



UNIVERSIDAD DEL ISTMO

**Prototipo didáctico de un videojuego para el aprendizaje de
estructuras secuenciales, selectivas e iterativas orientado a niños**

**PROTOCOLO DE TESIS QUE PARA OBTENER EL TÍTULO DE
INGENIERO EN COMPUTACIÓN**

Presenta

Carlos Esteban Sánchez Jiménez

Directora de tesis

M.A.G.N. Silvia Reyes Jiménez

Santo Domingo Tehuantepec, Oaxaca, noviembre 2021

Índice

1.	Generalidades de la investigación.....	3
1.1.	Antecedentes.....	3
1.2.	Planteamiento del problema	5
1.3.	Justificación	6
1.4.	Objetivos.....	7
1.4.1.	Objetivo general	7
1.4.2.	Objetivos específicos.....	7
1.5.	Alcances.....	7
1.6.	Limitaciones	7
1.7.	Trabajos relacionados	8
1.8.	Metodología.....	12
2.	Marco teórico	14
2.1.	Videojuego.....	14
2.1.1.	Tipos de videojuegos.....	15
2.1.2.	Videojuego educativo.....	16
2.1.3.	Motor de videojuego	16
2.2.	Estructuras de control	17
2.2.1.	Estructura secuencial.....	18
2.2.2.	Estructura selectiva.....	18
2.2.3.	Estructura iterativa	18
2.3.	Pensamiento computacional	18
2.4.	Metodología de desarrollo de videojuegos SUM	19
2.4.1.	Concepto.....	20
2.4.2.	Planificación.....	21
2.4.3.	Elaboración.....	21
2.4.4.	Beta.....	22
2.4.5.	Cierre	22
2.4.6.	Gestión de riesgos	22
2.5.	Modelo conceptual para el diseño de videojuegos educativos	22
2.5.1.	Mecánica	23
2.5.2.	Retos.....	23

2.5.3.	Escenario	23
2.5.4.	Caracterización de personajes	24
2.5.5.	<i>Feedback</i>	24
2.5.6.	Socialización	24
2.5.7.	<i>Debriefing</i>	24
2.5.8.	<i>Storyline</i>	25
2.5.9.	Recompensa.....	25
2.5.10.	Persistencia.....	25
2.5.11.	Servicios	25
2.5.12.	Interfaz del juego e interacción	26
3.	Cronograma de actividades	27
4.	Referencias.....	28

1. Generalidades de la investigación

Este apartado contiene los antecedentes, el planteamiento del problema, la justificación, el objetivo general y los objetivos específicos, los alcances y limitaciones, los trabajos relacionados con la presente investigación y la metodología a seguir en el desarrollo del proyecto.

1.1. Antecedentes

La tecnología ha sido parte fundamental para el desarrollo de la sociedad, se han creado medios de comunicación que nos permiten estar conectados con el mundo de distintas maneras. Uno de estos medios son los dispositivos móviles, que también pueden ser usados para diferentes fines, siendo uno de ellos el ocio y más en específico, jugar videojuegos. Los videojuegos, en palabras de Zhao (2017) “son una manifestación de las nuevas tecnologías que caracterizan a nuestra sociedad y cuyas transformaciones son consideradas por muchos autores como más profundos y con mayor impacto en nuestras vidas cotidianas de las que tuvieron lugar con la Revolución Industrial” (p. 15). La popularidad de los videojuegos para dispositivos móviles ha ido en aumento (véase la Figura 1), ya que son más accesibles que los juegos de consola o de computadoras personales, además de que se pueden jugar en cualquier sitio.

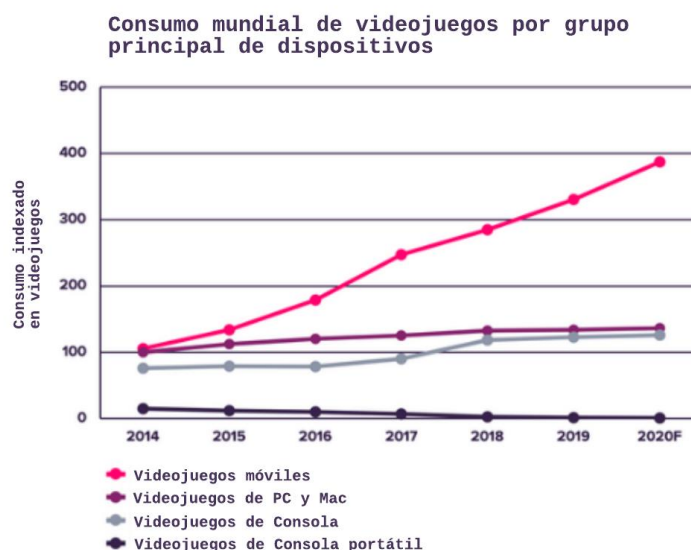


Figura 1. Consumo de videojuegos por plataforma. Fuente: App Annie & IDC, 2020, App Annie (<https://www.appannie.com/en/insights/market-data/gaming-spotlight-2020-review/>)

Durante mucho tiempo los videojuegos han sido sólo herramientas para el ocio y el entretenimiento, sin hacer distinción de edades ni de géneros, debido a esa razón se tiene una idea de que los mismos

son carentes de valor productivo y educativo, siendo sólo medios de distracción. Esta creencia ha estado siendo refutada desde los 70's por autores como Clark Abt (1970), quien plantea formas de usar los juegos de mesa y de cartas para aprender, y ya no con el único objetivo de divertirse, bajo la denominación de Juegos Serios (*Serious Games*).

Según Zyda (2005) la participación de la pedagogía (actividades para educar o instruir, impartiendo así conocimiento o habilidad) es lo que hace que los juegos sean considerados como 'serios'. La adición de características del videojuego para generar motivación y experiencias de aprendizaje inmersivas a la idea de los juegos serios, genera el concepto de los videojuegos serios, también conocidos como videojuegos educativos, y que están enfocados a enseñar mientras el jugador se divierte, haciendo uso de la gamificación (también llamado ludificación) como agente para motivar al jugador a terminar el juego.

La gamificación es un término que se utiliza para describir aquellas características de un sistema interactivo que tienen como objetivo motivar y comprometer a los usuarios finales a través del uso y la mecánica de estímulos comúnmente incluidos en los juegos. "Algunos elementos de gamificación utilizados son: puntos, recompensas, progresión, insignias y tablas de posiciones" (Zatarain, 2018, p. 116).

Por otro lado, existe un concepto que ha sido objeto de investigación de manera reciente, este es el pensamiento computacional, el cual según Basogain *et al.* (2015) es una metodología basada en la implementación de los conceptos básicos de las ciencias de la computación para resolver problemas cotidianos, diseñar sistemas domésticos y realizar tareas rutinarias. También menciona Zapata (2015) que aprender pensamiento computacional, no es aprender a programar computadoras, incluso algunas habilidades propias del pensamiento computacional no están relacionadas con ordenadores, es una habilidad que permite resolver problemas de manera inteligente e imaginativa.

Por lo tanto, el presente trabajo busca prototipar un videojuego serio con características de gamificación para dispositivos móviles orientado a niños de primaria, el cual, pretende, mediante temas básicos de la programación como lo son las estructuras secuenciales, iterativas y selectivas (la selección de estos temas se debe a que son los principios básicos de la programación, los cuales permiten idear un algoritmo capaz de resolver un problema de manera lógica y estructurada), incentivar el desarrollo del pensamiento computacional como una habilidad más para resolver situaciones cotidianas con un enfoque computacional.

1.2. Planteamiento del problema

El pensamiento computacional como área de investigación es relativamente nueva, desarrollada de manera masiva a principios de la década pasada (Téllez, 2019), por lo que, los materiales y experiencias en esta línea de investigación son recientes, lo que da a entender que los procesos para crear videojuegos educativos enfocados a desarrollar el pensamiento computacional se encuentran en exploración.

