

DOI: <https://doi.org/10.46296/ig.v8i15.0247>

PROPUESTA DE UNA METODOLOGÍA DE DESARROLLO DE VIDEOJUEGOS PARA EQUIPOS INDEPENDIENTES

PROPOSAL FOR A VIDEO GAME DEVELOPMENT METHODOLOGY FOR INDEPENDENT TEAMS

Loor-Cobeña Luis Enrique ¹; Alcívar-Cevallos Roberth Abel ²

¹ Universidad Técnica de Manabí. Facultad de Ciencias Informáticas. Portoviejo, Ecuador.
Correo: luisenrique49@live.com. ORCID ID: <https://orcid.org/0009-0007-8407-5235>

² Universidad Técnica de Manabí. Facultad de Ciencias Informáticas. Portoviejo, Ecuador.
Correo: roberth.alcivar@utm.edu.ec. ORCID ID: <https://orcid.org/0000-0001-6282-8493>

Resumen

Este trabajo presenta una solución a la falta de metodologías de desarrollo disponibles para equipos independientes, que recoge características ágiles que ayudan al desarrollo rápido, la comunicación efectiva y la transparencia. El resultado final se presenta mediante Agile Puzzle, una metodología de desarrollo ágil basada en el marco de trabajo Scrum, del cual recoge elementos considerados importantes y de rápido entendimiento para desarrolladores con poca experiencia en el control de proyectos, que busquen una solución fácil de aplicar y flexible para el proceso de desarrollo. Dicha metodología destaca por su capacidad de adaptación ante problemas encontrados durante el tiempo de vida del proyecto, al tener facultad de iteración, lo que le permite al equipo de desarrollo reconocer los problemas, decidir cómo actuar ante ellos y afrontarlos con un enfoque más experimentado.

Palabras clave: Videojuegos, scrum, independiente, desarrollo, metodología.

Abstract

This work presents a solution to the lack of development methodologies available for independent teams, which includes agile features that help rapid development, effective communication and transparency. The final result is presented through Agile Puzzle, an agile development methodology based on the Scrum framework, which includes elements considered important and quick to understand for developers with little experience in project control, who are looking for an easy-to-apply solution and flexible for the development process. This methodology stands out for its ability to adapt to problems encountered during the lifetime of the project, by having the power of iteration, which allows the development team to recognize problems, decide how to act on them and confront them with a more experienced approach.

Keywords: Video games, scrum, independent, development, methodology.

Información del manuscrito:

Fecha de recepción: 10 de diciembre de 2024.

Fecha de aceptación: 27 de febrero de 2025.

Fecha de publicación: 06 de marzo de 2025.



1. Introducción

Los desarrolladores independientes de videojuegos a menudo enfrentan una falta de opciones con respecto a una metodología formal de desarrollo, lo que resulta en proyectos que dependen en gran medida de la experiencia individual y que no siempre cumplen con sus expectativas. Para abordar esta deficiencia, se propone la implementación de una metodología ágil basada en Scrum, adaptada específicamente para equipos independientes. Esta metodología debe ser flexible para manejar cambios y errores durante el ciclo de vida del proyecto, contar con suficiente documentación para registrar todos los puntos importantes y ayudar a los desarrolladores a crear productos que cumplan con los estándares necesarios.

La metodología ágil basada en Scrum ofrece procesos iterativos, retroalimentación continua y adaptabilidad, elementos cruciales para equipos con recursos limitados. Al adoptar Scrum, los desarrolladores pueden gestionar los cambios con agilidad, priorizar tareas de manera efectiva y fomentar

una comunicación transparente. Además, la personalización de Scrum permite mantener la creatividad del equipo mientras se sigue un marco disciplinado. Con esta metodología, no solo se busca cumplir con los plazos, sino también mejorar la calidad e innovación de los productos, convirtiéndose en un aliado estratégico para los equipos independientes en un mercado competitivo y en constante evolución.

Estado del arte

- **Metodologías ágiles de desarrollo**

Estas tienen su base en el Manifiesto Ágil [1], que resume entre valores y principios aquellas prácticas que aportan valor al desarrollo acorde a los gustos del cliente, capacidad de adaptación y motivación del equipo [2], más la flexibilidad y capacidad de modificación [3]. La aplicación de estas metodologías ágiles se puede extender al desarrollo de cualquier tipo de proyecto o producto [4]. Otro punto importante es la búsqueda de un resultado rápido [5] y la comunicación constante con el usuario [6], lo cual permite a los desarrolladores trabajar en paralelo

con las necesidades del mismo. De estas metodologías, las que más renombre tienen son Extreme Programming (XP), Kanban y Scrum [7] [8].

- **Scrum**

Dicha metodología se basa en los procesos controlados y la experiencia mediante el empirismo [9], poniendo a disposición varios elementos que ayudan a dar forma a un proceso ágil mediante fundamentos y requisitos tales como la entrega de resultados en plazos cortos de tiempo, flexibilidad y adaptación, definición de los riesgos y mayor productividad y calidad [10].

Para extender el entendimiento de la metodología Scrum se recurre a dos personalidades que trabajaron en conjunto a otros expertos en desarrollo de software, Ken Schwaber y Jeff Sutherland, quienes en La Guía Scrum [11] aseguran que: "Scrum es un marco ligero que ayuda a las personas, equipos y organizaciones a generar valor a través de soluciones adaptables para problemas complejos.".

- **Videojuegos**

Para poder formar una definición, se recurre al análisis profundo de lo que

se entiende como videojuego entre las consideraciones de varios autores, las cuales M. Wolf y B. Perron recogen y sintetizan bajo cuatro elementos presentes dentro del espectro cambiante que es la definición de un videojuego: algoritmo, interfaz, gráficos y la actividad del jugador [12].

Entendiendo cada uno de estos elementos como:

Algoritmo: Programa o compilación de líneas de código con una secuencia lógica, que contiene todos los elementos e instrucciones que permiten ejecución del videojuego.

Interfaz: Todo elemento visual que muestra al jugador información sobre los elementos que observamos en pantalla, y que sirve al jugador como herramienta para controlar varios elementos del videojuego.

Gráficos: Método de representación mediante píxeles que retrata todo elemento dentro del videojuego.

Actividad del jugador: Todo movimiento o acción que el jugador debe llevar a cabo para interactuar con el videojuego.

A consideración personal, entre los cuatro elementos listados

anteriormente, no se incluye a dos elementos importantes a la hora de llevar a cabo la acción de interactuar con un videojuego, que son:

Método de contención: Dispositivo o tecnología que contienen la información digital del videojuego y que permite la obtención o lectura del mismo para poder transmitirlo a través de un medio externo. Véase como método de contención discos compactos, la nube, disco duro, entre otros.

Medio de transmisión: Todo dispositivo o tecnología que hace posible la transmisión (visual o auditiva) del videojuego para que el usuario sepa cómo interactuar con el mismo.

