

SEP

TNM

INSTITUTO TECNOLÓGICO DE CULIACÁN



Una Herramienta Inteligente y Gamificada para el Desarrollo del
Pensamiento Computacional a través del Reconocimiento de
Patrones

TESIS

PRESENTADA ANTE EL DEPARTAMENTO ACADÉMICO DE ESTUDIOS DE POSGRADO
DEL INSTITUTO TECNOLÓGICO DE CULIACÁN EN CUMPLIMIENTO PARCIAL DE LOS
REQUISITOS PARA OBTENER EL GRADO DE

MAESTRO EN CIENCIAS DE LA COMPUTACIÓN

POR:

Jorge Abraham Romero Polo
INGENIERO DE SOFTWARE

DIRECTOR DE TESIS:

Dra. María Lucía Barrón Estrada

CULIACÁN, SINALOA

SEPTIEMBRE 2019



"2019, Año del Caudillo del Sur, Emiliano Zapata"

Culiacán, Sin., 7 de Agosto del 2019

DIVISIÓN DE ESTUDIOS DE POSGRADO E INVESTIGACIÓN

OFICIO: DEPI: 331/VIII/2019

ASUNTO: **Autorización Impresión**

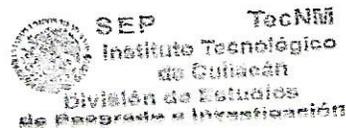
ING. JORGE ABRAHAM ROMERO POLO
ESTUDIANTE DE LA MAESTRÍA EN CIENCIAS DE LA COMPUTACIÓN
PRESENTE.

Por medio de la presente y en virtud de que ha completado los requisitos para el examen de grado de la **Maestría en Ciencias de la Computación**, se concede autorización para la impresión de la tesis titulada: **"UNA HERRAMIENTA INTELIGENTE Y GAMIFICADA PARA EL DESARROLLO DEL PENSAMIENTO COMPUTACIONAL A TRAVÉS DEL RECONOCIMIENTO DE PATRONES"** bajo la dirección del(a) **Dra. María Lucía Barrón Estrada**

Sin otro particular reciba un cordial saludo.

ATENTAMENTE
Excelencia en Educación Tecnológica®


M.C. MARÍA ARACELY MARTÍNEZ AMAYA
JEFE(A) DE LA DIVISIÓN DE ESTUDIOS DE
POSGRADO E INVESTIGACIÓN



C.c.p. archivo

MAMA/lucy *





**“UNA HERRAMIENTA INTELIGENTE Y
GAMIFICADA PARA EL DESARROLLO DEL
PENSAMIENTO COMPUTACIONAL A TRAVÉS DEL
RECONOCIMIENTO DE PATRONES”**

Tesis presentada por:

ING. JORGE ABRAHAM ROMERO POLO

Aprobada en contenido y estilo por:

Dra. María Lucía Barrón Estrada
Director de Tesis

Dr. Ramón Zatarain Cabada
Secretario

M.C. Gloria Ekaterine Peralta Peñuñuri
Vocal -1

M.C. Rosalío Zatarain Cabada
Vocal -2

M.C. María Aracely Martínez Amaya
Jefe(a) de la División de Estudios de
Posgrado e Investigación



Dedicatoria

A mi familia más cercana: mi prometida, mi madre, mi hermano y mi abuela.

Agradecimientos

Al Conacyt, por otorgar los recursos económicos para realizar este posgrado de tiempo completo, incluyendo mi estancia de investigación en Ciudad de México.

A mi madre Alma Lilia y mi abuela Dionisia Ofelia, porque con sus oraciones y su incondicional amor me mantuvieron fuerte.

A mi novia, mi futura esposa, quien siempre me motivó, me acompañó y me apoyó en esos momentos difíciles. Te amo.

A mi asesora, la Dra. María Lucía Barrón Estrada, encargada de redireccionarme por el camino adecuado e incentivarme finalizar con éxito este trabajo.

A todos los que intercedieron en mi durante mi estancia en la maestría; mis profesores, Ricardo, Ramón, Héctor, quienes enriquecieron mis conocimientos; mis compañeros y amigos, Héctor, Brandon, Francisco, Ramón y Emmanuel, compartimos mucho más que lugares en este posgrado, y a Ekaterine y Lucy, por su oportuna administración para iniciar y finalizar todo este proceso.

A mi amiga, la Lic. Laura Armenta, que me facilitó la ejecución del primer experimento con su grupo de quinto año.

Declaración de autenticidad

Por la presente declaro que, salvo cuando se haga referencia específica al trabajo de otras personas, el contenido de esta tesis es original y no se ha presentado total o parcialmente para su consideración para cualquier otro título o grado en esta o cualquier otra universidad. Esta tesis es resultado de mi propio trabajo y no incluye nada que sea resultado de algún trabajo realizado en colaboración, salvo que se indique específicamente en el texto.

Jorge Abraham Romero Polo.

Culiacán, Sinaloa, México, 2019

Resumen

El Pensamiento Computacional (PC) es el proceso mental para la formulación de un problema y su solución. Son competencias que se aplican a cualquier disciplina, se desarrollan con o sin un equipo de cómputo y se consideran tan elementales como leer y escribir.

Actualmente, el sistema de educación en México no incluye específicamente el PC en los programas de estudio de educación básica, sin embargo, algunas habilidades relacionadas con el PC, que los estudiantes deben adquirir, se abordan en diferentes temas y áreas de estudio mediante actividades de diversos tipos. Los profesores utilizan el método tradicional de enseñanza con explicaciones grupales y los estudiantes resuelven una limitada cantidad de ejercicios proporcionada en los libros de texto; por lo tanto, existe una ausencia de seguimiento personalizado sobre el aprendizaje de los estudiantes.

En esta tesis se presenta una propuesta en forma de aplicación móvil, bautizada como *Patrony*, que busca promover el desarrollo del PC a través de actividades diseñadas para reconocer patrones. El reconocimiento de patrones es una sección del PC que involucra encontrar similitudes o patrones en problemas pequeños con el fin de resolver problemas más complejos.

Patrony funge como una solución para impulsar en general la educación hacia el PC y en particular el mejorar el reconocimiento de patrones de sus usuarios. Las actividades incluidas en *Patrony* fueron basadas en temas incluidos en el contenido curricular mexicano correspondiente al tercer periodo escolar, es decir, de cuarto a sexto grado, y pertenecen al libro de desafíos matemáticos.

Para evaluar la efectividad de la aplicación móvil se utilizó la técnica pre-test/post-test en dos escuelas primarias a un total de 34 de estudiantes cuyos resultados fueron analizados, demostrando una primera buena valoración con los estudiantes del grupo experimental que presentaron mejores resultados que los del grupo de control.

Conocer y desarrollar las competencias del PC mediante la práctica formará parte de importantes herramientas para la resolución de problemas de este siglo.

Palabras clave

Pensamiento Computacional

Reconocimiento de Patrones

Gamificación

Educación con TIC

Desarrollo Móvil Multiplataforma

Índice general

1. Introducción.....	1
1.1. Aplicaciones móviles en la educación.....	1
1.2. Planteamiento del problema	2
1.3. Objetivos	3
1.3.1. Objetivo general.....	3
1.3.2. Objetivos específicos.....	3
1.4. Hipótesis.....	3
1.5. Justificación.....	4
1.6. Organización del documento.....	4
2. Marco teórico.....	5
2.1. Educación con tecnologías de la información y comunicación.....	5
2.1.1. Aprendizaje electrónico (<i>e-learning</i>)	5
2.1.2. Aprendizaje móvil (<i>m-learning</i>).....	6
2.1.3. Aprendizaje mezclado (<i>b-learning</i>).....	6
2.2. Desarrollo móvil multiplataforma	6
2.3. Gamificación	7
2.4. Pensamiento computacional.....	8
2.4.1. Reconocimiento de patrones	12
2.4.2. El pensamiento computacional en la educación.....	13
2.5. Ingeniería de software	14
2.5.1. Modelo de mejora iterativa.....	15
2.5.2. Modelo de prototipado	15
3. Estado del arte.....	16
3.1. El pensamiento computacional en la educación.....	16
3.2. Aplicaciones móviles educativas	16
3.2.1. Edumóvil.....	17
3.2.2. ARGeo.....	17
3.3. Aplicaciones orientadas al pensamiento computacional	18
3.3.1. <i>Prototipo para edades tempranas</i>	18
3.3.2. EasyLogic.....	19
3.4. Videojuegos con patrones	20
3.4.1. Candy Crush Saga.....	20

3.4.2.	4 fotos 1 palabra	22
3.5.	Comparación de aplicaciones	23
4.	Desarrollo del proyecto	24
4.1.	Propuesta de solución al problema	24
4.2.	Proceso de desarrollo de software	32
4.3.	Análisis.....	34
4.3.1.	Análisis de requerimientos	34
4.3.2.	Actores	37
4.3.3.	Diagrama de contexto.....	38
4.3.4.	Arquetipos	39
4.4.	Diseño	40
4.4.1.	Arquitectura.....	40
4.4.2.	Modelo de datos	43
4.4.3.	Interfaz gráfica	45
4.5.	Implementación.....	48
4.5.1.	React native.....	48
4.5.2.	Componentes react	48
4.5.3.	Persistencia con async storage	49
4.5.4.	React navigation.....	50
4.6.	Depuración y pruebas.....	51
4.6.1.	Pruebas	52
4.6.2.	Dificultades	53
5.	Experimentos y resultados.....	54
5.1.	Intervención con estudiantes	54
5.2.	Análisis de resultados.....	55
5.2.1.	Resultados de pre-test.....	55
5.2.2.	Resultados de post-test	57
5.2.3.	Resultados comparativos pre-test y post-test	58
5.2.4.	Ganancia relativa.....	59
5.2.5.	Datos recopilados por <i>Patrony</i>	60
5.3.	Conclusiones	62
6.	Conclusiones y trabajo futuro.....	64
6.1.	Conclusiones	64
6.2.	Trabajo futuro.....	65
	Bibliografía	66

Anexos	71
A. Examen Pre-Test	A-1
B. Examen Post-Test.....	B-1

Índice de figuras

Figura 2-1. Los pilares del pensamiento computacional.....	11
Figura 2-2. Modelo de mejora iterativa (Jalote, 2008).	15
Figura 2-3. Modelo de prototipado (Jalote, 2008).....	15
Figura 3-1. Actividad en ARGeo.....	18
Figura 3-2. Prototipo de app de reconocimiento de patrones para edades tempranas.	19
Figura 3-3. Ejercicio de estructura repetitiva en EasyLogic.	20
Figura 3-4. Formando un dulce especial en Candy Crush.	21
Figura 3-5. Nivel 217 en Candy Crush, del tipo "bajar frutas".	21
Figura 3-6. Mapa de niveles de Candy Crush.	21
Figura 3-7. Nivel 10 de 4 fotos 1 palabra.	22
Figura 3-8. Nivel resuelto de 4 fotos 1 palabra.	22
Figura 4-1. Primera parte de los ejercicios del libro de 4°.	25
Figura 4-2. Segunda parte de los ejercicios del libro de 4°.	26
Figura 4-3. Patrones numéricos con fracciones en libro de 5°.	27
Figura 4-4. Patrones numéricos con enteros en el libro de 5°.	28
Figura 4-5. Patrones de figuras geométricas en el libro de 5°.	29
Figura 4-6. Primera parte de los ejercicios de patrones en el libro de 6°.	30
Figura 4-7. Segunda parte de los ejercicios de patrones en el libro de 6°.	31
Figura 4-8. Proceso de desarrollo Prototipado-Iterativo.	33
Figura 4-9. Diagrama de contexto del sistema.....	38
Figura 4-10. Diagrama de arquetipos.....	39
Figura 4-11. Arquitectura del sistema de reconocimiento de patrones <i>Patrony</i>	41
Figura 4-12. Simbología de draw.io.....	44
Figura 4-13. Modelo de datos del sistema de reconocimiento de patrones.	44
Figura 4-14. Estructura de una actividad tipo secuencia.	45
Figura 4-15. Estructura de una actividad tipo matemáticas.	45
Figura 4-16. Mockup de la interfaz con las secciones principales.	46
Figura 4-17. Mockup de actividad general.....	46
Figura 4-18. Mockup de una actividad de tipo secuencia.....	47
Figura 4-19. Ejemplo de separación de componentes en un mockup.....	48
Figura 4-20. Diagrama del modelo de datos de implementación.....	49
Figura 4-21. <i>Patrony</i> ejecutándose en un Moto G5 Plus.	51
Figura 4-22. <i>Patrony</i> ejecutándose en una Galaxy Tab S2.	51
Figura 4-23. <i>Patrony</i> ejecutándose en un iPad Air 2.....	52
Figura 5-1. Procedimiento de evaluación pre-test/post-test.....	54
Figura 5-2. Histograma de calificaciones del examen pre-test.	56
Figura 5-3. Estudiantes aprobados y reprobados del examen pre-test.....	56
Figura 5-4. Histograma de calificaciones del examen post-test.	57
Figura 5-5. Estudiantes aprobados y reprobados del examen post-test.	58
Figura 5-6. Relación de calificaciones pre-test y post-test en grupos de control.	58
Figura 5-7. Relación de calificaciones pre-test y post-test en grupos experimentales.....	59

Figura 5-8. Comparación de ganancia relativa de los estudiantes.....	60
Figura 5-9. Puntuaciones promedio por actividad de <i>Patrony</i>	61
Figura 5-10. Errores promedio por actividad de <i>Patrony</i>	61
Figura 5-11. Tiempo promedio por actividad en <i>Patrony</i>	62

Índice de tablas

Tabla 2-1. Terminología de elementos de juego.....	8
Tabla 3-1. Tabla comparativa del estado del arte.....	23
Tabla 4-1. Requisitos funcionales para el sistema de reconocimiento de patrones.....	36
Tabla 4-2. Requisitos de calidad para el sistema de reconocimiento de patrones.....	37
Tabla 4-3. Restricciones para el sistema de reconocimiento de patrones.	37
Tabla 4-4. Arquetipos del sistema de reconocimiento de patrones.	39
Tabla 4-5. Componentes del grupo Navegadores.....	42
Tabla 4-6. Componentes del grupo Contenedores.	42
Tabla 4-7. Componentes del grupo Componentes.	43
Tabla 4-8. Entidades de Datos.....	50
Tabla 4-9. Relación de comentarios y acciones tomadas.	52

Capítulo 1

1. Introducción

Es indudable que los dispositivos móviles como los teléfonos inteligentes (*smartphones*) y tabletas (*tablets*) se han incluido en la vida cotidiana de la sociedad y son inherentes a las nuevas generaciones. Este estilo de vida actual debe ser aprovechado en todos los sentidos posibles. Si existe una relación cotidiana con los dispositivos, incrustar la educación en ellos directa o indirectamente daría pie a una generación con alcances educativos superiores.