Por otro lado, los videojuegos orientados a enseñar programación han proliferado más en el mercado y a día de hoy existen muchas propuestas de este tipo, no así los videojuegos orientados al desarrollo del pensamiento computacional, lo que no quiere decir que no se fomente el pensamiento computacional en estos videojuegos, ya que al aprender programación se desarrolla indirectamente el pensamiento computacional (Álvarez, 2017; Basogain *et al.*, 2015; Compañ *et al.*, 2015), esto más bien es un señalamiento de la carencia de videojuegos destinados a desarrollar esta habilidad en particular.

Basado en lo anterior, se observó que, en los videojuegos educativos para niños orientados a enseñar programación, uno de los mecanismos más utilizados, no siendo el único, es el uso de un personaje “llamativo” al que se le tiene que asignar instrucciones secuenciales para cumplir una tarea u objetivo haciendo que el niño empiece a familiarizarse con la estructura de los lenguajes de programación. Algunos ejemplos de este tipo de juegos son: *Rabbids coding!* (<https://store.ubi.com/ofertas/rabbids-coding/5d96f9075cdf9a2eacdf68c9.html>), *Codemonkey* (<https://app.codemonkey.com>), *LightBot* (<https://lightbot.com>) y *Aprende a programar con el Chavo* (<https://civat.es/app/aprende-a-programar-con-el-chavo/>).

De igual manera se observó que los videojuegos mencionados cuentan con apartados visuales coloridos y divertidos los cuales están hechos con la intención de llamar la atención del niño, no obstante, se identificaron problemáticas relacionadas con los mecanismos de interacción entre el juego y el niño, haciendo que en algunos casos sea complicada de entender la tarea a realizar o el propósito de la misma, siendo este el principal problema al desarrollar el pensamiento computacional con esta clase de videojuegos que están orientados a enseñar programación, ya que hace que el niño pierda interés en seguir jugando, sin contar que muchos de estos juegos están en inglés lo que hace que el interés de los niños para aprender sobre computación se vea más reducida.

A raíz de la inesperada pandemia desatada por el coronavirus SARS-COV2 en 2020, el uso de TIC's para el trabajo y la educación se ha integrado como parte de la nueva normalidad que se vive en todo el mundo. Debido al reciente confinamiento global y, de manera específica, en México según Sada (2020) aumentó considerablemente el tiempo que las personas destinan al uso del teléfono celular, siendo gran parte de este tiempo destinado a cuestiones de entretenimiento.

Tomando en cuenta la problemática mencionada sobre el poco desarrollo de videojuegos orientados a fomentar el pensamiento computacional, el de los juegos educativos para aprender a programar, y el aumento en el uso de dispositivos móviles; resulta adecuado explorar el desarrollo del pensamiento computacional mediante videojuegos para dispositivos móviles, que sin necesidad de estar conectado a internet para su funcionamiento, permitan a los niños visualizar y aplicar este razonamiento en situaciones de su vida diaria, fomentando su interés en aprender más sobre las temáticas relacionadas con la computación y sus distintas aplicaciones.

1.3. Justificación

La pandemia, y en consecuencia el confinamiento, también ha incrementado las actividades relacionadas a los videojuegos (Montás, 2020), bajo la idea inicial que los videojuegos no sirven más que para distraer y perder el tiempo, sin embargo, esto también ha provocado que muchos jóvenes se sientan atraídos hacia la programación de videojuegos como un posible trabajo futuro, aunque siguen carentes de información y conocimientos sobre el área, lo que causa que dejen de lado ese deseo.

Una manera de afrontar esta carencia de información desde edades tempranas, es mediante el desarrollo del pensamiento computacional, ya que como bien sugiere Wing (2006), el pensamiento computacional debería ser una habilidad que todos deberían de tener, no solamente los que se dedican a esta área. Esta sugerencia se debe a que el pensamiento computacional permite la resolución de distintos problemas mediante técnicas nativas de la informática sin importar el área de estos. De esta manera, el pensamiento computacional facilitará el aprendizaje sobre computación a los niños que quieran dedicarse a esta área.

Con el desarrollo de este trabajo se espera fortalecer el pensamiento computacional en niños mediante mecánicas implementadas en un videojuego con elementos de gamificación para motivarlos a jugar y aprender en el transcurso del juego sin recurrir a interfaces complicadas basadas en instrucciones como lo hacen los videojuegos convencionales de programación.

1.4. Objetivos

1.4.1. Objetivo general

Desarrollar un prototipo de videojuego serio para dispositivos móviles que fomente el pensamiento computacional en niños de quinto y sexto grado de primaria.

1.4.2. Objetivos específicos

1. Plantear el concepto y las mecánicas del juego de tal manera que fomente el desarrollo del pensamiento computacional.
2. Desarrollar un videojuego para dispositivos móviles empleando el motor de videojuegos Godot.
3. Realizar pruebas individuales con niños radicados en el Istmo de Tehuantepec para medir su comprensión sobre estructuras secuenciales, selectivas e iterativas después de jugar el videojuego.

1.5. Alcances

- El videojuego podrá ejecutarse sobre cualquier dispositivo móvil con Android.
- Será jugable sin necesidad de una conexión a internet.
- El videojuego sólo tendrá como ejes temáticos las estructuras secuenciales, selectivas e iterativas al resolver problemas, ya que las principales características del pensamiento computacional se pueden fomentar con estos tres.
- Se llevarán a cabo pruebas de usabilidad para medir la satisfacción del usuario con el videojuego y su comprensión de los temas planteados en el juego.
- El resultado de este proyecto será un prototipo de videojuego educativo que no será lanzado al mercado.
- El proyecto estará dirigido principalmente a la comunidad zapoteca de Santa María Xadani y alrededores.

1.6. Limitaciones

- El videojuego desarrollado será exclusivo para dispositivos móviles con sistema operativo Android.
- El videojuego sólo podrá ser jugado por niños que ya sepan leer y ser capaces de seguir instrucciones sencillas.

- Debido a la contingencia generada por el virus SARS-CoV2, el tamaño de la muestra para las pruebas será reducido, contemplando únicamente a niños de Santa María Xadani y Juchitán.
- El trabajo no estará financiado por lo que se optará por utilizar software libre o gratuito.

1.7. Trabajos relacionados

En el trabajo de Rahman (2017) se describe el desarrollo de un videojuego educativo en 3D para dispositivos móviles con Android para enseñar la lógica relacionada a la programación a personas de 5 a 18 años. El jugador controla a un personaje al que se le asignan instrucciones que va ejecutando una a una para derrotar enemigos y superar obstáculos con el objetivo de alcanzar un gato ubicado en el mapa.

Manquilef (2021) redacta en su trabajo el desarrollo de un videojuego educativo en 3D para computadoras personales, enfocado a desarrollar el pensamiento computacional en los jugadores mediante la resolución de puzzles. Utiliza un sistema de comandos introducidos en una consola gráfica dentro del juego que afectan principalmente a unas plataformas que están repartidas en el mapa con el objetivo de escapar de la sala del nivel, las instrucciones emulan estructuras cíclicas y selectivas básicas de programación.

Por otro lado, Ticante *et al.* (2019) en su trabajo desarrolla un videojuego educativo de tipo RPG (juego de rol por sus siglas en inglés) de aventuras en 2D, para computadoras personales, como apoyo para comprender principios básicos de programación y desarrollar habilidad lógica en niños. La interfaz está constituida por una consola en la que el jugador tendrá que escribir instrucciones para que el personaje recorra el nivel y derrote a los enemigos que se vaya encontrando.

Además de los trabajos mencionados anteriormente existen videojuegos educativos orientados a la computación creados por desarrolladoras para distintas plataformas. Estos tienen un objetivo más comercial y algunos no son precisamente trabajos de investigación, por lo tanto, el desarrollo de estos no está documentado lo que dificulta adquirir nuevo conocimiento sobre el desarrollo de videojuegos educativos. La Tabla 1 lista algunos videojuegos educativos disponibles en el mercado, su información y algunos puntos buenos y carencias respecto a cada uno.

