



FACULTAD DE INGENIERÍA

CARRERA DE INGENIERÍA DE SISTEMAS
COMPUTACIONALES

“DESARROLLO DE UN VIDEOJUEGO EDUCATIVO
BASADO EN RNA PARA FACILITAR LA EVALUACIÓN
DEL CURSO DE MATEMÁTICA DEL 4TO. GRADO DE
PRIMARIA”

Tesis para optar el título profesional de:

Ingeniero de Sistemas Computacionales

Autores:

Giancarlo Daniel Calderón Pereda
Diego Enzo Figueroa Rodríguez

Asesor:

Mg. José Alberto Gómez Avila

Trujillo – Perú
2017

APROBACIÓN DE LA TESIS

El asesor y los miembros del jurado evaluador asignados, **APRUEBAN** la tesis desarrollada por los Bachiller **Calderón Pereda Giancarlo Daniel** y **Figueroa Rodríguez Diego Enzo**, denominada:

**“DESARROLLO DE UN VIDEOJUEGO EDUCATIVO BASADO EN RNA PARA
FACILITAR LA EVALUACIÓN DEL CURSO DE MATEMÁTICA DEL 4TO.
GRADO DE PRIMARIA”**

Ing. José Alberto Gómez Avila
ASESOR

Ing. Lourdes Roxana Díaz Amaya

**JURADO
PRESIDENTE**

Ing. Lain Jardiel Cárdenas Escalante
JURADO

Ing. Rolando Javier Berru Beltrán
JURADO

DEDICATORIA

Dedicamos la presente tesis a nuestros padres que nos ayudaron a superar todas las brechas en nuestra etapa universitaria, a los escolares del Perú dado que el software está hecho para ayudarles a tener una mejor experiencia en sus evaluaciones, y a los docentes que usarán el software ya que se tiene el objetivo de facilitar sus labores.

AGRADECIMIENTO

Agradecemos a nuestro asesor por todo el tiempo y esfuerzo que dedicó a la tesis con nosotros. A nuestros profesores de la carrera quienes a pesar de las dificultades y retos que nos pusieron, lograron encaminarnos por el sendero del éxito. Al Dr. Víctor Burgos por ayudarnos con la parte educativa de la investigación y los docentes del colegio Niños en Acción por su tiempo para probar el sistema.

ÍNDICE DE CONTENIDOS

Contenido

<u>APROBACIÓN DE LA TESIS</u>	ii
<u>DEDICATORIA</u>	iii
<u>AGRADECIMIENTO</u>	iv
<u>ÍNDICE DE CONTENIDOS</u>	v
<u>ÍNDICE DE TABLAS</u>	vi
<u>ÍNDICE DE FIGURAS</u>	vii
<u>RESUMEN</u>	viii
<u>ABSTRACT</u>	ix
CAPÍTULO 1. INTRODUCCIÓN	10
CAPÍTULO 2. MARCO TEÓRICO	13
2.1.1. <i>Redes Neuronales Artificiales</i>	15
2.1.2. <i>Evaluación del aprendizaje</i>	19
2.1.3. <i>Videojuegos</i>	21
2.1.4. <i>Mejores prácticas en Aseguramiento de la Calidad / Testeo</i>	24
2.1.5. <i>Metodologías de desarrollo</i>	25
CAPÍTULO 3. METODOLOGÍA	29
3.1. Operacionalización de variables	29
3.2. Diseño de investigación	32
3.3. Unidad de estudio	32
3.4. Población	32
3.5. Muestra (muestreo o selección)	32
3.6. Técnicas, instrumentos y procedimientos de recolección de datos	32
3.7. Métodos, instrumentos y procedimientos de análisis de datos	33
CAPÍTULO 4. RESULTADOS	34
4.1. CONTRASTACIÓN	34
CAPÍTULO 5. DISCUSIÓN	44
CONCLUSIONES	46
RECOMENDACIONES	46
REFERENCIAS	47
ANEXOS	50

ÍNDICE DE TABLAS

Contenido

Tabla 1 Cuadro comparativo de las RNA según su aprendizaje.....	18
Tabla 2 Cuadro comparativo de metodologías de desarrollo de software	28
Tabla 3 Operacionalización de la variable independiente.....	29
Tabla 4 Operacionalización de la variable dependiente	31
Tabla 5 Recolección de datos.....	32
Tabla 6 Análisis de datos	33
Tabla 7 Prueba de Hipótesis	34
Tabla 8 Resultados Indicador Tiempo de corrección	35
Tabla 9 Resultados Indicador Estructuración de la información	38
Tabla 10 Resultados Indicador Demanda de preparación de examen	41
Tabla 11 Discusión Indicador Tiempo de corrección	44
Tabla 12 Discusión Indicador Estructuración de la información.....	44
Tabla 13 Discusión Indicador Demanda de preparación de examen	45
Tablas A1 – A45 (Anexos)	50

ÍNDICE DE FIGURAS

Contenido

Figura N°1. Representación gráfica de un Perceptrón Simple	16
Figura N°2. Representación gráfica de un Perceptrón Multicapa.....	17
Figura N°3. La evaluación del aprendizaje	20
Figura N°4. Gráfico Resultados Indicador Tiempo de corrección	37
Figura N°5. Gráfico Resultados Indicador Estructuración de la información	40
Figura N°6. Gráfico Resultados Indicador Demanda de preparación de examen.....	43
Figuras A1 – A42 (Anexos).....	50

RESUMEN

El presente trabajo de investigación tiene como propósito facilitar el proceso de evaluación de los docentes mediante el uso de un videojuego educativo basado en redes neuronales artificiales, dirigido a escolares de 4 grado de nivel primaria del colegio Niños en Acción de la ciudad de Trujillo, cumpliendo satisfactoriamente los objetivos específicos de la siguiente forma:

Se logró determinar el tiempo promedio de evaluación por los docentes del curso de matemáticas 4to grado de primaria, reduciendo en un 100 %.

Se logró determinar la correcta gestión de información de la evaluación de los escolares en el curso de matemática, en incremento de 169,14 %.

Se logró calcular el tiempo promedio de elaboración de las evaluaciones por el docente en el curso de matemática, reduciendo en un 33,33 %.

Palabras clave: LibGDX, videojuego, evaluación, extreme programming.

ABSTRACT

This research aims to facilitate the process of evaluation of teachers through the use of an educational video game based on artificial neural networks, aimed to primary school students of 4th grade at the school “Niños en Acción” in the city of Trujillo, satisfying the specific objectives as follows:

It was possible to determine the average time of evaluation by the teachers of the math course of 4th grade of primary, reducing by 100%.

It was possible to determine the correct information management of the evaluation of the students in the mathematics course, by an increase of 169.14%.

It was possible to calculate the average time of elaboration of the evaluations by the teacher in the mathematics course, reducing by 33.33%.

Keywords: LibGDX, video game, evaluation, extreme programming

CAPÍTULO 1. INTRODUCCIÓN

1.1. Realidad problemática

En la actualidad, la evaluación tradicional tal y como la conocemos no cumple un verdadero rol educativo, como lo exponen Condemarín y Medina (2000), agregando que “las prácticas tradicionales de evaluación tienden a comparar los resultados de los estudiantes, distribuyéndolos en una escala de buenos, regulares y malos”, en lugar de preocuparse en trabajar con las diferencias y particularidades de cada estudiante. En investigaciones como la de Vaillant (2013) se menciona que este hecho viene sucediendo en nuestro país, y se presenta a las TICs (Tecnologías de Información y Computación) como un medio viable para tratar de cambiar esta realidad, pero señala también que no se cuenta con una gran documentación sobre integración de las TICs u otras innovaciones en la formación docente en el Perú.

El uso de las tecnologías de información en la actualidad es tan amplio que se han logrado incorporar, tal y como afirma Cabrol & Severin (2010), en espacios educativos, donde si bien no existe el desarrollo riguroso de estas herramientas en nuestra región, se destaca el potencial que conllevan las TIC debidamente implementadas como una buena alternativa para hacer posible el acceso a cada vez más información y conocimientos, propiciando que la calidad de la educación mejore. Kaztman (2010) señala el hecho de que las tecnologías de información no son de amplio uso en la educación en la región Latinoamericana, lo cual pone de manifiesto una brecha considerable con los países desarrollados, y ya que si bien el acceso a la tecnología tanto en hogares o en instituciones educativas no llega a ser masivo, se buscan universalizar el acceso al mundo digital con lo que se tiene al alcance, y es la escuela una de las instituciones que debe hacerse cargo de liderar este cambio en la educación.

Sin embargo, como indica Vaillant (2013), el acceso a las TICs por parte de los docentes no asegura que le den un uso efectivo, ya que para que ello ocurra, es necesario una constante formación de los docentes en estas tecnologías y disponer cómo es que se integrarán en su labor, considerando que esta integración está fuertemente asociada con el interés que posean para emplearlas. Cardemil et al. (2011) destaca la importancia que posee el rol del docente en el uso de las TICs en el aula, ya que estos recursos o herramientas perderían todo su potencial si se desconoce el papel que desempeñan. Es la concepción de la enseñanza uno de “los principales factores del docente que limitan la posibilidad de transformar sus prácticas pedagógicas”, puesto que aquellos quienes tengan una visión más tradicional de enseñanza tienden a no incorporar las TICs en su práctica, mientras que aquellas con una “concepción más constructivista tienden a utilizarlas más frecuentemente promoviendo el trabajo autónomo tanto como el colaborativo en sus alumnos”.

Gómez y Macedo (2010) sostienen que el uso de las TICs en la educación, además de representar una herramienta que permite el desarrollo de diversas capacidades cognitivas del escolar favoreciendo así su proceso de aprendizaje, constituyen una gran ayuda para los

docentes ya que actúan como instrumentos procesadores de información y de gestión administrativa, lo cual les da la posibilidad de establecer un mayor acercamiento con sus alumnos y conocer de cerca sus progresos individuales, asimismo exime al docente de tareas repetitivas y rutinaria, facilitan la evaluación, y constituyen un buen medio de seguimiento detallado al tener almacenado el recorrido que le tomó a cada escolar para llegar a sus respuestas.

Entre las herramientas educativas existentes al día de hoy encontramos a e-Adventure, desarrollada por Pérez Padilla et al. (2007), el cual es un motor de aventuras gráficas con fines educacionales que permite a los docentes crear su propio videojuego sin la necesidad de que posean conocimientos avanzados en informática, estableciendo qué elementos, personajes, situaciones y ambientes se harán presentes, de entre un abanico de diversas opciones. Al ser videojuegos creados sólo en base a imágenes se presentan restricciones en interacción, siendo necesario tener personajes y elementos que se muevan libremente y que se adapten según el nivel alcanzado por el usuario para evitar que el videojuego se torne repetitivo, canalizando adecuadamente la información que quiere presentar el docente; y en la creación de contenido, llegando a ser una tarea un poco desordenada, tediosa y lenta, al tener que ir generando y agrupando imágenes para cada contenido nuevo, por lo que es necesario que la herramienta facilite este aspecto favoreciendo a que la elaboración del contenido suceda en un tiempo prudencial. Además, el e-Adventure posee un sistema de gestión de aprendizaje el cual permite el seguimiento de avances, evaluación y posterior retroalimentación. En este punto, es necesario tener una forma más precisa y en tiempo real de los avances de los estudiantes, para conocer hasta qué punto la evaluación se realiza de manera correcta, y no depender de sistemas externos a la herramienta educativa. Es así que, a partir de las necesidades identificadas y al entrevistar a docentes del curso de matemáticas, se determinaron criterios u objetos a evaluar como: demanda de trabajo y preparación, tiempo de corrección y estructuración de la información.

Al evidenciarse que es posible recurrir a las TICs como herramientas que ayuden en la labor docentes, se toma en consideración para desarrollarlas a la Inteligencia Artificial, la cual según Russel y Norvig (2004), es un campo bastante amplio al cual han contribuido para su desarrollo diversas disciplinas como filosofía, matemáticas, economía, neurociencias, psicología, lingüística y la ingeniería computacional, buscando desarrollar agentes o sistemas inteligentes que al recibir señales externas del medio ambiente sean capaces de responder desde un punto de vista racional, es decir que realice lo correcto en base a su conocimiento. Entonces, la IA tiene un amplio rango de acción, los cuales a través de diferentes estudios académicos se demuestra su utilidad y aplicación en áreas como la robótica, biométrica, sistemas financieros, medicina o en la educación.

1.2. Formulación del problema

¿De qué manera el desarrollo de un videojuego educativo basado en Redes Neuronales Artificiales facilita la evaluación a los docentes en el curso de matemática del 4to. grado de nivel primaria de la institución educativa Niños en Acción en la ciudad de Trujillo?

1.3. Justificación

Para fundamentar la necesidad de realizar este trabajo de investigación, se consideran muchos elementos. Existen softwares educativos, pero muchos se limitan a exponer la información como si de un libro de texto se tratara, es por ello que se cree conveniente aportar al campo educativo realizando una aplicación de este tipo con técnicas de programación como son las redes neuronales.

Además, es pertinente utilizar las tecnologías disponibles para ayudar al docente en su labor, muchas veces sobrecargada, brindándole una aplicación de apoyo que complemente lo hecho en aula, propiciando participación más activa de los escolares, constituyendo esto una justificación aplicativa o práctica.

Del mismo modo, se puede señalar que si bien el producto de este proyecto es en principio una aplicación educativa, se presentará en forma de videojuego, el cual posteriormente usarán los escolares, representando una forma novedosa y dinámica para el docente al realizar el seguimiento del progreso de sus escolares, estableciendo así una justificación valorativa.

Finalmente, como justificación académica, cabe destacar el hecho de que no se le ha dado a las redes neuronales artificiales un amplio uso en el campo de la enseñanza, sino que la mayor parte de investigaciones se dirigen a lo que respecta a pronósticos y reconocimiento de imágenes o de manuscritos.

1.4. Limitaciones

- Poca disponibilidad de instituciones educativas para la aplicación de la investigación. Se superó la limitación exponiendo una demo del videojuego al momento de presentarles nuestra propuesta a las instituciones.
- Realizar la investigación dentro de un determinado espacio de tiempo para no interferir con épocas de evaluaciones y actividades internas de la institución educativa. Se superó la limitación coordinando con el área académica de la institución las fechas de visita durante el periodo escolar.

1.5. Objetivos

1.5.1. Objetivo general

Facilitar la evaluación realizada por los docentes del curso de matemática del 4to grado de nivel primaria de la institución educativa Niños en Acción en la ciudad de Trujillo, mediante el desarrollo de un videojuego educativo basado en RNA.

1.5.2. Objetivos específicos

- Determinar el tiempo promedio de evaluación por los docentes del curso de matemáticas 4to grado de primaria
- Determinar la correcta gestión de información de la evaluación de los escolares en el curso de matemática
- Calcular el tiempo promedio de elaboración de las evaluaciones por el docente en el curso de matemática

CAPÍTULO 2. MARCO TEÓRICO

a) Antecedentes

Como primer antecedente tenemos la investigación “Evaluación y diseño de videojuegos: generando objetos de aprendizaje en comunidades de práctica” de la Dra. Esther del Moral Pérez et.al (2012), la cual se centra en determinar si el uso de un videojuego educativo facilita la labor de los profesores. La herramienta sometida a análisis es “e-Adventure”, desarrollada por Pérez Padilla et al. (2007), la cual demuestra cómo insertar videojuegos en el aula desde el punto de vista educativo. La experiencia que se tuvo fue bastante alentadora en sus conclusiones, sin embargo se encontraron problemas como el hecho de que a pesar de que sea una herramienta poderosa, su uso inadecuado podría pervertir su fin, por ende es necesario que el docente asegure una metodología de trabajo coherente con el videojuego educativo. Otra de las dificultades que se encontró fue la del tiempo que le toma al docente crear una línea de juego para que los alumnos puedan utilizarlo. Cabe resaltar que, al menos los videojuegos por sí mismos no generan una mejora en los procesos cognitivos ni en la calidad educativa, sino que dependen de la labor educativa y planificación del docente para orientarla de la mejor manera, esto quiere decir que el desempeño de los alumnos al usar el videojuego depende completamente de cómo el docente prepara el material. A pesar de estos aspectos, también se concluyó que al incluir elementos audiovisuales en el proceso educativo se favorece un aprendizaje más constructivista, el cual permite a los alumnos tener su propia manera de resolver los problemas, en vez de enseñarles un único camino posible. De igual manera, los autores de la investigación recomiendan que la creación de videojuegos educativos como e-Adventure y su posterior publicación en comunidades supone una gran ventaja, ya que permite recolectar experiencias, ideas, mejoras y recursos (retroalimentación), asegurando que el software se encuentre en constante mejora para, alcanzando así una correcta incorporación del mismo en el aula.

El siguiente antecedente “Análisis, Diseño e implementación de un videojuego en 2D orientado a la ejercitación de la memoria y el desarrollo de la inteligencia espacial” es obra de Quispe (2013), de la Pontificia Universidad Católica del Perú, en la ciudad de Lima, el cual busca promover el desarrollo de las habilidades cognoscitivas de las personas por medio de un videojuego. El objetivo es perfeccionar las habilidades y destrezas a nivel cerebral, abarcando aspectos como el de la mejora de la memoria, atención, velocidad mental y la resolución de problemas. El autor de esta tesis toma los datos de la prueba PISA 2009, la cual evidencia el bajo rendimiento de los escolares peruanos en comparación al resto del mundo, así como las dificultades en sus competencias cognitivas puesto que no comprenden lo que leen y no tienen la capacidad de aplicar sus conocimientos de manera correcta. Por ello, propone una herramienta con un enfoque lúdico para apoyar el ejercicio de la memoria, el razonamiento lógico, y desarrollar habilidades cognitivas referentes a la inteligencia espacial. Se desarrolló el proyecto desde dos enfoques, uno orientado a la gestión del proyecto, tomando como modelo el Pmbok, y otro orientado al desarrollo del software, la metodología RUP. Como resultado, se obtuvo un correcto desarrollo del videojuego, implementando algoritmos de Inteligencia artificial, como son: búsqueda A* para manejar el movimiento, máquinas de estado finitos para toma de decisiones, y algoritmos evolutivos para las estrategias de los personajes en el videojuego. En conclusión, se desarrolló un videojuego implementado con algoritmos de inteligencia artificial, verificando su factibilidad y eficiencia. Este trabajo brinda información básica necesaria para la realización de un videojuego al usar distintas técnicas de Inteligencia Artificial.

La investigación “Implementation of a Computerized Assessment System by using Backpropagation Neural Networks with R and Shiny”, desarrollada por Casafranca y Gutiérrez (2015), de la Universidad del Pacífico, en la ciudad de Lima, pone de manifiesto uno de los problemas a los que se enfrentan muchos estudiantes durante su etapa universitaria como es el manejo del material de estudio, por lo cual se propone un software desarrollado con redes neuronales artificiales para evaluar de una mejor manera a los alumnos en temas de informática básica puesto que determinaron que las personas aprenden a distintos ritmos. El objetivo es que mediante el uso de algoritmos y técnicas de ciencias computacionales unidos a las posibilidades predictivas que ofrecen las RNA esta herramienta de evaluación ajuste automáticamente la dificultad de los ejercicios, brindándoles a los estudiantes una forma de evaluación más dinámica y de retroalimentación inmediata, para así no desmotivarlos y estimularlos a estudiar más. Los resultados fueron los esperados por los autores, debido a que con el uso de su sistema de evaluación automático de RNA con entrenamiento back propagation, y considerando aspectos como el de nivel de dificultad alcanzado, calificaciones, y tiempo utilizado para resolver las preguntas, se mostró que los alumnos que utilizaron el sistema lograron un incremento en el logro de los objetivos del curso donde fue aplicado. Como aporte importante de este trabajo se debe señalar los componentes que se consideran para evaluar el aprendizaje, y que la RNA la red comienza a aprender en qué casos debe cambiar la dificultad utilizando como entradas la

dificultad alcanzada, tiempo de resolución y el puntaje total. Esta red neuronal multicapa es de tipo supervisada ya que es el docente quien brinda los casos de entrenamiento con ajustes mediante retro propagación para que la red haga estimaciones a futuro. Los autores recomendaron que en trabajos posteriores, como el presente, se puedan aplicar redes neuronales artificiales agregando más valores de entrada, dejando libertad para que se puedan aplicar en distintas áreas como matemática, historia, ciencias, etc., además de plantearlo de una manera más amigable y dinámica para los alumnos.

b) Bases teóricas

Como bases teóricas para la sustentación de la presente tesis, se realizó un estudio de Redes Neuronales Artificiales (RNA), recalcando qué son, su clasificación y cómo se desarrollan. Además, como se planteó desde un comienzo que sería un software educativo en forma de videojuego, es pertinente especificar qué importancia tienen las metodologías de desarrollo así como las métricas de calidad.

2.1.1. Redes Neuronales Artificiales

2.1.1.1. Fundamentos de Inteligencia Artificial

Dentro del área de Inteligencia Artificial, existe un paradigma llamado Redes Neuronales Artificiales (RNA). García Serrano (2013) explica que las RNA son un intento de emular la forma de trabajar del cerebro humano tomando como base el funcionamiento de las neuronas y sus conexiones, analizando un conjunto de datos (entradas), al igual como lo hacen las personas para tomar una decisión. Por ejemplo, si un joven quiere escoger una carrera universitaria, tendrá que conocer sus fortalezas y habilidades, qué le gusta hacer, la demanda de carreras, etc. Dependiendo de aquellos datos de entrada, se podrá determinar que carreras van de acuerdo al perfil del joven.

2.1.1.2. Predicción de las Redes Neuronales Artificiales

Una de las aplicaciones de las RNA son la de predecir posibles soluciones dependiendo de una serie de datos. Lee (1989) cita la investigación de Lapedes y Farber del Laboratorio de Investigación de los Álamos, quienes demuestran que las RNA aplicadas a la predicción pueden lograr una mayor exactitud numérica que otras técnicas, además de brindar una alta tasa de éxito en situaciones caóticas mediante backpropagation (técnica la cual se explicará más adelante). Retomando el ejemplo del joven que quiere escoger su carrera universitaria, mediante una RNA sería posible predecir qué carreras son las que mejor se ajustan a su perfil, e incluso la probabilidad de éxito si llegara a estudiar dicha carrera.

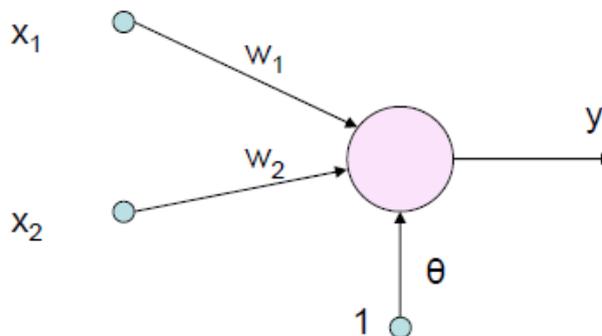
2.1.1.3. Elementos de las Redes Neuronales Artificiales

Como se mencionó anteriormente, las RNA tratan de imitar el funcionamiento del cerebro humano, por lo que García Serrano (2013) explica que según la neurociencia actual, las neuronas funcionan como transistores (encendido o apagado), con el agregado de que las prendidas pueden estar poco activas o muy activas y que estas neuronas se disparan en función a las dendritas (entradas). En caso que las dendritas lleguen a un umbral determinado, la neurona se activa y envía señales a otras neuronas mediante los axones (salidas). Las RNA tienen casi el mismo funcionamiento, tienen entradas, y de acuerdo al nivel de excitación del perceptrón (neurona), este brinda una respuesta apropiada.

2.1.1.3.1. Perceptrón

Galván e Isasi (2004), se refieren al perceptrón como la unidad de las RNA, siendo la emulación de una neurona. Estas a su vez poseen entradas (X), pesos por cada entrada (W), un umbral de disparo (θ) y una salida (Z o Y). En la imagen inferior se puede apreciar de una mejor manera cada una de las partes del perceptrón simple.

Figura N°1. Representación gráfica de un Perceptrón Simple



Fuente: Página web del Laboratorio del Departamento de Informática de la Universidad Carlos III de Madrid

Dependiendo de los pesos, las entradas tendrán mayor relevancia para que así el umbral se excite. La fórmula para calcular la salida está dada por la sumatoria de las entradas con sus respectivos pesos, es decir $z = \sum_i w_i * x_i$ (García Serrano, 2013).

Diversos autores, como Galván e Isasi (2004), Martín del Brío y Sanz (2006), y García Serrano (2013), explican que dicha fórmula tiene como objetivo verificar si se pudo disparar el umbral θ , de forma matemática se puede expresar como una condición:

Si $\sum > \theta$ entonces: $z = 1$; y , Si $\sum < \theta$ entonces: $z = 0$

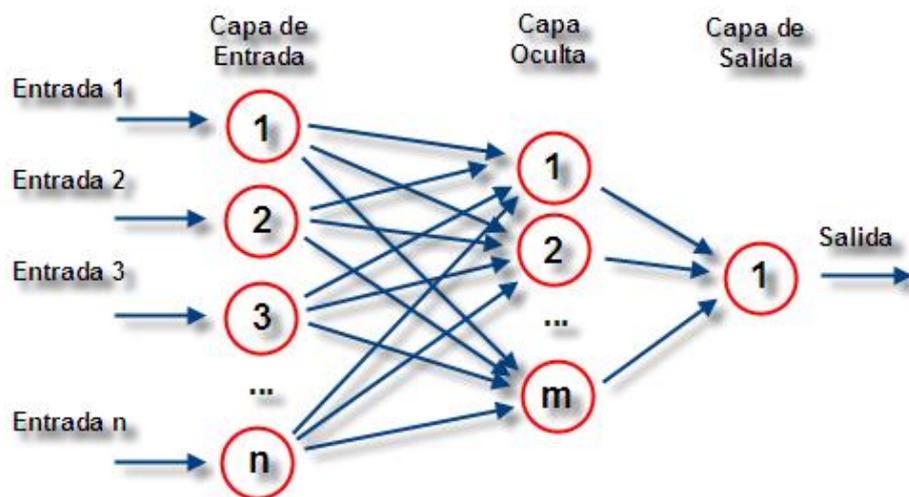
La función de \sum sumatoria en caso logre alcanzar el umbral objetivo θ , dará como resultado la activación de la neurona y por consecuencia una respuesta positiva a z. Existirán veces que no se logrará dicho objetivo y en tales situaciones la salida será distinta. Por ejemplo si uno desea

saber si debería ir al doctor, si se llega a un nivel de malestar que supere las expectativas, entonces se procederá a ir, de lo contrario se optará por una solución diferente. Matricialmente la misma fórmula puede expresarse en $z = x * w$.