1.1. Aplicaciones móviles en la educación

El uso de dispositivos móviles en la educación amplía las opciones de enseñanza cambiando el paradigma habitual del uso de libros. Estos dispositivos ya se encuentran arraigados a la sociedad como una herramienta de comunicación, de creación, de entretenimiento y de educación. En México, el Instituto Nacional de Estadística, Geografía e Informática (INEGI, 2018) reportó que, en 2017, el 72.2% de la población de seis o más años utiliza de alguna forma los teléfonos celulares, es decir, al menos siete de cada diez personas disponen de un teléfono celular inteligente.

Un ejemplo de implementación de tecnología móvil en el área de la educación en México fue “*Mati-Tec: Educación para Todos*”, realizado por el Tecnológico de Monterrey en colaboración con la Secretaría de Educación Pública (SEP), quien otorgó permisos de aplicación; la Fundación Telefónica México, quien aportó capital, el servicio 3G, y los dispositivos móviles para el desarrollo del proyecto, y el Consejo Nacional de Ciencia y Tecnología (Conacyt), quien brindó apoyo a capital (Instituto Tecnológico de Monterrey, 2016). El objetivo de *Mati-Tec* fue cambiar el proceso de enseñanza-aprendizaje a través de recursos pedagógicos innovadores y dispositivos móviles. Con su aplicación, *Mati-Tec* logró beneficiar a más de 6,000 niños.

1.2. Planteamiento del problema

El Pensamiento Computacional es el proceso mental para la formulación de un problema y su solución. Son competencias que se aplican a cualquier disciplina, se desarrollan con o sin un equipo de cómputo y se consideran tan elementales como leer o escribir. Estas competencias se practican en temas de todos los niveles en la educación de México, aunque el Pensamiento Computacional no se incluya como un tema específico en el esquema curricular.

El Pensamiento Computacional comprende varios temas, en este trabajo se da enfoque a uno en específico: el Reconocimiento de Patrones. Dicho tema supone encontrar similitudes o patrones en los problemas pequeños que han sido descompuestos con el fin de resolver problemas más complejos.

Para establecer claramente el problema, se realizó una exploración de los libros de texto gratuitos de educación básica (primaria) correspondientes a los grados cuarto, quinto y sexto. La búsqueda se enfocó en el contenido relacionado con la conceptualización y uso del reconocimiento de patrones. Adicionalmente, se realizó una entrevista abierta con una docente de quinto año de primaria, con el fin de obtener información sobre los métodos de enseñanza utilizados para el tema de reconocimiento de patrones.

Como producto de estas acciones, se obtuvo una descripción de los métodos de enseñanza usados por el docente sobre el tema de reconocimiento de patrones y una lista de las actividades para reconocer patrones incluidas en los libros de texto de educación primaria, específicamente en cuarto, quinto y sexto grado. Estas actividades involucran sucesiones de figuras, sucesiones numéricas y reglas gramaticales, entre otras.

Estos productos sirvieron para definir el problema a resolver: el método de enseñanza tradicional utilizado por los docentes de nivel básico está limitado a exposiciones y explicaciones grupales, lo que no permite hacer un seguimiento personalizado del aprendizaje de los estudiantes. Además, la cantidad de ejercicios contenidos en los libros de texto es escasa y algunos estudiantes no logran adquirir la habilidad de reconocer patrones en el tiempo planeado para el aprendizaje de los temas. Por otra parte, se detectó también

que, dentro de los grupos, una porción de estudiantes no están motivados a realizar los ejercicios y actividades de la bibliografía debido a lo rutinario del proceso de aprendizaje.

1.3. Objetivos

En este apartado se presentan el objetivo general y los objetivos específicos que fueron establecidos para este tema de investigación.

1.3.1. Objetivo general

Desarrollar y evaluar la efectividad de una aplicación móvil que implemente un juego educativo que promueva y motive el desarrollo del pensamiento computacional en los niños de 8 a 10 años, a través del reconocimiento de patrones.

1.3.2. Objetivos específicos

Con el fin de lograr el objetivo general, se establecieron cinco objetivos específicos que se presentan a continuación:

1. Definir y seleccionar actividades y ejercicios para el reconocimiento de patrones.
2. Desarrollar el prototipo del software (aplicación móvil multiplataforma) que promueva el desarrollo del reconocimiento de patrones.
3. Seleccionar e implementar el mecanismo de gamificación que contribuya a la motivación de los niños.
4. Realizar la intervención con los estudiantes para evaluar la efectividad del sistema respecto al tema reconocimiento de patrones y la motivación de los estudiantes.
5. Analizar los resultados de la intervención para generar conclusiones sobre la efectividad del sistema.

1.4. Hipótesis

Los estudiantes que usen la aplicación móvil con el juego educativo desarrollarán de manera más efectiva el reconocimiento de patrones que aquellos estudiantes que utilicen los medios tradicionales.

1.5. Justificación

El pensamiento computacional está formado por un conjunto de habilidades empleadas en la formulación y resolución de problemas, sin estar atada a ninguna disciplina en particular. El poder que conlleva adquirir y desarrollar este conocimiento está destinado a mejorar la calidad de vida de la sociedad.

Científicos, empresas y gobiernos en varios países se encuentran investigando, apoyando y aplicando el pensamiento computacional en niños y jóvenes principalmente a través de las ciencias de la computación en forma de talleres, recursos virtuales abiertos al público, e incluso anexándolo en su currículo.

Por otra parte, el uso de la tecnología resulta atractivo para los niños por la sensación de independencia y entretenimiento a través de retroalimentación visual y auditiva.

El tener una herramienta adicional que aproveche la ventaja tecnológica y las mecánicas y dinámicas de los videojuegos como acelerador de motivación y diversión en el aprendizaje ayudará a desarrollar la habilidad de reconocer patrones en los estudiantes.

1.6. Organización del documento

La tesis está dividida en seis capítulos, el objetivo de los restantes se describe a continuación:

Capítulo 2. Este capítulo presenta el marco teórico de la tesis, los temas principales necesarios para comprender el trabajo realizado.

Capítulo 3. Este capítulo contiene el estado del arte, lo que ya existe respecto al tema abordado.

Capítulo 4. Este capítulo describe el proceso de desarrollo del sistema de software producido y las tecnologías requeridas para su implementación.

Capítulo 5. Este capítulo exhibe los resultados de la evaluación de efectividad del software desarrollado.

Capítulo 6. Este capítulo expone las conclusiones alcanzadas y propone trabajo futuro para esta tesis.

Capítulo 2

2. Marco teórico

En este capítulo se incluye la base teórica de esta investigación. El capítulo se divide en cinco secciones: Educación con tecnologías de la información y comunicación, Desarrollo móvil multiplataforma, Gamificación, Pensamiento computacional y por último Ingeniería de software. Dentro del apartado destinado al Pensamiento Computacional se abordará profundamente el tema del Reconocimiento de Patrones, ya que es el tema principal de este trabajo de investigación.

2.1. Educación con tecnologías de la información y comunicación

El uso de la tecnología computacional en la educación generó algunos conceptos como aprendizaje electrónico (*e-learning*), aprendizaje móvil (*m-learning*) y aprendizaje mezclado (*b-learning*). Estos conceptos involucran elementos comunes, todos están enfocados al uso de dispositivos electrónicos y sistemas de software durante el proceso de enseñanza-aprendizaje, sin embargo, existen diferencias entre ellos que deben ser clarificadas.

2.1.1. Aprendizaje electrónico (*e-learning*)

E-learning ya es un término bastante usado en estos tiempos y se hace referencia a él cuando se adquiere conocimiento a través de cualquier dispositivo electrónico, con o sin internet.

Clark y Mayer establecen que: “El *e-learning* es una instrucción impartida en un dispositivo digital (como una computadora de escritorio, laptop, tableta o teléfono inteligente) que tiene como objetivo apoyar el aprendizaje” (Clark & Mayer, 2016).

Se ha demostrado que con un sistema de educación *e-learning* el estudiante mantiene una actualización constante de información, mejora sus habilidades lectoras y se vuelve más crítico de la información que maneja, según el trabajo de (Suárez-Suárez & Custodio-Najar, 2014).

2.1.2. Aprendizaje móvil (*m-learning*)

El aprendizaje móvil, comúnmente conocido como *m-learning*, es un concepto relativamente reciente asociado a la educación mediada por dispositivos móviles. Este tipo de aprendizaje se materializó cuando la tecnología necesaria (dispositivos móviles, internet, redes inalámbricas, etc.) estuvo disponible para que cualquier persona desde cualquier lugar y a cualquier hora pudiera aprender utilizando un dispositivo móvil como una tableta electrónica, una laptop o un teléfono inteligente.

En 2013, (Crompton, 2013) definió el *m-learning* como “aprender a través de múltiples contextos, a través de interacciones sociales y de contenido, utilizando dispositivos electrónicos personales”. También, aclaró que la palabra contexto describe situaciones que pueden o no ser formales, planeadas o adecuadas pero que comparten un fin común: aprender.

2.1.3. Aprendizaje mezclado (*b-learning*)

El aprendizaje mezclado (*b-learning* por blended learning) se ha moldeado a lo largo de los años a causa del avance tecnológico y la manera en que la tecnología se ha aplicado en la educación. Conocido también como aprendizaje híbrido (Bryan & Volchenkova, 2016), es un método de enseñanza que combina las técnicas presenciales más efectivas y la colaboración interactiva en línea, constituyendo un sistema que funciona en una constante correlación y formando un todo (Krasnova, 2015).

El *b-learning* es una técnica de enseñanza muy ordinaria, y su existencia se puede reconocer con la simple aplicación de una plataforma de gestión de aprendizaje para compartir información de una asignatura, así como la entrega de tareas digitales o la aplicación de exámenes digitales, actividades que son complementarias a las clases presenciales tradicionales.

2.2. Desarrollo móvil multiplataforma

En un mundo donde la presencia del software es indispensable en la mayor parte de los dispositivos electrónicos, surgen los marcos de trabajo (*framework*) multiplataforma, que permiten compartir una misma base de código para generar aplicaciones dirigidas a distintos sistemas.

Grandes compañías en la industria de software han desarrollado algunos *frameworks* multiplataforma para propiciar el desarrollo móvil, entre los que se encuentran: Xamarin, de Microsoft; Flutter, de Google y React Native, de Facebook.

React Native es un *framework* multiplataforma para desarrollar, con el lenguaje JavaScript, aplicaciones móviles principalmente para los dos sistemas operativos dominantes de los dispositivos actuales: iOS y Android (Facebook Inc., 2019b). En React Native las interfaces gráficas se construyen bajo los mismos lineamientos de la popular librería para interfaces de usuario web React, realizada por la compañía Facebook.

2.3. Gamificación

La gamificación pudiera ser considerada como el arte de incorporar en una tarea aburrida algunos conceptos y mecánicas de la industria de los videojuegos que motiven a una persona a realizar dicha tarea.

En general, el diseño de un videojuego envuelve mecánicas, capacidades disponibles para el jugador; dinámicas, fenómenos que genera el jugador al aplicar las mecánicas, y estéticas, lo que el juego quiere transmitir en el jugador.

De manera más formal en (Hunicke, Leblanc, & Zubek, 2004) estos conceptos se definen como sigue:

- **Mecánicas:** componentes particulares del juego, a nivel de representación de datos y algoritmos.
- **Dinámicas:** describen el comportamiento en tiempo de ejecución de las mecánicas que actúan en las entradas del jugador y en las salidas de cada una de ellas a lo largo del tiempo.
- **Estéticas:** describen las respuestas emocionales deseables evocadas en el jugador, cuando interactúa con el sistema de juego.

La gamificación puede ser empleada en el ámbito educativo, laboral y en la vida misma. De acuerdo con Wang: “*La gamificación describe una serie de principios, procesos y sistemas*

de diseño utilizados para influir, involucrar y motivar a individuos, grupos y comunidades para impulsar conductas y lograr los resultados deseados” (Wang, 2011).

En (Seaborn & Fels, 2015) se enfatiza que existen términos en el ámbito de juego que pueden emplearse en distintas situaciones y se explican en la Tabla 2-1 para evitar confusiones.