<i>Videojuego</i>	Enlace	Plataforma	Características	Observaciones
<i>LightBot</i>	https://lightbot.com/	<ul style="list-style-type: none"> • En línea • iOS 	<ul style="list-style-type: none"> • Videojuego para aprender 	<ul style="list-style-type: none"> • Interfaz amigable

		<ul style="list-style-type: none"> • Android 	<p>programación basado en la resolución de puzles.</p> <ul style="list-style-type: none"> • El personaje debe de pasar por los cuadros apagados para prenderlos. • Implementa el uso de instrucciones a través de comandos gráficos. 	<ul style="list-style-type: none"> • Traducción a varios idiomas • Tiene costo • La versión en línea requiere de la obsoleta tecnología Flash para poder jugar. • Requiere de una conexión a internet.
<i>Code Monkey</i>	https://www.codemonkey.com/	<ul style="list-style-type: none"> • En línea 	<ul style="list-style-type: none"> • Videojuego para aprender programación basado en la resolución de puzles. • El juego consiste en conseguir el plátano del monito. • Las instrucciones se asemejan a un lenguaje de programación escrito. 	<ul style="list-style-type: none"> • Entorno muy amigable para los niños. • Tiene mucha ayuda y diferentes planes de aprendizaje. • Tiene planes para maestros y padres. • Incluye niveles de prueba sin costo, pero se tendrá que pagar por un plan para seguir jugando. • Las instrucciones que se le dan al monito están en inglés. • Requiere de una conexión a internet.
<i>Code Combat</i>	https://codecombat.com/	<ul style="list-style-type: none"> • En Línea 	<ul style="list-style-type: none"> • Videojuego de aventura, basado en la exploración de mazmorras para aprender programación. • El objetivo del juego es obtener el tesoro en cada nivel evitando los peligros de la mazmorra. • Las instrucciones se asemejan a un 	<ul style="list-style-type: none"> • El entorno se ve muy llamativo parecido a un juego de aventuras. • Permite la mejora y elección del personaje. • No tiene costo. • Permite utilizar Python,

			lenguaje de programación escrito.	JavaScript, Coffe Script o C++ como sintaxis para la programación. <ul style="list-style-type: none"> Para utilizar algunas sintaxis es necesario estar registrado en la página del juego.
<i>Cargo-Bot</i>	https://apps.apple.com/es/app/cargo-bot/id519690804	<ul style="list-style-type: none"> iOS 	<ul style="list-style-type: none"> Videojuego de rompecabezas para aprender a programar instrucciones. El objetivo del juego es introducir instrucciones a un brazo robótico para replicar un patrón. Las instrucciones están representadas de manera gráfica para que el niño pueda guiarse sin necesidad de escribir nada. 	<ul style="list-style-type: none"> El juego está para su descarga gratuita en la tienda de Apple. El ambiente es muy colorido con una temática caricaturesca. Está disponible únicamente en inglés. No requiere de una conexión a internet para jugar.
<i>RoboZZle</i>	<ul style="list-style-type: none"> https://apps.apple.com/es/app/robozzle/id350729261 https://play.google.com/store/apps/details?id=com.team242.robozzle https://amzn.to/2HGC4gC 	<ul style="list-style-type: none"> iOS Android Kindle Fire 	<ul style="list-style-type: none"> Videojuego para aprender principios de programación basado en la resolución de puzles. El objetivo es recorrer el nivel mediante instrucciones recogiendo todas las estrellas del camino. Las instrucciones son flechas que se arrastran al área de instrucciones. 	<ul style="list-style-type: none"> El videojuego está disponible de manera gratuita en la tienda de Apple y la App Store de Android, con compras dentro de la app. La versión para Kindle Fire tiene precio. La interfaz de usuario es poco intuitiva. El juego está en inglés. La versión de Kindle Fire tiene

				traducción a ruso y francés. <ul style="list-style-type: none"> Los niveles son desarrollados por diferentes usuarios del juego.
<i>Daisy the Dinosaur</i>	https://apps.apple.com/es/app/daisy-the-dinosaur/id490514278	<ul style="list-style-type: none"> iOS 	<ul style="list-style-type: none"> Aplicación para aprender principios básicos de programación. El objetivo de la aplicación es hacer que el dinosaurio baile y se mueva por la pantalla mediante instrucciones. 	<ul style="list-style-type: none"> La aplicación es gratuita en la tienda de Apple. La interfaz es muy amigable y sencilla. Las instrucciones se introducen en forma de bloques con una etiqueta de lo que hará el dinosaurio. Tiene un tipo de juego libre en el que el niño hace lo que quiera y un modo reto donde se plantean objetivos a cumplir. La aplicación está únicamente en inglés.
<i>Kodable</i>	https://apps.apple.com/es/app/kodable/id577673067	<ul style="list-style-type: none"> iOS 	<ul style="list-style-type: none"> Videojuego para aprender a programar mediante la resolución de retos. El objetivo del juego es resolver los retos ambientados en el recorrido de unos alienígenas en su planeta. 	<ul style="list-style-type: none"> El videojuego es gratuito en la tienda de Apple. Tiene un modo para aprender JavaScript desde cero, pero requiere de un pago monetario en el juego. El juego es muy amigable gráficamente, muy colorido y atractivo para niños.

				<ul style="list-style-type: none"> • No es necesario saber leer para jugar, ya que el juego explica todo mediante gráficos y voz. • Disponible únicamente en inglés.
--	--	--	--	--

Tabla 1. Videojuegos educativos publicados orientados a programación.

De los videojuegos relacionados mencionados y descritos en la Tabla 1 se pueden observar que algunos fueron desarrollados para dispositivos móviles con iOS como sistema operativo, cabe destacar que estos dispositivos son costosos de adquirir y por lo tanto, no toda la población puede contar con uno; también hay juegos jugables en línea, lo que quiere decir que requieren de un navegador web y de una conexión permanente a internet para poder jugar, estos navegadores están disponibles en cualquier equipo de cómputo y dispositivo móvil, aunque es más común y cómodo su uso en equipos de cómputo, que de igual manera que con los dispositivos móviles con iOS, estos equipos pueden ser costosos y no todas las familias pueden darse el privilegio de contar con uno en casa. (revisar datos de la INEGI sobre familias con equipos de cómputo y celulares)

1.8. Metodología

Se usarán en conjunto una metodología de desarrollo de videojuegos y un modelo de conceptualización de videojuegos educativos para el desarrollo de este proyecto. Es de mencionar, que al momento no se ha identificado en la literatura una metodología estandarizada para el desarrollo de videojuegos educativos. Por lo anterior, la propuesta de una metodología basada en una metodología en particular y un modelo de conceptualización en el desarrollo del videojuego es un diferenciador de este trabajo de tesis.

Esta propuesta metodológica está basada principalmente en la metodología SUM para el desarrollo de videojuegos; descartando las etapas de planificación y gestión de riesgos de la metodología SUM, las razones son que la documentación del proyecto ya incorpora una planificación de tiempo de todo el proyecto donde cada etapa de la metodología ya está contemplada y debido a que al ser un proyecto de desarrollo pequeño y contar con un equipo conformado por una persona, los problemas surgidos se pueden gestionar de manera inmediata y tomar la mejor decisión dependiendo de la situación actual del proyecto.

También, al ser SUM una metodología enfocada al desarrollo de videojuegos en general, no contempla características principales a tomar en cuenta al momento de conceptualizar videojuegos educativos, por esta razón la etapa de concepto de la metodología SUM será reemplazada por el modelo conceptual para el diseño de videojuegos educativos planteado por Ruíz (2014), el cual es un modelo orientado a la conceptualización de videojuegos con valor educativo.