2.1.1.3.2. Capas

Galván e Isasi (2004) mencionan que las RNA son como un conjunto de grafos ponderados cuyos nodos se subdividen en tres grupos, los de entrada, salida y los intermedios. Las capas de una RNA le permiten tener una mayor cantidad de condiciones, perfeccionando la red.

Figura N°2. Representación gráfica de un Perceptrón Multicapa



Fuente: Página web del Laboratorio del Departamento de Informática de la Universidad Carlos III de Madrid

2.1.1.4. Clasificación de las Redes Neuronales Artificiales

Según Martín del Brío y Sanz (2006) se pueden clasificar a las RNA según su arquitectura y según su aprendizaje.

2.1.1.4.1 Según su Arquitectura

- Redes Monocapa: Como su mismo nombre lo dice, son redes neuronales de una sola capa la cual intercambia señales entre ellas, siendo algunas entradas la salida de otros nodos. Hopfield citado por Galván e Isasi (2004), propuso en la década de los 80 un modelo de neuronas no lineal encerrado en una sola capa donde todas las neuronas están conectadas con todas las demás, siendo

la salida de unas, la entrada del siguiente nodo. Ejemplo: Perceptrón simple, Adaline, Red de Hopfield (considerada también como red recurrente)

- Redes Multicapa: Son redes de muchas capas las cuales pueden dividirse en las de entrada, las ocultas o intermedias y las de salida. García Serrano (2013) las menciona como clave en el desarrollo de aprendizajes supervisados mediante el algoritmo de back propagation. Ejemplo: Perceptrón multicapa.

2.1.1.4.2 Según su aprendizaje

Galván e Isasi (2004) explican ampliamente que existen 2 tipos de aprendizaje para entrenar a las redes neuronales.

Tabla N°1. Cuadro comparativo de las RNA según su aprendizaje

Supervisado	No Supervisado
Está presente la supervisión humana para garantizar de que las respuestas emitidas son correctas o no.	No necesitan de intervención humana, tampoco de un conjunto de datos cuya respuesta ya se conoce.
Utilizan un conjunto de datos con respuestas predefinidas para adaptar los pesos, error y el umbral.	Conforme más datos se agregan a la red, esta va aprendiendo nuevos escenarios.
Usualmente son Multicapa	Usualmente son Monocapa
Determinan su salida por excitación del perceptrón, pesos de cada entrada, error y umbral establecidos después del entrenamiento.	Determinan su salida por familiaridad, componentes principales, agrupamiento u otras posibilidades que se desee integrar.
Ejemplos: Algoritmo de Retropropagación, Algoritmos estadísticos, etc.	Ejemplos: Algoritmo de agrupamiento, Redes de Kohonen, etc.
Híbrido: No es hablado mucho en los libros teóricos dado que es un modelo nacido de la combinación del aprendizaje supervisado y el no supervisado para brindar soluciones más precisas. Usa parte del aprendizaje supervisado en un parte del algoritmo y no supervisado en la otra.	

Fuente: Datos tomados de la investigación de Galván e Isasi (2004)

2.1.1.5. Metodología de desarrollo de una Red Neuronal Artificial

En el desarrollo de las redes neuronales artificiales se pasa por 4 importantes pasos, los cuales, según autores como García Serrano (2013) y Fausett (1994) son:

- Definición de la RNA: Una vez determinada el tipo de red neuronal que se utilizará, se define: las entradas, las salidas, las capas, los pesos w , los umbrales iniciales y otros datos relevantes en la RNA.
- Entrenamiento: Puede ser supervisado o no supervisado, el entrenamiento sirve para ajustar los pesos $w_1 \dots w_n$ y el umbral θ . Con estos datos ajustados a lo real, se podrá estimar con mayor precisión las respuestas.
- Aprendizaje: fase de ejecución de la red, donde de acuerdo al entrenamiento que se tuvo, se muestran las respuestas a las entradas ingresadas en un entorno real. Sin embargo, a medida que transcurre el tiempo podrán surgir nuevos conjuntos de entradas reales aún desconocidas por la RNA, y por consiguiente será necesario un aprendizaje.

2.1.2. Evaluación del aprendizaje

En este punto se detalla lo concerniente a la evaluación del aprendizaje, abarcando el propósito, su objeto, y el método de evaluación, todo ello tomando como referencia el libro de Aguilera et al. (2011).

2.1.2.1. Objeto del Proceso ¿Que evaluar?

Todo parte de la definición de los conocimientos, habilidades y actitudes circunscritos en los objetivos de aprendizaje que se persiguen, lo cual trae una implicación directa en la “planeación de la enseñanza y cómo es que se comunican estos objetivos tanto a alumnos, padres de familia y miembros de la comunidad escolar, además de la retroalimentación sobre los procesos de enseñanza aprendizaje”.

Para el caso específico de las Matemáticas, y en base al nivel de complejidad, los aspectos tomados en cuenta son: escribir definiciones de conceptos matemáticos y fórmulas, y resolución de operaciones; resolución de problemas con contengan operaciones; y la creación de problemas.

2.1.2.2. Finalidad ¿Para qué evaluar?

Es a partir de los propósitos que el docente decide qué debe evaluar en sus alumnos, además de los métodos de evaluación y de qué forma se comunican los resultados. Estos propósitos son las metas que se quieren conseguir o “productos a obtener”, los cuales se transforman en “una razón o motivación para actuar”.

Se pueden identificar dos propósitos a nivel general: la mejora del proceso educativo; e informar a los actores del proceso los logros obtenidos. En un plano más específico se

pueden mencionar los siguientes propósitos: identificar problemas en el aprovechamiento, valorar lo que se ha aprendido, saber cómo apoyar, planear y conducir las clases, asignar calificaciones y brindar retroalimentación.

En resumen, los docentes consideran que la el propósito de la evaluación es la “identificación de áreas de oportunidad, para generar acciones de mejora tanto en el aprendizaje del alumno como en las prácticas de enseñanza” del docente.

2.1.2.3. Control del Proceso ¿Cómo evaluar?

Las consideraciones que se deben tomar para realizar la evaluación deben ser precisas, es decir, válidas y confiables. Por lo cual, y de acuerdo con la INEE, el docente debe tomar en cuenta qué evaluar (tipos de objetivos de aprendizaje), cómo evaluar (método de evaluación) y la relación entre el qué y el cómo, es decir, que el método tome en consideración el objetivo de evaluación y su nivel de complejidad. Para que esto ocurra, el docente debe “identificar el tipo y nivel de complejidad de cada objetivo de aprendizaje que pretende conseguir”.

Figura N°3. La evaluación del aprendizaje



Fuente: Adaptado a partir de Stiggins, et al., 2007.

Fuente: Imagen tomada del libro “Evaluación de los aprendizajes en el aula”, obra de Aguilera et al. (2011)

2.1.3. Videojuegos

2.1.3.1. ¿Qué son los videojuegos?

Actualmente el videojuego es considerado desde una forma de trabajo hasta algo lúdico e inclusive un arte. De acuerdo a lo citado por Acerenza et al. (2009), quienes se apoyan en el trabajo de Wolf, un videojuego es tanto una herramienta potencial para la educación como también un objeto de estudio para la psicología del comportamiento; un medio para la interacción social, un juguete y un medio de distracción. Asimismo, aseveran que el campo emergente de la teoría del videojuego también constituye muchos enfoques en distintas áreas como el arte, la narratología, la estética, la programación, la inteligencia artificial y la semiótica. Todos estos enfoques convergen en un solo resultado el cual les permite a las personas y desarrolladores expresar su creatividad en su máximo esplendor. Los videojuegos también varían en tipos, género, rating y modelo de ingresos, con el fin de entretener, motivar a los jugadores a resolver nuevos acertijos, problemas o enemigos finales, ejercitando su mente y haciéndoles pasar un buen rato en el proceso.

2.1.3.2. Tipos de Videojuegos

De acuerdo con la información recogida en el trabajo de Acerenza et al. (2009), algunos tipos de videojuego que son de incumbencia para el presente trabajo de tesis y se desarrollaran en qué consisten, se describe su modelo de ingresos y plataformas que se suelen utilizar, así como su público objetivo. Estas descripciones se basan en la International Game Developers Association IGDA.

- Casuales: Son aquellos videojuegos que son para pasar el rato. Se caracterizan por ser fáciles de aprender y de jugar pero suelen ser altamente adictivos por saber captar la atención y compromiso del jugador. Las plataformas principales de estos juegos “casuales” son los dispositivos móviles. Extrañamente el público objetivo de estos videojuegos son hombres y mujeres entre los 35 y 65 años pues buscan algo fácil de jugar, sin embargo este rango está ampliándose cada día más puesto que los jóvenes y niños tienen mayor acceso a dispositivos móviles y les resulta altamente adictivo este tipo de juegos. Estos juegos usan el modelo Suscripción y Advergaming como modelo de ingreso.
- Educativos: Estos videojuegos tienen el fin de educar, entrenar las habilidades de rapidez mental y mejorar la forma de razonar de las personas. Los géneros varían entre plataformas, simuladores, etc. Teniendo en cuenta el género por excelencia de este tipo de juegos, los Puzle. Cabe resaltar que el presente trabajo tomará este tipo de videojuegos pues el fin del mismo será de determinar su influencia en el rendimiento

académico de escolares. No existe un público objetivo específico para este tipo ya que se han venido realizando juegos educativos para todas las edades, desde simuladores espaciales para astronautas hasta juegos para mejorar la capacidad cognitiva como es el Profesor Layton para el Nintendo DS. En éste caso ya se tiene especificado el público objetivo en la variable de estudio, el cual son los escolares.

- Hardcore: No podemos dejar de lado a los videojuegos que han marcado no sólo la infancia de algunos sino de generaciones enteras. Este tipo de juegos se caracteriza por ser más difíciles que los casuales, con más complejidad en los controles, en la trama y en los personajes, brindando así una experiencia única y completa al jugador. Las consolas principales son las videoconsolas y la PC, puesto que requieren mayor capacidad y RAM que un dispositivo móvil. El público objetivo de estos juegos son los jóvenes entre 18 y 34 años aunque en el mundo real sean jugados por personas de todas las edades, desde niños hasta adultos.

2.1.3.3. Roles en el desarrollo de videojuegos

En muchos libros sobre desarrollo de videojuegos se hace referencia a programadores, diseñadores, compositores, etc. Sin embargo, Rogers (2010) muestra de manera sencilla los roles que realmente son desempeñados en el desarrollo de videojuegos, siendo pionero en el negocio pues fue uno de los diseñadores del mundialmente conocido videojuego Pac-man.

2.1.3.3.1 Programador

Es uno de los roles más importante del desarrollo de videojuegos, ya que sin él no se podría desarrollar el juego en mente por los diseñadores y gestores del proyecto. Mediante los lenguajes de C++, C# o Java, le da vida a los personajes mediante código y técnicas de programación. En cuanto a la programación del juego, el programador crea el sistema de movimiento de cámaras, el mundo de juego, los menús principales, los objetos, la física del juego e inclusive la inteligencia artificial de los enemigos; por ende deben tener buenos conocimientos de matemáticas discretas, física y razonamiento matemático.

2.1.3.3.2 Artista

Los artistas tienen el deber de crear el arte conceptual y de plasmar las ideas del equipo en personajes y dibujos que más adelante se moldearán y diseñarán mejor. Los sub roles que ejercen estas personas son:

- Artista conceptual, el cual se dedica a crear el arte conceptual (personajes, enemigos, etc.) del cual se extraerán las mejores ideas.
- Artista del guión gráfico, quien realiza las posibles historias que tendrá el videojuego.

2.1.3.3.3 Diseñador

Su tarea es la de crear las ideas y reglas que abarca un juego. Debe saber diferenciar entre lo que hace bueno, o no, a un videojuego, y saber el porqué, es decir, saber sustentar sus ideas.

- Diseñador de nivel: crean los mapas, su estructura, los elementos que interactúan dentro de ellos, como los personajes y enemigos.
- Diseñador del sistema: quien desarrolla cómo se interrelacionan los elementos del juego.
- Scripter (Guionista): mediante herramientas escriben código que permite que ocurran acciones dentro del juego.
- Diseñadores de combate: especializados en el combate del jugador contra el enemigo, y de balancear la experiencia del jugador.
- Director creativo: quien mantiene la visión general del juego al supervisar el trabajo de los demás diseñadores, así como de dar sugerencias para mejorar su trabajo.

2.1.3.3.4 Productor

Quien ve el desarrollo completo del videojuego. Se encarga de contratar y formar los equipos, escribir contratos, contribuir al diseño del juego, administrar el cronograma de los equipos, resolver desacuerdos entre los creativos y programadores, coordina la

creación de recursos externos como el arte, música y escenas. Son la cara pública del juego

2.1.3.3.5 Tester

El tester trabaja muchas horas para probar el juego, lo cual requiere que tenga mucha habilidad. Debe tener paciencia, persistencia y buenas habilidades de comunicación para reportar cualquier problema o bugs que encuentre. El aseguramiento de calidad es crucial para el videojuego, ajustándose a estándares de calidad rigurosos.

2.1.4. Mejores prácticas en Aseguramiento de la Calidad / Testeo

Según los estándares definidos por la International Game Developers Association (IGDA) (2003) existen diversas maneras de probar los videojuegos:

2.1.4.1. Pruebas automatizadas

Aquí el factor humano es removido, y se deja que la computadora haga el trabajo, estableciendo una serie de pruebas de depuración integradas a los procesos más complicados. Estas pruebas se realizan para encontrar errores que son más difíciles de encontrar para las personas por el tiempo que implica encontrarlos, lo cual resulta en algunos casos más eficiente.

Aunque esto podría resultar más caro al tener que pedir al desarrollador que escriba más líneas de código para que luego pase a un tester, y siempre está la posibilidad de que el software que haga las pruebas también posea errores.

2.1.4.2. Grupos o individuos especializados para estándares de prueba

Todo los software tienen criterios o estándares que deben ser cumplidos, lo cual es especialmente cierto para juegos de consola. Existen muchos estándares distintos para cada una de las varias plataformas y es difícil conocerlas todas.

2.1.4.3. Subcontratar el departamento de aseguramiento de calidad

Aseguramiento de la calidad es una de las últimas líneas de defensa de un producto. Sin los testers, los productos serían entregados con toda clase de errores y problemas. Mientras más desarrolladores aparezcan, es más difícil para los publicadores y desarrolladores realizar las pruebas. Es un costo que muchas veces los pequeños desarrolladores no pueden costear, así que optan por tercerizar.

2.1.4.4. Escribir el plan de pruebas

Este plan puede ser escrito tanto al tener el primer nivel jugable como al principio del proceso de desarrollo. De la misma forma, quien lo escriba puede ser por el desarrollador o por el editor del juego. Todo depende de los factores que existan en el entorno del proyecto.

2.1.5. Metodologías de desarrollo

Dentro de las metodologías de desarrollo de software más utilizadas encontramos a RUP, y las metodologías ágiles Extreme programming y Scrum. A continuación una breve explicación de cada una de ellas y por qué se ha seleccionado para el desarrollo de esta tesis la metodología Extreme programming (XP).

2.1.5.1. Metodología Principal: eXtreme Programming (XP)

Como bien señala Beck (2000), es una metodología ágil, eficiente, de bajo riesgo, flexible, predecible y una manera más divertida de desarrollar software. La programación extrema es una disciplina de desarrollo de software debido a que hay cosas que se deben realizar al utilizarla, ya que de lo contrario no sería extrema. Es diseñada para trabajar con proyectos realizados por un equipo de software compuesto de 2 hasta 10 programadores. Lo que la hace distinta a otras metodologías y lo que ha sido determinante para su utilización en este proyecto es lo siguiente:

- Su concreta, temprano y continua retroalimentación de ciclos cortos, ya que permite verificar con el cliente si lo que se va realizando se ajusta a sus requerimientos.
- Enfoque de planificación incremental, el cual rápidamente surge con un plan general el cual se espera evolucione a través del ciclo de vida del proyecto.
- Habilidad de programar de manera flexible la implementación de las funcionalidades, respondiendo a los cambios que el negocio necesite. En este caso, los docentes podrán proponer nuevos temas a incluir en el videojuego a lo largo del año escolar, permitiendo XP un adecuado control para estas actualizaciones.
- Dependencia de pruebas automatizadas escritas por programadores y clientes para monitorear el progreso del desarrollo, lo cual permitirá identificar defectos en una etapa temprana.
- Comunicación oral, pruebas y código fuente para comunicar la estructura del sistema con su propósito.

- Procesos de diseño evolutivos que duran tanto como el sistema

La innovación de XP radica en colocar todas estas prácticas bajo un mismo techo, asegurándose de que sean ejecutadas tan minuciosamente como sea posible y de que cada una de ellas apoye a las otras en la mayor medida de lo posible.

2.1.5.2. Rational Unified Process (RUP)

De acuerdo con Kroll y Kruchten (2003), RUP es un enfoque de desarrollo de software iterativo, centrado en la arquitectura y guiado por los casos de uso. Sus procesos de ingeniería de software son bien definidos y bien estructurados, estableciendo claramente responsabilidades y que cosas hacer en el momento que corresponda, proveyendo en resumen el ciclo de vida del software.

Los principios fundamentales de RUP, y que representan la esencia el mismo son: atacar los riesgos importantes desde la fase de inicio, asegurarse de que se entrega un producto apegado a los requerimientos y por lo tanto que implique ser de valor para el cliente, mantenerse concentrado en el correcto funcionamiento del software, dar cabida a los cambios al principio del proyecto, trabajo en equipo, considerar la calidad como un estilo de vida y testear la arquitectura a usar.

RUP funciona como una secuencia de iteraciones, en las cuales se incluyen algunas de las disciplinas de desarrollo. Cada iteración tiene objetivos definidos y producen una parte parcial del trabajo del sistema final. Las primeras iteraciones ponen énfasis en los requerimientos, el análisis y el diseño, mientras que las posteriores lo hacen sobre la implementación y el testeo.

RUP divide el proyecto en 4 fases: Inicio, Elaboración, Construcción y Transición. Cada fase pone especial énfasis en determinadas disciplinas, y se desarrollan mediante un número de iteraciones determinado por el cumplimiento de los objetivos de la fase, el equipo de trabajo y la complejidad del proyecto.

- Fase de Inicio: Se establece cuál es el sistema a construir al obtener comprensión de alto nivel de los requerimientos y estableciendo el alcance del sistema. Se modela el negocio.
- Fase de Elaboración: Se encarga del diseño, establecer una línea base de la arquitectura, y de las primeras etapas de la implementación y pruebas.
- Fase de Construcción: Aquí se realiza la mayor parte de la implementación del sistema. Se despliegan diversas versiones del sistema para asegurarse de que se cumplan con las necesidades del usuario. Se concluye al lanzar una versión completamente funcional del sistema, incluyendo la instalación y la documentación de soporte.

- Fase de Transición: Asegura que el software cumple con las necesidades de sus usuarios, para lo cual se hacen pruebas de aceptación y se concluye con algún documento de aceptación. La retroalimentación del cliente sirve en este punto para poner a punto el sistema, en resolver cuestiones con la configuración, instalación y temas de usabilidad.

Los elementos del proceso RUP son los siguientes:

- Roles: indican quién realizará qué actividades. En RUP los roles definen cómo los individuos deben realizar el trabajo, y ellos especificar las competencias y responsabilidades que cumplen. Una persona ejecuta uno o más roles, y muchas personas pueden ejecutar un mismo rol.
- Actividades: unidad de trabajo que un individuo en un determinado rol debe ejecutar. La actividad tiene un propósito claro usualmente expresado en términos de crear o actualizar uno o un número pequeño de artefactos.
- Artefactos: pieza de información que es producida, modificada i usada por un proceso. Son elementos tangibles del proyecto, objetos que son producidos o usados mientras se desarrolla el software.
- Flujos de trabajo: son una forma de describir secuencias significativas de actividades que producen algún resultado con valor y muestra interacciones entre roles. Los flujos de trabajo más conocidos son las Disciplinas.

Disciplinas: son los contenedores lógicos que agrupan todos los elementos de los procesos (como son los roles, actividades o artefactos), y son nueve: seis de procesos y 3 de soporte. Sin embargo la lista no es definitiva debido a que cualquier compañía puede hacer las extensiones que crea conveniente. Las disciplinas se enfocan en los siguientes puntos:

- Disciplinas de procesos: Modelado de negocio, Administración de requerimientos, Análisis y Diseño, Implementación, Pruebas, y Despliegue.
- Disciplinas de apoyo: Gestión del cambio, Gestión del proyecto, y Entorno.

Ventajas: Se genera documentación más completa, el desarrollo es más organizado y estructurado, se verifica la calidad del software, existe un control de cambios, manejo de errores, centrado en la arquitectura y guiado por los casos de uso. Baja probabilidad de falla.

Desventajas: Implica mayor gasto de recursos como tiempo y dinero. Una vez superada una fase, es muy difícil implementar cambios.

2.1.5.3. Scrum

Citando a Dimes (2015), Scrum es “un marco de referencia para crear software complejo y entregarlo a tiempo de una forma mucho más sencilla”. Este marco de referencia ágil se concentra en el desarrollo de la función que es de mayor valor para los clientes, y se basa en los principios de adaptación, inspección continua, autogestión e innovación. Hace énfasis en comunicaciones informales continuas y está abierto al cambio constante, haciéndola adaptativa. Sus iteraciones suelen ser muchas y frecuentes, pudiendo ir desde una semana hasta un mes, además de realizar reuniones diarias, los Daily Meetings, y semanales, Reunión Sprint, por la misma naturaleza del proyecto que trata esta metodología, que es la de cambios constantes en los requerimientos y la necesidad de validarlos y comunicarlo a todo el equipo de trabajo. Para cumplir con los objetivos del proyecto se hace uso de un Product backlog (pila de productos) donde se listan las características del producto y se priorizan para que el equipo sepa qué trabajar primero. Los ciclos iterativos de Scrum.

Ventajas: Es de desarrollo rápido, constante comunicación y revisión de avances, ideal para proyectos con requisitos cambiantes, es más barata y con un retorno sobre la inversión.

Desventajas: Necesita que los miembros del equipo sean experimentados. Si un integrante se separa del equipo ocasionará retrasos y pérdidas considerables debido al ritmo que lleva el desarrollo del proyecto mientras se le supe. Control de calidad difícil de implementar y cuantificar.

A continuación, se presenta una tabla comparativa entre las mencionadas metodologías de desarrollo, de entre las cuales se seleccionará una para el desarrollo del presente proyecto de investigación. Toda esta información es tomada de los distintos libros de Kroll y Kruchten (2003), Dimes (2015) y Beck (2000).

Tabla N°2. Cuadro comparativo de metodologías de desarrollo de software

SCRUM	XP	RUP
Desarrollo ágil	Desarrollo ágil	Desarrollo óptimo pero toma más tiempo.
Usado en desarrollo de videojuegos con un equipo pequeño de programadores y buen capital inicial	Usado en desarrollo de videojuegos con pocos programadores (1 - 5).	Usado en el desarrollo de videojuegos grandes, dado que se necesitan artistas, compositores, testers, etc. Participación de no menos de 100 personas.
Permite cambios hasta antes del desarrollo del Sprint.	Permite cambios de último momento	Todo el proceso ya está estandarizado y es estricto

Testeo al final de cada Sprint	Testeo a lo largo del proyecto	Testeo al final del módulo
Iteraciones cada 1 - 4 semanas	Entregables cada 1 - 3 semanas	Entregables cada 3 - 5 semanas

Fuente: Datos tomados de las investigaciones de Kroll y Kruchten (2003), Dimes (2015) y Beck (2000)

De acuerdo al cuadro comparativo, dado que la cantidad de programadores es 2, el tiempo para la realización de la investigación es corto, y debido a su flexibilidad de incluir cambios en cualquier fase del proyecto, es que se opta por la metodología ágil desarrollo XP.

c) Hipótesis

El desarrollo de un videojuego educativo basado en redes neuronales artificiales facilitará la evaluación realizada por los docentes en el curso de matemática del 4to. grado de nivel primaria de la institución educativa Niños en Acción en la ciudad de Trujillo.

CAPÍTULO 3. METODOLOGÍA

3.1. Operacionalización de variables

Tabla N°3. Operacionalización de la variable independiente

VARIABLE	DEFINICIÓN CONCEPTUAL	DIMENSIONES	INDICADORES
Videojuego educativo basado en redes neuronales	De acuerdo a lo citado por Acerenza et al. (2009), según Wolf (2003) un videojuego es una herramienta potencial para la educación como también un objeto de estudio para la psicología del comportamiento; un medio para la interacción social, un juguete y un medio de distracción en forma de software. Estos videojuegos tienen el fin de educar, entrenar las habilidades de rapidez mental y mejorar la forma de razonar de las personas.	Usabilidad	Si el videojuego cumple con criterios de buen diseño de interfaces, comprensión de las funciones del sistema y facilidad de uso para los escolares. <ul style="list-style-type: none"> Atracción ¿Es atractivo el diseño del software? <ul style="list-style-type: none"> Entendimiento ¿Es fácil de entender y reconocer la estructura y la lógica y su aplicabilidad?
		Funcionalidad	Si todas las partes del software se han implementado por completo y funcionan correctamente: <ul style="list-style-type: none"> Adecuidad. ¿Tiene el conjunto de funciones apropiadas para las tareas especificadas?