Tabla 2-1. Terminología de elementos de juego.
Esta tabla fue obtenida de (Seaborn & Fels, 2015).

Término	Definición	Alternativas
Puntos	Unidades numéricas indicando el progreso.	Puntos de experiencia; puntuación.
Insignias	Íconos visuales que significan logros.	Trofeos.
Tablas de clasificación	Muestra de rangos para comparaciones.	Clasificaciones, tablas de puntuación.
Progresión	Hitos indicando el progreso.	Nivelación, subir de nivel.
Estado	Apodos textuales indicando el progreso.	Títulos, rangos.
Niveles	Ambientes con dificultad creciente.	Etapas, áreas, mundos.
Recompensas	Objetos deseables y tangibles.	Incentivos, premios, regalos.
Role	Elementos de rol del personaje.	Clase, personaje.

Según (Chou, 2014) la gamificación “es el arte de derivar elementos divertidos y atractivos que se encuentran típicamente en los juegos y aplicarlos cuidadosamente a actividades productivas o del mundo real”.

2.4. Pensamiento computacional

Jeanette Wing en 2006 se refirió al Pensamiento Computacional (PC) como una habilidad fundamental para todos, una habilidad que involucra resolución de problemas, diseño de

sistemas y entendimiento del comportamiento humano y no una habilidad solo para científicos de la computación (Wing, 2006).

A partir de la publicación de Wing en 2006, empezó un movimiento científico, empresarial y gubernamental que crece año con año. Los esfuerzos buscan incrustar el Pensamiento Computacional a la sociedad actual y reconocerlo como un conocimiento básico, haciendo hincapié en lo indispensable que será en un futuro próximo. El Pensamiento Computacional representa la herramienta principal para los retos y problemas que el siglo XXI depara.

Dentro del movimiento científico los investigadores han revisado en múltiples ocasiones la definición de Pensamiento Computacional y en la actualidad aún no hay un consenso que establezca una definición oficial. Sin embargo, a continuación, se presentan dos definiciones que se consideraron importantes en este trabajo.

“El pensamiento computacional es un tipo de pensamiento analítico. Comparte con el pensamiento matemático las formas generales en las que podríamos abordar la resolución de un problema. Comparte con el pensamiento ingenieril las formas generales en que podríamos enfocar el diseño y la evaluación de un sistema grande y complejo que opera dentro de las limitaciones del mundo real. Comparte con el pensamiento científico las formas generales en las que podríamos abordar la comprensión de la computabilidad, la inteligencia, la mente y el comportamiento humano”
(Wing, 2008).

Más tarde, en 2014, Wing decidió ser más descriptiva respecto a su antigua definición que más bien era una abreviación de “pensar como un científico de la computación”, actualizándola con ayuda de sus colegas:

“El pensamiento computacional es el proceso de pensamiento involucrado en la formulación de un problema y en la expresión de su(s) solución(es) de tal manera que un computador – humano o máquina – pueda llevar a cabo de manera efectiva” (Wing, 2014).

El Pensamiento Computacional describe la actividad mental que se requiere tanto para formular un problema como para crear su solución computable. Es importante tomar en

consideración que esta solución puede ser empleada tanto por un humano como por una computadora.

Saber qué es y desarrollar el pensamiento computacional añade ventajas en la vida de las personas. En (Wing, 2010) se exponen las capacidades que todos pueden adquirir mediante el PC:

- Entender qué aspectos de un problema son susceptibles a ser computados.
- Evaluar la correspondencia entre las herramientas y técnicas computacionales y un problema.
- Entender las limitaciones y el poder de las herramientas y técnicas computacionales.
- Aplicar o adaptar una herramienta o técnica computacional a un nuevo uso.
- Reconocer una oportunidad del uso de la computación de una nueva manera.
- Aplicar estrategias computacionales como *divide y vencerás* en cualquier dominio.

Además, existen actitudes que son esenciales del PC y se desarrollan al mismo tiempo que mejoran las competencias de la persona, estas incluyen, según (ISTE & CSTA, 2011):

- Confianza al lidiar con complejidad.
- Persistencia al trabajar con problemas difíciles.
- Tolerancia a la ambigüedad.
- Habilidad para lidiar con problemas no estructurados.
- Habilidad para comunicarse y trabajar con otros para alcanzar una meta o solución común.

El PC tiene cuatro pilares de habilidades (Figura 2-1): descomposición, abstracción, reconocimiento de patrones y el diseño del algoritmo (BBC, 2019; Google LLC, 2019).

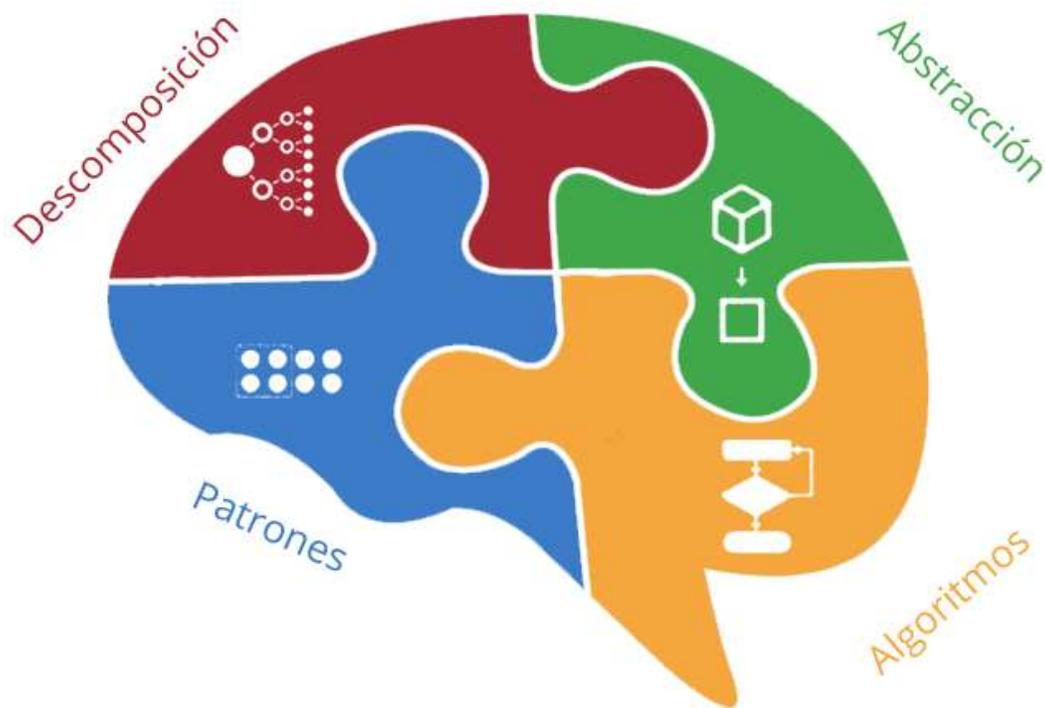


Figura 2-1. Los pilares del pensamiento computacional.

La **descomposición** involucra romper un problema o sistema complejo en partes pequeñas, manejables y fáciles de entender (BBC, 2019).

La **abstracción** hace referencia a la competencia que permite a una persona lidiar con la complejidad al identificar las partes esenciales del objeto o fenómeno que se analiza (Zapotecatl, 2018).

El **diseño de algoritmos** consiste en crear una serie ordenada de instrucciones para resolver un problema (Grover & Pea, 2013).

El **reconocimiento de patrones** se aborda en 2.4.1 de manera más minuciosa por ser el tema principal de esta tesis.

En (Hsu, Chang, & Hung, 2018) se presenta una clasificación muy detallada del pensamiento computacional a través de 19 pasos de pensamiento, entre los que se incluyen:

1. Abstracción.
2. Diseño de algoritmos.
3. Automatización.

4. Análisis de datos.
5. Recolección de datos.
6. Representación de datos.
7. Descomposición.
8. Paralelización.
9. Generalización de patrones.
10. Reconocimiento de patrones.
11. Simulación.
12. Transformación.
13. Lógica condicional.
14. Conexión a otros campos.
15. Visualización.
16. Detección de errores y depuración.
17. Eficiencia y desempeño.
18. Modelado.
19. Resolución de problemas.

2.4.1. Reconocimiento de patrones

No existe una definición única y universalmente admisible para el Reconocimiento de Patrones (RP). En esta sección se presentan diversas definiciones acuñadas por varios autores, las cuales serán usadas para generar la base para el desarrollo del sistema producto de esta tesis.

Para Ruiz, el reconocimiento de patrones es una zona de conocimiento que utiliza diversos elementos para identificar o clasificar objetos:

“Por RP se podría identificar a la zona del conocimiento (de carácter interdisciplinario) que se ocupa del desarrollo de teorías, métodos, técnicas, y dispositivos computacionales para la realización de procesos ingenieriles, computacionales y/o matemáticos, relacionados con objetos físicos y/o abstractos, que tienen el propósito de extraer la información que le permita establecer propiedades y/o vínculos de o entre conjuntos de

dichos objetos sobre la base de los cuales se realiza una tarea de identificación o clasificación” (Ruiz, 2013).

El mismo concepto se aborda en la Enciclopedia Británica que lo define como sigue:

“La imposición de la identidad de los datos de entrada (habla, imágenes, texto) mediante el reconocimiento y la delineación de los patrones que contengan y sus relaciones. Las etapas en el RP pueden incluir medición del objeto para identificar atributos distintivos, la extracción de características para los atributos definitorios y la comparación con patrones conocidos para determinar una coincidencia o no coincidencia” (Encyclopaedia Britannica, 2018).

Por otra parte, en el recurso de Introducción al Pensamiento Computacional de BBC Bitesize se establece que el Reconocimiento de Patrones involucra encontrar similitudes o patrones en los problemas pequeños producto de una descomposición previa, con el fin de resolver eficientemente problemas más complejos (BBC, 2019).

Una definición más concreta la aporta Zapotecatl estableciendo que “El área de reconocimiento de patrones concentra algoritmos, metodologías, teorías y sistemas que permiten diferenciar, asociar y clasificar datos de forma automatizada” (Zapotecatl, 2018).

Actualmente el Reconocimiento de Patrones se asocia directamente con la Ciencia de Datos, donde el poder de cómputo tiene un papel fundamental en la detección de patrones.

2.4.2. El pensamiento computacional en la educación

Muchos profesores han abordado la enseñanza del Pensamiento Computacional a través de la programación argumentando que es la manera más fácil y apropiada de hacerlo. Sin embargo, se ha demostrado que el pensamiento computacional tiene amplio uso en áreas como Matemáticas, Biología, Ciencias de la computación y Lenguaje (Hsu et al., 2018).

De acuerdo con Hsu y colegas, es posible abordar diferentes estrategias de aprendizaje en las actividades de aprendizaje del Pensamiento Computacional, entre estas se encuentran:

1. Aprendizaje basado en problemas.
2. Aprendizaje colaborativo (trabajo en equipo).

3. Aprendizaje basado en proyectos.
4. Aprendizaje basado en juegos.
5. Demostraciones y acompañamiento (*Scaffolding*).
6. Sistema de resolución de problemas.
7. Narración de historias (*Storytelling*).
8. Estrategias computacionales sistemáticas.
9. Experiencia estética.
10. Aprendizaje basado en conceptos.
11. *HCI teaching*.
12. Aprendizaje basado en diseño.
13. Aprendizaje encarnado (*embodied*).
14. Lectura centrada en el profesor.
15. Literatura computacional crítica.
16. Diseño universal para aprendizaje.

En algunos países del mundo ya se ha incorporado el Pensamiento Computacional a los planes de estudio desde nivel preescolar hasta bachillerato (K-12), como Estados Unidos y Reino Unido y, en otros más, se ha integrado hasta o exclusivamente en el nivel profesional. Esta inclusión ha sido mediante diversas actividades que empiezan desde un nivel cognitivo bajo (como clasificación de botones de distintos colores y tamaños) hasta un intenso trabajo mental (como programación con ciclos), de forma gradual y semejante a las matemáticas, por mencionar una asignatura básica para la vida. De hecho, la abstracción se emplea en el álgebra, que es una asignatura considerada compleja por muchos alumnos, por lo que si un alumno tiene trayectoria en el Pensamiento Computacional trazada desde la infancia es muy probable que el álgebra no le resulte compleja después de todo.

2.5. Ingeniería de software

En todos los trabajos de desarrollo de software debe de existir un proceso de ingeniería de software adecuado al tipo de proyecto. Los modelos de prototipado y mejora iterativa contienen características aplicables al desarrollo del proyecto de tesis, dando como resultado un modelo híbrido descrito en la sección 4.2.

2.5.1. Modelo de mejora iterativa

En el modelo de mejora iterativa (ver Figura 2-2) se basa en una regla que indica que el software debe ser desarrollado en incrementos, cada incremento añade funcionalidad adicional al sistema hasta completarlo. Al iniciar el proyecto se consideran los requerimientos clave, que son fáciles de entender e implementar.

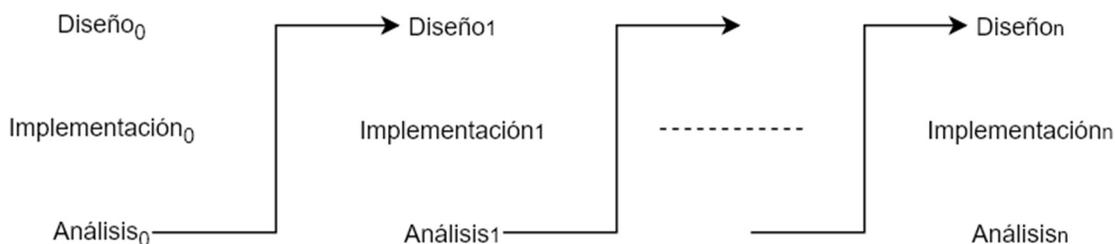


Figura 2-2. Modelo de mejora iterativa (Jalote, 2008).