Del modelo conceptual de Ruíz se descartarán las entidades de socialización y servicios, debido a que, el juego no contará con algún tipo de interacción entre jugadores y solamente se desarrollará el apartado jugable del videojuego y no incluirá ningún tipo de medio cuyo propósito no sea jugar. De esta manera, en la Figura 2 se puede observar el proceso metodológico que se llevará a cabo en el proyecto.

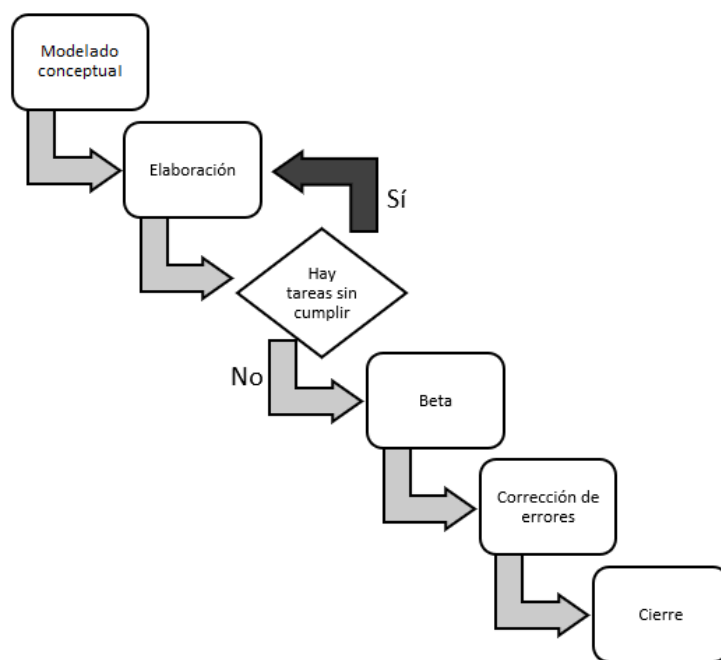


Figura 2. Etapas de la metodología de trabajo del proyecto.

A continuación, se describen cada una de las etapas de la metodología que se llevarán a cabo en el transcurso del proyecto.

- **Modelado conceptual:** se plantearán las mecánicas que regirán la jugabilidad del videojuego, los retos que el jugador tendrá que afrontar durante el juego, los escenarios que adornarán cada nivel del juego, el diseño y caracterización de los personajes, el *feedback* que el juego brindará al jugador con cada acción realizada mientras está jugando, la historia

de fondo del videojuego, las recompensas que el jugador obtendrá al cumplir ciertos requisitos, los métodos de persistencia que tendrá el jugador para continuar jugando en cualquier momento y la definición de la interfaz de usuario.

- **Elaboración:** se llevará a cabo la programación del videojuego tomando como base todo lo definido en la etapa anterior. Todo el concepto será dividido en tareas que serán enlistadas para su desarrollo individual en esta etapa. Este proceso de desarrollo será iterativo e incremental y se puede observar en la Figura 2. Cuando ya no haya tareas por desarrollar es cuando esta etapa llega a su fin y se pasa a la siguiente.
- **Beta:** es una versión temprana del videojuego, en esta versión se llevan a cabo pruebas de jugabilidad para detectar errores y *bugs* en la programación. También permite comprobar que las mecánicas y el comportamiento del juego funcionen de manera correcta.
- **Corrección de errores:** los errores y *bugs* detectados en la beta, se enlistan y se corrigen permitiendo refinar el resultado final.
- **Cierre:** se obtiene la versión final del videojuego después de haber corregido los errores detectados en la beta, se libera el juego para realizar pruebas con los usuarios finales.

Por último, a pesar de que se menciona la liberación del juego dentro de la metodología, el resultado de este proyecto será un prototipo de videojuego educativo que no será lanzado al mercado y cuyos usuarios finales estarán compuestos por un grupo de niños seleccionados, también cabe recalcar que cada etapa será plasmada en el respectivo documento de tesis del proyecto.

2. Marco teórico

En este apartado se define el concepto de videojuegos y abordan temas como las estructuras de control, el pensamiento computacional, metodología de desarrollo de videojuegos SUM y un modelo conceptual para el diseño de videojuegos educativos.

2.1. Videojuego

Un videojuego es un juego digital interactivo, creado con fines de entretenimiento en general y se basa en la interacción con personas o personajes ficticios (Ticante *et al.*, 2019). De una manera más amplia Frasca (2001) define a los videojuegos como cualquier forma de software de entretenimiento por computadora, ya sea textual o basado en imágenes, usando cualquier

plataforma electrónica como computadoras personales o consolas e involucrando a uno más jugadores en un entorno físico o de red.

Así como cada persona tiene un diferente punto de vista de un mismo tema, diferentes autores tienen una concepción distinta de lo que un videojuego es, sin embargo, Wolf y Perron (2003) identifican cuatro elementos fundamentales que comparten todos los videojuegos, estos son gráficos, interfaz, actividad del jugador y algoritmo. Los gráficos son un conjunto de visualizaciones cambiantes en una pantalla que produce una imagen basada en píxeles; la interfaz marca la frontera entre el jugador y el videojuego, es el medio por el cual el mundo real se comunica con el videojuego, cuantifica las acciones del jugador y las convierte en datos; las actividades del jugador consisten en la entrada de datos por medio de la interfaz, así una acción en el videojuego será el resultado de una acción realizada físicamente por el jugador; y el algoritmo es el que define el videojuego, las reglas, las respuestas de este y principalmente controla todo.

Dicho de una manera más sencilla, son juegos que se juegan en un medio electrónico. Estos utilizan distintos dispositivos de interacción para capturar las acciones del jugador, como teclados de computadora, *mouse*, *joysticks* de consola, pantallas táctiles, entre otros.

2.1.1. Tipos de videojuegos

Los videojuegos se pueden clasificar de acuerdo a su género, esta clasificación se hace en referencia a las reglas del juego, mejor conocidas como mecánicas del juego. En la Tabla 2 (Empantallados, s.f.) se mencionan algunos géneros de videojuegos.

<i>Tipo</i>	<i>Descripción</i>
<i>Acción</i>	Se basan principalmente en eliminar enemigos y su contenido suele ser violento. Implican una inmersión grande, concentración y por lo general se relacionan con exposiciones cortas de tiempo.
<i>Aventura</i>	El protagonista del juego atraviesa niveles, lucha contra enemigos y recoge objetos de valor. Casi siempre son juegos de larga duración con argumentos extensos y dependen completamente de la trama para su valoración.
<i>Arcade</i>	Son juegos de plataformas, laberintos y aventuras en donde el usuario debe superar pantallas para seguir jugando. Imponen un ritmo rápido de juego y requieren tiempo mínimos de reacción por parte del jugador.
<i>Deportivo</i>	Recrean diversos deportes y requieren de habilidad, rapidez y precisión. No suelen contener tramas argumentales de ningún tipo y contienen un alto grado de socialización.
<i>Estrategia</i>	Consisten en trazar una estrategia para superar al contrincante. Exigen concentración, saber administrar recursos, pensar y definir estrategias de juego. Se caracterizan por tener una curva de aprendizaje de algún tipo, ya que se pueden perder las primeras partidas,

	pero el jugador va aprendiendo de esto. Pueden ser útiles para desarrollar la inteligencia y pueden llegar a ser complicados de dominar.
<i>Simulación</i>	Permiten experimentar el funcionamiento de máquinas, fenómenos, situaciones y asumir el mando en estas. Representan acciones que se pueden vivir fuera de un contexto de peligro real, como volar un avión o conducir un tanque de la segunda guerra mundial.
<i>Juegos de mesa</i>	La tecnología informática sustituye al material tradicional del juego e incluso al adversario por un personaje controlado por inteligencia artificial.

