bc752beb0cfa616f40959a5_img-waterfall-model-template%20%402x.png.

Durante la preproducción, el diseñador, el director de arte y el artista conceptual colaboran para establecer el estilo artístico del juego, por lo que establecer la Biblia de Arte desde el principio ayuda a orientar a los nuevos artistas que ingresan al proyecto y asegura que el producto final tendrá un estilo consistente en todo momento.

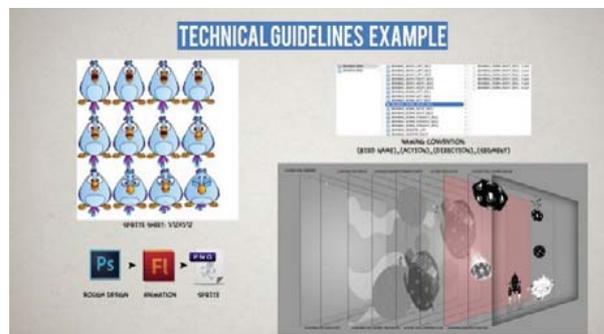
Capítulo 3. Producción de Videojuegos

3.1 Fases de Producción



3.125 User Interface

jacobduniam.com. (2020). User Interface [imagen]. Recuperado 27 Marzo 2020, de <https://i1.wp.com/imagen.slidesharecdn.com/pencilatikshirajngdc2013talk-131117032017-phpapp01/95/game-art-bible-secret-sauce-to-making-great-game-art-23-638.jpg>.



3.126 Technical Guidelines

jacobduniam.com. (2020). Technical Guidelines [imagen]. Recuperado 27 Marzo 2020, de <https://i1.wp.com/imagen.slidesharecdn.com/pencilatikshirajngdc-2013talk-131117032017-phpapp01/95/game-art-bible-secret-sauce-to-making-great-game-art-25-638.jpg>.



3.127 Diseño de Personaje

pinterest.com.mx. (2020). Diseño de Personaje [imagen]. Recuperado 27 Marzo 2020, de <https://i.pinimg.com/originals/aa/f2/47/aaf247bd5efdaaff1e301696ab765ed8.jpg>.



3.128 Diseño de Personaje

4.bp.blogspot.com. (2020). Diseño de Personaje [imagen]. Recuperado 27 Marzo 2020, de https://4.bp.blogspot.com/-lissW_MV9TI/T9UVX_P3VpI/AAAAAAAAAiU/9gOBv-6Lt-M/s640/MLP-Fighting-is-Magic---Applejack---Early-Concept-Sketches---By-Elosande.png.

El Documento de Diseño del Juego, uno de los documentos más esenciales en el proceso de producción de videojuegos, ya que detalla exhaustivamente todo lo que

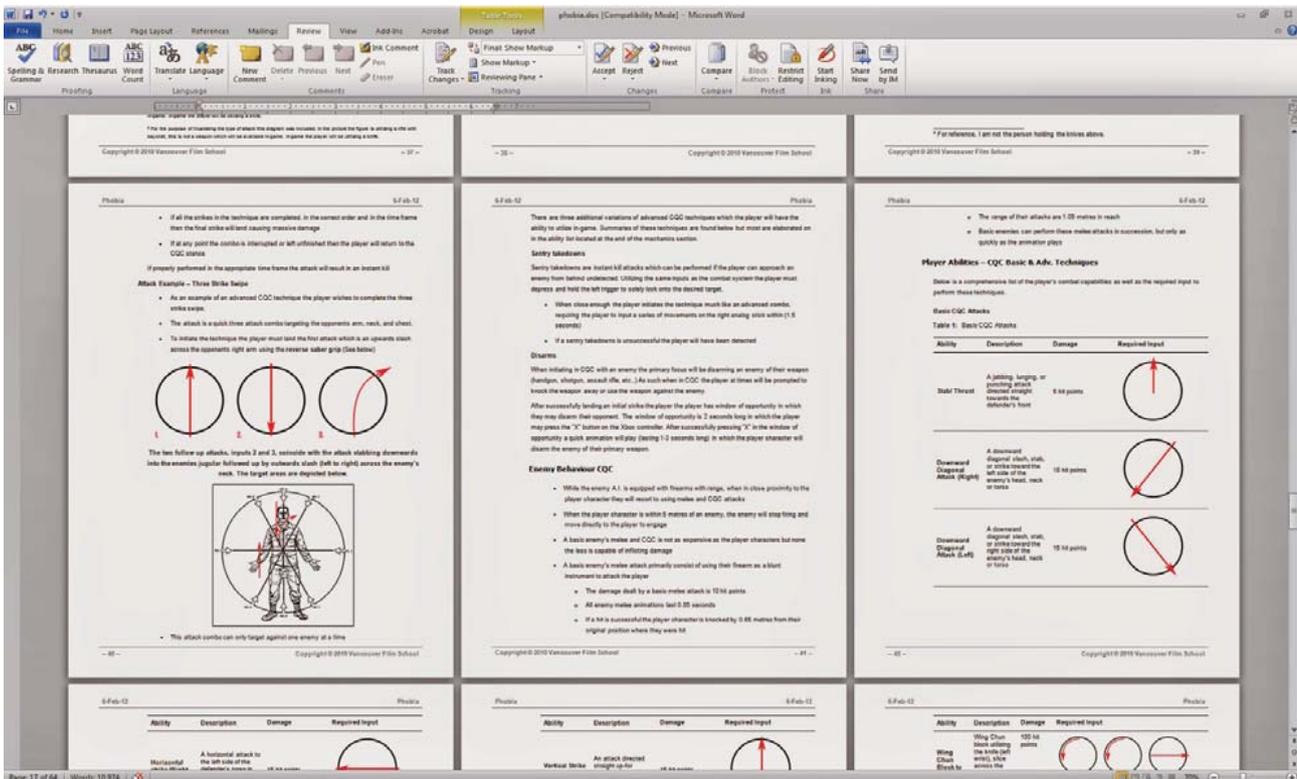
Capítulo 3. Producción de Videojuegos

3.1 Fases de Producción

el jugador experimentará en el juego.

Consiste en una descripción coherente de los componentes básicos, sus interrelaciones, direcciones y un vocabulario compartido para un desarrollo eficiente.

Incluye información sobre la historia, los personajes, el nivel, *Gameplay*, arte, sonido, Interfaz de usuario, monetización entre otros.



3.129 Game Design Document

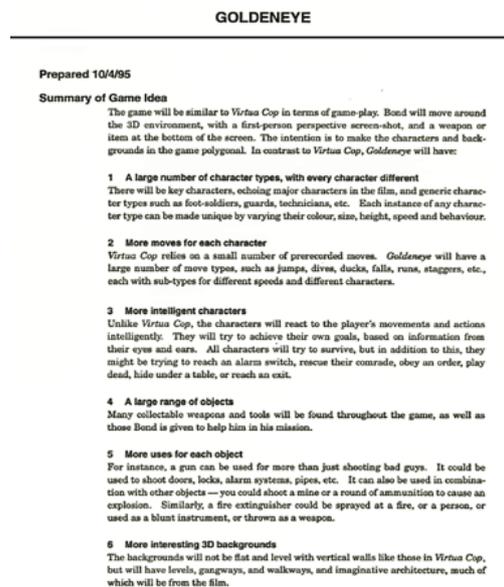
fnjdane.blogspot.com. (2020). Game Design Document [imagen]. Recuperado 27 Marzo 2020, de http://2.bp.blogspot.com/-CP-B_A_-V-60/VOOxTaxApiI/AAAAAAAAA8c/eJh2Rsqj3h0/s1600/combat2.jpg.

Capítulo 3. Producción de Videojuegos

3.1 Fases de Producción



3.130 GoldenEye 007: Game Design Document
goldenedecoded.blogspot.com. (1995). GoldenEye 007: Game Design Document (By Martin Hollis - Abril 1995) [imagen]. Recuperado 27 Marzo 2020, de https://4.bp.blogspot.com/-Tb-G3qeTfi80/UpshRnifh9I/AAAAAAAAAjU/N_HbuZ7L9oA/s1600/ge007_gdd_100495.png.



3.131 GoldenEye 007: Game Design Document
goldenedecoded.blogspot.com. (1995). GoldenEye 007: Game Design Document (By Martin Hollis - Abril 1995) [imagen]. Recuperado 27 Marzo 2020, de https://2.bp.blogspot.com/-Uts-qL3pANMk/UpsiLk9D2JI/AAAAAAAAAj0/ZCm_sVPdhvE/s1600/ge007_gdd_100495_p4.png.

El Documento Técnico de Diseño es escrito por el director técnico o programador principal, como una visión general de todos los aspectos técnicos de la producción del juego. Incluye detalles de todo el software, componentes de *hardware* y el *Game Engine* que se utilizarán durante la producción, y lo que esas herramientas proporcionarán al equipo de desarrollo. También incluye un resumen de la descripción de cómo se escribirá el código para el juego y los pipelines para diferentes áreas.

El conocimiento y uso de los documentos de preproducción es una actividad que debe de ser abordada con autoridad y dedicarle el suficiente tiempo a su elaboración utilizando formatos estandarizados ya que es la columna central en la que se basará la realización del juego, además si llegan a existir cambios y revisiones en fases posteriores ya que la creación de un juego se da por mejoras a través de ciclos no lineales, estos documentos permiten ser la guía eficiente para los responsables de cada área, llegando a delimitar las modificaciones para evitar caer en el caos y la falta de rumbo, eficientando el trabajo en equipo ya sea que se trate de una gran

Capítulo 3. Producción de Videojuegos

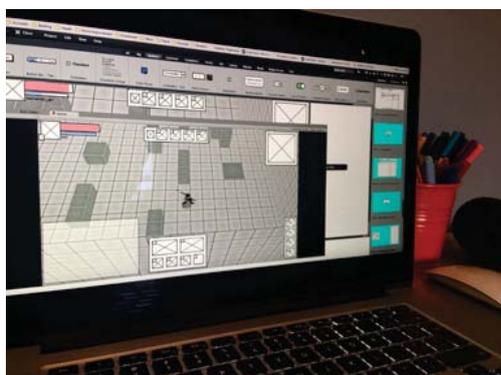
3.1 Fases de Producción

producción donde exista una multitud de profesionales ejerciendo su profesión o un pequeño grupo de creativos creando juegos independientes; de esta forma la documentación principal provee el control al diseñador del juego para cumplir su visión generando organización y sosteniendo realmente el desarrollo.

Es importante mencionar que idealmente la creación de estos documentos involucra a los jefes responsables de cada área con el diseñador de juego, donde por medio de juntas periódicas se busca llegar a concretar acciones aterrizando las ideas para actuar en un entorno que sea factible, de esta forma cada especialista aporta su visión y experiencia respaldándose en las capacidades de su equipo de trabajo.

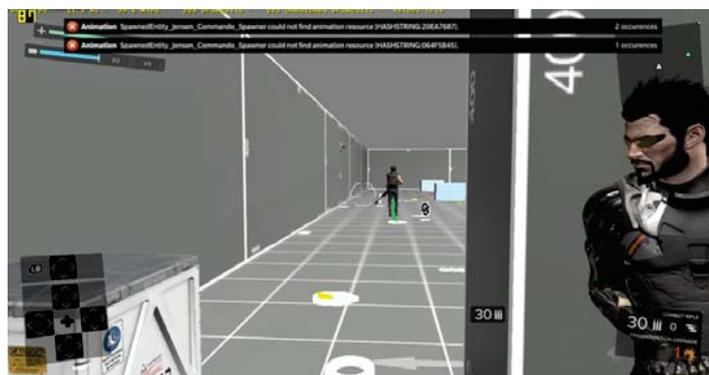
Prototipo

En esta fase se crea un prototipo, es decir una pieza de software funcional que sea basado en el concepto inicial y que capture su esencia en lo más que se pueda, con él se puede comprobar si las mecánicas de juego funcionan y si el juego es divertido y cumple con las expectativas. Una forma rápida y efectiva es crear el prototipo con un **Game Engine** (motor de juegos) ya existente.



3.132 Prototipo de Videojuego

blog.balsamiq.com. (2020). Prototipo de Videojuego [imagen]. Recuperado 27 Marzo 2020, de https://blog.balsamiq.com/wp-content/uploads/2017/01/IMG_2598.jpg.



3.133 A “grey Box” level from the development of Deus Ex: Mankind Divided. Screenshot courtesy of Eidos Montreal.

vice.com. (2020). A “GREY BOX” LEVEL de THE DEVELOPMENT OF DEUS EX: MANKIND DIVIDED . SCREENSHOT COURTESY OF EIDOS MONTREAL. [imagen]. Recuperado 27 Marzo 2020, de https://vice-imagens.vice.com/imagens/content-imagens-crops/2016/10/16/why-its-so-hard-to-make-a-video-game-body-imagen-1476658493-size_1000.png.

La fase del prototipado funciona a varios niveles resolutivos dependiendo de los recursos que se tengan contemplados para ese determinado periodo, por ejemplo, la cantidad de programadores y artistas dedicados, los fines que se tengan en esa etapa inicial y la solvencia en las capacidades productivas.

Los objetivos de un prototipo pueden variar ya que puede servir para mostrar un juego nuevo a posibles inversores y así ganar el presupuesto de producción, probar

Capítulo 3. Producción de Videojuegos

3.1 Fases de Producción

un estilo o modo de juego innovador dentro de un proyecto estable ya aprobado cambiando el concepto original, funcionar en la producción de diferentes versiones sobre el mismo prototipo que posibilitan mejoras pero conservando la idea original, y como resultado de investigación académica donde los resultados favorecen el ciclo productivo de la industria por medio de ideas completamente nuevas.

El prototipado en los videojuegos es un ciclo muy flexible y enriquecedor ya que tanto puede ser generado basado en una idea inicial sin documentación donde no esté concretado el concepto, sin tener planeadas las fases posteriores y con poco presupuesto ayudando a definir el rumbo del posible juego bajo la idea original del autor, así como también siendo realizado bajo una documentación rigurosa donde ya se tenga definido gran parte del concepto del juego, su tiempo de producción y los elementos mismos del juego con el fin de trabajar sobre una idea ya establecida con rumbo fijo.

El desarrollo de un prototipo puede ser concebido por medio de assets genéricos donde el diseño de estos sea un factor no relevante ya que lo que se busca es tener una idea de la funcionalidad del prototipo y siguiendo esta idea, también se pueden reutilizar *assets* que fueron pensados bajo lineamientos de otra producción para acelerar la creación del prototipo. También es común que en grados iniciales de prototipado se elabore sólo un nivel de juego con el objetivo de captar las funcionalidades principales probando jugabilidad y mecánicas.

En un entorno productivo práctico el diseñador de juego programa la interacción y genera también los *assets* buscando eficientar costos al máximo teniendo de esta manera el control creativo estando seguro de generar la visión correcta.

Así también el control de versiones es un sistema productivo usado para llevar un orden sobre la cantidad de cambios hechos en un ciclo lleno de prototipos.

Producción

En esta fase el prototipo ha sido aprobado por lo que se empieza la fase de producción. La fase de producción es una parte central del desarrollo del juego y todos los requerimientos deben estar claros antes de que comience. Cuando comienza la fase de producción, se espera que cada división, equipo o especialista sepa qué hacer. Esta fase se debe asegurar que el proyecto sea completado bajo el presupuesto y el cronograma planeado. La fase de producción es la parte más larga del desarrollo del juego y puede llevar varios años ya que cada departamento empieza con su cometido, desde creación de elementos de arte, animaciones, sonidos, guiones, hasta piezas de código interactivo. La fase termina con el lanzamiento del juego.

Capítulo 3. Producción de Videojuegos

3.1 Fases de Producción



3.134 Animación 3D en Maya

imagen.winudf.com. (2020). Animación 3D en Mayoa [imagen]. Recuperado 27 Marzo 2020, de <https://imagen.winudf.com/v2/imagen/Y29tLmxtZGdmdmRSLnBsdGZqaGdoLmZiZHJ0bnBkX3NjcmVlbl81XzE1MDk3MDgxNDFFMDUz/screen-5.jpg?fakeurl=1&type=.jpg>



3.135 Container Truck - Zbrush R7

artstation.com/artwork/Arr8V. (2020). Container Truck - Zbrush R7 [imagen]. Recuperado 27 Marzo 2020, de <https://cdnb.artstation.com/p/assets/images/images/000/494/549/medium/cristian-chihaia-carrier-truck-final-s.jpg?1443930988>.



3.136 Punisher Zbrush

artstation.com/artwork/0XBNk4. (2020). Punisher Zbrush [imagen]. Recuperado 27 Marzo 2020, de <https://cdna.artstation.com/p/assets/images/images/016/314/886/medium/marco-plouffe-keos-masons-punisher-z-03-bw.jpg?1551714349>.



3.137 Substance Painter 2020

store.steampowered.com. (2020). Substance Painter 2020 [imagen]. Recuperado 28 Marzo 2020, de https://steamcdn-a.akamaihd.net/steam/apps/1194110/ss_df-ce8bfb82192e21deae83389112b57a1ebaaf4.1920x1080.jpg?t=1578434760.

Capítulo 3. Producción de Videojuegos

3.1 Fases de Producción

Testing Phase

Esta es la fase de pruebas para testear si el juego funciona, además esta fase está linkeada con la fase de producción ya que, si hay alguna falla o variante, se reitera y se realiza el cambio.

Fase Alpha: Es la primera entrega completa del programa terminado de principio a fin, listo para ser probado. Aquí todavía se pueden hacer cambios en el arte y demás departamentos ya que no son los finales, pero cercanos.

Fase Beta: Todas las piezas finales son concebidas en un solo programa que será probado exhaustivamente para buscar errores o bugs antes de dar salida comercial al videojuego.

Fase Gold: El juego está completo y listo para ser enviado a manufactura.

Postproducción

Al final del ciclo de vida de producción, la compañía se preocupa principalmente por identificar y corregir errores en juegos ya lanzados al mercado. Cuando se encuentra un error, se registra en la base de datos de errores, de esta manera se mandan updates, contenido descargable, parches y expansiones para obtener que el juego dure más en su ciclo útil corrigiendo los errores, además se crea el post mortem que es la oportunidad para que la compañía de desarrollo de juegos reflexione sobre la reciente producción del juego y evalúe lo que salió bien y lo que salió mal.

También en esta fase entra el marketing, las ferias de desarrolladores, los artículos y la publicidad a la que estará expuesto el videojuego (muchas de esta publicidad estará desde antes de la salida del videojuego).

Capítulo 3. Producción de Videojuegos

3.2 Roles de desarrollo

3.2 Roles de desarrollo

En los primeros días del desarrollo del juego, las áreas necesarias para creación de videojuegos no eran muy definidas mezclándose unas a otras porque el trabajo del diseñador, programador y creador de contenido podía ser realizado por una sola persona. Cuando la industria evolucionó y se hizo más grande, surgió la necesidad de dividir el trabajo por medio de profesionales especializados.

Estos especialistas normalmente se identifican por su disciplina, y esas disciplinas se definen por el dominio del trabajo dentro del proceso de desarrollo del juego.

Publisher

Es una compañía distribuidora y editora que se encarga de la financiación, manufactura, comercialización, distribución, así como el marketing y la publicidad de un videojuego.

Pueden desarrollar el videojuego internamente en uno de sus estudios o externamente por medio de un Developer.

Algunos de los principales Publishers son:

Tencent Games, Sony, Apple, Microsoft, Activision Blizzard, NetEase, Google, EA, Nintendo, Bandai Namco, Ubisoft, Konami, Capcom, Square Enix entre otros.

Developer

Un **Video Game Developer** es un desarrollador de software especializado en Videojuegos; puede ser un gran estudio con grandes presupuestos y diversas áreas de producción organizadas de acuerdo a los estándares de la industria, así como también pueden ser desarrollos en estudios independientes llamados **Indie** formados por un grupo pequeño de profesionales donde varios roles son compartidos por una o pocas personas ya que estos estudios cuentan con presupuestos inferiores a los grandes developers, teniendo como principal característica la libertad creativa sobre sus creaciones que los diferencia en la industria.

Game Designer

El diseñador de juego genera el concepto y las ideas principales del videojuego y las concreta a través de arte y demos jugables, es el líder creativo en la producción por lo que debe tener conocimientos en gráficos, arte, animación, programación, plataformas y de lo más nuevo en la industria.

Hideo Kojima y Shigeru Miyamoto son diseñadores de juego destacados en la industria.

Capítulo 3. Producción de Videojuegos

3.2 Roles de desarrollo

Game Developer

Término usado para definir a la persona encargada de casi cualquier aspecto de la producción y programación del juego, guiando y tomando el rumbo sobre la idea principal en el aspecto técnico a través del desarrollo.

Creative Director

Desarrolla la parte de como se ve y siente el videojuego, creando congruencia en áreas como el arte, animación y sonido. Está presente en la planeación del *Gameplay* y tiene la capacidad para realizar propuestas que mejoren esta área.

Producer

Es un administrador de proyecto en todas las fases de creación, supervisa el presupuesto y el tiempo de desarrollo de acuerdo a lo planeado para la entrega de un producto final.

Art Director

Está envuelto en casi todas las áreas del proyecto supervisando el proceso de producción artística, desde conceptos visuales previos hasta postproducción y marketing. Trabaja con artistas 2D y 3D, animadores, modeladores y otro personal artístico para coordinar que todos los elementos visuales tengan congruencia y estén unificados creando una guía de estilo que servirá como base a lo largo del desarrollo del juego.

Leads

Son los coordinadores de cada área responsables de guiar a su equipo de trabajo para conseguir los objetivos acordados y ver que se cumplan en tiempo, calidad y presupuesto.

Estos mini equipos de trabajo se dan en áreas como animación, audio, programación y arte.

Audio Director

Es responsable de la creación de toda la banda sonora, desde el diálogo, el ambiente, los efectos de sonido y la música hasta sonidos interactivos para el juego

Animation Director

Es el encargado de supervisar la producción de animaciones 2d o 3d para el juego, teniendo sólidos conocimientos de animación y motion capture, además debe ser crítico y con gran ojo en la interpretación de movimiento basándose en los principios de animación.

Está en contacto cercano con el departamento de arte y *rigging* para la correcta creación de elementos destinados a la animación.

Capítulo 3. Producción de Videojuegos

3.2 Roles de desarrollo

Writer

El encargado de construir los textos y guiones para el juego, puede realizar escritos de narrativa, historia, diálogos, textos para el manual de juego, descripciones e indicaciones técnicas para *Gameplay* y traducciones.

Level Designer

Es el creador de los niveles en los que interactúa el jugador, puede generar desde la parte visual-conceptual, pasando por la estructura del mundo, ubicándolo en tiempo y espacio hasta encontrarse diseñando elementos interactivos del *Gameplay* como acciones y eventos.

System Designer - Mechanics

Realiza las mecánicas de juego para obtener un juego divertido y satisfactorio. Una mecánica de juego es un sistema de reglas que el jugador sigue para obtener resultados y volver a entrar en nuevas mecánicas. Es uno de los pilares del *Gameplay*.

Gameplay Designer

Este profesional programa los diferentes tipos de interacción que el jugador establece en el videojuego incluyendo inteligencia artificial y el comportamiento del personaje en su relación con el mundo, busca en todo momento la mejor jugabilidad.

Interface Designer

El diseñador de interfaz crea las interfaces que permiten interactuar al jugador con el juego, ya que a través de ella el jugador puede entender y decidir sobre las acciones que se encuentran en los mundos previamente pensados. Puede generar interfaces en las áreas de *User Experience (UX)* y *User Interface (UI)*.

Environment Artist

Son los generadores de los mundos que formarán el videojuego, siendo especialistas en modelado 3d con sólidos conocimientos de arquitectura, arte, paisajes, diseño de interiores e iluminación.

Quality Assurance (tester)

El Tester trabaja para el aseguramiento de la calidad, los responsables se encargan de probar por completo el juego y descubrir errores o bugs que pueden ser cambiados, tanto en etapas iniciales como en etapas finales de producción.

Technical Director

Es el encargado de coordinar la parte tecnológica de la producción determinando si es posible su realización, supervisando áreas de programación como Game Engine, físicas, **Gameplay**, etc.

Trabaja siendo responsable por establecer un enlace entre la visión artística y su

Capítulo 3. Producción de Videojuegos

3.2 Roles de desarrollo

implementación técnica generando herramientas específicas para el proyecto.

Programmer

Llamado a menudo programador de videojuegos o desarrollador, es el encargado de medir los requerimientos técnicos que llevará el videojuego en base a las especificaciones de interactividad y mediante diversos lenguajes programación generar código para poder dar instrucciones al *Game Engine* y generar Inteligencia Artificial, físicas, interacciones, *Gameplay*, efectos visuales, gráficos, experiencias online, entre otros.

Concept Artist

En base a los requerimientos conceptuales del videojuego realiza visualizaciones para generar propuestas por medio de dibujos y bocetos sobre ambientes, personajes, objetos y storyboards.

Animator

Es el encargado de dar la ilusión de vida a los personajes u objetos que requieran movimiento en base a principios de animación, además el campo de desarrollo se divide en animación 2d, 3d y **Motion Capture** siendo un estándar en la industria.

3D Modeler

En esta área entran los artistas dedicados al modelado tridimensional poligonal de personajes, *props* y escenarios que se usarán en el juego, usando software como *Maya*, *3ds Max*, *Zbrush* o *Cinema 4d* por citar algunos.

Rigger

Después de tener el objeto modelado, el artista de *rigging* se encarga de crear un sistema de esqueleto dentro del objeto que sostendrá a la geometría 3d para pasársela al área de animación. Cualquier objeto puede contener un rig.

Texture Artist

Genera las texturas que recubrirán los modelos en 3d.

FX Artist

Este artista tiene la función de diseñar y crear efectos especiales como, explosiones, fuego, láseres, partículas, fluidos, fracturas, etc.

Lighting Artist

Recrea los fundamentos reales de la luz para ajustarlos al proyecto y crear ambientes y efectos de luz natural y artificial.

Capítulo 3. Producción de Videojuegos

3.2 Roles de desarrollo

Technical Artist

El artista técnico actúa como un puente entre los artistas y los programadores, asegurando que el contenido y las características del arte se integren fácilmente en el juego, respetando la visión artística y las limitaciones técnicas del proyecto.

Tiene la tarea de optimizar el arte para garantizar que los archivos digitales se ejecuten de manera eficiente dentro de la memoria y las velocidades de carga del motor y la consola. El artista técnico también es responsable de preparar y diseñar sistemas de personajes como esqueletos, telas, emisores de efectos y colisiones.

El trabajo realizado en cualquiera de las áreas que ofrece el desarrollo de juegos debe de seguir metodologías productivas formuladas de forma específica, de ésta manera cada área conserva *workflows* ya comprobados y estandarizados que ayudan a optimizar la creación de *assets* del juego para mantener un ritmo de trabajo efectivo donde el equipo se enfoque en la obtención de resultados de acuerdo a las fases de planeación efectivamente teniendo en mente un producto que cumpla con calidad, sea creado en tiempo, esté dentro del presupuesto destinado y aproveche al máximo las capacidades de todas las partes involucradas, además una característica puntual es que cada área en sí misma tenga capacidad de respuesta ante mejoras y cambios sobre la marcha en diferentes periodos de desarrollo, para este fin los profesionales en el ámbito deben de considerar tener ciertas cualidades que los harán desenvolverse, como una actitud creativa, buena disposición y flexibilidad a cambios, conocimiento de áreas relacionadas, el tener una capacidad de actualización constante, trabajar en equipo y conocimiento así como dominio del idioma Inglés ya que es la forma de comunicación adoptada en la industria.

Otro punto importante que debe de ser cubierto por los profesionales una vez estandarizados los pasos y metodologías individuales que dan forma a cada *asset* del juego es el del conocimiento y dominio del *pipeline* o sistema de trabajo de la producción entera ya que esta parte se da de forma práctica por medio de la comunicación entre áreas y la generación e intercambio de archivos y *assets* de producción, conectando y sincronizando la cadena de procesos, creando un flujo de trabajo bastante atractivo donde cada área necesita de sus contrapartes para continuar dentro de las fases planeadas para dar resultados donde la administración del proyecto pueda ser balanceada y efectiva a lo largo de los ciclos, por lo tanto es común que en el desarrollo de videojuegos, siendo un producto con requerimientos de software se usen metodologías de producción como *Agile* y *Scrum* altamente adoptadas para la ejecución y eficiencia, donde el trabajo es realizado mediante la colaboración de equipos auto organizados y multidisciplinarios, inmersos en un proceso de toma de decisiones a corto plazo.

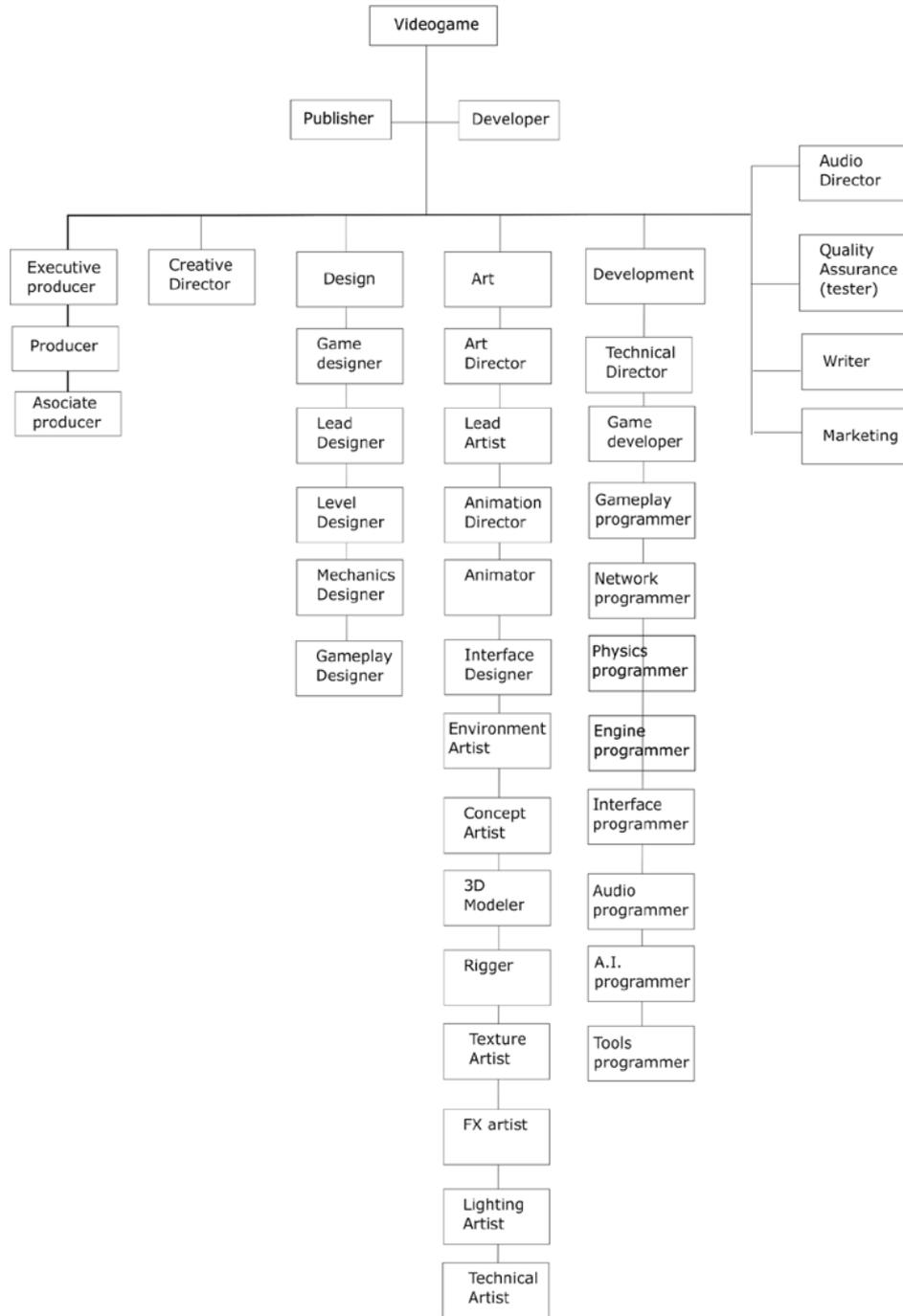
Capítulo 3. Producción de Videojuegos

3.2 Roles de desarrollo

En este sentido tomando en cuenta las posibilidades metodológicas dadas en la actualidad ante el desarrollo tecnológico que facilita la producción a estudios pequeños con presupuestos menores, varios roles en la creación de videojuegos en ocasiones son acaparados por pocas personas para el aprovechamiento del presupuesto al máximo, además de que el mismo factor de desarrollo tecnológico dado por herramientas más poderosas y especializadas han provocado que los roles artísticos y creativos sean influenciados cada vez más por roles donde interviene la programación, generando una necesidad por conocimientos y dominio de la programación. actualidad ante el desarrollo tecnológico que facilita la producción a estudios pequeños con presupuestos menores, varios roles en la creación de videojuegos en ocasiones son acaparados por pocas personas para el aprovechamiento del presupuesto al máximo, además de que el mismo factor de desarrollo tecnológico dado por herramientas más poderosas y especializadas han provocado que los roles artísticos y creativos sean influenciados cada vez más por roles donde interviene la programación, generando una necesidad por conocimientos y dominio de la programación.

Capítulo 3. Producción de Videojuegos

3.2 Roles de desarrollo



3.138 Esquema que describe las jerarquías y los roles de trabajo para la elaboración de un videojuego.

Ortuño F., 2020, Esquema que describe las jerarquías y los roles de trabajo para la elaboración de un videojuego., Mapa Conceptual

Capítulo 3. Producción de Videojuegos

3.3 Recursos Online para aprender a programar

3.3 Recursos Online para aprender a programar

El uso de *Game Engines* con programación visual ha favorecido la creación de videojuegos de principio a fin por parte de personas que no cuentan con conocimientos técnicos en desarrollo de software, solo basta tener interés y curiosidad en el tema y tras consultar la documentación necesaria en forma de manuales y tutoriales, se pueden generar videojuegos sencillos.

Por lo tanto, la programación visual está enfocada para la práctica a nivel introductorio sobre conceptos de pensamiento computacional de forma amigable.

Sin embargo, en el caso específico de haber alcanzado el desarrollo de un juego básico, el siguiente paso lógico es concebir un juego que sea más cercano a los juegos profesionales, aumentando la complejidad en la calidad de la jugabilidad ya que la programación visual genera interacciones preestablecidas que son replicadas por cientos de usuarios a su manera, pero bajo los mismos principios creando juegos parecidos. De esta forma la alternativa que surge para personalizar un juego y volverlo único se da en principal medida por medio de la programación, ya que cada módulo del juego puede ser codificado para expandir sus capacidades, generando enemigos más inteligentes, efectos visuales más impresionantes, personajes con mejores movimientos o misiones más avanzadas por mencionar algunos ejemplos.

De esta forma la programación de videojuegos involucra conocimientos en ingeniería de software, conocimiento avanzado que se enseña en universidades y colegios, sin embargo una manera de acercarse al tema de forma económica y accesible es por medio de los recursos online que internet brinda sobre programación, ya que los recursos en línea desde la década pasada han venido favoreciendo y consolidando un modelo educativo donde el principal objetivo es mostrar alternativas a la enseñanza tradicional, como son el acceso a una gran cantidad de recursos temáticos relacionados, el conocimiento en específico de algún tema de interés comparando varias fuentes, la comodidad y funcionalidad de tener acceso a los cursos desde cualquier computadora con internet no importando la locación, el aprendizaje a ritmo personal ya sea lento o acelerado definiendo horarios adecuados para el estudio, contenidos enriquecido en diferentes formatos multimedia complementando los textos como videos, imágenes, interactivos e hipervínculos, la actualización temática es constante y los costos de las cuotas son generalmente más bajos que en presencial.

A continuación, se presenta un listado de sitios web que ofrecen recursos introductorios para aprender a programar, algunos son plataformas de video tutoriales, otros son *MOOC (Massive Online Open Course)* y otros son sitios interactivos donde el usuario puede practicar sus habilidades para programar por medio de juegos y desafíos con ejemplos de código.

Capítulo 3. Producción de Videojuegos

3.3 Recursos Online para aprender a programar

<p>www.codecademy.com Es una plataforma interactiva en línea que ofrece clases de programación gratuitas y de pago.</p>	<p>www.code.org Organización sin ánimo de lucro que anima a estudiantes a aprender ciencias de la computación, por medio de clases sencillas.</p>
<p>www.codecombat.com CodeCombat es un videojuego que enseña ciencias de la computación, donde los estudiantes escriben código y ven a sus personajes interactuar en tiempo real. www.coursera.org Aprendizaje 100% en línea de las mejores universidades y empresas del mundo. Cada curso es como un libro de texto interactivo, con videos pregrabados, cuestionarios y proyectos.</p>	<p>www.coursera.org Aprendizaje 100% en línea de las mejores universidades y empresas del mundo. Cada curso es como un libro de texto interactivo, con videos pregrabados, cuestionarios y proyectos.</p>
<p>www.khanacademy.org Organización no comercial con la misión de proveer educación de clase mundial gratis para todos, donde sea.</p>	<p>www.lynda.com www.linkedin.com/learning LinkedIn Learning (también Lynda.com) es un sitio estadounidense que imparte video cursos enseñados por profesionales del mundo real.</p>
<p>www.udacity.com Plataforma digital educativa que enseña habilidades tecnológicas que demanda el sector profesional y laboral actual.</p>	<p>www.teamtreehouse.com Escuela online que ofrece cursos sobre tecnología para novatos como avanzados.</p>
<p>www.tynker.com Plataforma educativa de programación para enseñar los principios de programación de juegos y programas.</p>	<p>scratch.mit.edu Scratch es un lenguaje de programación visual, permite el desarrollo de habilidades mentales mediante el aprendizaje de la programación sin tener conocimientos profundos sobre el código.</p>

Capítulo 3. Producción de Videojuegos

3.4 Introducción a la Programación

ocw.mit.edu MIT OpenCourseWare - Iniciativa del Massachusetts Institute of Technology (MIT) para publicar los contenidos educativos de sus cursos de forma gratuita, online.	www.edx.org edx es una plataforma de cursos abiertos masivos en línea, conocidos por sus siglas en inglés como MOOC (Massive Online Open Course)
www.udemy.com Udemy es la plataforma de cursos online más grande del mundo.	

Capítulo 3. Producción de Videojuegos

3.4 Introducción a la Programación

3.4 Introducción a la Programación

La programación es una forma de decirle a las computadoras que hagan una tarea específica. Para que realicen estas tareas, a las computadoras se les debe proporcionar un conjunto de instrucciones escritas en un idioma que ellas puedan entender. Las computadoras entienden las instrucciones escritas mediante una sintaxis específica llamada lenguaje de programación.

Técnicamente un lenguaje de programación es un sistema notacional para describir computaciones en una forma legible tanto para el ser humano como para la máquina. Una computación es una operación o cálculo realizado por la computadora. Por lo tanto, un programador debe generar instrucciones por medio de un lenguaje de programación que puedan ser entendidas y ejecutadas por una computadora. Estas instrucciones reciben el nombre de programa y pueden ser simples como sumar o restar un par de números o complejas donde puedan involucrar una secuencia de instrucciones múltiples.

El dominar un lenguaje de programación involucra desarrollar habilidades para la resolución efectiva de problemas, estas habilidades son adquiridas mediante el denominado pensamiento computacional que engloba cualidades en abstracción, pensamiento algorítmico, automatización, descomposición, depuración y generalización.

La programación es una de las áreas más importantes dentro de la producción de un videojuego ya que está directamente relacionada con la ejecución de las tareas principales que dan forma al producto tales como interacción, Inteligencia Artificial, contenidos Online, efectos visuales - FX, multijugador, entre otros.

Si el diseñador continúa con el deseo de emprender y especializarse en el desarrollo de videojuegos tras utilizar programación visual, lo más probable es que para obtener juegos más avanzados y complejos es que deberá comprender temáticas de programación. Por lo que para motivar a que se continúe investigando sobre el tema en el siguiente apartado se mencionan los elementos básicos que deben conocerse para empezar en el mundo de la programación, dando una explicación sobre el término presentado y su desarrollo mediante pseudocódigo y diagramas de flujo que explican la estructura y el funcionamiento del código.

Capítulo 3. Producción de Videojuegos

3.4 Introducción a la Programación

3.4.1 Componentes básicos de programación

3.4.1 Componentes básicos de programación

Variable

Es un contenedor de datos que estaremos usando después en el programa, puede cambiar la información almacenada a lo largo del programa.

Constante

Es también un contenedor de datos, con la diferencia que conserva la información original y no la cambia a lo largo del programa.

Operadores

Son los elementos que permiten realizar operaciones para calcular datos y definir el flujo de la aplicación.

OPERADORES ARITMÉTICOS

SUMA	+
RESTA	-
MULTIPLICACIÓN	*
DIVISIÓN	/
MÓDULO	%
SUMAR 1	++
RESTAR 1	--

OPERADORES DE ASIGNACIÓN

IGUAL	=
MÁS IGUAL	+=
MENOS IGUAL	-=

LÓGICOS

Operador AND (Y)	&&
Operador OR (O)	
Operador NOT (NO)	!
MAYOR QUE	>
MENOR QUE	<
MAYOR O IGUAL QUE	>=
MENOR O IGUAL QUE	<=
IGUAL A ...	==
DIFERENTE DE...	!=

Capítulo 3. Producción de Videojuegos

3.4 Introducción a la Programación

3.4.1 Componentes básicos de programación

Arrays (Arreglos)

Son estructuras de datos que sirven como un contenedor que nos permite tener diferentes datos al mismo tiempo almacenados.

Function (funciones)

Una función o método es un conjunto de acciones que se van a ejecutar, invocadas con un nombre, facilita realizar tareas en bloque.

Ejemplo:

```
funcion nombre () {  
    acciones a ejecutar  
}  
  
nombre ()
```

Parámetros o Argumentos

En ciertos casos las funciones pueden requerir de datos extra para hacer sus cálculos y ejecutar acciones, estos datos extra son los parámetros. Son datos necesarios que las funciones pueden necesitar para realizar acciones.

Ejemplo:

```
función nombre (parámetro 1, parámetro 2, ...  
parámetro n) {  
    acciones a ejecutar  
}  
  
nombre (param1, param2)
```

Capítulo 3. Producción de Videojuegos

3.4 Introducción a la Programación

3.4.1 Componentes básicos de programación

Estructuras de control:

En lenguajes de programación, las estructuras de control permiten modificar el flujo de ejecución de las instrucciones de un programa.

Las estructuras de control se pueden clasificar en: *secuenciales*, *selectivas* y *repetitivas*. Esta es una de las cosas que permiten que la programación se rija por los principios de la programación estructurada.

Los lenguajes de programación modernos tienen estructuras de control similares. Básicamente lo que varía entre las estructuras de control de los diferentes lenguajes es su sintaxis; cada lenguaje tiene una sintaxis propia para expresar la estructura.

Capítulo 3. Producción de Videojuegos

3.4 Introducción a la Programación

3.4.2 Estructura secuencial

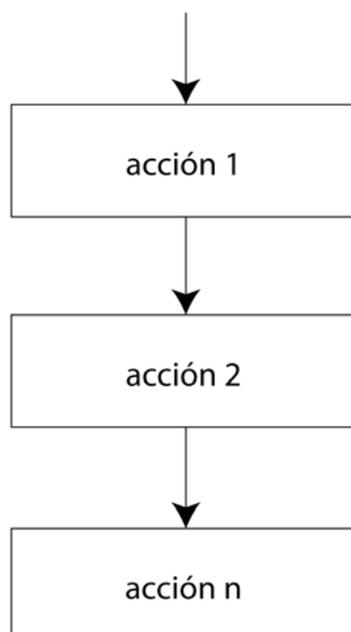
3.4.2 Estructura secuencial

Una estructura secuencial es aquella en la que una acción (instrucción) sigue a otra en secuencia.

Las tareas se suceden de tal modo que la salida de una es la entrada de la siguiente y así sucesivamente hasta el final del proceso.

Ejemplo:

```
inicio
    <acción 1>
    <acción 2>
fin
```



3.139 Diagrama de flujo estructura secuencial

Estructuras secuenciales. Virtual.itca.edu.sv. (2020). Recuperado 13 Mayo 2020, de https://virtual.itca.edu.sv/Mediadores/dlp/23_estructuras_secuenciales.html.

Capítulo 3. Producción de Videojuegos

3.4 Introducción a la Programación

3.4.2 Estructura secuencial

La estructura secuencial es el orden básico de ejecución en los lenguajes de programación, y su orden es de arriba a abajo, siendo la base para la escritura de algoritmos.

Se representa gráficamente como una caja después de otra, ambas con una sola entrada y una única salida.

De esta forma en las estructuras secuenciales una instrucción no se ejecuta hasta que finaliza la anterior, es así que las instrucciones se suceden de tal forma que el resultado de la anterior puede afectar la siguiente.

Ventajas en el uso de la estructura secuencial	Desventajas en el uso de la estructura secuencial
<p>Los programas son más fáciles de leer o entender.</p> <p>La estructura de los programas es clara, puesto que las instrucciones están ligadas entre sí.</p> <p>Existe una reducción de esfuerzo en las pruebas y depuración y en los costos de mantenimiento.</p> <p>Los programas son más sencillos y rápidos de confeccionar.</p>	<p>El principal inconveniente es que se obtiene un único bloque del programa, que cuando se hace demasiado grande puede resultar problemático para el manejo del código fuente.</p>

Capítulo 3. Producción de Videojuegos

3.4 Introducción a la Programación

3.4.3 Estructuras Selectivas

3.4.3 Estructuras Selectivas

Las estructuras selectivas se utilizan para tomar decisiones lógicas; de ahí que se suelen denominar también estructuras de decisión o alternativas.

En las estructuras selectivas se evalúa una condición y en función del resultado de la misma se realiza una opción u otra. Las condiciones se especifican usando expresiones lógicas.

Los tipos de declaración selectiva son:

if, if/else, if/else/if, switch

if

La estructura alternativa simple if ejecuta una determinada acción cuando se cumple una determinada condición.

Ejemplo:

Si la condición es verdadera, entonces ejecuta la acción S1
Si la condición es falsa, entonces no hacer nada.

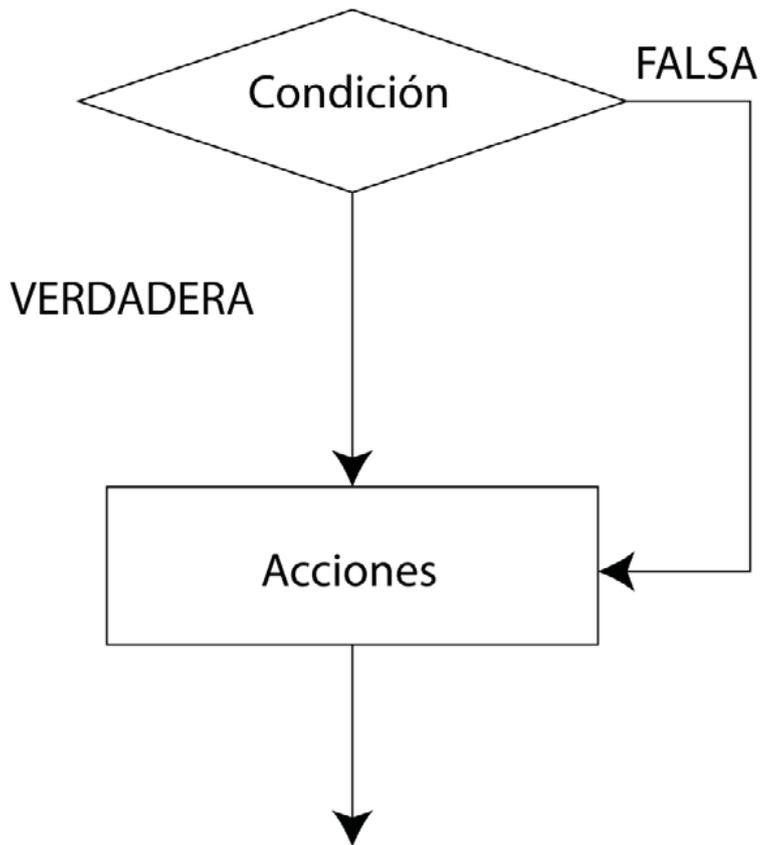
Ejemplo en pseudocódigo en español:

```
si <condición> entonces  
<acción S1>  
fin _ si
```

Ejemplo en pseudocódigo en inglés:

```
if <condición> then  
<acción S1>  
endif
```

Capítulo 3. Producción de Videojuegos
3.4 Introducción a la Programación
3.4.3 Estructuras Selectivas



3.140 Diagrama de flujo de la estructura selectiva simple if

González Bustamante, O. (2007). Guías y Textos de Cómputo, Introducción a la Programación (p. 36). UNAM.

Capítulo 3. Producción de Videojuegos

3.4 Introducción a la Programación

3.4.3 Estructuras Selectivas

if - else

Si la condición es verdadera, se ejecuta la acción S1 y, si es falsa, se ejecuta la acción S2.

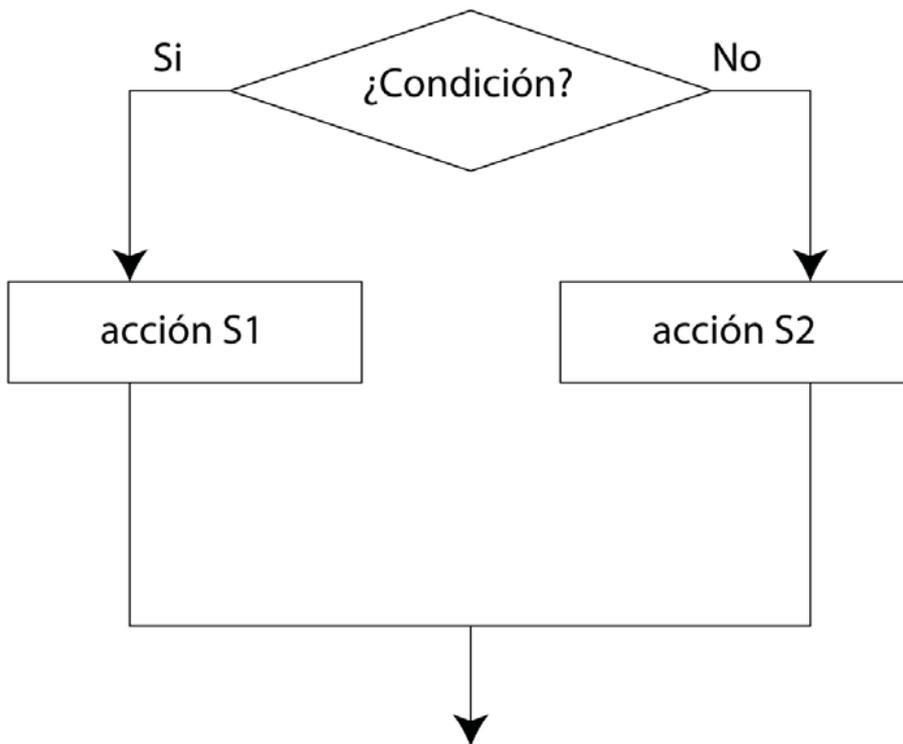
Ejemplo en pseudocódigo en español:

```
si <condicion> entonces  
<accion S1>  
si _ no  
<accion S2>  
fin _ si
```

Ejemplo en pseudocódigo en inglés:

```
if <condicion> then  
<accion S1>  
else  
<accion S2>  
endif
```

Capítulo 3. Producción de Videojuegos
3.4 Introducción a la Programación
3.4.3 Estructuras Selectivas



3.141 Diagrama de flujo de la estructura selectiva doble if – else

González Bustamante, O. (2007). Guías y Textos de Cómputo, Introducción a la Programación (p. 37). UNAM.

Capítulo 3. Producción de Videojuegos

3.4 Introducción a la Programación

3.4.3 Estructuras Selectivas

if - else- if

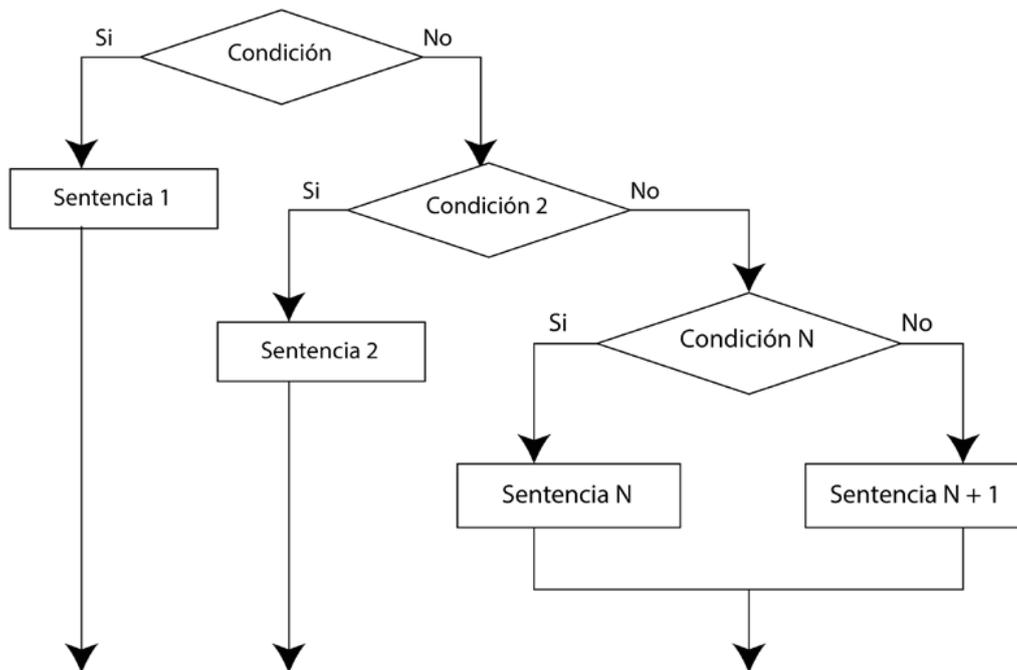
Condiciones que se tienen hasta que se logre cumplir el objetivo.

Se pueden agregar tantas veces se necesite tener un sí.

Ejemplo:

```
si (condición 1) entonces
    Sentencias 1;
Sino
    Si (condición 2) entonces
        Sentencias 2;
    Sino
        Si (condición N) entonces
            Sentencias N;
        Sino
            Sentencias N+1;
        Fin Si
    FinSi;
FinSi;
```

Capítulo 3. Producción de Videojuegos
3.4 Introducción a la Programación
3.4.3 Estructuras Selectivas



3.142 Diagrama de flujo de la estructura de selección múltiple if - else - if

González Bustamante, O. (2007). Guías y Textos de Cómputo, Introducción a la Programación (p. 41). UNAM.

switch

La estructura de decisión múltiple evaluará una expresión que podrá tomar n valores distintos, 1, 2, 3, 4, ..., n . Según que elija uno de estos valores en la condición, se realizará una de las n acciones, o lo que es igual, el flujo del algoritmo seguirá un determinado camino entre los n posibles.

Capítulo 3. Producción de Videojuegos

3.4 Introducción a la Programación

3.4.3 Estructuras Selectivas

Ejemplo 1:

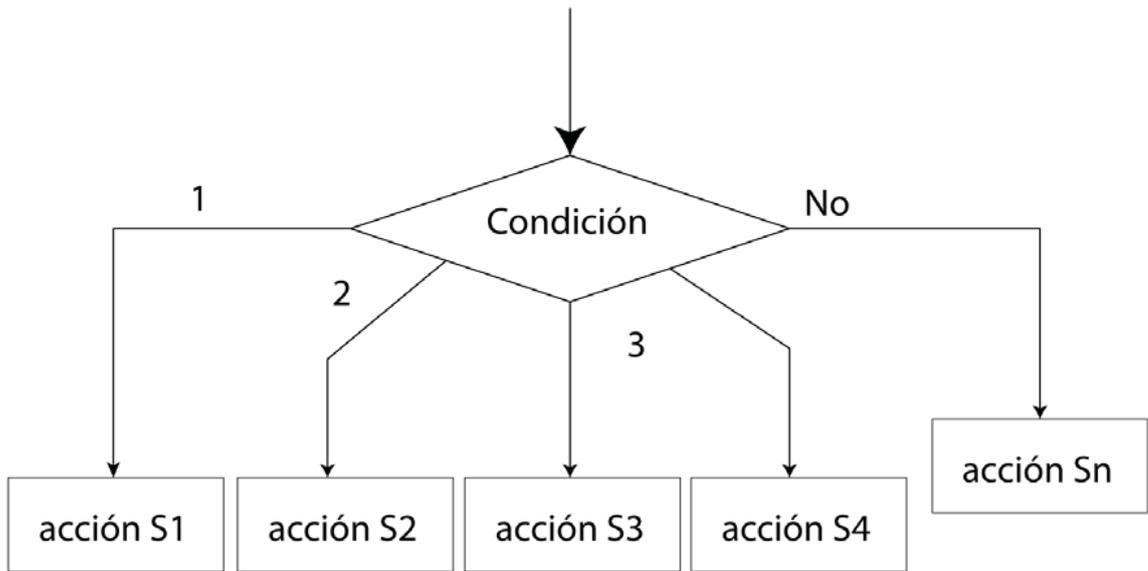
```
switch (expresión)
{
case valor1:
sentencia1;
sentencia2;
sentencia3;
break;
case valor2:
sentencia1;
sentencia2;
sentencia3;
break;
case valor3:
sentencia1;
sentencia2;
sentencia3;
break;
default:
sentencia1;
sentencia2;
sentencia3;
} // fin de la sentencia compuesta
```

Ejemplo 2

```
según_sea expresion (E) hacer

e1: accion S11
      accion S12
      accion S1a
e2: accion S21
      accion S22
      accion S2b
en: accion S31
      accion S32
      accion S3p
si-no
      accion Sx
fin_según
```

Capítulo 3. Producción de Videojuegos
3.4 Introducción a la Programación
3.4.3 Estructuras Selectivas



3.143 Diagrama de flujo de la estructura de selección múltiple switch

González Bustamante, O. (2007). Guías y Textos de Cómputo, Introducción a la Programación (p. 52). UNAM.

Capítulo 3. Producción de Videojuegos

3.4 Introducción a la Programación

3.4.4 Estructuras repetitivas

3.4.4 Estructuras repetitivas

Se describe y analiza el concepto de bucle como la sección de código que se repite y que se denomina así ya que cuando termina la ejecución de la última sentencia el flujo de control vuelve a la primera sentencia y comienza otra repetición de las sentencias del código. Cada repetición se conoce como iteración o pasada a través del bucle.

Estructuras repetitivas usadas son:

while, do-while, for

while

La estructura repetitiva mientras (en inglés *while*) es aquella en que el cuerpo del bucle se repite mientras se cumple una determinada condición.

Cuando se ejecuta la instrucción *mientras*, la primera cosa que sucede es que se evalúa la condición (una expresión booleana). Si se evalúa falsa, no se toma ninguna acción y el programa prosigue en la siguiente instrucción del bucle.

Si la expresión booleana es verdadera, entonces ejecuta el cuerpo del bucle, después de lo cual se evalúa de nuevo la expresión booleana. Este proceso se repite una y otra vez mientras la expresión booleana (condición) sea verdadera.

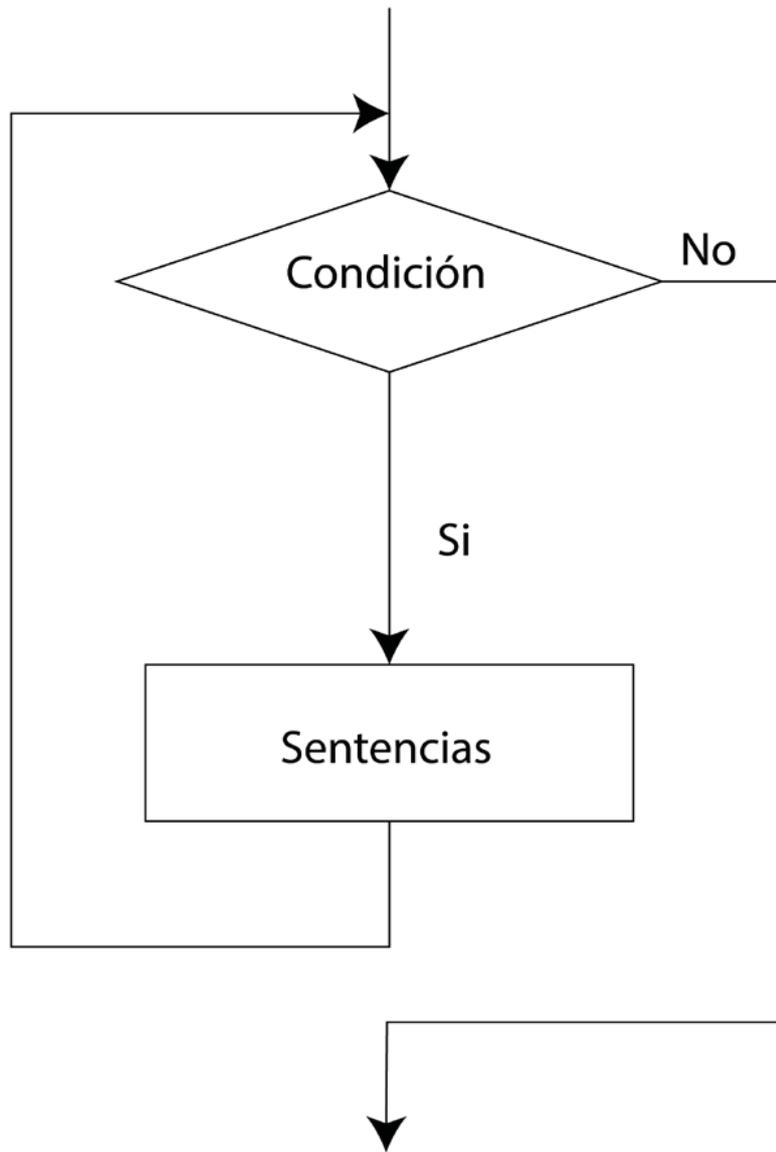
Ejemplo de pseudocódigo en español:

```
mientras condicion hacer
accion S1
accion S2
    acción Sn
fin _ mientras
```

Ejemplo de pseudocódigo en inglés:

```
while condicion do
<acciones>
endwhile
```

Capítulo 3. Producción de Videojuegos
3.4 Introducción a la Programación
3.4.4 Estructuras repetitivas



3.144 Diagrama de flujo de la estructura repetitiva while

González Bustamante, O. (2007). Guías y Textos de Cómputo, Introducción a la Programación (p. 44). UNAM.

Capítulo 3. Producción de Videojuegos

3.4 Introducción a la Programación

3.4.4 Estructuras repetitivas

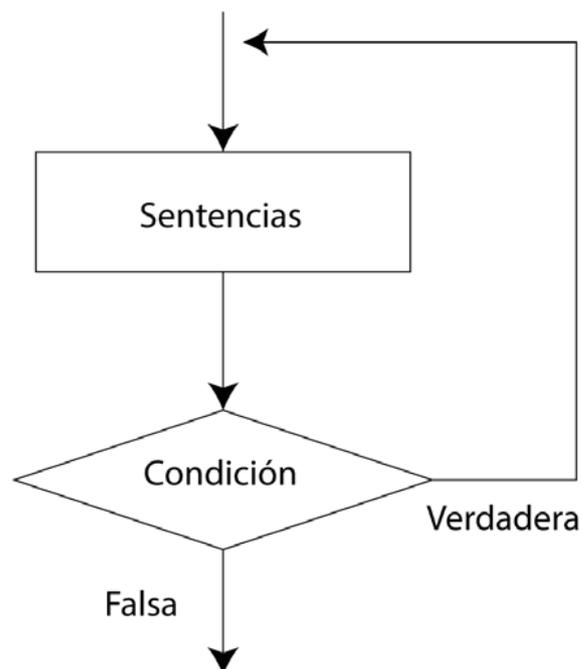
do-while

El bucle do-while es análogo al bucle while y el cuerpo del bucle se ejecuta una y otra vez mientras la condición (expresión booleana) sea verdadera.

Existe, sin embargo, una gran diferencia y es que el cuerpo del bucle está encerrado entre las palabras reservadas do y while, de modo que las sentencias de dicho cuerpo se ejecutan, al menos una vez, antes de que se evalúe la expresión booleana. En otras palabras, el cuerpo del bucle siempre se ejecuta, al menos una vez, incluso aunque la expresión booleana sea falsa.

Ejemplo de pseudocódigo:

```
hacer  
<acciones>  
mientras (<expresión>)
```



3.145 Diagrama de flujo de la estructura repetitiva do – while

Joyanes Aguilar, Luis. (2008). Fundamentos de Programación, Algoritmos, estructura de datos y objetos. (p.166) 4a Ed. España. McGraw-Hill.

Capítulo 3. Producción de Videojuegos

3.4 Introducción a la Programación

3.4.4 Estructuras repetitivas

for

En muchas ocasiones se conoce de antemano el número de veces que se desean ejecutar las acciones de un bucle.

En estos casos, en el que el número de iteraciones es fijo, se debe usar la estructura for

La estructura for ejecuta las acciones del cuerpo del bucle un número especificado de veces y de modo automático controla el número de iteraciones o pasos a través del cuerpo del bucle.