2.5.2. Modelo de prototipado

La idea principal del modelo de prototipado es la creación de un prototipo desechable que ayude a entender los requerimientos. El desarrollo del prototipo no es riguroso ni formal.

El prototipo permite experimentar una sensación real del sistema deseado, lo que resulta en requerimientos que cambiarán con menos frecuencia y serán más estables (Jalote, 2008). Al tener requerimientos más refinados, se procede al proceso de desarrollo del sistema deseado. El proceso de este modelo está ilustrado en la Figura 2-3. El bloque más a la izquierda de la figura representa el desarrollo del prototipo desechable que establece las bases para entender los requerimientos y posteriormente desarrollar el sistema integrando requerimientos más estables en el software.

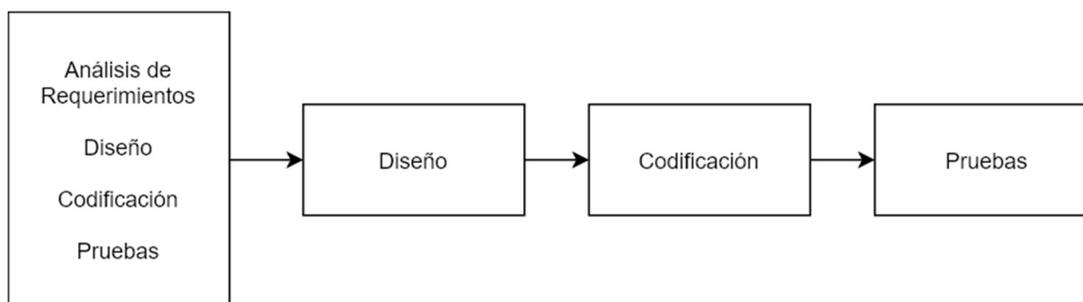


Figura 2-3. Modelo de prototipado (Jalote, 2008).

Capítulo 3

3. Estado del arte

A continuación, se agrupan los trabajos de investigación que se relacionan con las áreas de la educación, pensamiento computacional y aplicación tecnológica. Estos temas dieron forma al presente trabajo de tesis. El capítulo se encuentra seccionado en los temas siguientes: El pensamiento computacional en la educación, Aplicaciones móviles educativas, Aplicaciones orientadas al pensamiento computacional, Videojuegos con patrones y Comparación de aplicaciones.

3.1. El pensamiento computacional en la educación

El desarrollo del pensamiento computacional se ha abordado usando diferentes elementos, por ejemplo, sistemas de software como Scratch (MIT Media Lab, 2019), Disney (Disney, 2019); plataformas educativas completas como Code.org y Duolingo (Code.org, 2019; Duolingo, 2019); y en diferentes trabajos de investigación (Calderon, Crick, & Tryfona, 2015; Grover & Pea, 2013; Hsu et al., 2018; Roscoe, Fearn, & Posey, 2014; Yadav, Hong, & Stephenson, 2016; Yadav, Mayfield, Zhou, Hambrusch, & Korb, 2014).

En México, el Programa *Cuantrix* dio inicio en octubre del año 2017 gracias a IZZI Telecom, la Asociación Mexicana de la Industria de Tecnologías de la Información (AMITI) y Fundación Televisa en colaboración con Code.org con el objetivo de incentivar a los estudiantes a aprender sobre las Ciencias de la Computación y Programación (Cuantrix, 2019). Cuantrix dispone de material descargable en formato de documento portátil (PDF) en su página web para los docentes que deseen formar parte de la inclusión de este conocimiento en sus escuelas de diferentes niveles educativos, desde primaria hasta preparatoria.

3.2. Aplicaciones móviles educativas

Investigadores trabajan en la creación de aplicaciones educativas de diversos temas y disciplinas de conocimiento con el fin de desarrollar mejores herramientas de aprendizaje

para todos los alumnos pero, principalmente, se han enfocado en aquellos estudiantes que presentan mayor dificultad de comprensión, baja evaluación o tasa de aprobación.

3.2.1. Edumóvil

Edumóvil, es una aplicación para mejorar el proceso enseñanza-aprendizaje de los niños de nivel primaria apoyándose en la tecnología móvil incorporada en el aula (Gerónimo-Castillo & Rocha-Trejo, 2014). Edumóvil pretende utilizar los beneficios de la tecnología móvil como el bajo costo, la movilidad, las responsabilidades individuales y la oportunidad de trabajo colaborativo.

Este sistema contiene aplicaciones para Matemáticas de primer año y Ciencias Naturales para tercero, cuarto y quinto grado, y se planeaba el desarrollo de aplicaciones para temas de Español, Matemáticas de segundo y tercer grado, e Historia.

En este trabajo se realizaron pruebas de usabilidad con varios dispositivos móviles en donde se observó que: las niñas tienen más dificultad que los niños en el uso de los botones, los niños tienen menos cuidado con un *stylus* (pluma para pantalla) que las niñas y que en general, aquellos niños que ya tenían experiencia en videojuegos presentaban mayor agilidad en el uso de los dispositivos.

El trabajo concluye que se deben realizar pruebas de usabilidad en dispositivos actuales y enfocarse en la enseñanza de matemáticas, que es donde se presentan más dificultades. Edumóvil no pretende sustituir al profesor, sino servir de apoyo en el salón de clases.

3.2.2. ARGeo

ARGeo es una aplicación móvil que utiliza realidad aumentada y computación afectiva para mejorar el aprendizaje en temas de geometría. La realidad aumentada es aplicada mediante marcadores, pequeños papeles con símbolos que son detectados por el software para representar, dentro de él, una figura tridimensional con la cual el usuario debe interactuar. Esta aplicación fue pensada para que los alumnos conocieran y entendieran de mejor manera las figuras en tres dimensiones que normalmente se enseñan usando un espacio de dos dimensiones como es el pizarrón (Zatarain-Cabada, Barrón-Estrada, Ibañez-Espiga, & Uriarte-Portillo, 2018). Una actividad de ARGeo se muestra en la Figura 3-1 donde se emplea la colisión de objetos en tres dimensiones.

Se realizó una primera evaluación a profesores que imparten la materia en el nivel secundaria y se concluyó que los estudiantes podrían aumentar su motivación en aprender estos temas.



Figura 3-1. Actividad en ARGeo.

3.3. Aplicaciones orientadas al pensamiento computacional

A partir de los trabajos sobre Pensamiento Computacional que impulsó Wing, los investigadores también han trabajado en la generación de aplicaciones en este tema, ya sea de manera conceptual, abarcando muchos aspectos o enfocándose en pocas o una sola habilidad en específico para tratar de mejorarla.

3.3.1. Prototipo para edades tempranas

En (Calderon et al., 2015) se desarrolló una app para iOS con ejercicios de reconocimiento de patrones dirigido a niños de 3 a 5 años. La aplicación muestra un conjunto de cuatro imágenes donde el usuario debe seleccionar las imágenes idénticas, que incluyen objetos cotidianos y cuerpos geométricos. La Figura 3-2 presenta dos escenarios de la aplicación.

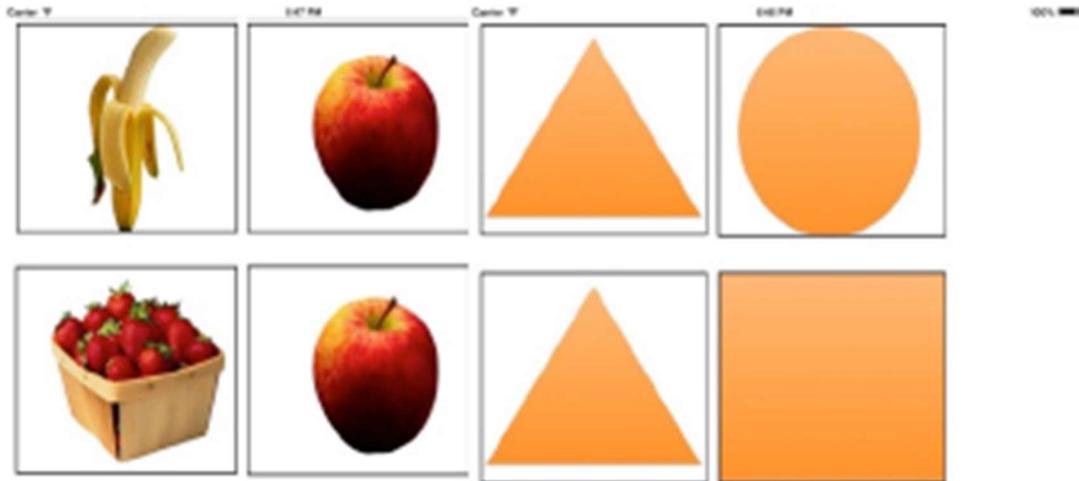


Figura 3-2. Prototipo de app de reconocimiento de patrones para edades tempranas.

3.3.2. EasyLogic

EasyLogic es sistema tutor inteligente afectivo que apoya el aprendizaje de lógica algorítmica y programación presentado en (Zatarain-Cabada, Barrón-Estrada, & Ríos-Félix, 2017). Se desarrolló para web y se apoya haciendo uso de bloques gráficos provistos por la librería Blockly. EasyLogic se divide en 3 secciones principales:

- Aprende: provee ejercicios para enseñar cada estructura de control existente en la programación.
- Imagina y crea: permite al estudiante crear sus propios algoritmos y ejecutarlos.
- Codifica: posibilita al estudiante programar directamente en JavaScript en vez de utilizar bloques (aún en desarrollo).

En la Figura 3-3 se muestra la interfaz de usuario del sistema EasyLogic mostrando un ejercicio de estructura repetitiva. La interfaz tiene 8 secciones que se utilizan para brindar información al usuario, estas son:

1. Encabezado con información del jugador.
2. Estatuto principal que se utiliza en el ejercicio.
3. Nivel del ejercicio a realizar.
4. Opciones de estatutos disponibles.

5. Sección que muestra el algoritmo creado usando bloques.
6. Descripción gráfica del problema a resolver.
7. Información para el usuario.
8. Código de programación que representa al algoritmo creado.

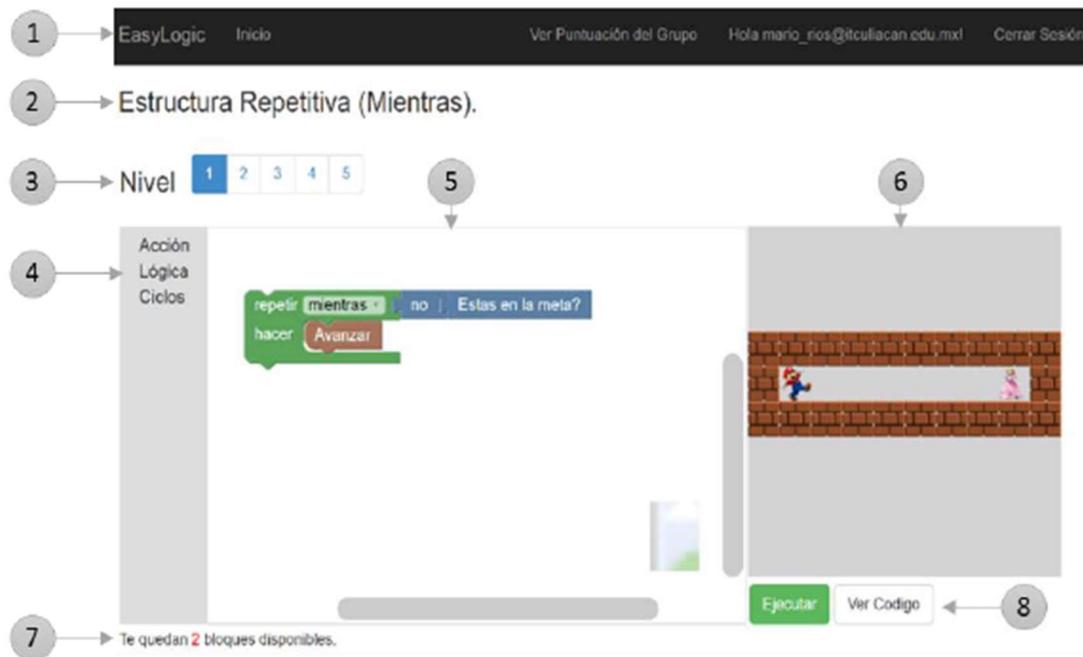


Figura 3-3. Ejercicio de estructura repetitiva en EasyLogic.

3.4. Videojuegos con patrones

Existen algunos juegos populares que a pesar de ser destinados al entretenimiento exigen al jugador agilidad mental como resolver problemas de matemáticas, estrategia, lógica o reconocer patrones.

3.4.1. Candy Crush Saga

Candy Crush Saga (King.com LTD, 2019) es un juego de puzzle desarrollado por King.com Ltd lanzado en 2012 originalmente como aplicación para Facebook. Hoy en día el videojuego está disponible también en iOS, Android, Microsoft Store y en King.com y tiene millones de jugadores alrededor del mundo.



Figura 3-4. Formando un dulce especial en Candy Crush.