Uniendo las piezas mencionadas, se puede definir a un videojuego como todo programa electrónico, independiente de su medio de contención o de transmisión, creado mediante la unión de líneas de código lógicas y ordenadas, que implica una interacción temporal o permanente con el usuario; que proporciona, mediante la interfaz, información importante hacia el jugador, y que se representa mediante algún método gráfico.

No se puede ignorar la relevancia de estos productos en la actualidad, que gracias a numerosos y novedosos avances tecnológicos, han tenido una gran cabida alcance cada vez mayor, así como recibiendo reconocimiento más esparcido dentro de otras industrias y áreas de estudio como la psicología [13], el cine [14] e incluso siendo considerados como herramienta para el aprendizaje [15]. Además, es importante mantener dentro del contexto del videojuego el componente lúdico y la inclusión de la diversión que este aporta, creando así experiencias más remarcables, como suele buscarse en campos como la psicología, para que el aprendizaje o la obtención de habilidades diversas sea más amena al usuario [16].

- **Videojuegos Independientes o “Indies”**

La definición de videojuego independiente es muy ambigua incluso dentro de la industria de la que surge, siendo además un término acuñado a principios del siglo presente [17], pero podemos recurrir a puntos de vista que pueden darnos pistas sobre lo que se desea tener claro, tomando como

referencia a la página oficial Game Maker, donde se expone que un videojuego independiente viene por parte de un desarrollador independiente y que no está asociado a ninguna empresa de desarrollo [18]. Así mismo, en la web de Toronto Film School, se menciona la ambigüedad del término "indie" [19].

Para fines de este artículo podemos definir que un videojuego independiente es todo aquel producto digital e interactivo que es desarrollado y publicado por un equipo de desarrolladores independientes.

En años recientes, el mercado independiente de videojuegos ha tenido un repunte gracias al surgimiento de numerosos proyectos con una innovación atrayente [20] [21], sumado al apoyo de compañías como Nintendo con su apartado Indie World, espacio en el que muestra desarrollos de videojuegos indie que llegarán a su plataforma, iniciado en el año 2019 [22].

- **Desarrolladores independientes**

Según lo revisado anteriormente con la definición de los videojuegos

independientes, fácilmente se puede definir a un desarrollador independiente como aquel individuo que cumple sus objetivos a través de una planificación personal y cuyos recursos surgen directamente de su capacidad de adquisición; es decir, no está contratado por ninguna empresa para crear sus productos, sino que el proceso creativo, de investigación, desarrollo y financiación viene de sí mismo. Dicha definición sigue los lineamientos que se exponen la página Kevuru Games [23].

Desarrolladores de este tipo cada vez tienen mayor apoyo que encontrar en la web, dándose a notar Pro Indie Dev [24], una iniciativa independiente para apoyar al conocimiento y formación independiente.

- **Desarrollo de videojuegos**

Proceso mediante el cual un desarrollador o un equipo de desarrolladores trabajan con un objetivo en común en varias áreas como: diseño visual [25], programación [26], sonido [27], entre otros. Este considera también puntos como la edad de los usuarios a los que va dirigido un videojuego [28], la

accesibilidad [29], plataformas (consolas de sobremesa, PC, consolas portátiles), el marketing [30] o el género (FPS, Terror, Mundo abierto), puntos clave que pueden ser cruciales para considerar el éxito del título en desarrollo.

- **Metodologías de desarrollo para equipos independientes**

Anteriores trabajos demuestran el interés por fusionar las propuestas de una metodología de desarrollo con las mejores prácticas del marco de trabajo SCRUM, dando como resultado trabajos interesantes con los que un desarrollador independiente puede encontrarse y basar su trabajo en dichas metodologías.

Dos ejemplares de estos casos son SUM y Huddle [31] [32], de las cuales ambas tienen la característica de aprovechar elementos presentes en el marco de trabajo Scrum, como las herramientas, procesos o roles, que dan forma a trabajos bastante válidos y que sin duda pueden aportar valor a un posible desarrollo.

2. Propuesta

Metodología Agile Puzzle

La metodología se ha denominado como Agile Puzzle recogiendo dos puntos importantes que involucra la misma:

1. Primero, la palabra Agile viene dada por contar con procesos y herramientas ágiles que provean de flexibilidad al desempeño de la misma
2. Segundo, la palabra Puzzle, por su traducción al español, rompecabezas, refleja la naturaleza de la metodología al poder ser empleada como un rompecabezas donde las piezas pueden encajarse en el orden en que más convenga al desarrollador, sin importar la fase en la que se encuentre

Agile Puzzle propone su agilidad al disponer de herramientas que aportan valor a la adaptabilidad ante dificultades que puedan surgir durante el desarrollo del videojuego, además, las tareas propuestas se podrán explorar en el orden que se decida según el desarrollo lo requiera, ya que estas tienen un enfoque iterativo y superpuesto [33].

Esto aporta una característica de valor ante el desenvolvimiento del proyecto, el cual podrá ser abarcado de distintas maneras según se requiera, teniendo la facilidad no solo de volver a explorar tareas que aporten al incremento de las características, sino que también podrán abarcar actividades que corresponden a una fase distinta a la que se encuentra el desarrollo. Es importante aclarar tres puntos:

1. Agile Puzzle no es una herramienta de gestión del proyecto, sino un conjunto de procesos y herramientas que pueden ser utilizados en un orden acorde a las necesidades del proyecto para dar forma al mismo
2. Agile Puzzle podrá ser usada en un proyecto individual o grupal (una persona o un equipo)
3. De ser utilizada Agile Puzzle en un equipo desarrollador, es recomendable que el mismo no supere un margen de 10-15 personas, esto debido a que las actividades en una metodología ágil requieren flexibilidad y agilidad, además, de una comunicación efectiva y colaboración íntegra, elementos

fundamentales a la hora de utilizar una metodología Ágil.

Agile Puzzle, además, se separa de los roles presentes en SCRUM, como el Scrum Master o el Product Owner, al contar con la naturaleza multidisciplinaria del desarrollador independiente que, si bien en su abanico de posibilidades se pueden incluir los papeles anteriormente mencionados, es preferible desapegar al desarrollador independiente de la sobrecarga de roles y solapamiento de responsabilidades. Otro punto que sustenta esta omisión es la fácil comprensión de los procesos que se busca atribuir al desarrollador independiente, logrando que se adapte a la metodología de la forma más orgánica posible, incluso siendo aplicable desde equipos con nula experiencia en procesos ágiles o la metodología SCRUM, hasta aquellos equipos con mayor experiencia y adaptabilidad.

Por último, se puede recurrir a la analogía de que la metodología es un rompecabezas cuyas piezas pueden encajar perfectamente una con la otra, dando así lugar a que la parte desarrolladora pueda escoger qué piezas utilizar en qué momento,

o cuándo prescindir de una de ellas. Las tareas no deben estar estrictamente atadas a un momento específico del desarrollo. Por ejemplificar esto, la fase de Pruebas dispone de la tarea de Pruebas Internas, estas pueden ser aplicadas incluso desde fases iniciales del proyecto, sirviendo como un filtro para que los errores encontrados, incluso desde el inicio, puedan ser corregidos y que no supongan un problema en posteriores fases.