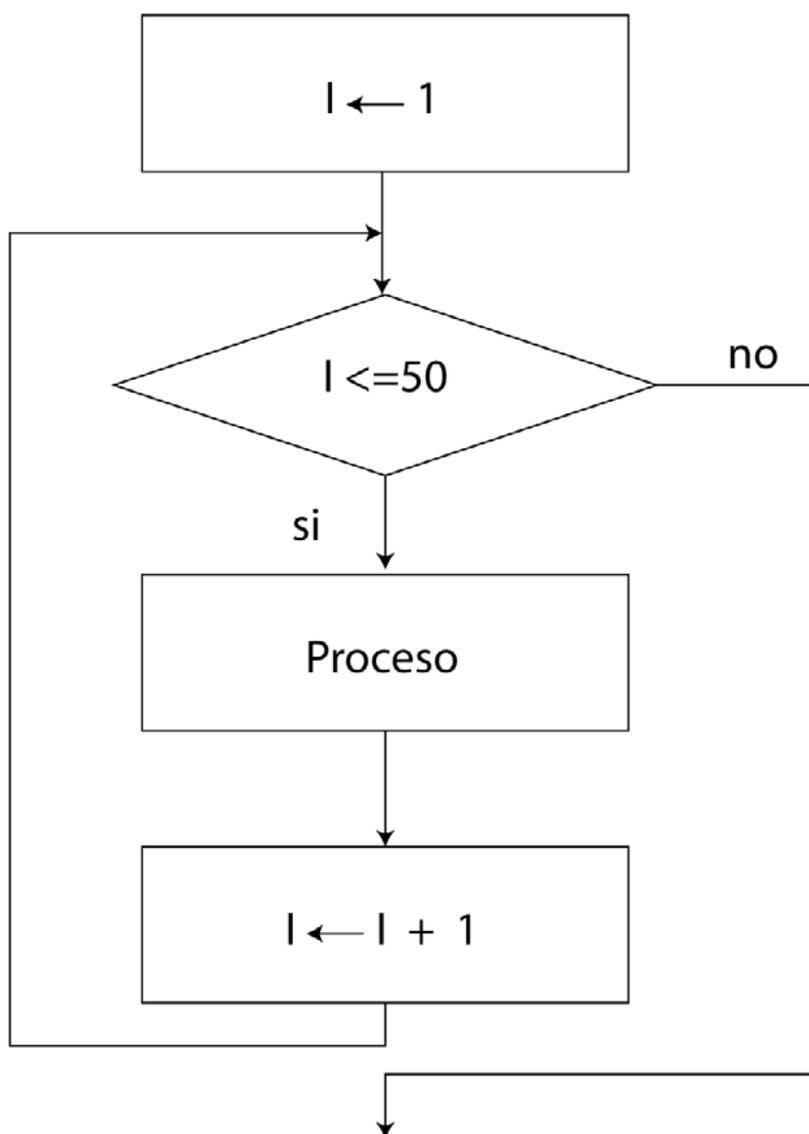
Ejemplo de pseudocódigo 1:

```
v: variable indice
vi, vf: valores inicial y final de la variable
desde v ← vi hasta vf [incremento incr] hacer
<acciones>
fin_desde
```

Ejemplo de pseudocódigo 2:

```
desde i ← 10 hasta 20 hacer
<acciones>
fin_desde
```

Capítulo 3. Producción de Videojuegos
3.4 Introducción a la Programación
3.4.4 Estructuras repetitivas



3.146 Diagrama de flujo de la estructura repetitiva for

Joyanes Aguilar, Luis. (2008). Fundamentos de Programación, Algoritmos, estructura de datos y objetos. (p.171) 4a Ed. España. McGraw-Hill.

Capítulo 3. Producción de Videojuegos

3.5 Software para elaborar arte para videojuegos (Game Art)

3.5 Software para elaborar arte para videojuegos (Game Art)

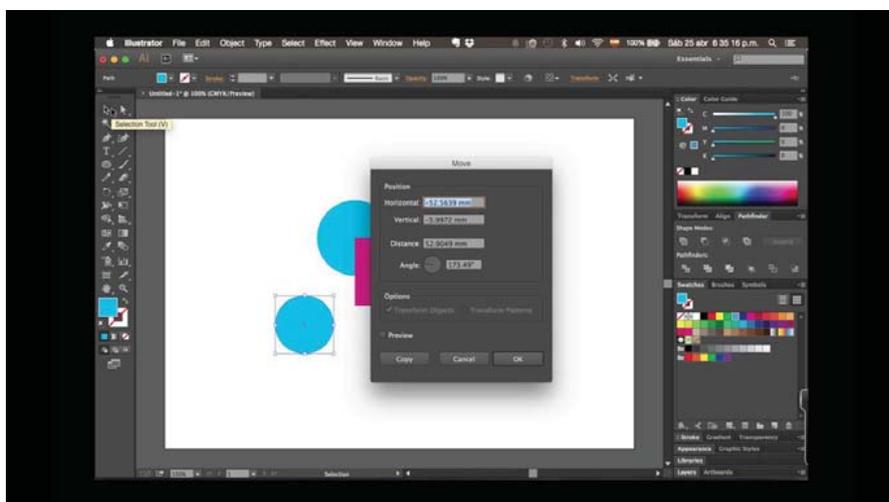
Una vez cubiertos los requerimientos conceptuales y de planeación, el proyecto pasa a la fase de producción, en esta fase la creación de los elementos del juego se da a través del equipo de *Game Art* ajustándose a los requerimientos técnicos del Engine para que el juego funcione fluidamente, así como el *look and feel* a lo largo del proyecto.

Cada año el software de creación digital lanza novedades y actualizaciones que mejoran el rendimiento y las capacidades de producción creativa, siendo un factor principal dentro del éxito del mercado de videojuegos.

Aunque el mercado de software está dominado por las grandes compañías con licencias propietarias costosas como Adobe y Autodesk, poco a poco el entorno productivo a visto surgir software Open Source que funciona de forma similar como Blender, Inkscape o ArmorPaint teniendo la ventaja de ser gratuito, generando una alternativa real posibilitando la creación de elementos de Arte.

Diseño 2d

El diseño en dos dimensiones engloba el uso de software especializado para la creación de imágenes en formato bitmap y vector. Es el área más genérica de producción de imágenes y en la industria se usa para la materialización de ideas por medio de concept art en diseño de personajes, escenarios, vehículos, objetos, así como la planeación de secuencias en storyboards, interfaces de usuario, head up displays, menus, logos, texturas 3d y creación de sprites para animación.



3.147 Illustrator CC

Illustrator CC. (2020). Illustrator CC [imagen]. Recuperado 28 Marzo 2020, de <https://i.ytimg.com/vi/qz4jX24qE0Y/maxresdefault.jpg>.

Capítulo 3. Producción de Videojuegos

3.5 Software para elaborar arte para videojuegos (Game Art)

Software utilizado para la creación de imágenes en mapa de bits - ráster

Adobe Photoshop CC	Pixlr
GIMP	Krita
Corel PHOTO – PAINT	Sumopaint
Corel Painter 2020	Affinity Photo

Software utilizado para la creación de imágenes vectoriales - vector

Adobe Illustrator CC	Affinity Designer
InKscape	Vectr
CorelDRAW 2019	BOXY SVG

Animación 2d

Este tipo de animación tradicional se genera por medio de secuencias de imágenes continuas que contienen dibujos y compiladas unas tras otras simulan movimiento.

Este tipo de animación puede ser usada tanto en videos o gifs promocionales como dentro del juego mismo por medio de sprites, animando y dando vida a los personajes, enemigos, así como generando los efectos visuales como explosiones, fuego, viento, agua, poderes especiales, armas, y efectos de iluminación en el ambiente.

Software utilizado para la creación de animaciones 2d

Moho	Cel Action 2d
Toon Boom Harmony	Pencil 2d
OpenToonz	FlipBook



3.148 Harmony

blog.toonboom.com. (2020). Harmony 14 [imagen]. Recuperado 28 Marzo 2020, de https://blog.toonboom.com/hubfs/blog/blogpost_visual/har14-announcement/HAR14_spanish.jpg.

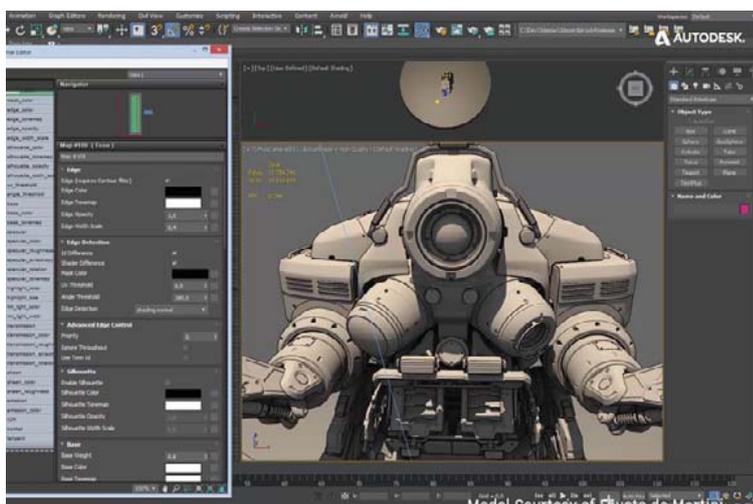
Capítulo 3. Producción de Videojuegos

3.5 Software para elaborar arte para videojuegos (Game Art)

Diseño y animación 3d

El 3d es la representación en tercera dimensión de un objeto por medio de puntos en el espacio, conectados por entornos geométricos en un entorno virtual.

Los objetos generados por el software de 3d pueden ser animados y renderizados dando como resultado secuencias de imágenes usadas en videos y en impresos, así también los objetos 3d y sus animaciones pueden ser usados en Game Engines en un entorno de rendero en tiempo real dando pauta a la interactividad. Algunos productos generados por medio del 3d son: personajes, escenarios, edificios, vehículos, objetos, vegetación, efectos visuales como explosiones, fluidos y partículas, además estos objetos pueden animarse creando animaciones faciales, transformaciones (*morphs*), *walk cycles* y gráficos en movimiento.



3.149 3ds Max 2020

709mediaroom.com. (2020). 3ds Max 2020 [imagen]. Recuperado 28 Marzo 2020, de https://www.709mediaroom.com/wp-content/uploads/2019/04/autodesk3dsmax2020_002.jpg.

Software utilizado para la creación de objetos y animaciones 3d

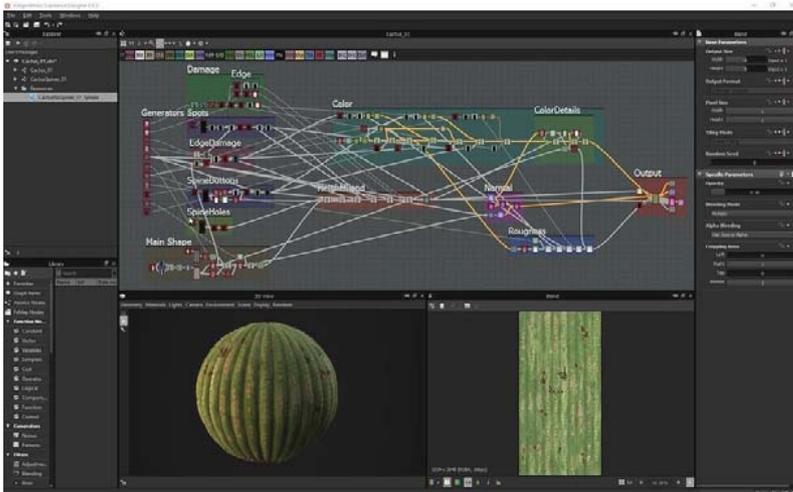
Autodesk Maya	Houdini
Autodesk 3DS Max	Daz 3d
Maxon Cinema 4d Blender	Blender
Fusion 360	Moment of Inspiration (MOI)
Modo	Zbrush
Mudbox	Sketchup
iClone	Adobe Fuse
Mixamo	Marvelous Designer

Capítulo 3. Producción de Videojuegos

3.5 Software para elaborar arte para videojuegos (Game Art)

Texturizado 3d

El texturizado 3d comprende el añadir texturas y materiales a los modelos 3d creados previamente, volviendo al objeto más estético y creíble, en ocasiones con acabados fotorrealistas.



3.150 Substance Designer

forums.cubebrush.co. (2020). Substance Designer [imagen]. Recuperado 28 Marzo 2020, de <https://i.pinimg.com/originals/74/fb/a0/74fba0cd810ed05a-c34a662e431e2690.jpg>.

Software utilizado para la creación de texturas en 3d

Substance Designer	Body Paint 3d
Substance Painter	Quixel Suite
Mari	ArmorPaint

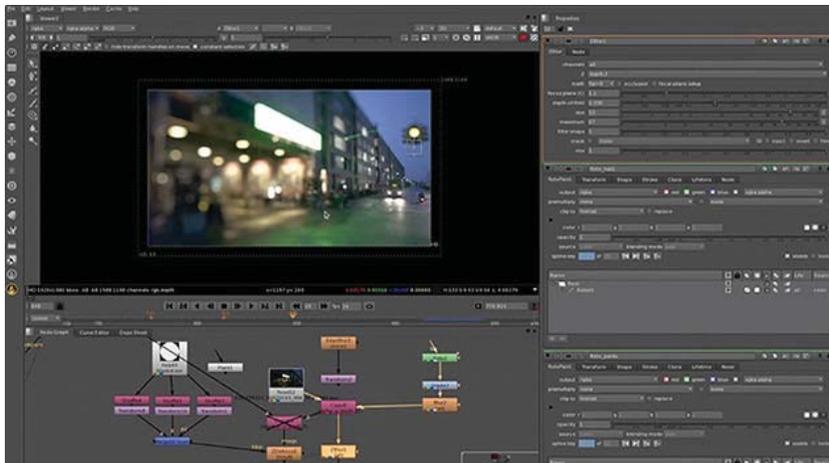
Edición de Video y VFX

El video es la producción de imágenes en secuencia para la representación de escenas en movimiento y estas escenas son complementadas por medio de composiciones de efectos visuales (VFX) que incrementan su valor estético.

La parte de marketing enfocada a video es usada para mostrar intros, promos, e historias sobre los juegos.

Capítulo 3. Producción de Videojuegos

3.5 Software para elaborar arte para videojuegos (Game Art)



3.151 The Foundry Nuke
studiodaily.com. (2020). The Foundry Nuke [imagen]. Recuperado 28 Marzo 2020, de https://www.studiodaily.com/wp-content/uploads/2012/11/630_nuke7.jpg.

Software utilizado para la creación de video

Adobe Premier Pro	Flame
After Effects	Fusion 16
Nuke	Natron
Pinnacle Studio	Shotcut

Capítulo 3. Producción de Videojuegos

3.6 Uso de Construct 2 para la creación de un videojuego

3.6 Uso de Construct 2 para la creación de un videojuego

En el inicio del proyecto surgieron dificultades que provocaban que la creación del mismo se viera algo lejano de realizar por la escasa documentación, el conocimiento desestructurado y desordenado y tutoriales costosos, además se tocaba con frecuencia un tema que no se llegaba a comprender de forma certera: la programación. En ésta búsqueda por el conocimiento en desarrollo de juegos se consideró en primer término el software *Unreal Engine* ya que al ser gratuito y tener los mejores gráficos de su tipo generaba gran atención, sin embargo las dificultades empezaron cuando la función principal que es la interacción se da por medio de la programación, actividad bastante compleja que al tener desconocimiento de su funcionamiento desestimó esfuerzos; posteriormente buscando alternativas se usó el *Engine Unity* ya que también es gratuito, sin embargo aunque su interfaz es un poco más amigable y tiene una tienda en línea con gran cantidad de *assets* preprogramados nuevamente surgió la barrera creativa por el uso de programación para dar vida a la interacción, por lo que al no haber alcanzado algo concreto con esos aproximamientos y después de no ver el avance esperado, la estrategia se modificó en el sentido de realizar una investigación más general sobre *Engines* más sencillos y fáciles de usar, así que el objetivo cambió a realizar un juego claramente funcional con todos sus requerimientos estructurales principales; en este sentido el Engine encontrado para realizar esta tarea fue *Construct 2*, un software que desde el primer uso deja en claro cuáles son sus funciones principales por medio de una interfaz sumamente sencilla, fácil de entender y amigable aún para quien nunca ha usado un *Game Engine*, de esta forma se crearon las primeras interacciones desde cero de forma intuitiva, obteniendo una motivación por lograr conseguir el concretar un juego de principio a fin.

En los párrafos siguientes se describirán las características más importantes de este divertido software, así como sus puntos fuertes y limitaciones.

Construct 2 fue puesto a la venta en 2011, y actualmente se cuenta con la última versión, salida al mercado en 2017 llamada *Construct 3*, que basada en el mismo principio que su predecesora, incluye soporte para nuevas plataformas.

Tiene 3 licencias de uso: Free Edition, Personal License (\$3,599.90 USD) y Business License (\$9,499.90 USD).

La promesa de venta es que se pueden realizar prototipos jugables en horas en lugar de semanas o meses.

La página oficial de *Construct 2* es <https://www.construct.net>

Capítulo 3. Producción de Videojuegos

3.6 Uso de Construct 2 para la creación de un videojuego

Construct 2 es un Game Engine que permite el desarrollo de videojuegos por medio de programación visual, desarrollado por la compañía Scirra Ltd., está basado en el lenguaje de marcado HTML5 funcionando en gran variedad de dispositivos.

Este Game Engine está diseñado únicamente para la producción de juegos 2D y su uso está enfocado a usuarios no programadores.

La interacción necesaria para lograr que el juego funcione se da por el uso de instrucciones en la sección de hojas de eventos (event sheets), esta es la sección principal del *Engine* ya que aquí es donde el usuario genera las reglas de interacción que gobernarán a los personajes y enemigos, por lo que para lograr este objetivo se usan eventos, condiciones y acciones sobre los objetos del juego, generando lógica del juego basándose en estructuras de programación usadas en el desarrollo de software como **while**, **for**, **if-else** entre otras, de esta forma se generan sistemas avanzados de lógica visual de forma intuitiva, fácil y rápida para la creación de prototipos.

Otra parte importante dentro del esquema de *Construct 2* son los comportamientos (*behaviors*), denominadas como acciones preestablecidas que se le asignan a los personajes u objetos del juego para que por medio de unos cuantos *clicks* y sin crear una sola línea de código estos elementos del juego adquieran propiedades interactivas avanzadas en varias categorías como son movimiento, velocidad, opacidad, color, *pathfinding* y *scroll* por citar algunos, permitiendo al usuario experimentar y jugar creativamente sin preocuparse por generar código laborioso; por ejemplo con el uso del *behavior Platform*, el objeto adquiere características del género de videojuegos *platformer* con movimientos de izquierda - derecha y salto sobre una plataforma específica; algunos otros comportamientos usados son *Car*, *Bullet*, *8 Direction* y *Rotate*.

Construct 2 tiene la capacidad de exportar para las siguientes plataformas:

HTML5 Navegador	Windows
Chrome Web Store	Mac
Facebook	Linux
Kongregate	Windows 8 Store
NewGrounds	Windows Phone 8 app
Firefox Marketplace	IOS
Scirra Arcade	Android

Capítulo 3. Producción de Videojuegos

3.6 Uso de Construct 2 para la creación de un videojuego

Tutoriales recomendados para dominar el software

Udemy - The Complete Game Developer course - Build 60 Games	Learn Construct 2 Creating a Pixel Platformer in HTML5 - Jimmy Alexander
Richard Sneyd - Learn Easy HTML5 Game Development in Construct 2	How to Create HTML5 And Mobile Games Using Construct 2 - C2 TUTS
Platform Game Creation with Construct 2 (HTML5)	Digital Tutors - Developing HTML5 Games with Construct 2
Canal de YouTube: Marcos Game Dev Recriando Megaman X Seção 1 - Movimentos Básicos do Megaman Recriando Megaman X Seção 2 - Chefes Recriando Megaman X Seção 3 - Armas Recriando Megaman X Seção 4 - Pedidos	Canal de Youtube: Construct Tutorials - by CKenYoshida [8-bit Glory]: MEGAMAN - Construct2

Libros recomendados

HTML5 Game Development from the Ground Up with Construct 2 - Roberto Dillon	Game Development with Construct 2, From Design to Realization - Lee Stemkoski, Evan Leider
Level 0 - Ankur Prasad, Allen Wu	Construct 2 Game Development by Example - John Bura
Learning Construct 2 - Aryadi Perwira Subagio	

Al usar Construct 2 por medio de la programación visual basada en eventos se obtiene una ventaja teórica sobre la introducción a la programación, ya que por medio de la práctica quedarán en claro las acciones y pasos necesarios a seguir para producir interacción haciendo referencia a estructuras básicas de programación, por lo que el diseñador podrá enfrentarse y resolver problemas reales con estructuras de control que cualquier lenguaje de programación utiliza y dependerá de la práctica, creatividad y documentación que tenga, para lograr productos complejos

Capítulo 3. Producción de Videojuegos

3.6 Uso de Construct 2 para la creación de un videojuego

e interesantes.

Otra ventaja es que se podrán generar de forma rápida y económica prototipos jugables que pueden ser modificados fácilmente, sirviendo como vaciado de ideas teniendo la ventaja que se podrá definir si el juego contiene un *Gameplay* de calidad, si el juego es divertido ya que una vez definido, aceptado y testeado el prototipo, se podrá crear por completo con un mayor presupuesto involucrando a desarrolladores de forma individual o colectiva donde la calidad del juego y por ende el éxito del mismo estará sujeto a varios factores como planeación, inversión, capacidades técnicas, márketing e idea original.

En cuanto al alcance que se puede lograr con *Construct 2*, es estandarizado ya que se puede generar un producto que cumpla con el nivel de aceptación suficiente que exige Apple Store o Google Play en sus tiendas virtuales permitiendo la monetización.

En síntesis, algunos puntos destacados que posicionan a *Construct 2* como uno de los *Game Engines* más accesibles y fáciles de usar del mercado son:

La iniciación en el mundo de la creación de videojuegos de forma rápida y sencilla, concretando conceptos e ideas, motivando al creador a seguir adelante.

La producción de un videojuego sin el requisito de tener conocimiento de lenguajes de programación complejos.

La práctica de los principios de programación de forma introductoria, divertida y atractiva.

El desarrollo de prototipos rápidos para la visualización del funcionamiento del juego, ahorrando presupuesto, esfuerzo y tiempo.

Capítulo 3. Producción de Videojuegos

3.7 Estructuras de control aplicadas a Construct 2

3.7 Estructuras de control aplicadas a Construct 2

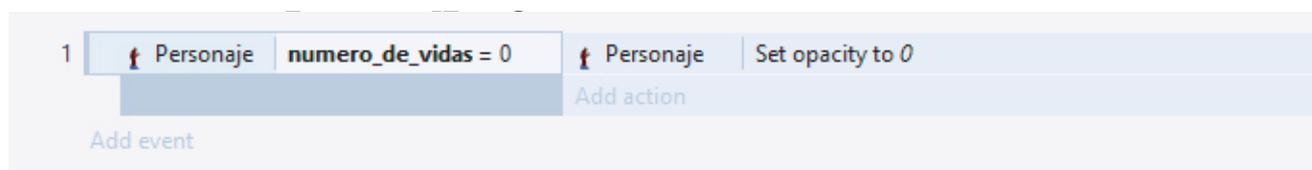
Para una mejor comprensión en el funcionamiento de las estructuras de control se presentará una analogía funcional entre las instrucciones que se generan por medio de programación tradicional y sus equivalentes en bloques visuales en *Construct 2*.

De esta forma el usuario que asimile las comparaciones debe entender la lógica interactiva que se da por la relación de bloques ya que la sintaxis está implícita para el software no para el usuario.

A nivel inicial esta práctica comparativa es benéfica en la producción de un videojuego porque da un orden a las instrucciones lógicas que se quieren generar incrementando la velocidad de producción, minimizando el proceso de prueba error y creando un sentimiento favorable en el usuario.

Cabe mencionar que sólo con constante práctica el usuario logrará comprender los fundamentos del pensamiento computacional.

Explicación de las estructuras de control básicas por medio de eventos



3.152 Hoja de Eventos de Construct 2

Construct 2: Game Making Software (Version Release 216 (64-bit)) [Computer software]. (n.d.). Recuperado September 30, 2021, de <https://www.construct.net/> Copyright @ 2014 Scirra Ltd All rights reserved Built at 13:46:39 en Oct 22 2015

La acción en la hoja de eventos dice:

Si el Personaje tiene un numero_de_vidas igual a cero, bajar la opacidad del Personaje a cero por ciento.

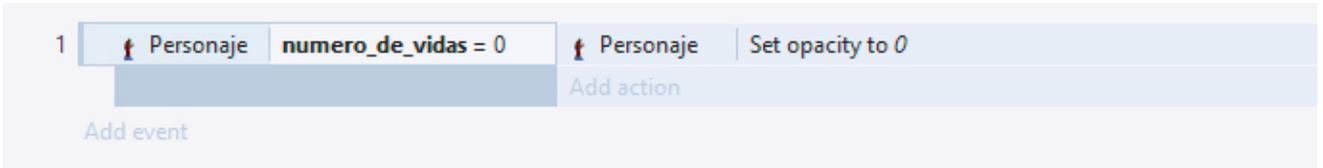
Este ejemplo en pseudocódigo significa:

```
if (el numero_de_vidas del Personaje es igual a cero) {  
  bajar la opacidad del Personaje a cero por ciento  
};
```

Capítulo 3. Producción de Videojuegos

3.7 Estructuras de control aplicadas a Construct 2

Estructura IF-ELSE en Construct 2



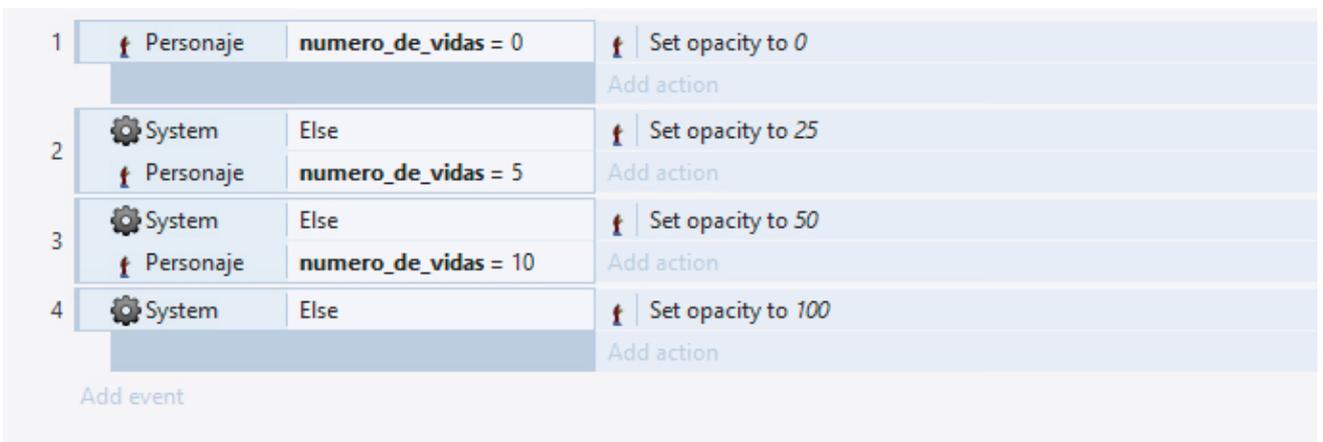
La acción en la hoja de eventos dice:

Si el Personaje tiene un numero_de_vidas igual a cero, bajar la opacidad del Personaje a cero por ciento; de otro modo, si el Personaje tiene un numero_de_vidas diferente a cero, subir la opacidad del Personaje a cien por ciento.

Este ejemplo en pseudocódigo significa:

```
if (el numero_de_vidas del Personaje es igual a cero) {  
  bajar la opacidad del Personaje a cero por ciento  
} else {  
  subir la opacidad del Personaje a cien por ciento  
};
```

Estructura IF-ELSE-IF en Construct 2



Capítulo 3. Producción de Videojuegos

3.7 Estructuras de control aplicadas a Construct 2

La acción en la hoja de eventos dice:

Si el Personaje tiene un numero_de_vidas igual a cero, bajar la opacidad del Personaje a cero por ciento.

De otro modo, si el Personaje tiene un numero_de_vidas igual a cinco, poner la opacidad del Personaje a veinticinco por ciento.

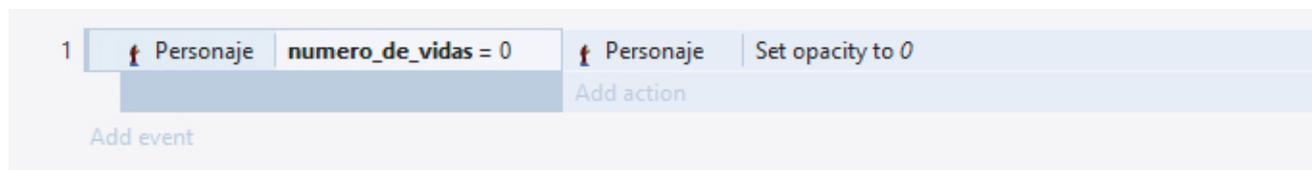
De otro modo, si el Personaje tiene un numero_de_vidas igual a diez, poner la opacidad del Personaje a cincuenta por ciento.

De otro modo, cualquier otra cifra diferente a las especificadas anteriormente para numero_de_vidas, poner la opacidad del Personaje a cien por ciento.

Este ejemplo en pseudocódigo significa:

```
if (el numero_de_vidas del Personaje es igual a cero) {
    bajar la opacidad del Personaje a cero por ciento
} else if (el numero_de_vidas del Personaje es
    igual a cinco) {
    poner la opacidad del Personaje a
    veinticinco por ciento
} else if (el numero_de_vidas del
    Personaje es igual a diez) {
    poner la opacidad del Personaje
    a cincuenta por ciento
} else {
    poner la opacidad del Personaje
    a cien por ciento
};
```

Estructura FOR en Construct 2



La acción en la hoja de eventos dice:

Cada 2 segundos, por medio del loop FOR llamado “repetición” (que empieza en uno y termina en diez) se repetirá diez veces la operación de crear el Personaje en la pantalla, de forma aleatoria.

Capítulo 3. Producción de Videojuegos

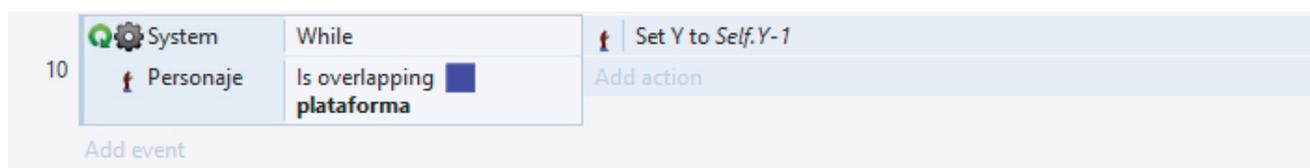
3.7 Estructuras de control aplicadas a Construct 2

Este ejemplo en pseudocódigo significa:

cada dos segundos crear el loop

```
for (empezando en uno, incrementando el valor de uno en uno hasta diez)
{
  Crear al Personaje una vez
};
```

Estructura WHILE en Construct 2



La acción en la hoja de eventos dice:

Mientras que el Personaje esté superponiéndose a la plataforma, colocar al Personaje arriba de ella.

Este ejemplo en pseudocódigo significa:

```
while (el Personaje esté superponiéndose a la plataforma)
{
  colocar al Personaje arriba de la plataforma
};
```

Capítulo 3. Producción de Videojuegos

3.8 Creación de Videojuego: Cazareliquias

3.8 Creación de Videojuego: Cazareliquias

La idea principal del proyecto ha sido presentar un apartado descriptivo sobre el diseñador, los videojuegos y la programación visual, sin embargo es importante mencionar que también se ha tenido la idea de presentar por medio de un ejercicio práctico un producto funcional que contenga todos los requerimientos de un videojuego de principio a fin, donde el lector pueda consultar en forma los documentos pertinentes a las hojas de eventos que contienen la lógica de interacción.

Desarrollo del videojuego

El videojuego creado tiene como nombre “Cazareliquias” y se logró por medio del uso del *Game Engine Construct 2* con programación visual.

La introducción al Engine se logró por medio de la lectura de la documentación pertinente a *Construct 2*, esta lectura permitió el primer acercamiento a la interfaz y el funcionamiento de todas sus partes y módulos, sin embargo una vez que ya se tenía dominada su forma operacional, fueron otros tipos de información los que facilitaron el proceso creativo, como tutoriales en YouTube, UdeMy y libros enfocados a describir las relaciones interactivas que propiciaban el *Gameplay* por medio de ejemplos prácticos así como bibliografía referencial sobre bases de programación.

El desarrollo del juego fue producto de un proceso donde la libertad creativa estaba supeditada a los recursos y referencias interactivas del Game Engine con los que se contaba, de esta forma primero se planearon los niveles en lápiz y papel y en base a las referencias bibliográficas sobre interacción se pudo definir el género, donde por medio de las capacidades de codificación que se tenían en ese momento se generaban prototipos básicos cuyo fin era testear el desenvolvimiento del personaje con el entorno y su interacción contra enemigos y los estados de victoria y derrota, de esta forma cada versión de prototipado realizaba una mejora a su versión anterior. Alrededor de 4 prototipos fueron creados hasta que se encontró la forma de unificar los elementos del juego.

Concepto y descripción

Para la realización del videojuego “Cazareliquias” se han tomado como inspiración los juegos de plataformas 2D de 8 y 16 Bit producidos en las décadas de los 80’s y 90’s pero que contrasta y se diferencia por medio de un estilo visual moderno, logrado con elementos de las estéticas Vaporwave y Outrun, estilos predominantes en 2010-2015, además incorpora como referencia el toque de locura visual que tiene el juego indie PUSS!¹⁶

¹⁶ Zasiadvok, I. (2018). PUSS! [Nintendo Switch, Android, Microsoft Windows, iOS, GNU/Linux, Mac OS,

Capítulo 3. Producción de Videojuegos

3.8 Creación de Videojuego: Cazareliquias

La jugabilidad ha sido pensada para mantenerla lo más sencilla posible (moviéndose horizontalmente, saltando y disparando) buscando un *Gameplay* sin pretensiones, tomando como inspiración el juego *Flappy Bird* y que recuerde también al videojuego de *Megaman* en cuanto a la forma en cómo se avanza a través de los niveles, así como la manera en que el personaje principal tiene que enfrentar a los enemigos por medio de frenéticas ráfagas de disparos, remarcando la facilidad de control y lo responsivo de este.

Se ha buscado que la experiencia de juego sea amigable, por la facilidad de uso del personaje, esto incluye 3 movimientos básicos que fueron pulidos cuidadosamente para reflejar una jugabilidad muy agradable y que incluye los movimientos laterales, saltos y disparos, se buscó que el juego tuviera esos movimientos para que el jugador se concentrara principalmente en mantenerse sobre las plataformas de izquierda a derecha, esquivando obstáculos como picos y huecos que le quitarían vidas, evitando caer al vacío por medio de saltos, retomando el género de *2D Platformer - Side Scroller Shooter* y una pizca de *Lateral Endless Runner*.

El principal activo dentro del juego es la velocidad de respuesta ante el terreno irregular en el eje X que pueden ser plataformas móviles, como plataformas fijas, pero con espacios abiertos una entre otra, por lo que el jugador tendrá que ser muy preciso a la hora del salto; también hay plataformas móviles en el eje Y con movimientos de arriba - abajo donde la coordinación de salto entre las plataformas tiene que ser sincronizada.

El otro punto fuerte en *Gameplay* es la acción que se ejerce por medio de disparos del personaje, se calculó que la velocidad de disparos fuera frenética para que la emoción fuera intensa además existen una gran cantidad de enemigos que pueden ser eliminados fácilmente, teniendo en mente, el placer que genera eliminar a un enemigo con su explosión pertinente, generando ese placer de victoria repitiendo una y otra vez.

La dificultad del juego aumenta un poco en el final ya que en el último nivel del juego al acercarse la recta final hay más espacios entre plataformas y un mayor número de enemigos necesitando una cantidad mayor de balas para derrotar a los enemigos por lo que se tiene que ser muy preciso en los reflejos si se quiere avanzar, esto con el fin de intensificar la emoción de alcanzar el final del juego.

Para el sistema de daños se utilizó la fórmula de que al estar en contacto con el enemigo este desaparece en efecto kamikaze inmovilizando al jugador un momento, restándole una vida, se definió que el jugador perdiera una vida en lugar de regresarlo al inicio del nivel, incrementado la facilidad de juego logrando mantener el ritmo del

Xbox One, PlayStation 4]. teamcoil.

Capítulo 3. Producción de Videojuegos

3.8 Creación de Videojuego: Cazareliquias

nivel que se está jugando obteniendo la atención del jugador por concluir el juego. A nivel visual se buscó que el personaje portara elementos mecánicos representativos, como una máscara antigases dando una sensación de dramatismo y objetividad, ya que, al no conocer su rostro, su personalidad pasa a segundo plano, sólo tiene un objetivo que es obtener las reliquias, además de esta forma el personaje sale de contexto y no se relaciona con algún escenario.

Se buscó que el jugador obtuviera un énfasis realista al controlar al personaje principal ya que el personaje ejerce movimientos muy expresivos, exagerando las caídas y dejando notar el esfuerzo al saltar y correr, volviéndolo como un ente real, para contrastar con los movimientos genéricos de los enemigos que denotan la mecanización de sus actitudes por medio de la repetición en el patrullaje cubriendo semi automáticamente una zona dada.

Tras un gusto por juegos no convencionales se decidió generar escenarios que salieran de lo común utilizando elementos geométricos fuertes por medio de línea, punto de fuga, repetición y matrices, todos estos marcados en posiciones donde existe contraste de forma, manejando una paleta de colores con un toque fresco, cítrico, ácido, juvenil que causa atención por las combinaciones de colores que de ninguna manera están presentes en la realidad, como un mar morado, montañas amarillas, líneas y retículas en punto de fuga, haciendo hincapié en la vivencia de una realidad alterna, distorsionada, fantasiosa, pero suave delicada, colorida y llena de vida; morados, rosas, fucsias, verdes así como los tonos en sus estados puros en RGB como azul, rojo y amarillo son colores que predominan. Fue usada también la estética *Glitch*.

Los niveles del juego están llenos de robots, estatuas antiguas, monstruos, así como elementos griegos clásicos y máquinas industriales, todos renderizados foto realísticamente con V-Ray, para descontextualizar el género visual y percibir un entorno atemporal donde no se distingue ni región ni época, se buscó que los niveles estuvieran llenos de estímulos visuales.

El personaje principal fue creado en el software Fuse de Adobe, software especializado en la creación de figuras humanoides customizadas, por lo que para fines prácticos se ha buscado el moldear un personaje que sea exagerada y desproporcionadamente musculoso con el fin de transmitir fuerza, poder, control, rapidez y movimiento al jugador y que tenga congruencia con el rol que tiene en el juego, un héroe que elimina enemigos con una uzi para obtener las reliquias; también el color de su atuendo en rojo, sus botas en azul y su arma uzi en amarillo han sido escogidos para crear una triada armónica de colores primarios puros con el fin de resaltar su presencia contrastando con los fondos coloridos de los niveles además genera la idea de presentarse como un personaje serio, enérgico y feroz.

Capítulo 3. Producción de Videojuegos

3.8 Creación de Videojuego: Cazareliquias

Los niveles y enemigos del juego son elementos que están descontextualizados unos de otros haciendo alusión a que cada uno representa y tiene características de su propia realidad en cuanto a forma y tiempo específico pero que conviven en el entorno 2D del videojuego para dar la sensación de inmersión en un entorno imaginario donde no existe una relación de causa y efecto, esto también es complementado presentando una gran carga de estímulos visuales que en su conjunto recrean un mundo exagerado, irreal, divertido, gracioso y fantasioso. Es de esta forma que los elementos del juego comparten un estilo visual estilizado con rasgos realistas unificados por medio de una paleta de color formada por degradados, colores fosforescentes y tonos puros y brillantes de morados, verdes, rosas y aqua para atraer la atención del jugador en todo momento, creando una experiencia de juego diferente a través de un ambiente onírico desarrollado por medio de un entorno no convencional, raro y experimental.

La obtención de reliquias es la parte fundamental del videojuego, ya que al final del nivel, si se eliminaron todos enemigos, se evitaron los obstáculos y se obtuvieron 3 llaves se tiene acceso a abrir el cofre que contiene una reliquia, donde se disfruta el momento de verle girar contrastando con los escenarios y los personajes fantasiosos; cada una es digna de presentarse como el premio principal del nivel con un espectacular color dorado que las recubren simulando ser oro, estas piezas que aparecen al final del nivel son reproducciones de escaneado 3d de objetos verdaderos de culturas antiguas, incrementando ese gusto por lo antiguo y lo místico que contrasta con los objetos estilizados del juego.

La música original es basada en caja de ritmos (como música de marcha) teniendo la idea de simplicidad y repetición simulando el trabajo arduo que el personaje tiene que cumplir en su misión; fue realizada a través de loops individuales forzados con diferentes efectos y mezclada a diferentes tiempos en el master final.

Workflow de desarrollo

El flujo de trabajo o *workflow* es la relación de tareas que se siguen para que un producto pueda ser completado en un proceso de desarrollo.

El primer punto a desarrollar en el videojuego “Cazareliquias” fue la creación de la estructura de niveles, se pensó la posición en que estarían las plataformas que sostendrían los elementos del juego, de esta forma inicialmente fue usado Photoshop para bocetar las delimitaciones gráficas, después una vez concretado el nivel se visualizó que tipo de movimiento y acciones tendrían los enemigos y sus zonas de patrullaje, en este punto para volver factible y posible el desarrollo fue que se utilizaron modelos 3d de personajes poco convencionales y modelos de máquinas que son gratis descargados de la página <https://sketchfab.com> por lo que en base a los

Capítulo 3. Producción de Videojuegos

3.8 Creación de Videojuego: Cazareliquias

modelos que se fueron encontrando se sugería cual debería ir en qué nivel, los que se adaptaran y tuvieran una congruencia entre ellos serían los elegidos, estos modelos fueron descargados en OBJ y FBX archivos estándar para edición de 3D, el paso siguiente fue abrir cada enemigo en CINEMA 4D colocarles un material metálico o plástico específico del motor de render VRAY, hacer pruebas de color, iluminación, sombras y generar un render de previsualización, después pasarlos por un *rigging* semiautomático con el constructor que tiene Cinema 4D llamado CHARACTER el cual en base a una plantilla predefinida (se pueden escoger rig humanoides, aves, cuadrúpedos etc.) permite a la geometría tener *rigging* y *skinning* de forma general rápidamente, el paso siguiente fue el empezar a animar cada personaje también en CINEMA 4D logrando un loop de movimiento de aproximadamente un promedio de 30 - 60 frames por cada enemigo a 20 fps, consiguiendo *walk cycles* eficientes, pensando en el menor peso posible del archivo con destino a *Construct 2*.

El paso final se obtuvo renderizando con V-Ray en secuencias de cuadros png con transparencia para ser incrustados en el Game Engine y así tener cada enemigo por separado con su propia animación.

Para la obtención del personaje principal la geometría fue descargada del constructor de personajes FUSE de Adobe el cual permitió customizar un personaje que sería musculoso y con una máscara como accesorio, después se coloreó el mapa de textura que lo acompañaba con tonos vibrantes en Photoshop, para después iniciar el mismo proceso de *rigging* animado y renderizado anteriormente descrito; cabe mencionar que al ser el personaje principal tendría más animaciones que los enemigos, con 9 estados que el jugador podría manipular con el control o el teclado, estos son:

IDLE	SALTO	DAÑO
SALTAR-DISPARANDO	SALTO-CAIDA	CORRER
CORRER-DISPARANDO	DISPARANDO	GANAR

El paso final en la construcción gráfica fue el colocar un fondo por cada nivel y empezar a ubicar los árboles, arbustos, así como la maquinaria como elementos decorativos, todo esto ya en el *Game Engine Construct 2*.

Una vez contemplados los elementos gráficos y los límites de patrullaje que tendría cada enemigo se empezaron a construir las hojas de eventos de *Construct 2*, desde colocar los tamaños del lienzo, que tipo de comportamientos tendrían los elementos gráficos, misiles, las físicas del juego, el *loading* de cada nivel, las interfaces gráficas, el movimiento predefinido del *platformer*, la lógica de ganar perder y explosiones.

Cabe mencionar que, para lograr el resultado interactivo final, se desarrollaron

Capítulo 3. Producción de Videojuegos

3.8 Creación de Videojuego: Cazareliquias

prototipos muy sencillos, primero para aprender las funcionalidades del programa, dominarlo y tras dominarlo pasar a cosas específicas que servirían en el proyecto final, como el movimiento del personaje, sus disparos, la lógica de recorrido de los enemigos entre otros.

Para tener un *Gameplay* interesante sobre el juego se realizó una investigación en libros y tutoriales en video que recreaban *Platformers* clásicos como *Mario Bros* o *Megaman*, el reto a seguir fue poder crear un desarrollo customizado, ya que las documentaciones al estar regadas sin tener un orden, se volvían generales en ocasiones por lo que el videojuego se tenía que complementar con extractos de diferentes fuentes, más lo que la propia lógica fuera diciendo después de comprender conceptos básicos de programación.

Ante este escenario el desarrollo del videojuego fue muy agradable y casi siempre se pudieron dar soluciones a la problemática que la interacción del juego pedía. Construct 2 es un gran motor de juego muy accesible de usar con muchas facilidades. Para las interfaces gráficas se utilizó Illustrator y Photoshop con elementos descargados de internet y otros originales.

La música fue realizada en una aplicación de iPad llamada DM1 The Drum Machine que dio las prestaciones para crear loops de sonidos distorsionados no convencionales siendo una App fácil de usar, después se hizo un armado con composiciones en Ableton Live para dar salida como mp3 a 320 kbps e importarlas en Construct 2 con una calidad de 192 kbps.

Una vez testado el juego se exportó el ejecutable para Linux, Windows y Mac Osx.

Después se subió a la página [itch.io](https://cazareliquias.itch.io/) y se creó un micrositio: <https://cazareliquias.itch.io/> que contiene información del videojuego donde cualquier persona puede descargar el videojuego de forma gratuita escogiendo su plataforma favorita.

Capítulo 3. Producción de Videojuegos

3.9 Documento de Diseño de Juego (GDD)

3.9 Documento de Diseño de Juego (GDD)

Un videojuego es creado gracias a la labor de un equipo multidisciplinario, donde diversas áreas profesionales entran en relación influyéndose unas a otras, sin embargo, la producción de un videojuego puede ser difícil por la cantidad y complejidad de los elementos dentro del juego por lo que para obtener proyectos exitosos la industria de los videojuegos utiliza documentos estandarizados que sirven para ordenar y estructurar el desarrollo a lo largo de las fases.

Estos documentos estandarizados son:

Documento de Concepto/Visión/Propuesta

Documento de Diseño de Juego

Documento de Diseño de Arte

Documento de Diseño Técnico

Cronograma del proyecto

Plan de Prueba de Software

Dependiendo de lo extenso, complejo y específico que vaya a ser el proyecto serán la cantidad de documentos que se generen, sin embargo, el documento que siempre debe de estar presente en un desarrollo es el **Documento de Diseño de Juego**.

El **Documento de Diseño de Juego (Game Design Document - GDD)** es un documento que se obtiene en la fase de preproducción y tiene como fin el describir a todos los miembros del equipo las partes y los requerimientos que serán implementados y desarrollados en el videojuego a lo largo del proyecto.

Este documento también detalla cada elemento presente dentro del juego y sus características por ejemplo el tipo de personajes, niveles, mecánicas de juego, menús, música, interfaces, pantallas de juego, **Gameplay**, historia, diálogos, etc. por lo que el GDD es la columna vertebral del desarrollo sirviendo como guía para el equipo de producción.

Para el **equipo de programación** el GDD funciona como guía para conocer las características y requerimientos técnicos que deben de implementar.

El **director de arte** utiliza este documento como guía para su equipo donde puedan integrar y unificar los elementos artísticos del juego, además de calcular todo el arte que hay que generar en base a los requerimientos.

El **equipo de marketing** debe interpretar los temas usados en el documento para

Capítulo 3. Producción de Videojuegos

3.9 Documento de Diseño de Juego (GDD)

generar las campañas y mensajes en la promoción del juego.

Los **diseñadores de juego** se basan en este documento para crear los niveles, mecánicas y el tipo de *Gameplay* que llevarán los niveles, teniendo control sobre el proceso creativo y la producción del juego.

Los **productores** deberán entender los componentes del juego para dividir el tiempo de producción eficientemente.

Los **diseñadores de audio** deben basarse en este documento para conocer los requerimientos de sonido, voz en off y música que llevará el videojuego.

Las características que debe incluir el GDD son:

Nombre del Juego, Realización, Introducción, Descripción, High Concept, Descripción del juego, Keywords, Género, Cámara, Plataforma, Lenguaje, Licencia, Engine, Audiencia, Referencias de juegos, Tipo de Gráficos, Número de Jugadores, Número de Niveles, Elementos de juego, Interfaz Gráfica de Usuario (GUI), HUD, Menús, Pantallas de juego, Música, FX, Créditos, Controles, Lógica de Juego, Requerimientos técnicos básicos, Mapa de Interacción, Prototipos tempranos

A continuación, se detallan las características que dan forma al videojuego Cazareliquias por medio del Game Design Document.

Capítulo 3. Producción de Videojuegos

4.0 GDD Videojuego Cazareliquias

4.0 GDD Videojuego Cazareliquias

Nombre del Juego:

Cazareliquias (Relic Hunter)

Realización:

Francisco Gabriel Ortuño Silva

Repositorio digital:

<https://cazareliquias.itch.io/relic-hunter>

[https://drive.google.com/drive/](https://drive.google.com/drive/folders/1ZHguL8aONbaHnrEN0Thr30n1Zzn1ZKqz?usp=sharing)

[folders/1ZHguL8aONbaHnrEN0Thr30n1Zzn1ZKqz?usp=sharing](https://drive.google.com/drive/folders/1ZHguL8aONbaHnrEN0Thr30n1Zzn1ZKqz?usp=sharing)

cazareliquias@gmail.com

Referencia:

Ortuño, F. (2021, September 30). Version (Versión 1). Cazareliquias (Relic Hunter). rojomedialab. Retrieved September 30, 2021, from <https://cazareliquias.itch.io/relic-hunter>. Copyright @ 2014 Scirra Ltd. All rights reserved.

Construct 2. Compiled: Release 233 (64-bit). Built at 13:47:42 on Aug 8 2016

Designed: Release 216 (64-bit) 13:46:39 Oct 22 2015

[https://drive.google.com/drive/u/1/](https://drive.google.com/drive/u/1/folders/1ZHguL8aONbaHnrEN0Thr30n1Zzn1ZKqz)

[folders/1ZHguL8aONbaHnrEN0Thr30n1Zzn1ZKqz](https://drive.google.com/drive/u/1/folders/1ZHguL8aONbaHnrEN0Thr30n1Zzn1ZKqz)

Fecha:

México CDMX 2021

Introducción:

Ante la demanda de juegos cada vez más realistas con grandes capacidades visuales se le suma un nicho más pequeño pero que está en auge, el de los juegos casuales; juegos que son fáciles de usar, cortos y de dificultad baja, desarrollados rápidamente, donde se satisface esa demanda por parte del mercado de devorar con rapidez los juegos de moda, para acceder al siguiente lanzamiento.

En base a estos principios se busca el hacer un juego corto, sencillo, divertido y fácil de jugar.

Cada mundo tiene un toque fantástico de colores chillantes con enemigos realizando rutinas de patrullaje, poniendo como primer punto el estar disparando todo el tiempo dejando en claro la rapidez de respuesta motriz del jugador.

Sólo si se consiguen todas las llaves y se obtienen todas las figurillas se podrá terminar el juego.

Descripción:

Únete a la aventura de un Cazareliquias y recupera las 6 piezas doradas perdidas a

Capítulo 3. Producción de Videojuegos

4.0 GDD Videojuego Cazareliquias

través de los diferentes niveles eliminando a los enemigos en este *casual shooter 2D platformer* con elementos de *Vaporwave* y *Outrun*.

High Concept:

La travesía de un Cazareliquias buscando las piezas extraviadas, luchando contra fantásticos y monstruosos enemigos.

Descripción del juego:

Videojuego casual de plataformas **shooter 2D** que consiste en ir recolectando llaves para abrir contenedores y así obtener reliquias, además de destruir enemigos por medio de disparos conforme se avanza el nivel, antes de que ellos lo hagan, se deberán tener todas las reliquias para obtener la victoria.

Keywords:

vaporwave, 2D platformer, shooter, pre rendered sprites, construct 2, outrun, arcade, V-Ray, casual game, glitch, mega man, free game, side scroller shooter, robots, esculturas, reliquias.

Género:

Arcade, Casual Game, 2D platformer, Side Scroller Shooter.

Cámara:

Horizontal (Izquierda - Derecha)

Plataforma:

Windows, Linux y MacOS

Lenguaje:

Español

Licencia:

itch.io Terms of Service

Digital Millennium Copyright Act ("DMCA")

Engine:

Construct 2

Release 216 (64-bit)

www.scirra.com

Audiencia:

Jugadores de 13 a 25 años

Referencias de juegos:

Flappy Bird, Megaman, Contra, PUSS!

Gráficos:

Gráficos estilizados semirealistas 2D prerrenderizados de 3D con colores vistosos, ácidos, frescos, inspirados en los estilos Glitch, Vaporwave y Out run.

Número de Jugadores:

1 Jugador

Capítulo 3. Producción de Videojuegos

4.0 GDD Videojuego Cazareliquias

Número de Niveles:

6 niveles

Elementos de juego:

1 Personaje controlado por el jugador

Elementos en los niveles:

Nivel 1: 4 elementos decorativos (arbustos), 1 fondo, 4 enemigos NPC - Non Player Character (2 humanoides y 2 Objetos) 1 diseño de nivel plataforma (pattern), 1 reliquia.

Nivel 2: 3 elementos decorativos (arbustos), 1 fondo, 3 enemigos NPC - Non Player Character (1 humanoide y dos objetos) 1 diseño de nivel plataforma (pattern), 1 reliquia.

Nivel 3: 3 elementos decorativos (arbustos), 4 elementos ornamentales (columnas), 1 fondo, 3 enemigos NPC - Non Player Character (1 humanoide y 2 Objetos), 1 diseño de nivel plataforma (pattern), 2 plataformas, 1 reliquia.

Nivel 4: 10 elementos decorativos (máquinas), 1 elemento ornamental (reja), 1 fondo, 3 enemigos NPC - Non Player Character (robots) 1 diseño de nivel plataforma (pattern), 3 plataformas, 1 reliquia.

Nivel 5: 1 fondo, 4 enemigos NPC - Non Player Character (robots) 2 diseño de nivel plataforma (pattern), 2 plataformas, 1 reliquia.

Nivel 6: 1 fondo, 19 elementos ornamentales (16 esculturas y 3 arbustos), 4 enemigos - NPC - Non Player Character (máquinas) 1 diseño de nivel plataforma (pattern), 1 reliquia.

Elementos genéricos presentes en todos los niveles

Las 3 llaves, el cofre contenedor de la reliquia del nivel, un helado que da puntos, la barra indicadora de fin del nivel (significa que has caído fuera del nivel), explosión activada cada vez que se toca o se destruye un enemigo, proyectil de disparo del jugador.

Interfaz Gráfica de Usuario (GUI)

HUD:

1 ícono de corazón marcando las vidas del jugador

1 ícono de helado marcando los puntos recolectados del nivel

1 ícono de 3 llaves marcando el número de llaves recolectadas a través del nivel

Menús:

Pantalla Inicio del Juego con BOTÓN de JUGAR

Pantalla Selección de Nivel con BOTÓN DE NIVELES 1-6 (cada botón tiene 2

Capítulo 3. Producción de Videojuegos

4.0 GDD Videojuego Cazareliquias

estados, abierto seleccionable al nivel y con candado bloqueado)

Pantalla Nivel Completado con 2 BOTONES SIGUIENTE NIVEL Y MENÚ PRINCIPAL

Pantalla Game Over Dentro del Juego con 3 BOTONES: VOLVER A INTENTAR, MENÚ PRINCIPAL Y QUITAR JUEGO.

Pantalla Último Nivel Completado con 2 BOTONES: SIGUIENTE Y MENÚ PRINCIPAL

Pantalla Juego Completado con 1 BOTÓN CRÉDITOS

Pantalla Game Over Créditos con 2 BOTONES MENÚ PRINCIPAL Y QUITAR JUEGO

Pantalla Pausa con 2 BOTONES MENÚ PRINCIPAL Y QUITAR JUEGO.

Pantallas de juego:

Pantalla logo de desarrollador

Pantalla logo de Game Engine

Música:

9 Composiciones musicales originales

FX:

Salto, disparo, explosión

Créditos:

Título de Juego:

Cazareliquias (Relic Hunter)

Título de Tesis:

Guía para el Diseño y Desarrollo de videojuegos con Construct 2, Introducción a la lógica de interacción para no programadores.

Director de Tesis:

Carlos Emilio Cardoso Olmedo

Realización:

Francisco Gabriel Ortuño Silva

<https://cazareliquias.itch.io/>

cazareliquias@gmail.com

[https://drive.google.com/drive/](https://drive.google.com/drive/folders/1ZHguL8aONbaHnrEN0Thr30n1Zzn1ZKqz?usp=sharing)

[folders/1ZHguL8aONbaHnrEN0Thr30n1Zzn1ZKqz?usp=sharing](https://drive.google.com/drive/folders/1ZHguL8aONbaHnrEN0Thr30n1Zzn1ZKqz?usp=sharing)

Licencia:

Digital Millennium Copyright Act (DMCA)

Facultad de Artes y Diseño (FAD) - Universidad Nacional Autónoma de México (UNAM)

México CDMX Septiembre 2021

Instrucciones de Instalación:

Capítulo 3. Producción de Videojuegos

4.0 GDD Videojuego Cazareliquias

Windows 32 y 64:

Descarga el archivo comprimido .RAR descomprimir y doble click en el archivo .EXE

MacOS 32 y 64:

Descarga el archivo comprimido .RAR descomprimir y doble click en el archivo APP

Linux:

Descarga el archivo comprimido .RAR descomprimir

Lógica de Juego:

Sistema de ganar y perder

El jugador cuenta con 6 vidas al principio del nivel, si el jugador es tocado por un enemigo se le resta una vida, al restar todas las vidas muere es Game Over y vuelve a aparecer en el principio del nivel.

En los niveles 1, 2, 3 y 4 un disparo del jugador sirve para eliminar al enemigo.

En los niveles 5 y 6 algunos ítems necesitan 5 balas para ser eliminados y otros 1

Se necesitan 3 llaves por nivel para obtener la figurilla, tomarla y pasar al siguiente nivel

El jugador gana y el juego termina si obtiene las 6 figurillas

Cada 40 ítems de helado se suma una vida extra

Requerimientos técnicos básicos:

Windows 7 o más reciente

2 GB RAM

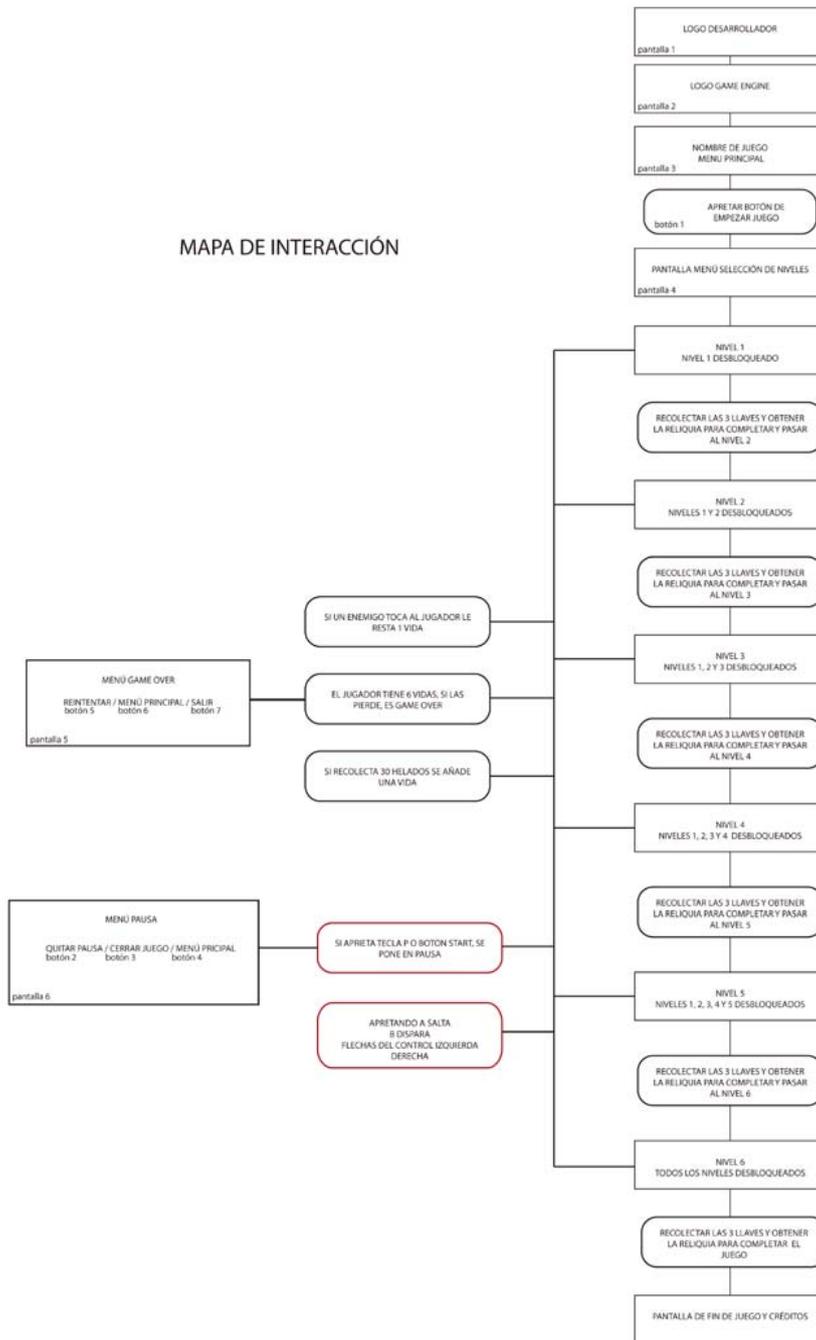
2 GHZ dual - core processor

Tarjeta gráfica NVIDIA o AMD con drivers actualizados

Capítulo 3. Producción de Videojuegos

4.0 GDD Videojuego Cazareliquias

MAPA DE INTERACCIÓN

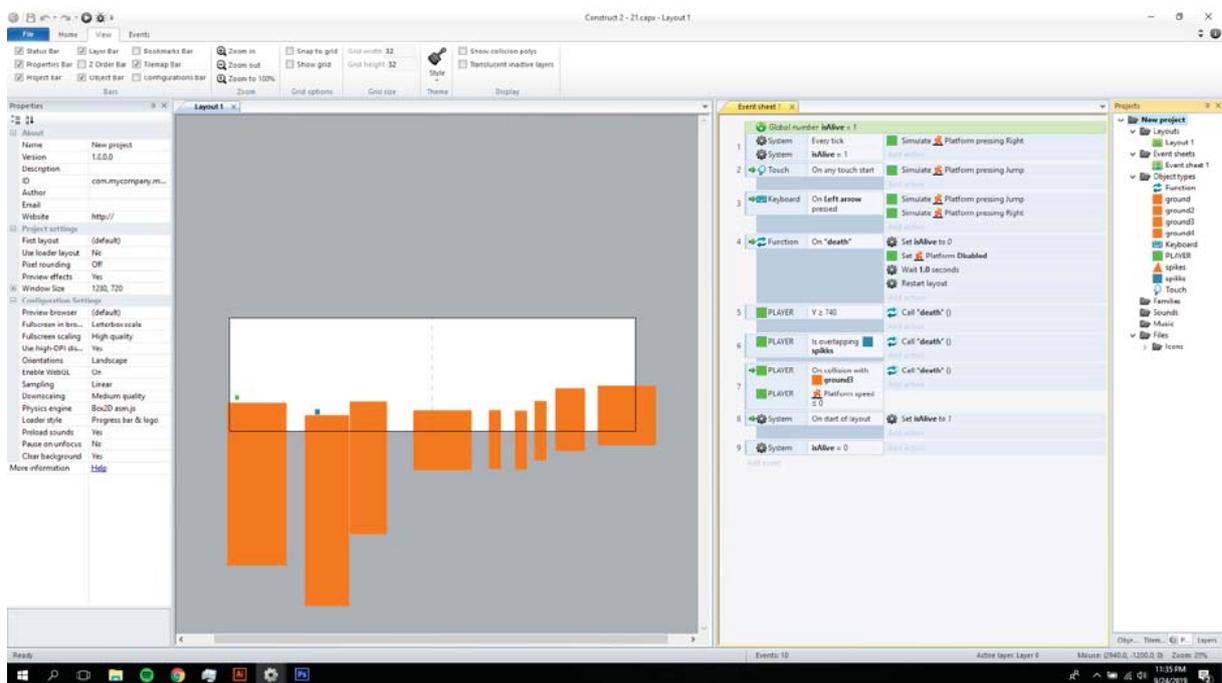


3.153 Mapa de interacción Videojuego Cazareliquias
Ortuño F., 2020, Mapa de Interacción Videojuego Cazareliquias

Capítulo 3. Producción de Videojuegos 4.0 GDD Videojuego Cazareliquias

La creación de prototipos involucró el testeo del *Gameplay* que se ejercía por parte del personaje en el entorno y la lógica que tendrían los enemigos, por lo que al principio no era necesario tener los gráficos definitivos.