Figura 3-5. Nivel 217 en Candy Crush, del tipo "bajar frutas".



Figura 3-6. Mapa de niveles de Candy Crush.

Candy Crush Saga tiene una temática sobre repostería, la mecánica principal consiste en unir tres o más dulces del mismo tipo dentro de un tablero intercambiando sus posiciones. El tablero es diferente en cada nivel, con distintos tipos de dulces y puede contener obstáculos estáticos o con cierto comportamiento. Las combinaciones de más de tres caramelos producen dulces especiales dependiendo el patrón formado, por ejemplo, una “L” forma un *dulce envuelto* (Figura 3-4). Dominar esta mecánica aumenta significativamente la probabilidad de victoria.

Los niveles varían en la condición de victoria, entre otros, llegar a cierta cantidad de puntos, consumir una cantidad de dulces en específico o bajar frutas (Figura 3-5). La puntuación impacta directamente en la cantidad de estrellas otorgadas en ese nivel al jugador.

Al ser un videojuego naturalmente cuenta con aspectos de gamificación, por mencionar algunos: puntos, nivel de dificultad, niveles de progresión y estrellas. En la Figura 3-6 se pueden apreciar algunos elementos de gamificación.

3.4.2. 4 fotos 1 palabra

4 Fotos 1 Palabra (GmbH, 2019) es un juego desarrollado por LOTUM GmbH que requiere que el usuario determine, con base en cuatro imágenes, la palabra a la que hacen alusión. Este juego está disponible tanto para Android como para iOS.

La mecánica principal disponible para el usuario consiste en responder tocando las letras ubicadas en la parte inferior de la pantalla. El usuario necesita reconocer los patrones en las ilustraciones que brinda la app para encontrar la palabra que los relaciona. Estas ilustraciones pueden o no ser elementos con los que el usuario esté familiarizado por lo que su desempeño se ve afectado por su conocimiento en general.

La Figura 3-7 muestra el nivel 10 de la aplicación y, en seguida, se responde correctamente en la Figura 3-8.



Figura 3-7. Nivel 10 de 4 fotos 1 palabra.



Figura 3-8. Nivel resuelto de 4 fotos 1 palabra.

3.5. Comparación de aplicaciones

La Tabla 3-1 presenta un concentrado de las características más importantes de los diferentes trabajos analizados en este capítulo, con el fin de realizar una comparación entre ellos. Como se puede observar, las aplicaciones comerciales contienen elementos de gamificación y trabajan en dispositivos móviles, sin embargo, no tienen un enfoque educativo ni se basan en los temas de educación primaria. Además, de los trabajos producto de investigaciones realizadas, ninguna de las aplicaciones desarrolladas para incentivar la habilidad del reconocimiento de patrones está enfocada en el esquema curricular de primaria y contienen algunos elementos de gamificación.

Tabla 3-1. Tabla comparativa del estado del arte.

App	Plataforma			Gamificación			Objetivo	Habilidades del PC	Basado en esquema curricular de primaria	Enfocado en la Educación
	Web	Móvil	Escritorio	Puntuación	Nivelación	Progresión				
Candy Crush Saga	X	X	X	X	X	X	Aplicación Comercial	Reconocimiento de Patrones	No	No
4 fotos 1 palabra		X		X	X	X	Aplicación Comercial	Reconocimiento de Patrones	No	No
Edumóvil		X					Investigación	N/A	Sí	Sí
ARGeo		X		X			Investigación	N/A	Sí	Sí
(Calderon et al., 2015)		X				X	Investigación	Reconocimiento de Patrones	No	Sí
EasyLogic	X				X	X	Investigación	Abstracción y Diseño de Algoritmos	No	Sí

Capítulo 4

4. Desarrollo del proyecto

Este capítulo describe el proceso de ingeniería de software empleado para el desarrollo del sistema inteligente y gamificado para el desarrollo del pensamiento computacional a través del reconocimiento de patrones de nombre *Patrony*.

Se abordan cinco temas que comprenden la propuesta de solución: el Proceso de desarrollo de software, el Análisis de la información obtenida de los interesados en el desarrollo del sistema, el Diseño del sistema que presenta la Arquitectura con sus componentes y relaciones además de la Interfaz gráfica de usuario para determinar la interacción humano-máquina, la Implementación con las herramientas seleccionadas, y finalmente la Depuración y pruebas del sistema.

4.1. Propuesta de solución al problema

En esta sección se describe la propuesta de solución al problema planteado en la sección 1.2, mostrando algunos ejercicios contenidos en el libro de desafíos matemáticos de los grados de cuarto, quinto y sexto de primaria.

En la Figura 4-1 y Figura 4-2, se presentan dos ejercicios del libro de 4to grado donde el estudiante debe encontrar la figura faltante en una secuencia que presenta dos figuras geométricas que se reproducen para formar figuras más complejas.

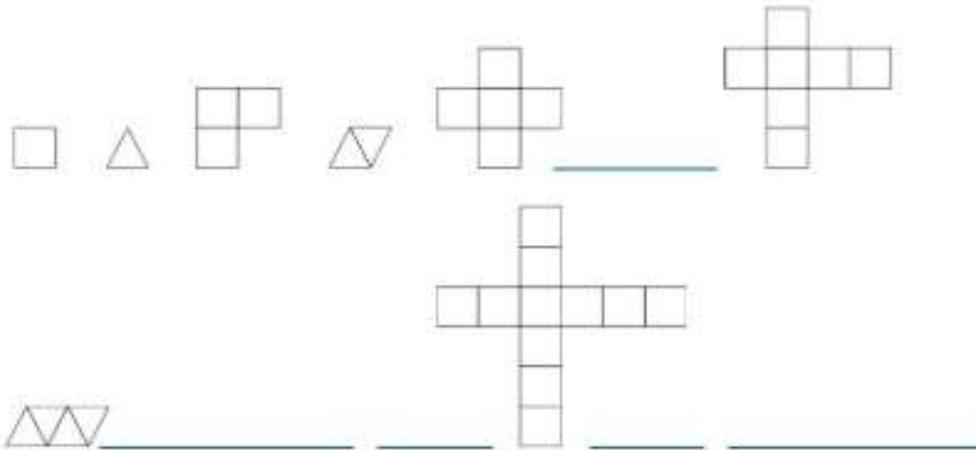
8

Y ahora, ¿cómo va?

Consigna

En equipos de tres, analicen, discutan y posteriormente resuelvan los ejercicios.

1. Encuentren los elementos faltantes en las siguientes sucesiones.



Encierran en un círculo las figuras que forman parte de la sucesión anterior e indiquen qué lugar ocupan.

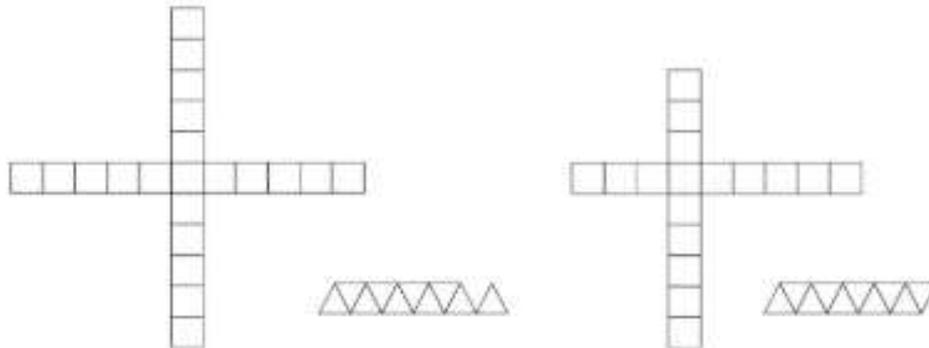
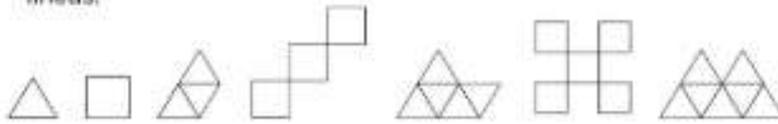
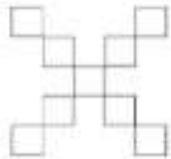


Figura 4-1. Primera parte de los ejercicios del libro de 4°.

2. ¿Qué elementos faltan en esta sucesión? Dibújenlos sobre las líneas.





Estas figuras forman parte de la sucesión anterior; anoten qué lugar ocupan.

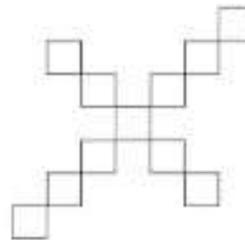


Figura 4-2. Segunda parte de los ejercicios del libro de 4°.

En la Figura 4-3 aparece un ejercicio de patrones numéricos que se encuentra en el libro de 5to grado. Como se puede observar, en este ejercicio el estudiante debe utilizar el concepto *Regularidad* para responder a las preguntas que se plantean, además, debe encontrar términos en las sucesiones de fracciones.

62 Uso de patrones

Consigna

Reunidos en parejas, resuevan los siguientes problemas.

1. ¿Cuál de las siguientes descripciones corresponde a la regularidad de la sucesión $\frac{1}{2}, 1, \frac{3}{2}, 2, \frac{5}{2}, 3, \dots$?

La regularidad es que aumenta cada término de 2 en 2.
La regularidad es que al término anterior se le aumenta 2 al numerador.
La regularidad es que al término anterior se le suma $\frac{2}{2}$ para obtener el siguiente término.
La regularidad es que cada término se determina aumentando $\frac{1}{2}$ al término anterior.
2. ¿Cuál es la regularidad de la siguiente sucesión? Descríbanla.
 $\frac{1}{16}, \frac{5}{16}, \frac{9}{16}, \frac{13}{16}, \dots$

3. ¿Cuál es el término que falta en la siguiente sucesión?
 $\frac{1}{8}, \frac{1}{4}, \frac{3}{8}, \frac{\quad}{8}, \frac{5}{8}, \dots$
4. ¿Cuál es el término que continúa la siguiente sucesión?
 $\frac{1}{4}, \frac{1}{2}, \frac{3}{4}, 1, 1\frac{1}{4}, 1\frac{1}{2}, \dots$



118
Desafíos matemáticos

Figura 4-3. Patrones numéricos con fracciones en libro de 5°.

En la Figura 4-4 aparecen ejercicios de patrones numéricos con números enteros que se encuentran en el libro de 5to grado. En estos ejercicios el estudiante debe encontrar términos faltantes en las sucesiones además de describir como encontraron los términos.

82 **¿Cuál es el patrón?**

Consigna

En equipo, resuelvan los siguientes problemas. Pueden utilizar la calculadora.

- Encuentren los términos faltantes de las siguientes sucesiones:
 - 1, 4, 16, _____, 256, 1024, 4096, _____, _____
 - 4, 28, 196, 1372, _____, _____, _____, 3294172, ...
- ¿Cómo encontraron los términos faltantes en cada sucesión?

- En un estadio de fútbol, los patrocinadores de los equipos que jugaron la final regalaron una camiseta y una gorra auto-grafiadas por los jugadores a los aficionados cuyos boletos de entrada pertenecieran a la siguiente sucesión.

9, 27, 81, 243, 729, 2187...
 - Si Norberto tiene el boleto 19683, ¿se ganó la camiseta y la gorra? Argumenta tu respuesta.



Quinto grado | 161

Figura 4-4. Patrones numéricos con enteros en el libro de 5°.

En la Figura 4-5 se observan ejercicios de patrones de figuras geométricas que se encuentran en el libro de 5to grado. En estos ejercicios el estudiante debe diseñar y construir sucesiones con progresión geométrica, además de identificar la regularidad en las sucesiones construidas por sus compañeros de clase y explicarlas verbalmente a sus compañeros de grupo.

Bloque V

2. Diseñen una sucesión con progresión geométrica de 10 elementos como máximo. Consideren los siguientes pasos.

- Construyan la sucesión solicitada.
- Intercámbienla con otro equipo.
- Identifiquen la regularidad planteada en la sucesión que intercambiaron.
- Expliquenla a sus compañeros de grupo.



Figura 4-5. Patrones de figuras geométricas en el libro de 5°.

Por otra parte, en el libro de 6to grado, se presentan problemas relacionados con estructuras secuenciadas tal como se muestra en la Figura 4-6 y la Figura 4-7.

76 Estructuras secuenciadas

Consigna

En pareja, resuelvan los problemas.

1. Las siguientes estructuras están armadas con tubos metálicos y hojas cuadradas de vidrio.



Estructura 1



Estructura 2



Estructura 3



Estructura 4



Estructura 5

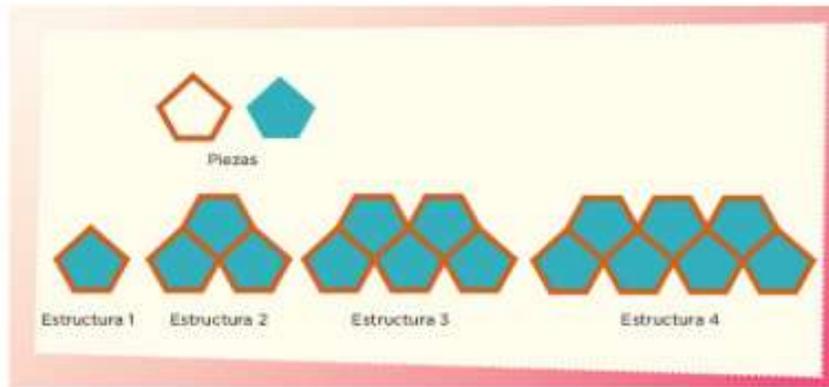
a) ¿Cuántos tubos metálicos se necesitan para hacer la estructura 4?

b) ¿Cuántos tubos metálicos se necesitan para hacer una estructura con 10 hojas de vidrio?

c) ¿Y con 15 hojas de vidrio?