Tabla 2. Clasificación de videojuegos por su género.

Cabe resaltar que existen más géneros de videojuegos con sus propias características diferenciadoras, también un género no es excluyente de otro, al contrario, se pueden combinar distintos géneros para generar un videojuego más variado en mecánicas.

2.1.2. Videojuego educativo

Este tiene un propósito más allá del entretenimiento del usuario, tiene entre sus objetivos según López (2016) crear entornos de aprendizaje virtuales que permitan experimentar con problemas del mundo real, con el propósito de hallar soluciones y obtener conocimiento sin temor a equivocarse ni tener repercusiones sobre sus decisiones. De esta manera se puede intentar un objetivo cuantas veces el jugador quiera sin importar cuantas veces falle, ya que sus errores en el videojuego no afectan al mundo real.

2.1.3. Motor de videojuego

Vallejo & Martín (2019) describen un motor de videojuegos como una herramienta de software la cual separa los componentes fundamentales de un videojuego (sistema de renderizado gráfico, sistema de detección de colisiones, sistema de audio, sistema de físicas) en módulos, y mediante una interfaz de usuario permite modificar y enlazar los módulos para desarrollar un videojuego.

En la actualidad existen varios motores de videojuegos, algunos más complejos que otros, algunos para uso exclusivo de empresas y otros de uso libre para los entusiastas del desarrollo de videojuegos; cada uno con sus ventajas y desventajas. En la Tabla 3, se listan algunos motores tomados en cuenta para el desarrollo del proyecto, todos con una curva de aprendizaje baja, la tabla contiene las características principales de cada motor.

<i>Motor</i>	<i>Características</i>
<i>Godot</i>	<ul style="list-style-type: none"> • Es de código abierto con licencia del MIT. • Cuenta con un sistema de programación gráfica por nodos. • Permite programación escrita para mayor control del juego mediante GDScript (lenguaje exclusivo de Godot y basado en Python), C# y C++. • Permite el desarrollo de videojuegos en 2D y 3D, aunque está más enfocado al 2D.

	<ul style="list-style-type: none"> • Exporta juegos para Windows, macOS, Linux, UWP, BSD, iOS, Android y HTML5 de manera nativa. • Exporta juegos a consolas Nintendo Switch, PS4, Xbox One mediante aplicaciones de terceros. • Cuenta con un sistema para crear animaciones.
<i>GameMaker Studio 2</i>	<ul style="list-style-type: none"> • Es un motor de pago y cuenta con una prueba gratuita de 30 días o una licencia gratuita para hacer pruebas. • Cuenta con un sistema de programación gráfica <i>Drag and Drop</i> (arrastrar y soltar) muy intuitivo. • Permite la programación escrita para mayor control del juego mediante GML (<i>GameMaker Language</i>) basado en C. • Especializado en el desarrollo de videojuegos en 2D. • Exporta juegos para Windows, macOS, Linux, HTML5, Android, iOS, PS4 y Xbox One.
<i>GameSalad</i>	<ul style="list-style-type: none"> • Es un motor de pago, pero cuenta con una prueba gratis. • Especializado en el desarrollo de videojuegos en 2D. • Cuenta únicamente con un sistema de programación gráfica <i>Drag and Drop</i> (arrastrar y soltar). • Exporta juegos para Windows, iOS, Android y HTML5.
<i>Construct 2</i>	<ul style="list-style-type: none"> • Especializado en el desarrollo de videojuegos 2D. • Exporta juegos para Windows, macOS, Linux, HTML5, Android y Nintendo Wii U. • Cuenta con una interfaz gráfica que permite la programación sin códigos. • Permite la programación escrita para mayor control del juego mediante HTML5 y JavaScript.

Tabla 3. Motores de videojuegos.

Se excluyeron los motores más populares y completos como *UnrealEngine*, *Unity* y *CryEngine* debido a su alta curva de aprendizaje y por la naturaleza del proyecto que estará enfocado al desarrollo de un videojuego 2D. Así, tomando en cuenta las características del motor y el precio por el uso del mismo; se usará Godot para el desarrollo del proyecto, esto debido a su libertad de uso y personalización del videojuego mediante código de ser necesario.

2.2. Estructuras de control

Existen tres estructuras de control básicas que guían el orden de ejecución de un programa de computadora: las estructuras secuenciales, selectivas e iterativas. Estas son esenciales para cualquier lenguaje de programación, ya que son las que dan lógica al programa desarrollado. Según Ristić *et al.* (2016) el aprender estos conceptos básicos de programación facilitará el aprendizaje de cualquier lenguaje de programación.

2.2.1. Estructura secuencial

Es una estructura de control en la que las instrucciones se ejecutan una después de otra en el orden en el que se encuentran escritas en un programa (Márquez *et al.*, 2011).

Una estructura secuencial se puede ver de la siguiente manera:

Instrucción 1

Instrucción 2

.

.

.

Instrucción n

La n representa el infinito y las instrucciones se ejecutan una por una de arriba hacia abajo hasta la n -ésima instrucción.

2.2.2. Estructura selectiva

Son aquellas que evalúan una expresión, usualmente una condición booleana, y a partir del resultado permiten tomar decisiones en una, dos o más opciones. Una expresión booleana no es más que una expresión que puede tener como valor verdadero o falso.

2.2.3. Estructura iterativa

Una estructura iterativa permite repetir una acción; la repetición es controlada por una expresión que es una condición booleana.

2.3. Pensamiento computacional

El pensamiento computacional implica resolver problemas, diseñar sistemas y comprender el comportamiento humano, haciendo uso de los conceptos fundamentales de la informática (Wing, 2006). La Sociedad Internacional de Tecnología en educación y la Asociación de Docentes en Ciencias de la Computación (ISTE & CSTA) (2011) consideran al pensamiento computacional como un proceso de resolución de problemas que incluyen (pero no están limitadas a) las siguientes características:

- Formulación de problemas de manera que se puedan utilizar computadoras y otras herramientas para su resolución.
- Organización lógica y analítica de datos.

- Representación de datos de manera abstracta, como modelados y simulaciones.
- Automatización de soluciones a través de pensamiento algorítmico (una serie de pasos ordenados).
- Identificación, análisis, e implementación de posibles soluciones con la finalidad de obtener la más eficiente y efectiva combinación de pasos y recursos.
- Generalización y transferencia de este proceso de resolución de problemas a un conjunto variado de problemas.

Otro concepto surgido a través de investigaciones científicas es definido por The Royal Society (2012) quienes consideran al pensamiento computacional como el proceso de reconocimiento de aspectos de la informática en el mundo que nos rodea, y la aplicación de herramientas y técnicas de la informática para comprender y razonar sobre los sistemas y procesos tanto naturales como artificiales.

Finalmente, Manquilef (2021) define que “el pensamiento computacional es más bien una idea de cómo pensar para dar solución a problemas, más que simplemente enseñar a programar o la interacción humano-computador”. De igual manera identifica tres componentes fundamentales que forman parte de esta forma de pensar:

- Abstracción: separar el problema y aislar cada parte de esta para su análisis individual para después hallar la relación entre todas las partes.
- Descomposición: complementa la abstracción, y es la característica de dividir el problema en partes más pequeñas, donde cada parte es un subproblema del principal.
- Reconocimiento de patrones: reducir el problema hasta encontrar un patrón de comportamiento, tanto en la entrada como en la salida, de la información relacionada al problema, para así hallar una solución general.

2.4. Metodología de desarrollo de videojuegos SUM

Existen gran cantidad de metodologías para desarrollar software, cada una con un diferente enfoque. No existe una metodología que sea mejor que otra, ya que; dependiendo de los objetivos del desarrollo, el tiempo, la cantidad de gente involucrada, el presupuesto, entre otros factores; se elige una metodología con la cual trabajar. Incluso puede darse el caso de que se combinen metodologías si el proyecto así lo amerita. Para este proyecto en específico, el desarrollo del

videojuego será llevado a cabo mediante la implementación de esta metodología, exceptuando la etapa de *Concepto* que será reemplazada por el modelo conceptual para el diseño de videojuegos educativos.