		Confiability	La red neuronal deberá tener una baja tasa de error y brindar confianza para un estudio correcto. <ul style="list-style-type: none">• Madurez. ¿Con qué frecuencia presenta fallas?
--	--	--------------	---

Fuente: Elaboración propia

Tabla N°4. Operacionalización de la variable dependiente

VARIABLE	DEFINICIÓN CONCEPTUAL	DEFINICIÓN OPERACIONAL	DIMENSIONES	INDICADORES	ÍTEMS	FÓRMULA
Evaluación del curso de matemática	Valenzuela, citado en Aquino et al. (2013), define la evaluación en el contexto educativo como “un proceso, y a la vez como un producto” ya que “su aplicación permite estimar el grado en el que un proceso educativo favorece el logro de metas para el cual fue creado”.	El proceso de la evaluación educativa posee dimensiones de clasificación, de las cuales en este trabajo se considera una: objeto del proceso, es decir bajo qué parámetros se realiza la evaluación.	Objeto del proceso	Tiempo de corrección	¿El tiempo empleado por el docente en la corrección de evaluación de los escolares es adecuado?	$TP = \left(\sum_{j=1}^d \frac{\sum_{i=1}^n t_i}{n} \right) / d$ <p>Donde: t: tiempo de corrección de la prueba (en minutos) n: número de pruebas corregidas d: número de docentes</p>
				Estructuración de la información	¿Es adecuada la forma en la cual los docentes gestionan la información de la evaluación de los escolares?	$EI = \left(\sum_{j=1}^d \frac{\sum_{i=1}^n c_i * p_i}{s} \right) / d$ <p>Donde: c: calificación de la pregunta del cuestionario (valor entre 1 a 5) p: peso (prioridad) de la pregunta s: suma de los pesos (prioridad) de las preguntas n: número de preguntas del cuestionario d: número de docentes</p>
				Demanda de preparación de examen	¿El tiempo empleado por el docente en la elaboración de la evaluación es adecuado?	$DPE = \left(\sum_{j=1}^d \frac{\sum_{i=1}^n t_i}{n} \right) / d$ <p>Donde: t: tiempo empleado para elaborar la prueba (en minutos) n: número de pruebas elaboradas. d: número de docentes</p>

Fuente: Elaboración propia

3.2. Diseño de investigación

Experimental Pre-Experimental.

Código de formalización: Diseño de Pre prueba – pos prueba con una sola medición

G: O -> X -> O

Dónde:

G: Grupo de personas o individuos

X: Tratamiento, estímulo o condición experimental

O: Medición a los sujetos de un grupo

3.3. Unidad de estudio

Docente de matemática de 4to, grado de primaria.

3.4. Población

Para esta investigación, se tomará en cuenta los docentes que dictan el curso de matemática del 4° grado de nivel primario en el colegio “Niños en acción”, ubicado en la ciudad de Trujillo, en el distrito de La Esperanza

3.5. Muestra (muestreo o selección)

Se considera la misma cantidad de docentes que en la población

3.6. Técnicas, instrumentos y procedimientos de recolección de datos

Tabla N°5. Recolección de datos

VARIABLES	TÉCNICAS	INSTRUMENTOS	FUENTES	INFORMANTES
Videojuego educativo basado en redes neuronales	Testeo del software	Pruebas de caja negra	Software educativo	Programadores
Evaluación del curso de matemática	Encuesta	Cuestionario	Docentes	Docentes

Fuente: Elaboración propia

Recolección de datos:

Para la variable del Videojuego educativo, se tendrán en consideración las dimensiones que se encuentran en el marco de la norma ISO 9126 para la calidad del software. Los

datos para medir cada una de estas características serán recogidas durante las pruebas de Caja Negra.

Para la variable de Evaluación del curso de matemática, el docente realizará una encuesta respecto a las labores que realiza, para luego hacer uso de la aplicación y contestar nuevamente la encuesta, verificando si es que existe un cambio en lo que el docente percibe como apoyo.

3.7. Métodos, instrumentos y procedimientos de análisis de datos

Tabla N°6. Análisis de datos

VARIABLES	TÉCNICAS	INSTRUMENTOS	FUENTES	INFORMANTES
Videojuego educativo basado en redes neuronales	Análisis de la calidad	Métricas ISO 9126: Se evalúan algunas características del software según la norma 9126 en base a métricas.	Software educativo	Programadores
Evaluación del curso de matemática	Estabilidad (Test - retest)	Coefficiente de correlación.	Docente	Docente

Fuente: Elaboración propia

Procedimiento de análisis de datos:

Para verificar si el videojuego resulta de apoyo para el docente, se comparan los resultados de los cuestionarios que realizó antes y después de usar la aplicación basada en redes neuronales artificiales (mediante la T de student, comparando los datos obtenidos en la recolección de datos).

Para el Análisis de la calidad del Videojuego educativo, se tendrán en cuenta algunas métricas con sus respectivas fórmulas (Anexo N°1).

CAPÍTULO 4. RESULTADOS

4.1. CONTRASTACIÓN

PRUEBA DE HIPÓTESIS

Para la contrastación de hipótesis se ha realizado el método Pre Test - Post Test, lo cual permitirá evaluar la eficacia de poner en práctica la hipótesis, y determinar su aceptación o rechazo. Así mismo, para la realización de este diseño se identificaron indicadores cuantitativos y cualitativos, los cuales se describen a continuación:

Tabla N°7. Prueba de hipótesis

	VARIABLE	INDICADOR	TIPO DE INDICADOR
01	Dependiente	Tiempo de corrección	Cuantitativo
02		Estructuración de la información	Cualitativo
03		Demanda de preparación de examen	Cuantitativo
04	Independiente	Si el videojuego cumple con criterios de buen diseño de interfaces, comprensión de las funciones del sistema y facilidad de uso para los escolares.	Cualitativo
05		Si todas las partes del software se han implementado por completo y funcionan correctamente:	Cualitativo
06		La red neuronal posee una baja tasa de error y brinda confianza para un estudio correcto.	Cuantitativo

Fuente: Elaboración propia

PRUEBA DE HIPÓTESIS INDICADOR DEPENDIENTE

Prueba de hipótesis para indicador 01 (cuantitativo)

Tiempo promedio de corrección de pruebas:

a) Definición de variables

TPCa: Tiempo promedio de corrección de pruebas con la metodología actual

TPCs: Tiempo promedio de corrección de pruebas con la herramienta propuesta

b) Hipótesis estadística

Hipótesis H_0 : Tiempo promedio de corrección de pruebas con la metodología actual es menor o igual que el tiempo promedio de corrección de pruebas con la herramienta propuesta (minutos)

$$H_0 = TPCa - TPCs \leq 0$$

Hipótesis H_a : Tiempo promedio de corrección de pruebas con la metodología actual es mayor que el tiempo promedio de corrección de pruebas con la herramienta propuesta (minutos)

$$H_a = TPSa - TPSs > 0$$

c) Nivel de significancia (α)

Para tomar la decisión de rechazar la hipótesis nula (H_0) cuando ésta sea verdadera (falso positivo). El nivel de confianza es del 95%, por tanto el nivel de significancia será de $\alpha = 0.05$

d) Estadígrafo de contraste

Debido a que la muestra es menor a 30, se usará la distribución t de Student.

$$t = \frac{\bar{d} - \bar{D}}{S_d / \sqrt{n}}$$

e) Resultados

Debido a que se está realizando una comparación de una característica de la muestra es que se usará la t de Student para dos muestras apareadas.

Tabla N°8. Resultados Indicador Tiempo de corrección

	TPSai	TPCsi	Diferencia (d)	$d_i - \bar{d}$	$(d_i - \bar{d})^2$
1	70	0	70	18	324
2	40	0	40	-12	144
3	40	0	40	-12	144
4	55	0	55	3	9
5	55	0	55	3	9
SUMATORIA			260		630
MEDIA MUESTRAL (\bar{d})			52	Varianza (s)	12,5499004

Fuente: Elaboración propia

Primero, se calcula el promedio de las diferencias obtenidas:

$$\bar{d} = \frac{\sum_{i=1}^n d_i}{n}$$

$$\bar{d} = \frac{260}{5}$$

$$\bar{d} = 52$$

Luego, se calcula la desviación estándar muestral de la diferencia:

$$s_d = \sqrt{\frac{1}{n-1} * \sum_{i=1}^n (d_i - \bar{d})^2}$$

$$s_d = \sqrt{\frac{1}{4} * \sum_{i=1}^n (d_i - 52)^2}$$

$$s_d = \sqrt{\frac{630}{4}}$$

$$s_d = 12,5499004$$

Para calcular t:

$$t = \frac{\bar{d} - \bar{D}}{s_d / \sqrt{n}}$$

$$t = \frac{52 - 0}{12,5499004 / \sqrt{5}}$$

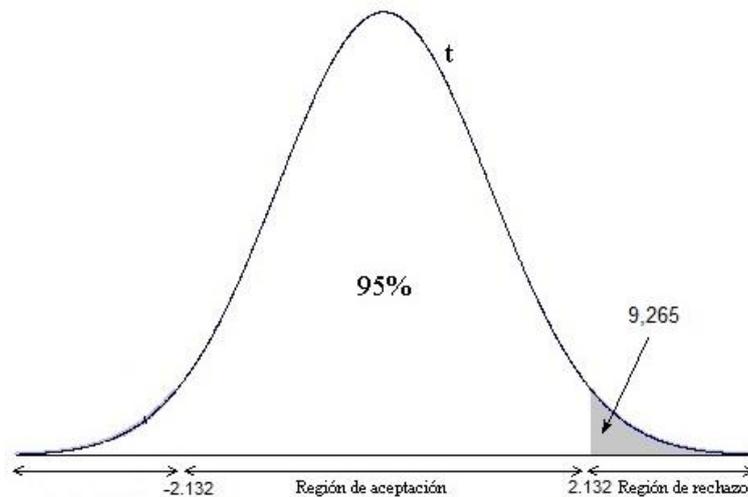
$$t = 9,265056386$$

$$t \cong 9,265$$

f) Región crítica

Para $\alpha = 0.05$ y $n-1$ grados de libertad = 4 encontramos $t = 2,132$. Lo cual significa que $P(T > 2,132) = 0.05$

Figura N°4. Gráfico Resultados Indicador Tiempo de corrección



Fuente: Elaboración propia

g) Conclusión

Debido a que $t = 9,265056386$ calculado, es mayor que 2,132 y encontrándose dentro de la región de rechazo, se puede concluir que H_0 es rechazada, por consiguiente H_a es aceptada. Entonces, el tiempo promedio de corrección de pruebas es menor con la herramienta propuesta que con la metodología actual con un nivel de error del 5% y un nivel de confianza del 95%.

Prueba de hipótesis para indicador 02 (cualitativo)

Valoración promedio de la Estructuración de la información:

a) Definición de variables

E_a: Valoración promedio de la Estructuración de la información con la metodología actual

E_s: Valoración promedio de la Estructuración de la información con la herramienta propuesta

b) Hipótesis estadística

Hipótesis H_0 : Valoración promedio de la Estructuración de la información con la metodología actual es mayor o igual que la Valoración promedio de la Estructuración de la información con la herramienta propuesta

$$H_a = EIa - EIs \geq 0$$

Hipótesis H_a : Valoración promedio de la Estructuración de la información con la metodología actual es menor que la Valoración promedio de la Estructuración de la información con la herramienta propuesta

$$H_0 = EIa - EIs < 0$$

c) Nivel de significancia (α)

Para tomar la decisión de rechazar la hipótesis nula (H_0) cuando ésta sea verdadera (falso positivo). El nivel de confianza es del 95%, por tanto el nivel de significancia será de $\alpha = 0.05$

d) Estadígrafo de contraste

Debido a que la muestra es menor a 30, se usará la distribución t de Student.

$$t = \frac{\bar{d} - \bar{D}}{S_d / \sqrt{n}}$$

e) Resultados

Debido a que se está realizando una comparación de una característica de la muestra es que se usará la t de Student para dos muestras apareadas.

Tabla N°9. Resultados Indicador Estructuración de la información

	EIai	EIsi	Diferencia (d)	$d_i - \bar{d}$	$(d_i - \bar{d})^2$
1	1,7	4	-2,3	0,44	0,1936
2	1,6	4	-2,4	0,34	0,1156
3	1,6	4,4	-2,8	-0,06	0,0036
4	1,3	4,7	-3,4	-0,66	0,4356
5	1,9	4,7	-2,8	-0,06	0,0036
SUMATORIA			-13,7		0,752
MEDIA MUESTRAL (\bar{d})			-2,74	Varianza (s)	0,43358967

Fuente: Elaboración propia

Primero, se calcula el promedio de las diferencias obtenidas:

$$\begin{aligned}\bar{d} &= \frac{\sum_{i=1}^n d_i}{n} \\ \bar{d} &= \frac{-13,7}{5} \\ \bar{d} &= -2,74\end{aligned}$$

Luego, se calcula la desviación estándar muestral de la diferencia:

$$\begin{aligned}s_d &= \sqrt{\frac{1}{n-1} * \sum_{i=1}^n (d_i - \bar{d})^2} \\ s_d &= \sqrt{\frac{1}{4} * \sum_{i=1}^n (d_i - -2,74)^2} \\ s_d &= \sqrt{\frac{0,752}{4}} \\ s_d &= 0,43358967\end{aligned}$$

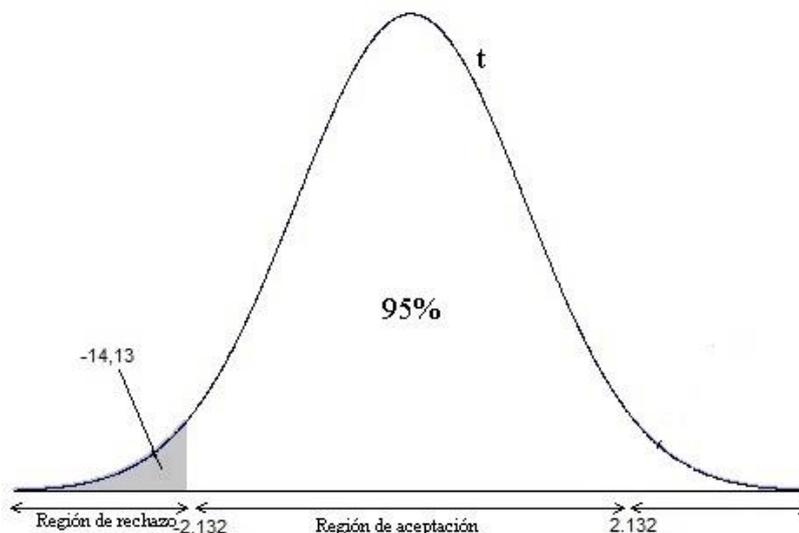
Para calcular t:

$$\begin{aligned}t &= \frac{\bar{d} - \bar{D}}{s_d / \sqrt{n}} \\ t &= \frac{-2,74 - 0}{0,43358967 / \sqrt{5}} \\ t &= -14,13047107 \\ t &\cong -14,13\end{aligned}$$

f) Región crítica

Para $\alpha = 0.05$ y $n-1$ grados de libertad = 4 encontramos $t = 2,132$. Lo cual significa que $P(T < 2,132) = 0.05$

Figura N°5. Gráfico Resultados Indicador Estructuración de la información



Fuente: Elaboración propia

g) Conclusión

Debido a que $t = -14,13047107$ calculado, es menor que $-2,132$ y encontrándose dentro de la región de rechazo, se puede concluir que H_0 es rechazada, por consiguiente H_a es aceptada. Entonces, la Valoración promedio de la Estructuración de la información es mayor con la herramienta propuesta que con la metodología actual con un nivel de error del 5% y un nivel de confianza del 95%.

Prueba de hipótesis para indicador 03 (cuantitativo)

Tiempo promedio para la elaboración de la prueba:

a) Definición de variables

DPEa: Tiempo promedio para elaboración de exámenes con la metodología actual

DPEs: Tiempo promedio para elaboración de exámenes con la herramienta propuesta

b) Hipótesis estadística

Hipótesis H_0 : Tiempo promedio para elaboración de exámenes con la metodología actual es mayor o igual que el tiempo promedio para elaboración de exámenes con la herramienta propuesta (minutos)

$$H_0 = DPEa - DPEs \geq 0$$

Hipótesis H_a : Tiempo promedio de corrección de pruebas con la metodología actual es mayor que el tiempo promedio de corrección de pruebas con la herramienta propuesta (minutos)

$$H_a = DPEa - DPEs < 0$$

c) Nivel de significancia (α)

Para tomar la decisión de rechazar la hipótesis nula (H_0) cuando ésta sea verdadera (falso positivo). El nivel de confianza es del 95%, por tanto el nivel de significancia será de $\alpha = 0.05$

d) Estadígrafo de contraste

Debido a que la muestra es menor a 30, se usará la distribución t de Student.

$$t = \frac{\bar{d} - \bar{D}}{S_d / \sqrt{n}}$$

e) Resultados

Debido a que se está realizando una comparación de una característica de la muestra es que se usará la t de Student para dos muestras apareadas.

Tabla N°10. Resultados Indicador Demanda de preparación de examen

	TPSai	TPCsi	Diferencia (d)	$d_i - \bar{d}$	$(d_i - \bar{d})^2$
1	195	140	55	2	4
2	135	80	55	2	4
3	150	95	55	2	4
4	160	105	55	2	4
5	155	110	45	-8	64
SUMATORIA			265		80
MEDIA MUESTRAL (\bar{d})			53	Varianza (s)	4,472135955

Fuente: Elaboración propia

Primero, se calcula el promedio de las diferencias obtenidas:

$$\bar{d} = \frac{\sum_{i=1}^n d_i}{n}$$

$$\bar{d} = \frac{265}{5}$$

$$\bar{d} = 53$$

Luego, se calcula la desviación estándar muestral de la diferencia:

$$s_d = \sqrt{\frac{1}{n-1} * \sum_{i=1}^n (d_i - \bar{d})^2}$$

$$s_d = \sqrt{\frac{1}{4} * \sum_{i=1}^n (d_i - 53)^2}$$

$$s_d = \sqrt{\frac{80}{4}}$$

$$s_d = 4,472135955$$

Para calcular t:

$$t = \frac{\bar{d} - \bar{D}}{s_d / \sqrt{n}}$$

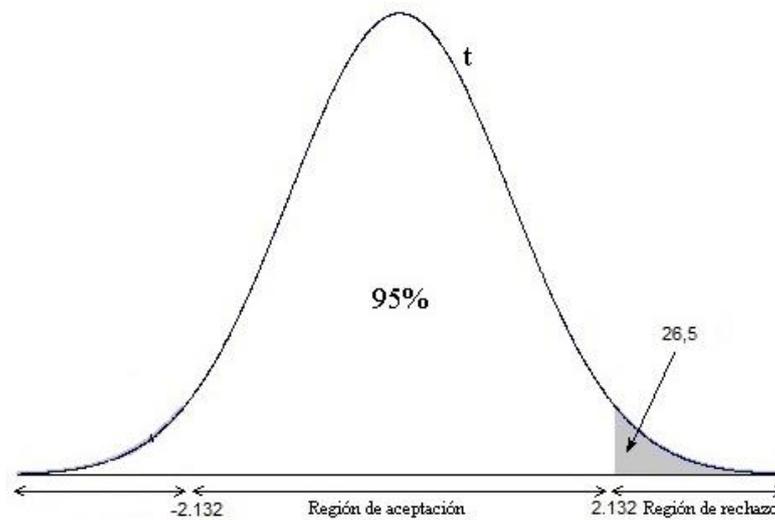
$$t = \frac{53 - 0}{4,472135955 / \sqrt{5}}$$

$$t = 26,5$$

f) Región crítica

Para $\alpha = 0.05$ y $n-1$ grados de libertad = 4 encontramos $t = 2,132$. Lo cual significa que $P(T > 2,132) = 0.05$

Figura N°6. Gráfico Resultados Indicador Demanda de preparación de examen



Fuente: Elaboración propia

g) Conclusión

Debido a que $t = 26,5$ calculado, es mayor que $2,132$ y encontrándose dentro de la región de rechazo, se puede concluir que H_0 es rechazada, por consiguiente H_a es aceptada. Entonces, el tiempo promedio para elaboración de exámenes es menor con la herramienta propuesta que con la metodología actual con un nivel de error del 5% y un nivel de confianza del 95%.

CAPÍTULO 5. DISCUSIÓN

Para que los colegios de educación primaria alcancen sus objetivos es de suma importancia la labor que cumplen los docentes durante las labores de evaluación y seguimiento con los escolares. Estas labores se logran con más facilidad, en el tiempo previsto, contando con las herramientas tecnológicas adecuadas que el docente debe contar.

Ante ello la presente investigación, luego de haber efectuado el análisis de resultados para el indicador Tiempo de corrección, alcanzó una media obtenida con el sistema actual de 52 minutos, y 0 (cero) minutos con el sistema propuesto, obteniendo un decremento del 100%. Esto se traduce y justifica que el sistema actual de los docentes es manual por el uso de recursos de oficina; por el contrario el sistema propuesto permite la corrección automática de exámenes, registro de resultados en una base de datos y reportes de avance de los escolares.

Tabla N°11. Discusión Indicador Tiempo de corrección

TPCa (100 %)	TPCs (TPCs %)	Decremento (%)
52 min.	0 min. (0 %)	52 min. (100 %)

Fuente: Elaboración propia

De similar forma, el análisis de resultados para el indicador de Estructuración de la información, que de acuerdo a los resultados de las encuestas (en base a puntaje 5) se obtiene que la diferencia entre la percepción docente previa al uso del sistema era bastante baja con un promedio de 1.62, en comparación de los resultados obtenidos después del uso del sistema con 4.36, lo que significa que la satisfacción en la gestión de información de los docentes incrementó en un 169,14 %. Esto se interpreta a que el sistema actual de los docentes no permite hacer consultas dinámicas ni rápidas y la información tienden a perderse por estar en un cuaderno; por el contrario el sistema propuesto permite manejar de forma rápida y eficaz la información, evitando pérdidas de información y generando reportes de avance de los escolares ya sea por individuo o grupos de escolares.

Tabla N°12. Discusión Indicador Estructuración de la información

Ela (100 %)	Els (Els %)	Incremento (%)
1,62	4,36 (269,14 %)	2,74 (169,14 %)

Fuente: Elaboración propia

Finalmente, en lo que respecta al indicador Demanda de preparación del examen, se obtuvo luego del análisis que a los docentes les toma, en promedio, 159 minutos prepara un examen con el

sistema actual, y 106 minutos, en promedio, con el sistema propuesto, lo que significa un decremento del 33.33%. La interpretación del resultado es que los docentes luego de esquematizar el tema del examen y de realizar las preguntas respectivas de un tema, tienden a seleccionar preguntas para las distintas filas de exámenes con el fin de evitar plagios, luego pasan estos formatos de exámenes a la computadora para finalmente imprimirlos, proceso que toma mucho tiempo; por el contrario, el sistema propuesto permite la generación rápida de exámenes y selección automática de preguntas de acuerdo al desempeño de cada escolar durante sus exámenes.

Tabla N°13. Discusión Indicador Demanda de preparación de examen

DPEa (100 %)	DPEs (DPEs %)	Decremento (%)
159 min.	106 min. (66,67 %)	53 min. (33,33 %)

Fuente: Elaboración propia

Finalmente, en investigaciones como el “Diseño e implementación de un videojuego en 2D orientado a la ejercitación de la memoria y el desarrollo de la inteligencia espacial” presentado por Luis Alberto Quispe, donde afirmó que el desarrollo de videojuegos para fines educativos, implementando algoritmos de Inteligencia artificial, resultan beneficiosos para el usuario final, mejorando su experiencia y promueve más el uso de este tipo de herramientas. En este espacio podemos afirmar que la presente investigación también logró resultados positivos en el mismo ámbito siendo precedente para más criterios de investigación posteriores.

De igual forma, la investigación “Implementation of a Computerized Assessment System by using Backpropagation Neural Networks with R and Shiny”, desarrollada por Fernando Casafranca y Juan Manuel Gutiérrez, donde exponen las ventajas de utilizar las predictibilidad de las redes neuronales artificiales para evaluar a estudiantes, universitarios en ese caso, incrementan el logro de los objetivos trazados, tal y como sucedió con los docentes en este trabajo de investigación.

CONCLUSIONES

Se concluye que, se logró facilitar la evaluación realizada por los docentes del curso de matemática del 4to grado de nivel primaria de la institución educativa Niños en Acción en la ciudad de Trujillo mediante el desarrollo de un videojuego educativo basado en RNA, cumpliendo los siguientes objetivos específicos:

- Se logró determinar el tiempo promedio de evaluación por los docentes del curso de matemáticas 4to grado de primaria, reduciendo en un 100 %
- Se logró determinar la correcta gestión de información de la evaluación de los escolares en el curso de matemática, en incremento de 169,14 %
- Se logró calcular el tiempo promedio de elaboración de las evaluaciones por el docente en el curso de matemática, reduciendo en un 33,33 %

RECOMENDACIONES

Tanto los docentes con quienes se trabajó como los que desarrollaron la presente tesis recomiendan que la siguiente fase del sistema tenga una versión móvil para que los escolares desde sus casas resuelvan los exámenes y así se incremente la competitividad. Actualmente el sistema tiene la opción de ser migrado al entorno de Android.

Se recomendó también que el apartado del docente pueda contar con un editor de fórmulas matemáticas para que los docentes puedan fácilmente generar fórmulas e insertarlas en las preguntas.

Finalmente se recomienda que el videojuego tenga la opción de desarrollar ejercicios paso a paso, para que los docentes puedan ver el desarrollo de los mismos. Esto significa una investigación más profunda en el campo de la enseñanza teniendo en consideración el constructivismo.