Figura 4-6. Primera parte de los ejercicios de patrones en el libro de 6°.

2. Estas estructuras están armadas con tubos metálicos y hojas pentagonales de vidrio.



a) ¿Cuál es la sucesión numérica que representa las cantidades de tubos de las estructuras?

b) ¿Cuántos tubos y cuántas hojas de vidrio se necesitan para formar la estructura 10?

c) ¿Y para la estructura 15?

Figura 4-7. Segunda parte de los ejercicios de patrones en el libro de 6°.

Los ejercicios encontrados en los libros de texto y la entrevista realizada a la docente, sirvieron para definir las características del sistema de software que se desarrolló para que los niños de educación básica, específicamente de cuarto a sexto grado, contaran con una herramienta tecnológica que les permita ejercitar el reconocimiento de patrones de una forma

dinámica y divertida, mejorando su habilidad para reconocer patrones y resolver problemas relacionados con secuencias de elementos.

El problema detectado consiste en que, el docente utiliza el método de enseñanza tradicional para todo el grupo y no es posible realizar un seguimiento personalizado del aprendizaje de los estudiantes. Además, la motivación de los estudiantes sobre los temas que involucran reconocer patrones es baja.

Como solución a este problema, se propone el desarrollo de una herramienta computacional que permita a los estudiantes desarrollar la habilidad de reconocer patrones utilizando un dispositivo móvil, que permita captar información del desempeño del estudiante y genere acciones, como ejercicios con mayor o menor grado de dificultad adaptándose al ritmo de aprendizaje de cada estudiante. La aplicación deberá ser dinámica, generando múltiples ejercicios con diferentes niveles de dificultad y además debe contener algunos elementos de gamificación para hacer el proceso de aprendizaje más divertido y motivante para los estudiantes.

4.2. Proceso de desarrollo de software

Para el desarrollo del sistema se combinaron dos modelos de desarrollo: el modelo de prototipado y el modelo de mejora iterativa. El modelo de prototipado se empleó para desarrollar un prototipo desechable y posteriormente se utilizó el modelo de mejora iterativa realizando dos iteraciones, con el fin de producir versiones incrementales del sistema. Este proceso de desarrollo que combina los dos modelos se presenta en la Figura 4-8.

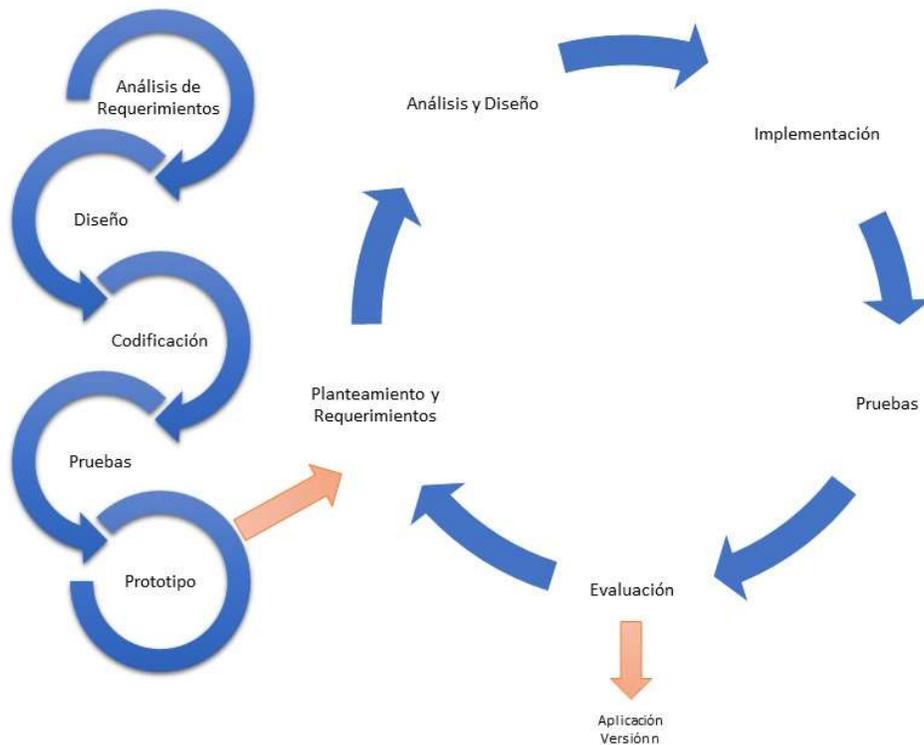


Figura 4-8. Proceso de desarrollo Prototipado-Iterativo.

A grandes rasgos el proyecto se construyó en tres etapas, las cuales se detallan a continuación:

- ✓ **Etapa 1.** Desarrollo de prototipo desechable. Con el fin de explorar las herramientas y tecnologías disponibles para el desarrollo multiplataforma, se desarrolló un prototipo del sistema representando aspectos fundamentales de la idea propuesta para el desarrollo del reconocimiento de patrones por medio de un juego didáctico. El prototipo generado se utilizó para ejecutar actividades de prueba de concepto definiendo la interacción entre el usuario y el sistema con interfaces simples y actividades estáticas. Asimismo, los datos que se requería almacenar para su posterior uso fueron determinados dentro de esta etapa.
- ✓ **Etapa 2.** Desarrollo de la aplicación considerando calidad. Tomando como base el prototipo generado en la etapa 1, se dio inicio a la primera iteración para desarrollar la primera versión del sistema incluyendo los requisitos funcionales generales y

considerando los atributos de calidad requeridos por el sistema. Como producto de esta etapa, se obtuvo un sistema que ofrecía las interfaces gráficas de usuario por medio de las cuales se establecía la interacción principal del usuario con el sistema y el mecanismo de generación de actividades.

- ✓ **Etapa 3.** Desarrollo de la aplicación incluyendo gamificación, perfiles de usuario y actividades. La versión del sistema generada en la etapa 2, se usó como entrada en la segunda iteración del desarrollo para generar una nueva aplicación que incluye nuevas características como son: diversos elementos de gamificación, la creación de perfiles de usuario para almacenar la información que se genera cuando un usuario utiliza la aplicación y una mejora en el generador de actividades. Adicionalmente se atendieron detalles y sugerencias detectadas durante las pruebas descritas en el 4.6.

En las siguientes secciones de este capítulo se detalla la información de las etapas 2 y 3 del desarrollo de la aplicación.

4.3. Análisis

Los requerimientos funcionales y de calidad del sistema, se establecieron con base en las actividades siguientes:

- Búsqueda de información.
- Investigación de aplicaciones existentes que pueden ser usadas o fueron diseñadas para desarrollar el pensamiento computacional.
- Realización de un prototipo desechable de acuerdo con el modelo de proceso de desarrollo de software de prototipado.
- Evaluación del prototipo por una docente de quinto año de primaria, quien ofreció retroalimentación para mejorar el prototipo.

4.3.1. Análisis de requerimientos

Como resultados del proceso de educación de requerimientos, se obtuvieron una serie de requisitos generales de usuario, los cuales se listan a continuación:

1. Aplicación móvil que funcione en iOS y Android.
2. Debe contener actividades que requieran el uso de la habilidad de reconocimiento de patrones por parte del usuario para resolverse como secuencias de figuras y números.
3. Las actividades deben iniciar en el nivel más básico e ir incrementando su dificultad de manera gradual.
4. La aplicación debe contener elementos de gamificación para impulsar la motivación del usuario, tales como nivelación, progresión y puntuación.
5. Debe manejar perfiles de usuario para conocer los datos generales y resultados de las actividades de cada uno.
6. Se debe incluir retroalimentación visual y auditiva al usuario.
7. La aplicación no debe necesitar conexión a internet para funcionar.

Los requerimientos de usuario fueron analizados y clasificados para definir las características del producto a generar en cada una de las etapas de desarrollo.

A continuación, se muestra la clasificación de los requisitos funcionales, de calidad y restricciones asociados al análisis del sistema. Todos contienen una clave de identificación, descripción y un grado de prioridad de atención para el desarrollo.

Requisitos funcionales

Los requisitos funcionales representan el comportamiento del sistema ante posibles entradas del usuario, así como también las salidas que genere hacia este usuario. Algunas funcionalidades del sistema pueden no requerir una entrada o no proveer una salida al usuario. En la Tabla 4-1 se listan los requisitos funcionales del sistema.

Tabla 4-1. Requisitos funcionales para el sistema de reconocimiento de patrones.

Identificador	Descripción	Prioridad
RF-01	El sistema debe de contener ejercicios que requieran reconocer patrones.	Alta
RF-02	El sistema debe organizar los ejercicios dentro de actividades. Las actividades serán los niveles (etapas o fases) del juego que el usuario tiene que superar.	Alta
RF-03	El sistema debe implementar diversos mecanismos de gamificación (ejemplo manejar un mecanismo de puntuación al responder los ejercicios).	Alta
RF-04	El sistema debe de contar las respuestas erróneas de las actividades.	Alta
RF-05	El sistema debe incluir un cronómetro para conocer el tiempo de finalización de las actividades.	Alta
RF-06	El sistema debe almacenar la cantidad de respuestas erróneas, el tiempo de finalización y la puntuación de cada actividad.	Alta
RF-07	El sistema debe crear perfiles de usuario para almacenar su información (Nombre, edad, género y estadísticas de las actividades) de manera individual.	Media
RF-08	El sistema debe de permitir cambiar el usuario actual por otro ya existente o uno nuevo.	Media
RF-09	El sistema debe contar con un módulo de lógica difusa para determinar el grado de dificultad del siguiente ejercicio.	Media
RF-10	El sistema debe contar con un módulo afectivo para determinar el estado emocional de estudiante.	Baja
RF-11	El sistema debe contener diversos niveles de dificultad de las actividades.	Alta

Requisitos de calidad

Estos requisitos sirven como condición para evaluar la conformidad del proyecto validando la aceptabilidad de un atributo o característica para la calidad del software y se encuentran listados en la Tabla 4-2.

Tabla 4-2. Requisitos de calidad para el sistema de reconocimiento de patrones.

Identificador	Atributo	Descripción	Prioridad
RC-01	Rendimiento	El sistema no debe tardar más de dos segundos en realizar cada función.	Alta
RC-02	Satisfacción	El sistema debe incluir música de fondo y retroalimentación a las acciones de usuario.	Alta
RC-03	Portabilidad	El sistema debe funcionar tanto en dispositivos Android como iOS.	Alta
RC-04	Usabilidad	El sistema debe incluir interfaces gráficas de usuario que sean fáciles de usar.	Alta

Restricciones

La información en la Tabla 4-3 corresponde a las restricciones del sistema. Las restricciones delimitan el alcance de las funcionalidades del sistema.

Tabla 4-3. Restricciones para el sistema de reconocimiento de patrones.

Identificador	Descripción	Prioridad
RN-01	El sistema debe evitar el uso de internet.	Alta
RN-02	Los ejercicios del sistema deben resolverse mediante toques, deslizamientos, arrastrado de objetos o rotaciones del dispositivo, evitando en mayor medida el uso del teclado en pantalla.	Alta
RN-03	La información que cada usuario puede visualizar en el sistema debe ser apropiada a su perfil.	Alta
RN-04	El sistema debe funcionar a partir de las versiones Android 7.0 y iOS 10.0.	Alta
RN-05	El sistema debe de funcionar solo en modo horizontal o <i>landscape</i> .	Alta

4.3.2. Actores

El único actor presente en el sistema es el alumno, que interactúa directamente con el sistema mediante toques en la pantalla del dispositivo móvil.

4.3.3. Diagrama de contexto

La interacción entre los actores e ilustración del flujo de datos se describe en el diagrama de contexto del sistema ilustrado en la Figura 4-9.

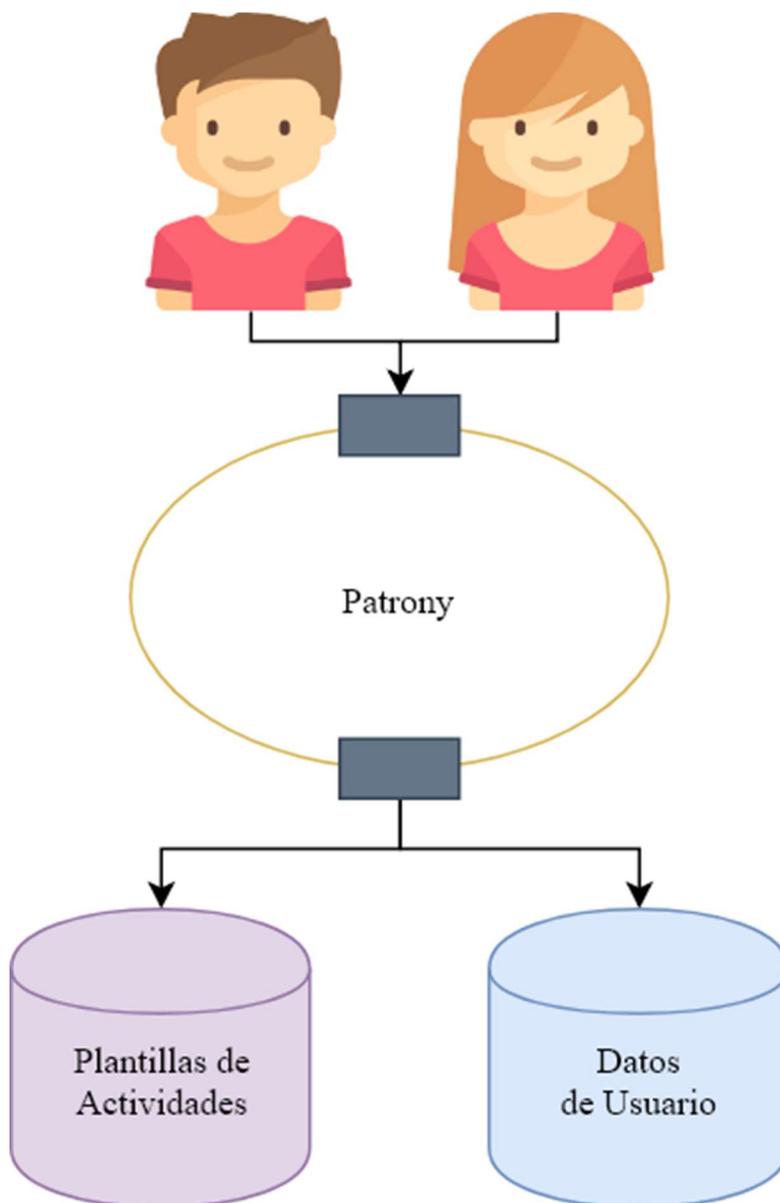


Figura 4-9. Diagrama de contexto del sistema.