Varios prototipos fueron realizados, siendo los 3 últimos los que reafirmaron el tipo de género que se buscaba dando pauta para el desarrollo total.

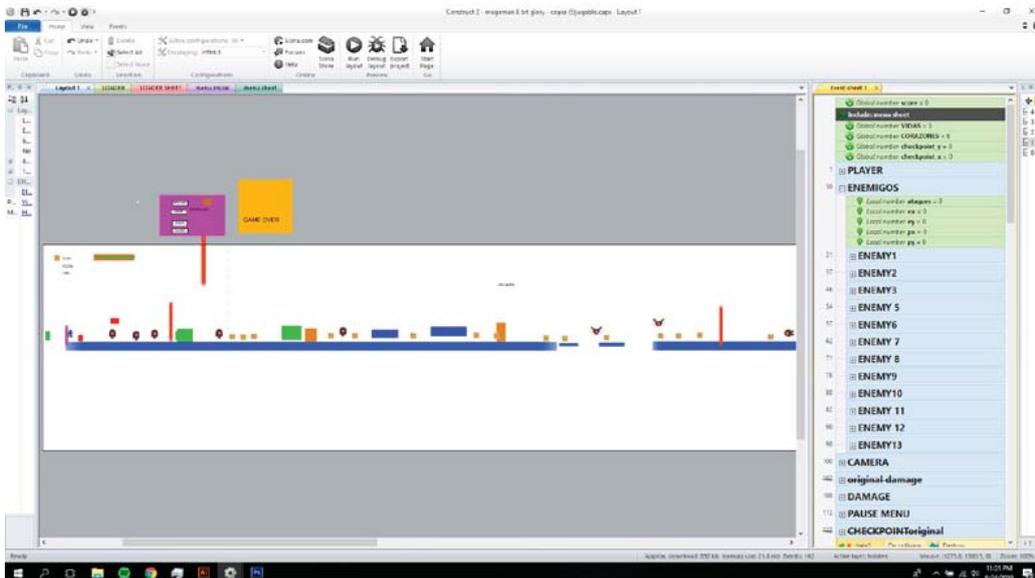


3.154 Construct 2 Prototipos Tempranos Videojuego Cazareliquias

Gullen, A., & Gullen, T. (2011). Construct 2 Prototipos Tempranos Videojuego Cazareliquias (Nº de versión Release 216 (64-bit)). Windows. Wandsworth, SW London: Scirra.

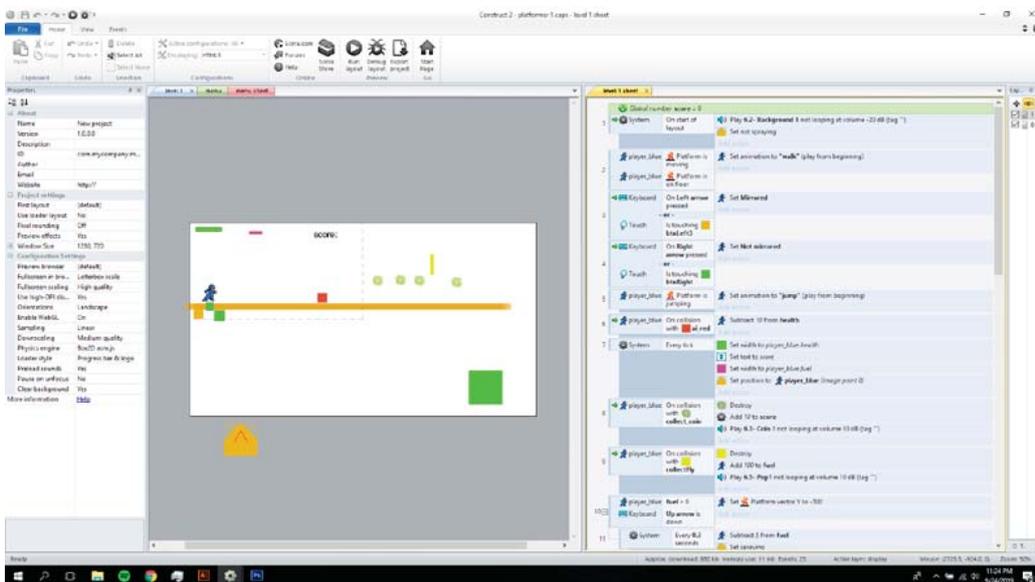
Capítulo 3. Producción de Videojuegos

4.0 GDD Videojuego Cazareliquias



3.155 Construct 2 Prototipos Tempranos Videojuego Cazareliquias

Gullen, A., & Gullen, T. (2011). Construct 2 Prototipos Tempranos Videojuego Cazareliquias (Nº de versión Release 216 (64-bit). Windows. Wandsworth, SW London: Scirra.



3.156 Construct 2 Prototipos Tempranos Videojuego Cazareliquias

Gullen, A., & Gullen, T. (2011). Construct 2 Prototipos Tempranos Videojuego Cazareliquias (Nº de versión Release 216 (64-bit). Windows. Wandsworth, SW London: Scirra.

Capítulo 3. Producción de Videojuegos

4.0 GDD Videojuego Cazareliquias

Para jugar se necesita usar el teclado y mouse para manipular los menús y un Control Xbox 360 o un Generic Controller por puerto USB para manipular al Jugador 1 (aunque también se puede manipular al jugador con el teclado).

Teclado:

“A” Mover a la izquierda al personaje

“D” Mover a la derecha al personaje

“J” Disparar el arma del personaje

“K” El personaje salta

“P” Se pausa el juego

Control USB XBOX 360 Controller o genérico:

“Joystick izquierdo” para mover al personaje izquierda - derecha, derecha- izquierda

“Botón Star” Pausa el juego

“Botón X” Disparo del jugador

“Botón A” Jugador Salta

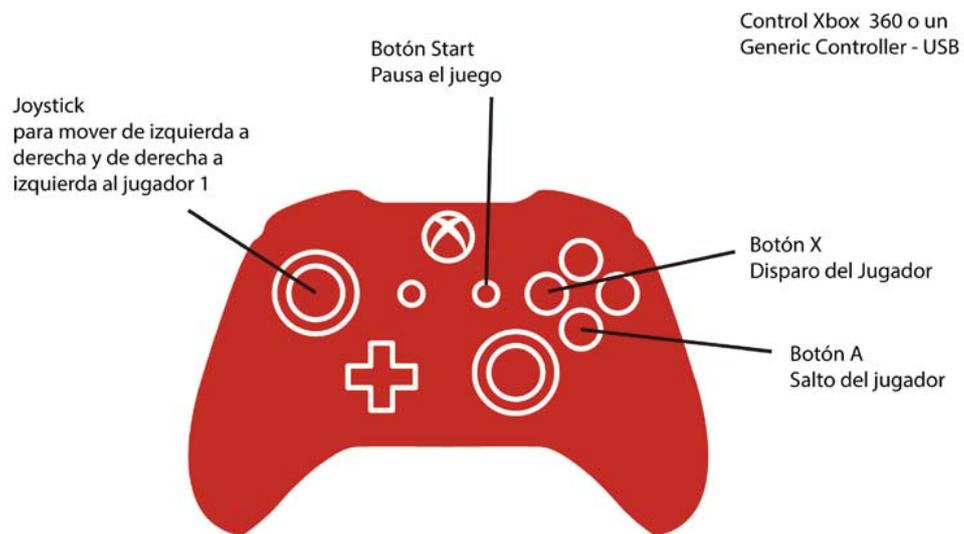


3.157 Controles del juego – Teclado.

Ortuño F., 2020, Controles del juego – Teclado

Capítulo 3. Producción de Videojuegos

4.0 GDD Videojuego Cazareliquias



3.158 Controles del juego – Xbox 360

Ortuño F., 2020, Controles del juego – Xbox 360.

Conclusión

Conclusión

Los videojuegos se refieren a productos dedicados al entretenimiento que generan experiencias interactivas por medio de retos cognitivos en mundos fantásticos virtuales provocando inmersión en el jugador, además los juegos han llegado a un desarrollo tan meticuloso y avanzado en gráficos, historia, jugabilidad, mecánicas y sonido que forman parte como bien cultural, llegando a ser considerados obras de arte.

Para entender el campo de acción que ejerce el Diseñador en la industria de los videojuegos y visualizar posibles nuevos roles y formas de trabajo a futuro, es necesario conocer el contexto tecnológico vigente.

Las principales características que engloban a la octava generación de consolas estando a un paso de concluir su ciclo de vida son las resoluciones 4k con gráficos HDRI, mundos abiertos inmensos, *multiplayer* Online masivo, gráficos y efectos fotorrealistas, juegos móviles, VR, AR y una creciente popularidad de *e-sports*.

A nivel tecnológico dentro de la sociedad sucede el inicio de la Cuarta Revolución Industrial, término acuñado en 2016 para describir la evolución sobre la informatización y digitalización de la producción por medio de varias tecnologías como *Internet of Things (IoT)*, *Realidad Virtual (VR)*, *Realidad aumentada (AR)*, *Cómputo Cuántico*, *Inteligencia artificial (AI)*, *Vehículos Autónomos*, *Nanotecnología*, *Big Data*, *Cloud Computing*, *Ciberseguridad*, *Impresiones en 3d* y *Bioingeniería Genética*, logrando una velocidad en el avance tecnológico sin precedentes, evolucionando a un ritmo exponencial.

El informe *The Future of Jobs Report 2018 del World Economic Forum* describe las tasas de adopción de nuevas tecnologías por industria y por empresas en los próximos años (2018-2020). De esta forma se muestran los tres principales porcentajes de adopción:

85% adoptarán el análisis de Big Data de usuarios y entidades.

75% adoptarán mercados habilitados para aplicaciones (Apps) y web.

75% adoptarán también tecnologías fundamentadas en el Internet de las cosas.

El principal fundamento que todas estas áreas tecnológicas presentan es el uso de recursos por medio del desarrollo de software a través de la programación.

Por lo tanto en esta etapa se prevé que los mercados laborales globales experimenten múltiples transformaciones marcados por la división de trabajos realizados por

Conclusión

humanos y los realizados por máquinas y algoritmos, generando grandes cambios sociales creando e interrumpiendo empleos, con argumentos que fundamentan que la automatización tendrá un impacto negativo en labores antes dedicadas a personas quitando trabajos; así también argumentos positivos como la mejora en la calidad del trabajo y en la productividad de los trabajadores.

Lo cierto es que existirá una apertura en entornos laborales que en un principio no estaban relacionados, ya que el desarrollo tecnológico provocará la mezcla de áreas productivas, esto generará un incremento en la búsqueda por perfiles calificados y una demanda de conocimiento especializado, impulsando el desarrollo de conocimiento.

Ante esta realidad México debe tomar la iniciativa a través del fortalecimiento e inversión en educación y preparación en áreas relacionadas a *STEM (Science, Technology, Engineering and Mathematics (Ciencia, Tecnología, Ingeniería y Matemáticas))* además de promover y dar las facilidades para la innovación y el emprendimiento creando vínculos académicos con el sector empresarial.

En el caso de los videojuegos siendo una industria que va a la par de la innovación en materia de cómputo generación tras generación, los principales avances que se reflejarán en generaciones próximas serán el Cloud Gaming, y la obtención de resoluciones visuales y procesamiento más poderoso exponenciando el mercado móvil por medio del 5G.

También el avance tecnológico dará la posibilidad de crear videojuegos de notable calidad, de forma más económica por medio de mejores herramientas, más fáciles de usar y en ocasiones gratuitas, dando pauta a que el diseñador se incorpore como factor productivo.

En la actualidad los videojuegos se han consolidado como el segmento productivo de la rama del entretenimiento con mayores ventas globales generando ingresos anuales de \$152.1 billones de dólares, superando a las industrias del cine, música y *streaming*, destacando como un sector económico muy rentable con crecimientos exponenciales; en este sentido la industria ha sabido sobreponerse a crisis como el *crash* de 1983 adaptándose a los gustos y necesidades del mercado de jugadores, sirviendo como ejemplo el modelo de juego actual *Free-to-Play*.

El factor de crecimiento se ha dado generación tras generación por medio de la evolución favorable en la calidad de las experiencias interactivas, generadas en la actualidad por medio de gráficos sorprendentes, mecánicas de juego novedosas, *Gameplay* divertido, experiencias *multiplayer* online, jugabilidad móvil e historias inmersivas, permitiendo a las empresas consolidar ventas a través de franquicias de

Conclusión

videojuegos exitosas.

El desarrollo tecnológico también ha potencializado el modelo productivo de juegos o estudios Indie, dando pauta para que estos estudios independientes puedan realizar juegos interesantes con bajo presupuesto.

Los juegos móviles (teléfonos inteligentes y tabletas) siguen siendo el segmento más grande en 2019, produciendo ingresos de \$68.5 mil millones, el 45% del mercado mundial de juegos.

América Latina es el mercado de juegos de más rápido crecimiento en el mundo con una TCAC de + 10.4%, teniendo ganancias de \$5.6 Billones de dólares anuales para 2019, siendo México el principal consumidor de videojuegos de América Latina con una ganancia de \$1.8 Billones de dólares anuales para 2019.

Entendiendo estos números suena muy prometedor el desarrollo de una industria Nacional de juegos que satisfaga contenidos regionales y que a la vez tenga la capacidad productiva sobre proyectos de grandes franquicias en México, creándolos a menor costo sin sacrificar calidad, empezando a ser un factor económico activo dentro de la industria del videojuego internacional, recuperando inversión y obteniendo un buen margen de ganancias.

Algunos estudios Mexicanos destacados que han empezado este camino publicando sus juegos en las principales plataformas son Hyperbeard, Estación PI, Lienzo, Bromio, GameCoder Studios, entre otros.

Dadas las pautas tecnológicas marcadas en la actualidad y las oportunidades de mercado regional para el desarrollo de juegos es que se abre la posibilidad para que más profesionales puedan integrarse de forma oportuna en el ramo, percibiendo una industria que está en búsqueda de nuevos talentos que puedan fortalecer y aportar ideas frescas e innovadoras, con el fin de exponer las ventas en mercados regionales o globales.

En este sentido la investigación cumple con el objetivo de contemplar e integrar al Diseñador en la cadena productiva de un juego, no sólo como creador en las áreas audiovisual y gráfica, sino como generador de prototipos y desarrollos independientes por medio de la obtención de conocimiento sobre el ciclo productivo de un videojuego y en específico sobre los requerimientos técnicos para generar interacción por medio de programación visual.

El diseñador visual basa su trabajo en resolver principalmente problemas de comunicación visual en tiempo y presupuesto basándose en modelos de

Conclusión

comunicación humana, creando mensajes basados en elementos formales de diseño, utilizando texto, hipermedia, imagen fija e imagen en movimiento.

La relación del diseñador con los videojuegos a variado con el tiempo ya que en los primeros desarrollos el programador era quien realizaba también los gráficos y no fue hasta generaciones posteriores a principios de los 80's donde la tecnología permitió que los creadores de imágenes pudieran especializarse aportando sus ideas al desarrollo de un juego, dando vida a gráficos memorables que han ido incrementando su complejidad partiendo del 2D pixel a pixel hasta el 3D fotorrealista.

En la actualidad los roles típicos en los que participa un diseñador en un videojuego son diseño de Interfaz y HUD, branding, identidad y manual corporativo, diseño de sitio web animación y video en 2d o 3d, diseño de empaque, ilustración, concept art, storyboard y diseño editorial.

Para que el diseñador dé un giro a su forma tradicional productiva y empiece a aportar ideas en el área interactiva por medio de la creación de sus propios prototipos es necesario el uso de Game Engines con programación visual ya que son fáciles de entender y ejecutar porque tienen acciones preestablecidas que el creador puede ejercer sobre los elementos del juego, de esta manera logrará interacción de forma accesible, rápida y económica, siendo la interactividad la base fundamental de los videojuegos.

La programación visual provoca que la creación de un videojuego pueda ser realizable para una mayor cantidad de personas interesadas, aún sin tener conocimientos técnicos.

Sin embargo también es cierto que la programación visual puede llegar a quedarse corta y ser básica en la generación de experiencias interactivas en comparación con el desarrollo profesional de videojuegos ya que éste involucra la creación de juegos más complejos con características propias que lo diferencian en el mercado, dado por el trabajo de ingenieros en software especializados en diversas ramas por medio del uso extensivo de la programación, involucrando conocimientos en las áreas de *STEM*, (*refiriéndose a Science, Technology, Engineering y Mathematics*).

De esta forma bajo una perspectiva integradora se plantea que algunos conceptos básicos en programación deben ser comprendidos por el diseñador para mejorar su labor y ampliar las capacidades creativas en la generación de prototipos más complejos, materializando ideas y reduciendo costos eficientando los presupuestos ya que varios roles pueden ser realizados por una persona.

Por lo tanto, para fines de la investigación, se ha logrado generar la documentación

Conclusión

y elaboración de un videojuego con programación visual, buscando también así la comprensión de conceptos básicos de programación tomando como base la formación que un diseñador y comunicador visual en orientación audiovisual e hipermedia tiene, con dominio de software de diseño, pero sin educación formal en *STEM*.

Las ventajas prácticas al aprender programación no se limitan a videojuegos, ya que el diseñador tras entender la lógica y sintaxis de programación podrá extender su formación solventando su trabajo por medio de la utilización de nuevos medios a través de diferentes formas de representar sus ideas, teniendo un abanico de posibilidades para expandirse y especializarse en medios interactivos, por lo que según estimaciones del World Economic Forum en materia de oportunidades laborales, en los próximos años las principales áreas de desarrollo serán Realidad Virtual (VR), Realidad Aumentada (AR), desarrollo de Apps (entre ellas videojuegos) y Web, así también existen otras áreas con potencial en arte, creatividad y diseño donde se podrá vincular la programación como es el uso de Processing, la Programación en 3d, After Effects expressions, Efectos Visuales VFX y el rol de *Technical Artist*. Además, ante el potencial productivo que tienen el diseño y la programación en su uso en conjunto cada vez es más común que en los entornos laborales se pida la integración de conocimientos estéticos y logros técnicos en áreas de programación, por lo que esto obligará que de manera académica la programación y el diseño lleguen a converger de manera muy probable en planes de estudio, modificando los programas académicos actuales en un tiempo reducido.

Así también un término muy importante que debe ser comprendido y adoptado por el Diseñador en su relación con la programación es el del Pensamiento Computacional referido a la capacidad que desarrolla una persona al planear, organizar y ejecutar un plan para la solución de problemas de forma eficiente, utilizando metodologías de ciencias de cómputo y programación, teniendo como ventaja que este modelo resolutivo puede aplicarse efectivamente a diferentes áreas del conocimiento, incluso en la vida diaria.

Los principales conceptos en los que se basa el pensamiento computacional son la abstracción, el pensamiento algorítmico, la automatización, la descomposición, la depuración y la generalización.

El Pensamiento Computacional sentará las bases para que las personas tomen decisiones fundamentadas en la razón, teniendo un pensamiento más ordenado, abstracto y fijo en objetivos, además de desarrollar la aptitud, disposición y eficiencia de forma natural hacia los cambios y actualizaciones tecnológicas, solventando la innovación sustentable dentro de la Cuarta Revolución Industrial.

Conclusión

Es importante para el Diseñador que aspira a desarrollar videojuegos el conocer la historia, definiciones, categorías, fines y objetivos de los videojuegos para tener solvencia en el tema y lograr así la producción de un producto funcional lleno de calidad.

De esta forma un Videojuego se define como un dispositivo electrónico con input y output que se proyecta en una pantalla y que hace uso de juegos electrónicos donde el objetivo es presentar una serie de retos posibilitando al jugador el experimentar situaciones que involucran el desarrollo del pensamiento y la agilidad mental con el objetivo de garantizar la diversión y entretenimiento, presentando simulaciones o realidades ficticias donde cada una de estas tiene su propio orden y lógica.

Cada videojuego es considerado como software y está compuesto por archivos que entran dentro de dos categorías principales: archivos de tipo Data y archivos en forma de Código, además los elementos fundamentales que dan forma al videojuego tales como gráficos, sonidos, interfaces, Gameplay, mecánicas de juego e historia o contexto se han mantenido presentes desde los orígenes de la industria.

De estos elementos dos son imprescindibles para que un videojuego AAA tenga éxito; en primer lugar, el *Gameplay* y en segundo las mecánicas de juego, aunque también pueden llegar a variar en porcentaje de importancia dependiendo del fin y la industria del producto interactivo. Para fines prácticos del Diseñador es necesario que pueda estudiar más a fondo las partes y las áreas de las que está compuesto un videojuego adquiriendo conocimientos, grados académicos y especializaciones en la rama, con el fin de tener un panorama más completo que le permita generar un producto que sea divertido, usable y que entre dentro de los estándares de calidad de la industria.

Desde la primera patente sobre videojuegos en 1947 echa por Thomas Goldsmith y Estle Ray Mann, pasando por los arcades, las primeras consolas caseras, el boom de los PC, el crash de 1983, el surgimiento de los juegos en 8, 16, 32 y 64 Bits dando la transición al 3D y capacidades multimedia, hasta la Octava generación de consolas en la actualidad, los videojuegos han abarcado una gran gama de plataformas y géneros mostrándose como una industria que está a la vanguardia en tecnología y experiencias interactivas.

La revisión histórica de los videojuegos es lo que da la pauta para que los nuevos creadores de juegos puedan comparar y reflexionar con vistas a la producción tomando en cuenta los casos de éxito reflejados en *Gameplay*, así como el conocimiento de los géneros originales que hoy en día se encuentran mezclados en subgéneros.

Los principales fines en los videojuegos son el entretenimiento, la interacción social,

Conclusión

educacional, *Serious Games*, salud, conciencia social, arte y marketing y *Advertising*. Además, en la creación de un videojuego lo primero que se contempla es definir el género tal como acción, aventura, *puzzle*, RPG, simulaciones, deportes, carreras, *Shooters*, *Platformers*, peleas, estrategia y multiplayer online por mencionar los relevantes.

Otro punto que se tiene que definir teniendo los fines y el género es la plataforma en la cual el juego será usado pudiendo ser para consola, PC, Handheld, smartphone o multiplataforma a través del Cloud Gaming; así también el modo de juego ya sea de un jugador, dos jugadores, multiplayer, LAN u Online Multiplayer, definirá el tipo de experiencia, la dificultad, modos de juego e interacción social.

De esta forma el creador de videojuegos podrá desarrollar su idea bajo las características mencionadas tomando en cuenta como primer principio la experiencia de usuario que el videojuego pueda proporcionar al jugador, generando un juego divertido y fácil de usar que aproveche los recursos de la plataforma.

En este sentido una plataforma muy viable para el desarrollo de juegos por parte de creadores novatos como Diseñadores es el smartphone ya que permite el uso de juegos que no necesitan tantos requerimientos técnicos de desarrollo, implementados en una computadora promedio, con tiempos de desarrollo y presupuestos bajos, aprovechando la multitud de *Game Engines* a buen precio que ofrecen exportar para smartphones, además como punto principal se tiene que la plataforma da la posibilidad de obtener monetización por medio de anuncios dentro del videojuego en forma de banners o rich media ads a través de compañías como *AdMob*, *Unity Ads*, *WebEye* y *mopub*.

Dentro del desarrollo de juegos es importante definir el público usuario, para esto se usa la clasificación de videojuegos por contenidos de acuerdo a las edades generada por el *ESRB (Entertainment Software Rating Board)*.

Un factor que ha posibilitado el desarrollo masivo de videojuegos en la actualidad es el uso de *Game Engines* (Motores de Juego) siendo programas que sirven para crear videojuegos o aplicaciones interactivas que contienen módulos preprogramados en su sistema -como los módulos de Física, Audio, Inteligencia Artificial, Renderizado Gráfico, Animación, Efectos especiales entre otros-, evitando al desarrollador programar cada módulo individualmente, logrando eficientar la producción obteniendo beneficios económicos, prácticos, y funcionales; además dos Engines son los principales del mercado: *Unreal Engine* y *Unity*. Los Game Engines han logrado la proliferación de producciones independientes y su uso se extiende en varias industrias como la arquitectura, ingeniería, cine, publicidad, sector automotriz e industria del transporte, por mencionar las principales.

Conclusión

Aunado al desarrollo de los *Game Engines* otro factor importante que ha propiciado la producción de juegos es la evolución y desarrollo de software para la creación de elementos o ítems artísticos (*assets*) como 3ds Max, Maya, Cinema 4D o Substance Designer.

Para escoger un *Game Engine* adecuado se debe tomar en cuenta el precio, las licencias de uso, qué tan complejo es de aprender, si el lenguaje de programación usado es conocido y el nivel de dificultad, su vigencia y soporte, la compatibilidad con software de creación de *assets*, la cantidad de documentación existente, el precio de la capacitación; además se debe de especificar el tipo de género y el fin del juego para constatar la viabilidad en el desarrollo especificando los alcances gráficos e interactivos del producto.

La función principal de un *Game Engine* se da por la interacción de elementos en un mundo virtual por parte del jugador, el programador es el encargado de generar esta función y lo hace creando instrucciones por medio de código de programación usando lenguajes de programación que contienen códigos de computadora, así la computadora interpreta las instrucciones y las traduce en acciones dentro del *Engine*.

La programación basada en texto y la programación visual son las dos formas más usadas para la programación de videojuegos y se diferencian en la facilidad de uso y la complejidad de los resultados que se piensen obtener.

Siendo el estándar en la industria, con la programación basada en texto (C#, C++, C) se pueden lograr resultados más avanzados teniendo conocimientos sobre matemáticas, física, e ingeniería de software; en cambio la programación visual surge como alternativa para personas sin conocimientos técnicos ya que se usan instrucciones previamente codificadas en un entorno visual, sencillo y amigable.

Al usar programación visual, el usuario comenzará por razonar de forma lógica para la resolución de un problema, en lugar de enfocarse en aprender la sintaxis de un lenguaje, además, podrá conocer, practicar, concretar y adoptar conceptos sobre pensamiento computacional, iniciándose en el área interactiva, mejorando en el área lógica matemática.

Los *Game Engines* más fáciles de usar que usan programación visual son Construct 3 y Buildbox.

Los pasos a seguir para un creador que inicia dependerán del tipo de videojuego y de la funcionalidad principal que se quiera llegar a producir, así como del presupuesto, la inversión en conocimiento y habilidades adquiridas enfocadas al desarrollo.

Conclusión

Ya sea que se use programación visual o programación basada en texto, la resolución de problemas interactivos es el punto principal que interviene en la creación de un juego, ya que para que éste pueda funcionar se necesita una lógica que le dé vida, por lo tanto, se deben de conocer puntualmente las estructuras lógicas antes de empezar a bocetar la idea. Para este fin se deben de conocer las estructuras de control fundamentales usadas en la mayoría de los lenguajes de programación siendo así la Secuencial (Sequential), de Selección (Selection) y de Repetición (Repetition). Las estructuras de control permiten que se realicen instrucciones y omitan otras, variando el orden de la secuencialidad en la ejecución de un programa de acuerdo a una condición, rompiendo la linealidad de la estructura.

Una vez dominada la programación visual el siguiente paso lógico es la generación de juegos más elaborados, logrados a través de la programación profesional de videojuegos en diferentes áreas y por medio de roles específicos como programador de *Engine*, programador de Físicas, programador de *Engine* Gráfico, programador de Inteligencia Artificial, programador de Red, programador de Interfaz y programador de Audio a través del uso de trigonometría, álgebra vectorial y lineal, cálculo diferencial, física, arquitectura e ingeniería de software, probabilidad y estadística.

El ser programador profesional implica tener ciertas habilidades mentales y capacidades cognitivas para destacar en la industria, además del gusto por videojuegos, planificación del trabajo por fases, flexibilidad a cambios en los entornos y tareas de trabajo, facilidad en la relación de conceptos y disciplina de aprendizaje constante. Para cubrir este grado profesional existen varias alternativas académicas que van desde universidades extranjeras y nacionales, pasando por masters y especializaciones hasta llegar a cursos y tutoriales online.

Para realizar un videojuego usualmente se divide la producción en fases donde en cada una de ellas se generan documentos especializados que sirven como guía para la creación de los elementos del juego, así como para sentar los procedimientos por los que el diseñador de videojuego y su equipo puedan conocer sus responsabilidades en los ciclos de trabajo, generando un producto de forma eficiente y ordenada. Las fases de desarrollo de un videojuego son: concepto, preproducción, prototipo, producción, testing y postproducción.

En el concepto se da la concepción inicial de la idea principal del juego.

En preproducción se desarrolla la propuesta a detalle y empieza la planeación del proyecto generando los documentos de *Plan de Producción*, *Biblia de Arte*, *Documento de Diseño de Juego (Game Design Document - GDD)* y el *Documento Técnico de Diseño (Technical Design Document - TDD)* que sirven como guía para la producción del juego.

Conclusión

El detalle en la realización de estos documentos debe de ser minucioso destinando tiempo suficiente a su elaboración por medio de formatos estandarizados, siendo la columna vertebral de la producción.

En la fase de Prototipado se realiza un producto interactivo básico donde se puedan testear los conceptos principales de jugabilidad, mecánicas de juego, así como el uso de assets, creando versiones mejoradas de cada desarrollo, comprobando si es funcional y divertido.

En la Producción todo debe de estar claro y cada quien conoce que se debe de hacer, en qué tiempo y de qué manera y de esta forma se empieza la realización del juego. La fase de testing involucra pruebas al software para buscar errores que deban ser corregidos, realizando mejoras y adecuaciones por medio de versiones; esta etapa está dividida a su vez en la fase Alpha, la fase Beta y la fase Gold siendo la fase Gold donde el juego está terminado y listo para manufactura.

En la fase de Postproducción se lanzan parches y extensiones al juego ya lanzado al mercado, también en esta fase entra el desarrollo de marketing.

La producción de un videojuego se da por el trabajo en conjunto de varios profesionales con roles especializados como productores, directores creativos, diseñadores de juego, diseñadores de niveles, diseñadores de *Gameplay*, directores de arte, animadores, artistas 3D, *riggers*, *technical artist*, *technical director*, programador de red, programador de físicas, programador de interfaces, programador de I.A., testers, escritores e involucra el conocimiento por parte de estos sobre los pipelines de trabajo de la producción con el fin de crear workflows efectivos en la creación de assets para el juego.

Es común encontrar principalmente en desarrollos reducidos o indies que varios roles en ocasiones son ejecutados por una misma persona para eficientar el presupuesto al máximo, además de que el mismo desarrollo tecnológico ha provocado una convergencia entre roles artísticos y técnicos generando una necesidad por conocimientos en Game Art y dominio de la programación.

Para generar los *assets* y el arte del juego que entran dentro de la categoría definida como *Game Art* se deben tener cubiertos los requerimientos de las fases conceptuales y de preproducción, siendo necesario el uso de software especializado en las áreas de diseño 2D como Photoshop, e Illustrator, en animación 2D con Moho y FlipBook, en animación y modelado 3D con 3ds Max, Zbrush, Maya o MOI, en texturizado 3D con Substance Designer y Painter y en edición de video y VFX con Nuke, After Effects y Premiere. Poco a poco el medio productivo ha visto crecer el software Open Source, que al ser gratuito rivaliza con las licencias propietarias, proponiendo una

Conclusión

alternativa real para la creación.

Programar significa dar instrucciones a una computadora para que realice cierta tarea, por medio de un lenguaje de programación en código que contiene una sintaxis específica que la computadora pueda entender.

El uso de pseudocódigo y diagramas de flujo son herramientas que sirven para estructurar la planeación y realización de un programa utilizando componentes básicos de programación como variables, constantes, operadores aritméticos, operadores de asignación, operadores lógicos, operadores relacionales, arrays, funciones y las estructuras de control que funcionan permitiendo modificar el flujo de ejecución de las instrucciones de un programa, siendo la estructura secuencial el orden básico de ejecución en los lenguajes de programación; seguido por las estructuras selectivas como **if**, **if/else**, **if/else/if**, **switch**; y las estructuras repetitivas formadas por **while**, **do-while**, **for**.

En este sentido los recursos en línea son una alternativa para iniciarse en la programación facilitando la comprensión del tema en un entorno diferente por medio de videotutoriales, *MOOC (Massive Online Open Course)* o sitios interactivos.

Con el fin de que el diseñador pueda entender conceptos básicos de programación y al mismo tiempo comprenda el ciclo productivo de un videojuego, se ha contemplado el desarrollo de un videojuego a través de *Construct 2, Game Engine* que permite el desarrollo de videojuegos 2D por medio de programación visual, donde el usuario puede generar interacción aún sin tener conocimientos sobre conceptos avanzados de programación, por medio de eventos, condiciones y acciones sobre los objetos del juego, logrando experimentar y jugar creativamente sin preocuparse por la creación de código, sin embargo aunque es posible crear la lógica del videojuego usando acciones preprogramadas, lo recomendable es que el creador del juego pueda conocer las estructuras de programación básicas como *while*, *for*, *if-else*, que le permitirá construir un juego más avanzado; de este modo por medio de la práctica el diseñador podrá enfrentarse y resolver problemas reales utilizando las estructuras de control que cualquier lenguaje de programación utiliza para posteriores desarrollos con diferente software.

En la sección estructuras de control aplicadas a *Construct 2* se ha generado una analogía funcional entre las instrucciones dadas por medio de programación tradicional y sus equivalentes en bloques visuales usados en las hojas de eventos de *Construct 2*, con el fin de comprender la lógica interactiva que se da por la relación de bloques para posteriormente relacionar temáticas avanzadas dentro de la materia, a través de estudios académicos en pensamiento computacional.

El videojuego creado para la investigación tiene como nombre “Cazareliquias” donde

Conclusión

la libertad creativa para su realización estaba supeditada a los recursos y referencias interactivas obtenidas sobre programación básica, el manual de uso, e-books y videotutoriales sobre *Construct 2*. Una vez realizada la investigación se definió el género y se procedió a la creación de prototipos para desarrollar el modo de juego entre el personaje y los enemigos, así como su interacción en el entorno; la mejora de los prototipos se dio a través de un ciclo constante logrando cuatro versiones previas hasta llegar a la quinta versión que serviría como base del desarrollo generando alrededor de diez versiones nuevas cada una con pequeños cambios y mejoras que culminarían con el producto jugable final.

Para la creación del videojuego se creó el Documento de Diseño de Juego (*Game Design Document - GDD*) que detalla los requerimientos técnicos, así como los elementos y características presentes en el juego tales como el tipo de personajes, niveles, mecánicas de juego, menús, música, interfaces, HUD, Gameplay, historia, diálogos, etc. funcionando como la columna vertebral del desarrollo.

El uso de Game Engines con programación visual facilita la creación de videojuegos a diseñadores y personas sin conocimientos técnicos, posibilitando concretar ideas propias con presupuestos bajos, sin embargo la producción de videojuegos como industria global marcada por un mercado de nativos digitales ansiosos por experiencias nuevas, requiere de productos que cumplan con procesos de calidad estandarizados bajo lo último en tecnología, esto significa que la producción de videojuegos profesionales involucra el uso de ingeniería de software y programación, por lo que la producción del videojuego “Cazareliquias” así como la relación comparativa propuesta en la investigación entre los eventos de programación visual y las estructuras básicas de programación, ayudarán al lector a identificar los principios funcionales de esta área de conocimiento con el fin de propiciar investigación y adopción académica a futuro, dentro de un contexto laboral propiciado por la Cuarta Revolución Industrial donde la programación es cada vez más requerida en áreas con relación directa o indirecta al diseño.

Anexo

Diseño de Interfaz de usuario - UI



Pantalla inicio del juego

Botón Jugar
archivo: Play



Pantalla selección de nivel

Botón Niveles
Nombre de archivo: Button_
Lithos Pro



Pantalla nivel completado
archivo: POPUP
Frame: 1

Botón Siguiete Nivel
archivo: nextLevel

Botón Siguiete Nivel
archivo: menuPrincipal

Diseño de Interfaz de usuario - UI



Pantalla Game Over dentro del juego
Frame: 0

Botón Volver a intentar
archivo: RESTART

Botón Menú principal
archivo: menuPrincipal

Botón Quitar Juego
archivo: BUTTONquit1



Pantalla Último Nivel Completado

Botón Siguiete
archivo:

Botón Menú principal
archivo: menuPrincipal



Pantalla Juego Completado

Botón Créditos
archivo:

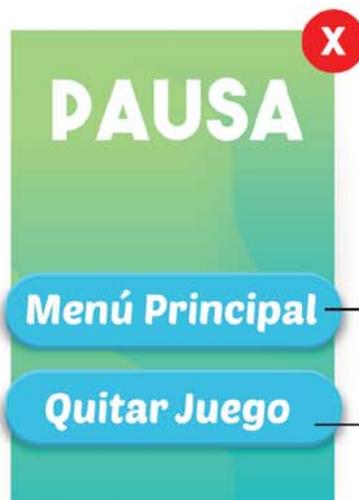
Diseño de Interfaz de usuario - UI



Pantalla Créditos

Botón Menú principal
archivo: menuPrincipal

Botón Quitar Juego
archivo: BUTTONquit1



Pantalla Pausa

Botón Menú principal
archivo: menuPrincipal

Botón Quitar Juego
archivo: BUTTONquit1

Splash Screen



Pantalla logo Desarrollador



Pantalla logo Game Engine

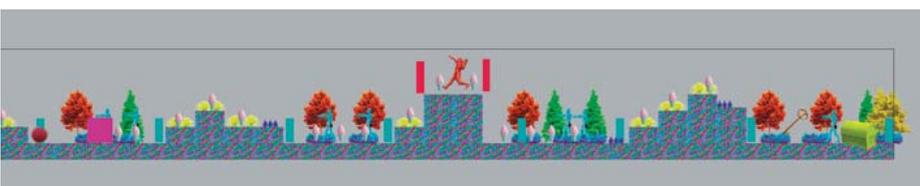
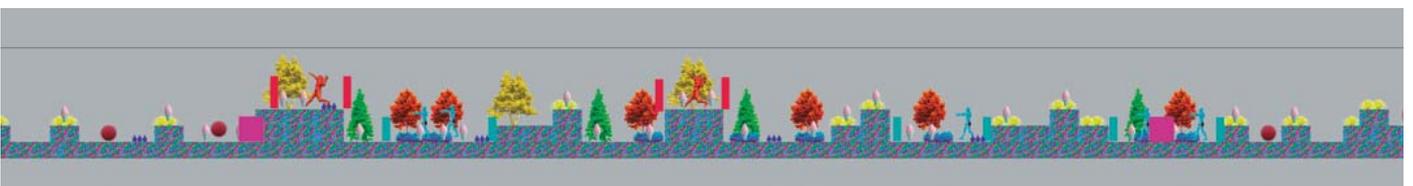
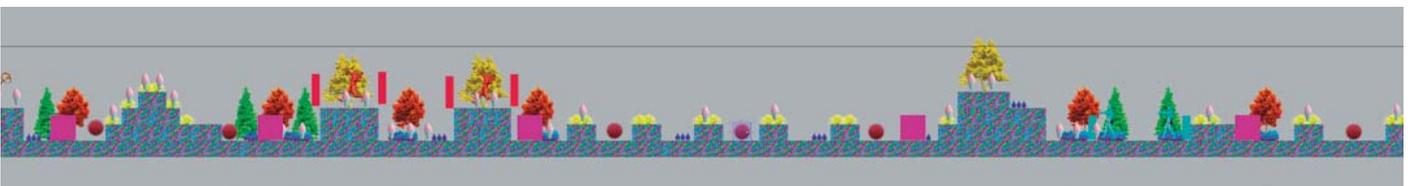
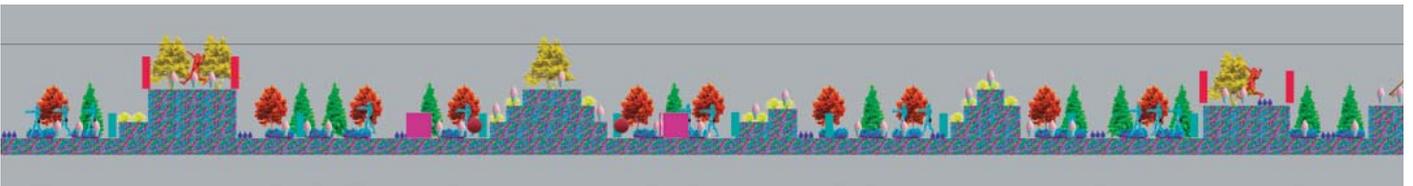
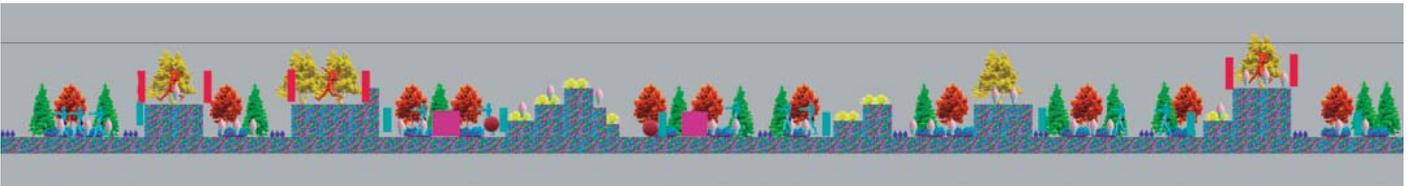
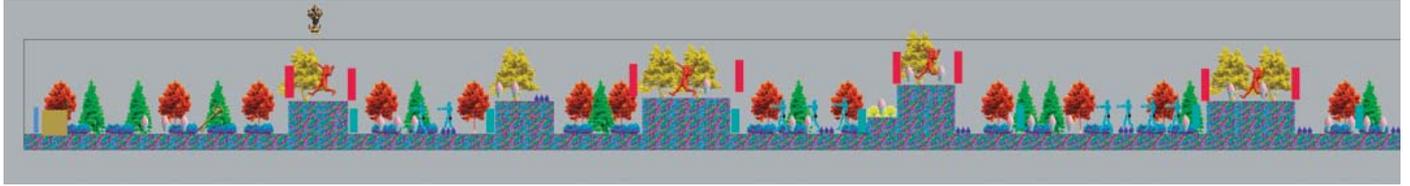
Nivel 1

Pantalla de nivel de juego



Nivel 1

Hoja de nivel de juego



Nivel 1

Elementos de juego. Ornamentos del nivel



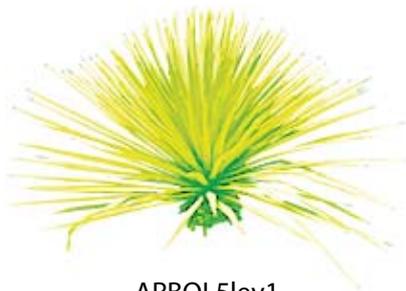
ARBOL2lev1



ARBOLESlev1



ARBOL3lev1



ARBOL5lev1



ARBOL4lev1

Nivel 1

Elementos de juego. Enemigos



Nombre de archivo: FIGURILLAlev1



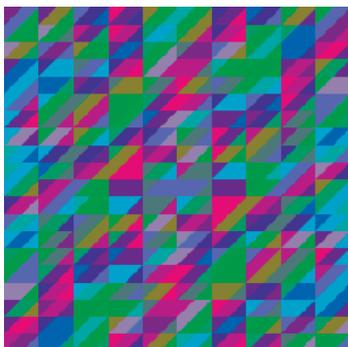
Nombre de archivo: enem2



Nombre de archivo: SpikesBallLev1



Nombre de archivo: PICOSLEV1



Nombre de archivo: platformLEVEL1



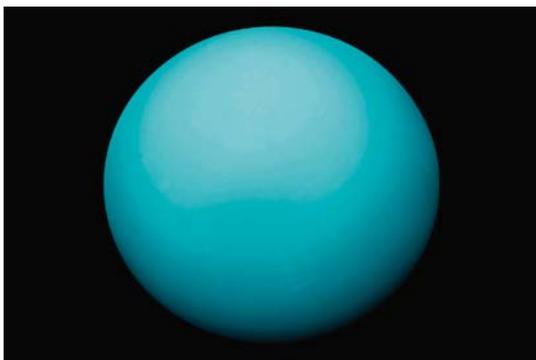
Nombre de archivo: enemy1

Nivel 1

Elementos de juego. Fondo



Elementos por separado



Fotografía de Urano



Render 3d simulación mar



Nubes realistas



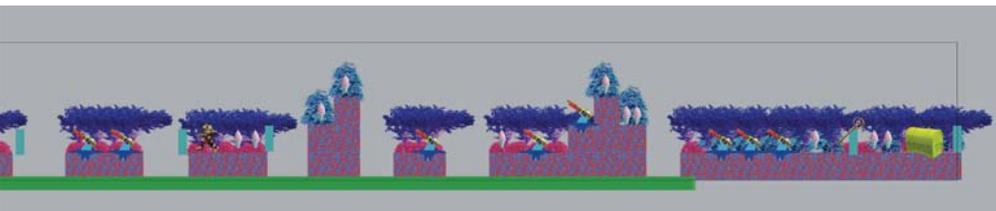
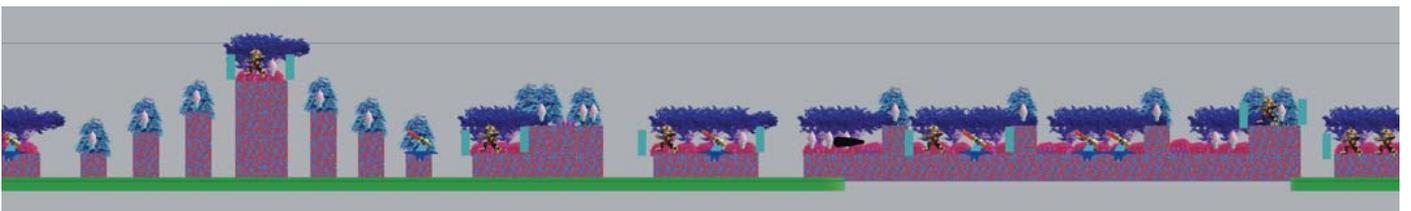
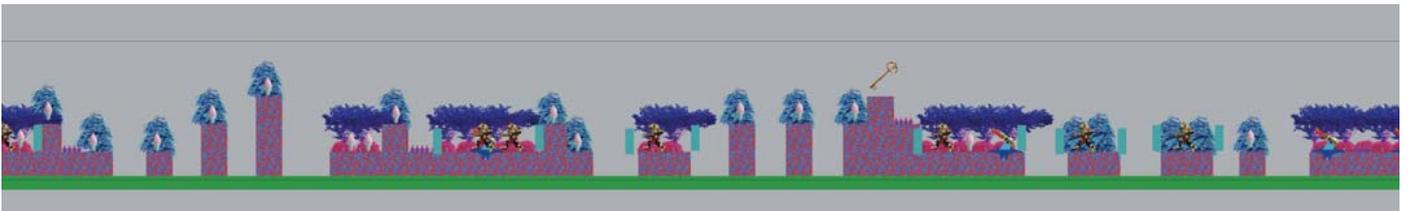
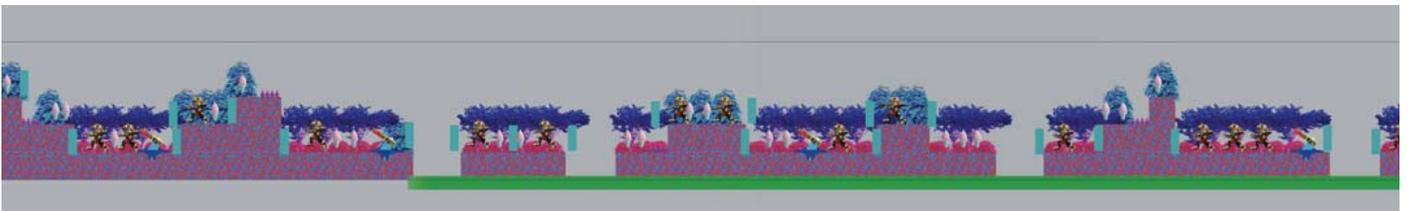
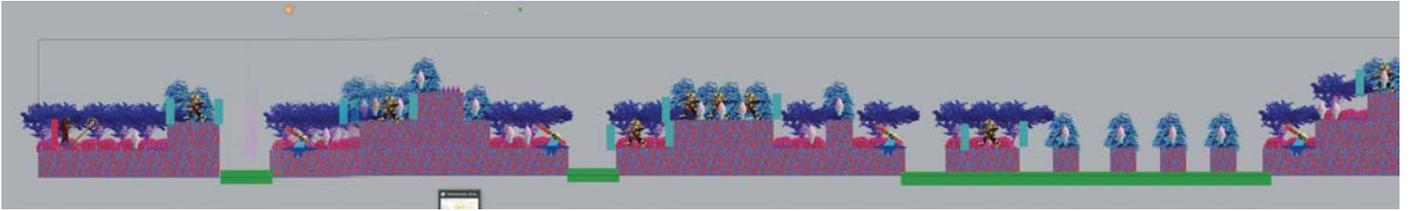
Nivel 2

Pantalla de nivel de juego



Nivel 2

Hoja de nivel de juego



Nivel 2

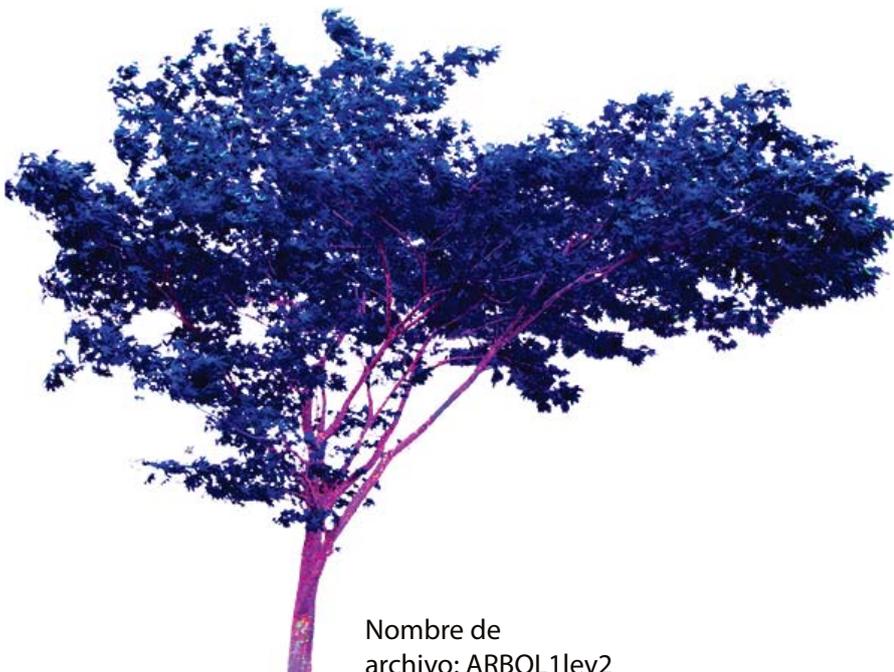
Elementos de juego. Ornamentos del nivel



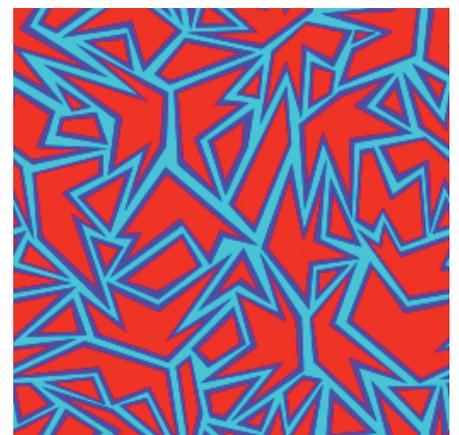
Nombre de
archivo: ARBOLES3lev2



Nombre de
archivo: arbol2LEV2



Nombre de
archivo: ARBOL1lev2



Nombre de
archivo: LEVEL2

Nivel 2

Elementos de juego. Enemigos



Nombre de
archivo: TROFEOLEV2



Nombre de
archivo: LEV2Torreta



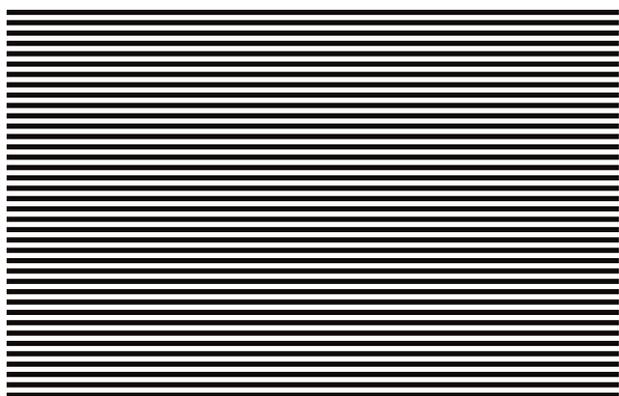
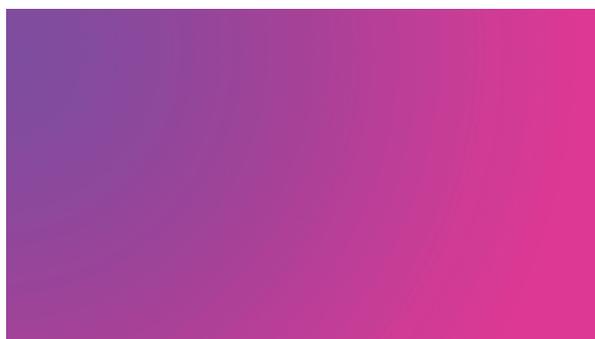
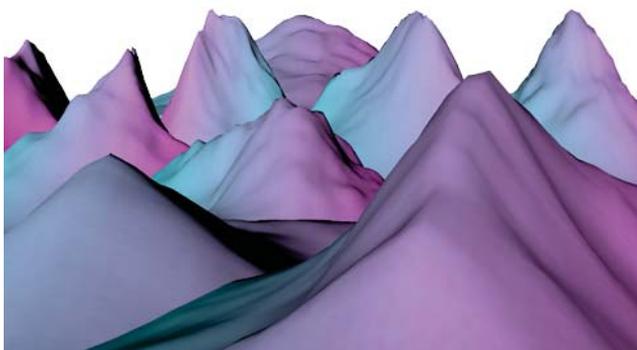
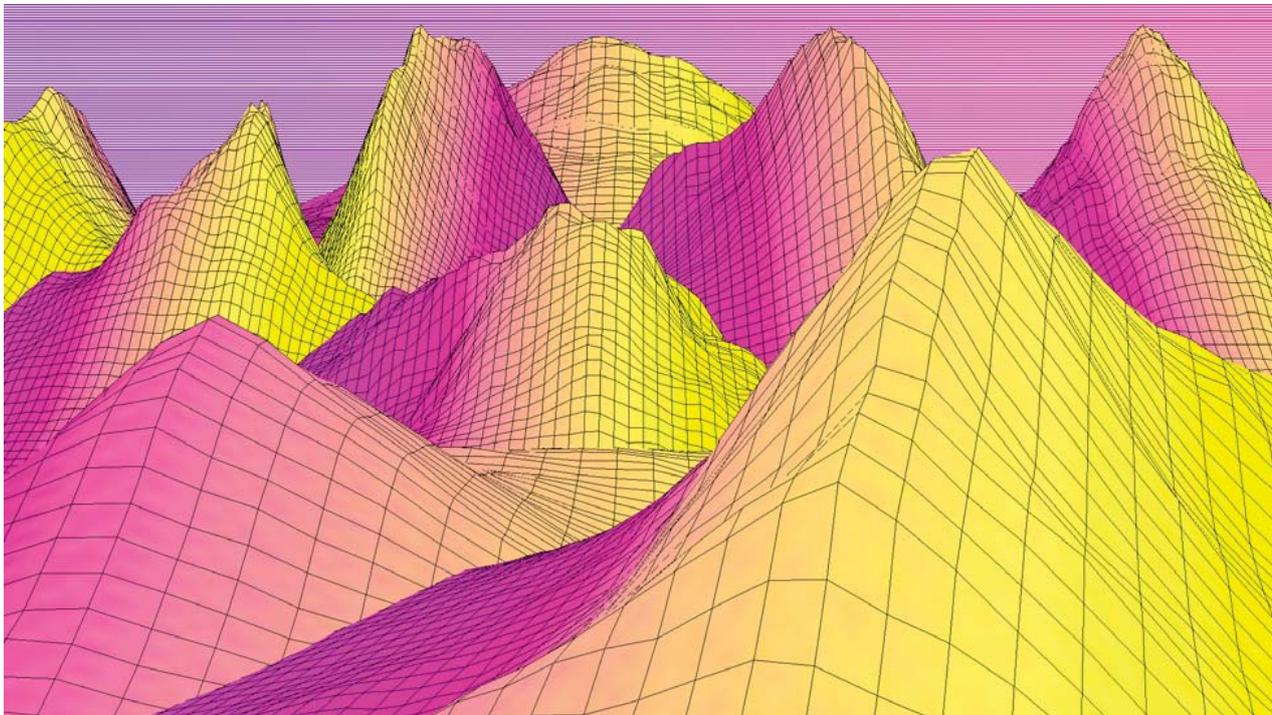
Nombre de
archivo: enemy1Level2



Nombre de
archivo: picoslev2

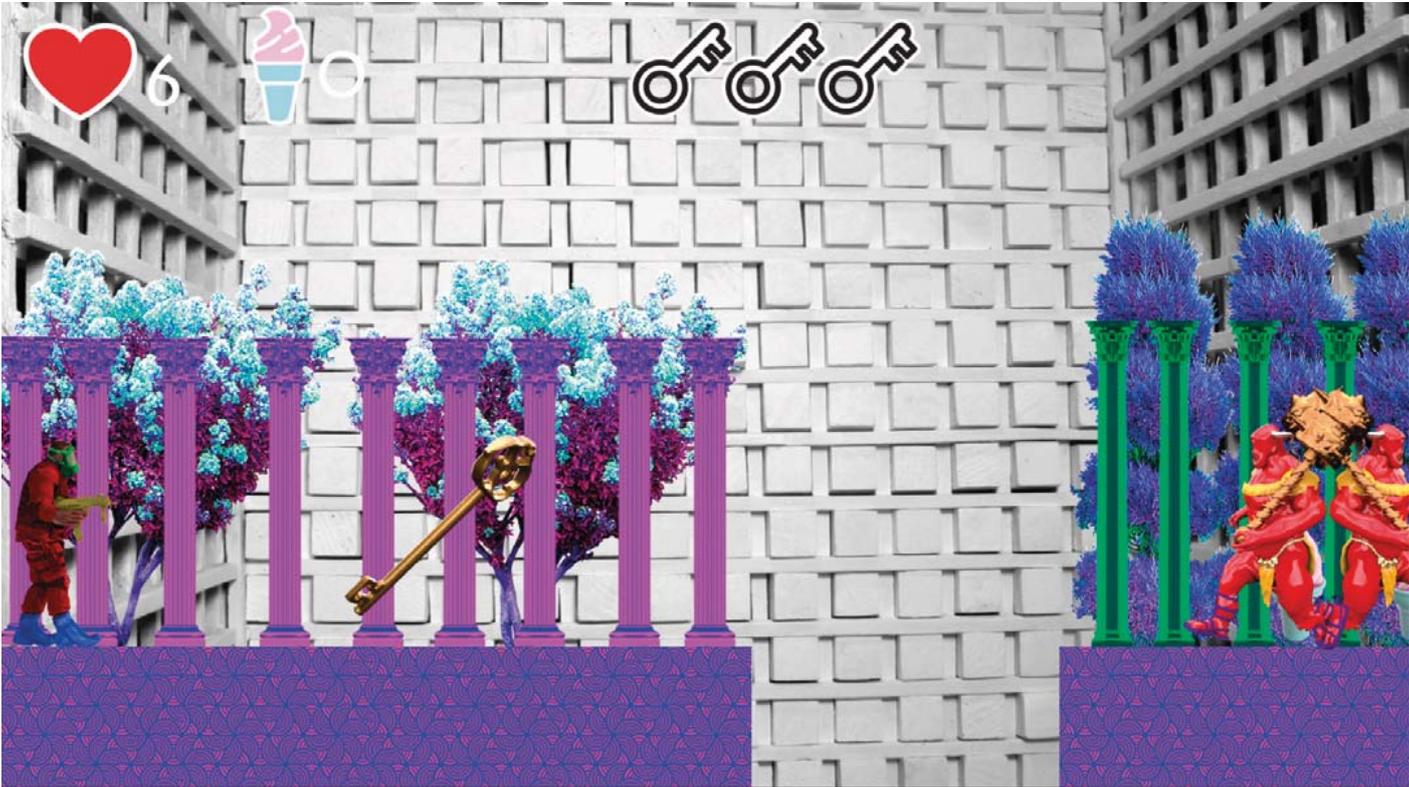
Nivel 2

Elementos de juego. Fondo



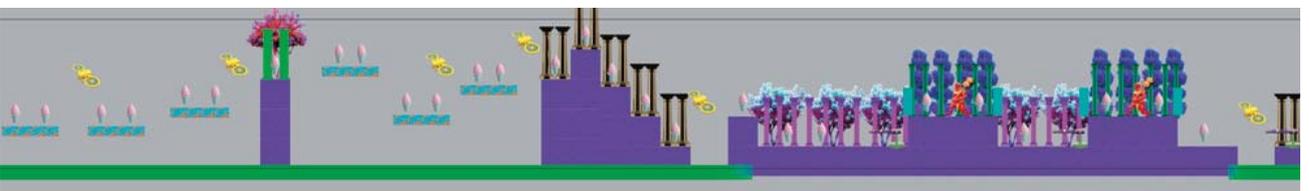
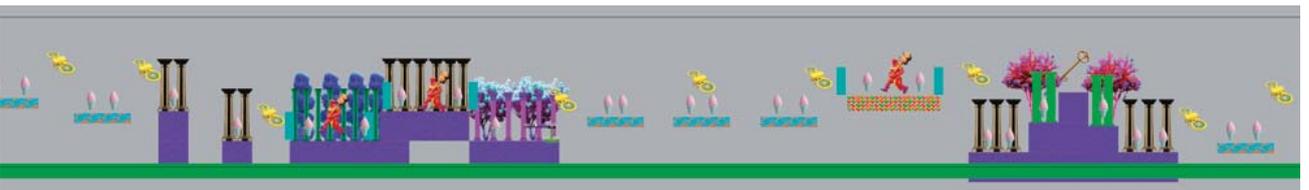
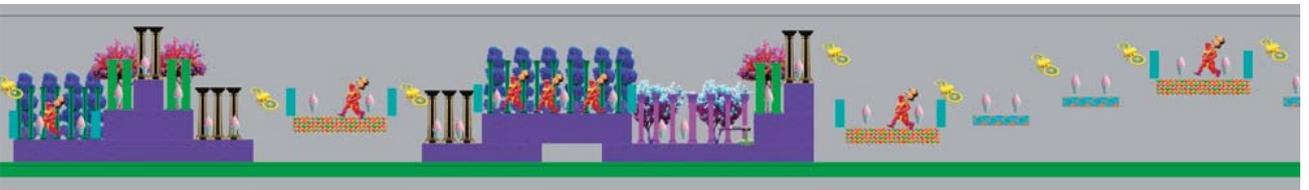
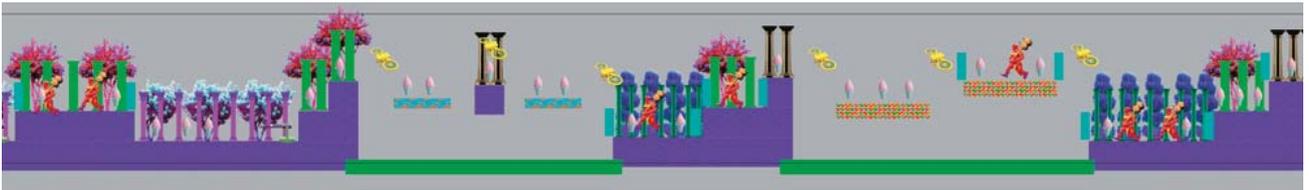
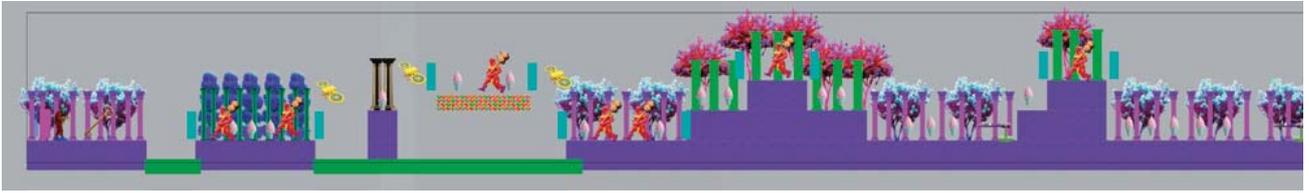
Nivel 3