REFERENCIAS

Acerenza, N., Coppes, A., Mesa, G. & Viera, A. (2009). Una metodología ágil para desarrollo de videojuegos (Paper) Universidad de la República, Uruguay.

Aguilera, M., García Medina, A., Muñoz, G. & Pérez Martínez, M. (2011). Evaluación de los aprendizajes en el aula. Opiniones y prácticas de docentes de primaria en México. México DF: Instituto Nacional para la evaluación de la educación. Recuperado de <http://www.inee.edu.mx/images/COEPRI/coeprifinal.pdf>

Beck, K. (2000). Extreme Programming Explained: Embrace Change. Estados Unidos: Addison-Wesley Professional. Recuperado de <https://books.google.com.pe/books?hl=es&lr=&id=G8EL4H4vf7UC&oi=fnd&pg=PR13&dq=extreme+programming&ots=j9wHAtfQAs&sig=exVRdlBxRPXsO9SICDeFr1uwLDg#v=onepage&q=extreme%20programming&f=false>

Cabrol, M. & Severín, E. (2010). TICs en educación: una innovación disruptiva. En BID Educación, N°2. Recuperado de <http://www.iadb.org/wmsfiles/products/publications/documents/35130690.pdf>

Cardemil, C., Carrasco, A. & Román, M. (2011). Enfoque y metodología para evaluar la calidad del proceso pedagógico que incorpora TIC en el aula. En Revista Iberoamericana de Evaluación Educativa, 4 (2) pp. 8-35. Recuperado de <https://dialnet.unirioja.es/servlet/articulo?codigo=4504889>

Casafranca, F. & Gutiérrez, J.M. (2015). Implementation of a Computerized Assessment System by using Backpropagation Neural Networks with R and Shiny. (Paper). Universidad del Pacífico, Lima, Perú. Recuperado de <http://repositorio.up.edu.pe/handle/11354/1087>

Condemarín, M. & Medina, A. (2000). Evaluación de los aprendizajes. Un medio para mejorar las competencias lingüísticas y comunicativas. Santiago de Chile: Mineduc (P900)

Del Moral, E., Esnaola, G., Villalustre, L. & Yuste, T. (2012). Evaluación y Diseño de videojuegos: generando objetos de aprendizaje en comunidades de práctica. (Paper). Universidad de Murcia, España. Recuperado de: https://www.um.es/ead/red/33/esther_et_al.pdf

Dimes, T. (2015). *Conceptos básicos de Scrum: Desarrollo de software Agile y manejo de proyectos Agile*. Recuperado de <https://books.google.com.pe/books?id=ETuXBgAAQBAJ&printsec=frontcover&hl=es#v=onepage&q&f=false>

Fausett, L. (1994). *Fundamentals of neural networks. Architectures, algorithms and applications*. New Jersey, Estados Unidos: Prentice Hall. Recuperado de <http://www.csbdn.in/csbdn-old/pdf/Fundamentals%20Of%20Neural%20Networks.pdf>

Galván, I.M. & Isasi, P. (2004) *Redes de Neuronas Artificiales. Un enfoque práctico*. Madrid: Perason Educación.

García Serrano, A. (2013). *Inteligencia Artificial: Fundamentos, práctica y aplicaciones*. México: Alfa Omega.

Gómez, L. & Macedo, J. (2010). Importancia de las TIC en la educación básica regular. En *Revistas de Investigación UNMSM*, 14 (25) pp. 209-224. Recuperado de <http://revistasinvestigacion.unmsm.edu.pe/index.php/educa/article/view/4776>

International Game Developers Association. (2003). *Best Practices in Quality Assurance/Testing*. Recuperado de https://c.yimcdn.com/sites/www.igda.org/resource/collection/FDB22FE1-269A-4EB8-B76A-7CD0BB88A008/IGDA_Best_Practices_QA_0.pdf

ISO Official Web Site. (2011). *Systems and software engineering — Systems and software Quality Requirements and Evaluation (SQuaRE) — System and software quality models*. Recuperado el 15 de octubre de 2015 de <https://www.iso.org/obp/ui/#iso:std:35733:en>

Kaztman, R. (2010). Impacto social de la incorporación de las nuevas tecnologías de información y comunicación en el sistema educativo. Recuperado de <http://repositorio.cepal.org/handle/11362/6171>

Kroll, P. y Kruchten P. (2003) *The Rational Unified Process Made Easy: A practitioner's Guide to the RUP*. Estados Unidos: Pearson Education. Recuperado de <https://books.google.com.pe/books?hl=es&lr=&id=Ea8qVou5ltEC&oi=fnd&pg=PT22&dq=Kroll,+P.+Kruchten+P.+2003+The+Rational+Unified+Process+Made+Easy:+A+practitioner%E2%80%99s+Guide+to+the+RUP&ots=-w6lrBVRGd&sig=iiRMGIhdhSaw-mVKXubfle9ftnHo#v=onepage&q=Kroll%2C%20P.%20Kruchten%20P.%202003%20The%20Rational%20Unified%20Process%20Made%20Easy%3A%20A%20practitioner%E2%80%99s%20Guide%20to%20the%20RUP&f=false>

Lee, Y.C. (1989). *Evolution, Learning and Cognition*. Singapur: World Scientific Publishing. Recuperado de [https://books.google.com.pe/books?id=KHUGCwAAQBAJ&pg=PA332&lpg=PA332&dq=how+neur+al+nets+work+ln+this+paper+we+review+and+extend+our+work+\(Lapedes+and+Farber+,+1987\)&s+ource=bl&ots=m1j6SbUWOz&sig=k2-](https://books.google.com.pe/books?id=KHUGCwAAQBAJ&pg=PA332&lpg=PA332&dq=how+neur+al+nets+work+ln+this+paper+we+review+and+extend+our+work+(Lapedes+and+Farber+,+1987)&s+ource=bl&ots=m1j6SbUWOz&sig=k2-)

ylors7W3sm64hMH92Jz8A_E&hl=es&sa=X&ved=0ahUKEwi6_pOn7r3PAhUMXB4KHViHD7sQ6A
EIKDAC#v=onepage&q&f=false

Marquès, P. (2000), Los docentes: funciones, roles, competencias necesarias, formación. Universitat Autònoma de Barcelona, España. Recuperado de http://www.uaa.mx/direcciones/dgdp/defaa/descargas/docentes_funciones.pdf

Martín del Brío, B. & Sanz, A. (2006), *Redes Neuronales y Sistemas borrosos*. (3.ª ed.). Madrid: RA-MA Editorial

Ministerio de Educación del Perú (2016), Sesiones de aprendizaje 2016. Recuperado de <http://www.minedu.gob.pe/rutas-del-aprendizaje/sesiones2016/unidad1/cuartogrado-mate.php>

Nareshkumar, N. (2005). *Simultaneous versus successive learning in neural networks*. (Tesis de Maestría). Universidad de Miami, Ohio, Estados Unidos. Recuperado de https://etd.ohiolink.edu/pg_10?0::NO:10:P10_ACCESSION_NUM:miami1134068959

Norvig, P. & Russell, S.J. (2004). *Inteligencia Artificial. Un Enfoque Moderno*. (2.ª ed.). Madrid: Pearson Educación.

Organización para la Cooperación y el Desarrollo Económico. (2013). Programa para la evaluación internacional de alumnos (PISA) PISA 2012 – Resultados. Recuperado de <https://www.oecd.org/pisa/keyfindings/PISA-2012-results-spain-ESP.pdf>

Pérez Padilla, F., Sollet, E. & Torijano, B. (2007). <e-Adventure>. (Paper). Universidad Complutense de Madrid, España. Recuperado de: http://eprints.ucm.es/8976/1/Documentacion_e-Adventure.pdf

Quispe, L.A. (2013). *Análisis, diseño e implementación de un videojuego en 2D orientado a la ejercitación de la memoria y el desarrollo de la inteligencia espacial*. (Tesis de Título). Pontificia Universidad Católica del Perú, PUCP, Lima, Perú.

Rogers, S. (2010). *Level Up! : The guide to great video game design*. Reino Unido: John Wiley & Sons.

Vaillant, D. (2013). Integración de TIC en los sistemas de formación docente inicial y continua para la Educación Básica en América Latina. Buenos Aires, Argentina: Fondo de las Naciones Unidas para la Infancia (UNICEF). Recuperado de <https://books.google.com.pe/books?hl=es&lr=&id=dR2ABAAAQBAJ&oi=fnd&pg=PA10&dq=Los+docentes+y+las+TICs+basica&ots=sXiddXZzuB&sig=dZ8tEAOWeXQ-a2MKHs7UhMuKsk#v=onepage&q=Los%20docentes%20y%20las%20TICs%20basicaper%C3%BA&f=false>

ANEXOS

ANEXO N°1

ISO 9126 para la calidad del software - Métricas

Tabla N°A1. Dimensión: Usabilidad - Indicador: Atracción

Nombre:	Buen diseño de las interfaces
Propósito:	Determinar si la interfaz cumple con patrones de diseño para que sea bien vista por los usuarios.
Método de aplicación:	de Contar los elementos y controles que hacen siguen los patrones de diseño (percepción, cognición, color, memoria, metáforas bien elaboradas, etc.)
Medición, fórmula:	$X = A/B$ A = número de elementos que siguen los patrones de diseño B = número total de elementos en la interfaz
Interpretación:	$0 < X \leq 1$ Entre más cercano a 1, más completa.
Tipo de escala:	absoluta
Tipo de medida:	X = entero/entero A = números enteros mayores que 0 B = números enteros mayores que 0
Fuente de medición:	de Especificación de requisitos, Diseño, Informe de revisión y Patrones de diseño establecidos

Dimensión: Usabilidad - Indicador: Entendimiento

Nombre:	Comprensión de las funciones del sistema
Propósito:	Medir el número de funciones que el usuario es capaz de realizar al usar el software.
Método de aplicación:	de Contar las funciones que se utilizaron con éxito durante las pruebas y comparar con el número de funciones descritas en la especificación de requisitos. La resta de ambos nos dará el número de funciones que no se entendieron.

Medición, fórmula:	$X = A/B$ A = número de funciones comprendidas con éxito B = número total de funciones
Interpretación:	$0 < X \leq 1$ Entre más cercano a 1, más completa.
Tipo de escala:	absoluta
Tipo de medida:	$X = \text{entero/entero}$ A = números enteros mayores que 0 B = números enteros mayores que 0
Fuente de medición:	Especificación de requisitos, Diseño, Código fuente, Pruebas del software e Informe de revisión

Tabla N°A2. Dimensión: Funcionalidad - Indicador: Adecuidad

Nombre:	Completitud de implementación funcional
Propósito:	Mide el nivel de completitud de la implementación funcional
Método de aplicación:	Contar las funciones no implementadas durante la evaluación y comparar con el número de funciones descritas en la especificación de requisitos.
Medición, fórmula:	$X = A/B$ A = número de funciones faltantes detectadas en la evaluación B = Número de funciones descritas en la especificación de requisitos.
Interpretación:	$0 < X \leq 1$ Entre más cercano a 1, más completa.
Tipo de escala:	absoluta
Tipo de medida:	$X = \text{entero/entero}$ A = números enteros mayores que 0 B = números enteros mayores que 0
Fuente de medición:	Especificación de requisitos Diseño Código fuente Informe de revisión

Dimensión: Confiabilidad - Indicador: Madurez

Nombre:	Suficiencia de pruebas
---------	------------------------

Propósito:	Determinar el porcentaje de casos de prueba que son cubiertos por la red neuronal.
Método de aplicación:	<ul style="list-style-type: none"> • Contar las pruebas que se realizaron con éxito en la RNA y compararlas con el número total de pruebas que se realizaron. • La medición se hará una vez realizado el entrenamiento de la RNA.
Medición, fórmula:	$X = A/B$ A = Número de pruebas con éxito. B = Número de total de pruebas.
Interpretación:	$0 < X \leq 1$ Entre más cercano a 1, mejor.
Tipo de escala:	Absoluta
Tipo de medida:	X = entero/entero A = números enteros mayores que 0 B = números enteros mayores que 0
Fuente de medición:	Código Fuente Pruebas del software

ANEXO N°2

Encuesta docente para evaluación de la herramienta educativa

Responder en el espacio en blanco con un número entre 1 a 5, donde 5 es muy satisfecho con el aspecto en cuestión.

Tabla N°A3. Encuesta

Indicador: Estructuración de la información	
Aspecto: ¿Es adecuada la forma en la cual los docentes gestionan la información de la evaluación de los escolares?	Prioridad: 4
1. ¿Qué tan conveniente le resulta asignar ejercicios en los exámenes? Considerando que cada vez que realiza una prueba utiliza exámenes distintos para cierta cantidad de escolares y así evitar el plagio. _____	
Aspecto: ¿Es adecuada la forma en la cual los docentes gestionan la información de la evaluación de los escolares?	Prioridad: 3
2. ¿Qué tan eficiente considera el método actual con el que prepara sus exámenes? _____	
3. ¿Qué tan cómodo le resulta desempeñar sus labores de seguimiento a los escolares?_____	

ANEXO N°3

METODOLOGÍA DE DESARROLLO: XP

1. FASE DE EXPLORACIÓN

1.1. Alcance del proyecto

1.1.1. Equipo de trabajo

Tabla N°A4. Equipo de trabajo

Actores	ROLES			
	Programador	Tester	Cliente	Guía
Calderón Pereda, Giancarlo Daniel	X	X		
Figueroa Rodríguez, Diego Enzo	X	X		
Ministerio de Educación				X
Colegios de nivel primaria de Trujillo			X	

Fuente: Elaboración propia

1.1.2. Solución Propuesta

Este videojuego es concebido como una herramienta que apoyará a los docentes durante sus actividades de evaluación. En base a ellos se obtendrán datos para su análisis y determinar si es que existe una mejora en sus tareas, así como también en lo concerniente a la recepción y asimilación de conocimientos por parte de sus escolares. El desarrollo se hará con el uso del framework XNA junto al IDE NetBeans, con el lenguaje de programación Java, y el motor de base de datos SQL Server Management Studio.

1.1.3. Usuarios

Los principales interesados de la herramienta educativa son los docentes de matemática del 4° grado de nivel primaria del colegio "Niños en acción", en la ciudad de Trujillo. No obstante, los escolares también tendrán interacción con la herramienta.

1.2. Historias de Usuario

Tabla N°A5. Historia de Usuario "Creación de personajes y recursos"

Historia de Usuario	
Número: 1	Usuario: Programador
Nombre historia: Creación de personajes y recursos	
Prioridad en negocio: Medio	Riesgo en desarrollo: Bajo
Iteración asignada: 1	
Programador responsable: Diego Figueroa	

Descripción: Como docente, necesito que la herramienta cuente con contenido agradable, ordenado, sencillo y amigable para los escolares, lo cual incluye personajes, ambientes y demás objetos que se presentan en el videojuego.
Observaciones:

Fuente: Elaboración propia

Tabla N°A6. Historia de Usuario “Usabilidad de Interfaz Docente”

Historia de Usuario	
Número: 2	Usuario: Programador
Nombre historia: Usabilidad de Interfaz Docente	
Prioridad en negocio: Alta	Riesgo en desarrollo: Medio
Iteración asignada: 1	
Programador responsable: Diego Figueroa	
Descripción: Como docente, necesito que la herramienta dedicada para hacer los exámenes sea fácil de usar, de tal forma que no recargue más tareas a las obligaciones docentes.	
Observaciones: <ul style="list-style-type: none"> • Desarrollo de una interfaz con criterios de usabilidad. • Los docentes no desean perder tiempo preocupándose por la historia del juego, generación del mapa del juego o por la posición de los enemigos y obstáculos. • Los docentes solo desean escoger la historia o tipo de juego. • Interfaz sencilla de aprender para que la corrección de los exámenes sea rápida y correcta. 	

Fuente: Elaboración propia

Tabla N°A7. Historia de Usuario “Elaboración de exámenes”

Historia de Usuario	
Número: 3	Usuario: Programador
Nombre historia: Elaboración de exámenes	
Prioridad en negocio: Alta	Riesgo en desarrollo: Medio
Iteración asignada: 2	
Programador responsable: Diego Figueroa	
Descripción: Como docente, necesito elaborar exámenes más rápido y de forma sencilla, de tal forma que las preguntas sean aleatorias para cada escolar, para tratar de evitar la incidencia de plagio Teniendo en cuenta que se debe ingresar: <ul style="list-style-type: none"> • Preguntas de examen, alternativas por pregunta y dificultad de la pregunta. 	
Observaciones: <ul style="list-style-type: none"> • Los docentes pierden mucho tiempo haciendo 2 o 3 listas de exámenes distintos sin perder la equidad del nivel de dificultad para todos sus escolares. El sistema deberá disminuir el tiempo invertido en este proceso. 	

- Tener un banco de preguntas que pueda ser utilizado luego por la red neuronal

Fuente: Elaboración propia

Tabla N°A8. Historia de Usuario “Estructuración de avance de escolares”

Historia de Usuario	
Número: 4	Usuario: Programador
Nombre historia: Estructuración de avance de escolares	
Prioridad en negocio: Alta	Riesgo en desarrollo: Bajo
Iteración asignada: 3	
Programador responsable: Giancarlo Calderón	
Descripción: Como docente, necesito visualizar de mejor forma el avance de notas de los escolares. Que cumpla con los siguientes requisitos: <ul style="list-style-type: none"> • Listar escolares por código, nombre u apellido. • Listar el avance de sus escolares, tanto individual como en conjunto (clase). • Mostrar un gráfico de los avances de los escolares. 	
Observaciones: <ul style="list-style-type: none"> • Los docentes en su mayoría tienen un cuaderno donde almacenan las notas y avances de los escolares. Les resulta difícil poder hacer un análisis rápido del avance de sus escolares dado que tienen múltiples exámenes a lo largo del Bimestre. 	

Fuente: Elaboración propia

Tabla N°A9. Historia de Usuario “Desarrollo de Interfaz del Videojuego educativo”

Historia de Usuario	
Número: 5	Usuario: Programador
Nombre historia: Desarrollo de Interfaz del Videojuego educativo	
Prioridad en negocio: Alta	Riesgo en desarrollo: Alto
Iteración asignada: 4	
Programador responsable: Giancarlo Calderón	
Descripción: Como docente, necesito que a lo largo del juego se presenten las preguntas o retos que el alumno debe resolver. El funcionamiento debe ser sencillo, de acuerdo a lo siguiente: <ul style="list-style-type: none"> • Cuando se topen con un enemigo, se mostrará la pregunta a resolver con las alternativas a marcar. • Después de resolver correctamente una pregunta, el enemigo deberá ser eliminado. 	
Observaciones: <ul style="list-style-type: none"> • El desarrollo de una interfaz limpia, fácil de usar y atractiva a la vista de los escolares supone una mayor aceptación de la herramienta educativa por parte de los docentes. • La cantidad de preguntas, enemigos y alternativas por pregunta ya debieron haber sido ingresados en la interfaz de la Historia de Usuario N°1 - Elaboración de exámenes 	

Fuente: Elaboración propia

Tabla N°A10. Historia de Usuario “Desarrollo de RNA”

Historia de Usuario	
Número: 6	Usuario: Programador
Nombre historia: Desarrollo de RNA	
Prioridad en negocio: Alta	Riesgo en desarrollo: Medio
Iteración asignada: 5	
Programador responsable: Equipo de programación	
<p>Descripción: Como docente, quisiera que las preguntas del videojuego se ajusten dinámicamente a las capacidades de los escolares para que estos puedan ser evaluados de acuerdo a su rendimiento individual.</p> <p>Además las preguntas deberán generarse dependiendo de: tiempo empleado, dificultad actual de la pregunta, cantidad de preguntas resultas y número de pregunta actual.</p>	
<p>Observaciones:</p> <ul style="list-style-type: none"> • El videojuego educativo hará uso de una Red Neuronal Artificial para el ajuste dinámico de la dificultad de las preguntas. • La RNA deberá ser probada y tener un margen de error aceptable. • Los docentes no pueden hacer exámenes personalizados para cada escolar dependiendo de su nivel, por lo tanto a los docentes les gustaría probar esta opción en el software. • Los docentes quieren hacer un estudio sobre el rendimiento de los escolares si es que la dificultad es diferente en cada uno de ellos y no una sola dificultad estándar para toda la clase. 	

Fuente: Elaboración propia

Tabla N°A11. Historia de Usuario “Integración de los módulos”

Historia de Usuario	
Número: 7	Usuario: Programador
Nombre historia: Integración de los módulos	
Prioridad en negocio: Alta	Riesgo en desarrollo: Alto
Iteración asignada: 6	
Programador responsable: Equipo de programación	
<p>Descripción: Como docente, necesito que haya un correcto funcionamiento entre los módulos que conforman la herramienta:</p> <ul style="list-style-type: none"> • RNA y base de datos terminada y validada. • Módulo del docente. • Módulo de escolares 	
<p>Observaciones:</p> <ul style="list-style-type: none"> • Los módulos deben funcionar de la mejor manera para que los docentes puedan apreciar mejor el prototipo y ayude con el visto bueno por parte de ellos. 	

Fuente: Elaboración propia

2. FASE DE PLANIFICACIÓN

2.1. Planificación Inicial

En la planificación inicial listaremos las historias de usuario de acuerdo a su **PRIORIDAD** (Bajo, Media o Alta según la importancia y relevancia que tenga), **RIESGO** (Bajo, Medio o Alto según la probabilidad de fallo de cada historia de usuario en el desarrollo), **ESFUERZO** (calificando 1, 2 o 3 según el tiempo y trabajo que demandará en desarrollar la historia de usuario) e **ITERACIÓN** (momento de implementación de cada historia), las cuales se acordaron con el docente.

Tabla N°A12. Planificación Inicial de Historias de Usuario

Nro.	Historia de Usuario	Prioridad de Negocio	Riesgo	Esfuerzo	Iteración
	Nombre				
001	Creación de personajes y recursos	Medio	Bajo	1	1
002	Usabilidad de Interfaz Docente	Alta	Medio	2	1
003	Elaboración de exámenes	Alta	Medio	2	2
004	Estructuración de avance de escolares	Alta	Bajo	2	3
005	Desarrollo de Interfaz del Videojuego educativo	Alta	Alto	2	4
006	Desarrollo de RNA	Alta	Medio	2	5
007	Integración de los módulos	Alta	Alto	3	6

Fuente: Elaboración propia

2.2. Estimación de tiempos

De acuerdo a las ponderaciones de la prioridad, riesgo y esfuerzo se ha estimado el tiempo de desarrollo de cada historia.

Tabla N°A13. Estimación de tiempos de Historia de Usuario

Nro.	Historia de Usuario	Prioridad de Negocio
	Nombre	
001	Creación de personajes y recursos	Medio
002	Usabilidad de Interfaz Docente	Alta
003	Elaboración de exámenes	Alta
004	Estructuración de avance de escolares	Alta
005	Desarrollo de Interfaz del Videojuego educativo	Alta
006	Desarrollo de RNA	Alta
007	Integración de los módulos	Alta

Fuente: Elaboración propia

Total estimado de 135 días.

2.3. Plan de entrega

Tabla N°A14. Plan de entrega

Entregable	Historias	Fecha Inicio	Fecha Término	Fecha Entrega
Entregable 1	01	06/01/2017	14/01/2017	17/01/2017
Entregable 2	02	07/01/2017	14/01/2017	17/01/2017

Entregable 3	03	14/01/2017	22/01/2017	24/01/2017
Entregable 4	04	24/01/2017	28/02/2017	28/02/2017
Entregable 5	05	01/03/2017	15/04/2017	18/04/2017
Entregable 6	06	16/04/2017	07/05/2017	09/05/2017
Entregable 7	07	08/05/2017	20/05/2017	23/05/2017

Fuente: Elaboración propia

3. FASE DE ITERACIONES

3.1. ITERACIÓN 1

3.1.1. HISTORIA DE USUARIO N°1

Tabla N°A15. Desarrollo de la Historia de Usuario "Creación de personajes y recursos"

Historia de Usuario N°01: Creación de personajes y recursos			Iteración No. 01
Sub historia	Fase	Tarea por Sub Historia	
Crear Personajes	Diseño	3.1.1.1	Diseño del Personaje
	Implementación	3.1.1.2	Establecer jugabilidad
	Tarjeta CRC	3.1.1.3	Tarjeta CRC Crear Personajes
	Pruebas	3.1.1.4	Testeo del personaje y su modo de jugabilidad
Crear Recursos	Diseño	3.1.1.5	Diseño de Recursos
	Implementación	3.1.1.6	Funcionalidad de TileMap
	Tarjeta CRC	3.1.1.7	Tarjeta CRC Crear Recursos
	Pruebas	3.1.1.8	Testeo de los recursos y su interacción en el TileMap

CREAR PERSONAJES

3.1.1.1. Diseño del Personajes

- Tarea Diseño del Personaje

Se establecen las imágenes que servirán para darle animación al personaje.

Figura N°A1.



3.1.1.2. Establecer Jugabilidad

Dirección: **Controles de dirección** ↑↓←→.

Acción: **Mouse, click izquierdo.**

Historia: Jake es un chico que tiene que salvar a su aldea de los demonios del fuego. Para derrotarlos deberá resolver los múltiples acertijos que ellos plantean.

3.1.1.3. Tarjeta CRC Crear Personajes

Tabla N°A16. CRC Personaje

Personaje	
Establece métodos para personajes	Sprite (gdx) Animation (gdx)

Tabla N°A17. CRC Personaje Principal

Personaje Principal	
Establecer animaciones del personaje principal del videojuego	Sprite (gdx) Animation (gdx) Personaje()

Tabla N°A18. CRC Animation

Animation	
Facilita el manejo de los frames y texturas para las animaciones.	LibGDX

Tabla N°A19. CRC Sprite

Sprite	
Ofrece métodos para manejar las propiedades de las imágenes que sirven para animaciones, tales como: ancho, alto, color, textura, coordenadas.	LibGDX

3.1.1.4. Testeo del personaje y su modo de jugabilidad

Al no existir un modelo establecido para realizar las pruebas de esta clase de herramientas – videojuegos – se optó por adecuar el modelo de partición de equivalencias.