El alumno interactúa mediante toques en la pantalla del dispositivo móvil con el fin de crear su perfil y realizar las actividades incluidas en *Patrony*. A su vez, *Patrony* necesita las plantillas de actividades que generará para el usuario y, además, almacena el perfil del usuario y los resultados de las actividades jugadas por el usuario.

4.3.4. Arquetipos

Los arquetipos son abstracciones que se relacionan entre sí y conforman al sistema, estos pueden ser o no directamente un componente del sistema. La lista de arquetipos se encuentra en la Tabla 4-4.

Tabla 4-4. Arquetipos del sistema de reconocimiento de patrones.

Arquetipo	Descripción
Actividad	Es la entidad que representa un conjunto de ejercicios que serán resueltos por el usuario.
Plantilla	Es la entidad que define el contenido de una actividad.
Perfil	Es la entidad que almacena la información del usuario, así como los resultados de los ejercicios resueltos por el mismo.
Usuario	Es la entidad que resuelve las actividades y a quien se le crea un perfil.

En la Figura 4-10 se visualiza la relación de los arquetipos. Las actividades del sistema dependen de plantillas. Estas plantillas definen una configuración para las actividades que será utilizada por el sistema para generar las actividades que el usuario tiene que realizar. El usuario debe crear un perfil donde se almacenarán sus datos generales, datos de configuración de la aplicación y los resultados de las actividades completadas.

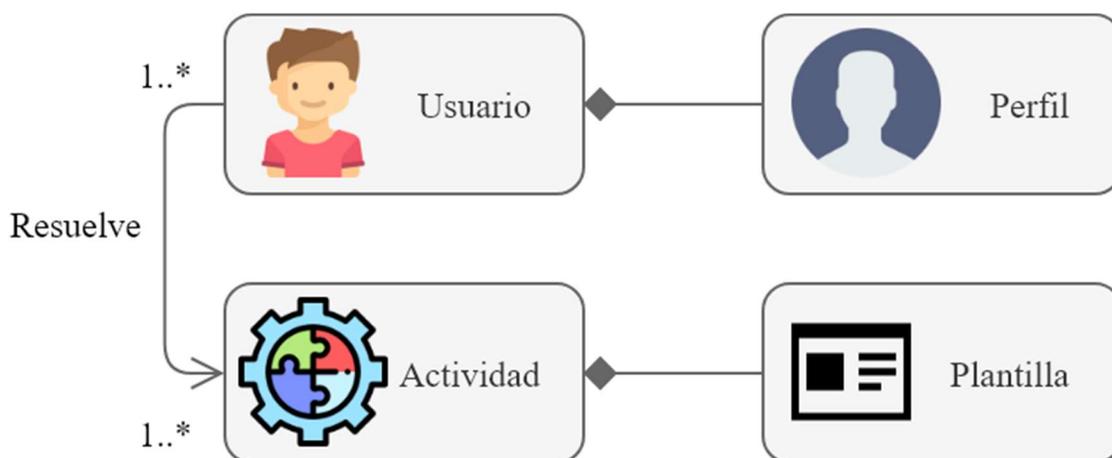


Figura 4-10. Diagrama de arquetipos.

4.4. Diseño

En esta sección se describen los diseños de la arquitectura, el modelo de datos y la interfaz gráfica con los que, en conjunto con el prototipo desechable, se inició el desarrollo de la aplicación.

4.4.1. Arquitectura

Para la elaboración del sistema se consideró el uso del framework React Native desde el diseño de la arquitectura.

El sistema completo se dividió en tres grupos de componentes: Navegadores, Contenedores y Componentes. Adicionalmente se consideraron dos elementos más: el módulo *React Navigation* y el componente *AsyncStorage*.

El diagrama de la arquitectura del sistema se presenta en la Figura 4-11. A continuación, se describen los componentes y grupos de componente antes mencionados.

El módulo *React Navigation* contiene componentes dedicados a la creación y apoyo de navegadores. Los navegadores son componentes que tienen la tarea de administrar la navegación entre las pantallas (contenedores) de la aplicación y están organizados en componentes individuales dentro del grupo Navegadores. Estos componentes se describen en la Tabla 4-5.

El componente *AsyncStorage* se encarga de administrar la persistencia de la información. El componente *Datos* del grupo *Componentes* hace uso de él para almacenar y consultar los datos de la aplicación.

En la 4.5 se profundizará tanto en *React Native* como en *React Navigation* y en *AsyncStorage*.

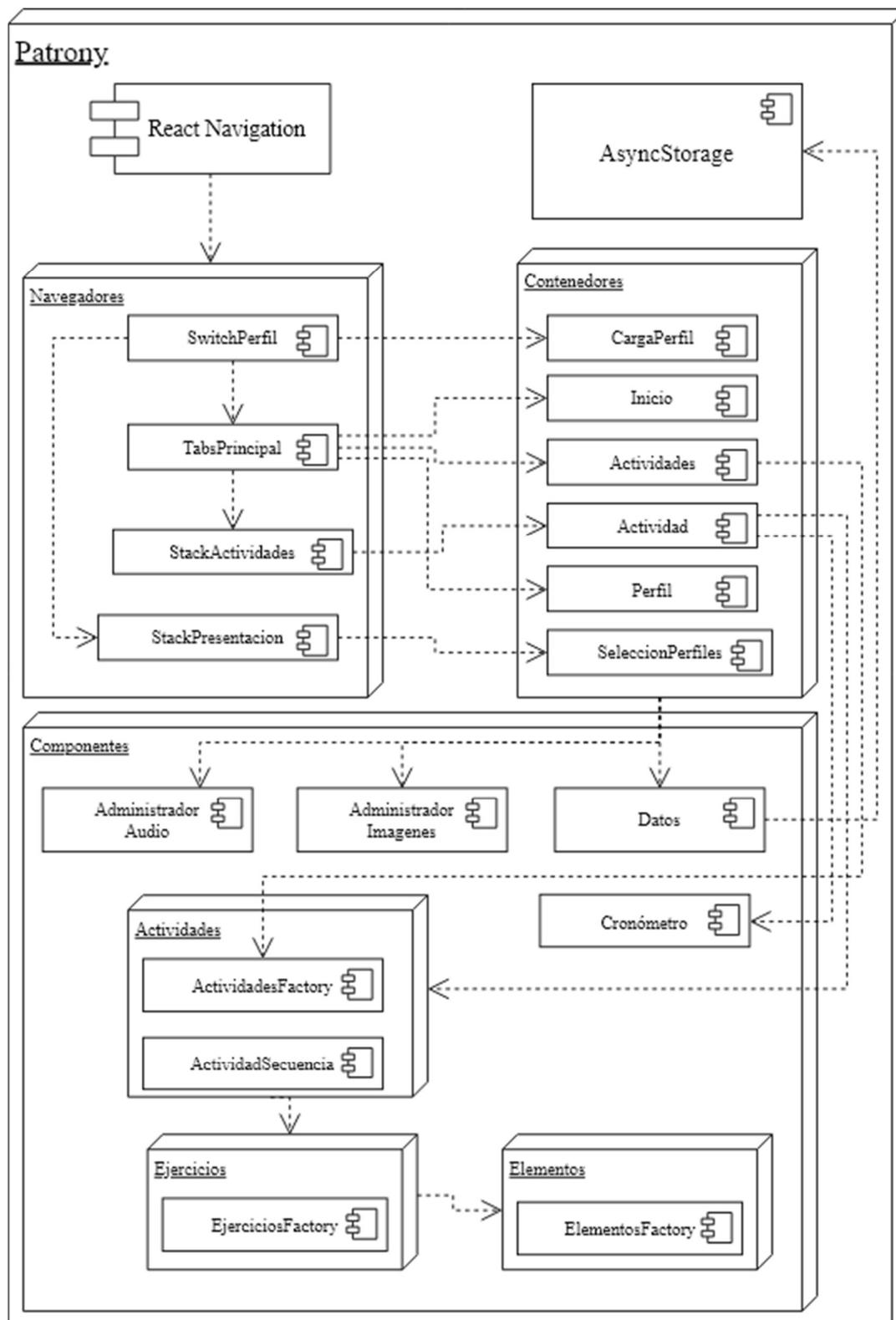


Figura 4-11. Arquitectura del sistema de reconocimiento de patrones *Patrony*.

Tabla 4-5. Componentes del grupo Navegadores.

Componente Navegador	Descripción
SwitchPerfil	Maneja las interfaces de los estados de autenticación, las interfaces que verá un usuario con y sin perfil.
TabsPrincipal	Maneja las interfaces del menú principal del sistema correspondiente a las pestañas de Inicio, Actividades y Perfil.
StackActividades	Maneja la transición entre el listado de actividades y la ejecución de las actividades.
StackPresentacion	Maneja las interfaces del flujo de autenticación, la creación y selección de perfil.

Los contenedores son componentes cuyo fin recae en contener a un conjunto de componentes para construir las pantallas principales, son los que conforman las rutas de los navegadores haciendo que la aplicación sea explorable, la Tabla 4-6 describe cada contenedor del grupo.

Tabla 4-6. Componentes del grupo Contenedores.

Componente Contenedor	Descripción
Inicio	Interfaz de la pestaña Inicio donde se le da la bienvenida al usuario e invita a iniciar o continuar con las actividades.
Actividad	Interfaz donde el usuario resuelve las actividades.
Actividades	Interfaz donde se encuentra el listado de actividades disponibles.
Perfil	Interfaz donde aparece la información del usuario, el detalle de las actividades que ha resuelto y la opción de cambiar de perfil.
CargaPerfil	Interfaz intermediaria de carga para determinar la interfaz a mostrar según: inexistencia de perfiles, existencia de algún perfil o perfil con sesión activa.
SeleccionPerfil	Interfaz en la que se listan los perfiles existentes.

El grupo Componentes contiene todos aquellos componentes que no son navegadores ni contenedores y añaden funcionalidad a otros componentes. Este grupo se desglosa en la Tabla 4-7.

Tabla 4-7. Componentes del grupo Componentes.

Componentes	Descripción
AdministradorAudio	Administra el sonido de fondo y los efectos de sonido de la aplicación.
AdministradorImagenes	Administra el suministro de las imágenes a los demás componentes.
Datos	Componente que hace uso de AsyncStorage para almacenar y extraer la información de la aplicación (4.5.3).
Cronómetro	Componente utilizado en el contenedor Actividad para medir el tiempo de duración de resolución de la actividad.
Actividades <ul style="list-style-type: none"> • ActividadesFactory • ActividadSecuencia 	El grupo Actividades contiene todos los componentes que se refieren a la creación y materialización de las actividades de <i>Patrony</i> . Se apoya del grupo Ejercicios. <ul style="list-style-type: none"> • ActividadesFactory es quien toma las plantillas de las actividades para crear las propiedades necesarias y la actividad adecuada para el usuario. • ActividadSecuencia es un tipo de actividad. Es un componente React.
Ejercicios <ul style="list-style-type: none"> • EjerciciosFactory 	El grupo Ejercicios contiene todos los componentes que se refieren a la creación y materialización de ejercicios de las actividades. Se apoya del grupo Elementos. <ul style="list-style-type: none"> • EjerciciosFactory genera los ejercicios de las actividades.
Elementos <ul style="list-style-type: none"> • ElementosFactory 	El grupo Elementos contiene todos los componentes que se refieren a la creación y materialización de elementos de los ejercicios. <ul style="list-style-type: none"> • ElementosFactory genera los elementos de los ejercicios.

4.4.2. Modelo de datos

El modelo de datos se detalla en la Figura 4-13 y su simbología se describe en la Figura 4-12. La aplicación requiere almacenar las plantillas de sus actividades, los perfiles de los usuarios que la utilicen. Los perfiles de usuario se conforman por un conjunto de datos generales, resultados de las actividades ya completadas y algunas preferencias de la aplicación.