Pantalla de nivel de juego



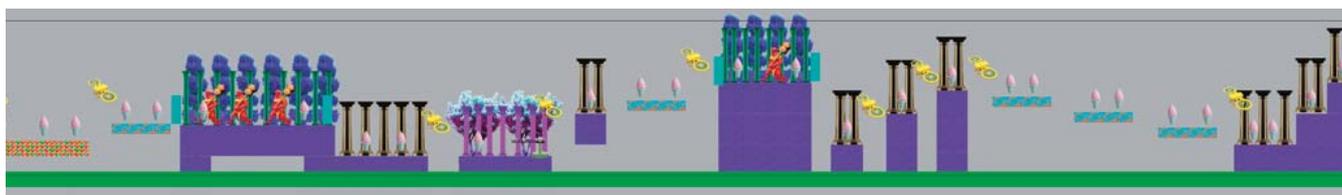
Nivel 3

Hoja de nivel de juego pag. v1



Nivel 3

Hoja de nivel de juego pag. 2



Nivel 3

Elementos de juego. Ornamentos del nivel



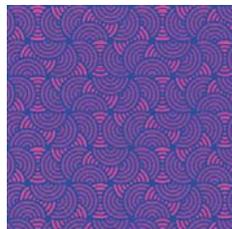
Nombre de archivo: ARBOL1lev3



Nombre de archivo: arbol2lev3



Nombre de archivo: arbol3LEV3



Nombre de archivo: level3Back



Nombre de archivo: TROFEO3



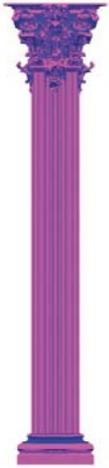
Nombre de archivo: platform



Nombre de archivo: barra

Nivel 3

Elementos de juego. Enemigos



Nombre de archivo: FONDO1lev3



Nombre de archivo: fondo2lev3



Nombre de archivo: FONDO3lev3



Nombre de archivo: FONDO4lev3



Nombre de archivo: Enemy1Lev3



Nombre de archivo: DRONlev3



Nombre de archivo: metraLev3CARAV

Nombre de archivo: METRAlev3Cuerpo

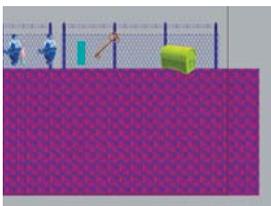
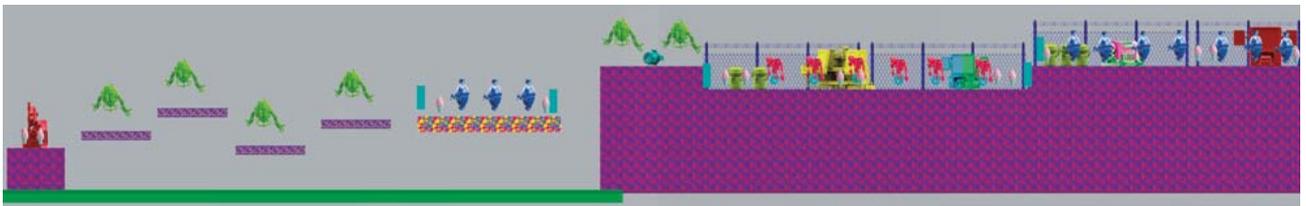
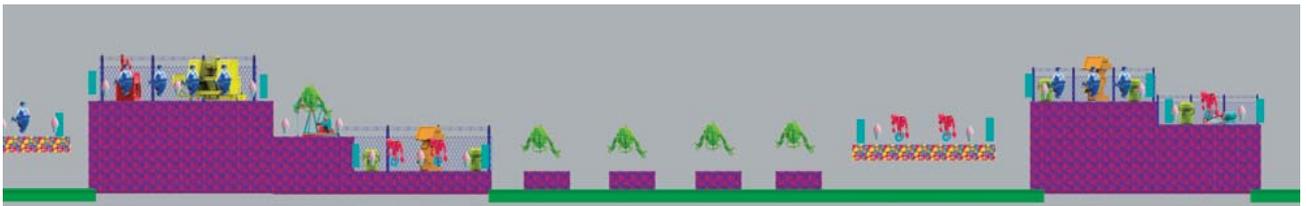
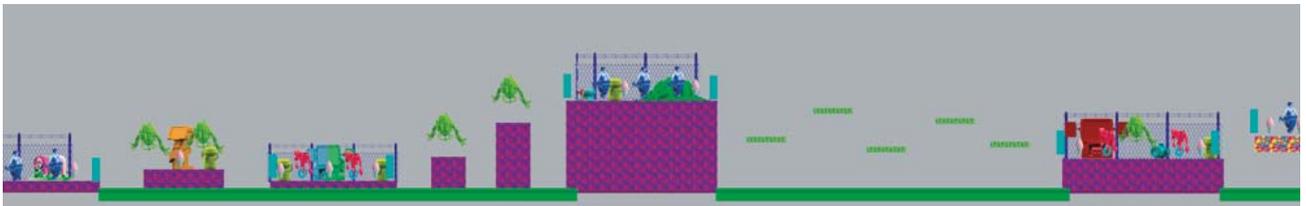
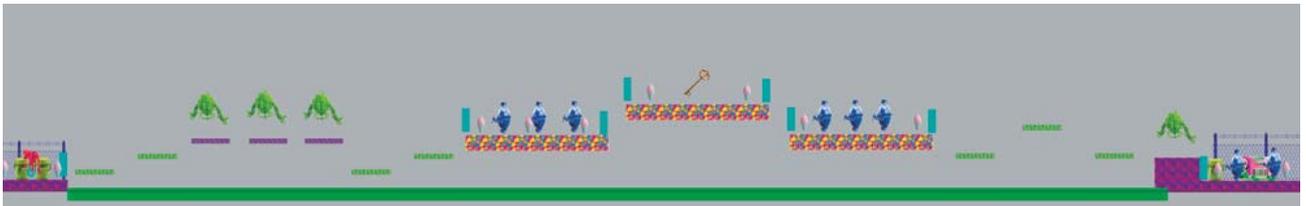
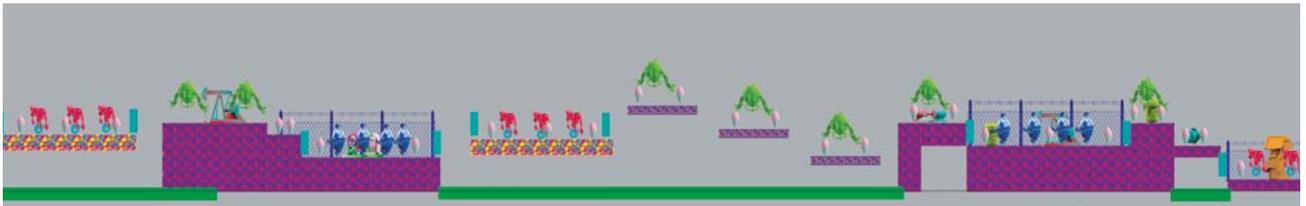
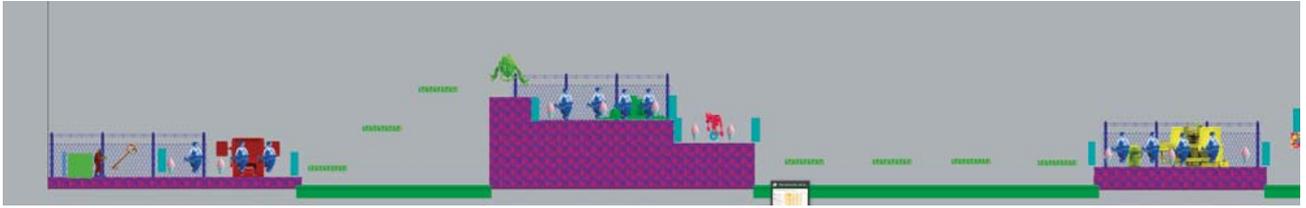
Nivel 4

Pantalla de nivel de juego



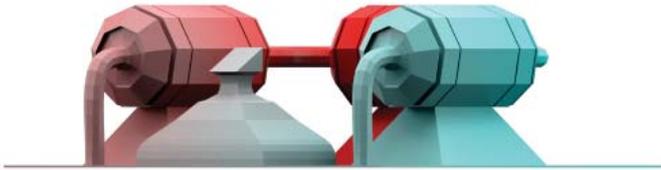
Nivel 4

Hoja de nivel de juego



Nivel 4

Elementos de juego. Ornamentos del nivel pag. 1



Nombre de
archivo: fondo9LEV4



Nombre de
archivo: fondo4LEV4



Nombre de
archivo: FONDO5lev4



Nombre de
archivo: fondo7LEV4



Nombre de
archivo: fondo2LEV4



Nombre de
archivo: FONDO6lev4

Nivel 4

Elementos de juego. Ornamentos del nivel pag. 2



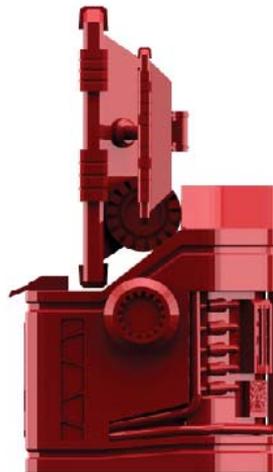
Nombre de
archivo: FONDO10lev4



Nombre de
archivo: 8fondolevel4



Nombre de
archivo: FONDO11lev4



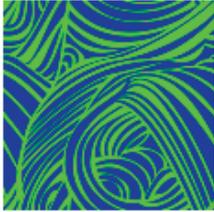
Nombre de
archivo: FONDO3lev4



Nombre de
archivo: TROFEO4

Nivel 4

Elementos de juego. Ornamentos del nivel pag. 3



Nombre de
archivo: lev4BACK



Nombre de
archivo: base



Nombre de
archivo: Sprite10



Nombre de
archivo: Sprite9

Nivel 4

Elementos de juego. Enemigos



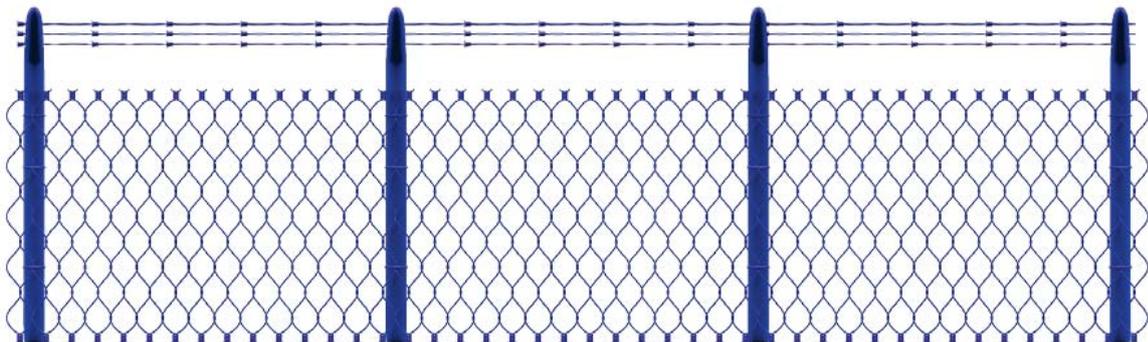
Nombre de
archivo: Enemy3Lev4



Nombre de
archivo: Enemy1Lev4



Nombre de
archivo: enemy2Lev4

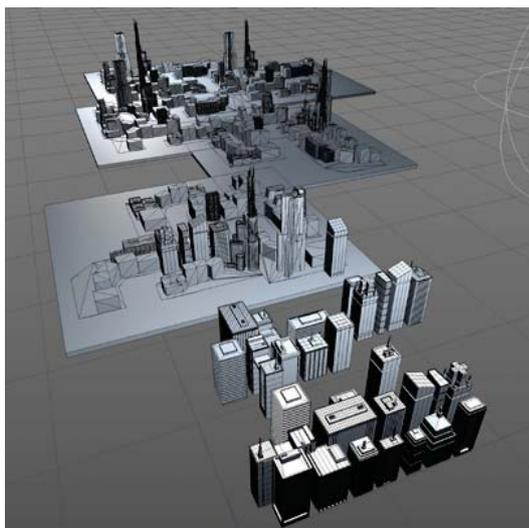


Nombre de
archivo: FONDO1lev4

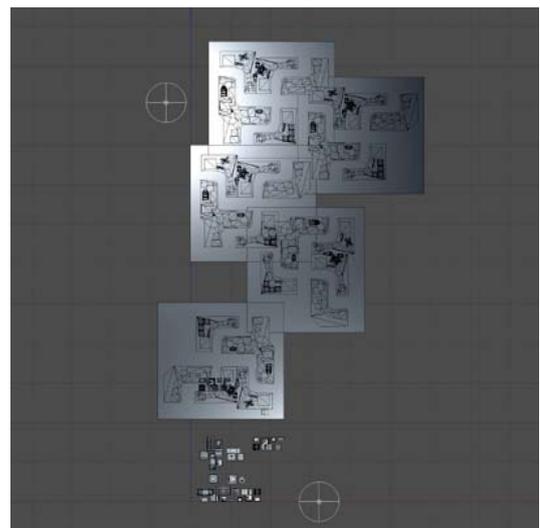
Nivel 4

Elementos de juego. Fondo

Nombre de
archivo: BACKlev4



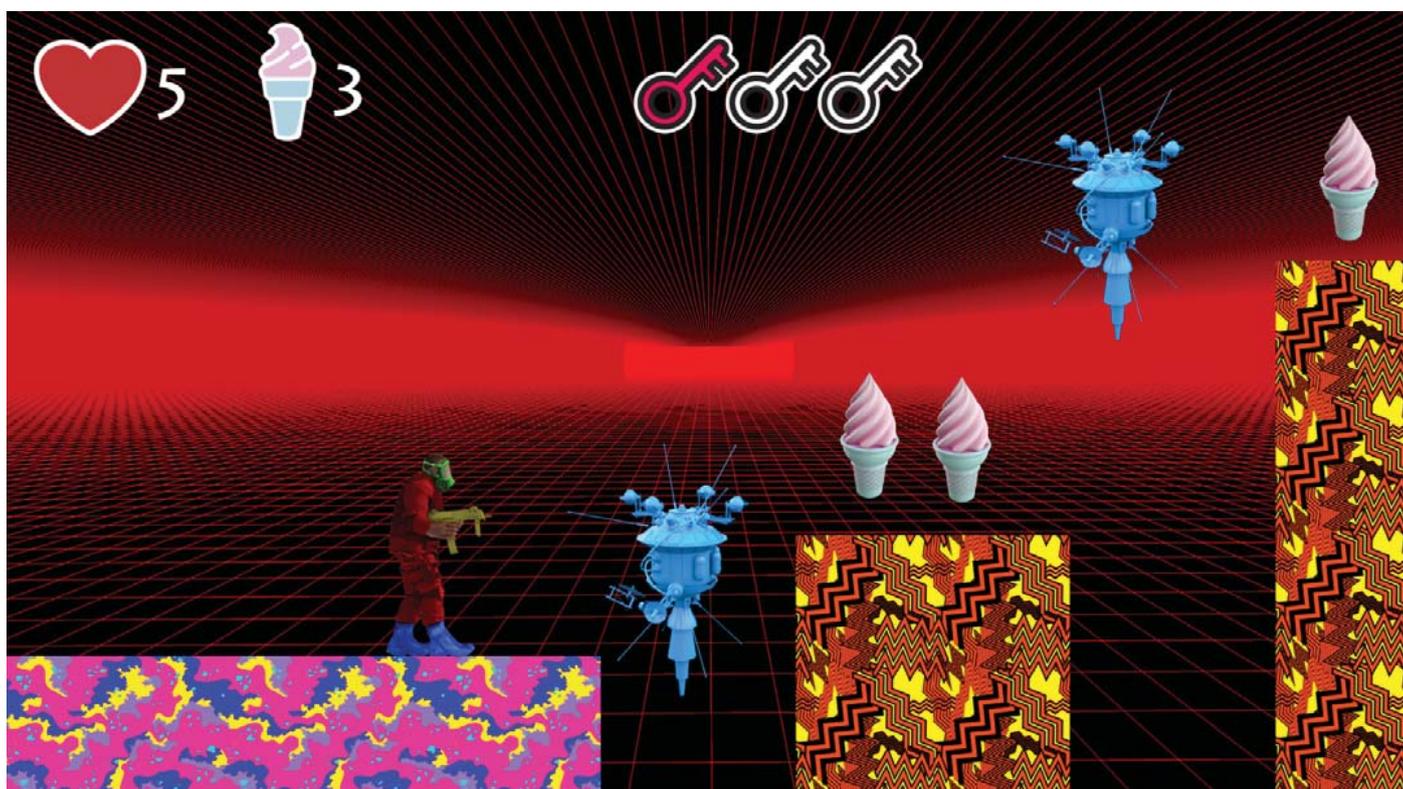
Vista en perspectiva del
fondo



Vista Top del fondo

Nivel 5

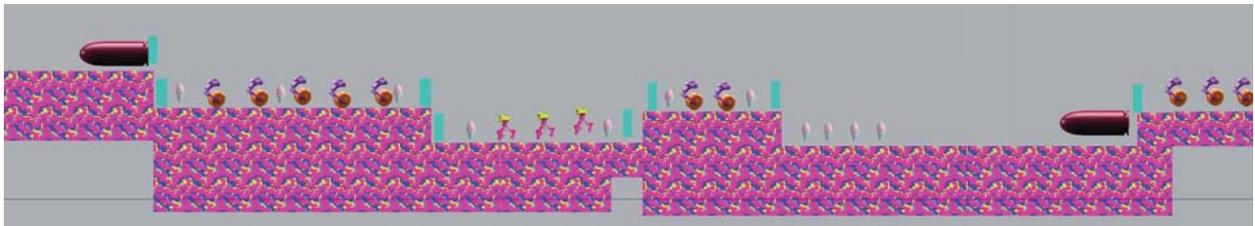
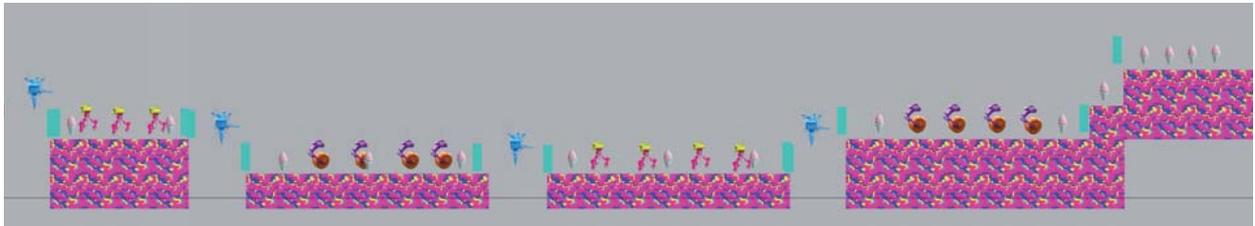
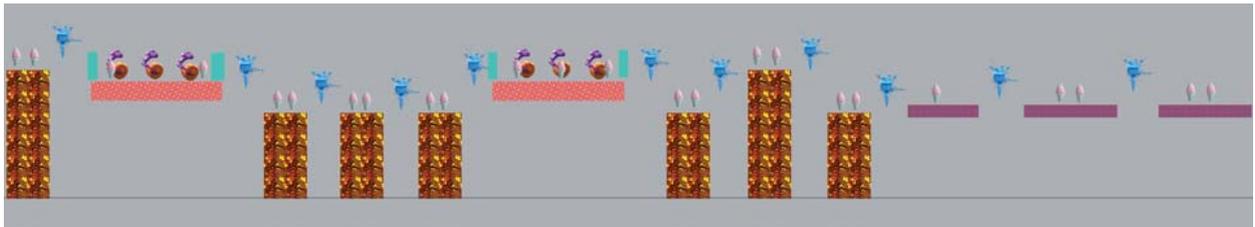
Pantalla de nivel de juego



Nombre de
archivo: Enemy3Lev4

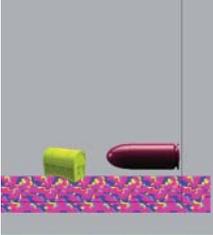
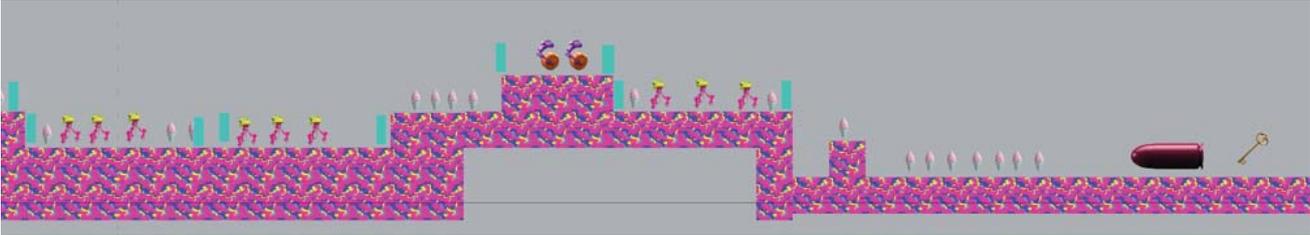
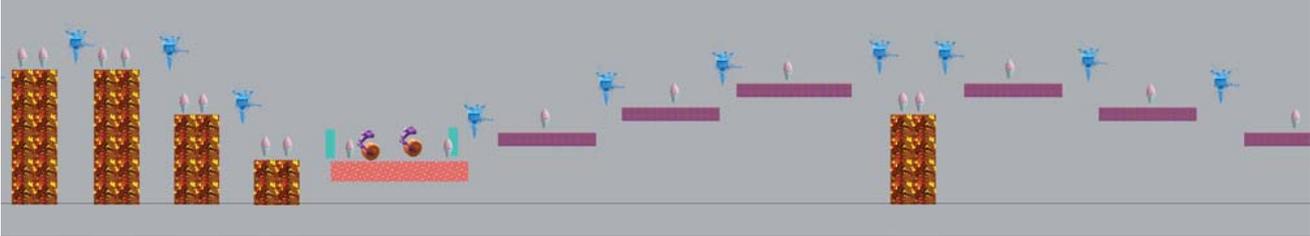
Nivel 5

Hoja de nivel de juego pag. 1



Nivel 5

Hoja de nivel de juego pag. 2



Nivel 5

Elementos de juego. Ornamentos del nivel



Nombre de
archivo: Sprite11



Nombre de
archivo: FONDO5



Nombre de
archivo: Sprite12



Nombre de
archivo: Sprite13

Nivel 5

Elementos de juego. Enemigos



Nombre de
archivo: enemy4Lev5



Nombre de
archivo: Enemy3Lev5



Nombre de
archivo: Enemy2Lev5



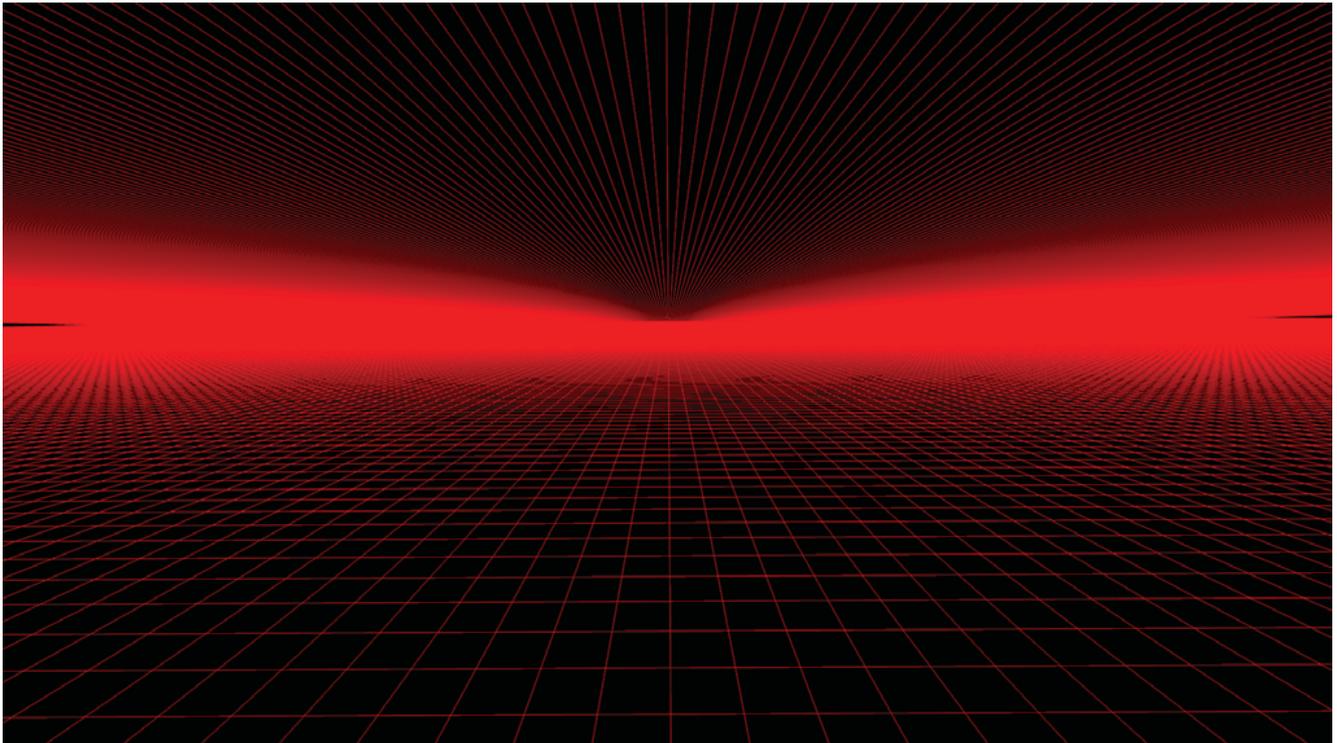
Nombre de
archivo: Enemy1Lev5



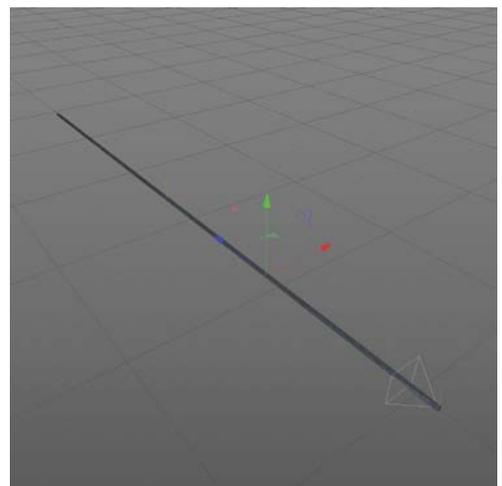
Nombre de
archivo: TROFEO5

Nivel 5

Elementos de juego. Fondo



Nombre de
archivo: Sprite12



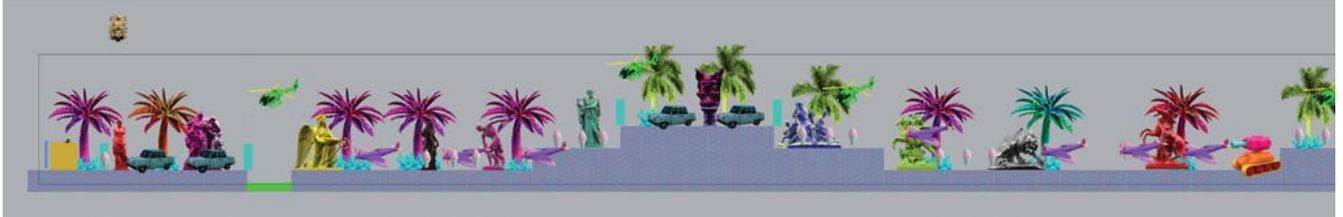
Nivel 6

Pantalla de nivel de juego



Nivel 6

Hoja de nivel de juego pag. 1



Nivel 6

Hoja de nivel de juego pag. 2



Nivel 6

Elementos de juego. Ornamentos del nivel pag. 1



Nombre de
archivo: ESCULTURA1lev6



Nombre de
archivo: ESCULTURA2lev6



Nombre de
archivo: ESCULTURA3lev6



Nombre de
archivo: ESTATUA5lev6



Nombre de
archivo: ESTATUA6lev6



Nombre de
archivo: ESCULTURA7lev6

Nivel 6

Elementos de juego. Ornamentos del nivel pag. 2



Nombre de
archivo: ESCULTURA4lev6



Nombre de
archivo: ESTATUA8lev6



Nombre de
archivo: ESCULTURA9lev6



Nombre de
archivo: ESTATUA10lev6



Nombre de
archivo: ESTATUA11lev6



Nombre de
archivo: ESTATUA12lev6

Nivel 6

Elementos de juego. Ornamentos del nivel pag. 3



Nombre de
archivo: ESTATUA13lev6



Nombre de
archivo: ESTATUA14lev6



Nombre de
archivo: ESTATUA15lev6



Nombre de
archivo: ESTATUAS16lev6



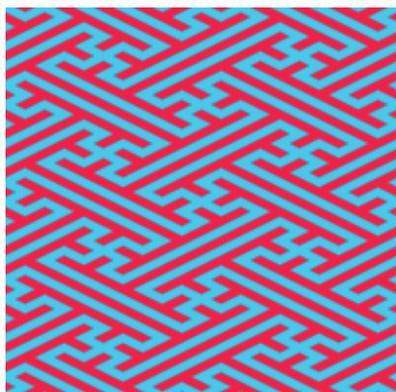
Nombre de
archivo: PLAMERA1lev6



Nombre de
archivo: palmera2LEV6

Nivel 6

Elementos de juego. Ornamentos del nivel pag. 4



Nombre de
archivo: Sprite14



Nombre de
archivo: ARBUSTOS1lev6



Nombre de
archivo: TROFEO6

Nivel 6

Elementos de juego. Enemigos



Nombre de
archivo: cocheLev6



Nombre de
archivo: helicopteroLev6



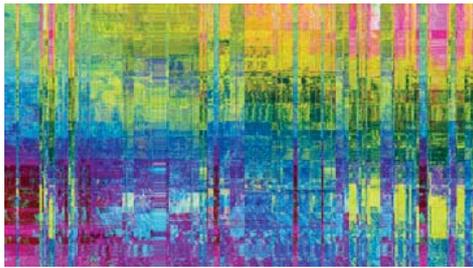
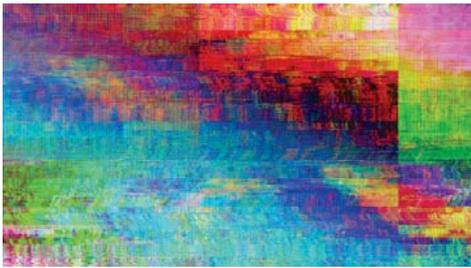
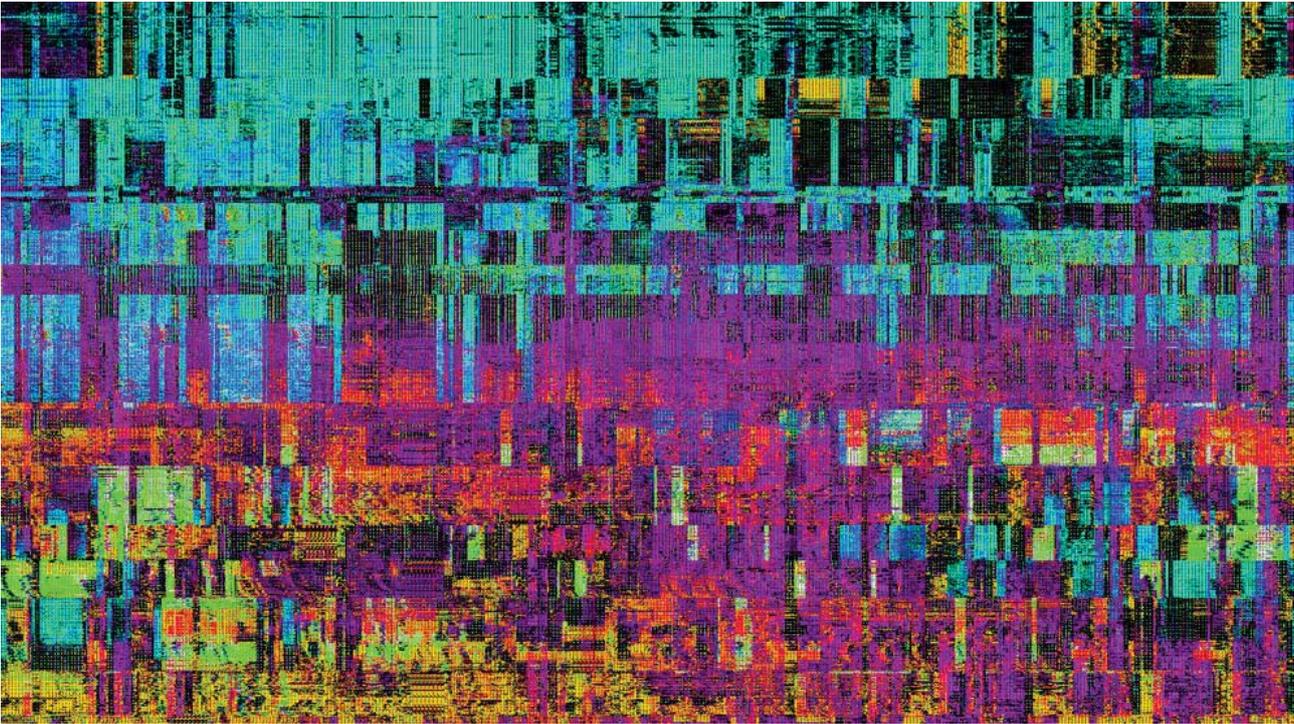
Nombre de
archivo: AvionLev6



Nombre de
archivo: TankLev6

Nivel 6

Elementos de juego. Fondo



Elementos Genéricos presentes en todos los niveles pag. 1



Llave para desbloquear el cofre
nombre de archivo: key2



Cofre de donde sale una figurilla dorada
nombre de archivo: Chest

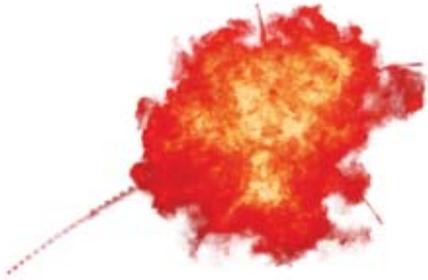


Helado, equivale a un punto en el HUD, si se suman 100 helados se agrega una vida al jugador
nombre de archivo: CRANEOLEV1

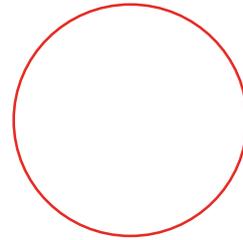


Si el jugador lo toca pierde sus vidas, indica caer al vacío
nombre de archivo: BASErestart

Elementos Genéricos presentes en todos los niveles pag. 2

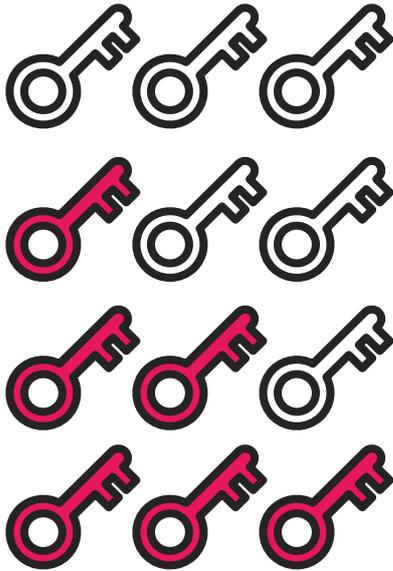
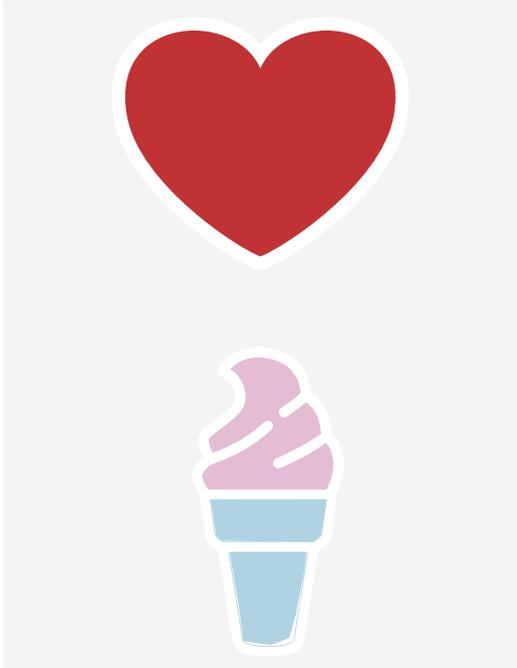
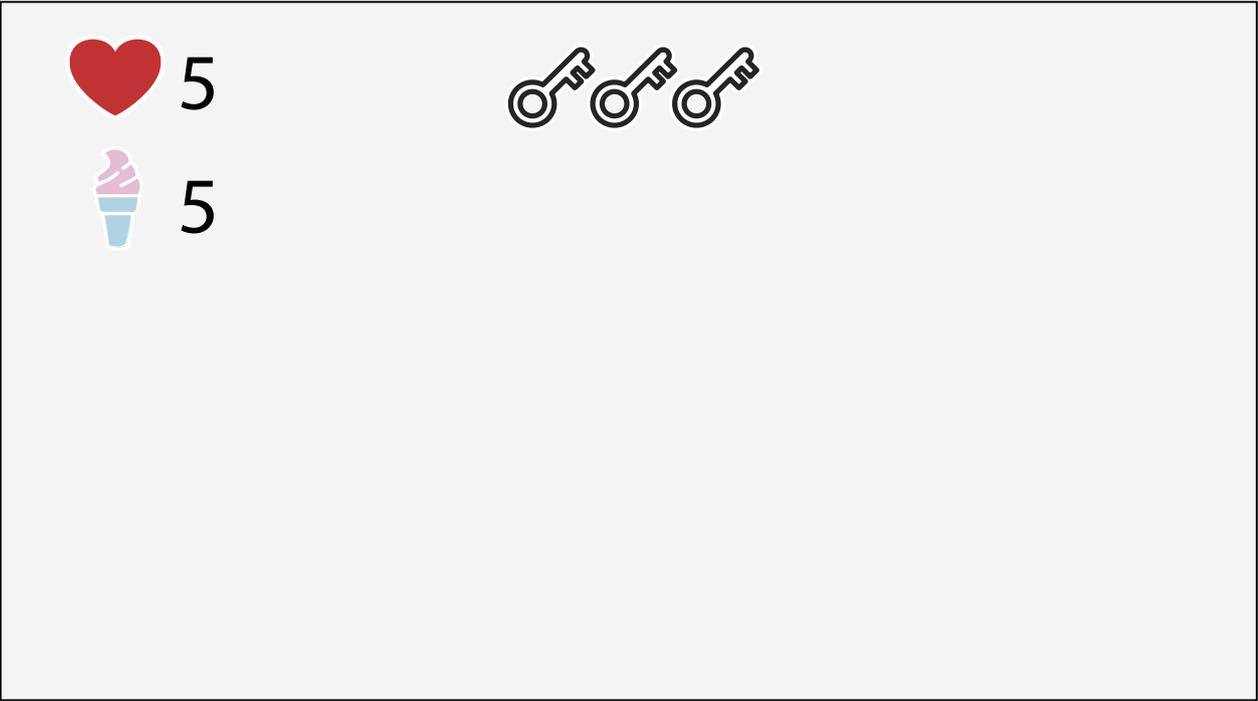


nombre de archivo: EXPLOSION



nombre de archivo: BALA1

HUD

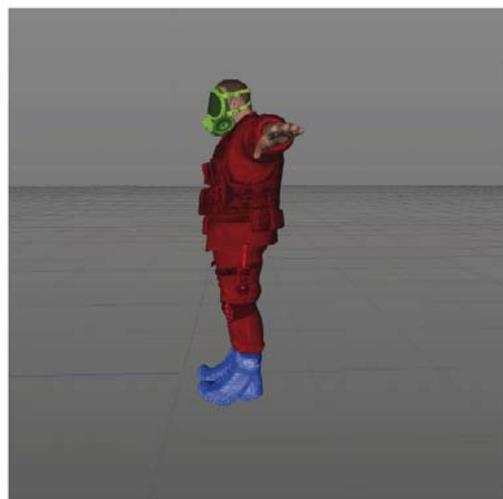
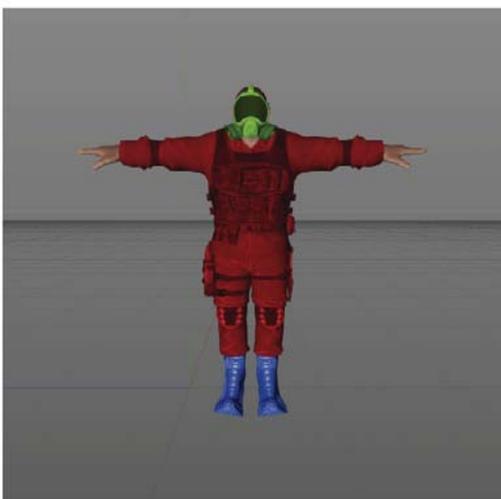
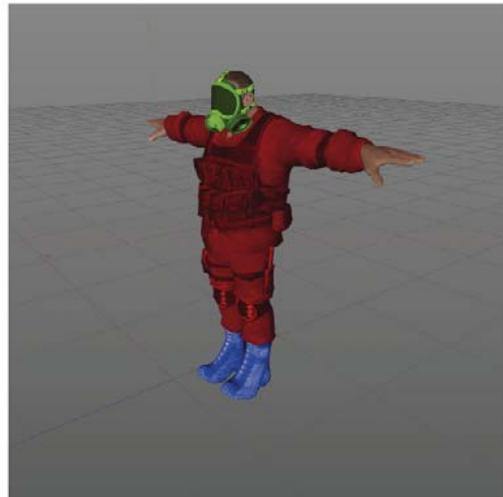


Personaje principal, modelo 3d



Personaje Principal

Modelo creado por medio de
Adobe FUSE
Nombre de archivo: PERSONAJE



Mapeo 3d



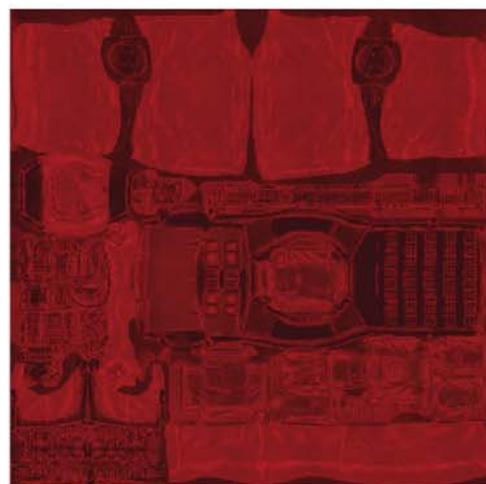
Bottom_Diffuse



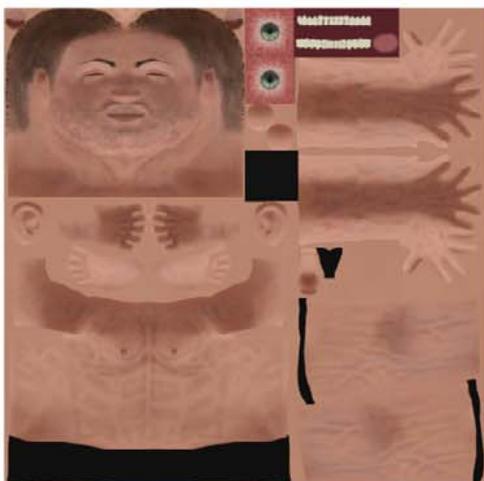
Shoes_Diffuse



Mask_Diffuse



Top_Diffuse



Body_Diffuse

Tabla de texturas usadas para mapear el personaje 3D pintadas en Photoshop

Animación de personaje

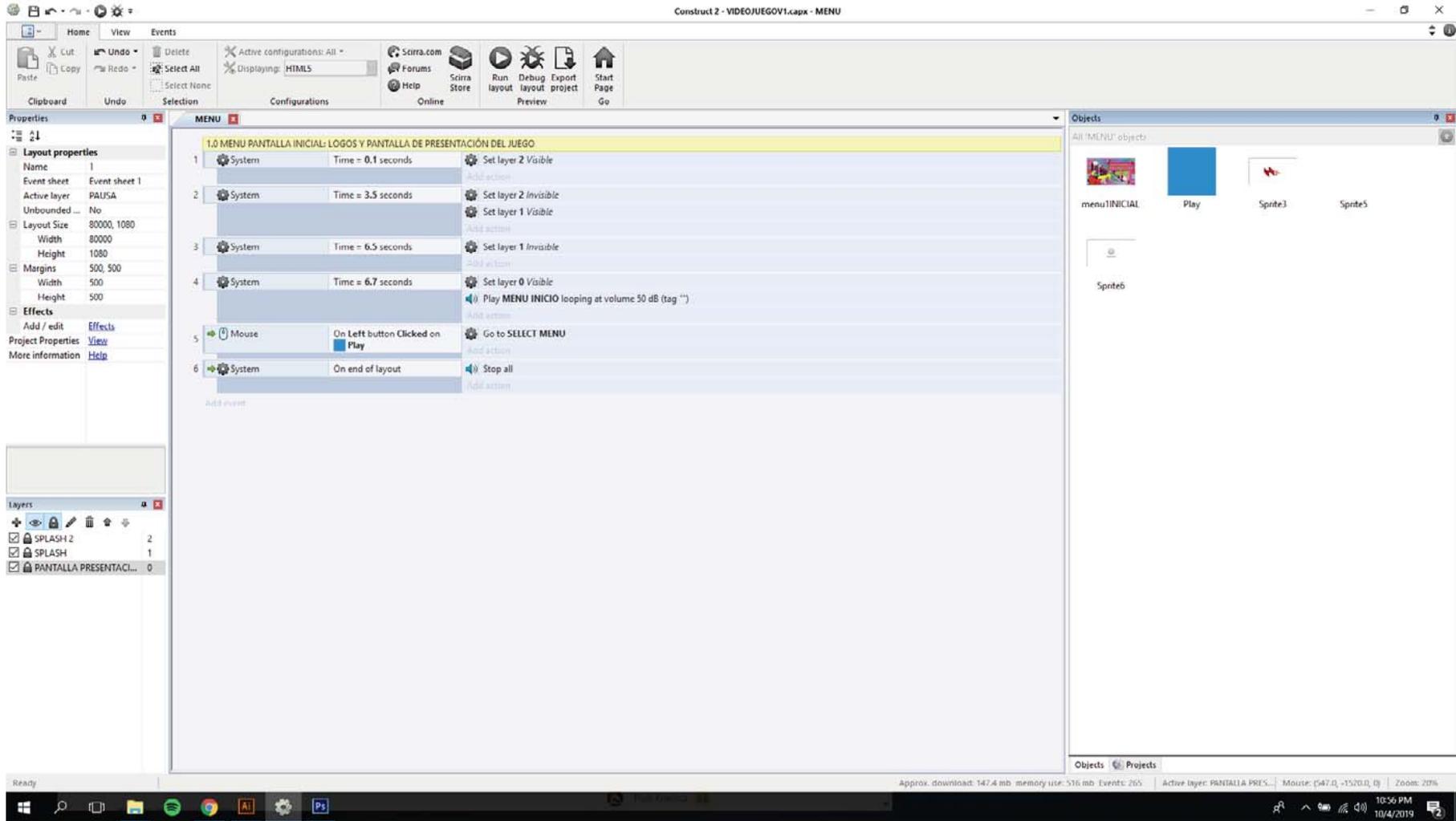


Cazareliquias

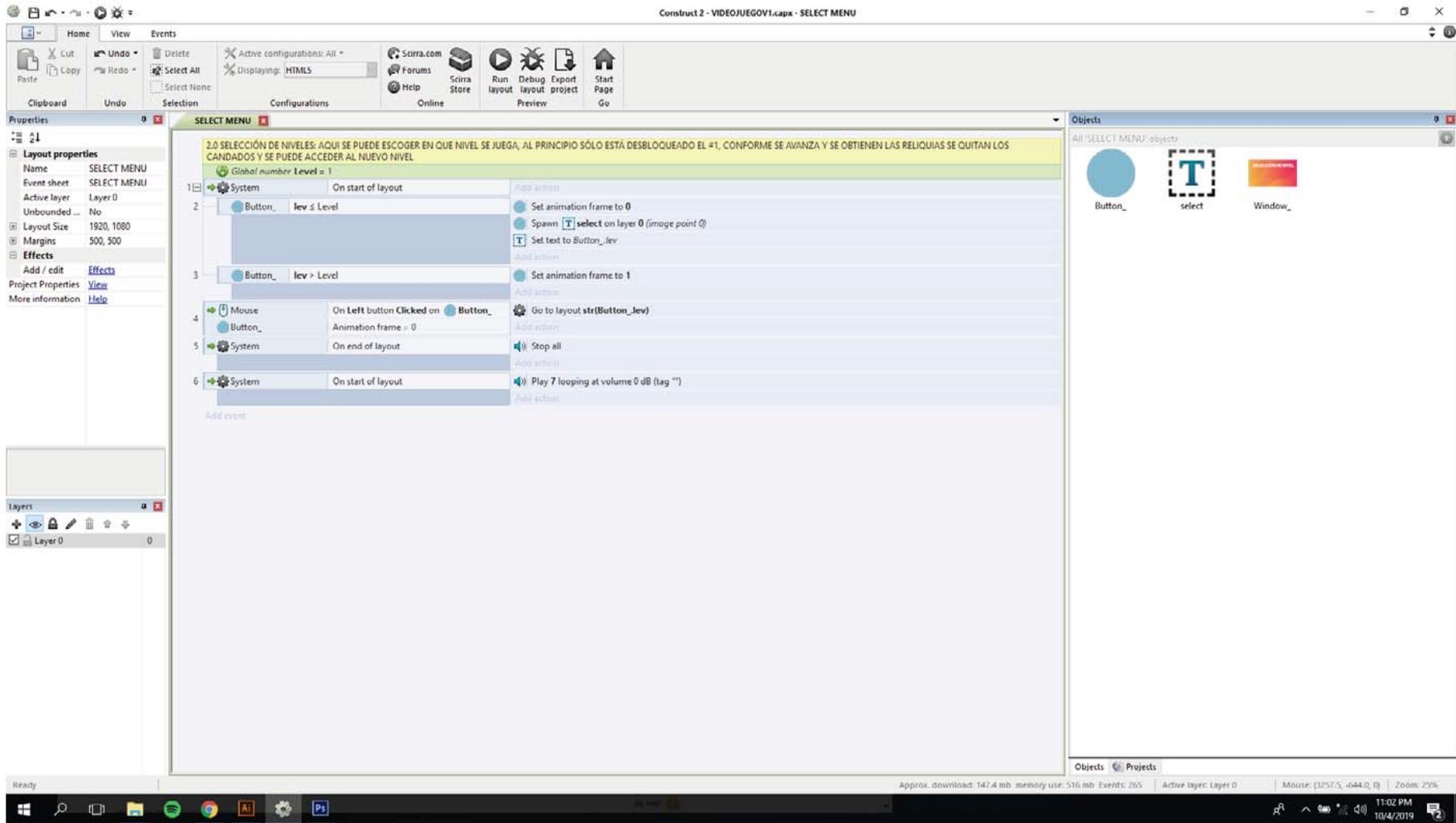
Construct 2

Hojas de Interacción

1.0 Menú principal



2.0 Selección de Nivel



3.0 Nivel 1

Construct 2 - VIDEOJUEGOV1.ccapx - Event sheet 1

Properties

- Layout properties
- Name: SELECT MENU
- Event sheet: SELECT MENU
- Active layer: Layer 0
- Unbounded: No
- Layout Size: 1920, 1080
- Margins: 500, 500
- Effects
- Add / edit: Effects
- Project Properties: View
- More information: Help

Event sheet 1

3.0 PRIMER NIVEL DEL JUEGO, SE DEBEN DE TOMAR LAS 3 LLAVES PARA ABRIR EL COFRE QUE CONTIENE LA RELIQUIA. ESTE NIVEL SIRVE DE BASE PARA LOS 3 NIVELES SIGUIENTES Y ESTA HOJA DE EVENTOS DEBE DE SER INCLUIDA EN LAS DEMAS HOJAS POSTERIORES.

- Global number PAUSA = 0
- Global number color = 0
- Global number FIGURILLA = 0
- Global text state = ""
- Global number playerHeart = 6
- Global number Gem = 0

1 System On start of layout

- System: Set layer "PAUSA" Invisible
- System: Set layer "POPUP" Invisible
- System: Set state to ""
- System: Set group "PLAYER" Activated

2 System On start of layout

- PERSONAJE: Set position to PLAYER (image point 0)
- PERSONAJE: Pin Pin to PLAYER (Position & angle)
- SpikesBallLev1: Set Platform Enabled
- System: Set time scale to 1.0
- System: Reset global variables to default

3 game logic

4 PLAYER On collision with CRANEOLEV1

- CRANEOLEV1: Destroy
- CRANEOLEV1: Spawn shockwave on layer "JUEGO" (image point 0)
- System: Add 1 to Gem

5 shockwave On any animation finished

- shockwave: Destroy

6 System Every tick

- System: Add action

7 numeroslevel type = "gem"

- numeroslevel: Set text to Gem

8 numeroslevel type = "heart"

- numeroslevel: Set text to playerHeart

9 System playerHeart ≤ 0

- System: Set group "PLAYER" Deactivated
- PLAYER: Set Platform Disabled
- System: Wait 1 seconds
- System: Set layer "POPUP" Visible
- System: Set state to "lose"

10 System Gem = 100

Layers

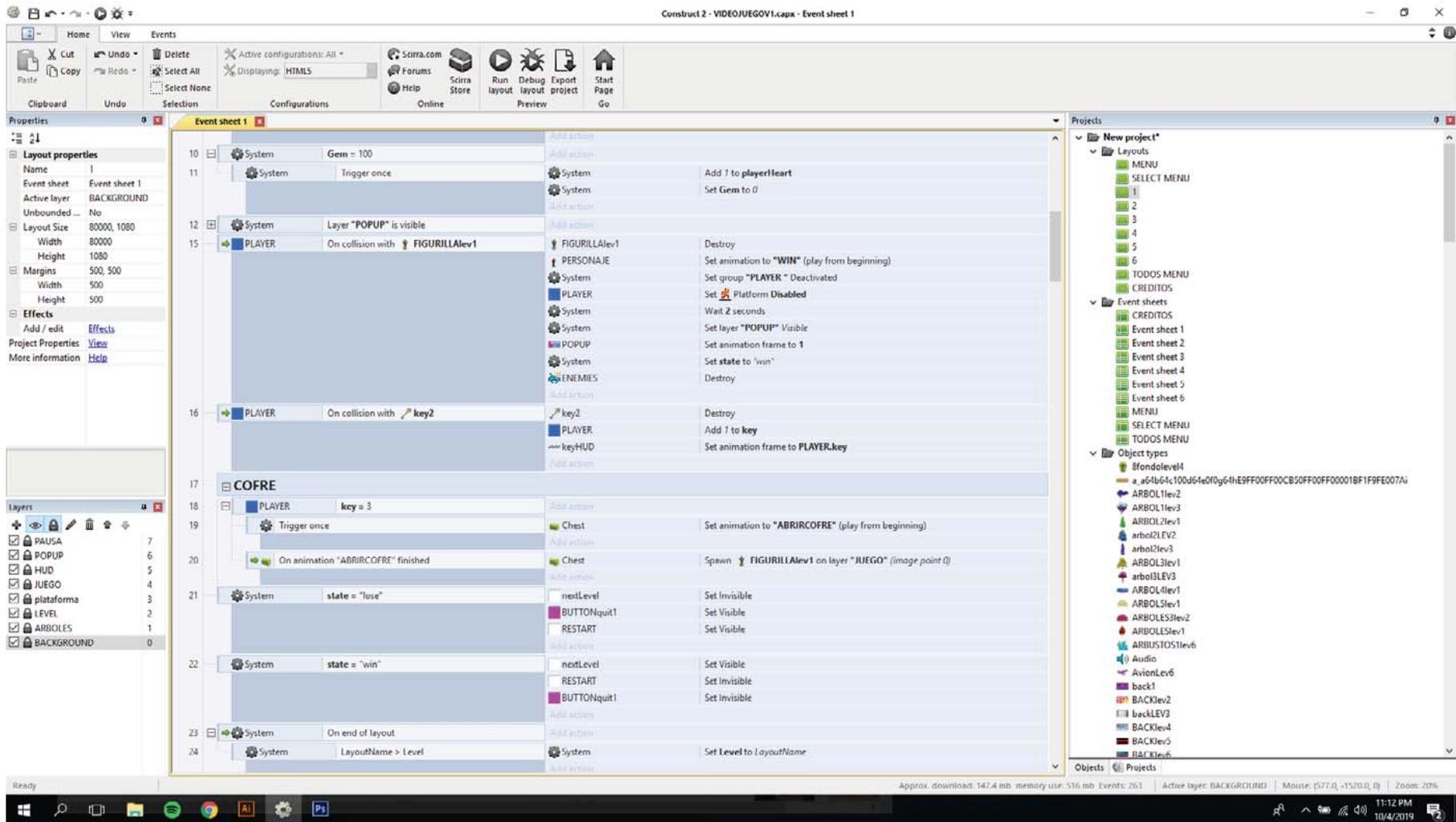
- PAUSA: 7
- POPUP: 6
- HUD: 5
- JUEGO: 4
- plataforma: 3
- LEVEL: 2
- ARBOLES: 1
- BACKGROUND: 0

Objects

- ARBOL2lev1, ARBOL3lev1, ARBOL4lev1, ARBOL5lev1, ARBOLESlev1, back1, BALA1, Barreraspik..., border_enc..., BTN_MEN..., BUTTONq..., CAMERA, CERRAR, Chest, CRANEOL..., enem2, enemy1, EXPLOSION, FIGURILLA..., hearth, HUDCOIN, key2, keyHUD, menuPrins..., muroenemy2, MUSIC1, nextLevel, numeroslevel, PAUSA, PAUSAback, PERSONAJE, PICOSLEV1, platformL..., PLAYER, POPIP, RESTART, shockwave, SpikesBall...