Tabla N°A20.

CONDICIÓN	CLASE VALIDA	CLASE NO VALIDA
El personaje puede moverse en dirección derecha sobre la pantalla.	1: Tecla direccional derecha.	2: Cualquier otro input que no sea la direccional derecha.
El personaje puede moverse en dirección izquierda sobre la pantalla.	3: Tecla direccional izquierda.	4: Cualquier otro input que no sea la direccional izquierda.
El personaje puede moverse en dirección arriba sobre la pantalla	5: Tecla direccional arriba.	6: Cualquier otro input que no sea la direccional izquierda.
El personaje puede moverse en dirección abajo sobre la pantalla	7: Tecla direccional abajo.	8: Cualquier otro input que no sea la direccional abajo.

Partición de equivalencias:

Casos de prueba:

Tabla N°A21.

NRO	CLASES DE EQUIVALENCIA	DATOS DE ENTRADA	RESULTADO ESPERADO
CP-01	1,3,5,7	Presionar teclas: direccional derecha, direccional izquierda, direccional arriba, direccional abajo	El movimiento del personaje sobre la pantalla debe ser el siguiente: derecha, izquierda, arriba, abajo.
CP-02	2,3,5,7	Presionar teclas: letra 'D', direccional izquierda, direccional arriba, direccional abajo.	El movimiento del personaje sobre la pantalla debe ser el siguiente: izquierda, arriba, abajo.
CP-03	1,4,6,8	Presionar teclas: direccional derecha, letra 'A', direccional arriba, direccional abajo.	El movimiento del personaje sobre la pantalla debe ser el siguiente: derecha, arriba, abajo.
CP-04	1,3,6,7	Presionar teclas: direccional derecha, direccional izquierda, letra 'W', direccional abajo	El movimiento del personaje sobre la pantalla debe ser el siguiente: derecha, izquierda, abajo
CP-05	1,3,5,8	Presionar teclas: direccional derecha, direccional izquierda, direccional arriba, letra 'S'	El movimiento del personaje sobre la pantalla debe ser el siguiente: derecha, izquierda, arriba.

Figura N°A2.



Figura N°A3.



Figura N°A4.



Figura N°A5.



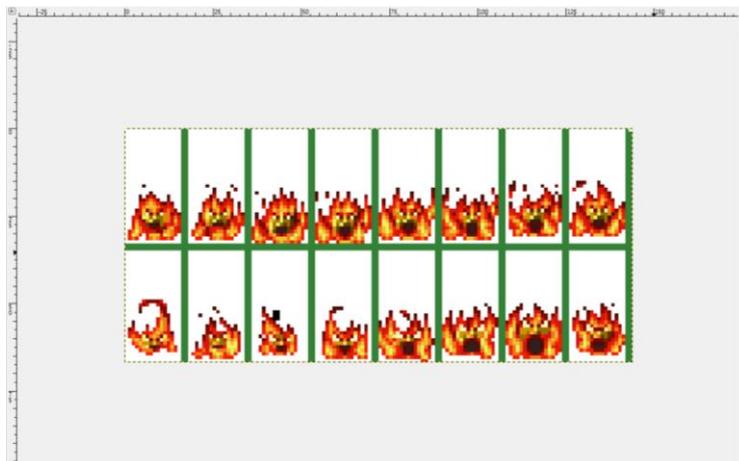
CREAR RECURSOS

3.1.1.5. Diseño de Recursos

- Tarea Diseño de Recursos

Se establecen las imágenes que servirán para darle animación a los demás recursos que se utilizarán en los mapas.

Figura N°A6.



3.1.1.6. Funcionalidad de TileMap

Implementamos el Tilemap para que pueda ser visto y testeado desde la aplicación del videojuego con las siguientes líneas de código que cargan el mapa con extensión .tmx

Figura N°A7.

```
viewport = new FitViewport(TesisMain.V_WIDTH, TesisMain.V_HEIGHT, gamecam);
//una escena del screen donde se visualiza el puntaje
currentScore = new Question(game.batch, this);
mapLoader = new TmxMapLoader();
map = mapLoader.load("levell-2.tmx");
//brindarle el mapa al renderizador de mapas y de paso darle mas Caché
renderer = new OrthoCachedTiledMapRenderer(map, 1, 5000);
//inicializar camara con posiciones
gamecam.position.set(600, 600, 0);
```

3.1.1.7. Tarjeta CRC Crear Recursos

Tabla N°A22.

Tesis Main	
Cargar SpriteBatch	Play Screen
Cargar Pantalla	SpriteBatch

Tabla N°A23.

Play Screen	
Generación de imágenes	Current Score
Cargar personajes de videojuego	Personaje Principal
Renderizar el mapa	Personaje
Establecer controles básicos de juego	

Tabla N°A24.

SpriteBatch	
Recoge y ejecuta toda la información (texturas) para que sea procesado por la GPU y se dibujada correctamente.	LibGDX

3.1.1.8. Testeo de los recursos y su interacción en el TileMap

Al no existir un modelo establecido para realizar las pruebas de esta clase de herramientas – videojuegos – se optó por adecuar el modelo de Partición de equivalencias.

Tabla N°A25.Casos de prueba:

NRO	ACCIÓN	EJECUCIÓN / DATOS DE ENTRADA	ESCENARIO ESPERADO	ESCENARIO CON ERROR
CP-01	Ejecución de la herramienta	Se ejecuta la herramienta.	Aparece la pantalla de la herramienta. Luego al chocar el personaje con algún elemento sólido debe detener su marcha.	No se muestra la pantalla.
				Aparece un mensaje de error.
CP-02	El personaje se mueve en 4 direcciones con total normalidad al accionarse las teclas.	Presionar las teclas direccionales (derecha, izquierda, arriba, abajo) para mover el personaje.	El personaje se mueve alrededor sin ningún problema.	El personaje no se mueve.
				Aparición de algún mensaje de error

Figura N°A8.



3.1.2. HISTORIA DE USUARIO N°2

Tabla N°A26. Desarrollo de la Historia de Usuario “Usabilidad de Interfaz Docente”

Historia de Usuario N°02: Usabilidad de Interfaz Docente		Iteración No. 01	
Sub historia	Fase	Tarea por Sub Historia	
Organizar Interfaz	Diseño	3.1.2.1	Diseño de la Interfaz
	Implementación	3.1.2.2	Establecer elementos de la Interfaz
	Tarjeta CRC	3.1.2.3	Tarjeta CRC Organizar Interfaz
	Pruebas	3.1.2.4	Testeo del funcionamiento de la Interfaz

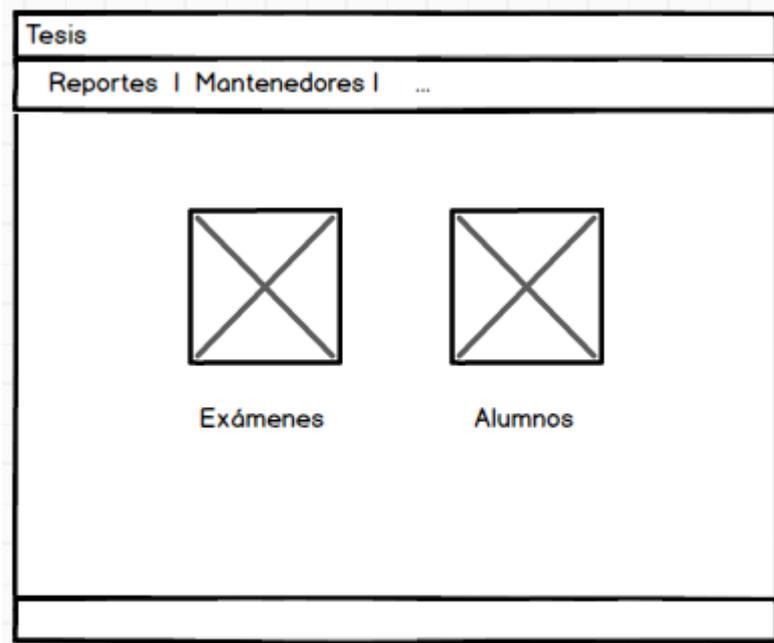
Fuente: Elaboración propia

3.1.2.1. Diseño de la Interfaz

Figura N°A9. Pantalla de Inicio

Fuente: Elaboración propia

Figura N°A10. Pantalla de Administración Docente



Fuente: Elaboración propia

3.1.2.2. Establecer elementos de la Interfaz

Figura N°A11.

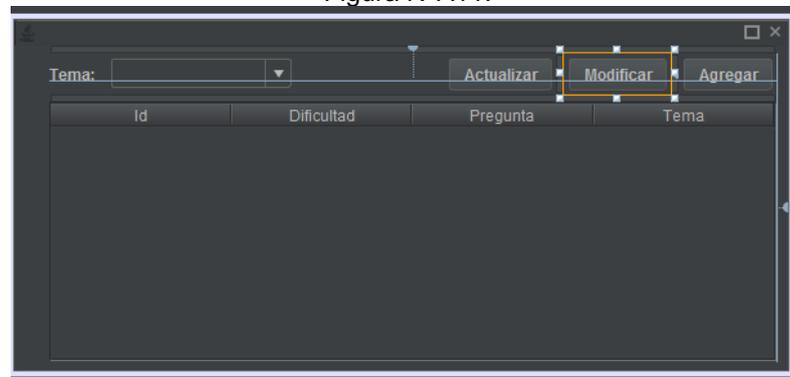
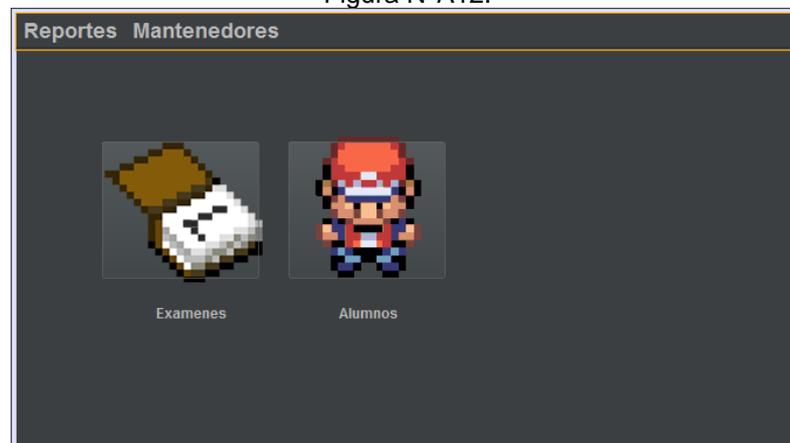


Figura N°A12.



3.1.2.3. Tarea Tarjeta CRC Organizar Interfaz

Tabla N°A27.

Tesis Main	
Cargar SpriteBatch Cargar Pantalla	Play Screen SpriteBatch

Tabla N°A28.

Play Screen	
Generación de imágenes Cargar personajes de videojuego Renderizar el mapa Establecer controles básicos de juego	Current Score Personaje Principal Personaje

Tabla N°A29.

SpriteBatch	
Recoge y ejecuta toda la información (texturas) para que sea procesado por la GPU y se dibujada correctamente.	LibGDX

3.1.2.4. Testeo del funcionamiento de la Interfaz

Casos de prueba para la ejecución de la herramienta para el docente.

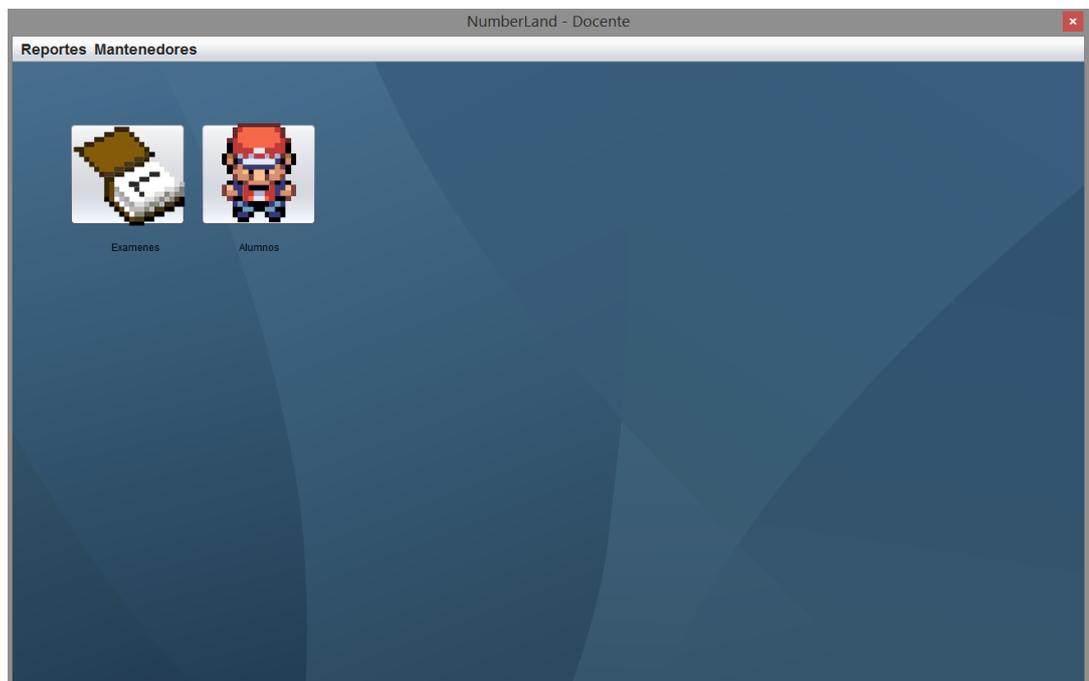
Tabla N°A30. Casos de prueba:

NRO	ACCIÓN	EJECUCIÓN DATOS ENTRADA	/ DE	ESCENARIO ESPERADO	ESCENARIO CON ERROR
CP-01	Ejecución de la herramienta	Se ejecuta la herramienta.		Se muestra la pantalla de autenticación	No se muestra la pantalla.
					Aparece un mensaje de error.
CP-02	Autenticación	Se ingresa datos de autenticación válidos en los campos de usuario y contraseña.		Se muestra la interfaz para la administración de la herramienta del docente.	No se muestra la pantalla.
					Aparece un mensaje de error.
CP-03	Ingreso a Generación de exámenes	Se selecciona la opción Exámenes		Se muestra el módulo para la gestión de Exámenes.	No se muestra la pantalla.
					Aparece un mensaje de error.
CP-04	Ingreso a Gestión de escolares	Se selecciona la opción Escolares		Se muestra la interfaz para la gestión de Escolares.	No se muestra la pantalla.
					Aparece un mensaje de error.

Figura N°12.



Figura N°A13.



3.2. ITERACIÓN 2:

3.2.1. HISTORIA DE USUARIO N°3

Tabla N°A31. Desarrollo de Historia de Usuario N°3 “Elaboración de exámenes”

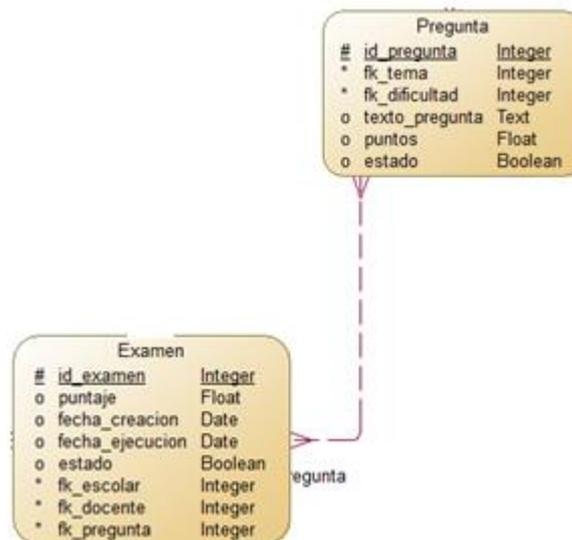
Historia de Usuario N°03: Elaboración de exámenes			Iteración No. 01
Sub historia	Fase	Tarea por Sub Historia	
Generar Examen	Diseño	3.2.1.1	Diseño de Generar Examen
	Implementación	3.2.1.2	Crear Generar Examen
	Tarjeta CRC	3.2.1.3	Tarjeta CRC Generar Examen
	Pruebas	3.2.1.4	Testeo de Generar Examen
Crear Pregunta	Diseño	3.2.1.5	Diseño de Preguntas
	Implementación	3.2.1.6	Crear pregunta
	Tarjeta CRC	3.2.1.7	Tarjeta CRC Crear Pregunta
	Pruebas	3.2.1.8	Testeo de Preguntas

Fuente: Elaboración propia

GENERAR EXAMEN

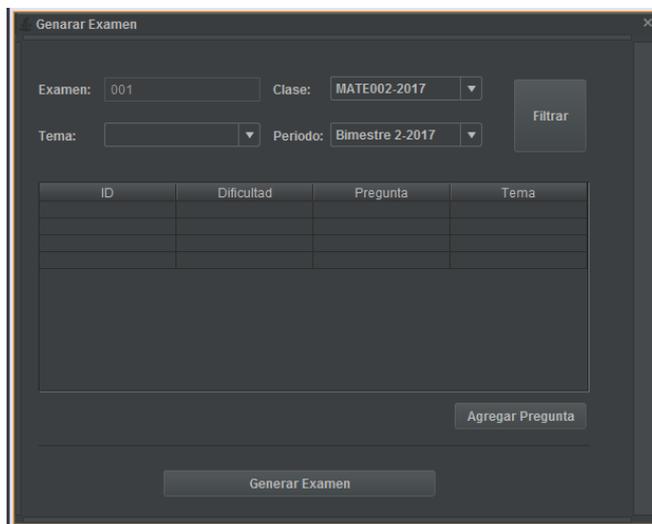
3.2.1.1. Diseño de Generar Examen

Figura N°A14



3.2.1.2. Crear Generar Examen

Figura N°A15.



3.2.1.3. Tarjeta CRC Generar Examen

Tabla N°A31.

Generar Examen	
llenarTabla_Pregunta() generarExamen()	ExamenDAO PreguntaDAO Examen Pregunta

3.2.1.4. Testeo de Generar Examen

Casos de prueba para la ejecución de la herramienta para el docente.

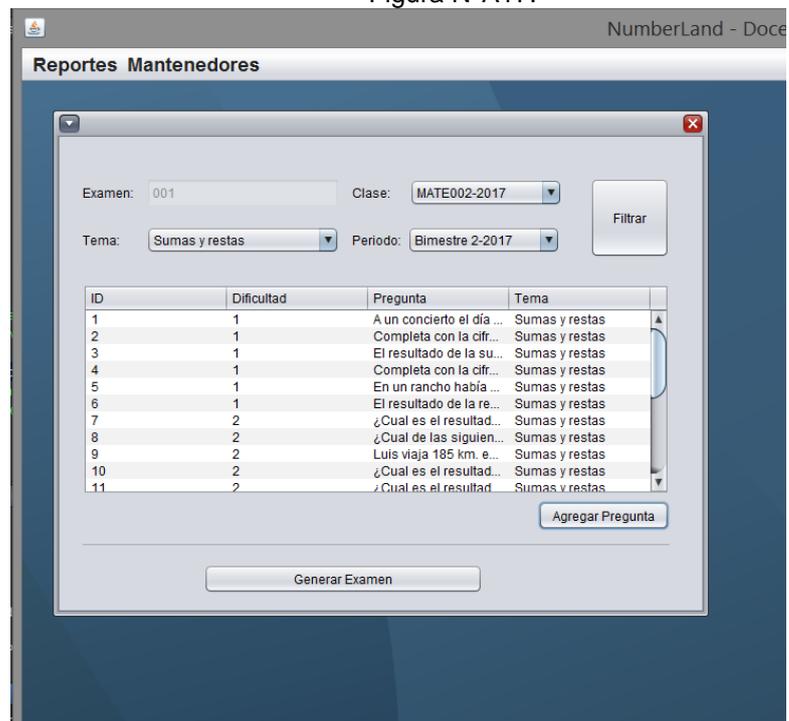
Tabla N°A32.Casos de prueba:

NRO	ACCIÓN	EJECUCIÓN DATOS ENTRADA	/ DE	ESCENARIO ESPERADO	ESCENARIO CON ERROR
CP-01	Módulo de Generación de Exámenes	Ingresar al módulo de Generación de Exámenes		Se muestra la pantalla con la lista de exámenes generados por el docente filtrados para cada clase que tiene, con la opción de Actualizar y Agregar uno nuevo.	No se muestra la pantalla.
					Aparece un mensaje de error.
CP-02	Agregar Examen	Se selecciona la opción Agregar.		Se muestra en la pantalla el Módulo de Preguntas con la lista de preguntas que se agregarán al examen que se está generando, así como las opciones de filtrado, Agregar Pregunta y Generar Pregunta.	No se muestra la pantalla.
					Aparece un mensaje de error.

Figura N°A16.



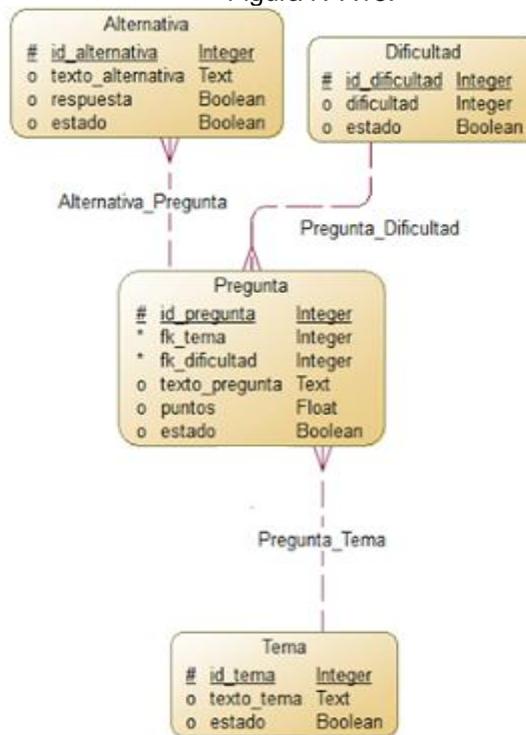
Figura N°A17.



CREAR PREGUNTA

3.2.1.5. Diseño de Pregunta

Figura N°A18.



3.2.1.6. Crear Pregunta

Figura N°A19.

PREGUNTA:

Tema:

Dificultad:

Texto:

ALTERNATIVAS: (seleccionar una como respuesta)

3.2.1.7. Tarjeta CRC Crear pregunta

Tabla N°A33.

PreguntaDAO	
Clase que guarda las preguntas de la BD: Listar(Tema) GetUltimo() Guardar(Pregunta)	Pregunta Tema

3.2.1.8. Testeo de Crear Preguntas

Casos de prueba para la ejecución de la herramienta para el docente.

Tabla N°A34. Casos de prueba:

NRO	ACCIÓN	EJECUCIÓN / DATOS DE ENTRADA	ESCENARIO ESPERADO	ESCENARIO CON ERROR
CP-01	Agregar Pregunta	Se ingresan los datos para cada campo requerido y se guardar la Pregunta.	Dentro del el Módulo de Preguntas se selecciona el Tema y dificultad de la pregunta a generar; así como su texto y sus alternativas, seleccionando la que será la respuesta correcta; y la opción Guardar.	No se muestra la pantalla. Aparece un mensaje de error.
CP-02	Lista de preguntas	Luego de guardar la Pregunta.	Se muestra la pantalla con la lista de preguntas actualizada, con las opciones Agregar Pregunta, y las opciones de filtrado por tema, clase y periodo (bimestre).	La pantalla no muestra las preguntas agregadas. Aparece un mensaje de error.

Figura N°A20.