La metodología SUM es una metodología basada en SCRUM, cuyo objetivo al desarrollar videojuegos es “la obtención de un software de calidad en tiempo y costo, así como la mejora continua del proceso para aumentar su eficiencia y eficacia” (Arenas, 2018, p. 24).

Se compone de cinco fases, *Concepto*, *Planificación*, *Elaboración*, *Beta* y *Cierre*, pero hay una fase que se lleva a cabo durante todo el desarrollo del proyecto y que se ejecuta a la par de la fase actual del desarrollo, esta es la *Gestión de riesgos*. La Figura 3 muestra la estructura y el flujo que la metodología sigue durante todo el proceso de desarrollo.

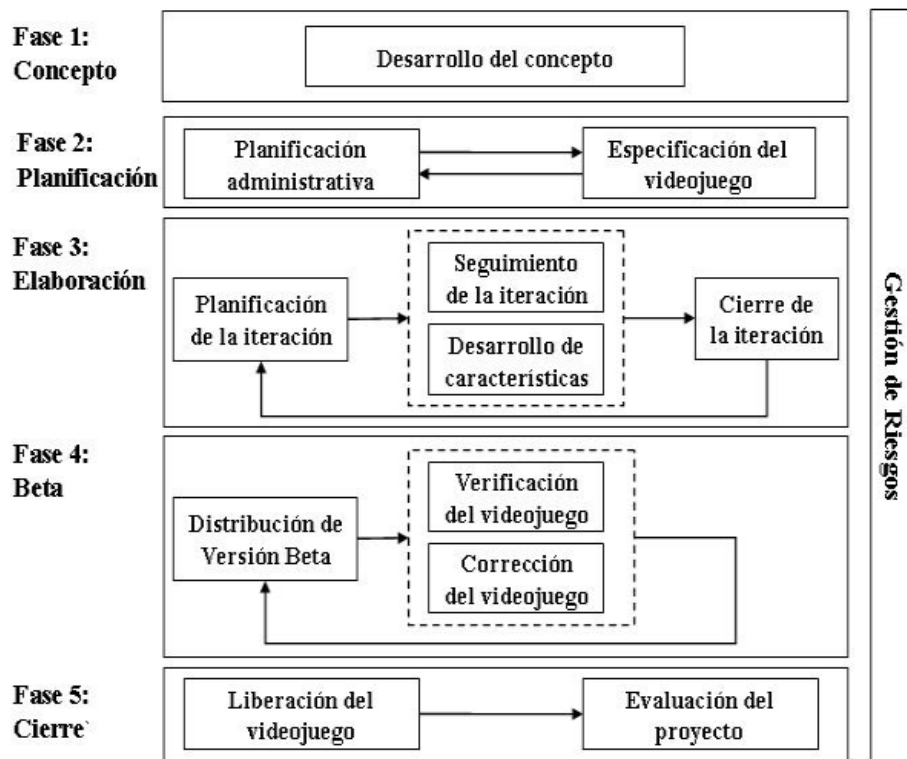


Figura 3. Estructura de la metodología SUM. Extraída de <http://www.gemserk.com/sum/>

2.4.1. Concepto

Esta primera fase contempla la idea principal del videojuego a desarrollar, que tipo de juego será, la historia, la ambientación y principalmente las mecánicas de juego, esto último permitirá definir un rumbo concreto al momento de programar el videojuego; aspectos de negocio del videojuego, a

quién va dirigido, modelo de negocio; y por último contempla el aspecto técnico del videojuego, el lenguaje de programación, el motor y las herramientas para su desarrollo (Acerenza *et al.*, 2009; Arenas, 2018; Murillo *et al.*, 2018).

2.4.2. Planificación

En esta fase lo principal es medir los tiempos que cada parte del desarrollo llevará, desde la fase de concepto hasta la fase de cierre y elaborar un cronograma de actividades.

2.4.3. Elaboración

Es la parte donde se lleva a cabo la programación, el desarrollo gráfico, y, por ende, el resultado funcional del videojuego. Esta fase se lleva a cabo de manera iterativa e incremental, dando como resultado en cada iteración una parte funcional del juego. Esta misma fase se divide en las siguientes actividades que se llevan a cabo igual en cada iteración:

1. Planificación de la iteración: se plantean los objetivos y características que se obtendrán al finalizar la iteración, estos se descomponen en tareas que se llevarán a cabo a lo largo de la iteración y también se establecen las validaciones que analizan que la parte realizada está cubierta.
2. Seguimiento de la iteración: controla el avance de la iteración en base a los objetivos establecidos.
3. Desarrollo de características: se llevan a cabo las tareas (véase Figura 4) establecidas para cubrir la característica planteada, una vez que una tarea está terminada se realizan pruebas para encontrar errores, en caso de haberlos, buscar maneras de corregirlos.
4. Cierre de la iteración: evalúa el estado actual del videojuego respecto a lo ocurrido en la iteración y así adaptar el plan de desarrollo con lo obtenido. Se llevan a cabo pruebas de usuario sobre la característica desarrollada y basado en eso se eliminan, agregan o modifican características para la siguiente iteración.

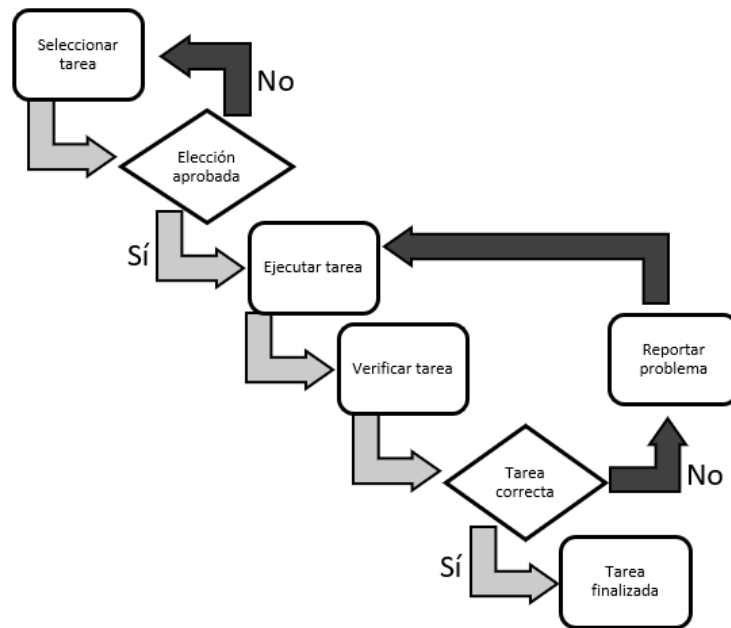


Figura 4. Proceso de desarrollo de tareas.

2.4.4. Beta

Es una versión de prueba del videojuego completo, que permite probar la funcionalidad del videojuego para encontrar errores y bugs para después corregirlos, también tiene como propósito verificar que las mecánicas y el comportamiento del videojuego son los correctos.

2.4.5. Cierre

Se llega al cierre después de haber refinado la versión beta, en esta etapa el videojuego se presenta al público objetivo para evaluar el rendimiento del videojuego y tomar consideraciones a futuro.

2.4.6. Gestión de riesgos

Su objetivo es reducir la ocurrencia y el impacto de problemas en todo momento mediante el análisis de problemas que podrían ocurrir durante el proceso y establecer métodos de solución para estos casos.