Ready | Approx. download: 147.4 mb | memory use: 516 mb | Events: 263 | Active layer: PAUSA | Mouse: (3257,5, -644,0) | Zoom: 25%

11:11 PM 10/4/2019



Construct 2 - VIDEOJUEGOV1.ccapx - Event sheet 1

Properties

Layout properties

- Name: |
- Event sheet: Event sheet 1
- Active layer: BACKGROUND
- Unbounded: No
- Layout Size: 80000, 1080
- Width: 80000
- Height: 1080
- Margins: 500, 500
- Width: 500
- Height: 500

Effects

Project Properties

Layers

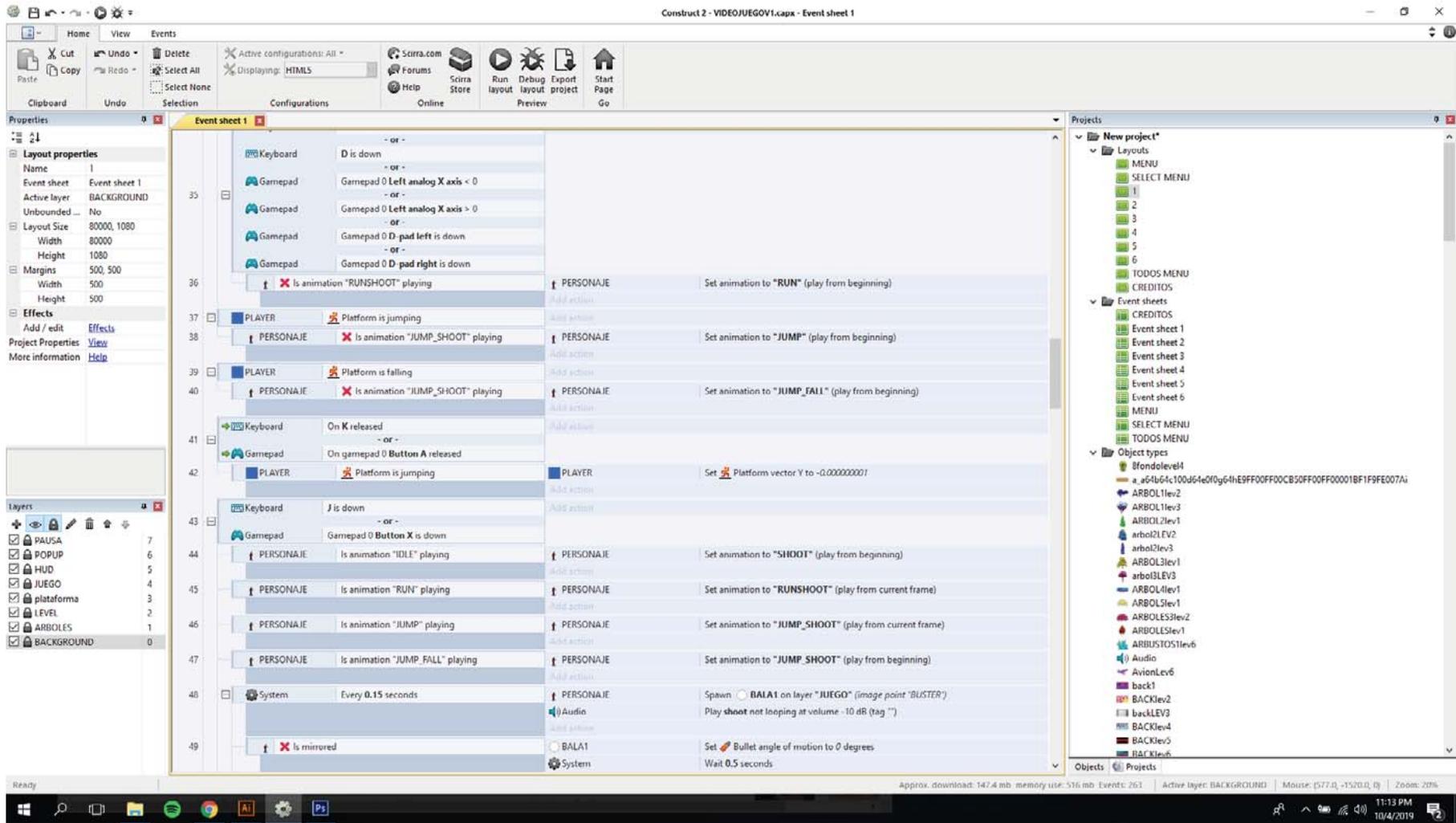
- PAUSA: 7
- POPOP: 6
- HUD: 5
- JUEGO: 4
- plataforma: 3
- LEVEL: 2
- ARBOLLES: 1
- BACKGROUND: 0

Event sheet 1

Event	Condition	Action
System	On end of layout	Add action
System	LayoutName > Level	System: Set Level to LayoutName
Mouse	On Left button Clicked on menuPrincipal	System: Go to SELECT MENU
System	Layer 6 is visible	Add action
Mouse	On Left button Clicked on menuPrincipal	System: Go to SELECT MENU
System	Layer "PAUSA" is visible	Add action
Mouse	On Left button Clicked on BUTTONquit1	Browser: Close
System	Layer "PAUSA" is visible	Add action
PLAYER		
Keyboard	A is down	PLAYER: Simulate Platform pressing Left
Gamepad	Gamepad 0 Left analog X axis < 0	PERSONAJE: Set Mirrored
Gamepad	Gamepad 0 D-pad left is down	- or -
Keyboard	D is down	PLAYER: Simulate Platform pressing Right
Gamepad	Gamepad 0 Left analog X axis > 0	PERSONAJE: Set Not mirrored
Gamepad	Gamepad 0 D-pad right is down	- or -
Keyboard	K is down	PLAYER: Simulate Platform pressing Jump
Gamepad	Gamepad 0 Button A is down	- or -
Keyboard	On K pressed	Audio: Play JUMP not looping at volume 300 dB (tag "")
Gamepad	On gamepad 0 Button A pressed	- or -
PLAYER	Platform is on floor	Add action
Keyboard	A is down	PERSONAJE: Set animation to "IDLE" (play from beginning)
Keyboard	D is down	Add action
Gamepad	Gamepad 0 Left analog X axis < 0	- or -
Gamepad	Gamepad 0 Left analog X axis > 0	- or -
Gamepad	Gamepad 0 D-pad left is down	- or -
Gamepad	Gamepad 0 D-pad right is down	- or -
Keyboard	A is down	Add action
Keyboard	D is down	- or -

Projects

- New project*
- Layouts
 - MENU
 - SELECT MENU
 - 1
 - 2
 - 3
 - 4
 - 5
 - 6
- TODOS MENU
- CREDITOS
- Event sheets
 - CREDITOS
 - Event sheet 1
 - Event sheet 2
 - Event sheet 3
 - Event sheet 4
 - Event sheet 5
 - Event sheet 6
- MENU
- SELECT MENU
- TODOS MENU
- Object types
 - 8fondolev4
 - a_a64b64c100d64e0f0y64hE9FF00FF00C850FF00FF00001BF1F9FE007A
 - ARBOL1lev2
 - ARBOL1lev3
 - ARBOL2lev1
 - arbol2lev2
 - arbol2lev3
 - ARBOL3lev1
 - arbol3lev3
 - ARBOL4lev1
 - ARBOL5lev1
 - ARBOL53lev2
 - ARBOL5lev1
 - ARBOL53lev2
 - ARBOL5lev1
 - ARBUSTOS1lev6
 - Audio
 - AvionLev6
 - back1
 - BACKlev2
 - backLEV3
 - BACKlev4
 - BACKlev5
 - BACKlev6



Construct 2 - VIDEOJUEGOV1.ccapx - Event sheet 1

Clipboard: Paste, Copy, Cut, Undo, Redo, Select All, Select None, Selection

Configurations: Active configurations: All, Displaying: HTML5

Online: Scirra.com, Forums, Help, Scirra Store

Preview: Run layout, Debug layout, Export project, Start Page Go

Properties: Layout properties (Name, Event sheet, Active layer, Unbounded, Layout Size, Margins, Effects)

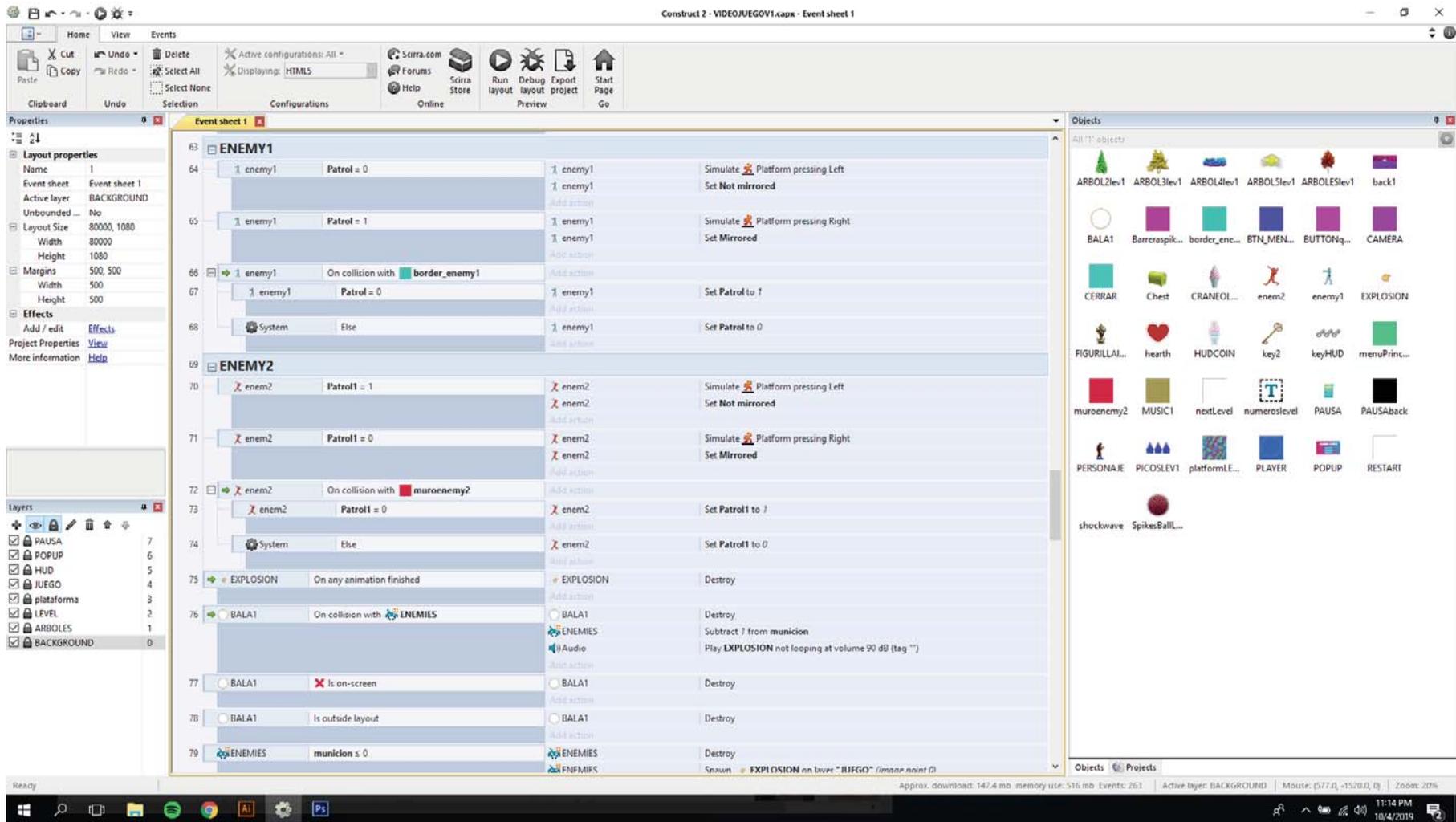
Event sheet 1

49	Is mirrored	BALA1	System	Set Bullet angle of motion to 0 degrees
		BALA1	System	Wait 0.5 seconds
		BALA1		Destroy
50	Is mirrored	BALA1	System	Set Bullet angle of motion to 180 degrees
		BALA1	System	Wait 0.5 seconds
		BALA1		Destroy
51	Keyboard	On J released		
	Gamepad	On gamepad 0 Button X released		
52	PERSONAJE	Is animation "RUNSHOOT" playing	PERSONAJE	Set animation to "RUN" (play from beginning)
53	PERSONAJE	Is animation "SHOOT" playing	PERSONAJE	Set animation to "IDLE" (play from beginning)
54	PERSONAJE	Is animation "JUMP_SHOOT" playing	PERSONAJE	Set animation to "JUMP" (play from current frame)
55	camera			
56	System	Every tick	CAMERA	Set position to (lerp(SeIf.X,PLAYER.ImagePointX(1),0.05) , lerp(SeIf.Y,PLAYER.ImagePointY(1),0.05))
57	WIN			
60	PLAYER	Is overlapping ENEMIES	Function	Call "ON_DAMAGE" ()
	PLAYER	On collision with ENEMIES	ENEMIES	Destroy
			ENEMIES	Spawn EXPLOSION on layer 1 (image point 0)
			Audio	Play EXPLOSION not looping at volume 90 dB (tag "")
61	Function	On "ON_DAMAGE"	PERSONAJE	Set animation to "DAMAGE" (play from beginning)
	PERSONAJE	Is animation "DAMAGE" playing	System	Set group "PLAYER" Deactivated
			System	Subtract 1 from playerHeart
			System	Wait 2.0 seconds
62	PERSONAJE	On animation "DAMAGE" finished	System	Set group "PLAYER" Activated
63	ENEMY1			
64	enemy1	Patrol = 0	enemy1	Simulate Platform pressing Left

Layers: PAUSA, POPUP, HUD, JUEGO, plataformas, LEVEL, ARBOLES, BACKGROUND

Objects: ARBOL2lev1, ARBOL3lev1, ARBOL4lev1, ARBOL5lev1, ARBOLESlev1, back1, BALA1, Barreraspik..., border_enc..., BTN_MEN..., BUTTONq..., CAMERA, CERRAR, Chest, CRANEOL..., enem2, enemy1, EXPLOSION, FIGURILLAL..., hearth, HUDCOIN, key2, keyHUD, menuPrins..., mureoenemy2, MUSIC1, nextLevel, numeroslevel, PAUSA, PAUSAback, PERSONAJE, PICOSLEV1, platformLE..., PLAYER, POPIP, RESTART, shockwave, SpikesBall...

Ready | Approx. download: 147.4 mb | memory use: 516 mb | Events: 263 | Active layer: BACKGROUND | Mouse: (577,0) -15200,0 | Zoom: 20% | 11:14 PM 10/4/2019



Construct 2 - VIDEOJUEGOV1.cpx - Event sheet 1

Clipboard: Paste, Copy, Cut, Undo, Redo, Select All, Select None, Selection

Configurations: Active configurations: All, Displaying: HTML5

Tools: Scirra.com, Scirra Store, Run layout, Debug layout, Export project, Start Page Go

Properties: Layout properties (Name, Event sheet, Active layer, Unbounded, Layout Size, Margins, Effects), Project Properties, More information

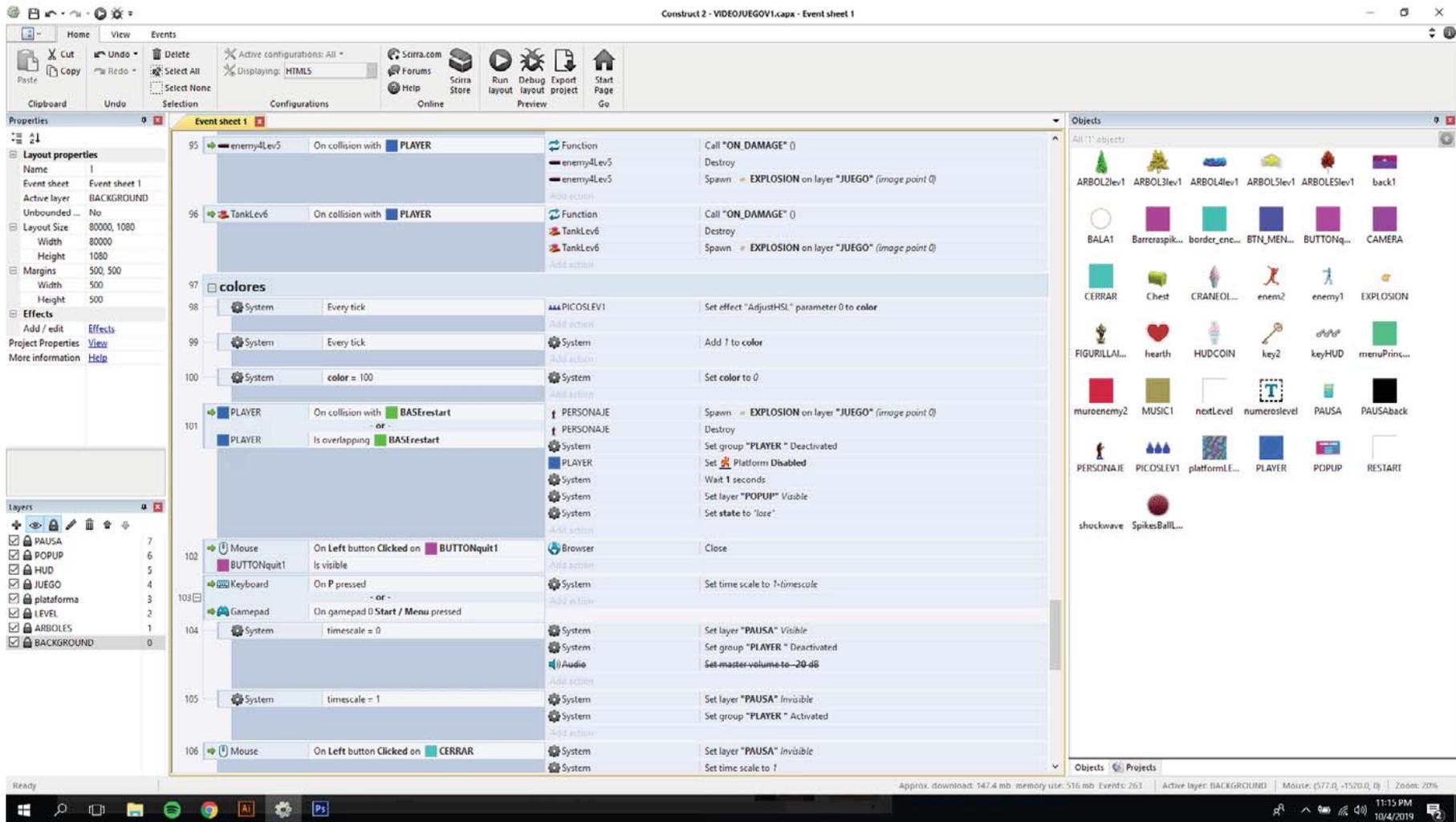
Event sheet 1

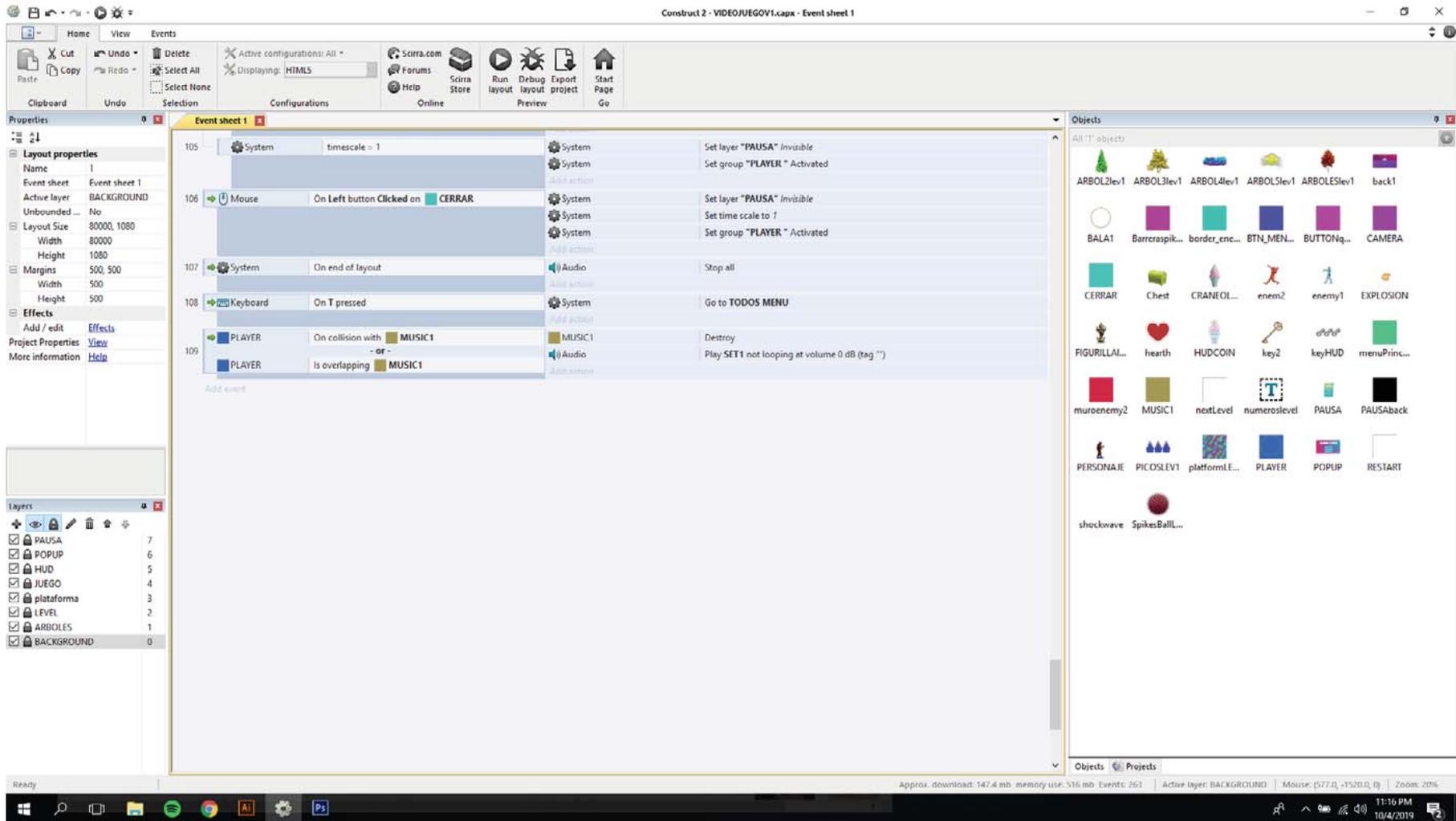
79	ENEMIES	municion <= 0	ENEMIES	Destroy
			ENEMIES	Spawn - EXPLOSION on layer "JUEGO" (image point 0)
80	BALA1	On collision with platformLEVEL1	BALA1	Destroy
81	BALA1	On collision with LEVEL2	BALA1	Destroy
82	BALA1	On collision with level3Back	BALA1	Destroy
83	BALA1	On collision with lev4BACK	BALA1	Destroy
84	BALA1	On collision with destruccionBalaBack	BALA1	Destroy
85	BALA3	On collision with level3Back	BALA3	Destroy
SPIKEBALL				
87	SpikesBall.ev1	Platform has wall to left	SpikesBall.ev1	Simulate Platform pressing Jump
88	SpikesBall.ev1	Platform has wall to right	SpikesBall.ev1	Simulate Platform pressing Jump
89	SpikesBall.ev1	Patrol = 0	SpikesBall.ev1	Simulate Platform pressing Right Set Not mirrored
90	SpikesBall.ev1	Patrol = 1	SpikesBall.ev1	Simulate Platform pressing Left Set Mirrored
91	SpikesBall.ev1	On collision with Barreraspikeball1	SpikesBall.ev1	Set Patrol to 1
92	SpikesBall.ev1	Patrol = 0	SpikesBall.ev1	Set Patrol to 0
93	System	Else	SpikesBall.ev1	Set Patrol to 0
94	PLAYER	On collision with Enemigosbaselev1	Function	Call "ON_DAMAGE" ()
95	enemy4Lev5	On collision with PLAYER	Function	Call "ON_DAMAGE" ()
			enemy4Lev5	Destroy
			enemultLev5	Spawn - EXPLOSION on layer "JUEGO" (image point 0)

Layers: PAUSA (7), POPUP (6), HUD (5), JUEGO (4), plataformas (3), LEVEL (2), ARBOLES (1), BACKGROUND (0)

Objects: ARBOL2lev1, ARBOL3lev1, ARBOL4lev1, ARBOL5lev1, ARBOLESlev1, back1, BALA1, Barreraspike..., border_enc..., BTN_MEN..., BUTTONq..., CAMERA, CERRAR, Chest, CRANEOL..., enem2, enemy1, EXPLOSION, FIGURILLA..., hearth, HUDCOIN, key2, keyHUD, menuPrins..., muroenemy2, MUSIC1, nextLevel, numeroslevel, PAUSA, PAUSAback, PERSONAJE, PICOSLEV1, platformE..., PLAYER, POPIP, RESTART, shockwave, SpikesBall...

Ready | Approx. download: 147.4 mb | memory use: 516 mb | Events: 263 | Active layer: BACKGROUND | Mouse: (577,0) -15200,0 | Zoom: 20% | 11:15 PM 10/4/2019





3.1 Nivel 2

Construct 2 - VIDEOJUEGOV1.cpx - Event sheet 2

Home View Events

Status Bar Layer Bar Bookmarks Bar
 Properties Bar Z Order Bar Tilemap Bar
 Project Bar Object Bar Configurations Bar

Zoom in Snap to grid Grid width: 32
 Zoom out Show grid Grid height: 32
 Zoom to 100% Show collision polys
 Zoom Grid options Grid size Theme Display

Show collision polys
 Translucent inactive layers

Properties

Layout properties

Name: 2

Event sheet: Event sheet 2

Active layer: PAUSA

Unbounded: No

Layout Size: 60000, 1280

Margins: 500, 500

Effects

Project Properties: [View](#)

More information: [Help](#)

Event sheet 2

3.1 NIVEL 2 - INCLUIR EVENT SHEET 1

Includes: Event sheet 1

1	System	Every 3.0 seconds	LEV2Torreta	Spawn BALA2 on layer "JUEGO" (image point 0) Set Bullet angle of motion to 215 degrees
2	PLAYER	On collision with BASTerstart	PERSONAJE PERSONAJE System PLAYER System System System	Spawn EXPLOSION on layer "JUEGO" (image point 0) Destroy Set group "PLAYER" Deactivated Set Platform Disabled Wait 1 seconds Set layer "POPUP" Visible Set state to "lose"
ENEMY3				
4	enemy1Level2	Patrol = 0	enemy1Level2	Simulate Platform pressing Left Set Not mirrored
5	enemy1Level2	Patrol = 1	enemy1Level2	Simulate Platform pressing Right Set Mirrored
6	enemy1Level2	On collision with border_enemy1		
7		Patrol = 0	enemy1Level2	Set Patrol to 1
8		Else	enemy1Level2	Set Patrol to 0
COFRE2				
10	System	On start of layout	System	Set group "COFRE" Deactivated
11	PLAYER	key = 3		
12		Trigger once	Chest	Set animation to "ABRIRCOFRE" (play from beginning)
13		On animation "ABRIRCOFRE" finished	Chest	Spawn TROFEOLEV2 on layer "JUEGO" (image point 0)
14		On collision with TROFEOLEV2	PERSONAJE	Destroy Set animation to "WIN" (play from beginning)

Objects

All 12 objects

ARBOL1lev2 arbol2LEV2 ARBOLES3L... BACKlev2 BALA1 BALA2

BASTerstart border_enc... CAMERA Chest CRANEOL... enemy1Lev...

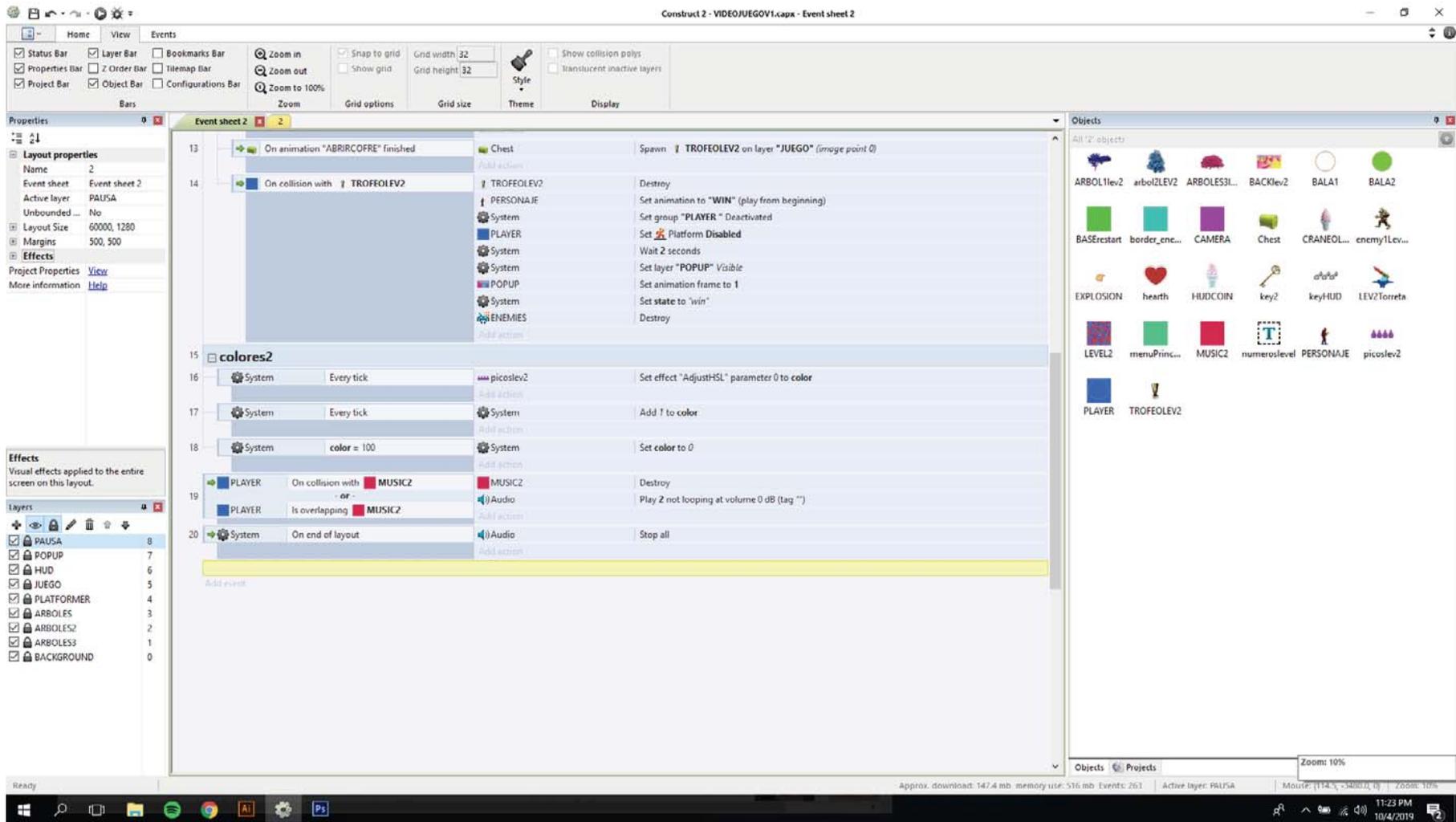
EXPLOSION hearth HUDCOIN key2 keyHUD LEV2Torreta

LEVEL2 menuPrinc... MUSIC2 numeroslevel PERSONAJE picoslev2

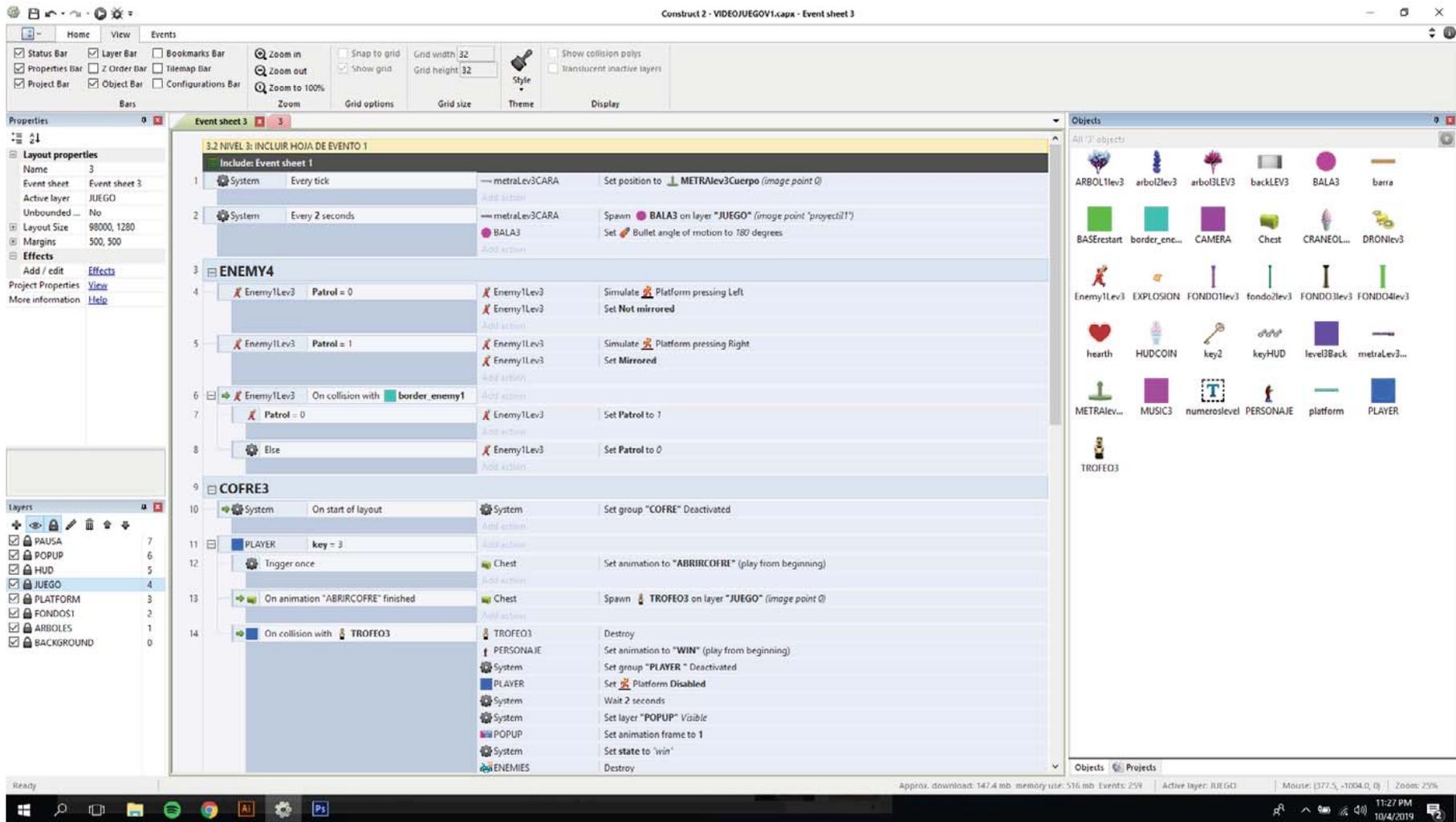
PLAYER TROFEOLEV2

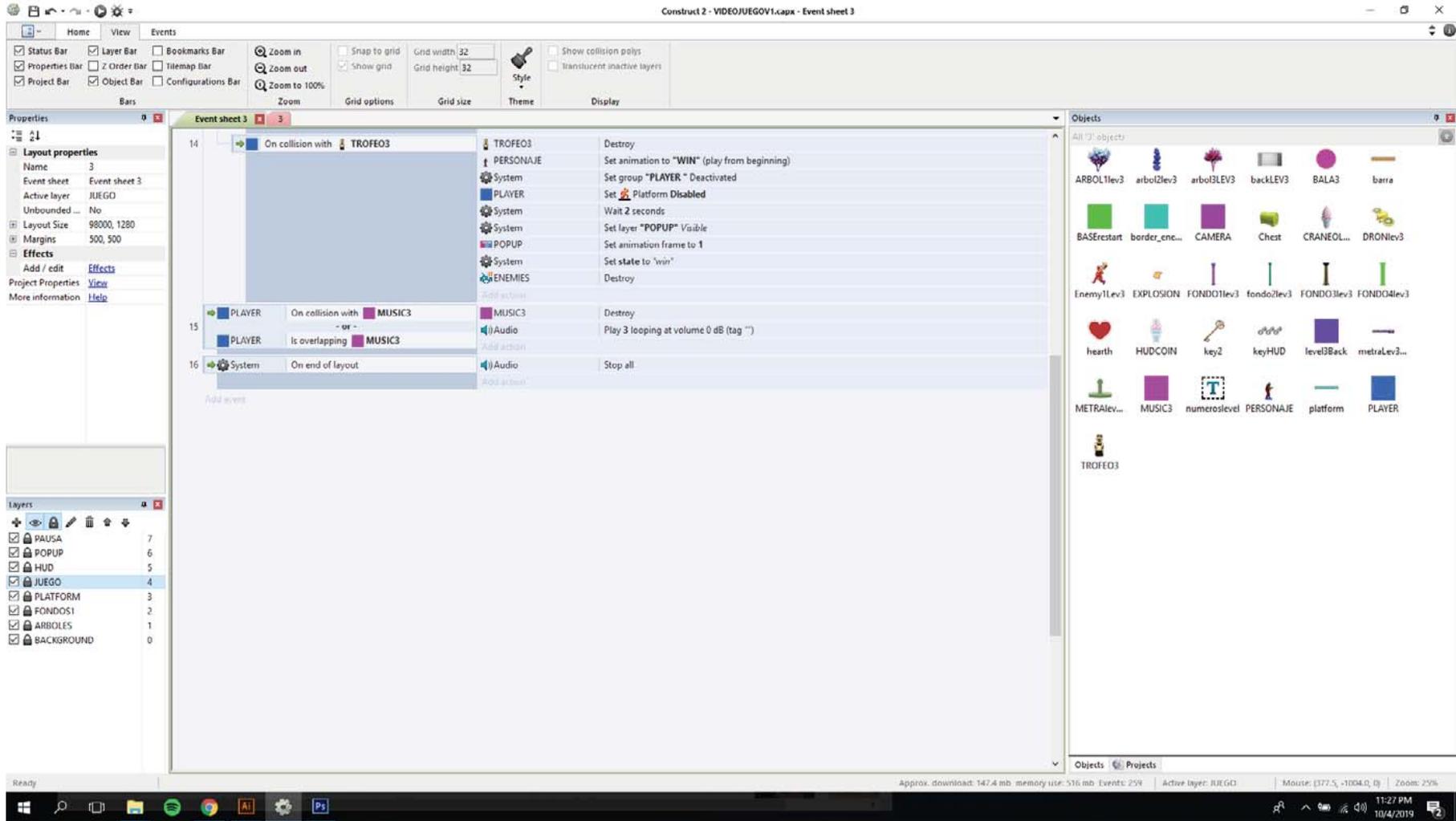
Ready Approx. download: 147.4 mb memory use: 516 mb Events: 263 Active layer: PAUSA Mouse: (114,5, -3480,0) Zoom: 10%

11:23 PM 10/4/2019



3.2 Nivel 3





3.3 Nivel 4

Construct 2 - VIDEOJUEGOV1.ccapx - Event sheet 4

Home View Events

Properties

Layout properties

Event sheet 4

3.3 NIVEL 4: INCLUIR HOJA DE EVENTO 1

Includes: Event sheet 1

Event	Condition	Action
1	ENEMY5	
2	enemy2Lev4 Patrol = 0	Simulate Platform pressing Left Set Not mirrored
3	enemy2Lev4 Patrol = 1	Simulate Platform pressing Right Set Mirrored
4	enemy2Lev4 On collision with border_enemy1	Set Patrol to 7
5	enemy2Lev4	Set Patrol to 0
6	System Else	Set Patrol to 0
7	ENEMY6	
8	Enemy3Lev4 Patrol = 0	Simulate Platform pressing Left Set Not mirrored
9	Enemy3Lev4 Patrol = 1	Simulate Platform pressing Right Set Mirrored
10	Enemy3Lev4 On collision with border_enemy1	Set Patrol to 7
11	Enemy3Lev4	Set Patrol to 0
12	System Else	Set Patrol to 0
13	COFRE4	
14	System On start of layout	Set group "COFRE" Deactivated
15	PLAYER key = 3	
16	System Trigger once	Set animation-frame to 1 Set animation-frame to 1 Set animation to "ABRIRCOFRE" (play from beginning)
17	Chest On animation "ABRIRCOFRE"	Spawn TROFEO4 on layer "JUEGO" (image point 0)

Objects

8fondolev4 BACKlev4 base BASRestart border_ene... CAMERA
Chest CRANEOL... Enemy1Lev4 enemy2Lev4 Enemy3Lev4 FONDO10L...
FONDO11L... FONDO12L... FONDO11ev4 fondo2LEV4 FONDO3lev4 fondo4LEV4
FONDO5lev4 FONDO6lev4 fondo7LEV4 fondo9LEV4 hearth HUCCOIN
key2 keyHUD levBACK MUSIC4 notLevel numeroslevel
PERSONAJE PLAYER Sprite10 Sprite9 TROFEO4

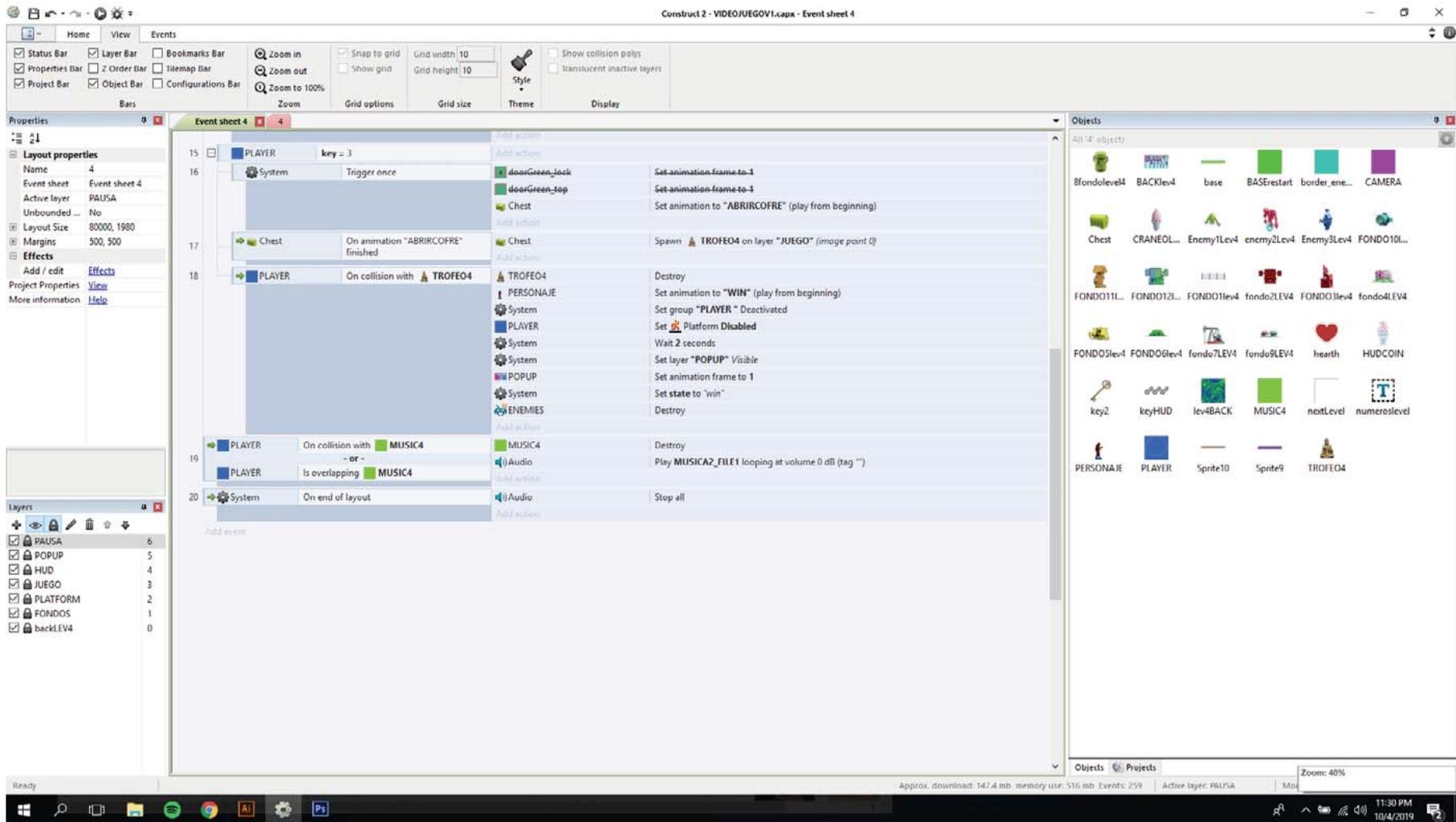
Layers

- PAUSA 6
- POPOP 5
- HUD 4
- JUEGO 3
- PLATFORM 2
- FONDOS 1
- backLEV4 0

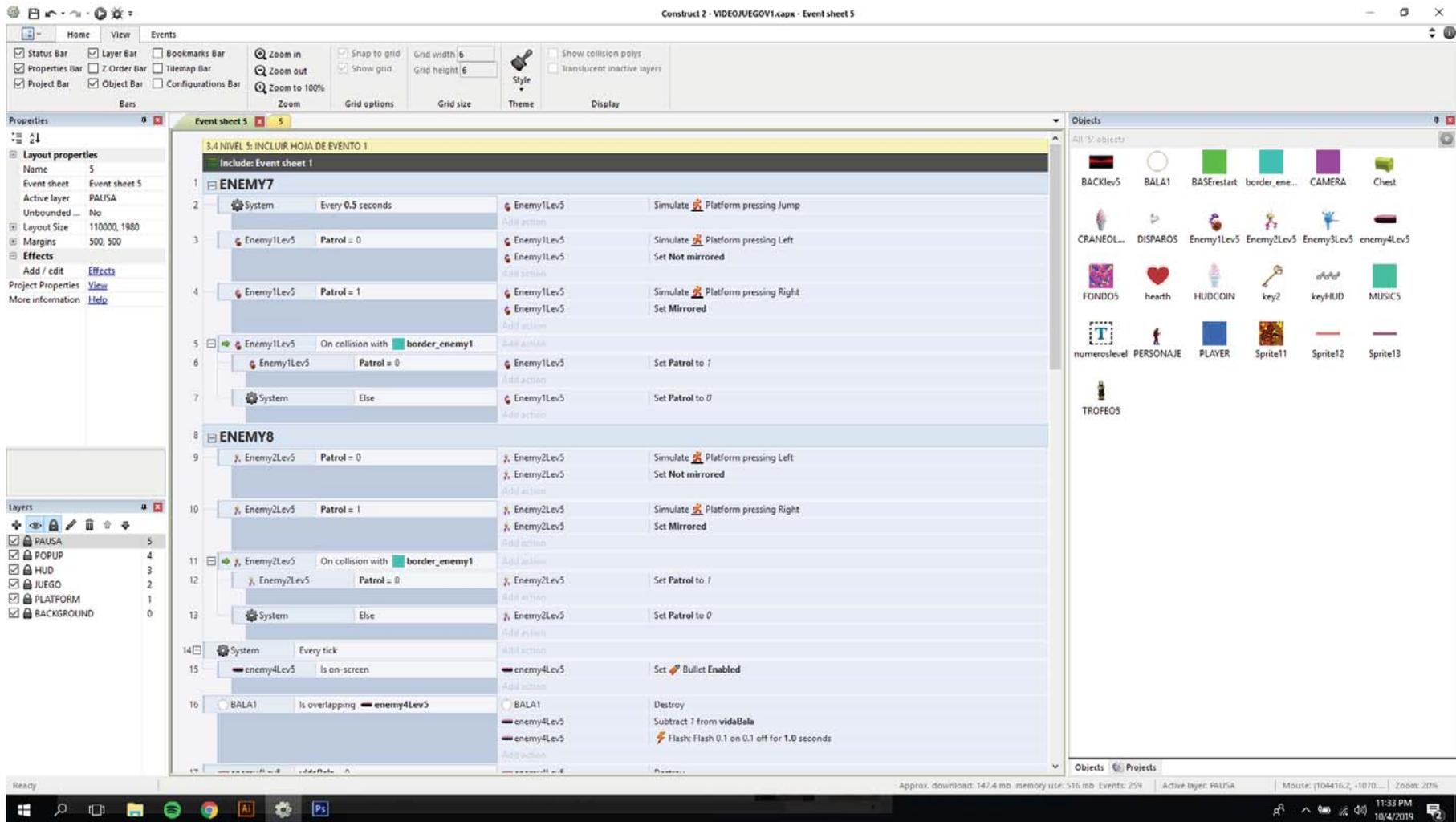
Ready

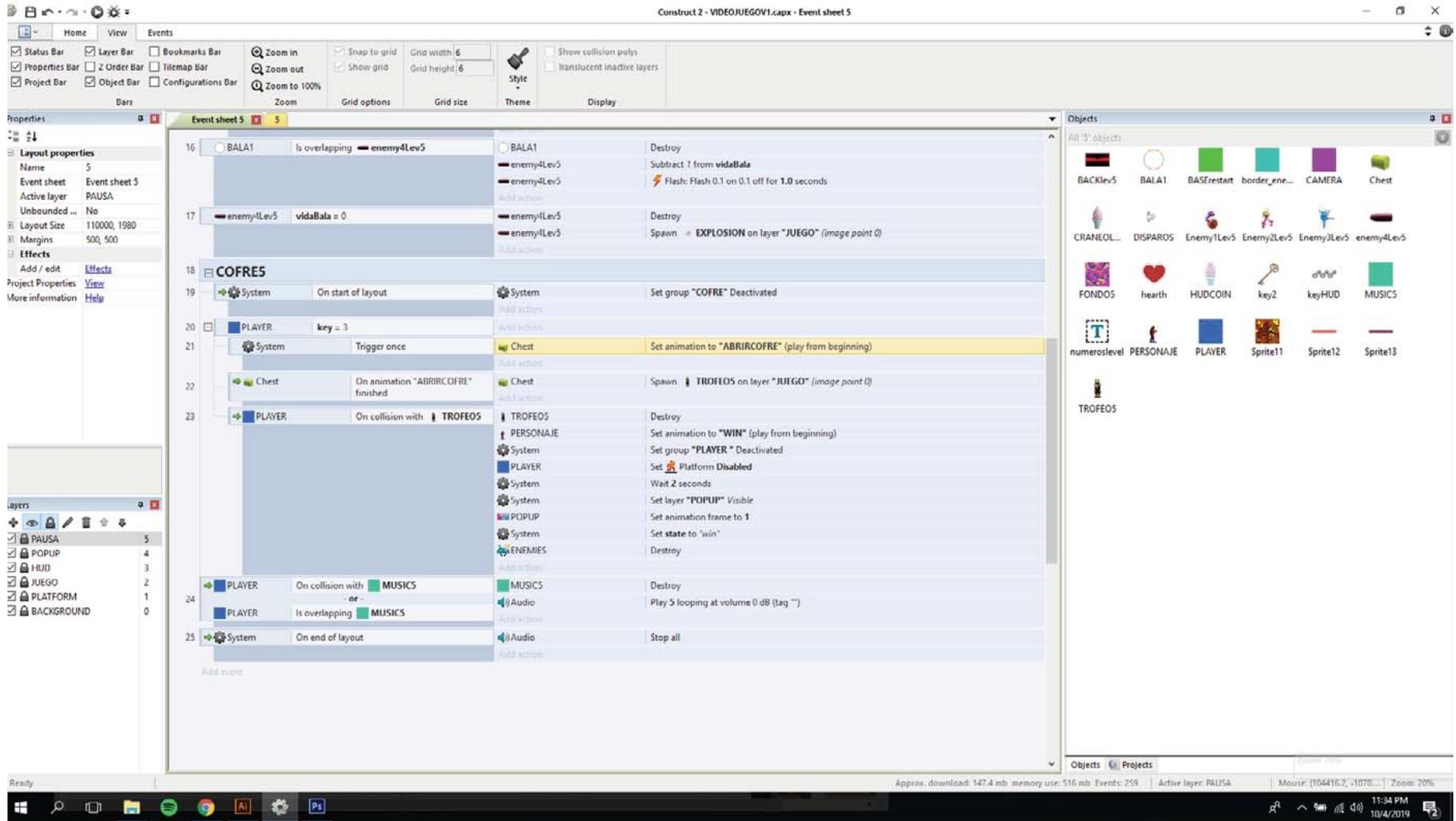
Approx. download: 147.4 mb memory use: 516 mb Events: 259 Active layer: PAUSA Mouse: (7745, 335) Zoom: 40%

11:30 PM 10/4/2019

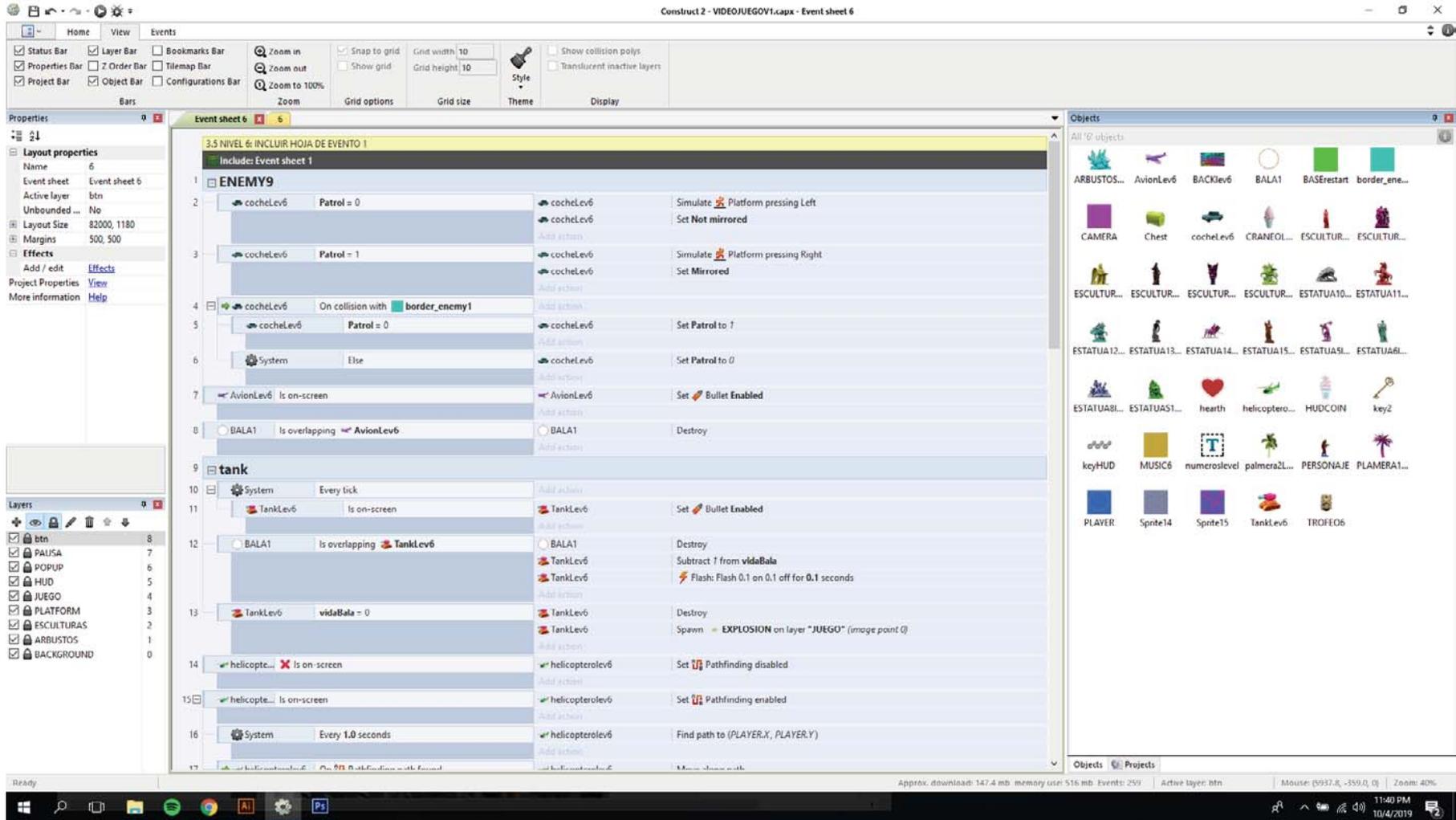


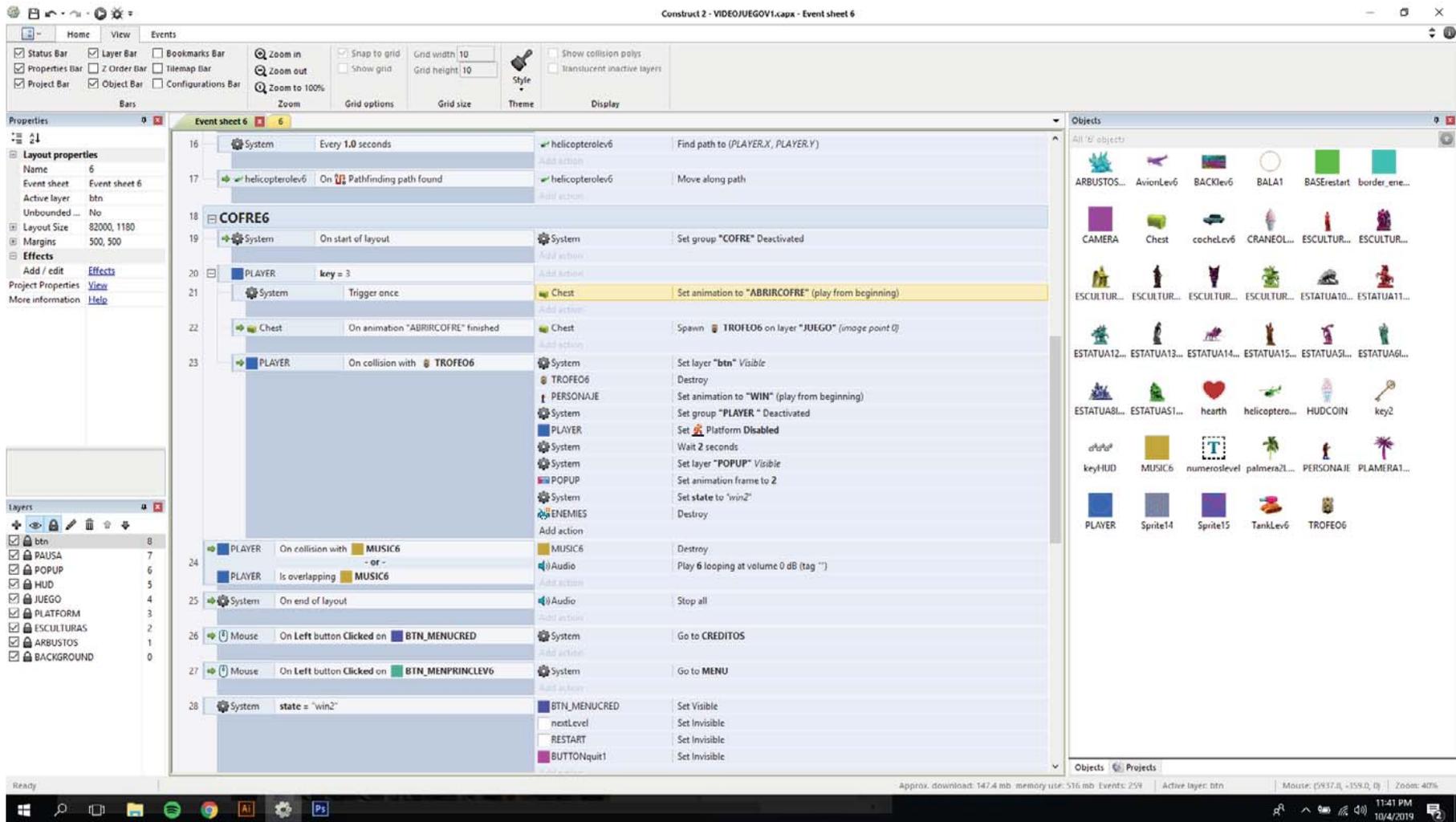
3.4 Nivel 5



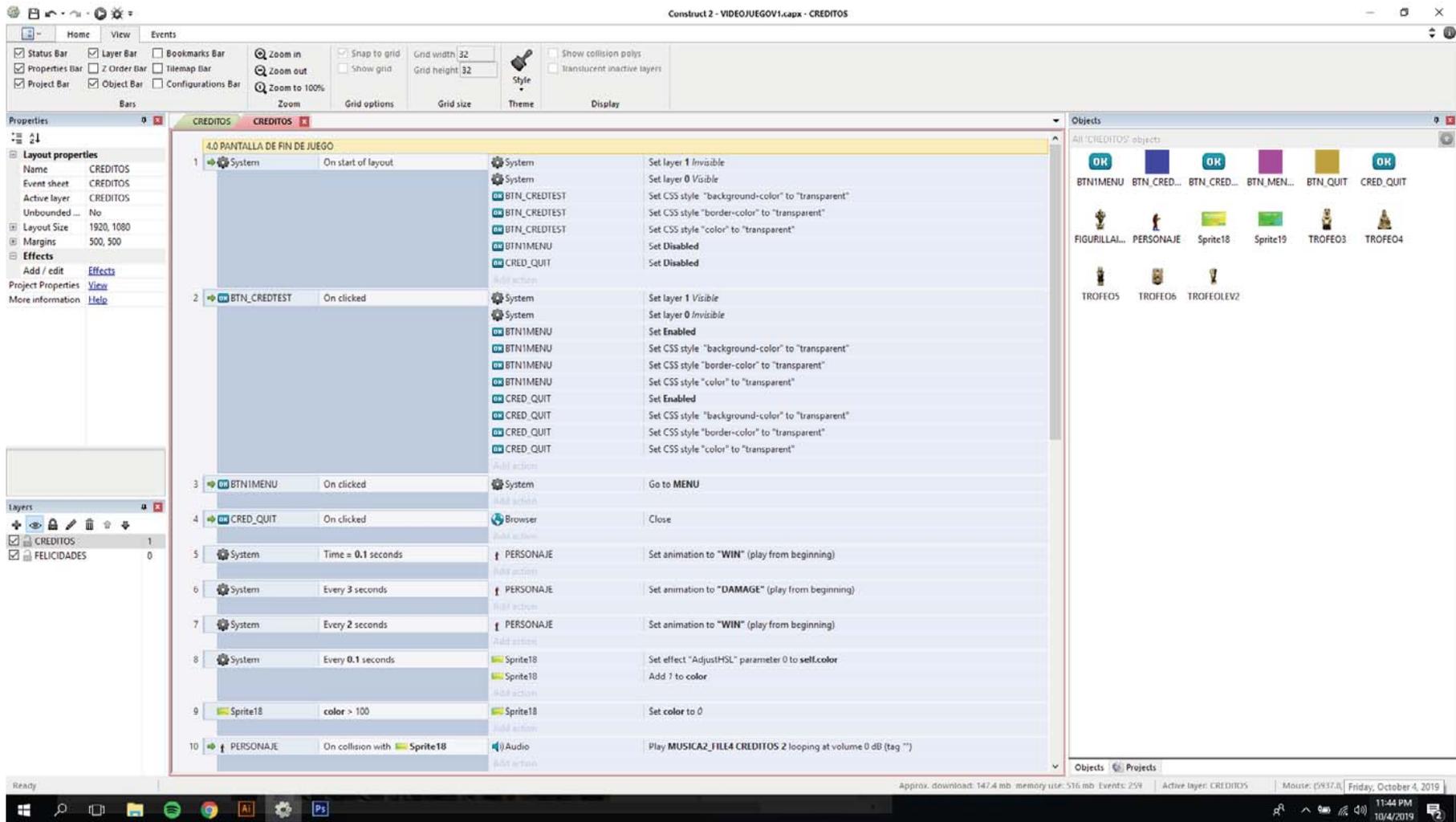


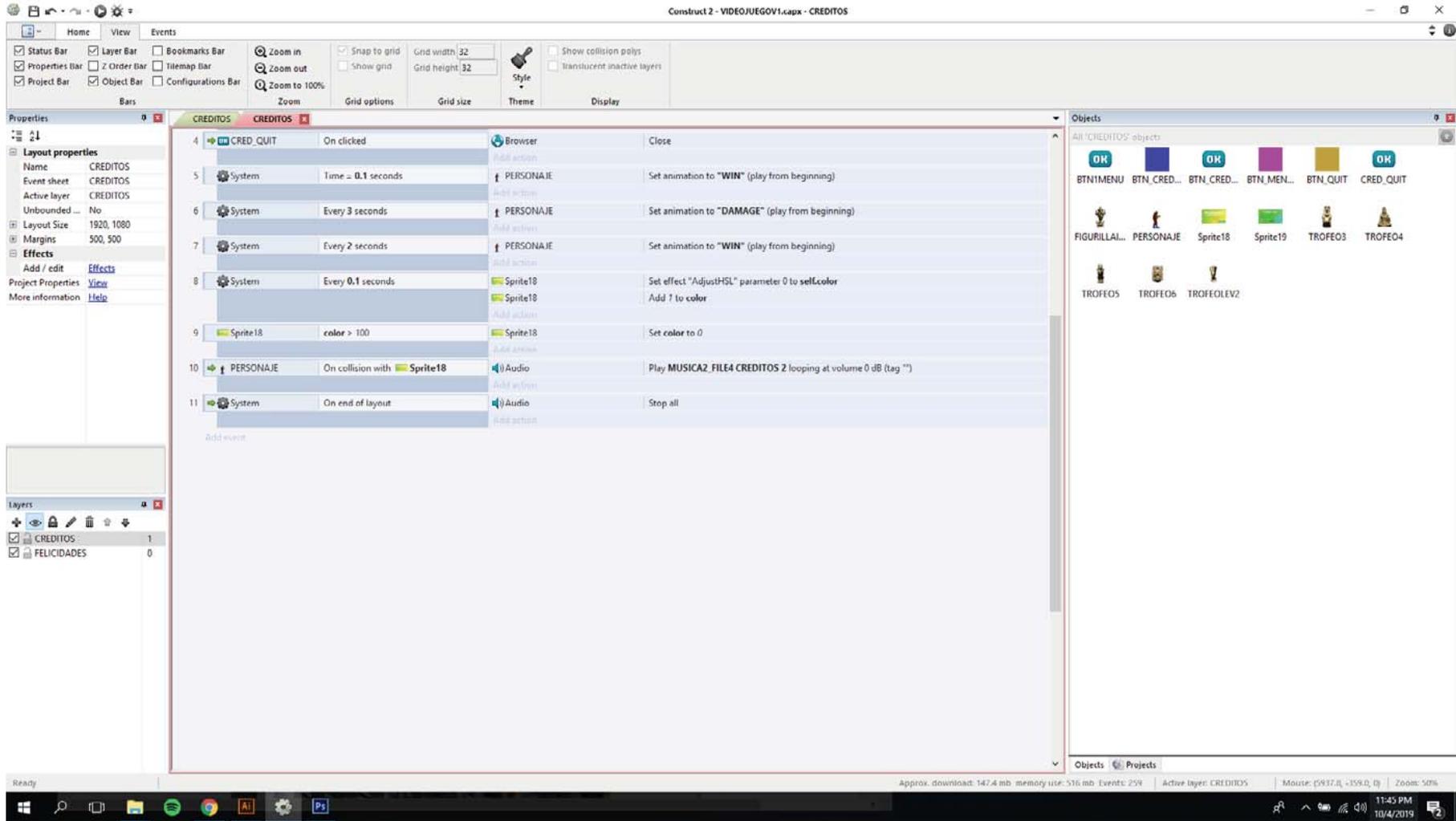
3.5 Nivel 6



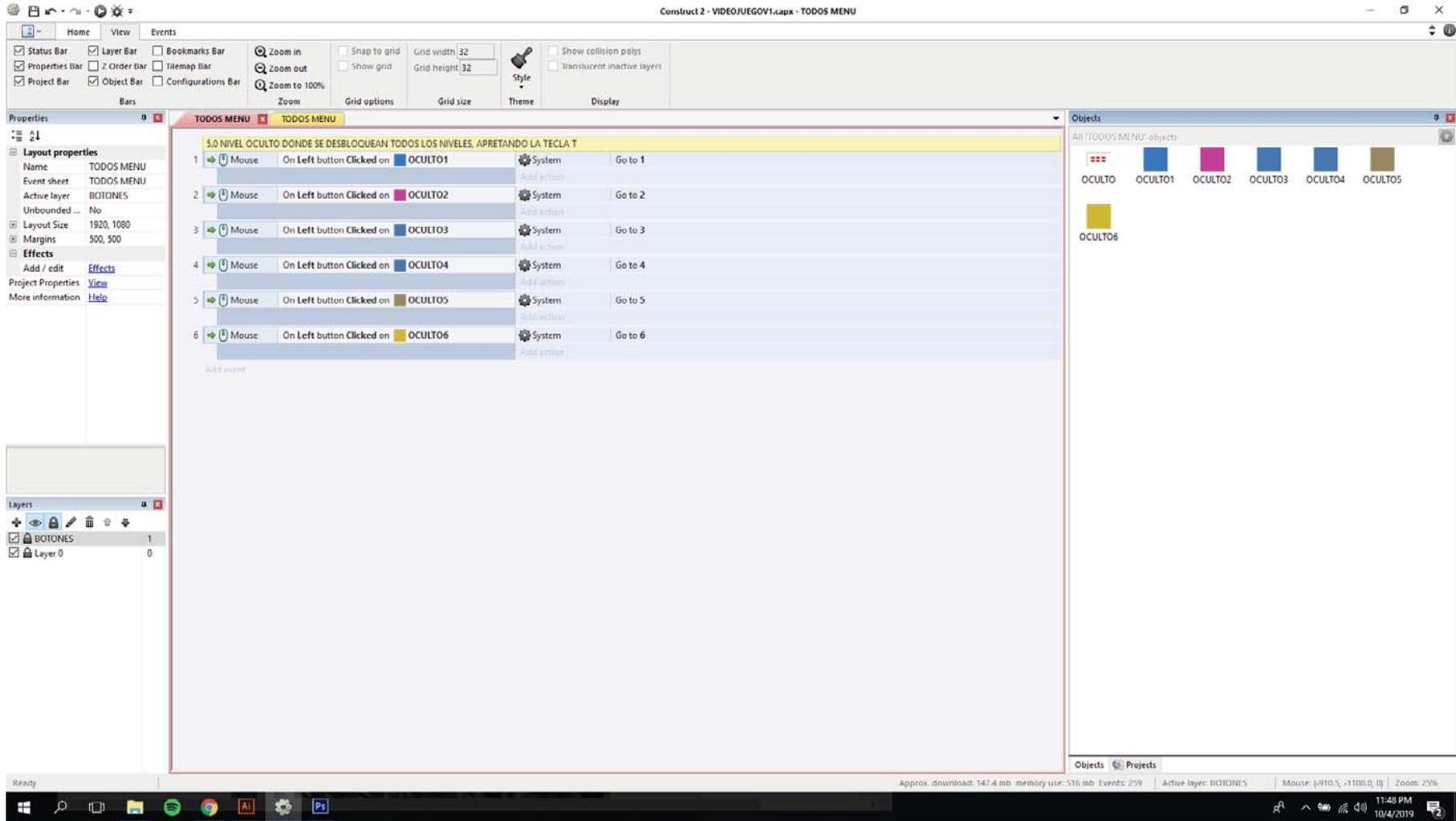


4.0 Fin del juego - Créditos

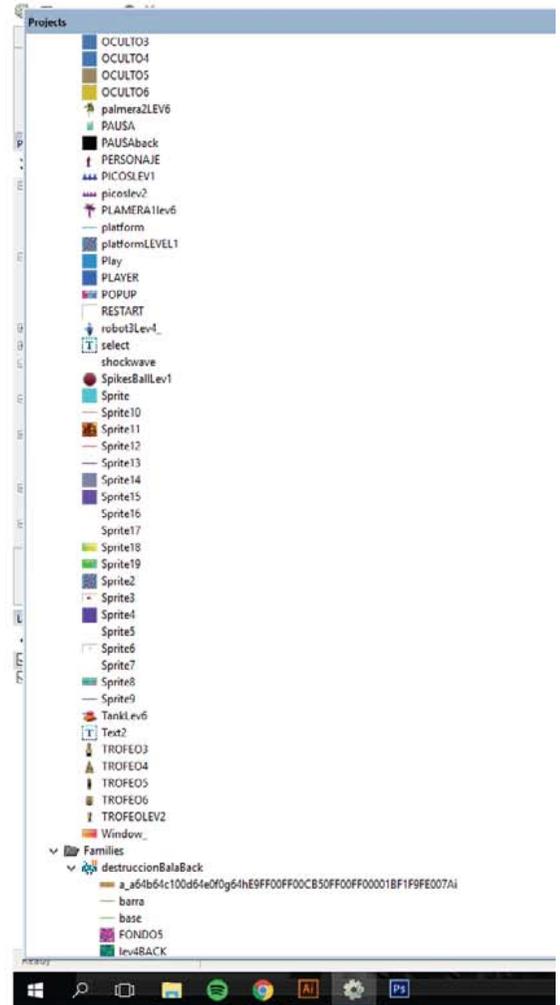
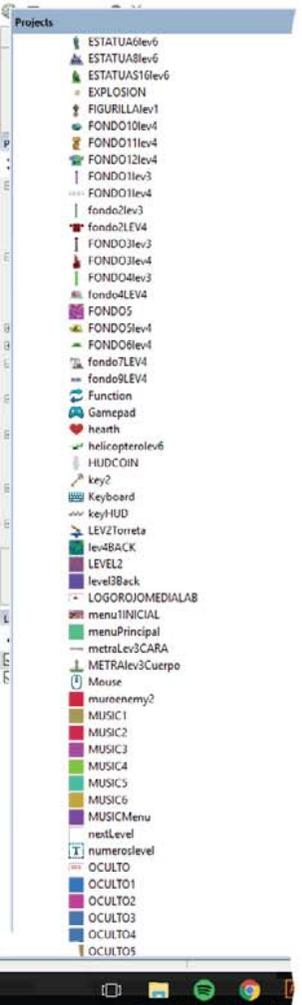
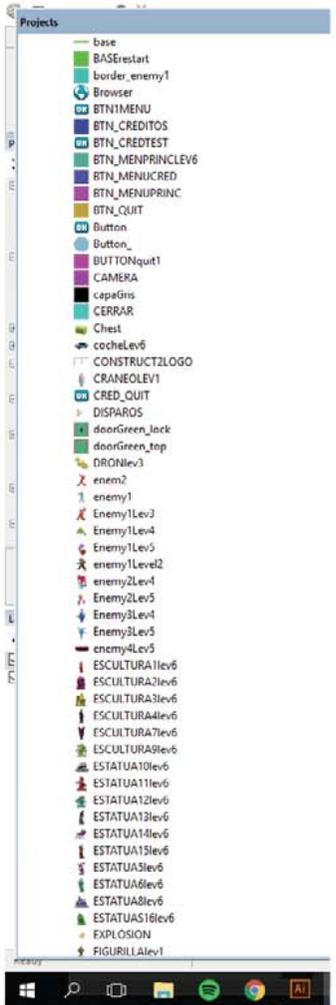
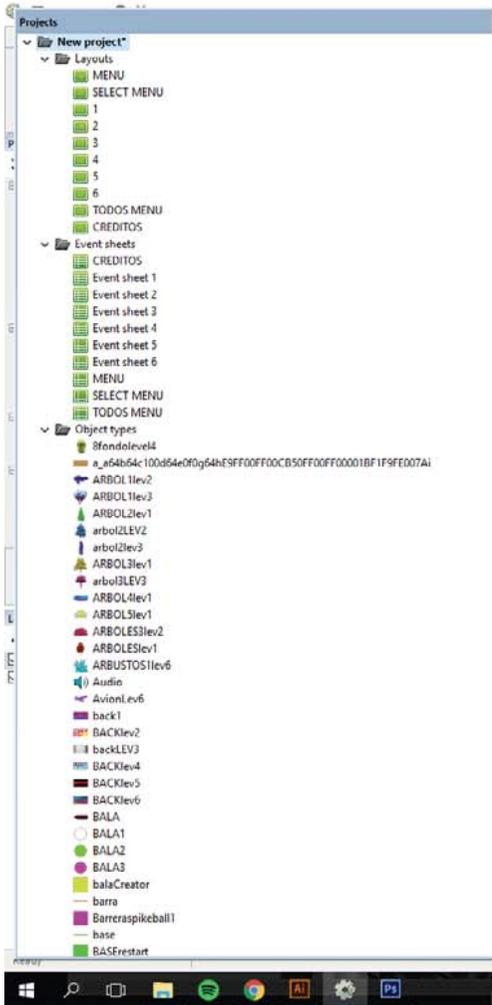