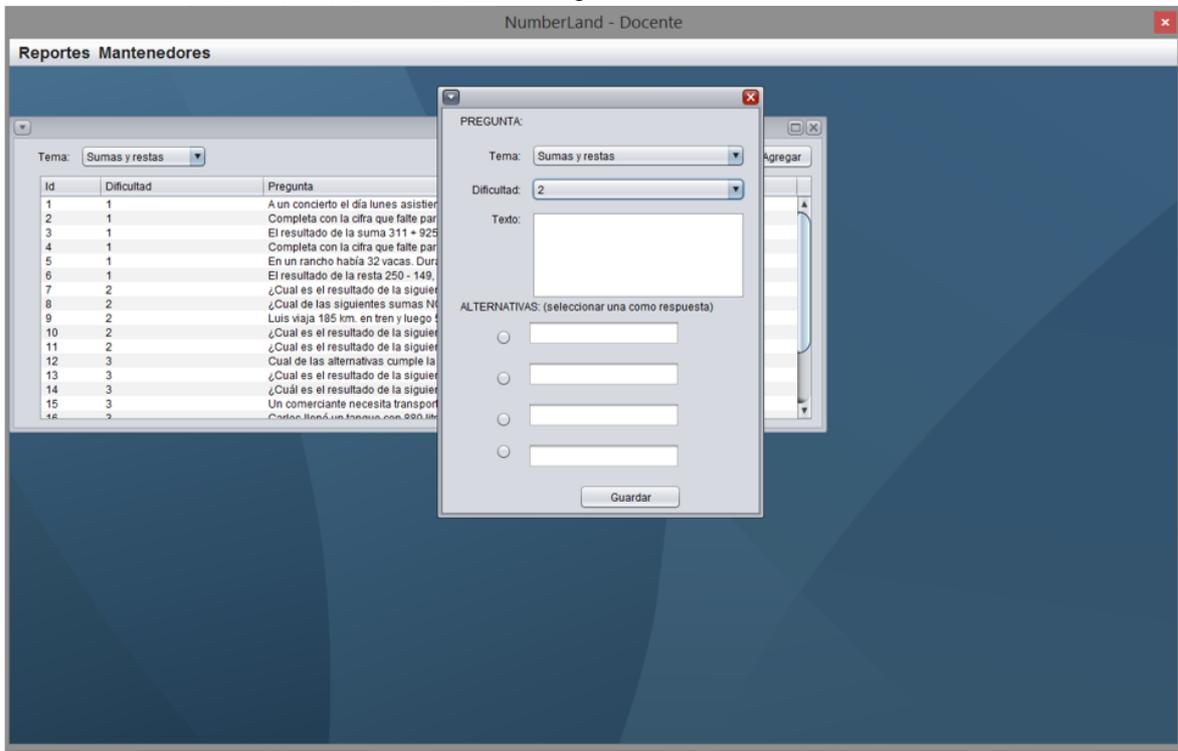
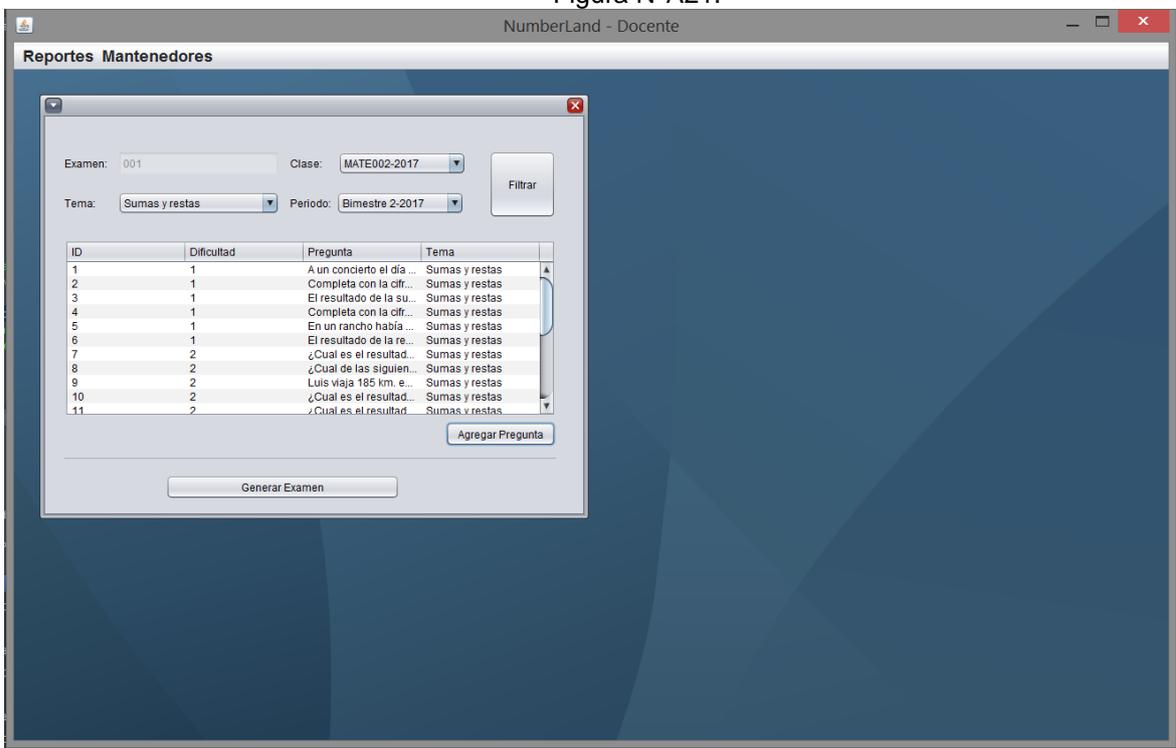


Figura N°A21.



3.3. ITERACIÓN 3:

3.3.1. HISTORIA DE USUARIO N°4

Tabla N°35. Historia de Usuario “Estructuración de avance de escolares”

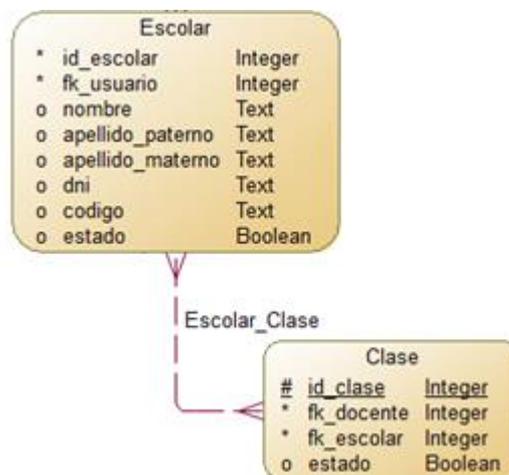
Historia de Usuario N°04: Estructuración de avance de escolares			Iteración No. 03
Sub historia	Fase	Tarea por Sub Historia	
Listado de escolares	Diseño	3.3.1.1	Diseño del listado
	Implementación	3.3.1.2	Implementar listado individual y grupal
	Pruebas	3.3.1.3	Testeo del funcionamiento del listado
Base de datos	Diseño	3.3.1.4	Diseño de base de datos
	Implementación	3.3.1.5	Implementación de Base de datos
Generación de gráficos de avance	Diseño	3.3.1.6	Diseño de Gráficos de avance
	Implementación	3.3.1.7	Implementar gráfico de avance
	Pruebas	3.3.1.8	Testeo de funcionamiento de listado y gráfico

Fuente: Elaboración propia

LISTADO DE ESCOLARES

3.3.1.1. Tarea Diseño del listado

Figura N°A22.



3.3.1.2. Implementar listado individual y grupal

Figura N°A23.



Tarea Tarjeta CRC Listar escolar

Tabla N°A36.

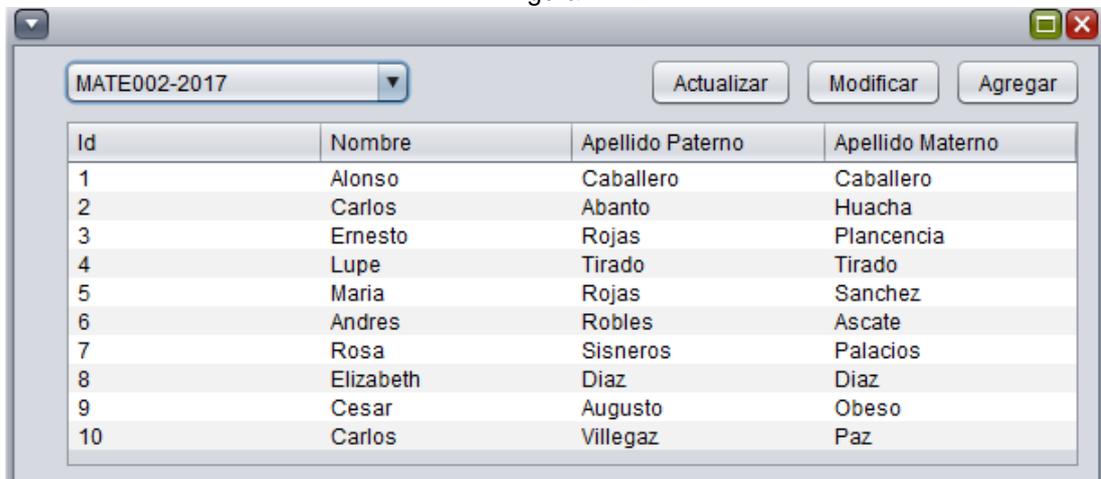
EscolarDAO	
Clase que guarda los escolares de la BD: Listar(Clase) Editar(Escolar)	Clase Escolar

3.3.1.3. Tarea Testeo del funcionamiento del listado

Casos de prueba para la ejecución de la herramienta para el docente.
Tabla N°A37. Casos de prueba:

NRO	ACCIÓN	EJECUCIÓN / DATOS DE ENTRADA	ESCENARIO ESPERADO	ESCENARIO CON ERROR
CP-01	Ejecutar módulo de mantenedor de escolares	Ingresar al módulo de mantenedor de Escolares	Se muestra la pantalla con la lista de escolares, con las opciones Actualizar, Modificar, Agregar, y la opción de filtrado por clase.	No se muestra la pantalla. Aparece un mensaje de error.

Figura N°A24.

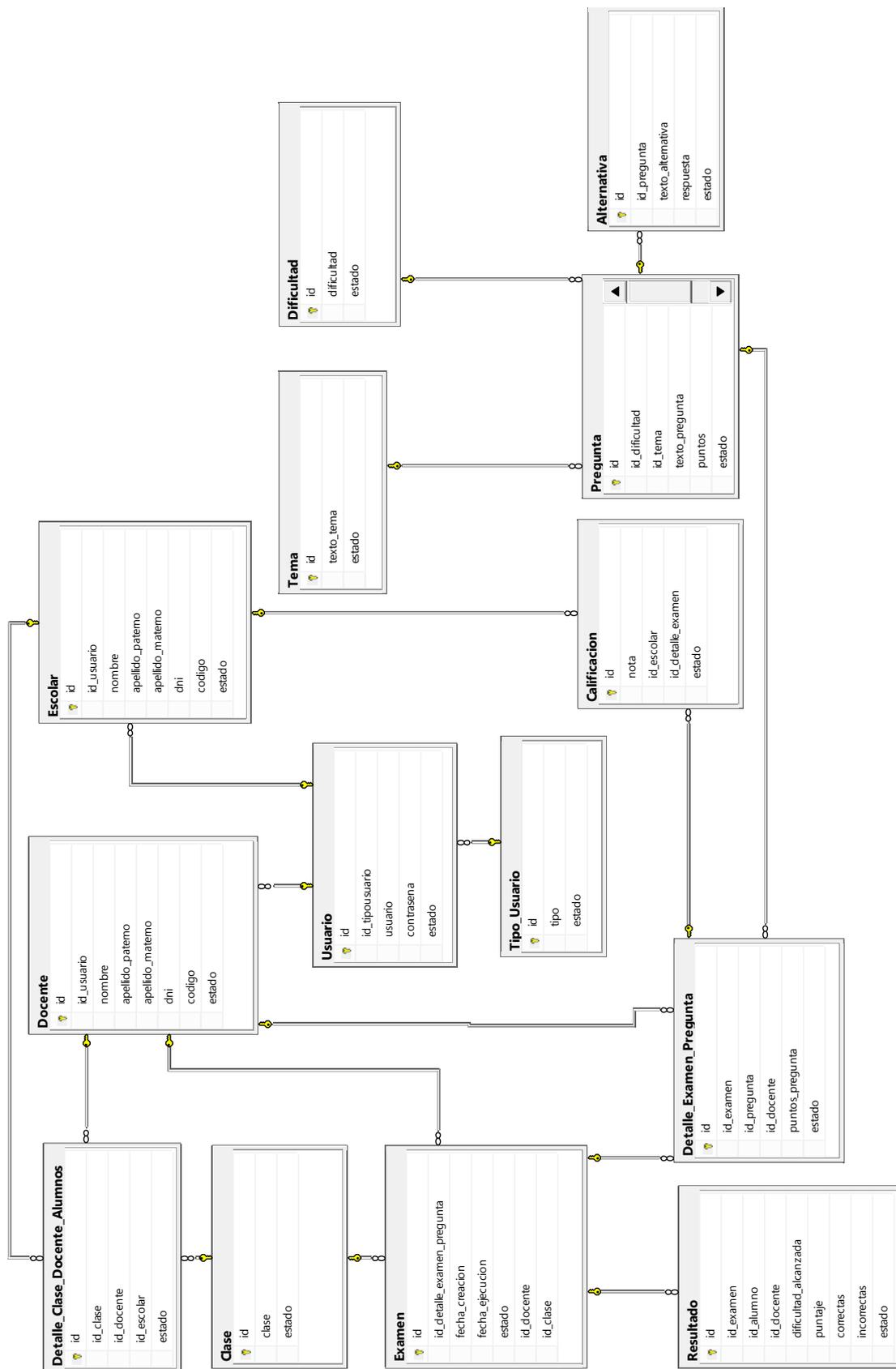


(Los nombres han sido modificados por cuestiones de seguridad)

BASE DE DATOS

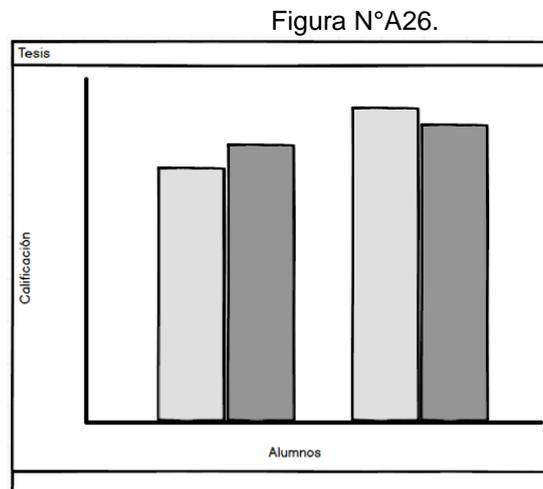
3.3.1.4. Tarea Testeo del funcionamiento del listado

Figura N°A25.



GENERACIÓN DE GRÁFICOS

3.3.1.5. Diseño de Gráficos de avance



3.3.1.6. Implementar gráficos de avance de escolares.

Utilizamos JFreeChart para los reportes y gráficos de avance de los escolares.

Figura N°A27.

```
JFreeChart chart = ChartFactory.createLineChart("Notas clase 001",
CategoryPlot plot = chart.getCategoryPlot();
plot.setRangeGridlinePaint(Color.black);
ChartFrame chartFrame = new ChartFrame("Students", chart, true);
chartFrame.setVisible(true);
chartFrame.setSize(700, 700);
ChartPanel chartPanel = new ChartPanel(chart);
```

3.3.1.7. Testeo del funcionamiento de listado y gráfico

Tabla N°A38. Casos de prueba para la ejecución de la herramienta para el docente:

NRO	ACCIÓN	EJECUCIÓN / DATOS DE ENTRADA	ESCENARIO ESPERADO	ESCENARIO CON ERROR
CP-01	Generar Reporte de Notas	Ingresar Reporte de Notas	Se muestra la pantalla para generar el reporte de acuerdo a los filtros: por Clase y Periodo (bimestre); o por escolar(es)	No se muestra el reporte en pantalla.
				Aparece un mensaje de error.
CP-02	Guardar Reporte de Notas	Guardar imagen del reporte generado	Click derecho sobre la imagen del reporte generado. Selecciona la opción "Guardar como...", seleccionar la extensión "PNG".	No se muestra el reporte en pantalla.
				Aparece un mensaje de error.

Figura N°A28.

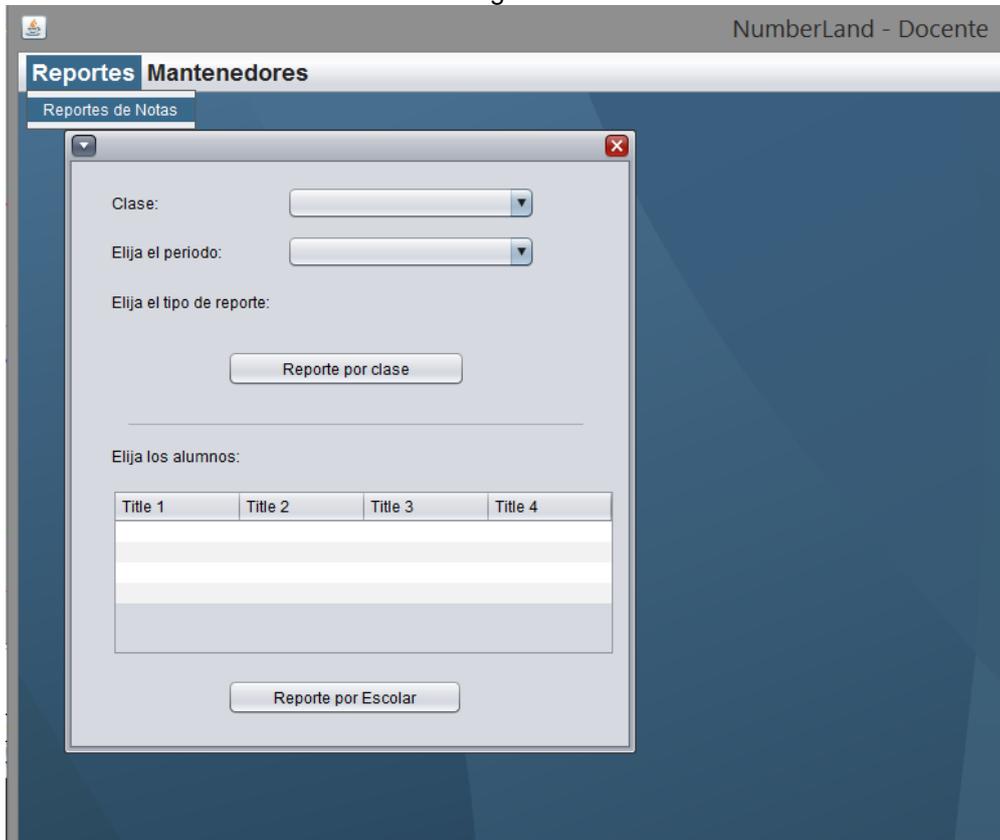
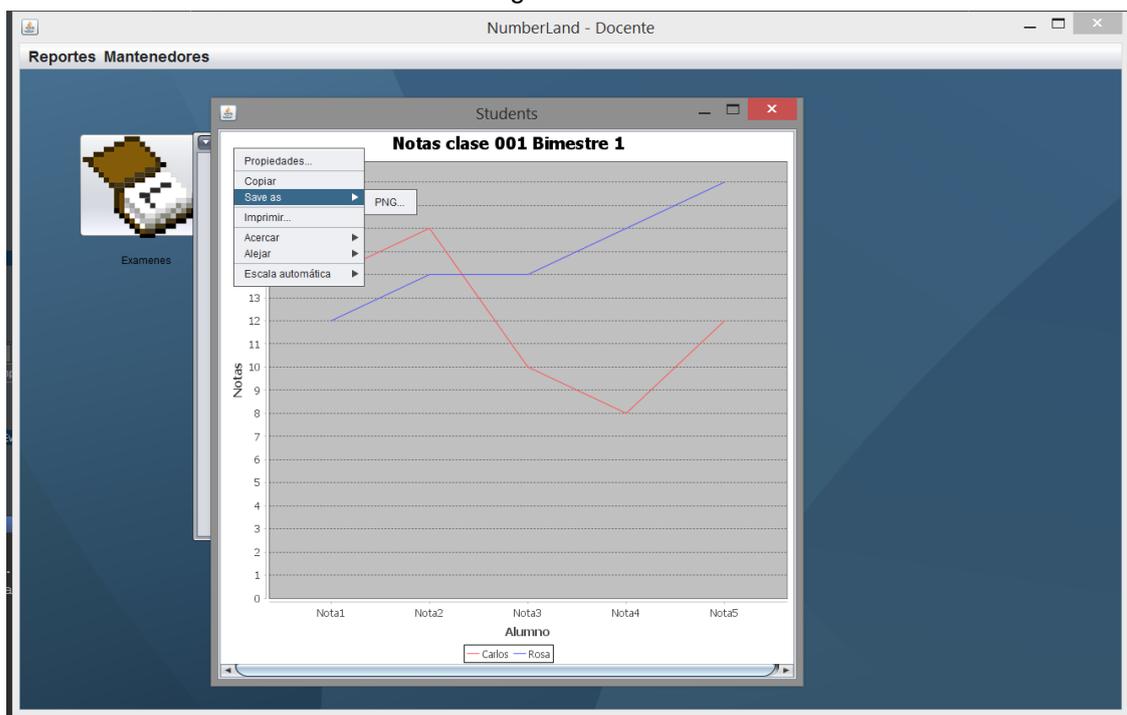


Figura N°A29.



3.4. ITERACIÓN 4:

3.4.1. HISTORIA DE USUARIO N°5

Tabla N°39. Historia de Usuario “Desarrollo de Interfaz del Videojuego educativo”

Historia de Usuario N°05: Desarrollo de Interfaz del Videojuego educativo		Iteración No. 04	
Sub historia	Fase	Tarea por Sub Historia	
Funcionalidad del videojuego	Diseño	3.4.1.1	Diseño de niveles
	Implementación	3.4.1.2	Implementar colisiones, animaciones y funcionalidad
	Pruebas	3.4.1.3	Testeo de funcionalidad del videojuego

Fuente: Elaboración propia

FUNCIONALIDAD DEL JUEGO

3.4.1.1. Diseño de niveles

Figura N°A30.



Diseño de los Tiles, los cuales son los obstáculos que harán que no se pueda pasar de largo. Existen varias capas de imágenes en el juego, primero están los tiles de obstáculos, después el fondo o en este caso las áreas verdes por donde transitaremos y finalmente, las capas de objetos que dibujarán las áreas no transitables y que en el videojuego se las programa para dicha función.

Figura N°A31.



3.4.1.2. Implementar colisiones y animaciones:

Ya en el software, estos objetos se traducen en rectángulos o formas geométricas, las cuales tienen una posición y un perímetro definido. De acuerdo a estos datos, se le programa las colisiones a cada uno de ellos mediante un for each de cada capa de objetos (capa de rectángulos transparentes de la imagen superior).

Figura N°A32.

```
for (MapObject object : map.getLayers().get("arboles").getObjects().getByType(RectangleMapObject.class)) {
    Rectangle rectangle = ((RectangleMapObject) object).getRectangle();
    bodyDef.type = BodyDef.BodyType.StaticBody;
    bodyDef.position.set(rectangle.getX() + rectangle.getWidth() / 2, rectangle.getY() + rectangle.getHeight() / 2);

    body = world.createBody(bodyDef);
    shape.setAsBox(rectangle.getWidth() / 2, rectangle.getHeight() / 2);
    fixtureDef.shape = shape;
    body.createFixture(fixtureDef);
}

for (MapObject object : map.getLayers().get("casas").getObjects().getByType(RectangleMapObject.class)) {
    Rectangle rectangle = ((RectangleMapObject) object).getRectangle();
    bodyDef.type = BodyDef.BodyType.StaticBody;
    bodyDef.position.set(rectangle.getX() + rectangle.getWidth() / 2, rectangle.getY() + rectangle.getHeight() / 2);

    body = world.createBody(bodyDef);
    shape.setAsBox(rectangle.getWidth() / 2, rectangle.getHeight() / 2);
    fixtureDef.shape = shape;
    body.createFixture(fixtureDef);
}
```

3.4.1.3. Testeo de funcionalidad del videojuego

Casos de prueba para la ejecución de la herramienta para el docente.

Tabla N°A40. Casos de prueba:

NRO	ACCIÓN	EJECUCIÓN / DATOS DE ENTRADA	ESCENARIO ESPERADO	ESCENARIO CON ERROR
CP-01	Colisionar con algún elemento del mapa (Tile) que no sea un personaje.	Movimientos con las teclas direccionales.	Nuestro personaje principal se choca y no podrá avanzar debido al obstáculo.	El personaje vuelve una posición atrás. Aparece un mensaje de error.

3.5. ITERACIÓN 5:

3.5.1. HISTORIA DE USUARIO N°6

Tabla N°41. Historia de Usuario "Desarrollo de RNA"

Historia de Usuario N°06: Desarrollo de RNA			Iteración No. 05
Sub historia	Fase	Tarea por Sub Historia	
Desarrollo de RNA	Diseño	3.5.1.1	Diseño de red neuronal artificial
	Implementación	3.5.1.2	Implementar red neuronal artificial
	Pruebas	3.5.1.3	Testeo de funcionamiento de RNA

Fuente: Elaboración propia

3.5.1.1. Diseño de la red neuronal artificial

Datos de entrada:

- Dificultad actual.
- Puntaje acumulado.
- Tiempo empleado en la pregunta.
- Numero de preguntas resueltas.

Datos de salida:

- Subir dificultad.
- Mantener dificultad.
- Bajar dificultad.

3.5.1.2. Implementar red neuronal artificial

METODOLOGÍA DE DESARROLLO DE LA RED NEURONAL

Lo primero es pasar el conjunto de datos (de excel) hacia una matriz numérica, ya que, la red neuronal trabaja con matrices para realizar los cálculos de cada neurona. Entonces se debe pasar los datos que ya se tienen como una base de conocimiento para que se puedan realizar las predicciones del nivel de juego al que pasará el jugador en base a las características con las que se cuenta.

1. Lo que tenemos que hacer es transformar la información en datos numéricos, y como se puede apreciar tenemos en los últimos valores los datos en forma de texto, los cuales se les asignará una etiqueta para asignarle un número. Las etiquetas son (mantener = 1, avanzar = 2, anterior = 0).

Tabla N°A42.

DATOS ORIGINALES	DATOS MODIFICADOS
0,2,2,1,mantener	0,2,2,1,1
0,4,6,2,mantener	0,4,6,2,1
0,6,2,3,avanzar	0,6,2,3,2
1,8,2,4,mantener	1,8,2,4,1
1,10,3,5,mantener	1,10,3,5,1
1,12,4,6,mantener	1,12,4,6,1
2,18,4,9,mantener	2,18,4,9,1
2,16,2,8,avanzar	2,16,2,8,2
2,14,5,9,anterior	2,14,5,9,0
0,16,2,9,avanzar	0,16,2,9,2
0,8,5,6,mantener	0,8,5,6,1
1,10,2,5,mantener	1,10,2,5,1
1,12,4,9,anterior	1,12,4,9,0
2,14,7,8,anterior	2,14,7,8,0
2,18,2,9,mantener	2,18,2,9,1
0,2,5,1,mantener	0,2,5,1,1
1,8,4,7,anterior	1,8,4,7,0

2. Una vez convertida la data separamos en dos matrices "X" y "Y", la matriz "X" contiene las características para indicar como se obtendrá el nuevo valor de predicción de "Y", en este caso todas las columnas excepto la última son las características seleccionadas para que trabajen con la red neuronal, y la matriz "Y" es la última columna es decir las etiquetas que debemos de encontrar con nuevos valores que nos dará aplicación del videojuego.