2.5. Modelo conceptual para el diseño de videojuegos educativos

En 2014, Mario Rafael Ruiz Vargas presentó en su investigación doctoral “Un Modelo Conceptual Para El Diseño de Videojuegos Educativos” (Ruiz, 2014), una propuesta para el diseño conceptual de videojuegos educativos, que en palabras del propio Ruiz, no se centra en los retos y problemas específicos asociados con un género de juego; más bien reúne y organiza las características que la

literatura de videojuegos considera a menudo como significativas dentro de la producción de una experiencia de videojuego atractiva, divertida y educativa. No está orientado como tal al desarrollo funcional del videojuego, más bien, es una guía para concebir la idea inicial y las entidades más importantes para un videojuego educativo.

Dicha propuesta identifica 12 aspectos fundamentales en un videojuego educativo: *mecánica, retos, escenario, caracterización de personajes, feedback, socialización, debriefing, storyline, recompensa, persistencia, servicios, interfaz del juego e interacción*. Descritas por Ruiz de la siguiente manera:

2.5.1. Mecánica

“Define los elementos o entidades principales del juego y las acciones e interacciones entre ellas que pueden tener lugar a lo largo de la partida. Determina qué es lo que puede suceder durante una partida” (Ruiz, 2014, p. 43).

2.5.2. Retos

Ruiz (2014) define los retos como “las metas que las distintas entidades deberán tratar de lograr durante el juego, junto con las situaciones o eventos que deberán evitarse” (p. 44). A partir de las mecánicas definidas, se podrán plantear los objetivos que el jugador deberá lograr en el juego, esto a su vez se podrá asociar con el objetivo de aprendizaje. Por ejemplo, separar perros y gatos de una manada de animales, este objetivo podría asociarse con “aprender a identificar”. Además de las metas, se pueden definir objetivos o eventos negativos que el jugador debe tratar de evitar para enriquecer la experiencia de juego del usuario.

2.5.3. Escenario

Definido por Ruiz (2014) como “el espacio virtual y las representaciones que serán utilizadas para encarnar las entidades definidas en la mecánica” (p. 44). Cuando en un mismo juego se haya definido más de una escena (lugares donde el juego puede tener lugar) puede ser necesario establecer enlaces entre ellas que determinen la forma en que los jugadores pueden navegar de unas a otras.

La descripción de una escena puede cubrir diversos aspectos incluyendo el tipo de dimensiones de la misma, el tipo de perspectiva, el área de visión de la escena, posiciones de la cámara, imágenes de fondo y los objetos o entidades con los que el jugador podría interactuar.

2.5.4. Caracterización de personajes

Ruiz (2014) recomienda que “el diseño de los personajes de juegos dirigidos a niños de corta edad suele utilizar representaciones gráficas cercanas a los dibujos animados con los que están acostumbrados a ver” (p. 51).

En relación con la descripción de las habilidades y comportamientos, es necesario tener en cuenta que cada personaje deberá estar asociado a alguna entidad de juego de las previamente definidas en la mecánica básica. El control de la entidad podrá estar o bien en manos de uno de los jugadores PCC (*Player Controlled Character*) o bien será llevado a cabo automáticamente por el propio ordenador NPC (*Non-Player Character*). En este último, dependiendo del grado de sofisticación del juego y de la necesidad de simular un comportamiento inteligente, estas reglas pueden ser implementadas en forma de un conjunto de eventos que determinen cuándo se dispara, o bien a través de algún algoritmo o técnica propia del área de inteligencia artificial.

2.5.5. *Feedback*

Lo define Ruiz (2014) como “los elementos y procedimientos que se utilizarán en cada momento para dar a conocer al jugador las consecuencias de sus acciones, su grado de progreso hacia los objetivos y en general, el estado y situación actual de la partida” (p. 44).

Además del *feedback* asociado a los objetivos del juego, suele ser habitual en los juegos educativos encontrarse con otro tipo de *feedback* que tiene por finalidad principal dar soporte al principio educativo del juego, suministrando información adicional que contribuya a la adquisición de conocimiento y al logro de los objetivos educativos.

2.5.6. Socialización

“Establece los mecanismos de comunicación e interacción entre jugadores que el juego soportará” (Ruiz, 2014, p.44). Incluir algún tipo de componente que dé soporte a la socialización entre jugadores, es sumamente importante para el caso de los videojuegos educativos, ya que no sólo puede contribuir a incrementar la motivación de los jugadores, sino que además proporcionaría la base para el desarrollo del aprendizaje como ejercicio socio cultural.

2.5.7. *Debriefing*

El término hace referencia al proceso durante el cual los participantes de un proceso educativo reflexionan y analizan sobre la actividad desarrollada, asociando lo aprendido con conceptos

previamente conocidos, y formando hipótesis que puedan ser verificadas en experiencias futuras. En otras palabras, “son actividades que no forman parte del juego en sí y cuyo fin es hacer recapacitar al jugador sobre lo que ocurre en el juego y relacionarlo con la actividad en el mundo real” (Ruiz, 2014, p. 44).

2.5.8. *Storyline*

“Define el contexto y el argumento del juego, ordenando los objetivos que se propondrán al jugador en distintas secuencias” (Ruiz, 2014, p.44). Contextualizar el desarrollo del juego dentro de una historia, permite añadir a los factores intrínsecamente motivantes asociados al propio juego, el de la curiosidad del jugador por conocer la evolución de la historia.

2.5.9. Recompensa

“Conjunto de premios que se activan en función de los logros y acciones del jugador y que tienen por fin aumentar su motivación de progresar en el juego” (Ruiz, 2014, p. 44). Es frecuente que los videojuegos hoy en día incluyan mecanismos muy explícitos para premiar al jugador, algunos ejemplos de esto pueden ser, la acumulación de puntos, la posibilidad de explorar áreas secretas, la activación de juegos extra o de bonificación, o la capacidad de personalizar el avatar con ciertos objetos o permitir hacer uso de habilidades especiales.

2.5.10. Persistencia

“Estos son un conjunto de mecanismos que el juego implementará para permitir o no al jugador afrontar los desafíos en varias sesiones de juego, y que establecerán la totalidad de la información que se mantendrá entre ellas” (Ruiz, 2014, p. 44).

En general, la elección de la información a persistir suele verse condicionada por el género al que el videojuego pertenece. De esta forma, en juegos de aventura o de rol, donde el mantenimiento de la continuidad de la historia es fundamental, es necesario almacenar el estado del juego prácticamente al completo. Por el contrario, en juegos de acción o plataforma es suficiente con almacenar el punto aproximado en el que se encuentra la historia y parte de la información relativa al estado actual del personaje.

2.5.11. Servicios

Ruiz (2014) define los servicios como “herramientas y aplicaciones incluidas en el juego, con el fin de dar soporte a otros elementos de diseño o de enriquecer y aumentar las posibilidades

ofrecidas por el juego” (p. 44). Por ejemplo, un videojuego puede incluir servicios de mensajería o de *chat* que den soporte a los distintos tipos de comunicación entre participantes especificados en el componente de socialización, servicios de foro que sirvan para llevar a cabo ciertas actividades de *debriefing*, o servicios de ranking que den apoyo a mecanismos de recompensa previamente definidos.

2.5.12. Interfaz del juego e interacción

Este elemento es definido por Ruiz (2014) como “el conjunto de medios, tanto de entrada como de salida, de los que el jugador dispondrá para comunicarse con el juego” (p. 44). El diseño de los controles de un juego educativo plantea problemas específicos, debido a que esta característica no sólo tiene que ver con el impacto en la capacidad de juego, sino también si estas características se pueden utilizar para mejorar las capacidades de aprendizaje y habilidades específicas.

En conclusión, con el presente trabajo de investigación se pretende desarrollar un prototipo didáctico de un videojuego para el aprendizaje de estructuras secuenciales, selectivas e interactivas orientado a niños, dando especial importancia al modelo conceptual para el desarrollo de videojuegos.