La propuesta de Agile Puzzle presenta cuatro fases, denominadas

de la siguiente manera: Planificación, Desarrollo, Pruebas y Publicación. Las fases nombradas anteriormente contienen tareas puntuales que se corresponden a una actividad específica para la realización de cada una de estas, mismas que incluyen herramientas que prioricen la transparencia [34] y procesos que aporten flexibilidad y agilidad [4].

La siguiente tabla representa la metodología con cada una de sus tareas a desarrollar:

Tabla 1. Metodología Agile Puzzle

Componentes de la Metodología Agile Puzzle	
Fases	Tareas
Planificación	Creación del GameLog
	Creación del Backlog
	Sprint Planning
Desarrollo	Daily Scrum
	Desarrollo Continuo
	Integración Continua
	Sprint Retrospective
	Refinamiento del Backlog
Pruebas	Pruebas Internas
	Beta Testing
	Corrección de Errores
Lanzamiento	Preparación para el Lanzamiento
	Marketing y Publicidad
	Lanzamiento del Juego

NOTA: La disposición de las tareas no refleja el orden o cronología a seguir para su aplicación final.

Las tareas presentes en la Tabla 1 se describen en la Tabla 2, donde se desglosa el desarrollo de cada una de estas.

Tabla 2. Tareas de Agile Puzzle

Creación del GameLog	Se establece una descripción global del juego, su público objetivo y las principales características desglosadas según el equipo lo requiera; al finalizar se recopilará toda la información en un documento llamado GameLog. Este debe ofrecer una guía clara para el equipo, y puede variar en cuanto a su contenido durante el desarrollo del videojuego, aunque su modificación no es recomendable, a menos que sea estrictamente necesario, ya que puede generar conflictos de entendimiento y visión en caso de haber más de un desarrollador involucrado. Su creación puede tomar más de una reunión según sea necesario, antes de comenzar con el desarrollo del proyecto.
Creación del Backlog	Este es una lista priorizada de todas las características, mejoras, correcciones de errores, requisitos técnicos y cualquier otra tarea que deba realizarse para completar el videojuego, a manera de historias de usuario. Es dinámico y estará en constante evolución a medida que el proyecto avance y se obtenga más feedback.
Sprint Planning	El equipo selecciona los ítems del Backlog que se compromete a completar durante el próximo Sprint, según la prioridad de los ítems y la capacidad del equipo. Durante esta reunión, el equipo descompone los ítems seleccionados en tareas más pequeñas y manejables, asegurándose de que cada tarea esté claramente definida y comprendida. Se establece un objetivo claro y alcanzable al final del Sprint.
Daily Scrum	Es una reunión diaria y breve, para fomentar la agilidad y la comunicación efectiva en los procesos de desarrollo. Durante esta reunión, cada miembro del equipo responde a tres preguntas clave: qué hizo ayer, qué planea hacer hoy y si hay algún impedimento en su camino. Esta reunión, de no más de 15 minutos, permite al equipo sincronizarse, identificar y abordar rápidamente los problemas y ajustar sus planes diarios para maximizar la productividad.
Desarrollo Continuo	Esta tarea es el núcleo del proyecto y dará como resultado todo trabajo que cumpla parcial o completamente los objetivos planteados para el Sprint. Se organiza en Sprints, que son intervalos de tiempo fijos, generalmente de dos a cuatro semanas (duración recomendada para establecer un deber de cumplimiento y mantener el principio de agilidad), durante los cuales el equipo de desarrollo se enfoca en completar un conjunto de ítems del Backlog.
Integración Continua	Práctica técnica que implica integrar el trabajo del equipo de desarrollo con frecuencia. Ayuda a encontrar y corregir problemas rápidamente, mejora la colaboración entre los desarrolladores y asegura que el producto siempre esté en un estado funcional y potencialmente entregable.
Sprint Retrospective	Tras cada Sprint, el equipo debe indagar en lo que funcionó bien, los problemas encontrados y cómo mejorar el desarrollo para el próximo Sprint, además de considerar medidas y prácticas para mejorar la calidad y satisfacción del desarrollo. Incluye la revisión de la Documentación Continua por parte de cada miembro.
Refinamiento del Backlog	Al final del Sprint se evalúan las historias de usuario, se aclaran detalles y requisitos de cada una de estas para generar una nueva prioridad en función de su valor y la complejidad técnica. El refinamiento continuo ayuda a mantener el flujo de trabajo del equipo, permitiendo que se enfoquen en las tareas más importantes y estén mejor preparados para las reuniones de planificación del Sprint.
Pruebas Internas	Durante las pruebas internas, los desarrolladores y testers del equipo ejecutan pruebas de usabilidad y rendimiento para verificar que cada componente del videojuego funcione correctamente e interactúen adecuadamente entre sí y detectar problemas, facilitando su identificación y corrección posterior a pruebas con usuarios externos.
Beta Testing	Los usuarios externos que participen de esta prueba serán denominados Beta Testers, y forman parte del público objetivo que proporcionará feedback sobre

	su experiencia de uso ante una versión preliminar del juego. Esta tarea busca nuevamente identificar errores que se hayan escapado en las Pruebas Internas para poder corregirlos posteriormente.
Corrección de Errores	Tarea continua que busca dar solución a los problemas encontrados principalmente en las Pruebas Internas y Beta Testing, y se basa en el feedback recibido. Es necesario que estas pruebas sean repetidas hasta verificar la mitigación de los errores encontrados, así como verificar que la corrección de errores anteriores no genere nuevos problemas.
Preparación para el Lanzamiento	Se genera la documentación necesaria para el usuario final, como manuales de usuario, guías de instalación, y notas de la versión que detallen las características y cambios del juego. Implica la verificación del funcionamiento correcto de aspectos técnicos, como servidores en línea para juegos cooperativos, sistemas de pago o mecanismos de actualización, aspectos que variarán según el videojuego desarrollado.
Marketing y Publicidad	Tarea fundamental para que el videojuego genere un alcance a una audiencia amplia y generar interés por el mismo, mediante una estrategia de publicidad que puede incluir la generación de videos tutoriales, tráilers, capturas de pantalla, y demás material publicitario.
Lanzamiento del Juego	Tarea culminante del desarrollo. Implica la finalización del videojuego acorde a los requerimientos del mismo, listo para estar a disposición del público en la o las plataformas y medios anteriormente seleccionados. Tras el lanzamiento, el equipo de desarrollo está preparado para abordar cualquier problema que pueda surgir, como errores no detectados previamente, problemas de rendimiento o dificultades con el sistema de distribución, considerando el feedback de los jugadores y las métricas de rendimiento del juego para aplicar mejoras continuas.