Figura N°A33. Matriz "X" de características

0.0	2.0	2.0	1.0
0.0	4.0	6.0	2.0
0.0	6.0	2.0	3.0
1.0	8.0	2.0	4.0
1.0	10.0	3.0	5.0
1.0	12.0	4.0	6.0
2.0	18.0	4.0	9.0
2.0	16.0	2.0	8.0
2.0	14.0	5.0	9.0
0.0	16.0	2.0	9.0
0.0	8.0	5.0	6.0
1.0	10.0	2.0	5.0
1.0	12.0	4.0	9.0
2.0	14.0	7.0	8.0
2.0	18.0	2.0	9.0
0.0	2.0	5.0	1.0
1.0	8.0	4.0	7.0

Figura N°A34. Matriz "Y" de objetivo

1.0
1.0
2.0
1.0
1.0
1.0
1.0
2.0
0.0
2.0
1.0
1.0
0.0
0.0
1.0
1.0
0.0

3. Ahora lo que se debe agregar a la matriz "X" es una columna con valores de unos (1.0) que se va a usar como valor de interceptación, es decir que este valor es el que permite minimizar el error en el cálculo de la red neuronal, es indispensable ya que sin este valor se pueden obtener porcentajes no tan buenos. Además que sirve para hacer los valores de predicción en cada capa oculta de la

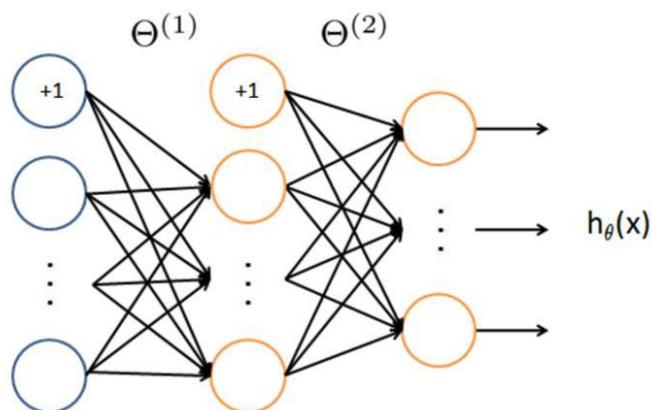
red neuronal, cuándo usamos backpropagation, este valor no se considera ya que el cálculo en backpropagation se realiza sólo de las neuronas que el algoritmo usa desde el inicio como características. El resultado es el siguiente:

Figura N°A35.

1.0	0.0	2.0	2.0	1.0
1.0	0.0	4.0	6.0	2.0
1.0	0.0	6.0	2.0	3.0
1.0	1.0	8.0	2.0	4.0
1.0	1.0	10.0	3.0	5.0
1.0	1.0	12.0	4.0	6.0
1.0	2.0	18.0	4.0	9.0
1.0	2.0	16.0	2.0	8.0
1.0	2.0	14.0	5.0	9.0
1.0	0.0	16.0	2.0	9.0
1.0	0.0	8.0	5.0	6.0
1.0	1.0	10.0	2.0	5.0
1.0	1.0	12.0	4.0	9.0
1.0	2.0	14.0	7.0	8.0
1.0	2.0	18.0	2.0	9.0
1.0	0.0	2.0	5.0	1.0
1.0	1.0	8.0	4.0	7.0

4. Luego de agregar el valor interceptor (columna de unos), se debe generar los valores que se usarán para predecir los nuevos datos que se obtengan por medio del videojuego, en este al usar dos capas, una capa oculta y otra capa para los valores de salida, la arquitectura de la red neuronal es de la siguiente manera.

Figura N°A36.



Se puede apreciar el valor (+1) que es el interceptor en capa. Ahora la matriz de características “theta” como se aprecia en la imagen el símbolo de “theta1” y “theta2” son las que crearemos para

almacenar los valores que obtendremos al final como valores óptimos para predecir nuevas instancias por medio del videojuego. Para “theta1” se obtiene lo siguiente:

Figura N°A37.

0.07478368	-0.7103865	0.42097843	0.87796664	-0.7378656
-0.5488599	0.8990146	-0.48546988	0.99944067	-0.90651673
-0.3494026	0.12020934	0.072169304	-0.123048484	0.6910342
0.92520523	-0.024458826	0.26533246	0.7092999	0.7304143

Los valores de “theta1” se obtienen de multiplicar la capa de nodos color azul con los valores aleatorios asignados a “theta1” que están en el rango [-1, 1]. Por cada capa se debe hacer el cálculo, entonces para “theta2” de igual manera se obtiene lo siguiente:

-0.45875812	-0.755332	-0.3932296	-0.18474251	0.86678505
0.15012002	0.20063698	-0.47189337	0.7944459	0.9061414
-0.5878922	0.6972444	-0.16342139	0.021477103	0.84054554

5. Una vez finalizado el cálculo de los valores de “theta1” y “theta2”, tenemos que realizar el cálculo de la capa final para obtener los valores de predicción para nuestro conjunto de datos que cargamos inicialmente desde el archivo de Excel, entonces obtenemos lo siguiente haciendo la multiplicación de los valores de “theta2” y nuestra capa oculta.

Figura N°A38.

0.32024574	0.19835022	0.87426525
0.29495403	0.14823246	0.89632285
0.30603945	0.17304897	0.8865914
0.29234362	0.16026127	0.89709324
0.2903535	0.14627586	0.9001298
0.2890063	0.13778074	0.9022407
0.28484178	0.12827215	0.90669674
0.28540915	0.13777687	0.9049223
0.28538838	0.14155735	0.90437186
0.29764193	0.1553515	0.89296466
0.29846144	0.16423537	0.89099914
0.2915043	0.15328224	0.8984724
0.2921251	0.16806187	0.89557457
0.2846653	0.12820464	0.9068324
0.28541708	0.13430403	0.90540177
0.2985996	0.15900092	0.89246255
0.29177365	0.1888948	0.89321524

6. Finalmente lo que debemos realizar es obtener el máximo valor de cada fila, ya que la red neuronal lo que ha realizado es el cálculo por cada uno de nuestros valores objetivos de nuestra

matriz “Y”, es por eso que obtenemos 3 columnas ya que tenemos (mantener = 1, avanzar = 2, anterior = 0). Al realizar la predicción se obtiene lo siguiente:

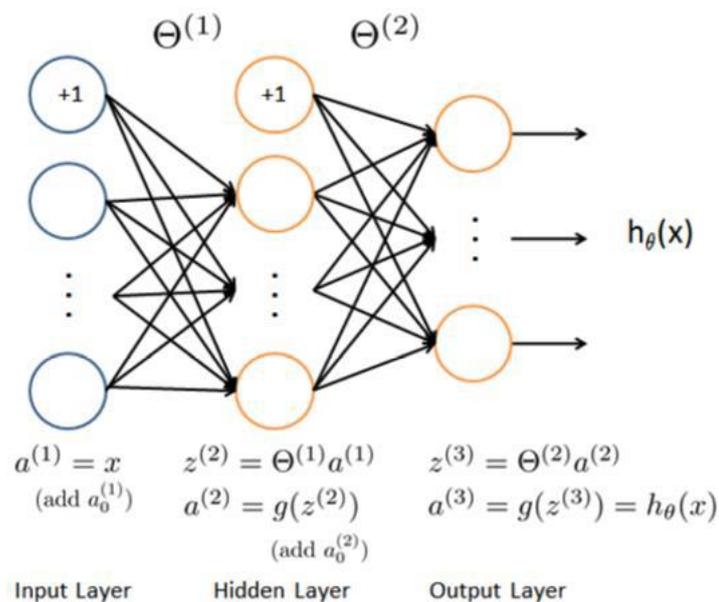
Figura N°A39.

2
 2
 0
 1
 1
 2
 1
 1
 2
 1
 2
 1
 2
 2
 1
 2
 2

Cada valor que se aprecia indica si es mantener = 1, avanzar = 2, anterior = 0. Que son las predicciones realizadas por la red neuronal para cada uno de los registros de nuestra base de conocimientos con la cuál entrenamos a la red neuronal.

Una formulación de manera matemática para el cálculo de la red neuronal es la siguiente:

Figura N°A40.



3.5.1.3. Testeo de funcionamiento de RNA

Después los resultados pueden ser mejorados mediante backpropagation con diversas iteraciones dependiendo de los datos de entrada proporcionados cuidadosamente por los docentes expertos en su área. Al principio se tenían datos algo decepcionantes como lo fue un 17% de efectividad de la RNA, pero conforme se mejoraba e iteraba, este porcentaje comenzó a subir hasta llegar a un 74.28% de efectividad, siendo la mejor matriz dado que otros resultados arrojan menores porcentajes.

Tabla N°A43. Evolución de la RNA.

0.87555695	-0.10438073	-0.9209528	-0.6187263	0.7383038
0.96224594	-0.18915367	0.9150449	0.49410355	0.57943773
-0.049113393	0.5839467	-0.5568322	-0.3568473	-0.7346476
0.44320858	0.6979226	-0.7467757	-0.65584946	0.8579843
0.75443137	0.43032277	0.17879224	0.43417943	0.018707156
0.6128949	-0.6188917	-0.23104316	0.78333616	-0.7653207
-0.8631481	0.67797005	0.6752267	-0.15058845	-0.580114
Exactitud: 17.647058				

-0.71269727	0.6397468	0.8696722	0.34563005	0.62932086
0.7835971	-0.96159303	0.5797709	0.18044591	-0.1882357
0.589674	0.67480826	-0.5384836	-0.67301166	-0.1229887
0.11752927	0.5359813	0.81241155	0.068455696	-0.9538241
0.82454395	0.0338943	-0.3059088	0.84053636	0.07471597
0.2751732	0.71718824	-0.91529554	0.4887588	0.67740655
-0.202546	-0.49684703	0.7341955	-0.3715198	-0.83313656
Exactitud: 23.529411				

0.23144495	-0.5513786	-0.44570112	-0.5477874	-0.05294907
------------	------------	-------------	------------	-------------

-0.32257527	-0.2810912	0.19292569	-0.6792966	0.28355956
0.117084265	0.8000704	-0.43300617	0.7585145	-0.4861874
0.40331817	0.77632487	0.14160705	0.36801124	-0.84371483

-0.73040557	0.6289425	-0.9160501	-0.35014063	0.49038804
0.64122224	0.09513223	-0.16749674	-0.013989985	0.64636064
-0.48736745	-0.79678273	0.030636907	-0.76205254	0.6030859

Exactitud: 35.294117

0.053230524	0.61621535	0.7334863	0.046730638	0.1676054
0.97590053	-0.738155	0.34511006	0.7600708	-0.032844245
-0.25765312	0.63977337	0.14207947	0.6402221	-0.7845162
-0.897733	-0.96194506	-0.70517945	0.36337996	-0.5242177

-0.18019837	0.06062913	0.35615444	-0.49528825	-0.84870386
0.13888717	-0.39802188	0.71420753	0.3844937	-0.4967168
0.48853648	-0.7197221	0.24405527	0.57117426	-0.94288325

Exactitud: 52.941177

-0.68932545	0.14047933	-0.88798904	0.19488978	0.59734297
0.42349434	-0.73966026	0.70806134	0.16964328	-0.44372332
0.25341642	0.9012064	-0.68316436	0.88984823	0.16938221
-0.032364666	-0.3498969	0.74300647	-0.27981007	0.9400208

-0.6204298	-0.96808857	-0.52469015	0.24015844	0.8033141
-0.14094496	0.84115887	0.23306417	0.35022545	0.11294508
-0.40778685	0.01740086	0.44240582	-0.86821496	0.9548421

Exactitud: 58.823532

0.64377904	0.44343078	0.65101624	0.10316265	0.42169106
0.57247293	-0.035933197	-0.5752312	-0.89577276	-0.18440348

-0.14927804 0.756981 -0.39909804 -0.563776 -0.7351458
0.8719286 0.53619945 0.90031445 0.82384956 -0.4479977

0.22237837 0.27627456 0.4444809 0.06549978 0.12682319
-0.39766884 -0.733552 0.27809703 0.8210653 -0.0996176
-0.55847067 -0.38470185 0.783358 0.9157351 0.4073

Exactitud: 61.11111

-0.6299952 0.6432874 -0.22910541 0.47125554 0.68747616
-0.23361069 0.072532296 0.2751963 0.699407 0.31801295
-0.83026516 -0.3502314 0.3278619 0.34004092 -0.22239423
0.57611036 -0.12195748 -0.741269 -0.98555267 -0.9288071

-0.118434906 0.5942913 -0.20650697 -0.46939415 0.9933286
0.48986888 0.95332026 -0.34676212 0.6407654 -0.6606896
-0.38422638 -0.31868714 0.094968915 -0.28311813 0.26218164

Exactitud: 61.11111

-0.6990369 0.9123814 -0.43233913 0.5203216 0.51722145
0.12720275 0.253834 -0.3905751 -0.117274344 -0.28158945
-0.32409966 -0.18377364 -0.9854439 0.14905798 0.89924014
-0.171722 -0.61716723 -0.16185385 0.16494358 -0.10593009

-0.88066995 0.49404085 -0.18015844 0.72779024 -0.42227387
0.55409205 0.27774084 0.5784241 -0.14253813 -0.35590833
-0.15018141 -0.39563394 0.52076316 0.81690836 0.08095348

Exactitud: 66.66667

-0.7937282 0.87927616 0.55587184 -0.29772323 0.18534577
0.409513 -0.54445654 0.42189682 -0.09114063 -0.34487623
-0.1987809 -0.56795967 0.19523323 -0.4073515 0.9488348
-0.24416238 -0.75932974 0.7178191 -0.4008544 -0.22330499

-0.73608875	0.6963028	0.40554106	-0.5575924	-0.15871388
0.64866495	0.77340674	0.93799317	0.38426793	0.20779383
0.2729671	0.28678107	0.41681874	-0.979764	-0.9261262

Exactitud: 72.0

0.788049	0.3937484	-0.4059915	0.7256384	0.35781467
-0.7131638	-0.03308022	-0.58234906	-0.0681206	-0.4916284
-0.24178159	-0.23088324	0.9355985	-0.95528775	-0.63557374
-0.65388036	-0.17232287	-0.18064606	-0.19174719	0.50000226

-0.62809277	-0.255719	-0.53882086	0.93018043	0.41162825
0.535043	-0.9010007	-0.4244659	0.30200493	0.6052946
-0.27627236	0.8737432	-0.08502293	0.13192105	-0.99869984

Exactitud: 73.333336

0.032107234	-0.48615962	0.79486394	0.855585	0.23853147
0.6753856	-0.6925923	0.75593495	-0.5667248	0.5786716
0.38746583	0.9439857	0.5932449	-0.71407723	-0.2859354
0.11284208	-0.6518376	-0.20413184	-0.97096103	-0.53691614

-0.535703	-0.9353592	0.70818007	-0.31604832	-0.42182356
0.22076297	0.57177126	-0.09266496	-0.5493983	0.38956165
-0.17309141	0.867291	-0.041444063	0.0052491426	-0.3624835

Exactitud: 74.28571

3.6. ITERACIÓN 6:

3.6.1. HISTORIA DE USUARIO N°7

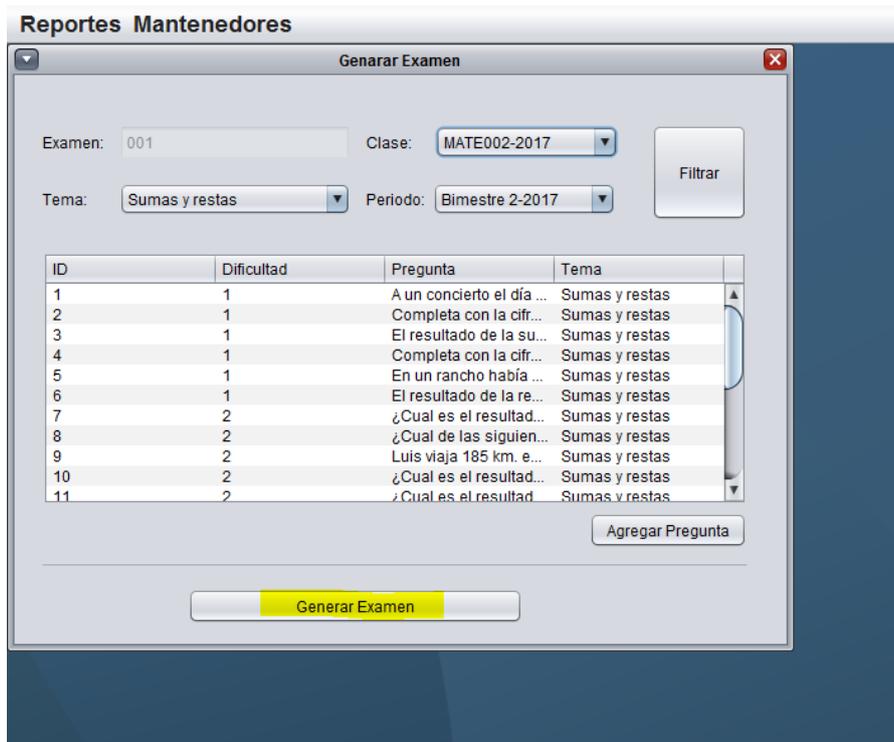
Tabla N°44. Historia de Usuario “Integración de los módulos”

Historia de Usuario N°07: Integración de los módulos			Iteración No. 06
Sub historia	Fase	Tarea por Sub Historia	
Integración del videojuego	Implementación	3.6.1.1	Implementar unión de módulos
	Pruebas	3.6.1.2	Testeo del funcionamiento de los módulos

Fuente: Elaboración propia

3.6.1.1. Implementar unión de módulos

Figura N°A41.



Reportes Mantenedores

Generar Examen

Examen: 001 Clase: MATE002-2017 Filtar

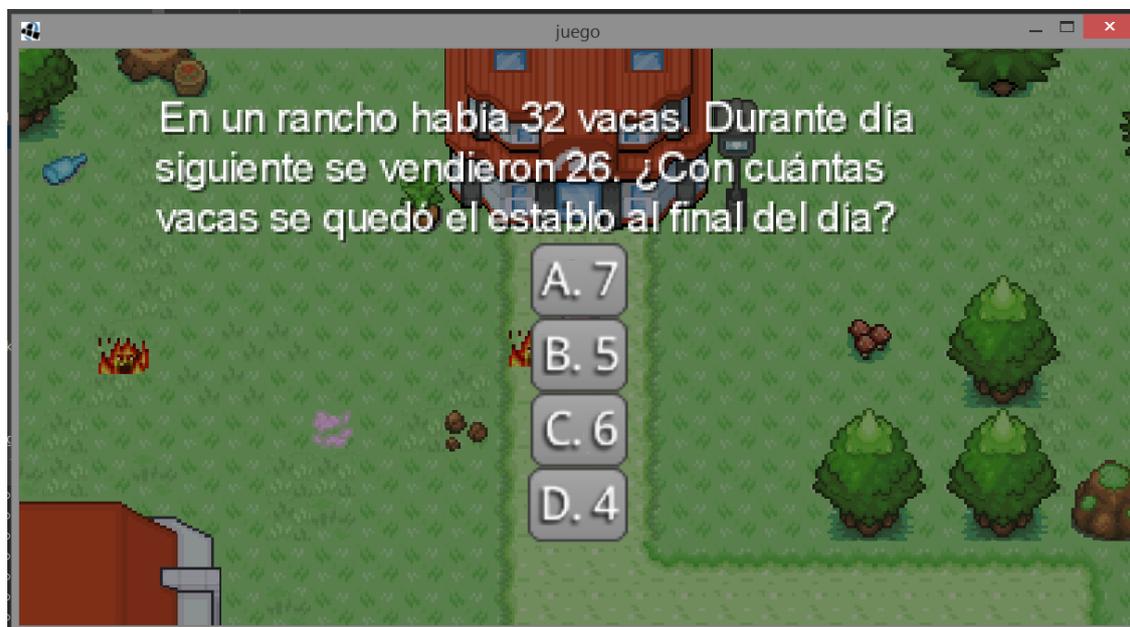
Tema: Sumas y restas Periodo: Bimestre 2-2017

ID	Dificultad	Pregunta	Tema
1	1	A un concierto el día ...	Sumas y restas
2	1	Completa con la cifr...	Sumas y restas
3	1	El resultado de la su...	Sumas y restas
4	1	Completa con la cifr...	Sumas y restas
5	1	En un rancho había ...	Sumas y restas
6	1	El resultado de la re...	Sumas y restas
7	2	¿Cual es el resultad...	Sumas y restas
8	2	¿Cual de las siguien...	Sumas y restas
9	2	Luis viaja 185 km. e...	Sumas y restas
10	2	¿Cual es el resultad...	Sumas y restas
11	2	¿Cual es el resultad...	Sumas y restas

Agregar Pregunta

Generar Examen

Figura N°A42.

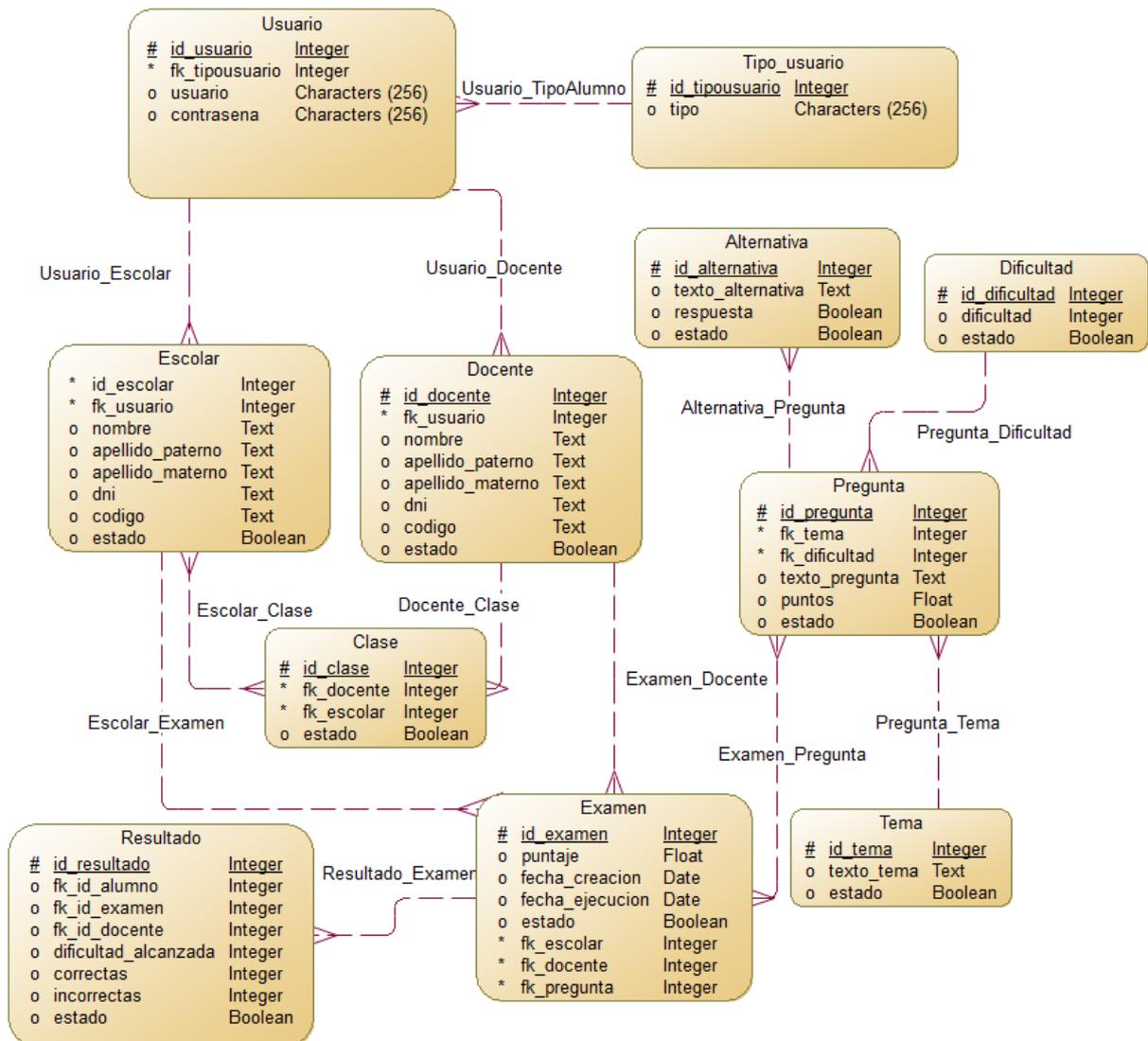


3.6.1.2. Pruebas de funcionamiento de los módulos

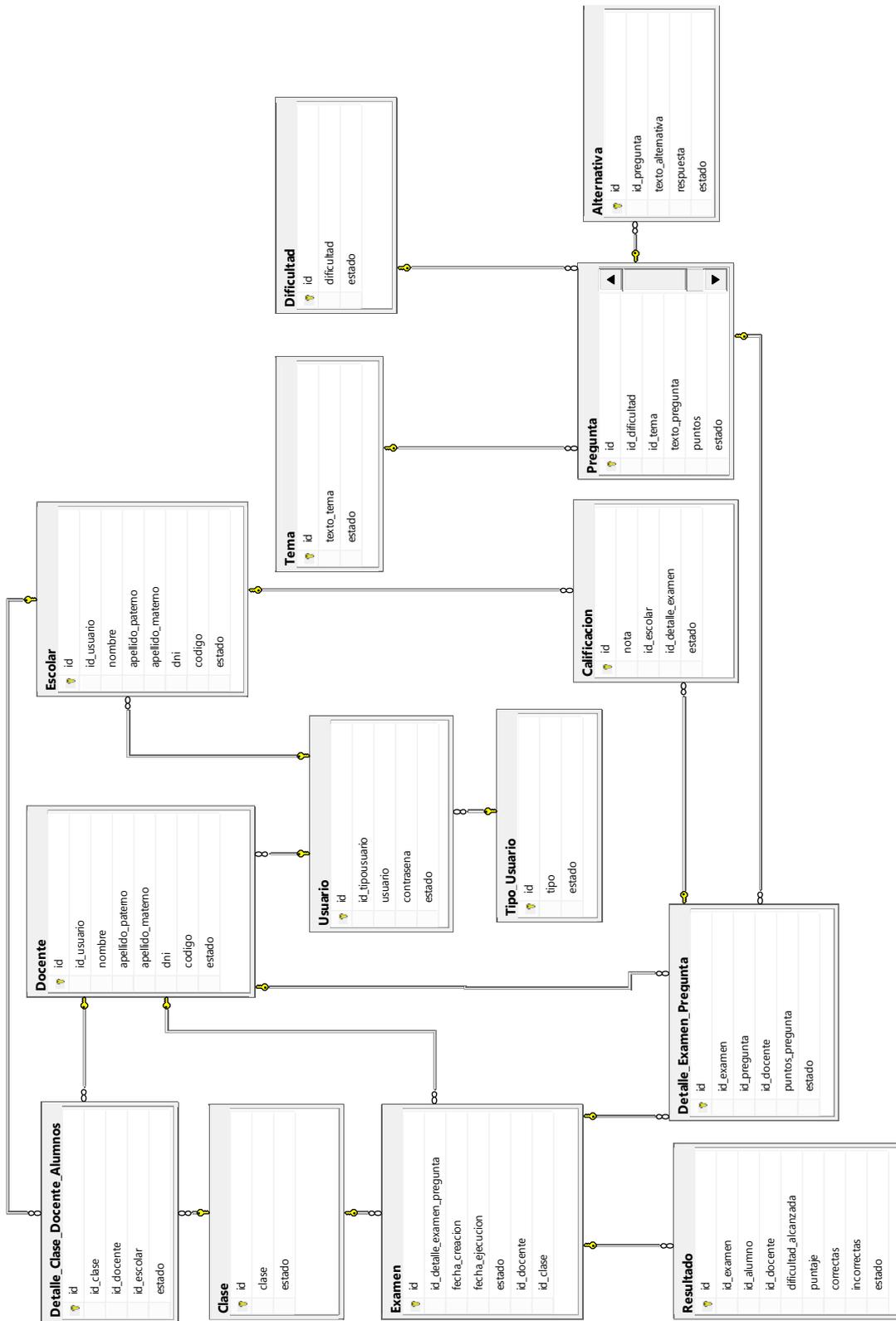
Tabla N°A45.