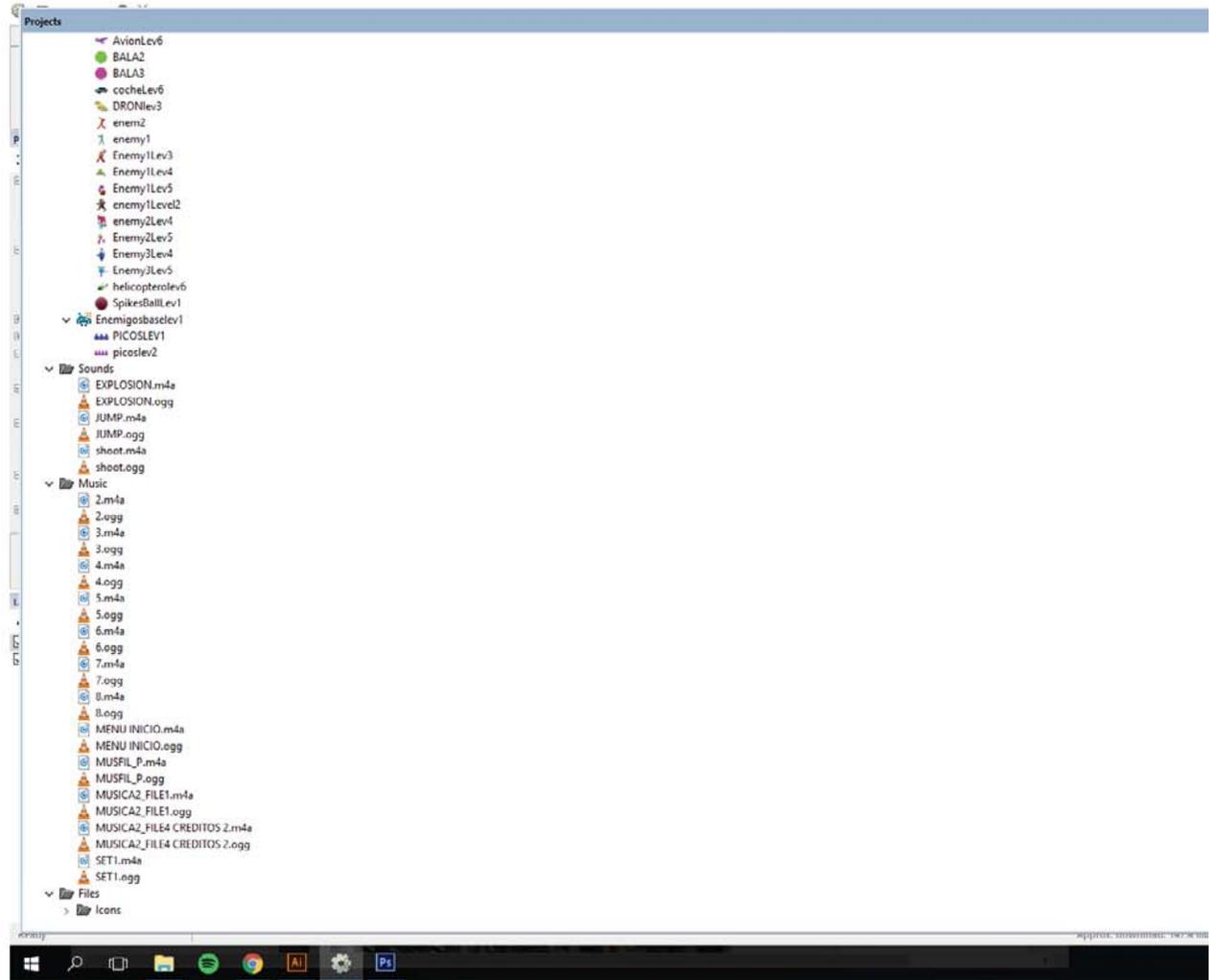
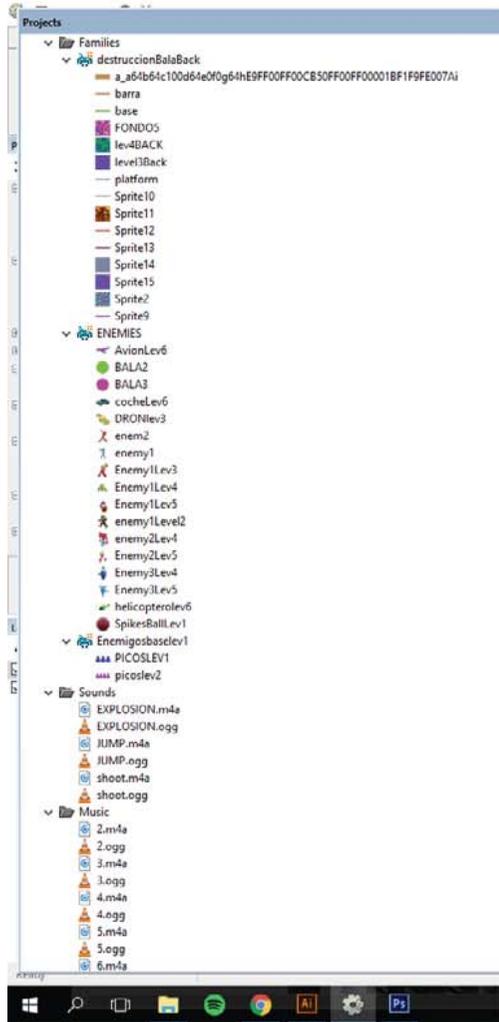