Documentos

GameLog: Documento clave que contiene toda la información primordial del videojuego a desarrollar, yendo desde el nombre del juego, género o plataformas, hasta aspectos más técnicos como las especificaciones requeridas para que el videojuego pueda ejecutarse de manera correcta. En la siguiente

plantilla se presentan dos columnas, Sección y Descripción, cada una con los elementos recomendados a tomar en cuenta al momento de armar dicho instrumento. El mismo trata de crear una visión del desarrollo transparente, que debe ser accesible por cualquier miembro del equipo de desarrollo. La Tabla 3 propone un esquema del documento en cuestión:

Tabla 3. Esquema del GameLog

Sección	Descripción
Visión General del Juego	Descripción general, género del juego, público objetivo, plataforma(s) objetivo.
Historia y Narrativa	Resumen de la trama, descripción de personajes principales, contexto y ambientación del juego.
Características Principales	Jugabilidad y mecánicas básicas, modos de juego, innovaciones o características únicas.

Arte y Estilo Visual	Estilo artístico, descripción visual de personajes y entornos, referencias visuales.
Sonido y Música	Estilo musical y ambiente sonoro, tipos de efectos de sonido, música de fondo y temas principales.
Interfaz de Usuario (UI) y UX	Diseño de la interfaz de usuario, flujo de menús y navegación, experiencia de usuario y accesibilidad.
Mecánicas de Juego Detalladas	Sistema de control, mecánicas de combate, mecánicas de puzzles o desafíos.
Sistemas del Juego	Progresión y niveles, sistema de recompensas y logros, economía del juego.
Inteligencia Artificial (IA)	Comportamiento de enemigos y NPCs, sistemas de IA para aliados, algoritmos de decisión.
Multijugador y Redes (si aplica)	Tipos de modos multijugador, sistema de matchmaking y servidores, comunicación y socialización en el juego.
Requisitos Técnicos	Motores de juego y herramientas a utilizar, requisitos mínimos y recomendados de hardware, compatibilidad y plataformas.
Plan de Desarrollo y Cronograma	Fases del desarrollo, hitos y entregables claves, cronograma estimado.
Marketing y Estrategia de Publicación	Estrategia de marketing y promoción, plan de lanzamiento, estrategia de post-lanzamiento y actualizaciones.

NOTA: Los elementos de la columna Descripción variarán según los aspectos a considerar en cada punto de la columna Sección, puesto que dependerá de la visión de cada proyecto y de cómo el o los desarrolladores quieran construir su videojuego.

Backlog: Documenta las historias de usuario, que son descripciones breves y simples de las funcionalidades desde la perspectiva de los distintos actores involucrados en el desarrollo. Se compone de un Actor (Usuario Final o Desarrollador), un Requerimiento que especifica la necesidad, un Prioridad (Alta, Media o Baja) y un Estado (Por Iniciar, En Curso, En Pausa, Finalizado, Atrasado). Esta disposición proporciona una herramienta fácil de entender y

utilizar durante las fases de Sprint Planning y Refinamiento del Backlog. La forma de leer cada historia de usuario será: Como (ACTOR) quiero (REQUERIMIENTO) para (NECESIDAD). Las columnas PRIORIDAD y ESTADO son estimaciones de qué tan importante es la historia de usuario y en qué estado de desarrollo se encuentra. La Tabla 4 propone un esquema del documento en cuestión:

Tabla 4. Esquema del Backlog

Actor	Requerimiento	Necesidad	Prioridad	Estado
Usuario final	Tener información sobre la munición disponible en mi arma	Saber cuánto puedo disparar y saber cuándo recargar	Alta	En Curso
Desarrollador	Crear una interfaz con estilo 3D	Representar los elementos correctamente	Media	Por Iniciar

Documentación Continua:

Es un proceso clave para garantizar la transparencia y la comunicación efectiva durante el desarrollo. Cada miembro del equipo debe mantener un registro detallado en un cuadro que incluye cinco columnas: "Barrera o Problema encontrado" para identificar los obstáculos que impiden el progreso; "Proceso Afectado" para señalar el proceso o tarea impactada; "Resolución o Mitigación" para documentar las soluciones o medidas de mitigación

propuestas; "Prioridad de Resolución" para clasificar la urgencia del problema (Alta, Media, Baja); y "Estado de Resolución" para actualizar el avance hacia la solución (Sin Resolver, En Proceso, Resuelto, Mitigado). Además, se incluye una sección para "Notas Personales y Fecha" donde cada desarrollador puede agregar comentarios relevantes, ya sea para consideración personal o del equipo en cuestión. La Tabla 5 propone un esquema del documento en cuestión:

Tabla 5. Esquema de la Documentación Continua

Barrera o Problema	Tarea afectada	Resolución o Mitigación	Prioridad	Estado de resolución
La licencia de la herramienta de modelos 3D está vencida	Creación de modelos 3D	Pagar la licencia de uso.	Alta	En Proceso
La animación de salto no se ejecuta	Animación	Revisión de animaciones o procesos en conflicto	Media	Sin Resolver
La creación de X elemento es inviable	Creación de escenas	Utilizar las herramientas disponibles para crear una solución viable	Media	Mitigado
Notas Personales:	Agregué una nota en el código del archivo Character_Moves, línea 3.		Fecha:	01/07/2024

Representación visual de la metodología Agile Puzzle

La siguiente representación de Agile Puzzle propone un escenario que sigue la secuencia revisada en la

Tabla 1, mas no tiene que ser la aplicación final recomendada para la metodología, puesto que implicaría que el desarrollo no ha tenido ningún problema en sus tareas ni ha enfrentado pruebas repetidas y

correcciones recurrentes. Se representa lo que se podría decir como un "trabajo perfecto" a través de Bizagi Modeler, una herramienta

gratuita de modelado de procesos. La Figura 1 y la Figura 2 muestran dicha representación.