NRO	ACCIÓN	EJECUCIÓN / DATOS ENTRADA DE	ESCENARIO ESPERADO	ESCENARIO O CON ERROR
CP-01	El examen generado deberá mostrarse para los escolares que ingresen al apartado del videojuego.	Colisionar con un enemigo para accionar una pregunta. Datos: dificultad actual	Generar la pregunta de la dificultad señalada	Aparece un mensaje de error.
CP-02	El examen deberá generarse sin problemas, considerando el Tema.	Clase, Tema, Periodo.	Se mostrarán todas las preguntas posibles que abarcará el examen y se genera el mismo en la base de datos para comunicarse después con el videojuego	Aparece un mensaje de error.

Modelo de Datos



Modelo Físico



Modelo de Componentes

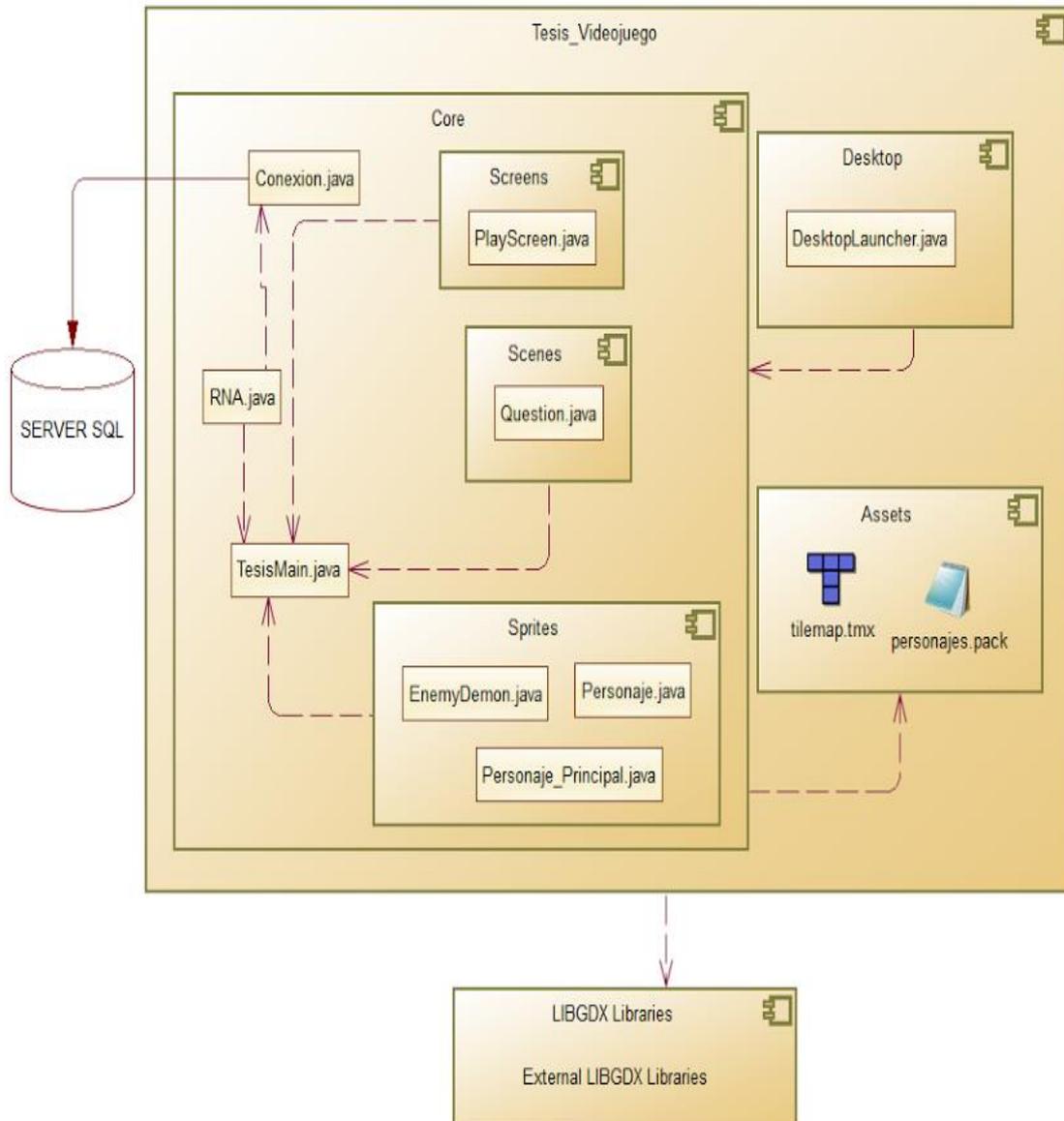
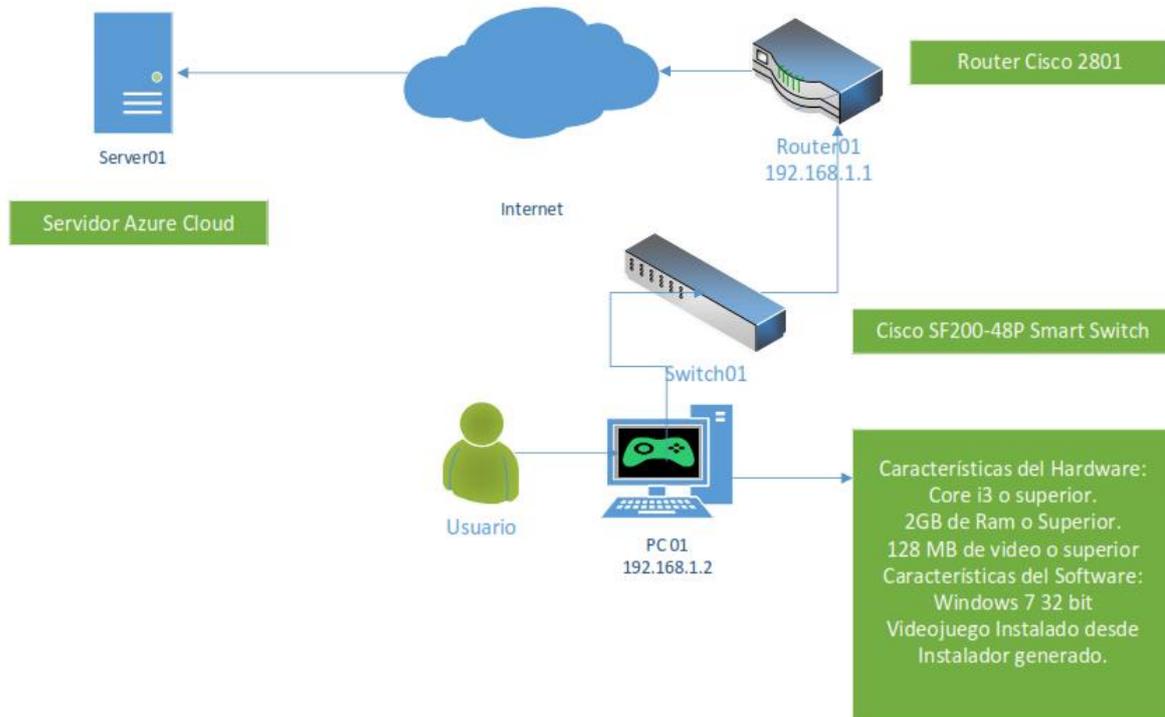
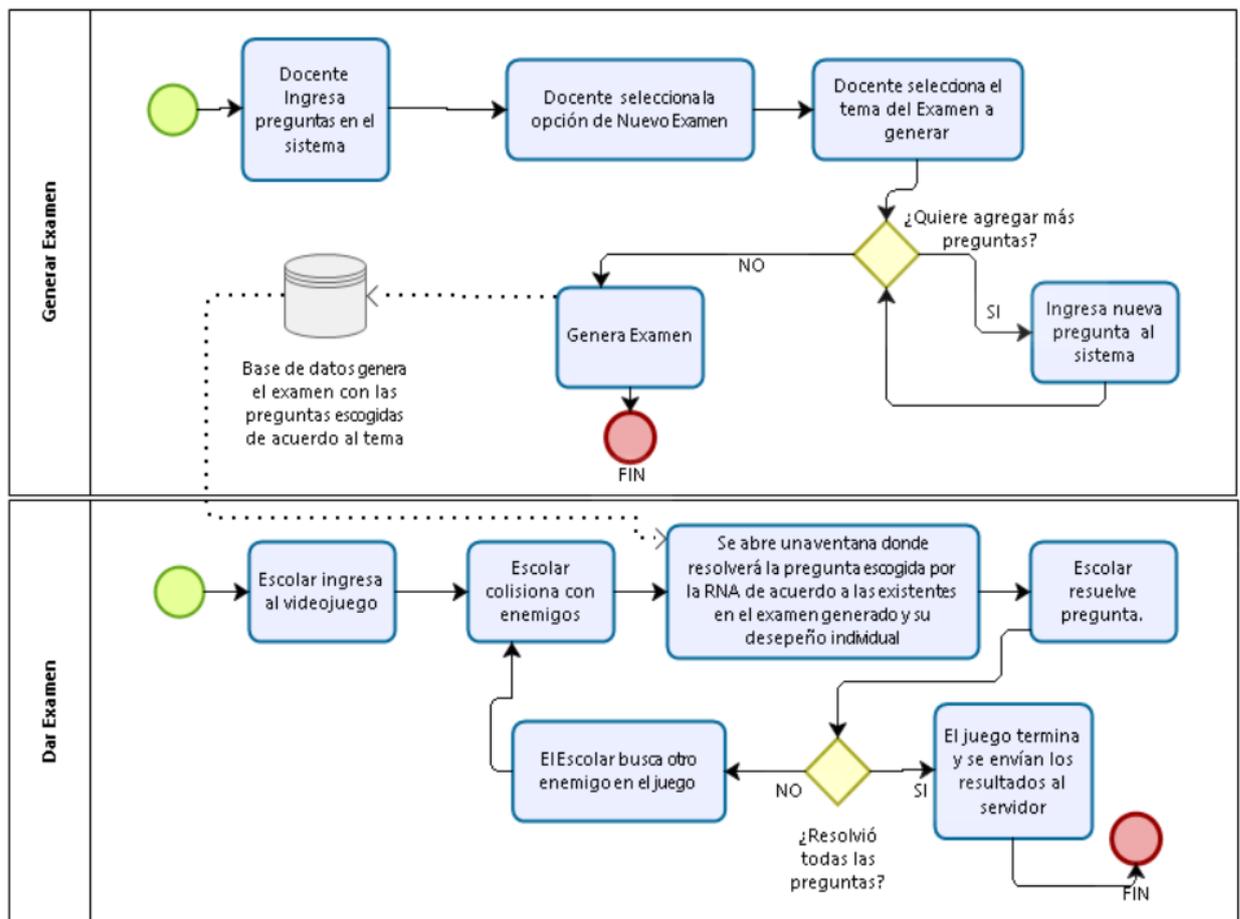


Diagrama de Despliegue



ANEXO N°4

DIAGRAMA DE PROCESOS



ANEXO N°5

RESULTADOS DE TIEMPOS DE PRUEBA.

- SIN EL VIDEOJUEGO.

Tareas / Docentes	Esquematizar y delimitar el tema	Esquematizar y realizar las preguntas	Seleccionar las preguntas para el examen	Hacer el formato del examen	Impresión y otros tiempos	Corregir exámenes	Pasar notas al registro
1	20	120	15	20	20	50	20
2	30	50	15	20	20	30	10
3	20	75	10	30	15	30	10
4	25	80	10	30	15	40	15
5	20	90	10	20	15	40	15

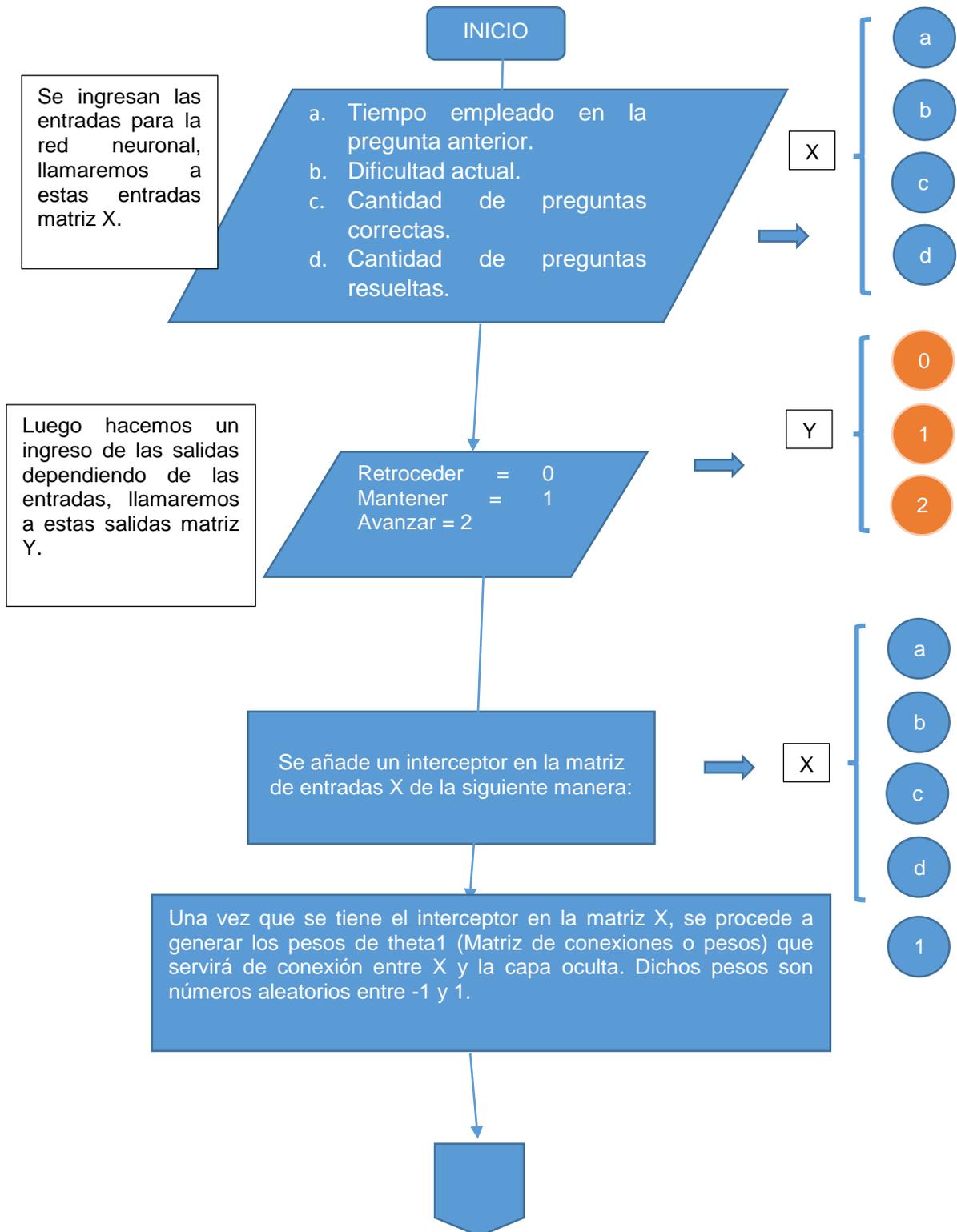
- Escala en minutos
- No incluye tiempos extra como: comidas o movilidad.
- Los tiempos son los aproximados brindados por los docentes.
- Escala de 30 escolares y 30 preguntas por examen.

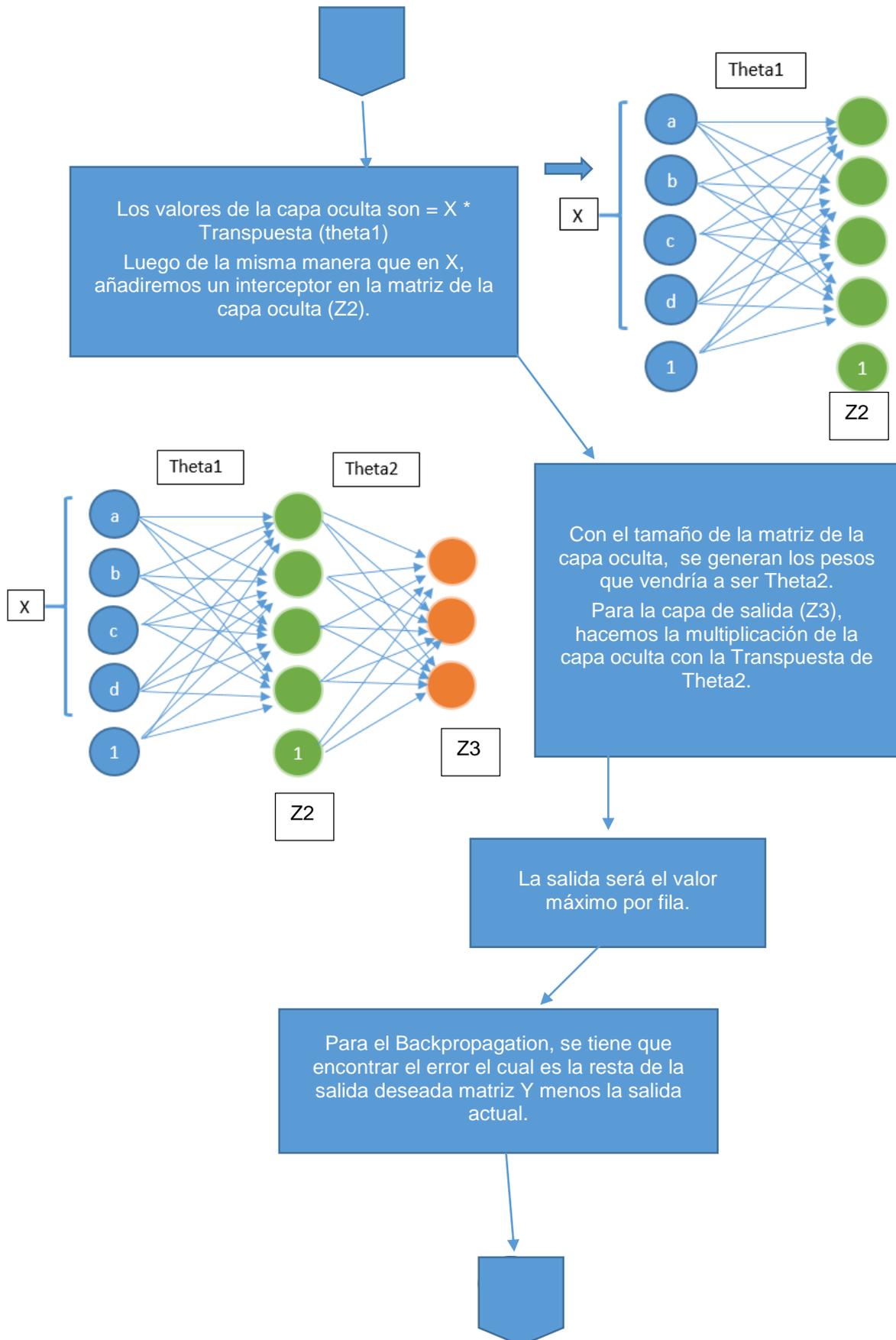
- CON EL VIDEOJUEGO

Tareas / Docentes	Esquematizar y delimitar el tema	Esquematizar y realizar las preguntas	RNA selecciona preguntas en tiempo real	Hacer el formato del examen	Impresiones y otros tiempos	Corregir exámenes	Pasar notas al registro
1	20	120	0.01	0	0	0	0
2	30	50	0.01	0	0	0	0
3	20	75	0.01	0	0	0	0
4	25	80	0.01	0	0	0	0
5	20	90	0.01	0	0	0	0

ANEXO N°6

Diagrama de Flujo de la Red Neuronal Artificial (Entrenamiento).





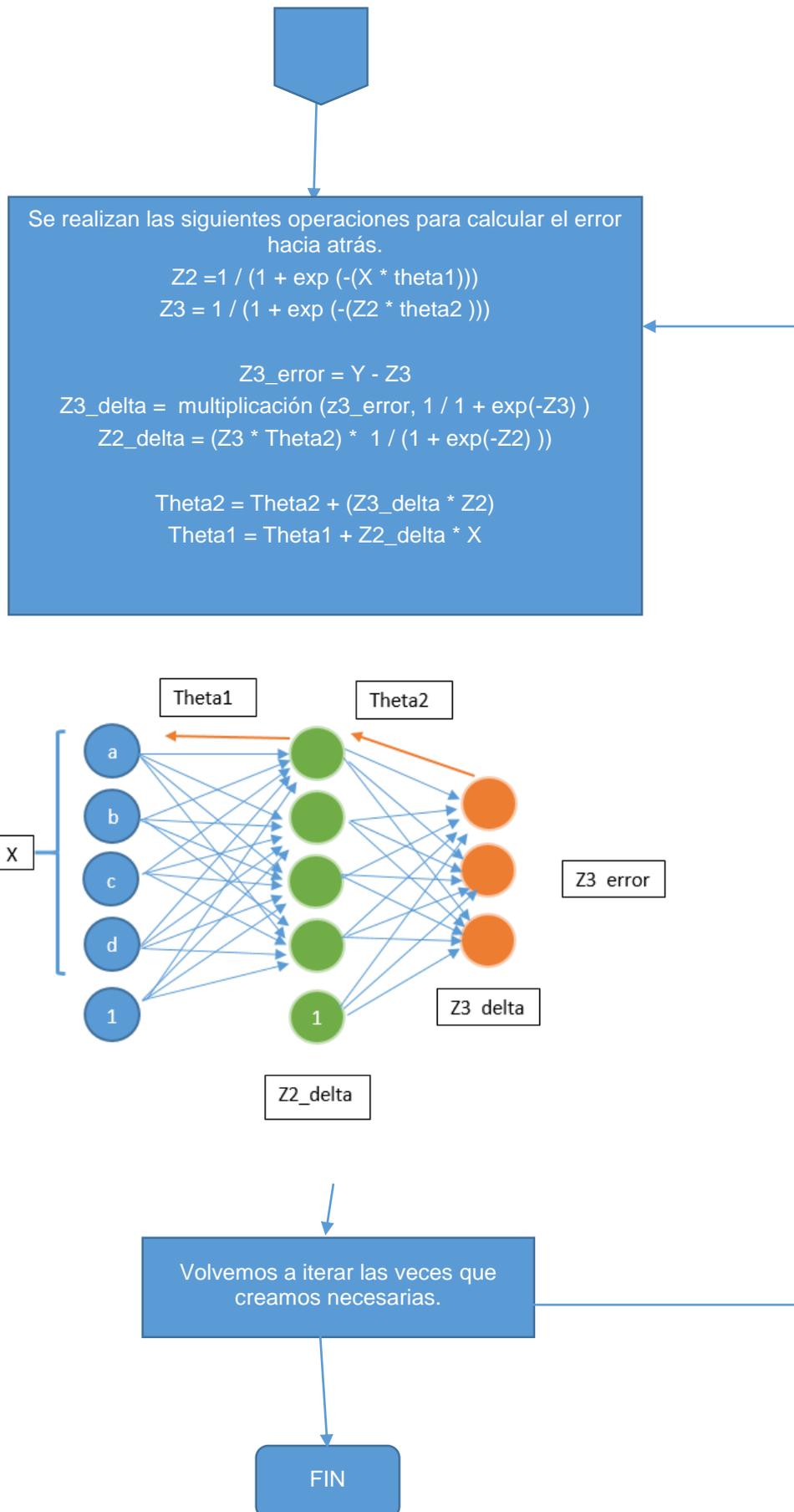


Diagrama de Flujo de la Red Neuronal Artificial (Salidas)