5.0 Todos los Niveles

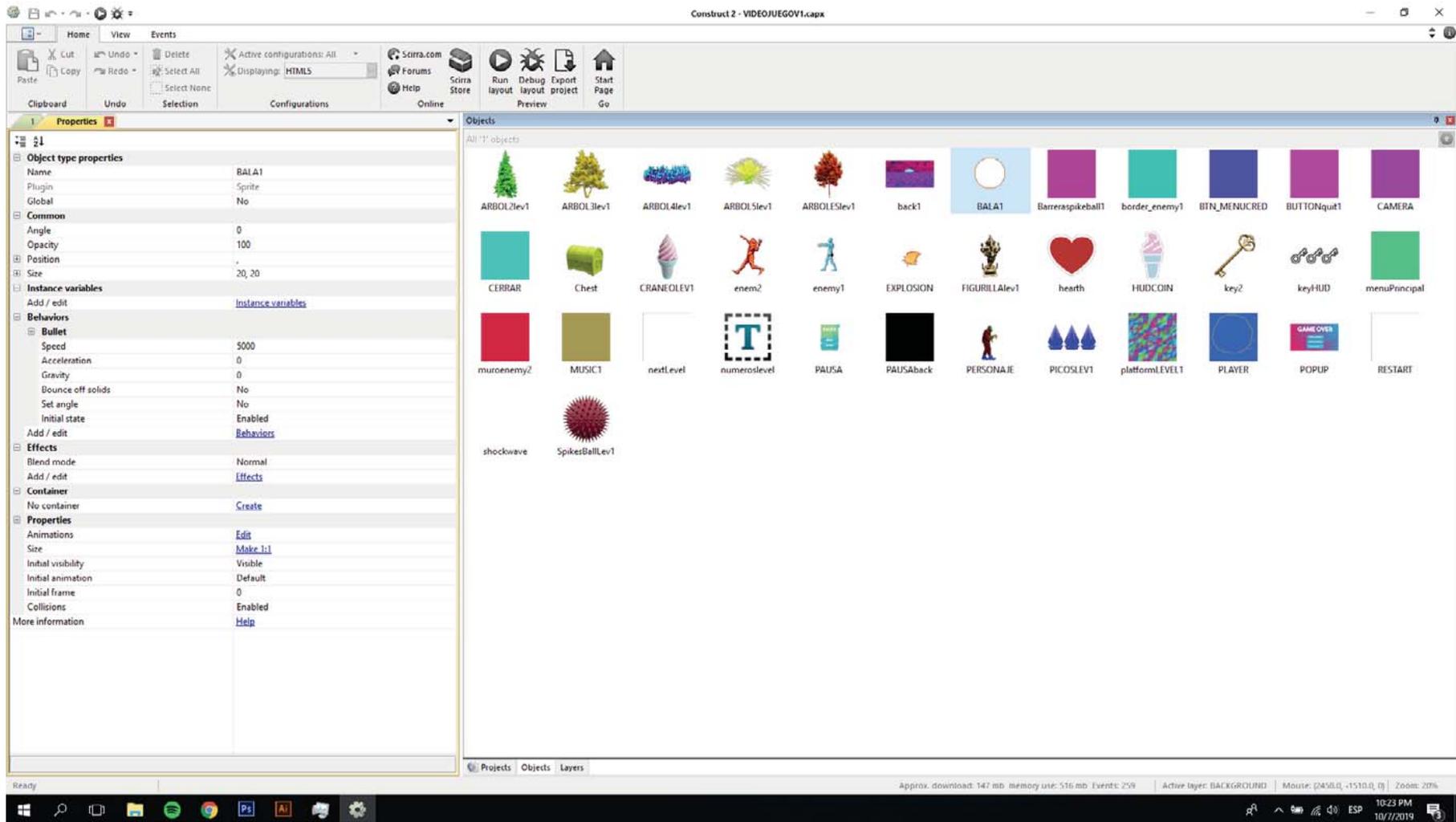


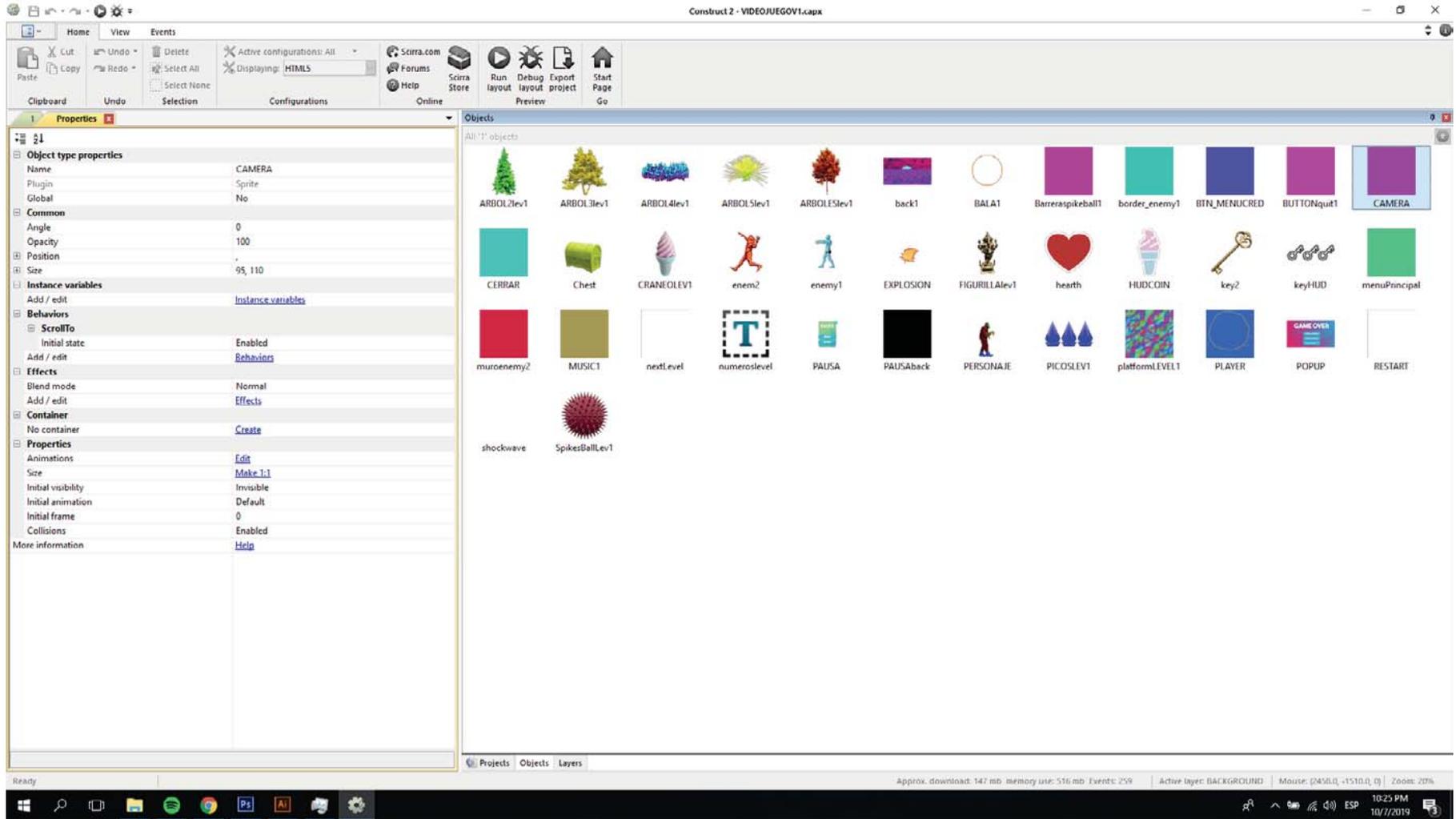
Assets del juego

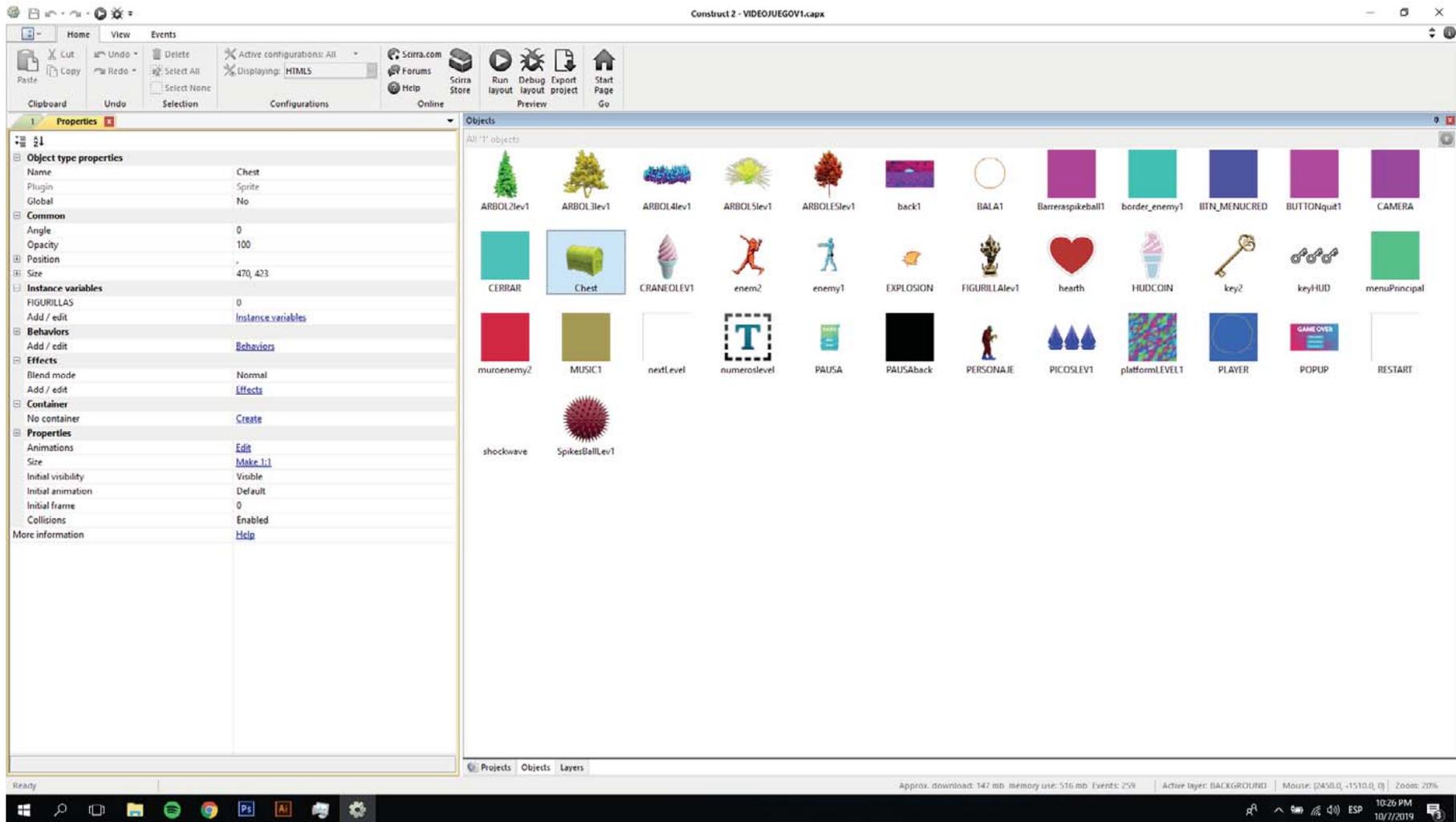


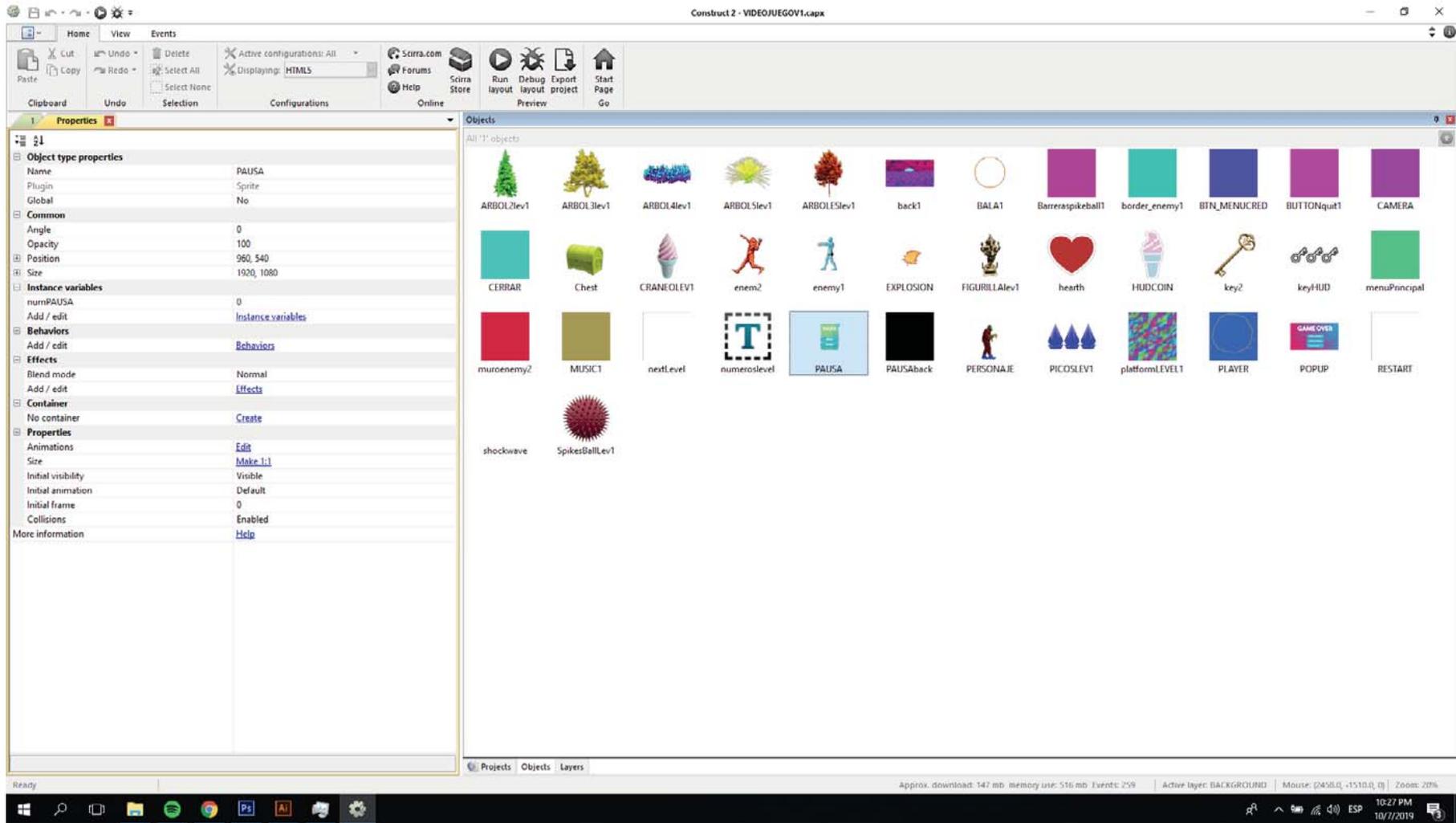


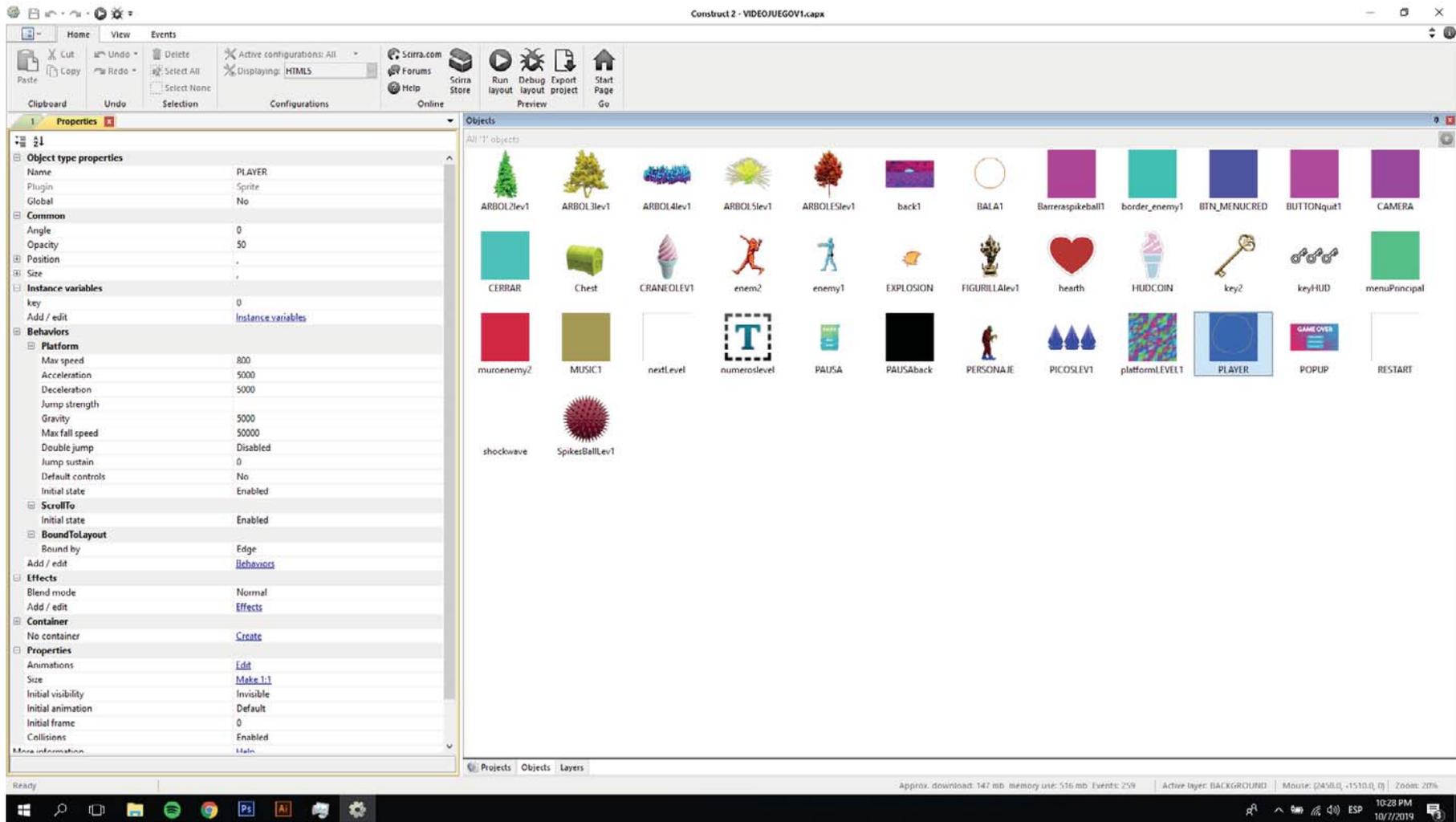
Propiedades de los Objetos Generales



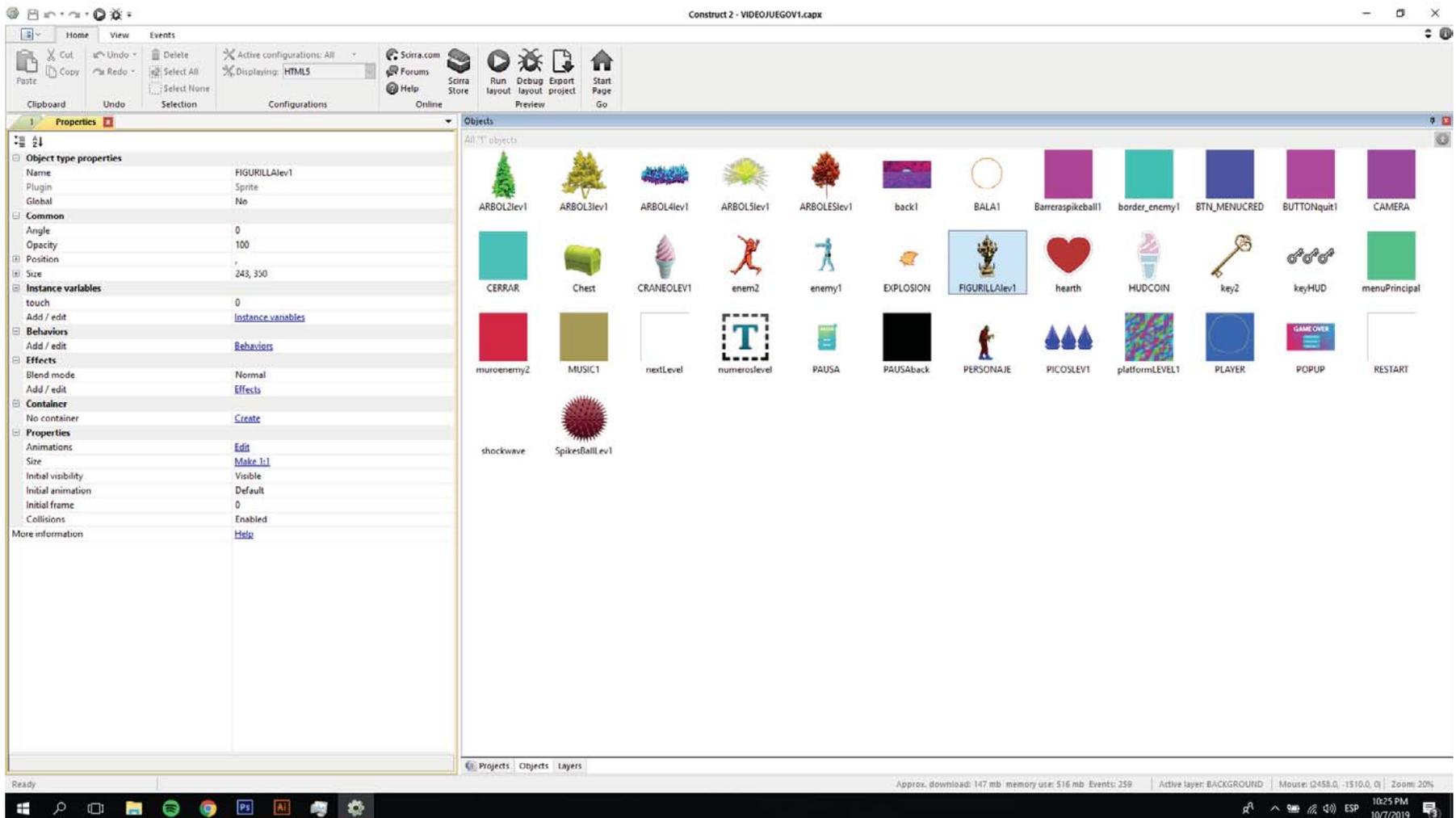


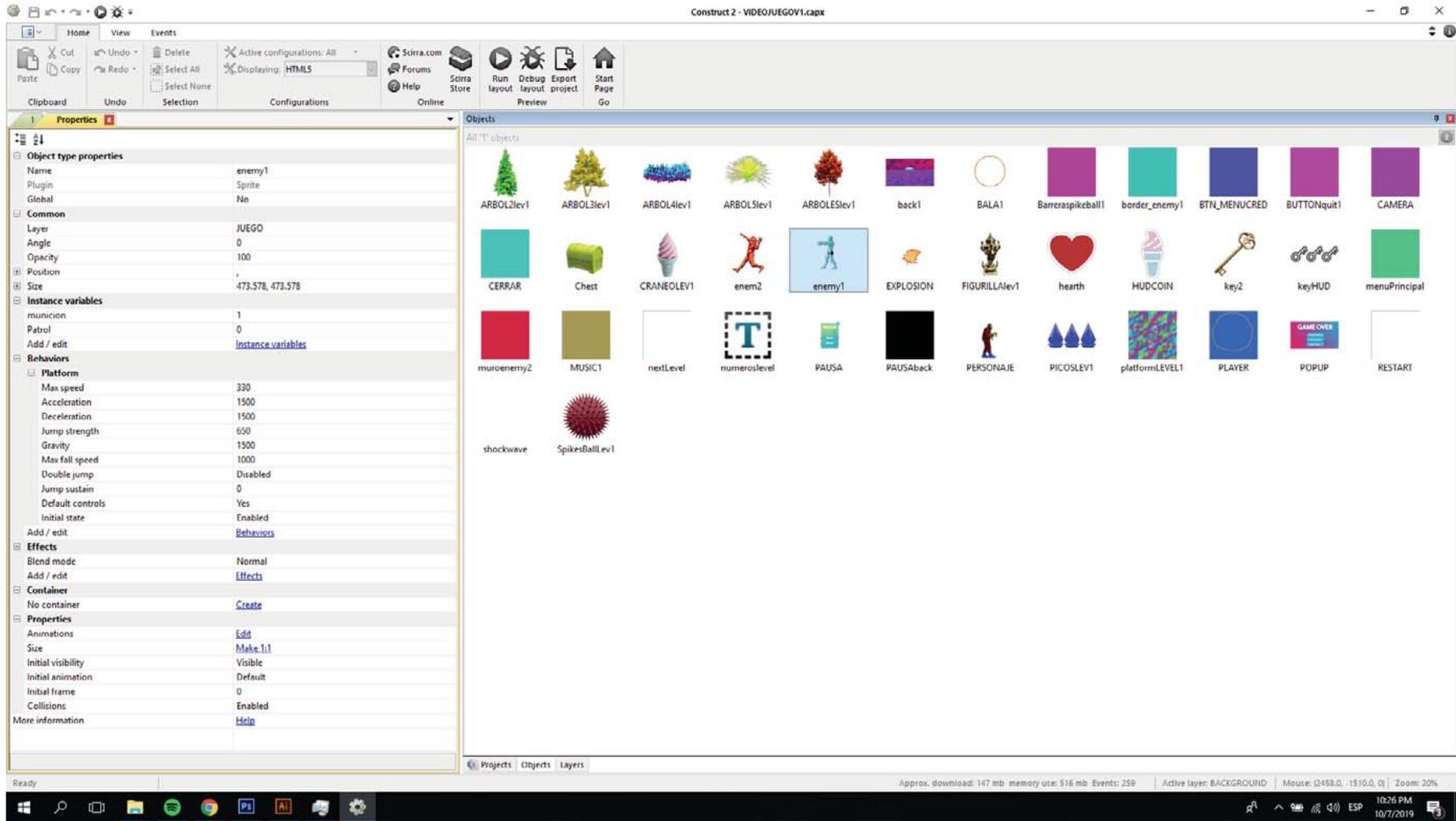


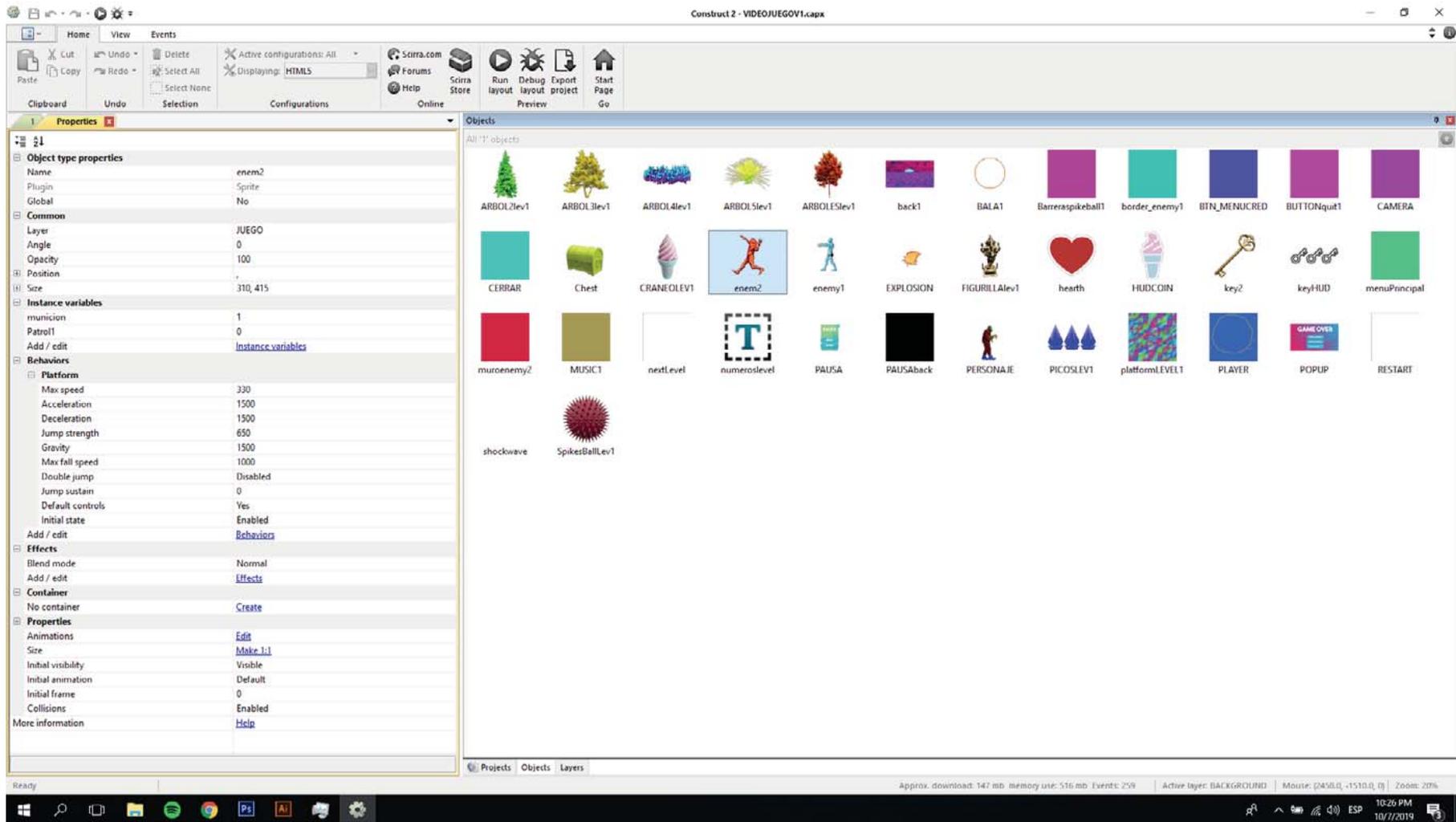


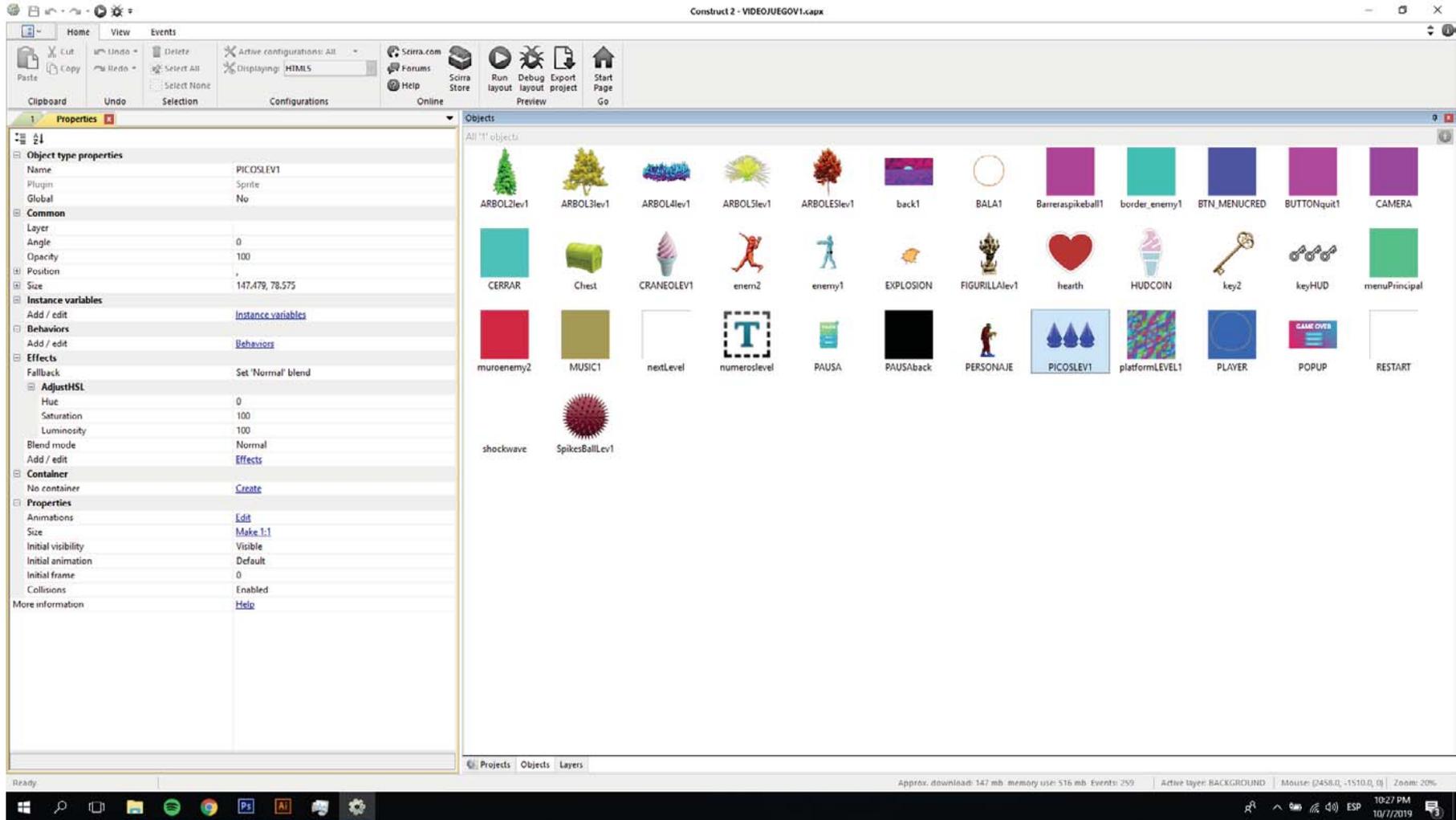


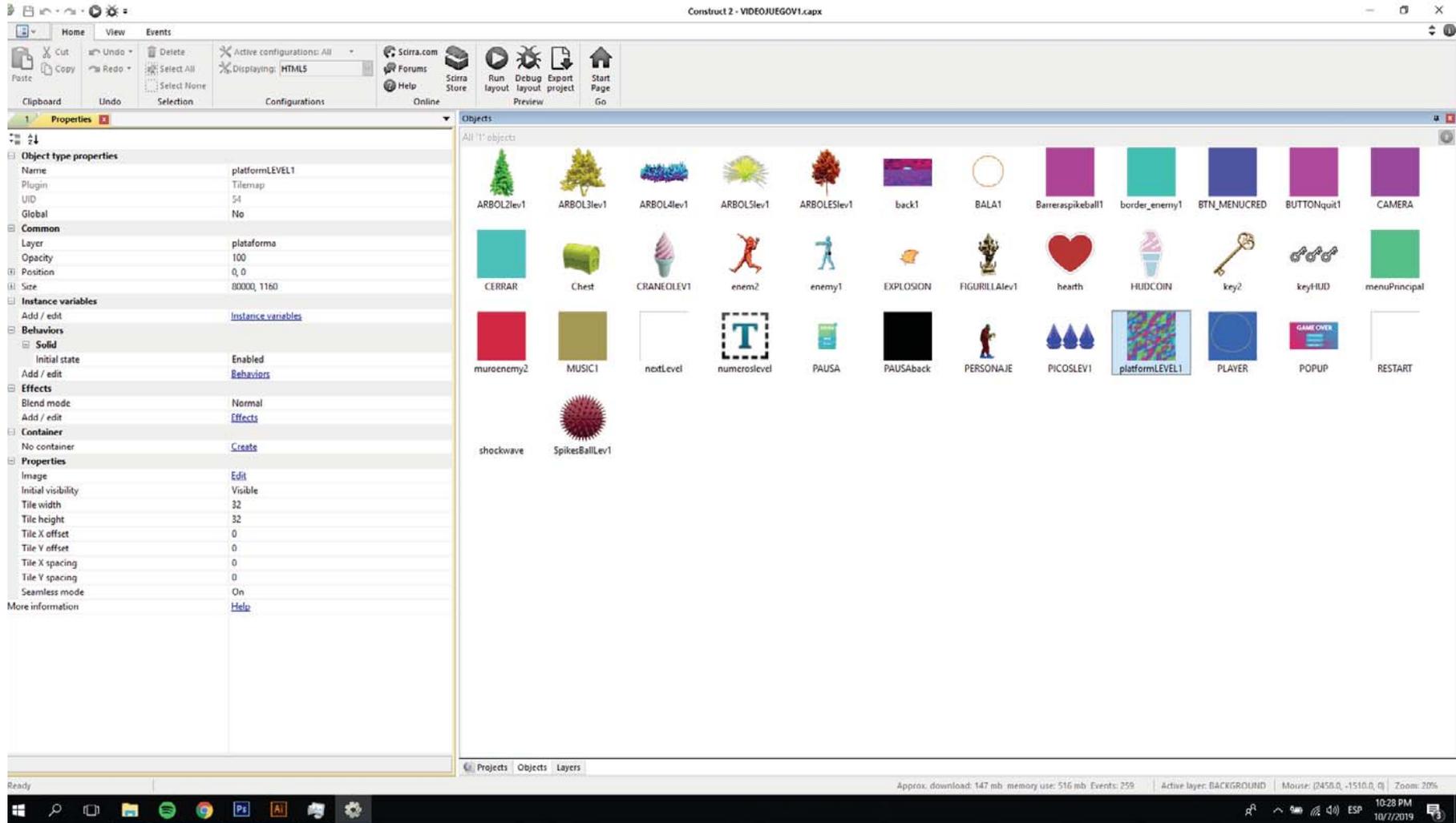
Propiedades de los Objetos Nivel 1

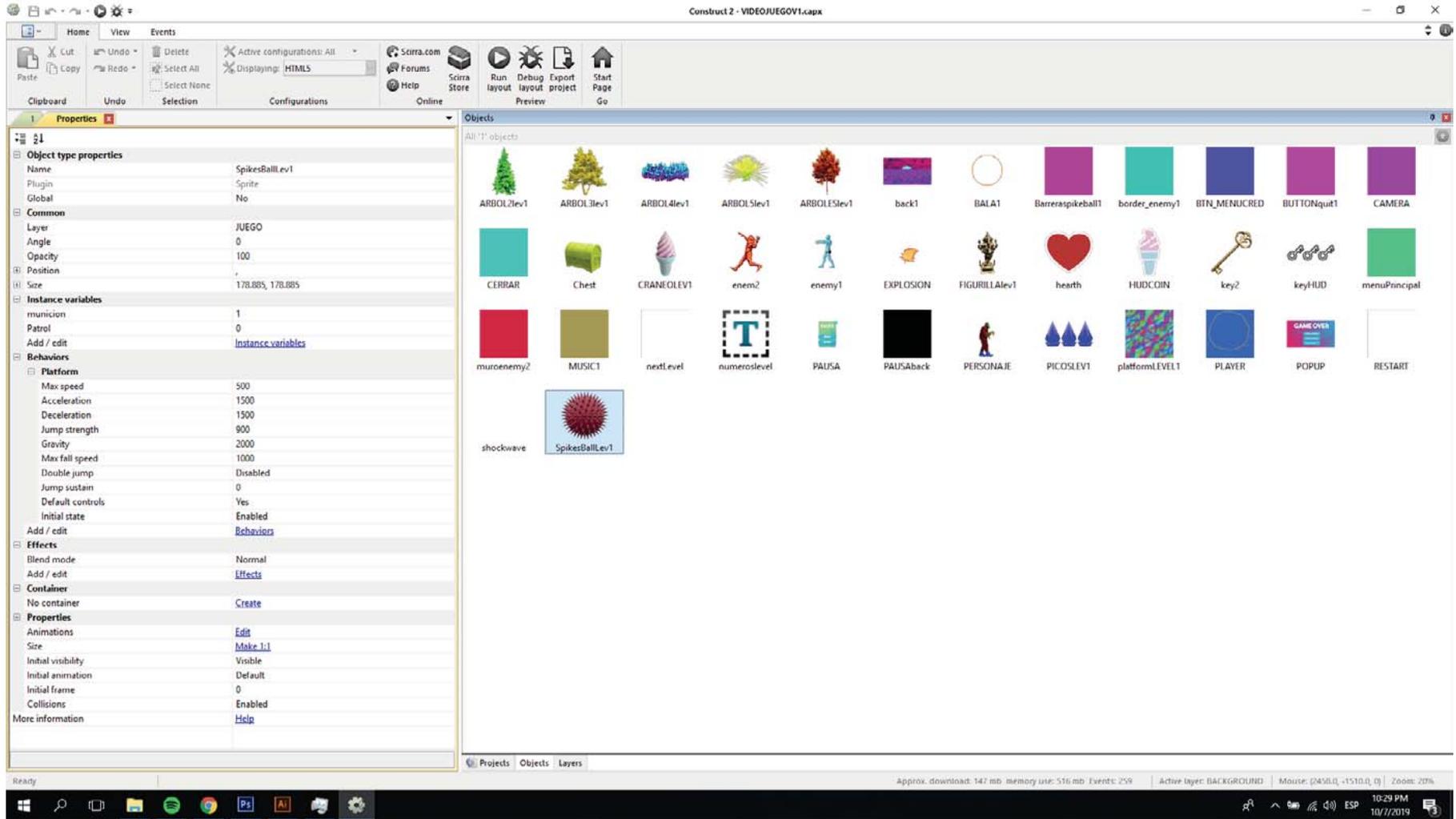




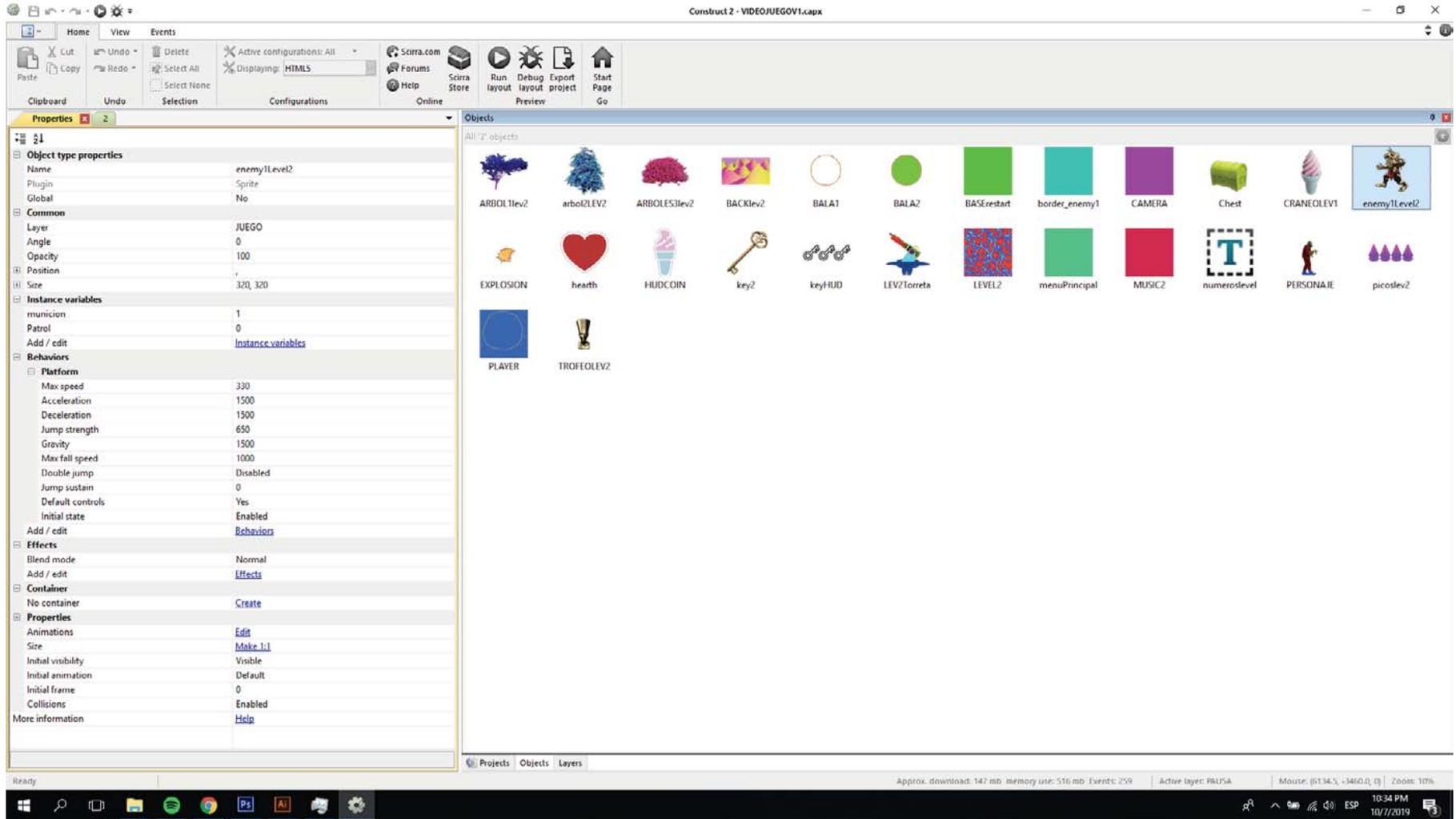


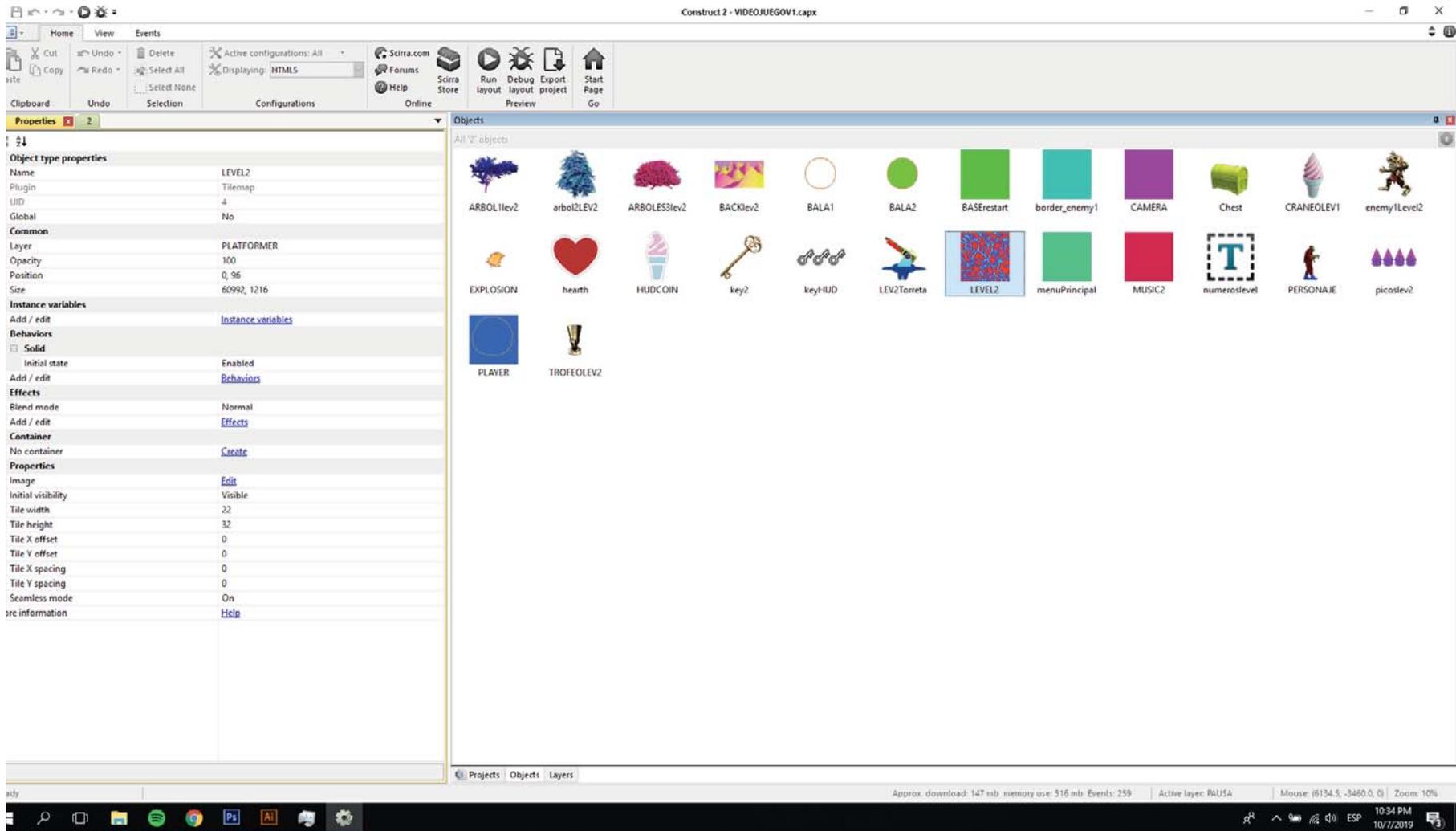


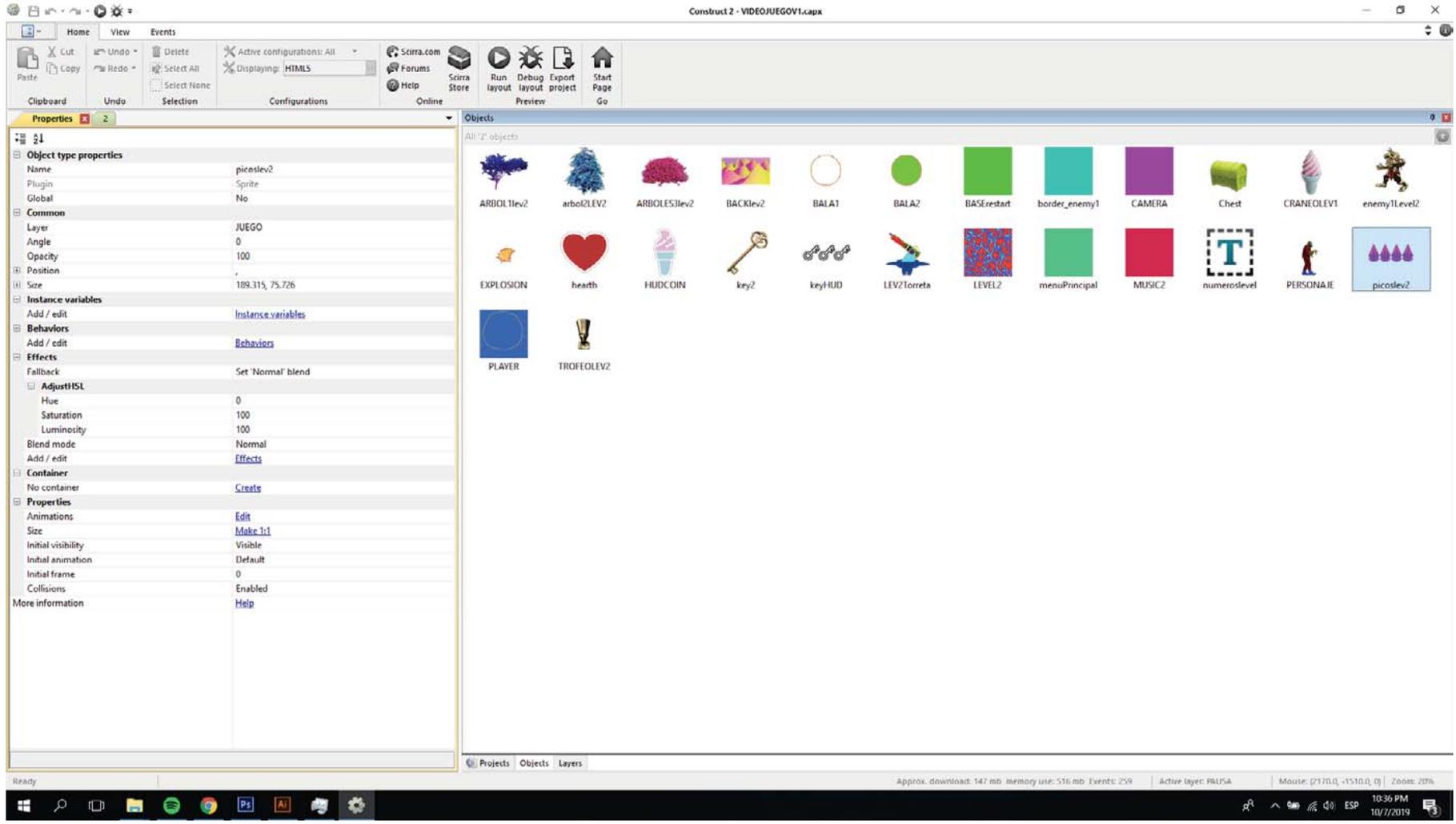




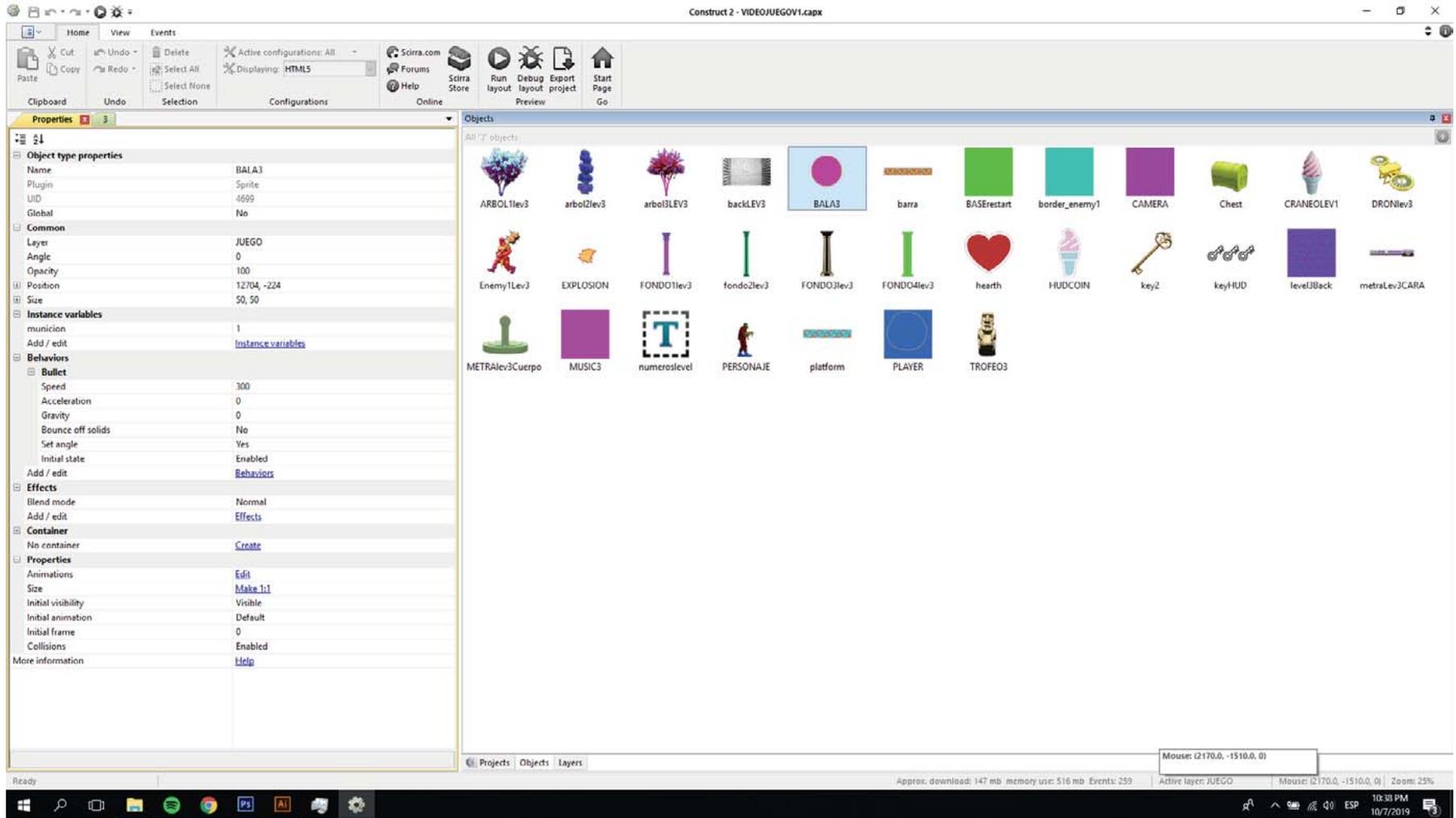
Propiedades de los Objetos Nivel 2

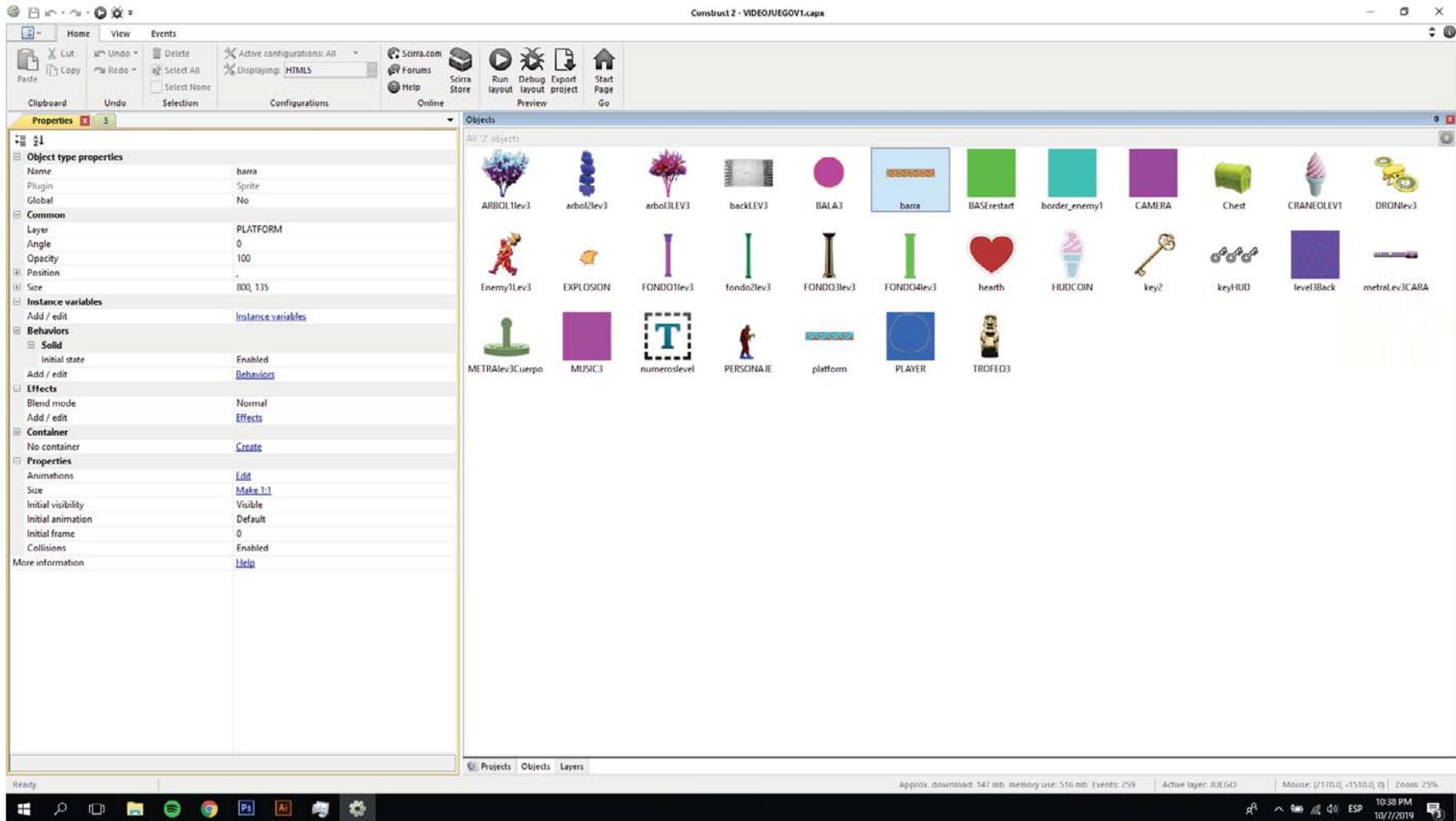


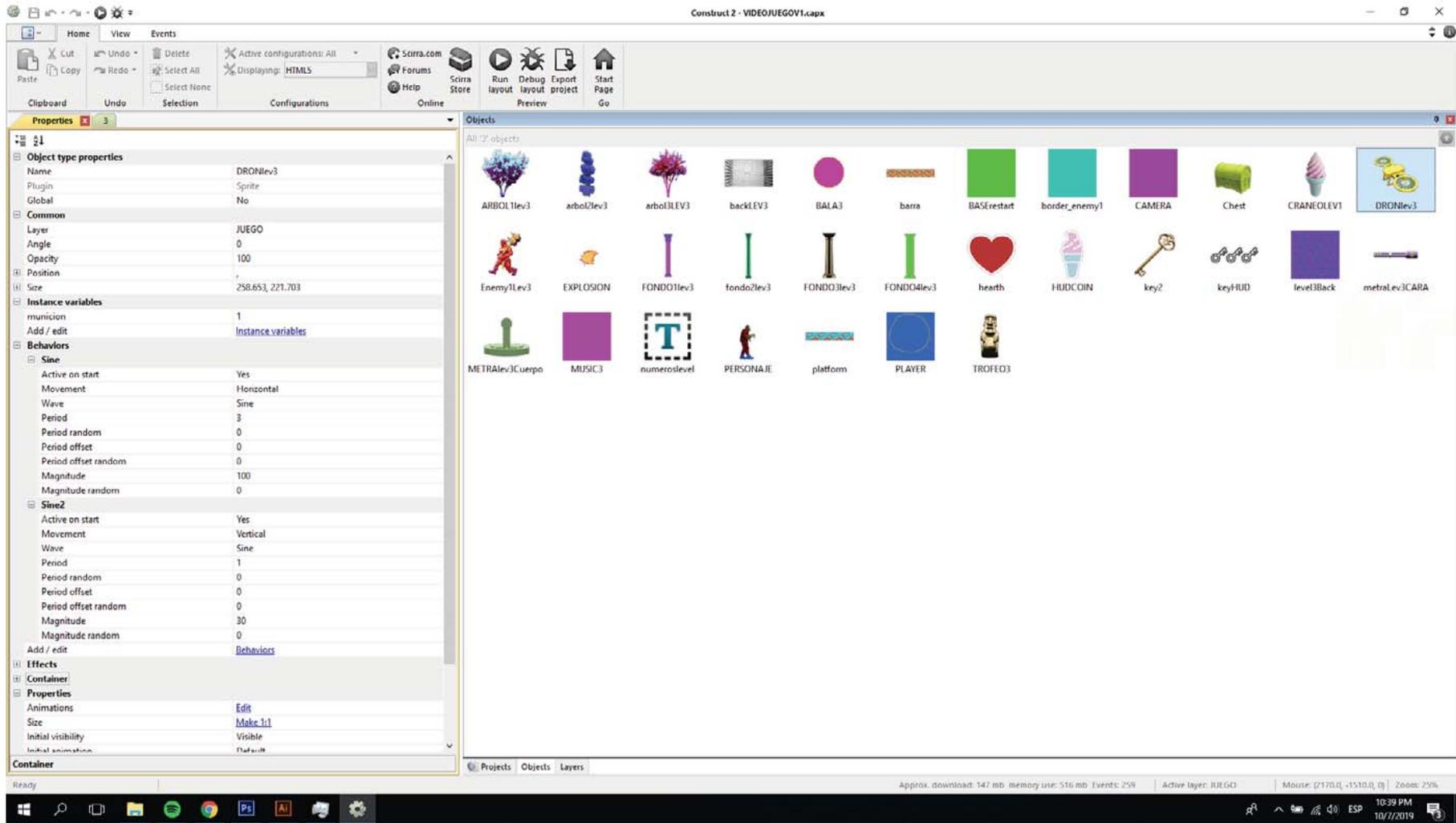


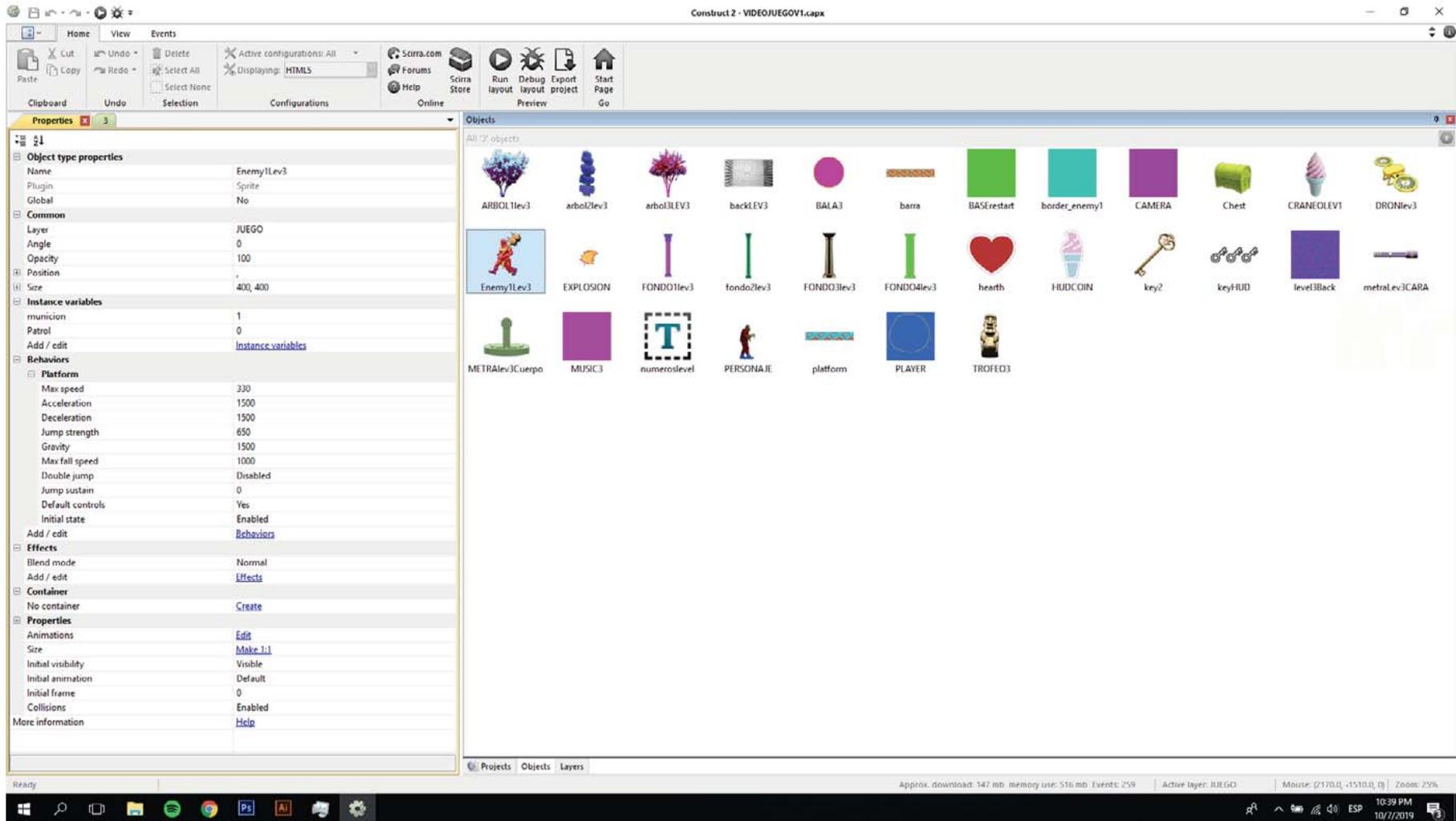


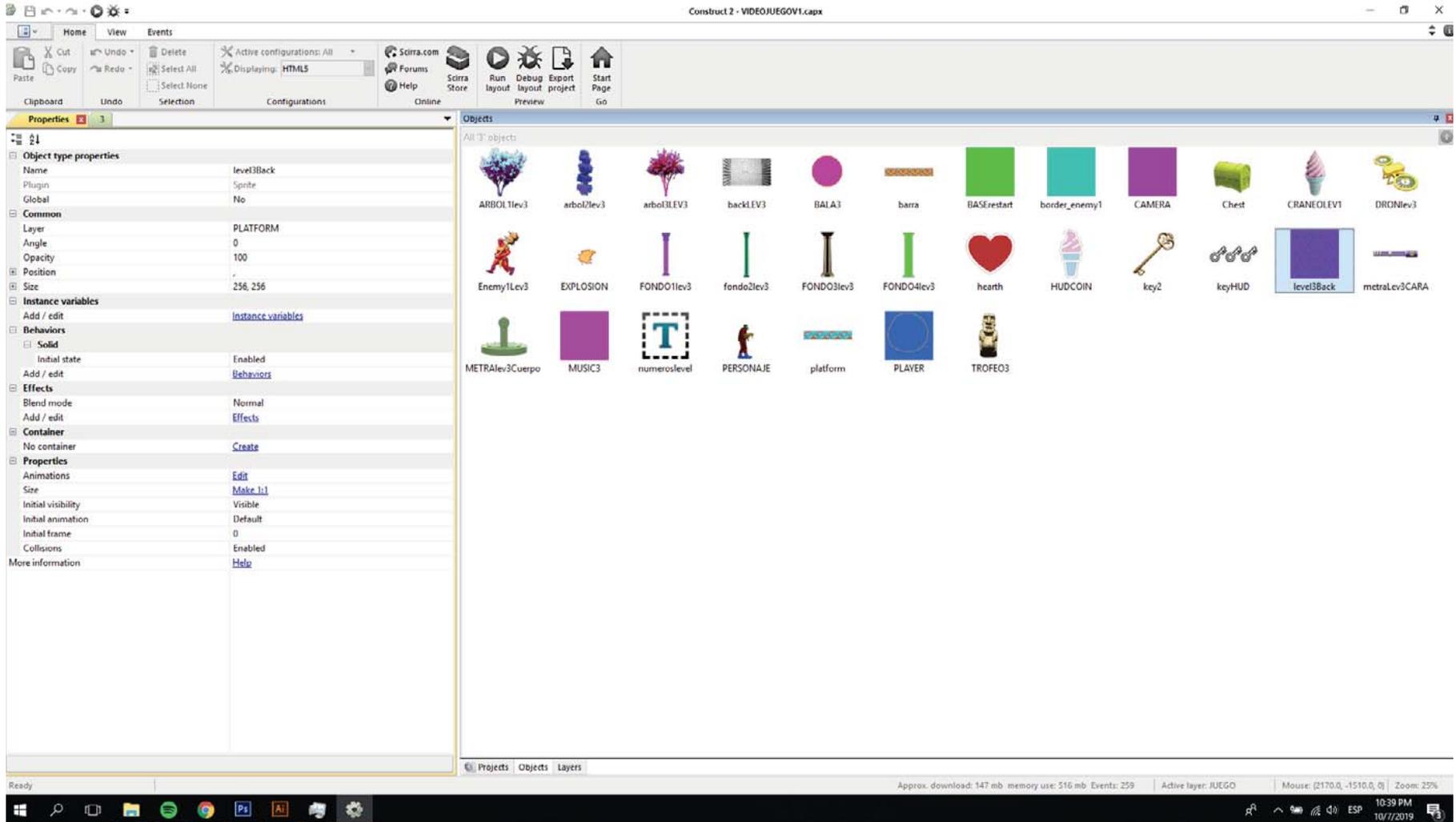
Propiedades de los Objetos Nivel 3

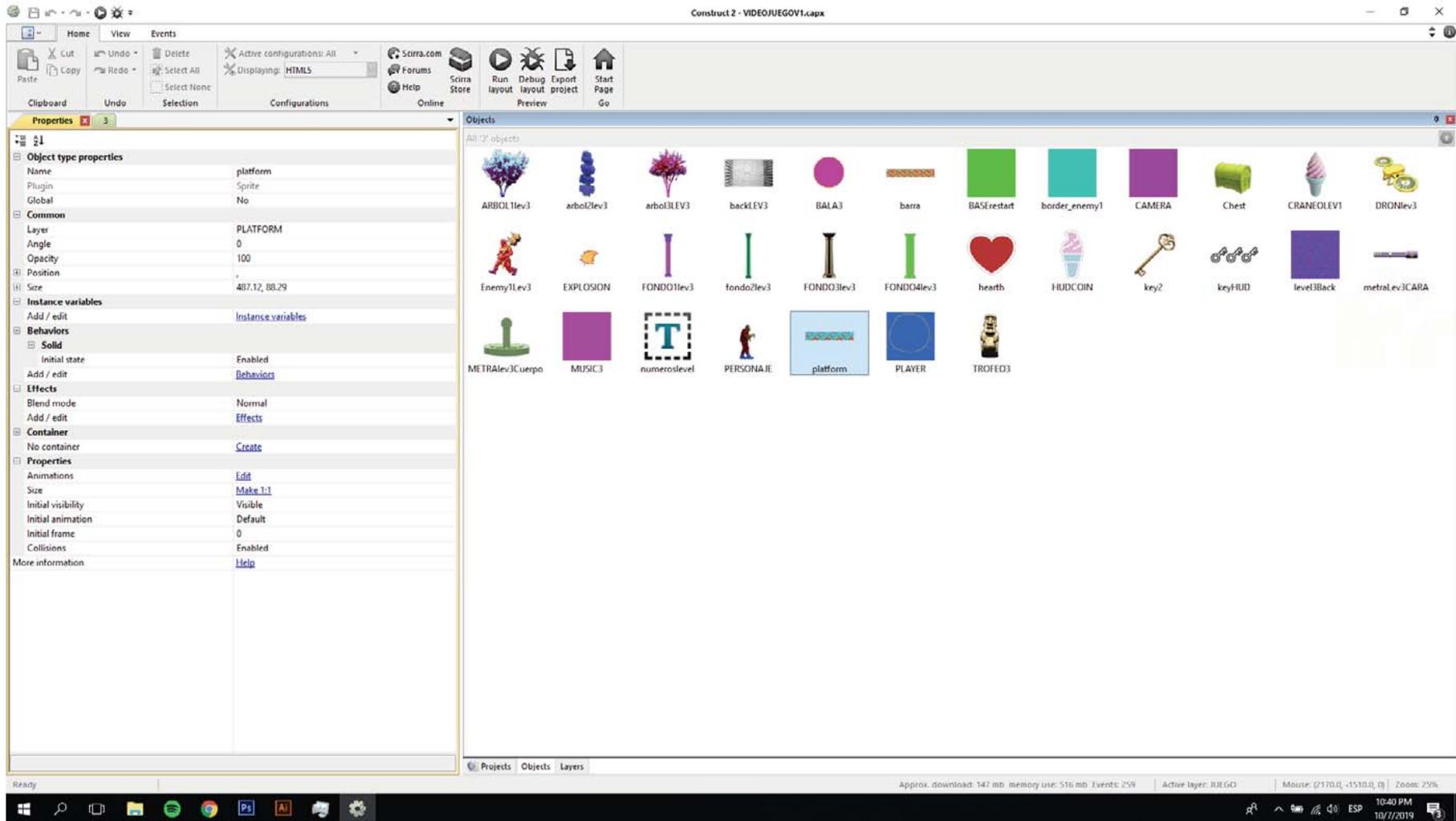




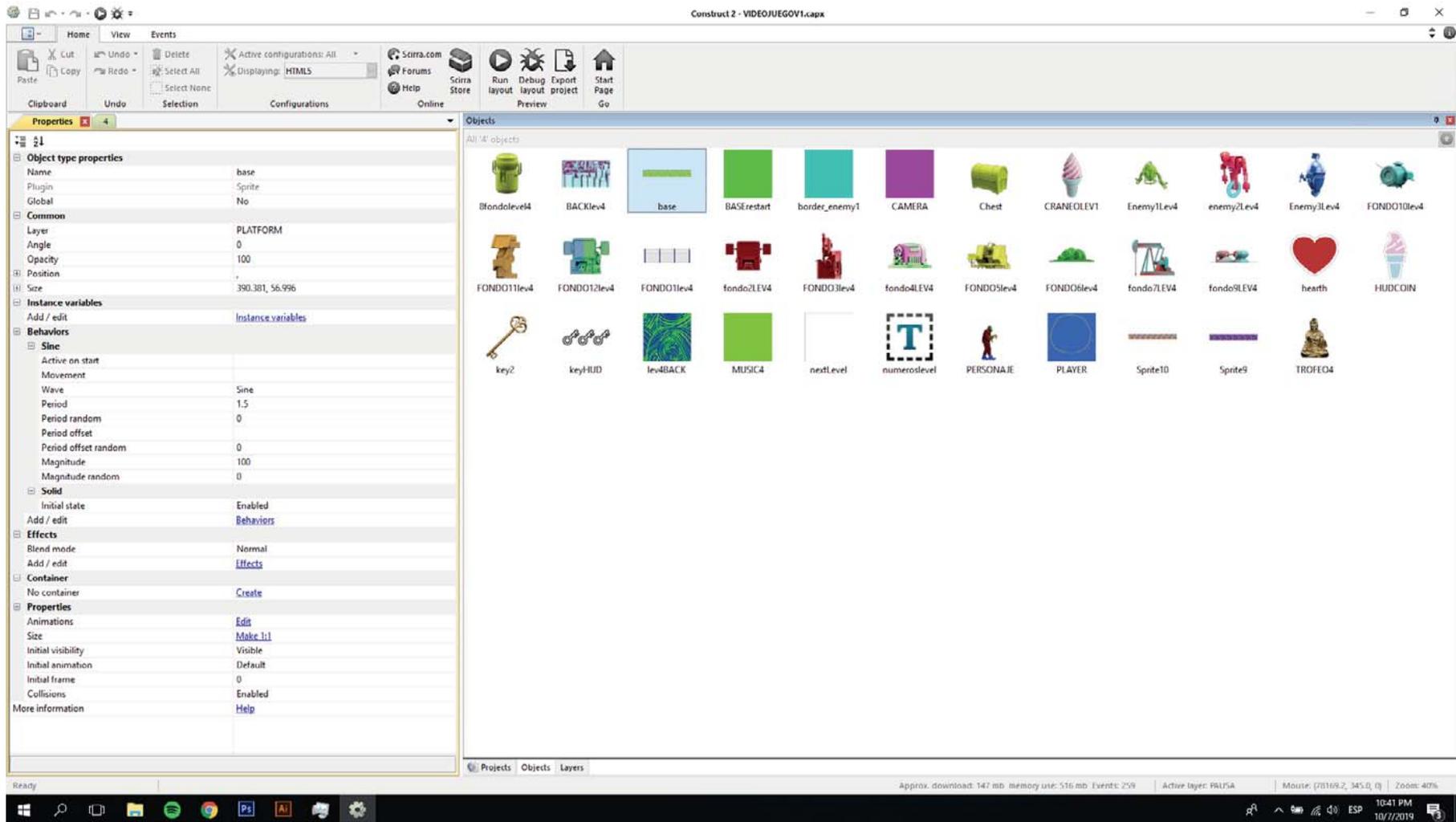


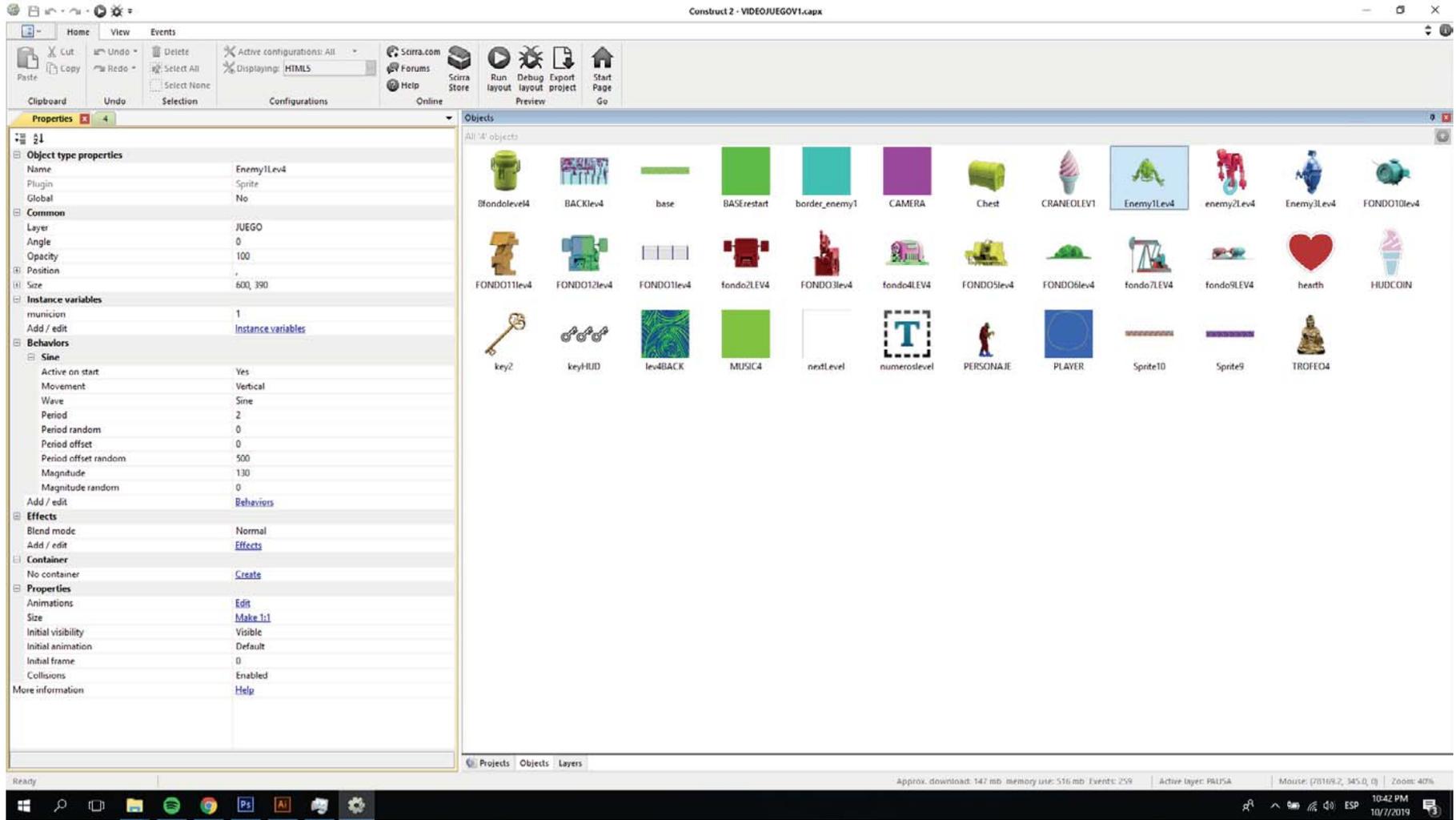


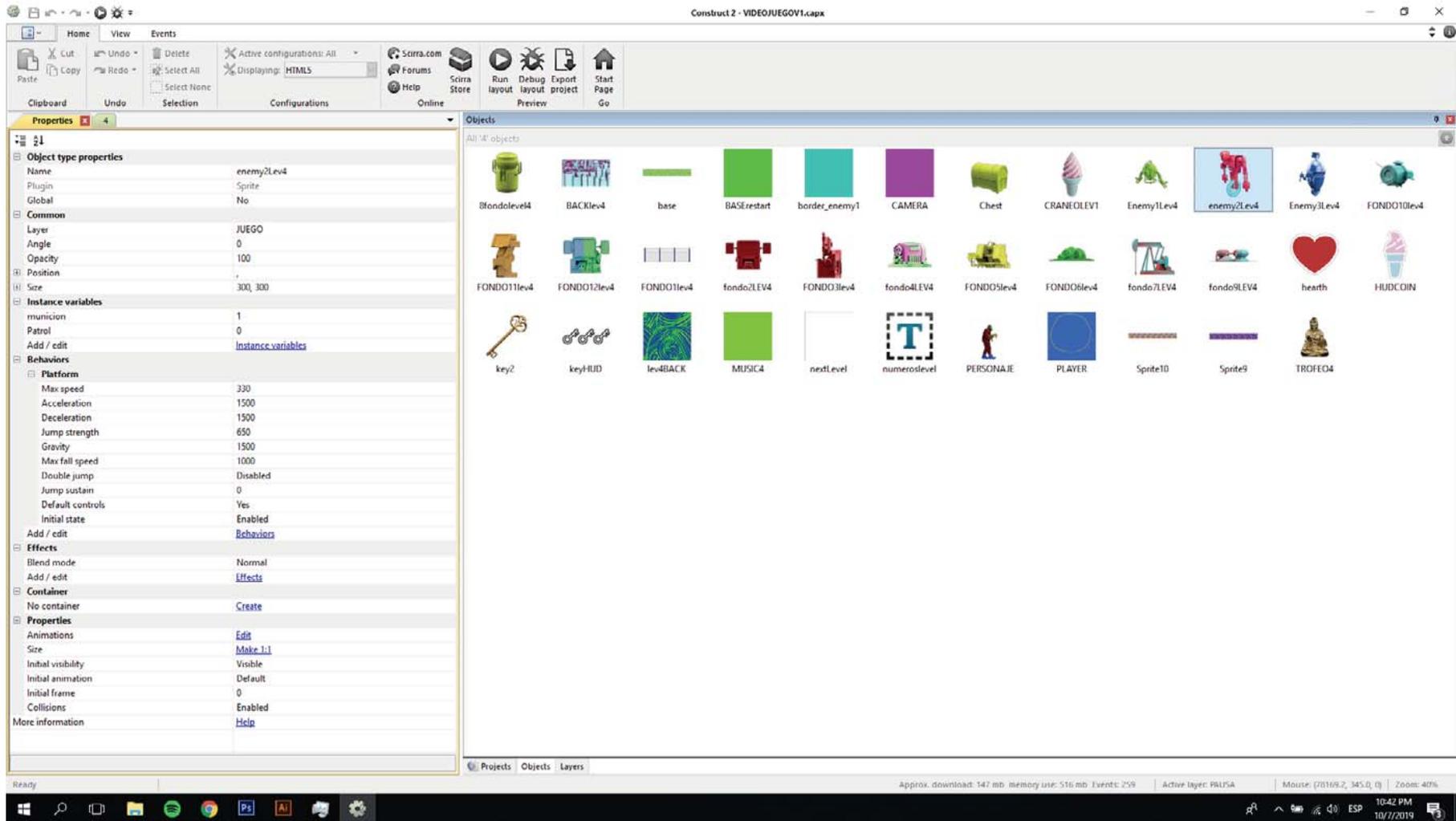


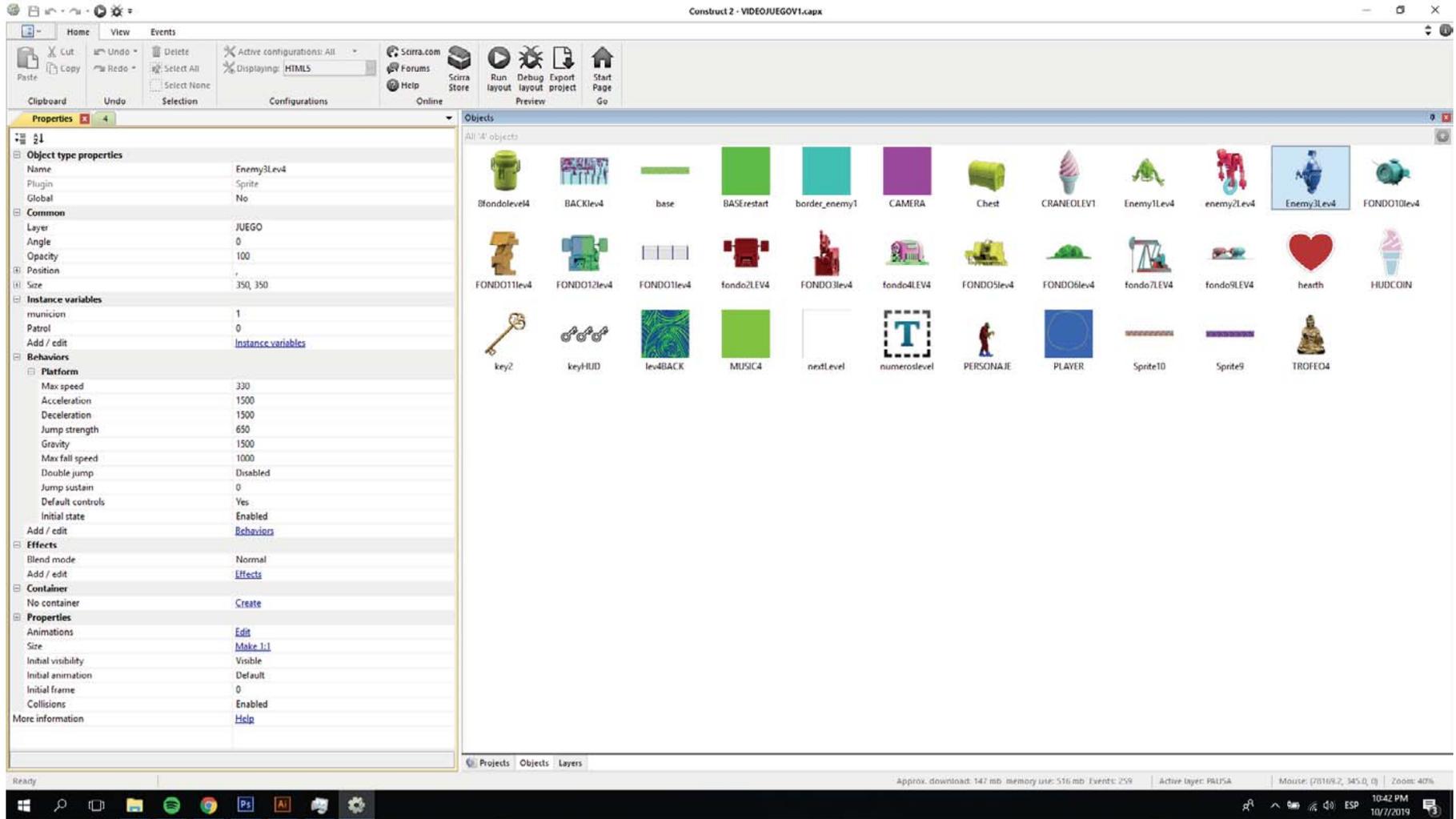


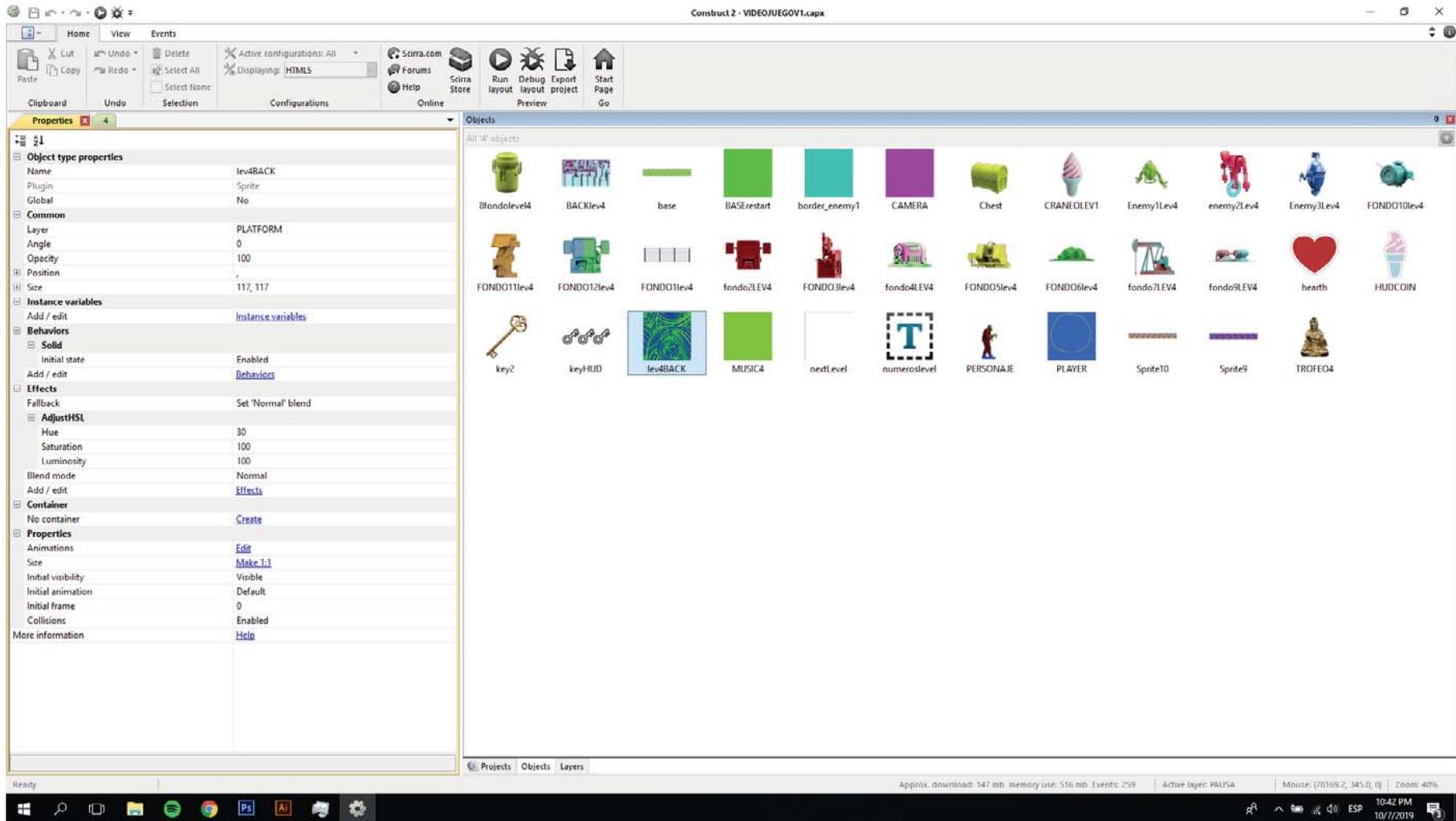
Propiedades de los Objetos Nivel 4

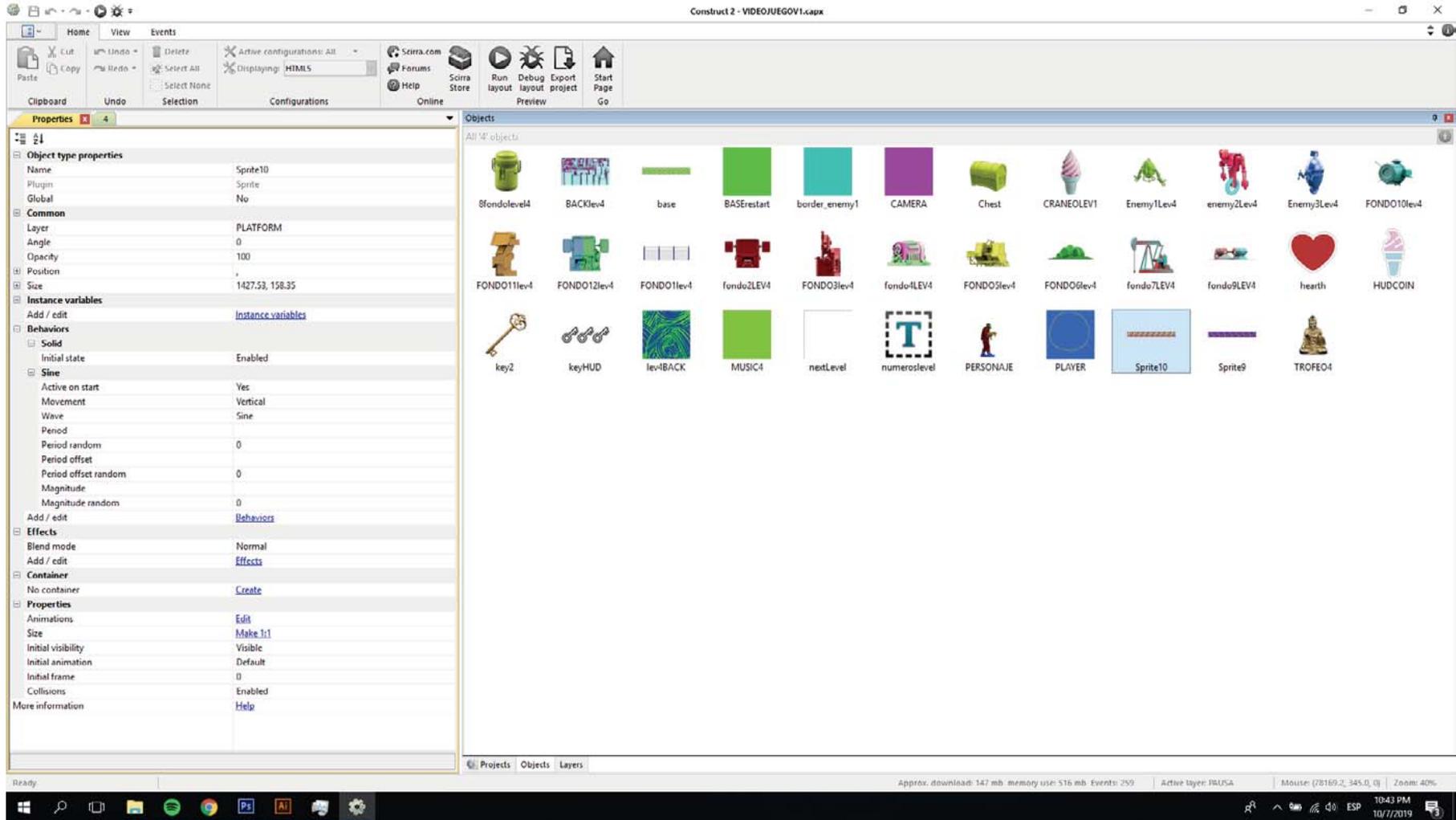


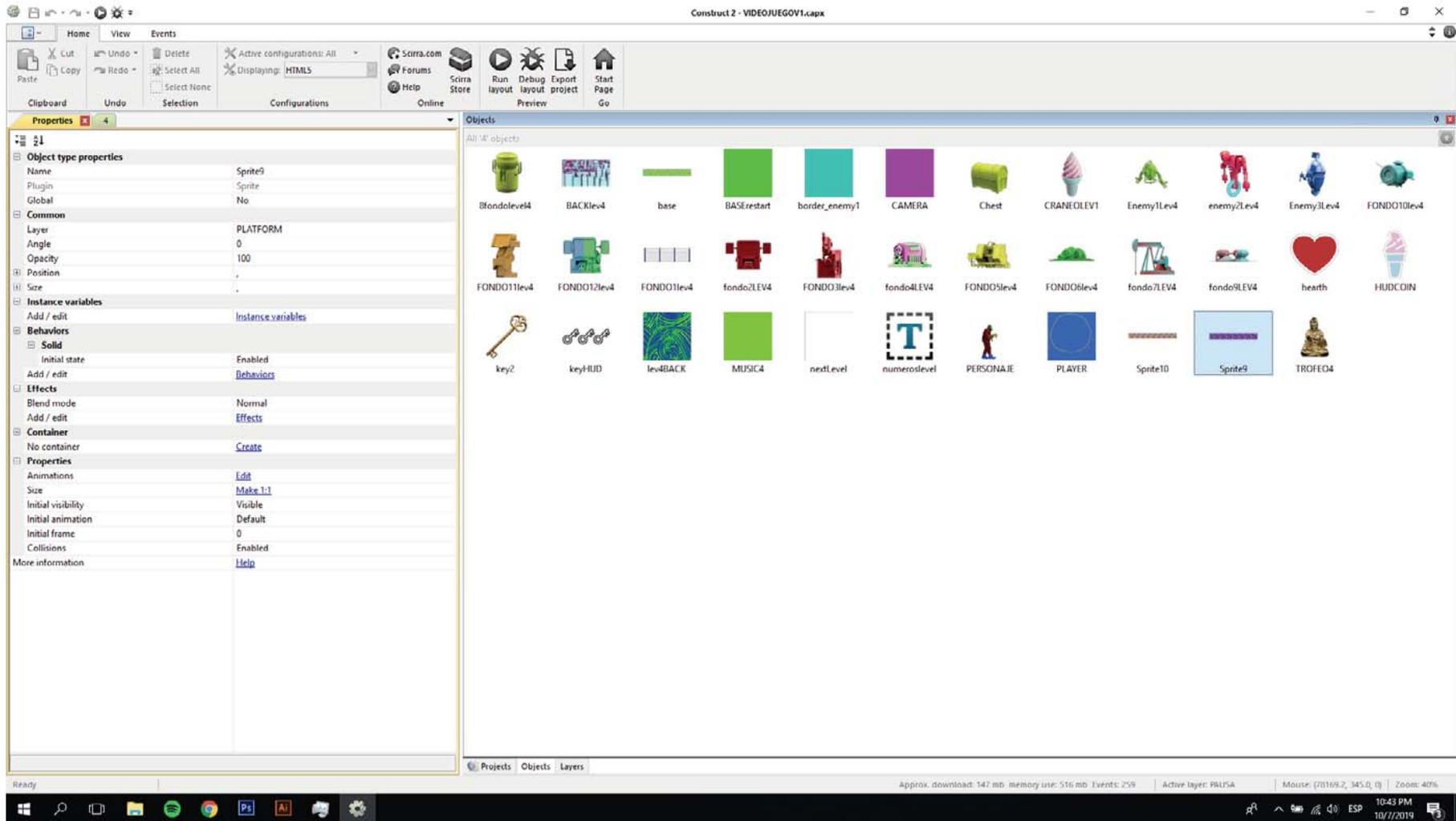




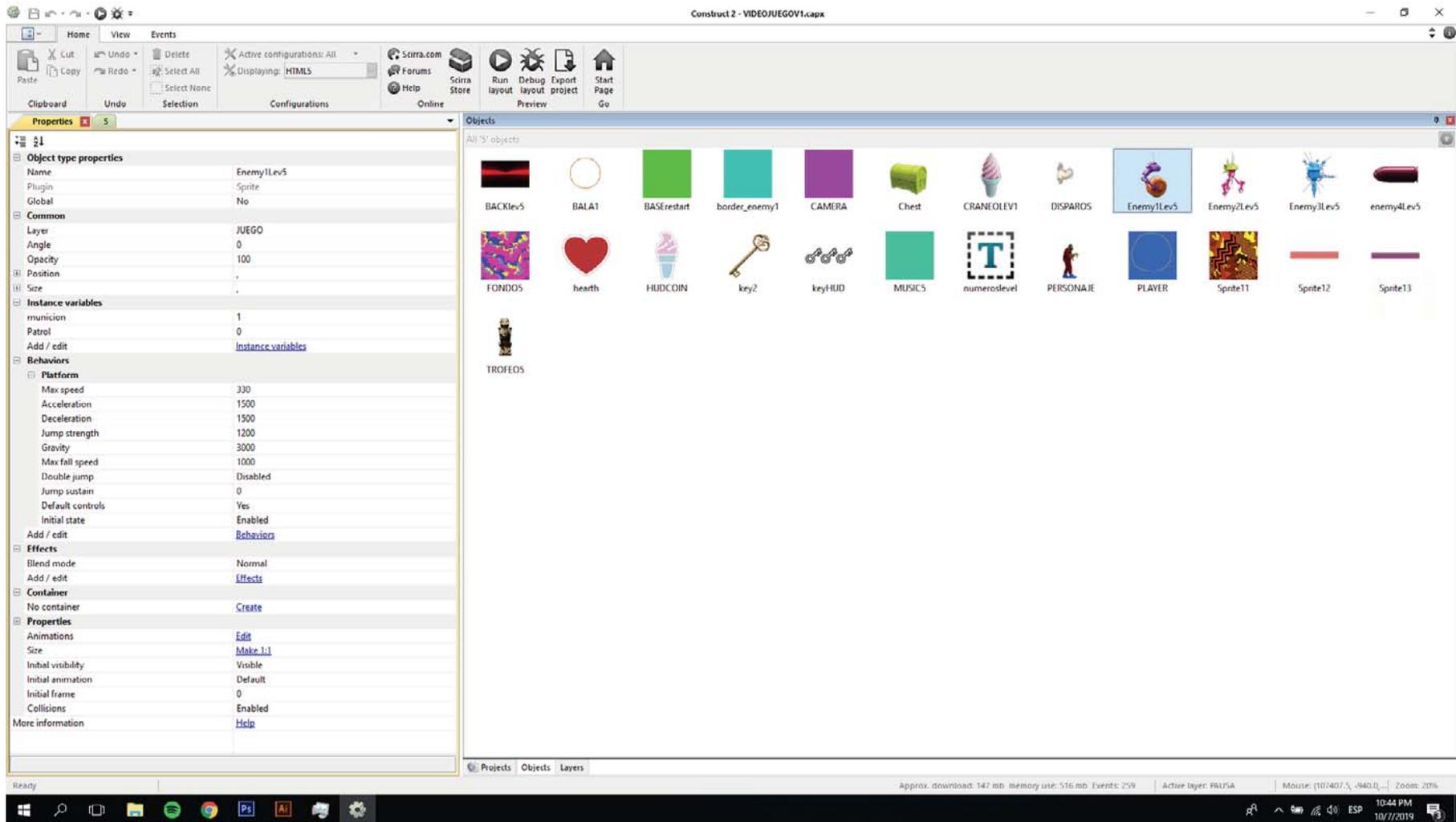


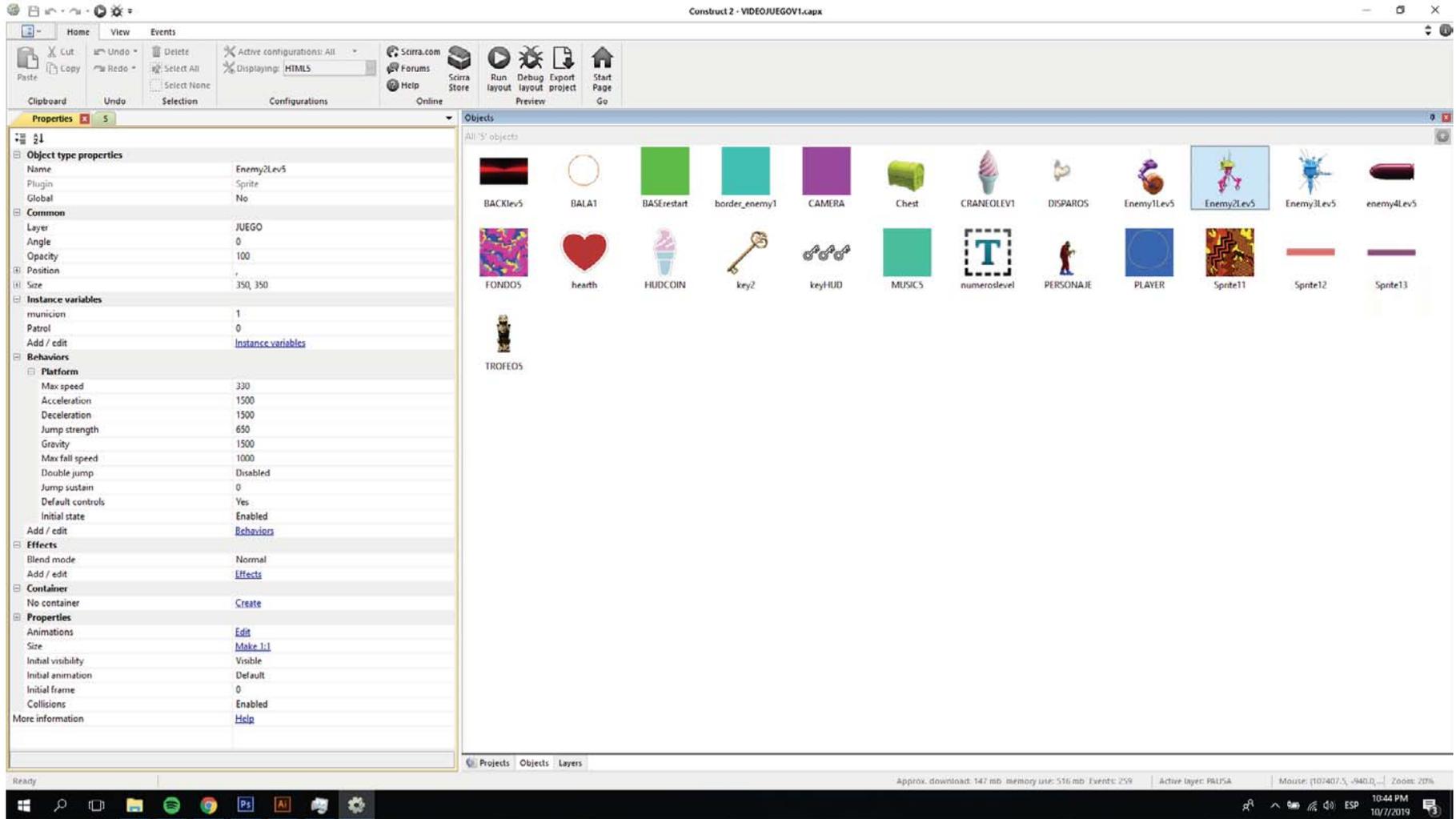


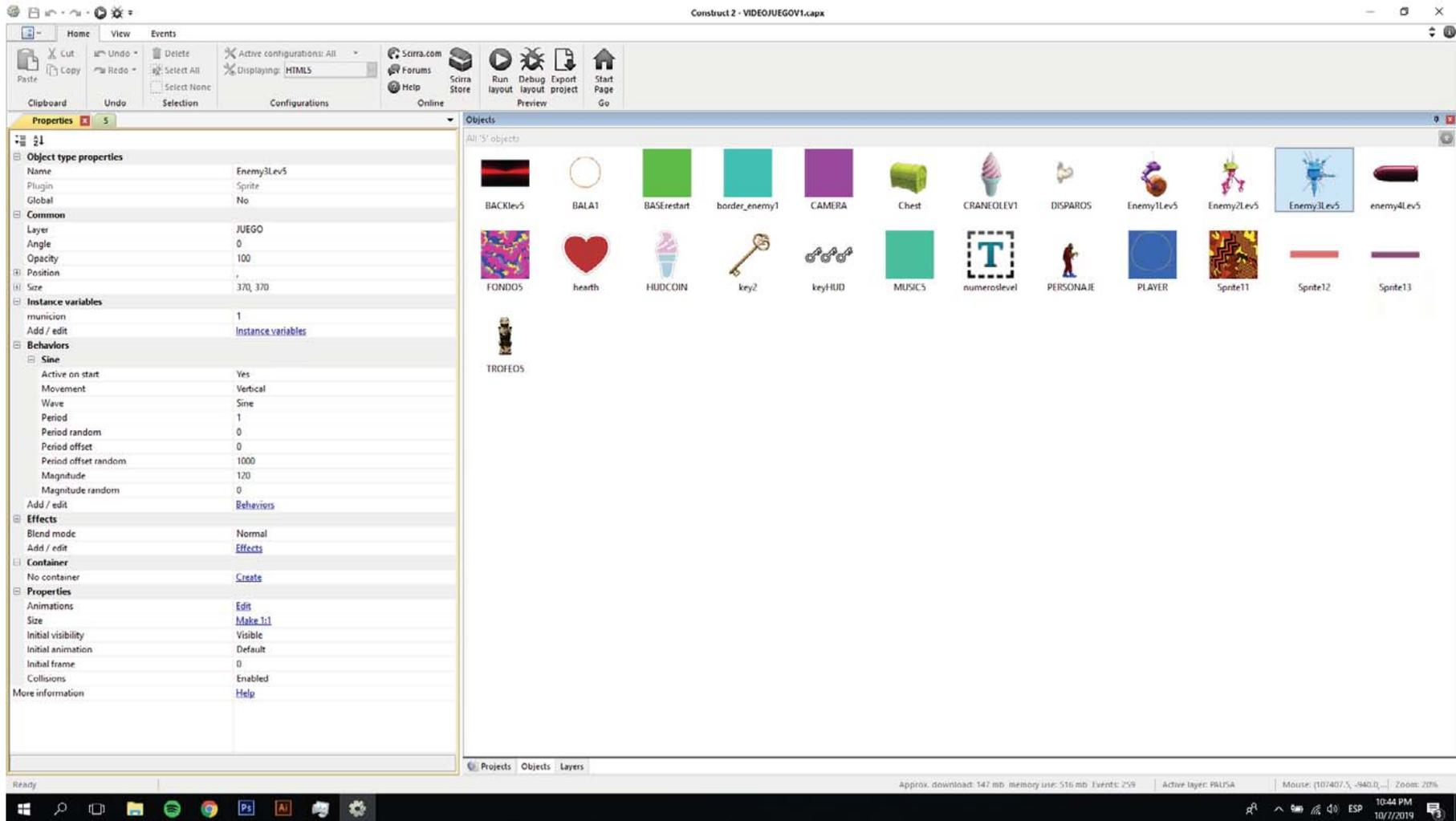


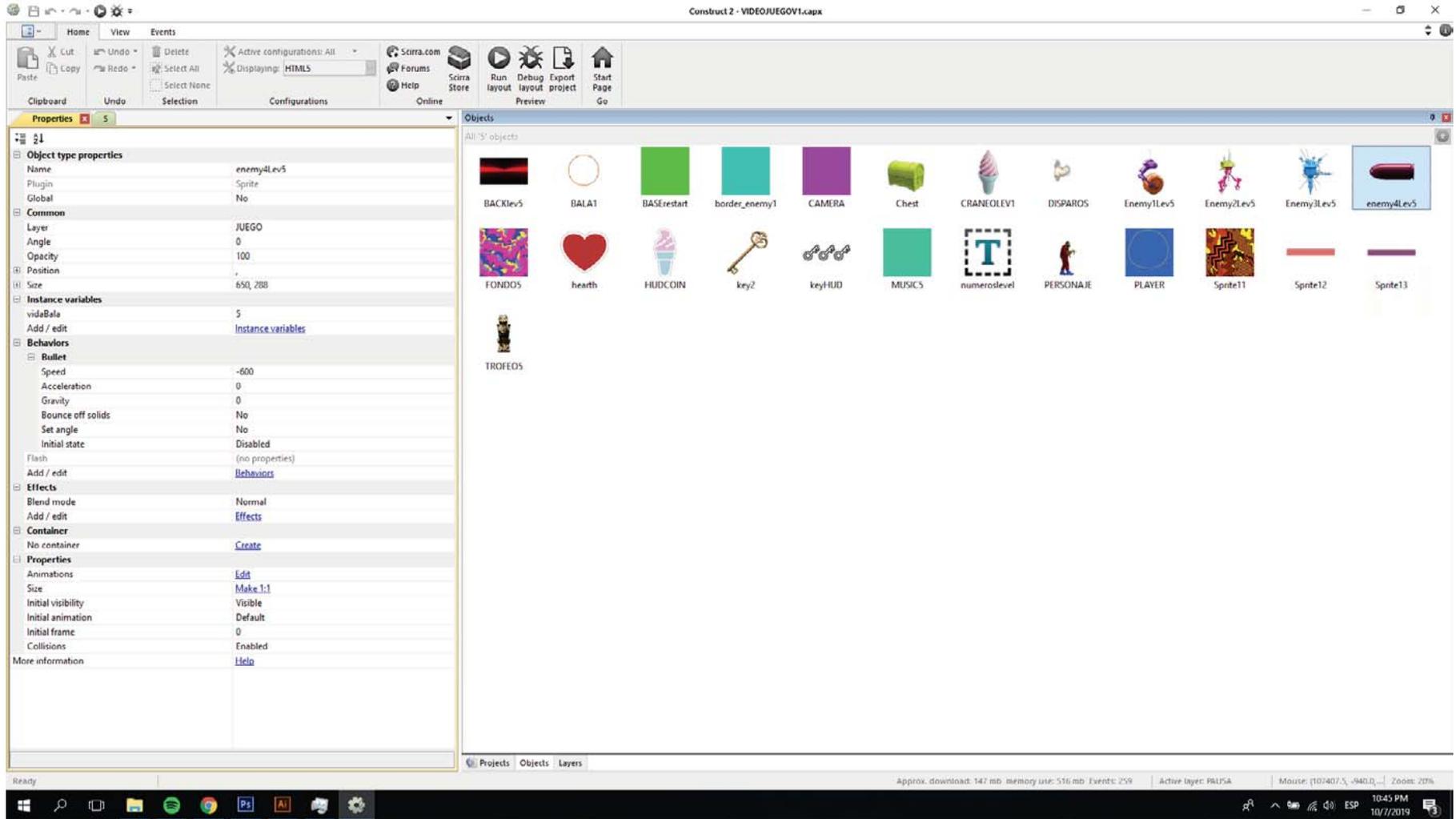


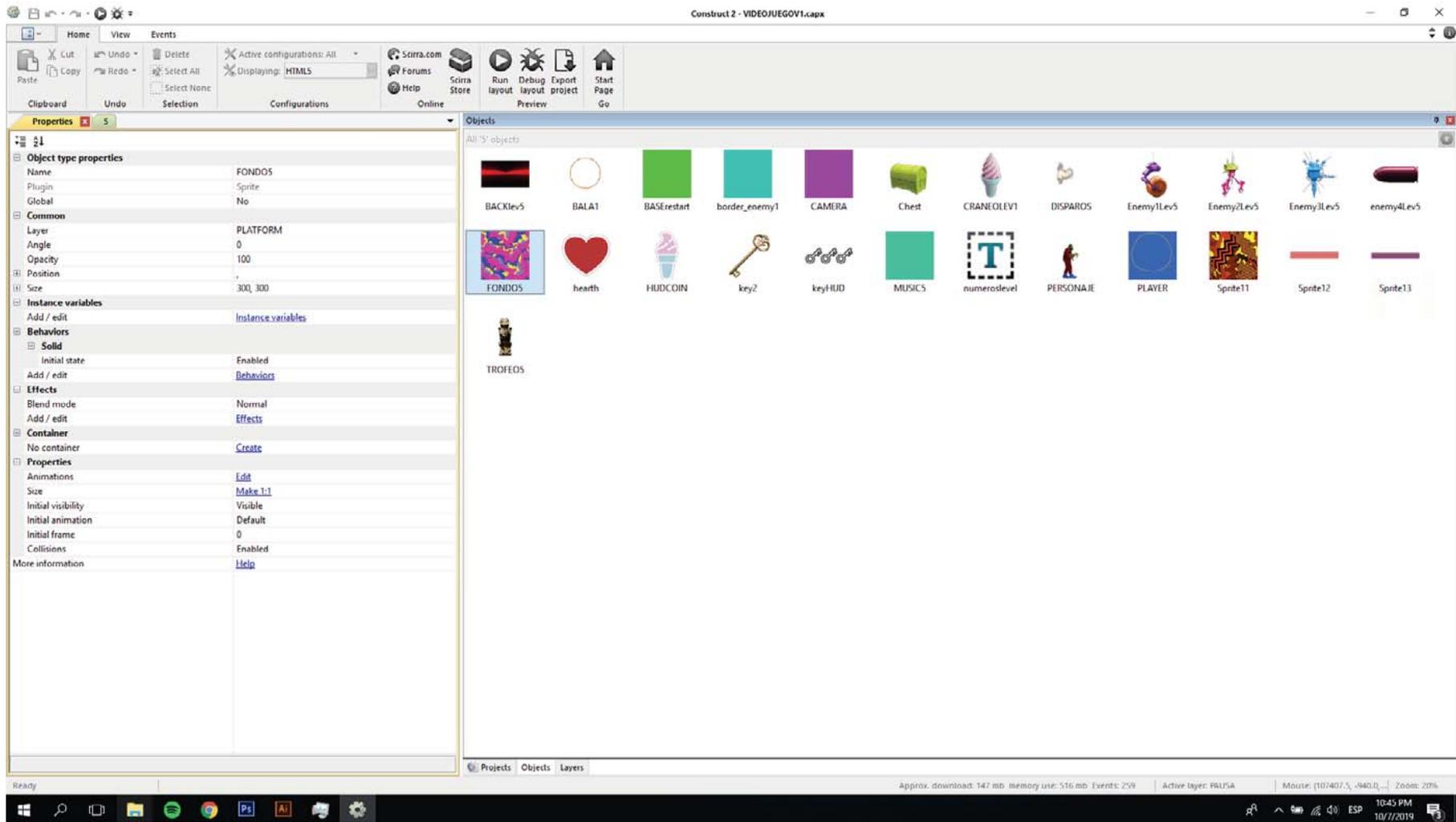
Propiedades de los Objetos Nivel 5

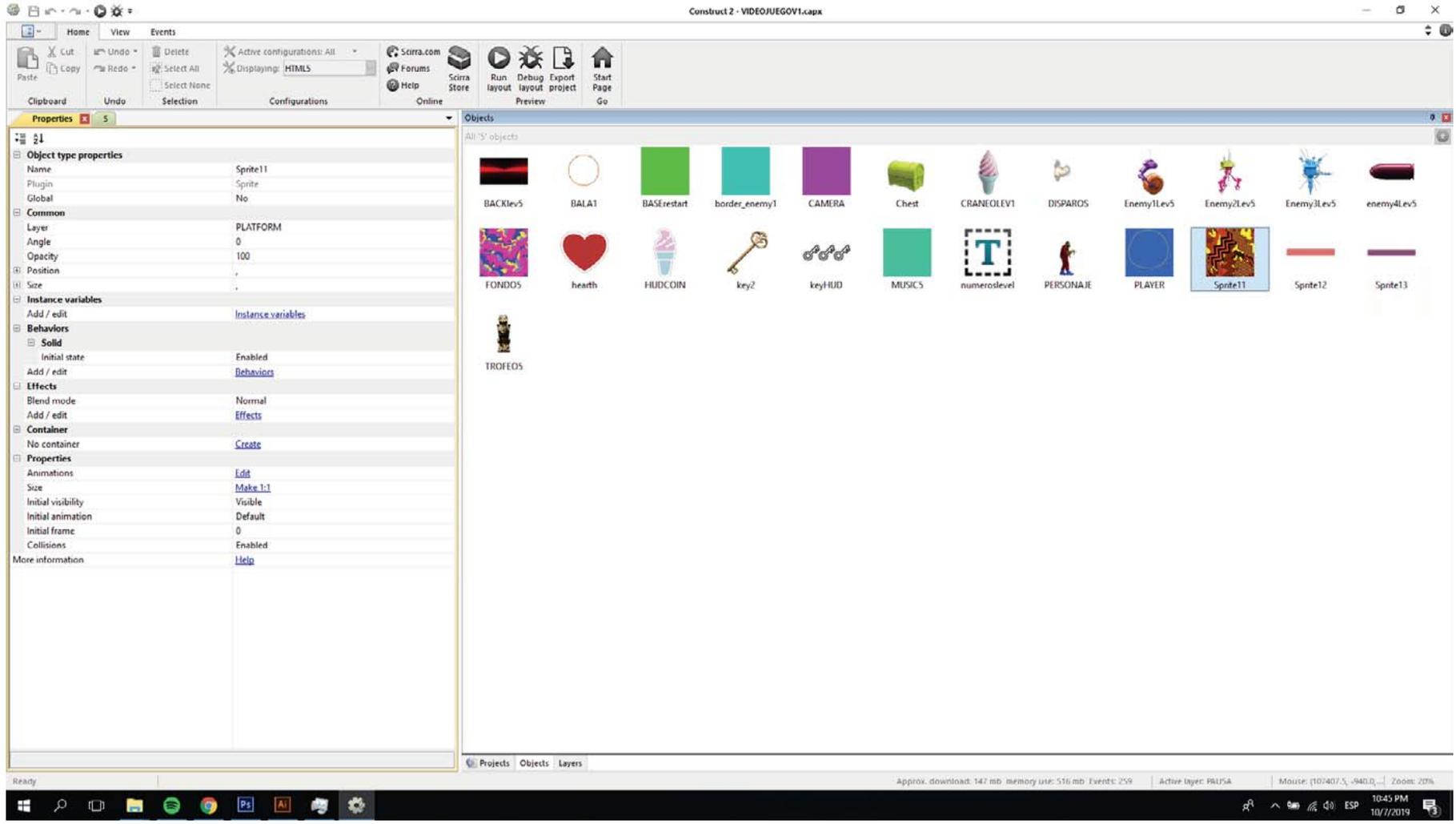


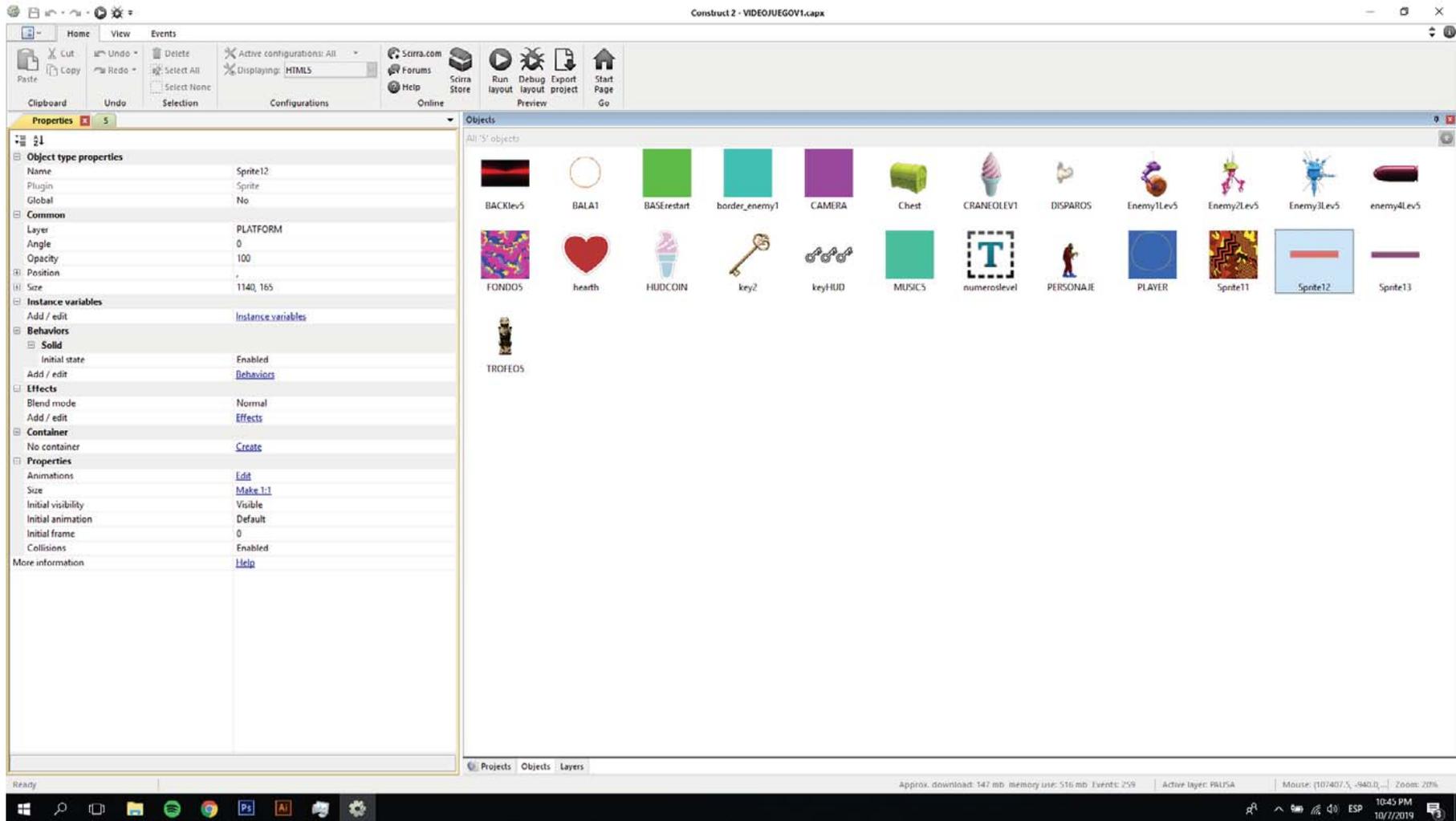


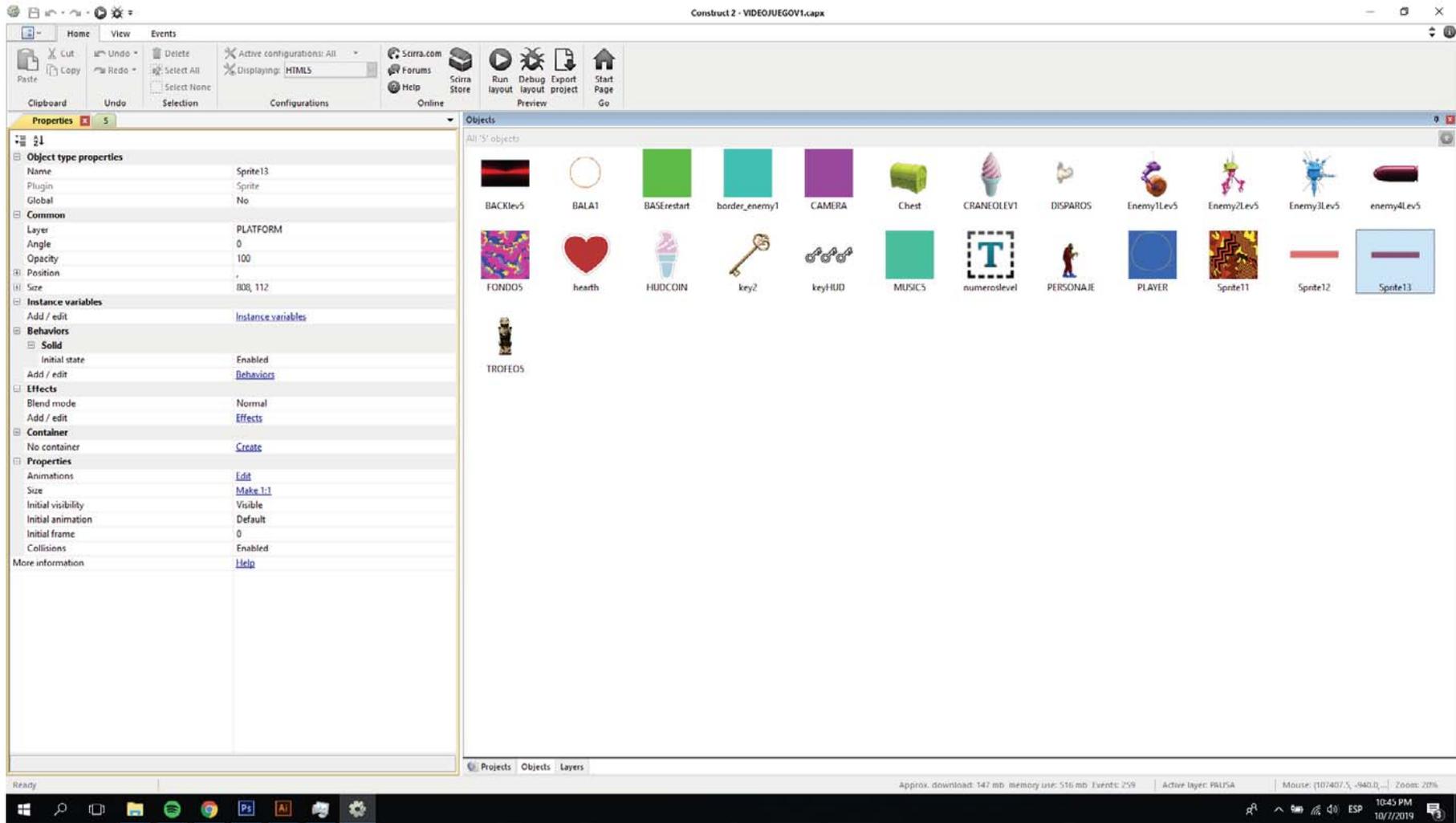




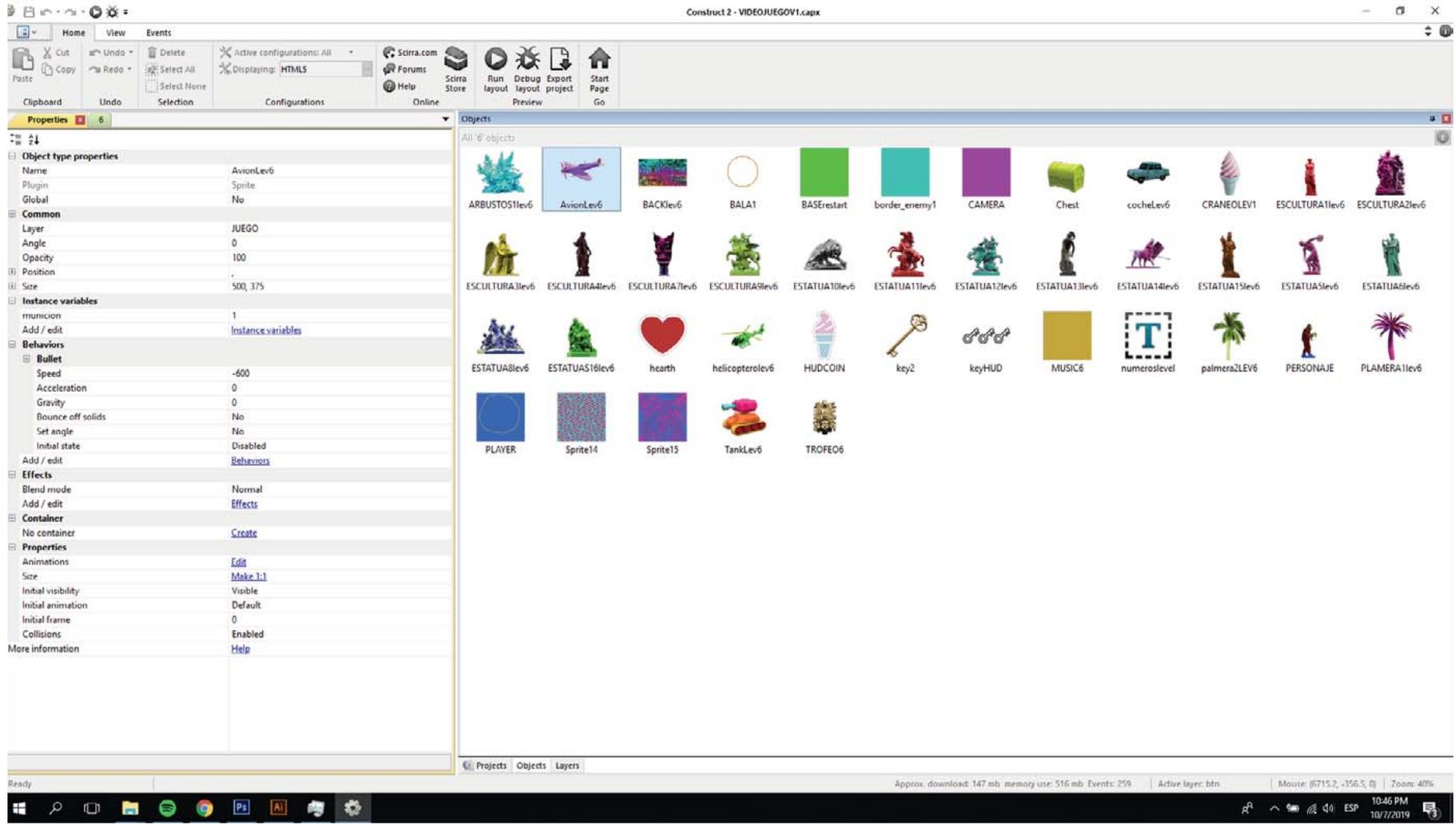


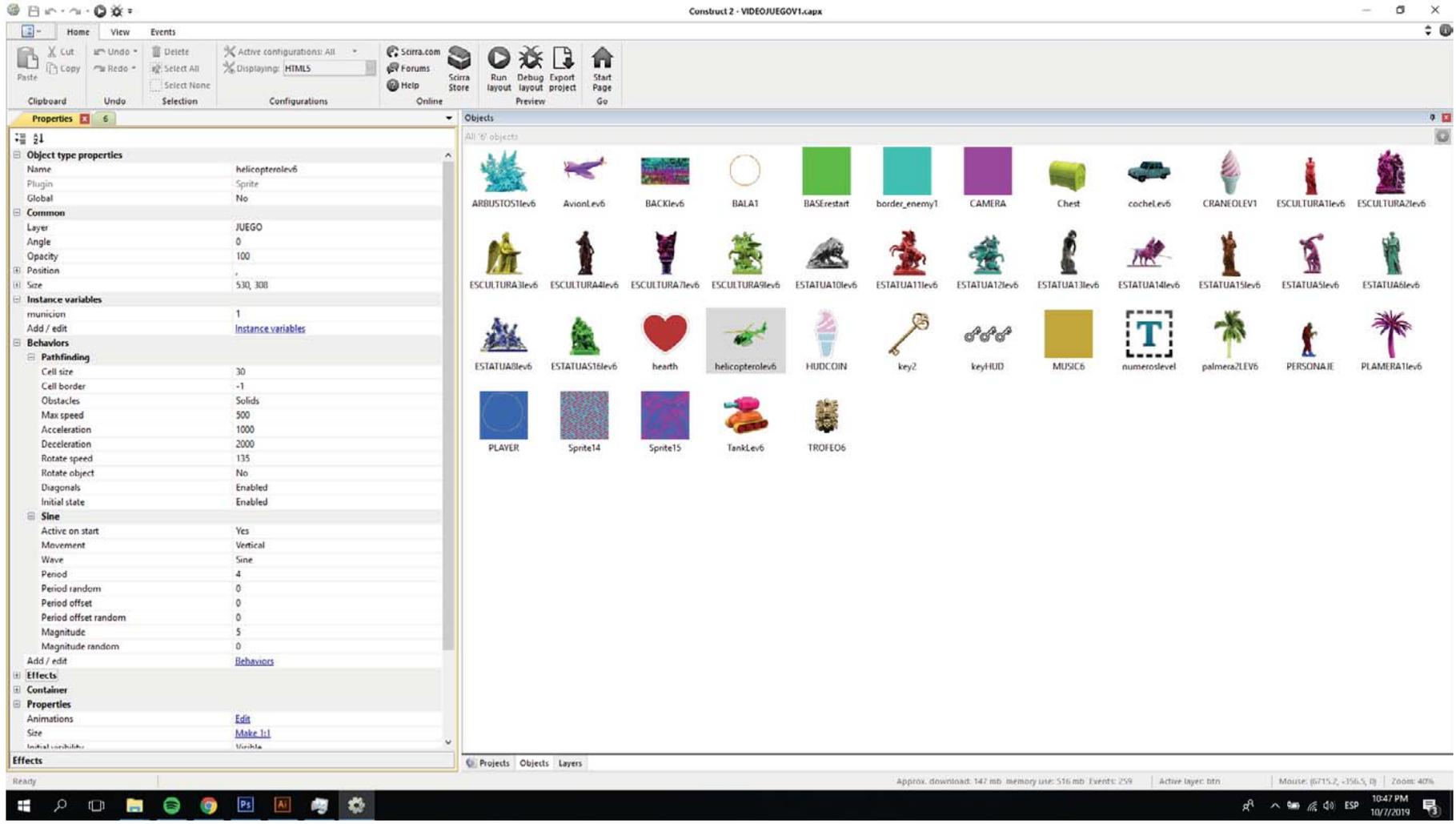


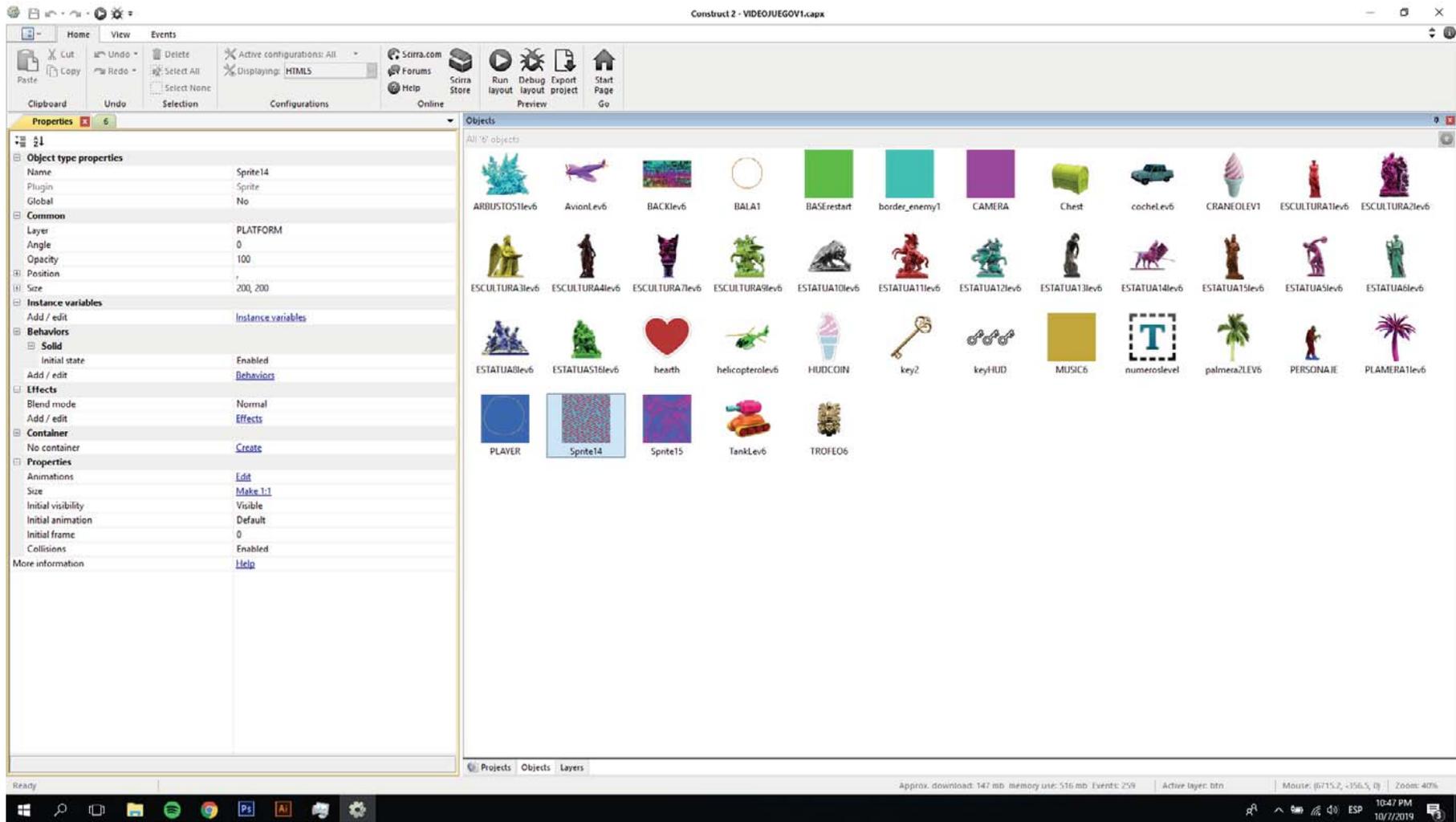


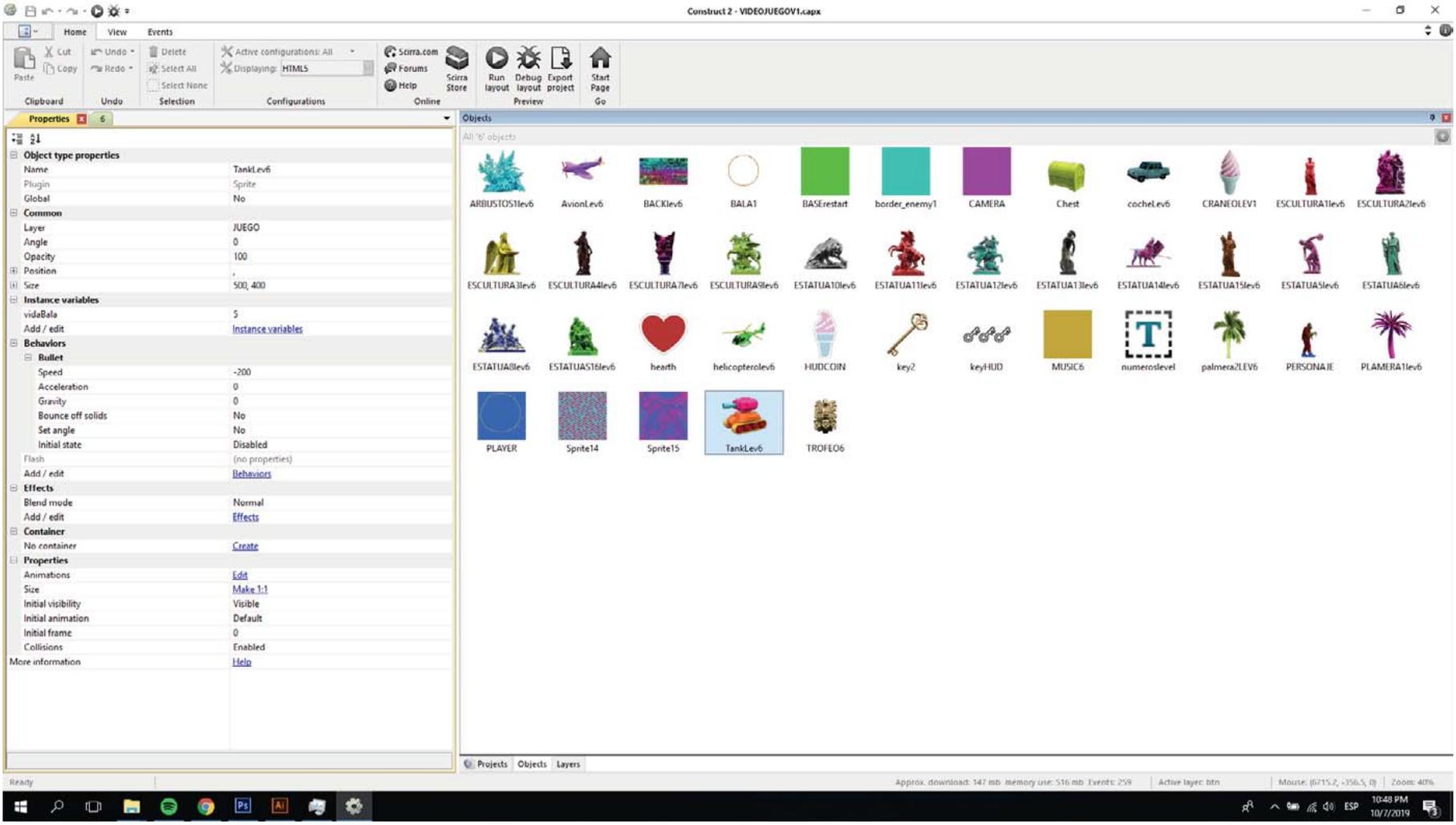


Propiedades de los Objetos Nivel 6









Glosario

0-9

2D graphics (Gráficos 2D): técnica de representación gráfica en una perspectiva bidimensional, a menudo utilizando sprites.

2.5D graphics (Gráficos 2.5D): técnica de representación gráfica de objetos tridimensionales establecidos en un plano de movimiento bidimensional. A menudo incluye juegos donde algunos objetos se representan como sprites.

3D graphics (Gráficos 3D): técnica de representación gráfica con objetos tridimensionales.

4K: hace referencia a varios formatos de distintos tamaños de imagen que tienen alrededor de 4000 píxeles de resolución horizontal. El DCI 4K 4096x2160 como estándar emergente para resolución en cine digital y en infografía, de relación 17:9, y por otra parte el 4K UHD TV (Ultra HD) 3840x2160 (2160p) usado en la industria de la televisión digital, de relación 16:9.

5G: en telecomunicaciones, 5G son las siglas utilizadas para referirse a la quinta generación de tecnologías de telefonía móvil.

8K: se refiere a una resolución de imagen o pantalla con un ancho de aproximadamente 8000 píxeles.

8-bit: hardware o software que surgió durante la tercera generación de consolas de videojuegos a comienzos de los 80 tras la crisis del videojuego dirigido a la arquitectura de computadoras de 8 bits.

16-bit: hardware o software que surgió durante la cuarta generación de consolas de videojuegos, dirigido a la arquitectura de computadoras de 16 bits.

32-bit: hardware o software que surgió durante la quinta generación de consolas de videojuegos, dirigido a la arquitectura de computadoras de 32 bits.

64-bit: hardware o software que surgió durante la quinta generación de consolas de videojuegos, dirigido a la arquitectura de computadoras de 64 bits.

A

Acciones (Actions): cómo reacciona un objeto al input o su interacción con otros objetos.

Algoritmo (Algorithm): un script de computadora que realiza los pasos necesarios para resolver un problema.

Alpha channel (Canal alfa): varía la opacidad del color desde la transparencia total hasta la opacidad total.

Alpha release (Lanzamiento alfa): una versión inicial e incompleta de un juego. Las versiones alfa generalmente se lanzan temprano en el proceso de desarrollo para probar la funcionalidad más crítica de un juego y los conceptos de diseño de prototipo.

Animación (Animation): Serie de frames reproducidos en secuencia con pequeñas diferencias entre cada uno que el cerebro interpreta como movimiento.

Glosario

Aplicación (Application - app): Programa para dispositivos de mano y teléfonos celulares.

Array: es una colección ordenada de datos (tanto primitivos u objetos dependiendo del lenguaje). Los arrays se emplean para almacenar múltiples valores en una sola variable, frente a las variables que sólo pueden almacenar un valor (por cada variable).

Arcade:

1. suele referirse a los juegos relativamente fáciles de jugar o que no responden fielmente a la gravedad y otras fuerzas.

2. es una máquina de entretenimiento que funciona con monedas, que generalmente se instala en empresas públicas como restaurantes, bares y salas de juegos. La mayoría de los juegos de arcade son videojuegos, máquinas de pinball, juegos electromecánicos, juegos de canje o exhibidores.

Art Designer (Diseñador de arte): crea los elementos visuales y los assets del juego.

Artificial Intelligence (AI) (Inteligencia artificial): Algoritmos utilizados para generar un comportamiento de juego sensible, adaptativo o inteligente, principalmente en personajes que no son jugadores. Distinto del concepto de ciencias de la computación sobre 'inteligencia artificial'.

Asset: Un asset es una representación de cualquier ítem que puede ser utilizado en el juego o proyecto.

Atari 2600: primer sistema de videojuegos comercialmente exitoso (1977) para hogares; permitió al jugador comprar individualmente cartuchos de juegos.

Audio engineer (Ingeniero de audio): Crea todos los assets de sonido.

Audio/visual effects (Efectos de audio/visuales): entretenimiento que puede ser visto u oído, pero que el observador no interactúa con él.

Augmented Reality (AR) (Realidad Aumentada): Complementa un entorno del mundo real con información perceptiva generada por computadora, que puede agregar o mejorar información del entorno físico.

Authorship restriction (Restricción de autoría): permite el uso de trabajo Creative Commons siempre que otorgue crédito al autor o creador original.

B

Background music (Música de fondo): Melodía que establece el estado de ánimo del nivel.

Backward compatibility (Compatibilidad con versiones anteriores): los juegos más antiguos se pueden jugar en consolas de videojuegos nuevas.

Battle Royale: videojuegos de acción tipo shooter (en primera o tercera persona). Es un todos contra todos donde cada jugador normalmente lucha por su cuenta en un campo de batalla y tiene que hacer todo lo posible por ser el último que quede en pie.

Beta: versión de prueba de un videojuego, donde el desarrollador del juego busca

Glosario

comentarios de los jugadores y evaluadores para eliminar errores antes del lanzamiento comercial del producto.

Big Data: es un campo que trata las formas de analizar, extraer sistemáticamente información o tratar conjuntos de datos que son demasiado grandes o complejos para ser manejados por el software tradicional de procesamiento de datos.

Bit: término informático para un único dígito binario de 0 o 1.

Bit Depth (Profundidad de bits): se refiere a la cantidad de bits de información necesarios para representar el color de un píxel en una imagen digital o en un framebuffer.

Bitmap (Mapa de bits): formato de archivo de imagen ráster que asigna digitalmente la ubicación y el color de cada píxel.

Blu-ray: es un formato de disco óptico de nueva generación, empleado para vídeo de alta definición (HD), 3D y Ultra HD con mayor capacidad de almacenamiento de datos de alta densidad que la del DVD.

Budget (Presupuesto): plan que detalla cuánto debería costar un proyecto.

C

Cámara: visualmente muestra el área del juego del videojuego y sigue al jugador donde quiera que vaya el personaje.

Casual Gamers (Jugadores casuales): jugadores que juegan con poca frecuencia cuando tienen tiempo libre.

Casual Gaming (Juegos casuales): los videojuegos casuales se distinguen por una curva de aprendizaje baja y facilidad de acceso, a menudo basada en la web para teléfonos móviles o computadoras personales. La mayoría de los juegos casuales tienen controles simplificados, con uno o dos botones que dominan el juego. Normalmente se pueden jugar en pequeños períodos de tiempo y es posible que no tengan una función de guardar.

Central process unit (CPU) (Unidad central de procesamiento):

1. la parte de la computadora o videojuego que ejecuta el programa de los juegos.
2. Una computadora personal PC.
3. Un personaje que no sea jugador controlado por el software del juego usando inteligencia artificial, usualmente sirviendo como oponente del jugador o jugadores.

Character Design Document (Documento de diseño de personaje): Bocetos que detallan la apariencia, poses y habilidades de un personaje.

Cloud Computing: es la disponibilidad on-demand de los recursos del sistema informático, especialmente el almacenamiento de datos y la potencia informática, sin una gestión activa directa por parte del usuario. El término se usa generalmente para describir los centros de datos disponibles para muchos usuarios a través de Internet.

Cloud Gaming (Juegos en la nube): funcionan con un servidor de juegos en la nube que ejecuta el juego y realiza todo el procesamiento, recibe acciones de entrada de los usuarios en red y transmite audio y vídeo del juego a estos usuarios. Esen-

Glosario

cialmente, los juegos en la nube son como un servicio de transmisión de video, pero interactivo.

Cheat codes (Códigos de trucos): atajos programados y códigos de habilidad que facilitan el trabajo del Tester del juego.

Cinematic cutscene (Escena cinematográfica): segmento similar a una película de un juego que no requiere que el jugador interactúe con ningún elemento del juego.

Color palette (Paleta de colores): conjunto de colores utilizados en una escena para mantener el estado de ánimo y la continuidad.

Commercial success (Éxito comercial): producto que genera suficientes ganancias para continuar produciéndose.

Compact disc, read-only memory (CD-ROM): proporciona videojuegos intercambiables en un disco de plástico económico; Tecnología de reemplazo para los cartuchos de juegos ROM.

Computational thinking (Pensamiento computacional): es un conjunto de métodos de resolución de problemas que implican expresar problemas y sus soluciones de manera que una computadora pueda ejecutar.

Concept art (Arte conceptual): bosquejo de la forma y el estilo básicos de un personaje u otro objeto.

Concept stage (Etapa conceptual): primera etapa de desarrollo que se enfoca en formalizar la idea principal del juego.

Console (Consola): una unidad de hardware de videojuegos que generalmente se conecta a una pantalla de video y controladores, junto con otro hardware. A diferencia de las computadoras personales, una consola generalmente tiene una configuración de hardware fija definida por su fabricante y no se puede personalizar.

Console generations (Generaciones de consolas): un conjunto de consolas de videojuegos en competencia directa por la cuota de mercado en una era determinada. El conjunto, como generación, está obsoleto en la introducción de la “próxima generación”.

Construct 2: es un Game Engine 2D basado en HTML5 desarrollado por Scirra Ltd., usa programación visual por medio de eventos.

Copyrighted: protección legal contra terceros que copian una creación original.

Core mechanics (Mecánica central): Programación dentro del motor del juego para hacer cumplir las reglas y los comandos del sistema. Es la actividad de juego esencial que los jugadores realizan una y otra vez en un videojuego.

Creación original: algo original de la mente o intelecto de alguien.

Creative Commons License (Licencia Creative Commons): permite que alguien use una obra creativa de forma gratuita, con algunas restricciones.

Creative director (Director creativo): establece la visión y dirección, selecciona personal, asigna roles de trabajo y se comunica entre la gerencia y el equipo de diseño.

Crowdfunding: es la práctica de financiar un proyecto o empresa recaudando pequeñas cantidades de dinero de un gran número de personas, generalmente a

Glosario

través de Internet.

D

Deep color (Color profundo): una profundidad de color que utiliza una profundidad de bits de 48, que produce más de mil millones de colores.

Developer (Desarrollador): la productora que fabrica un videojuego.

Difficulty (Dificultad): el nivel de dificultad que un jugador desea enfrentar mientras juega un juego; en niveles de dificultad más altos, el jugador generalmente se enfrenta a NPC más fuertes, recursos limitados o límites de tiempo más estrictos.

Documentos de diseño (Design Documents): documentos estandarizados utilizados por un equipo de diseño para registrar todas las decisiones y la visión del juego.

Digital video disc (Disco de video digital (DVD)): similar a un CD ROM o CD de música, pero puede contener mucha más información.

Dots per inch (dpi) (Puntos por pulgada (ppp)): número de píxeles por pulgada cuadrada; cuanto mayor es el dpi, más clara es la imagen.

E

Emulator (Emulador): entorno de juego simulado que tiene un aspecto similar al del juego original.

End User (Usuario final): consumidor o última persona en una cadena de suministro.

End User License Agreement (Acuerdo de licencia de usuario final (EULA)): conjunto de reglas que cada miembro debe aceptar antes de usar el software.

Endless Mode (Modo sin fin): modo de juego en el que los jugadores tienen el desafío de durar el mayor tiempo posible contra una amenaza continua con recursos o vidas limitados.

Entertainment Software Rating Board (ESRB): Organismo autorregulador sin fines de lucro que asigna clasificaciones de edad y contenido para videojuegos de computadora, hace cumplir las pautas publicitarias adoptadas por la industria y ayuda a garantizar prácticas responsables de privacidad en línea para la industria del software de entretenimiento interactivo.

Ergonomics (Ergonomía): estudio de cómo funciona el cuerpo y cómo los humanos interactúan con herramientas y entornos.

Esports: también deportes electrónicos, e-sports, eSports, juegos competitivos, cybersports y juegos profesionales. Competiciones organizadas en torno a videojuegos de competencia, a menudo jugados por premios económicos y reconocimiento.

F

Faces (Caras): superficies planas en un modelo 3D.

First playable level (Primer nivel jugable): primer objetivo alcanzado en la construcción del juego mediante el cual un nivel completo se ha programado de manera

Glosario

jugable.

First-person perspective (Perspectiva en primera persona): vista de juego donde el jugador ve a través de los ojos del personaje.

Floats: números con decimales.

Flowchart (Diagrama de flujo): organizador gráfico que muestra instrucciones a ser programadas para las interacciones del juego.

Frame rate (Velocidad de fotogramas): cuántas veces por segundo se muestra un nuevo fotograma.

Freemium: es una estrategia de precios mediante la cual un producto o servicio se proporciona de forma gratuita, pero se cobra dinero (premium) por funciones y servicios adicionales como bienes virtuales (online) o físicos (offline).

Free-to-Play (F2P o FtP): juegos que no requieren compra a un minorista para jugar, ya sea físico o digital.

G

Game (Juego): Actividad definida por reglas con un objetivo, meta o condición de victoria; se trata de un entorno de juego que permite jugar o simular.

Game Boy: dispositivo portátil lanzado por Nintendo en 1989 con procesador de 8 bits y cartuchos ROM intercambiables.

Game Designer (Diseñador del juego): establece las reglas del juego, finaliza los documentos de diseño y garantiza que el juego funcionará según lo previsto.

Game Design Document (GDD) (Documento de Diseño de Juego): es un documento vivo de diseño cuyo contenido es altamente descriptivo acerca del desarrollo de un videojuego. Un GDD está creado y editado por el equipo de desarrollo y es principalmente utilizado en la industria de videojuego para organizar esfuerzos dentro de un equipo de desarrollo.

Game-Development tools (Herramientas de desarrollo de juegos): hardware y software utilizados para simplificar y simplificar el proceso de diseño del juego.

Game Engine (Motor de juego): la plataforma de programación utilizada para crear y ejecutar un juego.

Game Tester (Probador del juego): persona que prueba cada parte del juego para verificar que funcione correctamente.

Game World (Mundo del juego): escenario en el que se desarrolla el juego.

Gameplay (Jugabilidad): lo que el jugador experimenta durante el juego como resultado de las mecánicas y las estructuras básicas del juego. Es la interacción de un jugador con un videojuego, definida a través de las reglas del juego, la interfaz del jugador, los desafíos, la trama y la conexión del jugador con el juego. Eficiente interacción por medio de coordinación mano-ojo.

Game mechanics (Mecánicas de juego): son las acciones básicas, procesos, visualizaciones y mecanismos de control que se utilizan para la gamificación. Son las reglas y recompensas que componen el juego y crean una experiencia atractiva.

Generación 1 de videojuegos: primera serie de videojuegos y sistemas; disponible

Glosario

para los consumidores a principios de la década de 1970.

Generación 2 de videojuegos: comenzó a mediados de la década de 1970 cuando los videojuegos domésticos se hicieron populares y se podían jugar en una plataforma de cartuchos intercambiables.

Generación 3 de videojuegos: comenzó a mediados de la década de 1980 y contó con sistemas informáticos de videojuegos para el hogar con procesadores de 8 bits que podían procesar mejores movimientos, gráficos y sonido.

Generación 4 de videojuegos: comenzó a fines de la década de 1980 y contó con sistemas de juegos de computadora con procesadores de 16 bits que permitieron mejorar los gráficos del juego.

Generación 5 de videojuegos: comenzó a mediados de la década de 1990; introdujo sistemas de juegos de 32 y 64 bits con avances en tecnología informática y almacenamiento de juegos.

Generación 6 de videojuegos: comenzó a principios de la década de 2000; sistemas de juego de 64 y 128 bits, tecnología DVD ROM y movimiento 3D realista.

Generación 7 de videojuegos: comenzó a mediados de la década de 2000; introdujo sistemas de juegos con procesador multinúcleo y nuevas interfaces de usuario; juego inmersivo destacado y juego en línea.

Generación 8 de videojuegos: incluye consolas lanzadas desde 2012 hasta 2018 por Nintendo, Microsoft y Sony.

Género: tipo o categoría principal de videojuegos con jugabilidad similar.

Glitch: error de programación dentro de un juego.

Gone Gold: el punto en el ciclo de desarrollo de software donde el software se considera final y listo para ser enviado.

Gráficos: Imágenes visuales vistas en la pantalla.

Graphical User interface (Interfaz gráfica de usuario (GUI)): es un programa informático que actúa de interfaz de usuario, permite a los usuarios interactuar con dispositivos electrónicos a través de iconos gráficos e indicadores de audio, representando la información y acciones disponibles en la interfaz.

H

Hand-Eye Coordination (Coordinación mano-ojo): capacidad de mover la mano en respuesta a un objeto de acción visto.

Handheld Console: una consola de juegos portátil; es decir, uno que no esté conectado a un televisor u otro dispositivo periférico. Game Boy de Nintendo es el ejemplo más reconocible.

HDTV: describe un sistema de televisión que proporciona una resolución de imagen de resolución sustancialmente más alta que la generación anterior de tecnología, el SD. A menudo abreviado como HDTV o HD-TV. Es el formato de video estándar actual utilizado en la mayoría de las transmisiones. Full HD corresponde a 1920x1080.

HDRI (High Dynamic Range Imaging): formato de imagen que contiene desde la

Glosario

sombra más profunda hasta la información más brillante de los niveles de blanco (highlights).

High Concept Document: se utiliza para presentar la idea o el concepto de un juego y responder las cinco preguntas esenciales.

Heads up display (HUD): Elementos gráficos que comunican información al jugador y ayudan a la interacción con el juego, como barras de salud, medidores de munición y mapas.