Figura 1. Representación visual de las fases Planificación y Desarrollo

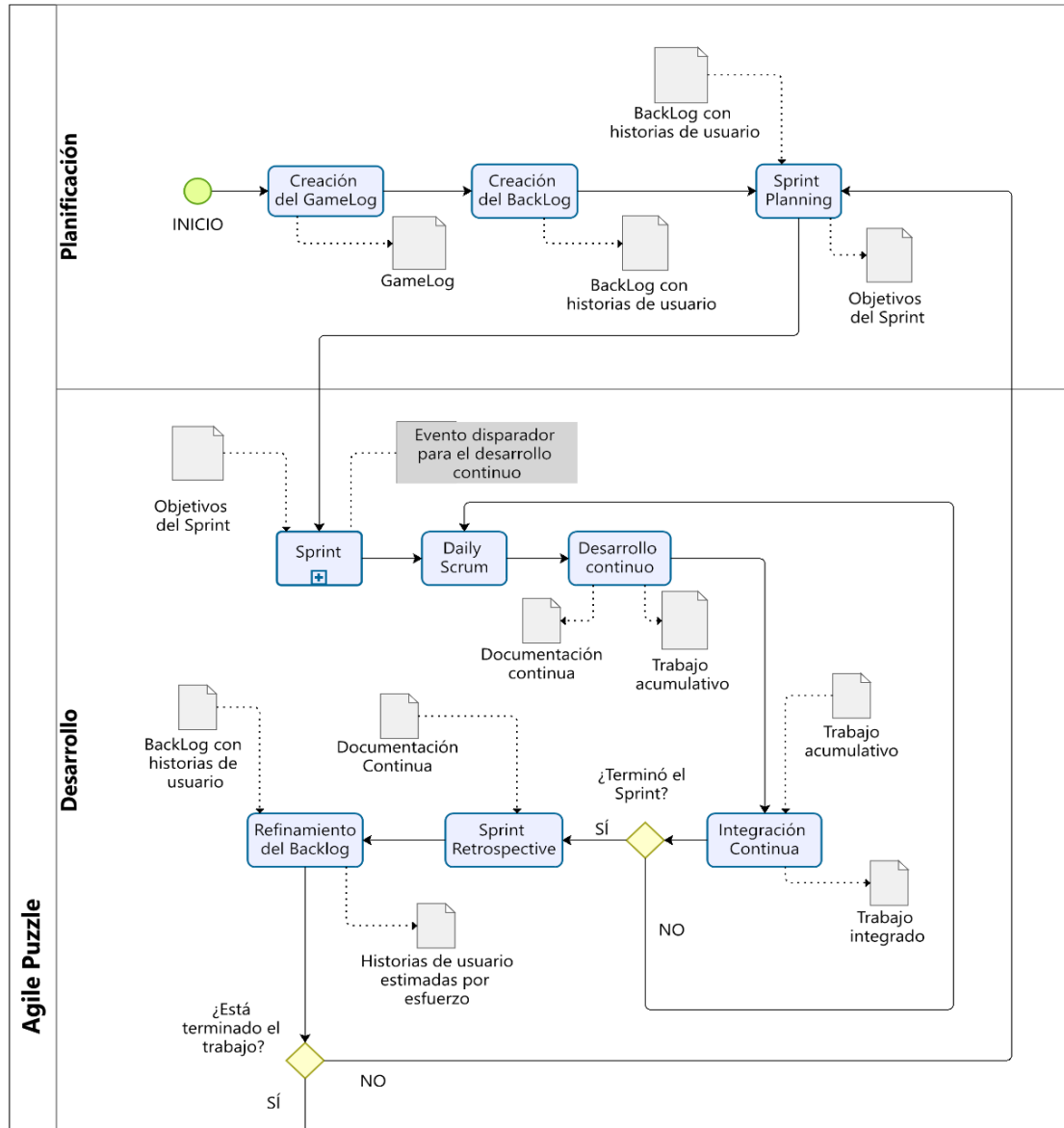
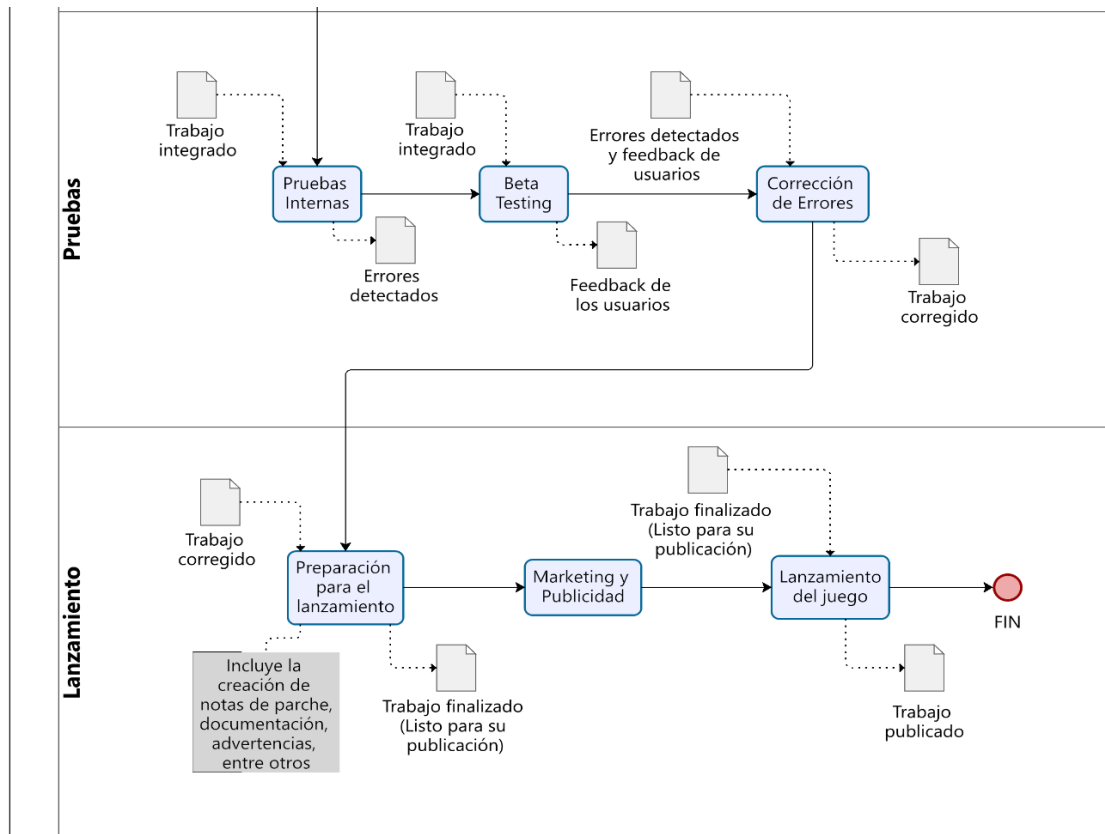
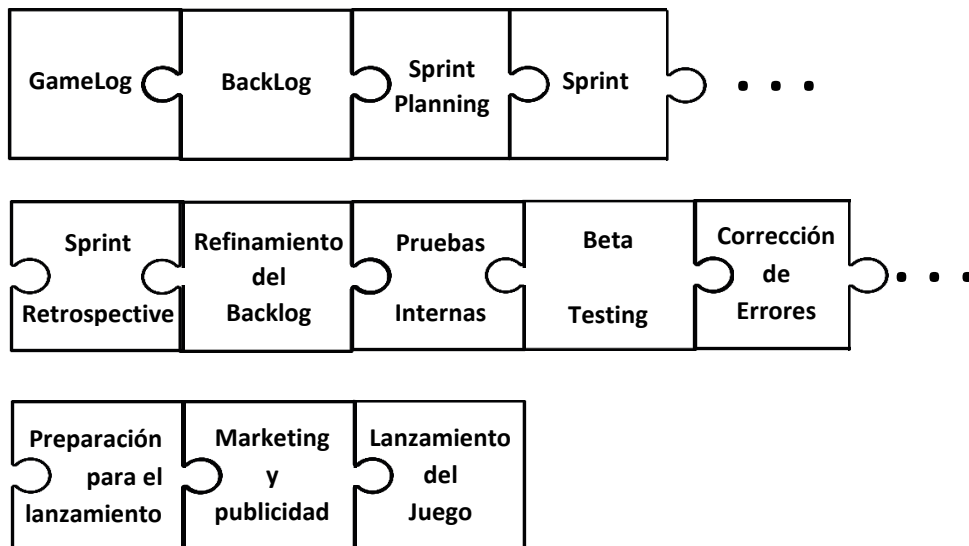