3. Cronograma de actividades

Actividad	Semanas del mes de Octubre - Febrero 2021-2022																					
	Noviembre				Diciembre					Enero				Febrero				Marzo				
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22
Conceptualización																						
Desarrollo (Iterativo e Incremental)																						
Pruebas Beta																						
Corrección de errores de programación																						
Entrega del primer borrador de tesis																						
Correcciones																						
Entrega final																						
Presentación y defensa de tesis																						

Tabla 4. Cronograma de actividades.

4. Referencias

- Abt, C. (1970). *Serious Games*. (1ra. ed.). Viking Press.
- Acerenza, N, Coppes, A, Mesa, G, Viera, A, Fernández, E, Laurenzo, T & Vallespir, D. (2009). Una Metodología para desarrollo de videojuegos :versión extendida. *Reportes Técnicos*, 09-13. <https://hdl.handle.net/20.500.12008/3420>
- Álvarez, M. (2017). Desarrollo del pensamiento computacional en educación primaria: una experiencia educativa con Scratch. *Revista de Ciències de l'Educació*, (2). 45-64. <https://doi.org/10.17345/ute.2017.2.1820>
- App Annie & IDC. (2020). Consumo de videojuegos por plataforma [Gráfico]. App Annie. <https://www.appannie.com/en/insights/market-data/gaming-spotlight-2020-review/>
- Arenas, K. M. (2018). *Desarrollo de un Serious Gaming para estudiantes de Primer Año de Primaria aplicando la Metodología SUM*. [Tesis de ingeniería, Universidad Peruana Unión]. <http://hdl.handle.net/20.500.12840/2973>
- Basogain, X., Olabe, M. A., & Olabe, J. C. (2015). Pensamiento Computacional a través de la Programación: Paradigma de Aprendizaje. *Revista de Educación a Distancia*, (46). 1-33. <https://doi.org/10.6018/red/46/6>
- Compañ, P., Satorre, R., Llorens, F., & Molina, R. (2015). Enseñando a programar: un camino directo para desarrollar el pensamiento computacional. *Revista de Educación a Distancia*, (46). 1-15. <https://doi.org/10.6018/red/45/11>
- Empantallados.com (s.f.). Tipos de videojuegos, ¿cómo se clasifican?. <https://empantallados.com/tipos-de-videojuegos/>
- Frasca, G. (2001). Videogames of the oppressed: videogames as a means for critical thinking and debate. [Tesis de maestría, Georgia Institute of Technology]. <https://ludology.typepad.com/weblog/articles/thesis/FrascaThesisVideogames.pdf>

- International Society for Technology in Education, & Computer Science Teachers Association (2011). Operational Definition of Computational Thinking for K-12 Education. https://cdn.iste.org/www-root/Computational_Thinking_Operational_Definition_ISTE.pdf
- López, C. (2016). El videojuego como herramienta educativa. Posibilidades y problemáticas acerca de los serious games. *Apertura*, 8(1). 1-15. http://www.scielo.org.mx/scielo.php?script=sci_arttext&pid=S1665-61802016000200010
- Manquilef, G. F. (2021). *Implementación de ciclos y condicionales para el desarrollo del pensamiento computacional en videojuegos*. [Tesis de ingeniería, Universidad Andrés Bello]. http://repositorio.unab.cl/xmlui/bitstream/handle/ria/18313/a131787_Manquilef_G_Implementacion_de_ciclos_y_condicionales_2021_Tesis.pdf
- Márquez, G., Osorio, S., & Olvera, N. (2011). *Introducción a la programación estructurada en C* (1ra. ed.). México: Pearson.
- Montás, G. (2020). *Adicción a los videojuegos en tiempos de pandemia*. Insight. <https://insight.com.do/adiccion-a-los-videojuegos-en-tiempos-de-pandemia/>
- Murillo, X. A., Gutierrez, A., Ibañez, A. W., Quiroz, J. A., Sahonero, G., & Diaz, F. (2018). *Implementacion de la metodologia SUM modificada para el desarrollo de videojuegos orientados al aprendizaje en Bolivia*. 15th Ibero-American Symposium on Education, Cybernetics and Informatics, SIECI 2018. Orlando, Estados Unidos. https://www.researchgate.net/publication/329525865_Implementacion_de_la_metodologia_a_SUM_modificada_para_el_desarrollo_de_videojuegos_orientados_al_aprendizaje_en_Bolivia
- Rahman, H. G. (2017). *Videojuego educativo en 3D para dispositivos móviles Android, enfocado al aprendizaje de la Lógica de Programación para usuarios entre los 5 a 18 años de edad*. [Tesis de ingeniería. Universidad Técnica de Ambato]. <https://repositorio.uta.edu.ec/jspui/handle/123456789/25705>

- Ristić, O., Milošević, D., & Urošević, V. (2016). *The importance of programming languages in education*. TECHNICS AND INFORMATICS IN EDUCATION, 6th International Conference, Faculty of Technical Sciences. Čačak, Serbia. [http://www.ftn.kg.ac.rs/konferencije/tio2016/Radovi%20TIO%202016/EN/3\)%20Information%20and%20Educational%20Technologies/319_046_Ristic%20i%20sar_EN.pdf](http://www.ftn.kg.ac.rs/konferencije/tio2016/Radovi%20TIO%202016/EN/3)%20Information%20and%20Educational%20Technologies/319_046_Ristic%20i%20sar_EN.pdf)
- Ruíz, M. R. (2014). *Un modelo conceptual para el diseño de videojuegos educativos*. Tesis doctoral. Universidad Carlos III de Madrid. <http://hdl.handle.net/10016/20595>
- Sada, C. (2020). La pandemia y el auge en el uso del celular en México. *MediaLab*. <https://medialab.up.edu.mx/noticias/la-pandemia-y-el-auge-en-el-uso-del-celular-en-mexico/>
- Téllez, M. (2019). Pensamiento computacional: una competencia del siglo XXI. *Educación superior*, 6(1), 23-32. http://www.scielo.org.bo/pdf/escepies/v6n1/v6n1_a07.pdf
- The Royal Society. (2012). *Shut down or restart? The way forward for computing in UK schools*. (informe nro. 1). Londres, Reino Unido: Autor. <https://royalsociety.org/-/media/education/computing-in-schools/2012-01-12-computing-in-schools.pdf>
- Ticante, A. C., Herrera, C. M., Arguijo, P., Meléndez, R. Á., & Vázquez, A. H. (2019). Videojuego educativo para ayudar a comprender los principios básicos de la programación y desarrollar la habilidad lógica en niños de educación básica. *Research in Computing Science*, 148(7), 127–139. <https://doi.org/10.13053/rcs-148-7-10>
- Vallejo, D. & Martín, C. (2019). *Desarrollo de Videojuegos: Arquitectura del Motor de Videojuegos*. Bubok.
- Wing, J. M. (2006). Computational thinking. *Communications of the ACM*, 49(3), 33-35. <http://dx.doi.org/10.1145/1118178.1118215>
- Wolf, M. & Perron, B. (2005). Introducción a la teoría del videojuego. *Formats: revista de comunicación audiovisual*. (4). <https://raco.cat/index.php/Formats/article/view/257329>

- Zapata, M. (2015). Pensamiento computacional: Una nueva alfabetización digital. *Revista de Educación a Distancia*, (46). 1-47. <https://revistas.um.es/red/article/view/240321>
- Zatarain, R. (2018). Reconocimiento afectivo y gamificación aplicados al aprendizaje de Lógica algorítmica y programación. *Revista electrónica de investigación educativa*, 20(3), 115-125. <https://doi.org/10.24320/redie.2018.20.3.1636>
- Zhao, Z. (2017). *Videojuegos, educación y desarrollo*. [Tesis de doctorado, Universidad Autónoma de Madrid]. <http://hdl.handle.net/10486/680698>
- Zyda, M. (2005). From visual simulation to virtual reality to games. *Computer*, 38(9), 25-32. <https://doi.org/10.1109/MC.2005.297>