I

Inmersión: grado en el que un jugador se conecta al mundo de un videojuego.

Indie Developer (Desarrollador independiente): creadores de juegos independientes y firmas independientes no convencionales.

Indie Game: definido generalmente como un juego hecho por una sola persona o un pequeño estudio sin ningún tipo de apoyo financiero, de desarrollo, marketing o distribución de un gran editor, aunque hay excepciones.

In-app purchase (IAP): una micro transacción en un juego móvil (o aplicación normal), generalmente para productos virtuales en juegos gratuitos o baratos.

Intellectual Property (Propiedad intelectual): producto que es una creación original.

Interactivity (Interactividad): cómo se comporta un objeto en relación con los objetos que lo rodean.

Interdisciplinary team (Equipo interdisciplinario): compuesto por muchos miembros con diferentes habilidades fuertes que trabajan para lograr el mismo objetivo.

Iteration (Iteración):

1. una sola ejecución a través de un loop de programación.
2. versión diferente de la misma producción del juego con muchas más mejoras.

Iterative process (Proceso iterativo): proceso mediante el cual se construye y mejora un juego prototipo básico a través de muchas versiones mejoradas hasta que se perfecciona la versión final.

Internet of Things (Internet de las cosas): es un sistema de dispositivos informáticos interrelacionados, máquinas mecánicas y digitales, objetos, animales o personas que cuentan con identificadores únicos (UID) y la capacidad de transferir datos a través de una red sin requerir interacción de humano a humano o humano computadora.

J

Jugar: Participación en una actividad de entretenimiento.

L

LAN Party (Fiesta LAN): los jugadores se reúnen en un lugar y conectan sus máquinas a una red de área local (LAN) para jugar un juego en línea.

Glosario

Lead artist (Artista principal): dirige al diseñador de arte y el flujo de trabajo (workflow) necesario para crear todos los assets del juego.

Level Designer (Diseñador de niveles): interpreta los documentos de diseño para crear un mundo de juego e interacciones para un solo nivel de juego.

M

Machine Code (Código de máquina): un lenguaje binario que consta de solo dos caracteres, 0 y 1.

Magnavox Odyssey: es la primera consola comercial de videojuegos domésticos. Fue desarrollado por un pequeño equipo dirigido por Ralph H. Baer.

Mainstream media (Medios principales): tecnología y entretenimiento que es popular en ese momento, aceptado por la mayoría de las personas y, en general, parte de la vida cotidiana.

Massive Online Open Course (MOOC) (Curso En-línea Masivo y Abierto): son cursos en línea dirigidos a un número ilimitado de participantes a través de Internet según el principio de educación abierta y masiva.

Massively Multiplayer Online (MMO) (Juegos multijugador masivo en línea): juegos interactivos que involucran a miles de personas jugando el mismo juego, cada uno con un avatar, conectándose al servidor del juego a través de Internet.

Mecánica del juego: término general que describe cómo funciona un juego en particular y que es posible dentro del entorno del juego; las reglas del juego. Un uso inesperado y novedoso de la mecánica del juego puede conducir a un juego emergente.

Mesh (Malla): una forma 3D creada con polígonos de interconexión pegados entre sí a lo largo de sus bordes.

Model (Modelo): asset tridimensional.

Model sheet: bocetos detallados y refinados que muestran al personaje en múltiples poses, vestimenta y expresiones emocionales.

Motion capture (Captura de movimiento): interfaz de usuario compleja que lee el movimiento del cuerpo y lo muestra en el mundo del videojuego.

Multi Platform (Multi Plataforma): un juego que se puede jugar en múltiples plataformas.

Multiplayer Online Battle Arena (MOBA) (Arena de batalla en línea multijugador (MOBA)): un género de videojuego popularizado por League of Legends que enfrenta a equipos de jugadores para defender su base de ataques enemigos.

N

Nintendo Entertainment System (NES): comenzó a venderse en 1985 y dominó el mercado de sistemas domésticos en ese momento.

Newbs: jugadores nuevos.

Non-Player Character (NPC) (Personaje no jugador): un personaje controlado por computadora o cualquier personaje que no esté bajo el control directo de un

Glosario

jugador.

O

Online Game (juego en línea): Un juego en el que parte del motor del juego está en un servidor y requiere una conexión a Internet. Muchos juegos multijugador admiten el juego en línea.

Online play (partida en línea): los jugadores se conectan a través de Internet para competir con otros jugadores.

Outsourcing: contratar parte de la producción o marketing a otras compañías con habilidades específicas o de menor costo que la compañía productora.

P

Patente: conjunto de derechos exclusivos concedidos por un Estado al inventor de un nuevo producto o tecnología, susceptibles de ser explotados comercialmente por un período limitado de tiempo, a cambio de la divulgación de la invención.

Patreon: es un sitio web de micromecenazgo para proyectos creativos, fundado en 2013.

PC Gaming: jugar videojuegos en una computadora personal en lugar de una consola de juegos dedicada.

Physics Engine (Motor de física): utiliza fórmulas matemáticas para crear reglas de movimiento, gravedad, velocidad, vuelo, trayectoria de un proyectil y otros movimientos del videojuego.

Piracy (Piratería): copia y distribución ilegal de videojuegos.

Pixel: elemento de imagen; el punto o punto de color más pequeño que puede generar una pantalla de computadora.

Pixelado: condición que resulta en una imagen borrosa; creado por cambio de tamaño incorrecto.

Pipeline: es el proceso de crear un videojuego desde el concepto hasta su finalización.

PlayStation: un sistema de videojuegos de 32 bits de Sony que utiliza tecnología de CD-ROM.

PlayStation 3 (PS3): consola de videojuegos de la generation 7 fabricada por Sony.

PlayStation 4 (PS4): es la cuarta videoconsola del modelo PlayStation fabricada por Sony. Forma parte de las videoconsolas de octava generación. Cuenta con un procesador AMD de 8 núcleos bajo la arquitectura x86-64.

Playstation 5 (PS5): es una consola de videojuegos de sobremesa desarrollada por la empresa Sony Interactive Entertainment. Forma parte de la Novena generación de Consolas.

Poly count (Conteo de polígonos): número de polígonos utilizados para hacer un objeto 3D.

Pong: Fue un videojuego de la primera generación de videoconsolas publicado por Atari, creado por Nolan Bushnell y lanzado el 29 de noviembre de 1972, basado en

Glosario

el deporte de tenis de mesa (o ping pong).

Professional Game testers (Probadores de juegos profesionales): es el que juega y prueba un juego y observa todos los problemas técnicos e imperfecciones antes de que se lance al público.

Project manager (Gerente de proyecto): coordina a todos los gerentes principales y garantiza que todos los componentes del proyecto estén produciéndose y bien administrados.

Prototipo: primera versión básica de un videojuego utilizada para probar el concepto.

Procedural generation: cuando el juego combina algorítmicamente elementos generados aleatoriamente.

Programming (Programación): es el proceso de diseño y construcción de un programa informático ejecutable para lograr un resultado informático específico.

Pseudo Code (Pseudocódigo): declaraciones lógicas no escritas en el código de la computadora que permiten a los programadores descomponer las interacciones del programa en pasos lógicos antes de convertir el código en un lenguaje de computadora

Q

Quality Assurance (QA) (Aseguramiento de la calidad): conjunto de actividades planificadas y sistemáticas aplicadas en un sistema de gestión de la calidad para que los requisitos de calidad de un producto o servicio sean satisfechos.

R

Raster images (Imágenes ráster): imágenes que están hechas de píxeles.

Rendering: proceso de generar imágenes fotorrealistas o no fotorrealistas a partir de un modelo 2D o 3D por medio de programas informáticos. Los resultados de mostrar dicho modelo pueden llamarse render.

Reverse Engineering (Ingeniería inversa): proceso de deconstrucción de un juego existente para comprender cómo funciona.

S

Scrolling (Desplazamiento): el marco del juego se mueve para que el jugador esté siempre en el área de juego visible.

Sega Genesis: primera consola exitosa de videojuegos de 16 bits (1989); presentaba gráficos, colores y movimientos mejorados en 2D.

Shoot 'em Up (SHMUP): Un subgénero del género de disparos, donde un solo personaje, tiene que disparar a los enemigos mientras todos ellos atacan o se mueven hacia él. Contiene el subgénero bullet hell.

Six-axis Controller (Controlador de seis ejes): dispositivo de interfaz para la PlayStation Sony que permite al jugador sacudir o girar el controlador para ingresar algunos comandos.

Glosario

Smartphone: es un tipo de ordenador de bolsillo con las capacidades de un teléfono móvil.

Source Code (Código fuente): conjunto de líneas de texto con los pasos que debe seguir la computadora para ejecutar un programa. Código de creación del juego.

Sprite: asset bidimensional.

Sprite sheets (Hojas de sprites): una única imagen de mapa de bits de todos los cuadros de animación para el movimiento de sprites.

STEM: es el acrónimo de los términos en inglés Science, Technology, Engineering and Mathematics (Ciencia, Tecnología, Ingeniería y Matemáticas) El término sirve para agrupar a las 4 grandes áreas de conocimiento en las que trabajan científicos e ingenieros.

Storyboard: Bosquejo de los momentos importantes del juego con ideas generales para movimiento, trampas y recompensas.

Super Nintendo: una consola de juegos de 16 bits introducida en 1991 y que utiliza un sistema de cartucho ROM.

Switch: es la novena consola de videojuegos desarrollada por Nintendo, considerada como consola híbrida. Se puede utilizar como consola de sobremesa con la unidad principal insertada en una estación de acoplamiento para conectarla con un televisor; también puede ser extraída de la base y utilizada de forma similar a una tableta a través de su pantalla táctil LCD, o colocada sobre una superficie siendo así visible por varios jugadores.

T

Tasa de crecimiento anual compuesto (TCAC): es un término específico de negocios e inversión para la ganancia anualizada lisa de una inversión sobre un periodo dado.

Technical Design Document (TDD) (Documento de diseño técnico): detalla los códigos de computadora, los códigos de eventos y las interacciones técnicas entre la interfaz de usuario y la mecánica principal. Proporciona un plan para que los ingenieros de software de su equipo implementen y codifiquen las características de su juego.

Testing and debugging: (Pruebas y depuración): etapa de construcción centrada en encontrar los problemas dentro de un juego.

Third-Person perspective (Perspectiva en tercera persona): el juego es visto por una persona que no es el personaje u oponente del jugador, sino una tercera persona neutral; vista del espectador

Three-dimensional (3D) Game (Juego tridimensional (3D)): juego con personajes 3D y objetos de fondo 3D que presenta jugabilidad en un entorno simulado tridimensional.

Triple A (AAA): juego de alto presupuesto con un gran equipo de desarrollo o estudios de juegos que los hacen. Los juegos AAA generalmente son multiplataforma o First-Party, tienen presupuestos multimillonarios y esperan vender millones de

Glosario

copias, son los juegos más populares y mejor financiados.

Two and One-Half-dimensional (2.5D) Games (Juegos de dos dimensiones y media (2.5D)): juego con gráficos de fondo bidimensionales, pero que utilizan personajes y obstáculos tridimensionales.

Two-dimensional (2D) Games (Juegos bidimensionales (2D)): juego con personajes y fondos que juegan en solo dos dimensiones: largo y ancho.

U

User Interface (Interfaz de usuario (UI)): controles utilizados por el jugador para interactuar dentro del juego.

V

Vector image (Imagen vectorial): una imagen compuesta de líneas, curvas y rellenos; no almacenan el valor de color y la ubicación de cada píxel; a cada píxel se le asigna un color a medida que se dibuja la imagen vectorial.

Vertex (Vértice): Punto único en un modelo 3D donde las esquinas de caras adyacentes se encuentran.

Videogame (Videojuego): producto de software electrónico que tiene todos los elementos de un juego; se juega combinando un entorno de juego generado por computadora por medio de una pantalla de video y una interfaz de usuario.

Video Game crash of 1983: fue una recesión a gran escala en la industria de los videojuegos que se produjo entre 1983 y 1985, principalmente en los Estados Unidos.

Video Game Designer (Diseñador de videojuegos): es una persona que diseña el juego, concibe y diseña las reglas y la estructura de un juego.

Virtual Reality (VR) (Realidad virtual): la realidad virtual es una experiencia interactiva generada por computadora que tiene lugar dentro de un entorno simulado, que incorpora principalmente estímulos auditivos y visuales, pero también otros tipos de retroalimentación sensorial como el háptico. Este entorno inmersivo puede simular el mundo real o puede ser fantástico, creando una experiencia que no es posible en la realidad.

Virtual World (Mundo virtual): mundo imaginario creado por un videojuego.

W

Wii: consola de juegos de 7a generación fabricada por Nintendo; presentó la introducción del controlador basado en movimiento.

Wireframe: vista que muestra los objetos como si estuvieran contruidos con alambre con bordes visibles y caras invisibles.

Workflow (flujo de trabajo): son los aspectos operacionales de una actividad de trabajo: cómo se estructuran las tareas, cómo se realizan, cuál es su orden correlativo, cómo se sincronizan, cómo fluye la información que soporta las tareas y cómo se le hace seguimiento al cumplimiento de las tareas.

Glosario

X

Xbox: primera consola de juegos producida por Microsoft.

Xbox 360: consola de videojuegos de séptima generación fabricada por Microsoft.

Xbox One: es la tercera videoconsola de sobremesa de la marca Xbox, producida por Microsoft formando parte de las Videoconsolas de Octava Generación.

Xbox Series X y Series S: es una línea de videoconsolas de sobremesa desarrolladas por Microsoft de Novena Generación.

Bibliografía

Bibliografía

Adams, Ernest. (2010). *Fundamentals of game design*. 2a Ed. Estados Unidos. New Riders.

Aranda, Daniel. (2018). *Game & Play: La cultura del juego digital*. 1a Ed. España. Ediciones Egregius.

Arntson, Amy E. (2007). *Graphic Design Basics*. 5ta Ed. Estados Unidos. Thomson Wadsworth.

Augmented Reality - The Fourth Industrial Revolution | UK | Waracle. (2020). Recuperado 3 Enero 2020, de <https://waracle.com/blog/augmented-reality/augmented-reality-the-fourth-industrial-revolution/>

ASALE, R. (2020). videojuego | Diccionario de la lengua española. Recuperado 3 Enero 2020, de <https://dle.rae.es/videojuego>

Barnard, Malcolm. (2005). *Graphic Design As Communication*. USA. Routledge.

Bates, Bob. (2004). *Game Design*. 2da Ed. USA. Thomson, Course Technology.

Beale, Ashley (2012). *La comunicación: de los orígenes a internet*. 1a edición. España. Editorial Gedisa.

Berens, Kate. Howard, Geoff. (2008). *The Rough guide to Videogames*. UK-USA. Rough Guides

Bernal, César A. (2010). *Metodología de la investigación*. 3era Ed. Colombia. Pearson Educación.

Bocconi, Stefania & Chiocciariello, Augusto & Dettori, Giuliana & Ferrari, Anusca & Engelhardt, Katja & Kampylis, Panagiotis & Punie, Yves. (2016). *Developing Computational Thinking in Compulsory Education. Implications for policy and practice*. EUR - Scientific and Technical Research Reports. 10.2791/792158.

Bura, John. (2014). *Construct 2 Game Development by Example*. UK. Packt Publishing.

Burgun Keith. (2013). *Game Design Theory A New Philosophy for Understanding Games*. USA. CRC Press.

Bibliografía

- Cloud Gaming Market | Growth, Trends and Forecast (2019 - 2024). (2020). Retrieved 3 Enero 2020, de <https://www.mordorintelligence.com/industry-reports/cloud-gaming-market>
- Chang-Hun, Kim. Sun-Jeong, Kim. Soo-Kyun, Kim. Shin-Jin, Kang. (2015). Real-Time Visual Effects for Game Programming. 1a Ed. Singapore. Springer Singapore.
- Ching, Francis D.K. (2007). Architecture, form, space, and order. 3a Ed. USA. John Wiley.
- Cohen, D.S., Bustamante II, Sergio A. (2010). Producing Games. Reino Unido. Elsevier.
- Del Cid Pérez, Alma; Méndez, Rosemary y Franco Sandoval Recinos. (2007). Investigación. Fundamentos y metodología. 1ra edición. México. Pearson Educación.
- Dillon, Roberto. (2014). HTML5 Game development, from the ground up with Construct 2. Reino Unido. CRC Press.
- Diseño de Experiencia de Usuario o UX Design | Platzi. (2020). Recuperado 3 Enero 2020, de <https://platzi.com/disenio-ux/>
- Diseño de Interfaces (UI). (2020). Retrieved 3 January 2020, from <https://platzi.com/interfaces-ui/>
- Frascara, Jorge (2000). Diseño gráfico y comunicación. 7a Ed. Argentina. Ediciones Infinito.
- Game programming. (2020). Recuperado 3 Enero 2020, from https://en.wikipedia.org/wiki/Game_programming
- Goldsmith, Mike & Jackson, Tom. (2011). Eyewitness books, Computer. USA. DK Publishing.
- Gullen, A., & Gullen, T. (2011). Construct 2 Prototipos Tempranos Videojuego Cazareliquias (Diseñado en N° de versión Release 216 (64-bit), Compilado en N° de versión Release 233 (64-bit). Windows. Wandsworth, SW London: Scirra.
- González Bustamante, Oscar Alejandro. (2007). Guías y Textos de Cómputo: Introducción a la Programación. México. Dirección General de Servicios de Cómputo Académico - Universidad Nacional Autónoma de México.

Bibliografía

Jacko, Julie A. (2012). *The Human - Computer Interaction Handbook*. 3a Ed. USA. CRC Press.

Jamieson, Harry. (2007). *Visual Communication, More than Meets the Eye*. United Kingdom. Intellect Books.

Joyanes Aguilar, Luis. (2008). *Fundamentos de Programación, Algoritmos, estructura de datos y objetos*. 4a Ed. España. McGraw-Hill.

Kilmer, Rosemary, ASID, W. Otie Kilmer, AIA. (2014). *Designing interiors*. 2da Ed. USA. John Wiley & Sons, Inc.

Mexico Games Market 2018. (2020). Recuperado 3 Enero 2020, from <https://new-zoo.com/insights/infographics/mexico-games-market-2018/>

Landa, Robin (2011). *Graphic design solutions*. 4a Ed. Estados Unidos. Wadsworth.

Lee Mitchell, Briar. (2012). *Game design Essentials*. Estados Unidos. Wiley.

Mealy, Paul. (2018). *Virtual & Augmented Reality For Dummies (For Dummies (Computer/Tech))*. 1a Ed. USA. Wiley.

Munari, Bruno (2016). *Diseño y comunicación visual*. 2da Ed. Barcelona. Editorial Gustavo Gili.

Newman, James. (2004). *Videogames*. United Kingdom. Routledge.

Novak, Jeannie. (2012). *Game Development Essentials: An introduction*. 3a Ed. Estados Unidos. Cengage Learning.

Ongallo, Carlos (2007). *Manual de Comunicación*. 2a Ed. Madrid. Editorial Dykinson S.L.

Pardew, Les. (2004). *Game design for teens*. 1a Ed. Reino Unido. Cengage Learning.

Poulin, Richard. (2011). *The Language of graphic design*. USA. Rockport Publishers, Inc.

Processing (programming language). (2020). Recuperado 3 Enero 2020, de [https://en.wikipedia.org/wiki/Processing_\(programming_language\)](https://en.wikipedia.org/wiki/Processing_(programming_language))

Bibliografía

Samara Timothy. (2014). Design Elements, a Graphic Style Manual. 2a Ed. USA. Rockport Publishers.

Staff, F. (2020). ¿Listos para la cuarta revolución industrial? • Forbes México. Recuperado 3 January 2020, de <https://www.forbes.com.mx/listos-para-la-cuarta-revolucion-industrial/>

Skilton, Mark & Hovsepian, Felix. (2018). The 4th Industrial Revolution, Responding to the Impact of Artificial Intelligence on Business. USA. Palgrave, Macmillan.

Technical Director for Visual Effects & Video Games - Programming Coding. (2020). Retrieved 3 January 2020, from <https://artfx.school/en/technical-director-visual-effetscs-video-games-programming/>

The Fourth Industrial Revolution: what it means and how to respond. (2020). Recuperado 3 Enero 2020, de <https://www.weforum.org/agenda/2016/01/the-fourth-industrial-revolution-what-it-means-and-how-to-respond/>

Victorian Certificate of Education Study Design. (2012). Visual Communication Design. Australia. Victorian Curriculum and Assessment Authority.

Wing, Jeannette. (2006). Computational Thinking. Communications of the ACM. 49. 33-35. 10.1145/1118178.1118215.

Wong, Wucius (1993). Fundamentos del diseño. España. Editorial Gustavo Gili.