Figura 2. Representación visual de la fase Pruebas



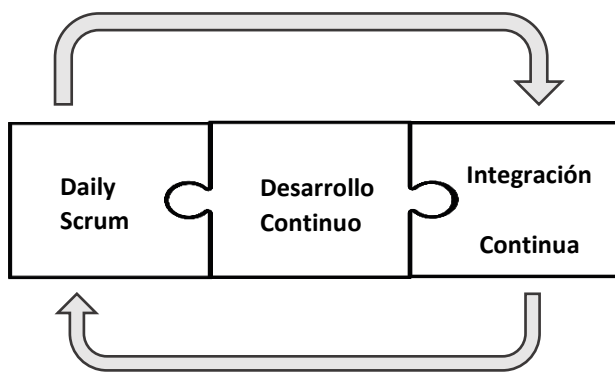
La Figura 3 muestra la representación a través de la analogía del rompecabezas:

Figura 3. Representación en Puzzle



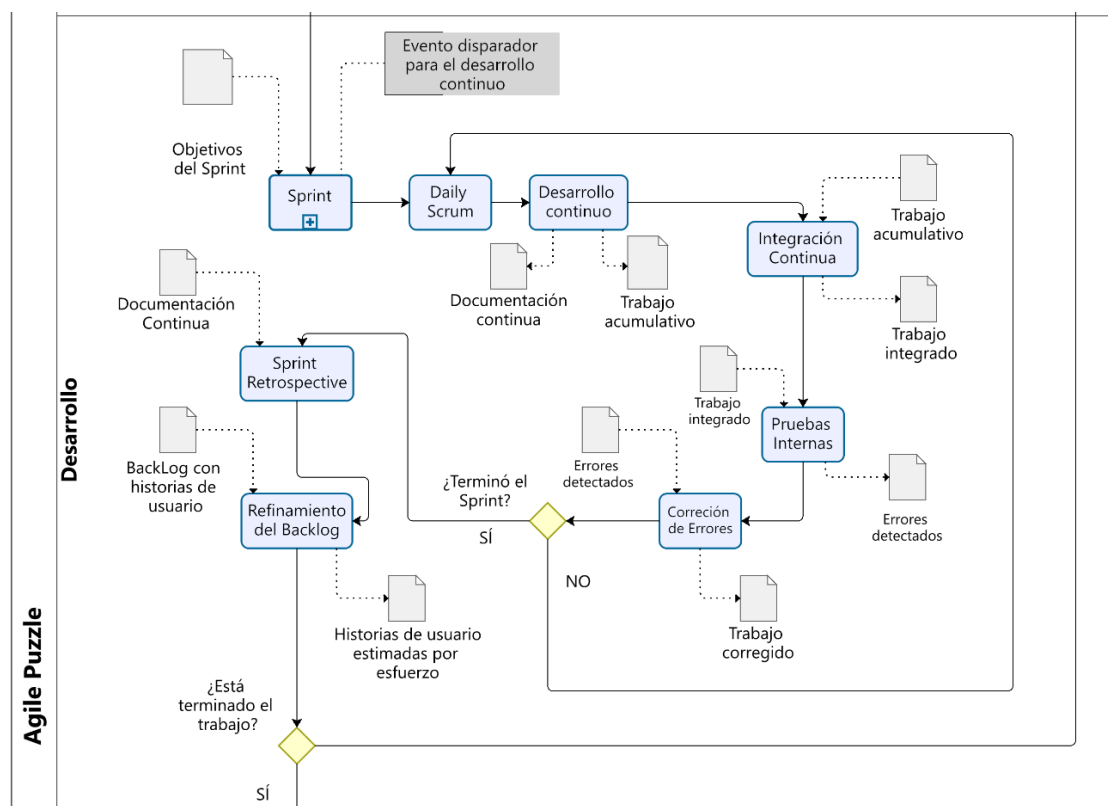
Y el Sprint de esa planificación a manera de Puzzle se muestra en la Figura 4:

Figura 4. Representación en Puzzle del Sprint 1



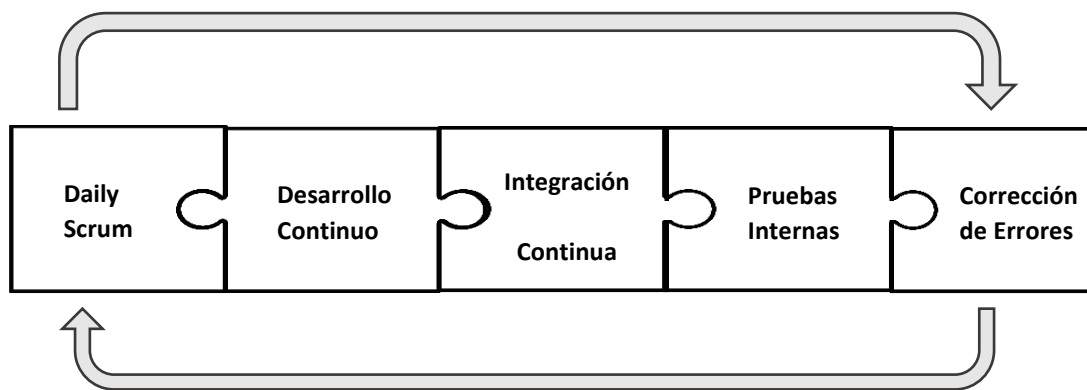
Otra manera de afrontar un Sprint sería la demostrada en la Figura 5:

Figura 5. Representación visual del Sprint 2



Donde la planificación del Sprint a manera de Puzzle sería la que propone la Figura 6:

Figura 6. Representación en Puzzle del Sprint 2



En esta representación de un nuevo Sprint se aplican pruebas y correcciones al trabajo realizado de manera periódica, sirviendo como filtro para eliminar errores que podrían acumularse en la fase de pruebas. Esta manera de trabajar puede atender a un modelo más rápido de desarrollo adoptado por el equipo. Decisiones como el cambio de configuración de un Sprint pueden tomarse incluso en la marcha, siendo que en el Sprint Planning se prevé incluir Pruebas Internas para eliminar futuros problemas en la fase de pruebas o el Beta Testing. Este ejemplo demuestra que las posibilidades de

las tareas flexibles son numerosas y que la planificación del proyecto puede modificarse según se necesite a través del ciclo de vida del mismo.

3. Resultados y discusión

Para efectos de evaluar Agile Puzzle, se decidió recurrir a una visión experimentada del espectro de desarrollo de software independiente, aplicando una encuesta que evalúa la usabilidad por parte de desarrolladores con respecto a los procesos y herramientas que ofrece Agile Puzzle. Sus resultados se muestran en la Tabla 6:

Tabla 6. Resultados de la encuesta

Claridad del GameLog		Utilidad del GameLog	
En escala del 1 al 5, ¿cómo calificarías la claridad y la organización del documento GameLog?	4,2	En escala del 1 al 5, ¿cuán útil consideras que es el GameLog para comprender el proyecto en general?	4,8
Utilidad de la Documentación Continua		Identificación y superación de barreras de desarrollo	

En escala del 1 al 5, ¿qué tan útil te parece la propuesta para Documentación Continua?		4,4	En escala del 1 al 5, ¿qué tan efectivo crees que es el enfoque propuesto para identificar y mitigar barreras en el desarrollo?		3,6
Pruebas y retroalimentación del usuario final		General			
En escala del 1 al 5, ¿qué importancia tiene el Beta Testing en la calidad del juego?	5	En escala del 1 al 5, ¿cuán probable es que recomiendes esta metodología a otros desarrolladores de videojuegos?	3,8	En escala del 1 al 5, ¿cuán probable es que utilices esta metodología en un proceso de desarrollo de un videojuego?	3,8

Los resultados promediados, a partir de desarrolladores independientes destacan a Agile Puzzle como una herramienta viable para el desarrollo de videojuegos, con herramientas útiles y una capacidad considerable para enfrentar barreras. Se puede notar que los encuestados muestran un entendimiento del documento GameLog, así como del BackLog, que son herramientas medulares para iniciar un desarrollo de videojuegos. Los desarrolladores valoran la Documentación Continua, aunque la identificación y superación de barreras es lo menos considerado para estos. Las Pruebas son las mejor evaluadas, y la metodología es vista positivamente para su uso y recomendación en cuanto a otros desarrollos de videojuegos independientes.

4. Conclusiones

La metodología propuesta en este artículo se estructura en cuatro fases que se desarrollan de manera cohesiva y comprensible, proporcionando a los desarrolladores un conjunto de herramientas y técnicas que fomentan la agilidad y adaptabilidad. Agile Puzzle es una metodología flexible que aporta valor desde las primeras etapas del desarrollo, permitiendo a los desarrolladores ajustar y reconfigurar actividades en cada fase según las necesidades del proyecto, superando la rigidez de la iteración en SCRUM. Agile Puzzle destaca con una adaptabilidad alta, permitiendo la planificación y modificación del desarrollo en cualquier fase, desde el inicio hasta las etapas finales, sin la rigidez característica de los ciclos iterativos

de SCRUM. Dicha flexibilidad da paso a la integración de actividades, como las pruebas de contenido durante un Sprint, lo que, aunque no esté detallado en la Tabla 1, es esencial para el desempeño eficaz de la metodología. Este enfoque flexible y adaptable constituye la base de la visión final de Agile Puzzle, asegurando que se pueda ajustar dinámicamente a las necesidades del proyecto.

Adicionalmente, en comparación con el marco de trabajo SCRUM, o de la metodología SUM [31], Agile Puzzle incorpora elementos de fácil comprensión y aplicación, eliminando la necesidad de un conocimiento profundo de SCRUM, donde se omiten roles específicos como el Scrum Master o el Product Owner, entendiendo que el desarrollador independiente actúa como un agente con recursos y conocimientos limitados que busca un proceso de desarrollo ágil y con una carga operativa mínima. La disposición limitada de recursos como el personal adecuado y capacitado o los conocimientos suficientes no supone un problema, si se compara con el marco de trabajo en el que se basa esta

metodología, pues recoge ciertos elementos básicos pero útiles para poder afrontar las distintas fases de desarrollo. Esto sin dejar atrás la transparencia y la efectividad de la comunicación, principios fundamentales que, cuando se utilizan adecuadamente, pueden facilitar la superación de barreras de entendimiento y conocimiento dentro de un equipo de desarrollo.

Además de la mencionada anteriormente SUM, también se debe considerar la propuesta encontrada bajo el nombre Huddle [32]. Con una visión muy parecida a Agile Puzzle, Huddle se presenta como una metodología bastante extensa en cuanto a la documentación del proyecto que, a comparación de la propuesta de este artículo, abarca mucho más en cuanto a la información reportada del proyecto. Es por esto último que Agile Puzzle se diferencia y destaca por la presencia de una propuesta de documentación más ligera y ágil, que no se muestra de manera abrumadora y extensa para el equipo independiente y facilitando la comunicación y entendimiento rápido de la información.

Como punto que deja en desventaja a Agile Puzzle se nota la documentación que propone, que, en comparación a otras metodologías mencionadas a través de este artículo, no es tan robusta y puede dejar a la deriva ciertos puntos a tomar en cuenta en el desarrollo del videojuego. La puesta en práctica requiere de una coordinación y comunicación alta en el equipo de desarrollo, esto es necesario para que todas las piezas encajen correctamente en su lugar y la forma final del proyecto sea la visión más cercana a las expectativas iniciales, esto puede suponer un punto decisivo para el uso de la herramienta, sobre todo en etapas tempranas del proyecto.

Adicional a esto, la metodología se presenta como una herramienta de desarrollo útil y de fácil acceso y entendimiento, misma que puede ser considerada para futuros trabajos prácticos donde se pueda identificar mediante un análisis empírico sus verdaderas ventajas y desventajas con respecto al equipo de desarrollo.

Agradecimientos

Este artículo lleva un arduo trabajo y dedicación tras de sí, elementos que no habría podido tener a mi completa

disposición si no hubiera sido por el apoyo constante de un gran número de personas que me han acompañado a lo largo de días y noches de investigación. En primer lugar, quiero mencionar a mis familiares más cercanos, mi padre, mi madre, mis hermanas y mi tía, quienes han sido un gran apoyo durante este largo proceso. Mi novia, quien me ha apoyado en tantos momentos en que el trabajo se notaba interminablemente agotador. Mis amigos universitarios, con quienes más tiempo compartí cursando las materias que dieron forma a mi perfil profesional, María, Michael, Leandro, Mateo, Bryan. Así mismo, mis amigos de la juventud, quienes a través de experiencias personales han sido un ejemplo claro de cómo todos tenemos nuestro propios deseos o metas, y que para cumplirlas debemos esforzarnos, a la par de llevar siempre una actitud positiva, ante todo: Ariel, Andrés, Luis, Yadin y Johan. En adición, quisiera dar una mención especial a todos mis maestros, profesionales que han desempeñado su papel correctamente para poder brindarme el conocimiento necesario y así poder ser el profesional que aspiro. Así mismo, a mi tutor Roberth

Alcívar, maestro y guía en la realización del presente trabajo.

Bibliografía

- [1] E. Herrera y L. Valencia, «Del manifiesto ágil sus valores y principios.,» Scientia et Technica., vol. 13, nº 34, pp. 381-386, Mayo 2007.
- [2] F. López, «Prácticas y herramientas para el desarrollo ágil de software seguro,» Universidad de Alcalá, p. 112, 2023.
- [3] A. Navarro, J. Morales y J. Fernández, «Revisión de metodologías ágiles para el desarrollo de software,» Prospectiva, vol. 11, nº 2, pp. 30-39, 2013.
- [4] E. Gómez, M. Marcillo y N. Ramírez, «Metodologías ágiles para el desarrollo de proyectos,» p. 9, 2021.
- [5] I. Morales, «Metodologías de Desarrollo Software. ¿Tradicional o Ágil?,» MoleQla: revista de Ciencias de la Universidad Pablo de Olavide, nº 19, p. 7, 2015.
- [6] B. Molina, H. Vite y J. Dávila, «Metodologías ágiles frente a las tradicionales en el proceso de desarrollo de software,» Espirales, vol. 2, nº 17, pp. 114-121, 2018.
- [7] E. Bautista-Villegas, «Metodologías ágiles XP y Scrum, empleadas para el desarrollo de páginas web, bajo MVC, con lenguaje PHP y framework Laravel,» Revista Amazonia Digital, vol. 1, nº 1, pp. e168-e168, 2022.
- [8] S. Gómez, «Aplicación de las Metodologías Ágiles al proceso de enseñanza-aprendizaje universitario,» Revista d'Innovació Docent Universitària, nº 12, pp. 62-73, 2020.
- [9] L. Straccia y M. Pollo-Cattaneo, «Metodología para el desarrollo de software en proyectos de I+D en el nivel universitario basada en Scrum,» Revista de la Red de Universidades con Carreras en Informática (RedUNCI), vol. 11, nº 1, pp. 535-545, 2016.
- [10] N. Tymkiw, J. Bournissen y M. Tumino, «SCRUM como Herramienta Metodológica para el Aprendizaje de la Programación,» Revista Iberoamericana de Tecnología en Educación y Educación en Tecnología, nº 26, pp. 81-89, 2020.
- [11] K. Schwaber y J. Sutherland, «La Guía Scrum,» Scrumguides. Org, vol. 1, p. 21, 2013.
- [12] M. Wolf y B. Perron, «Introducción a la teoría del

- videojuego,» *Formats: revista de comunicación audiovisual*, nº 4, 2005.
- [13] A. Garcia-Naveira, M. Jiménez, B. Teruel y A. Suárez, «Beneficios cognitivos, psicológicos y personales del uso de los videojuegos y esports: una revisión,» *Revista de Psicología Aplicada al Deporte y al Ejercicio Físico*, vol. 3, nº 2, pp. 1-14, 2018.
- [14] Á. Delgado y F. Granados, «Videojuegos trasladados al cine: análisis y caracterización de la narrativa audiovisual en ambos medios,» *Comunicación: revista Internacional de Comunicación Audiovisual, Publicidad y Estudios Culturales*, vol. 1, nº 10, pp. 63-77, 2012.
- [15] A. Roncancio, M. Ortiz, H. Llano, M. Malpica y J. Bocanegra, «El uso de los videojuegos como herramienta didáctica para mejorar la enseñanza-aprendizaje: una revisión del estado del tema,» *Ingeniería, Investigación y Desarrollo*, vol. 17, nº 2, pp. 36-46, 2017.
- [16] J. Pardo y N. Vela, *Revisión sistemática: Uso de los videojuegos en psicología, como herramienta terapéutica en el fortalecimiento de habilidades cognitivas*, Repositorio de Trabajos de Grado, 2022.
- [17] K. Karthikeyan, «Gameopedia,» *Gameopedia*, 17 Enero 2021. [En línea]. Available: <https://www.gameopedia.com/indie-games-everything-you-need-to-know/>.
- [18] R. Bramble, «Game Maker,» *Game Maker*, 14 Agosto 2023. [En línea]. Available: <https://gamemaker.io/es/blog/what-are-indie-games#earn-from-indie-games>.
- [19] N. Pajkovic, «Toronto Film School,» *Toronto Film School*, 25 Abril 2023. [En línea]. Available: <https://www.torontofilmschool.ca/blog/what-is-an-indie-game/>.
- [20] J. Herd, «Forbes,» *Forbes Agency Council*, 21 Agosto 2023. [En línea]. Available: <https://www.forbes.com/councils/forbesagencycouncil/2023/08/21/the-global-surge-of-independent-games-development-studios/>.
- [21] R. Benavent, «El videojuego independiente o "indie games" made in Spain,» *Con A de animación*, nº 9, pp. 42-51, 2019.
- [22] J. García, «IGN,» *IGN*, 19 Agosto 2019. [En línea]. Available: <https://es.ign.com/nintendo->

- switch/154162/feature/todas-las-novedades-del-nintendo-indie-world.
- [23] Kevuru Games, «Kevuru Games» Kevuru Games, 9 Marzo 2023. [En línea]. Available: <https://kevrugames.com/blog/indie-game-development-the-all-you-need-guide-to-revenues-most-profitable-genres-monetization-bonus-top-10-best-indie-games-2020/#:~:text=An%20indie%20developer%20is%20someone,financial%20support%20of%20large%20companies.>
- [24] Pro Indie Dev, «Pro Indie Dev,» Pro Indie Dev, [En línea]. Available: <https://www.proindiedev.com/>
- [25] A. Navarro, «Tales From The Capital: Desarrollo Visual Y Preproducción De Un Videojuego De Aventura Y Exploración 3D,» 2021.
- [26] E. Gonjar, «Desarrollo de un videojuego Tower Defense usando Unity3D y una API para la IA en lenguaje C#,» Universitat Politècnica de València, 2019.
- [27] J. Micó, «Producción del videojuego Part of Nick: gráfica 3D, diseño de sonido y demo Reel,» Universitat Politècnica de València, 2021.
- [28] J. del Río, «Evolución del desarrollo de videojuegos,» MoleQla: revista de Ciencias de la Universidad Pablo de Olavide, nº 36, p. 2, 2020.
- [29] I. Jiménez, «Análisis De La Accesibilidad En Videojuegos Comerciales: Xenoblade Chronicles 2,» 2023.
- [30] J. Parreño, Marketing y videojuegos, Madrid: ESIC Editorial, 2010.
- [31] N. Acerenza, A. Coppes, G. Mesa, A. Viera, E. Fernández, T. Laurenzo y D. Vallespir, «Una Metodología para Desarrollo de Videojuegos,» Mar del Plata, 2009.
- [32] G. Morales, C. Nava, L. Fernández y M. Rey, «Procesos de desarrollo para videojuegos,» Cultura Científica Y Tecnológica, vol. 7, nº 36, pp. 25-39, 2010.
- [33] Project Management Institute, «Project Management Body of Knowledge,» Project Management Institute, 1996.
- [34] P. Cababie y F. Troilo, «Metodologías ágiles en equipos de operaciones del área de tecnología de la información (TI),» Serie de Documentos de Trabajo, nº 783, 2021.