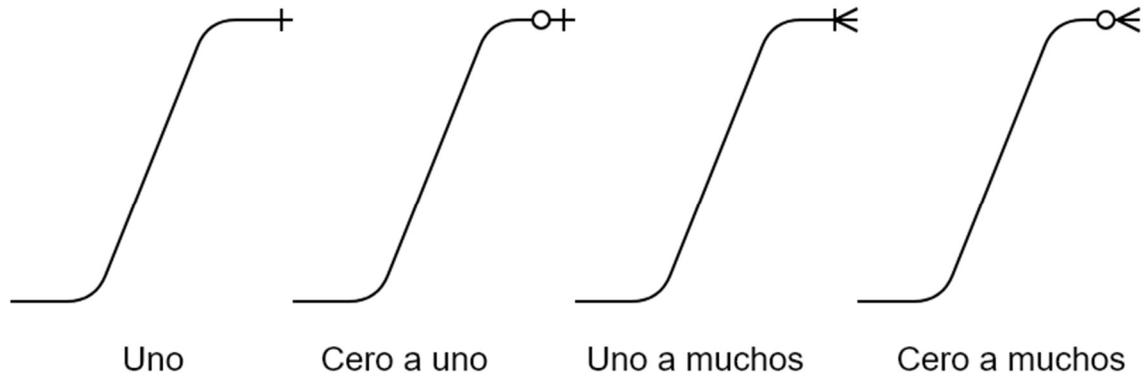


Figura 4-12. Simbología de draw.io.

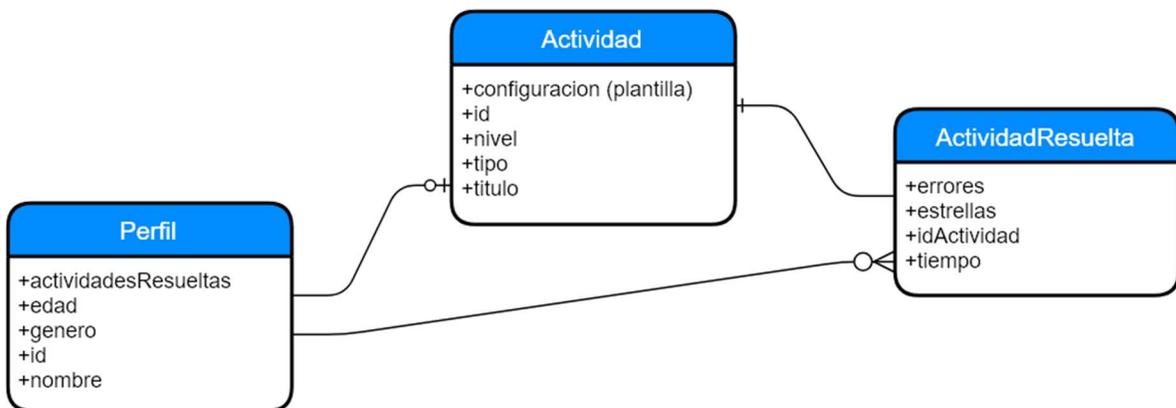


Figura 4-13. Modelo de datos del sistema de reconocimiento de patrones.

Las características de las actividades se estandarizaron en formato JSON (JavaScript Object Notation) y contienen su identificador, título, nivel de dificultad, tipo de actividad y la configuración de los ejercicios que incluye. Este último elemento es la plantilla que define cómo serán los ejercicios generados que deben resolverse para completar la actividad y varían de acuerdo al tipo de actividad. Esto se puede observar en la Figura 4-14 y la Figura 4-15.

```

{
  "id": "SEC001",
  "titulo": "Nivel básico",
  "nivel": 0,
  "tipo": "secuencia",
  "configEjercicios": {
    "cantidad": 10,
    "cantidadElementosDesconocidos": 2,
    "cantidadElementosRespuesta": 5,
    "cantidadElementosSecuencia": 9,
    "carpetaImágenes": "pokemon",
    "longitudSecuencia": 3,
    "tipoElementosDesconocidos": "final"
  }
},

```

Figura 4-14. Estructura de una actividad tipo secuencia.

```

{
  "id": "MAT001",
  "titulo": "Matebásico",
  "nivel": 0,
  "tipo": "matematicas",
  "configEjercicios": {
    "cantidad": 5,
    "cantidadNumerosDesconocidos": 3,
    "cantidadNumerosRespuesta": 4,
    "cantidadNumerosSecuencia": 6,
    "factor": 3,
    "numeroInicial": 2,
    "operacion": "multiplicar",
    "tipoNumerosDesconocidos": "final"
  }
}

```

Figura 4-15. Estructura de una actividad tipo matemáticas.

4.4.3. Interfaz gráfica

Se diseñaron mockups de las interfaces correspondientes al menú principal, a una actividad en general y a la actividad de secuencia de figuras.

La Figura 4-16 muestra las tres pestañas de las interfaces principales de la aplicación con la pestaña Actividades seleccionada en donde se muestra la lista de actividades disponibles. Aquí se introducen la nivelación y progresión como elementos de gamificación.



Figura 4-16. Mockup de la interfaz con las secciones principales.

A continuación la Figura 4-17 presenta el boceto general diseñado para las actividades donde se pueden apreciar diferentes elementos como son: el botón de menú (pausa), el nombre de la actividad, el nivel de dificultad, un cronómetro y otros dos elementos de gamificación: la puntuación del usuario y una barra de progresión para la cantidad de ejercicios de la actividad. Estos elementos están agrupados en la barra superior, el resto de la pantalla le corresponde a la actividad a realizar.

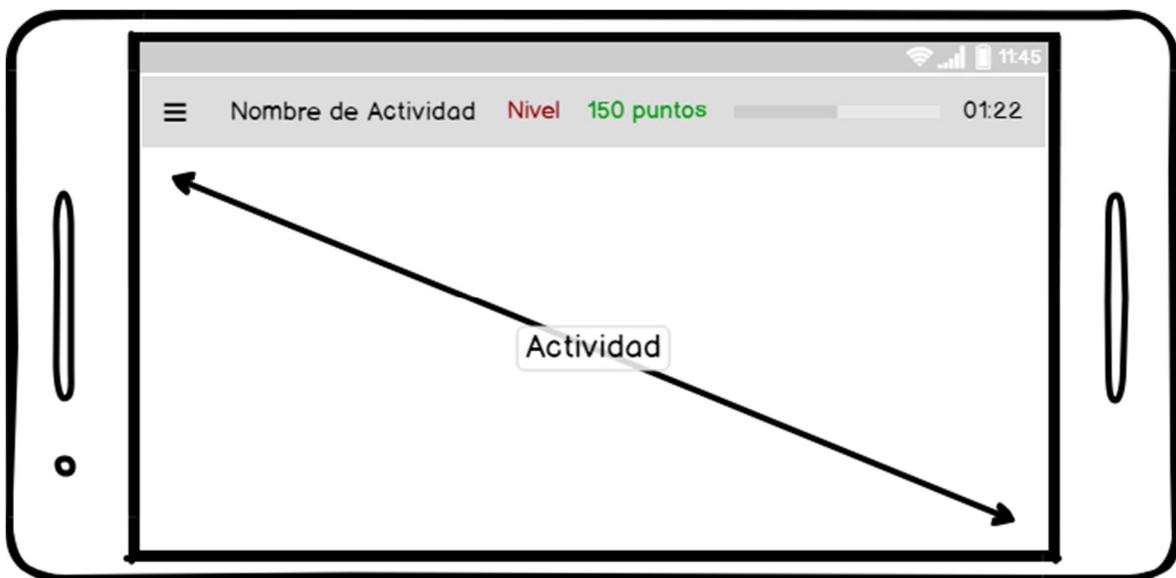


Figura 4-17. Mockup de actividad general.

Finalmente, la Figura 4-18 muestra un boceto de una actividad específica en dos momentos diferentes: antes y después de que el usuario responda a un ejercicio. El primero presenta dos secuencias de números, la lista de números en la parte superior muestra la secuencia incompleta de elementos que representan un patrón de números enteros, mientras que la lista de números de la parte inferior presenta las opciones disponibles para que el usuario seleccione el elemento que falta en la lista superior. El segundo, ofrece retroalimentación inmediata cuando el usuario responde a una actividad, el sistema evalúa la respuesta y marca como correcta la selección.

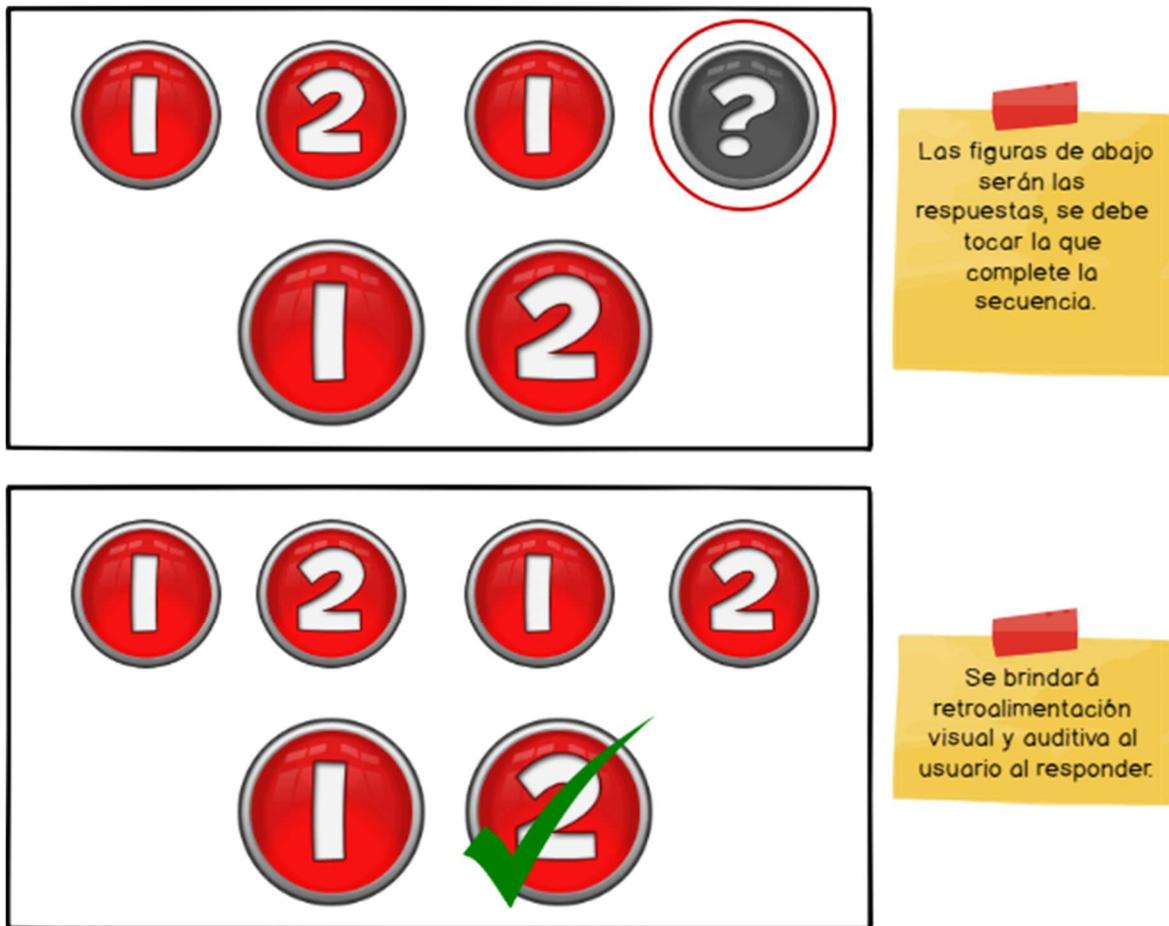


Figura 4-18. Mockup de una actividad de tipo secuencia.

4.5. Implementación

El sistema se construyó con el framework React Native, con su manejo nativo del estado de la aplicación, en conjunto con React Navigation 2.X para la navegación entre pantallas y AsyncStorage para la persistencia de los datos.

4.5.1. React native

La aplicación móvil incluida en este trabajo de tesis se desarrolló con la versión 0.57.4 de React Native. Toda la aplicación se rige bajo este framework y tiene relación directa con la librería React para el diseño de interfaces que provee *componentes React*.

4.5.2. Componentes react

Estos componentes son los elementos que visualizamos en la aplicación y se definen utilizando el principio de responsabilidad única, en la figura se puede mostrar un ejemplo de la definición de componentes React en un mockup (Figura 4-19) mostrado por (Facebook Inc., 2019a).



Figura 4-19. Ejemplo de separación de componentes en un mockup.

Comúnmente los componentes React se pueden clasificar en tres tipos: componentes funcionales (no tienen estado), componentes de clase (pueden tener estado) y contenedores. Los contenedores son componentes de clase que renderizan un espacio de agrupación para el

renderizado de un conjunto de componentes funcionales o de clase resultando en una interfaz completa de la aplicación.

4.5.3. Persistencia con async storage

React Native Async Storage es un sistema de almacenamiento asíncrono del tipo llave-valor, sin encriptación, persistente y global para React Native (Facebook Inc., 2019c). Internamente utiliza RocksDB o SQLite para almacenar la información. Su simplicidad de uso, el manejo de poca información y la ausencia de información sensible fueron las razones principales del uso de este componente incluido en las librerías de React Native.

Para la implementación del diagrama de datos mostrado en la más atrás se crearon las llaves *actividades*, *perfiles* e *idPerfilActual*. La Tabla 4-8 reseña las entidades mostradas en la Figura 4-20, cuya simbología se describió anteriormente en la Figura 4-12.

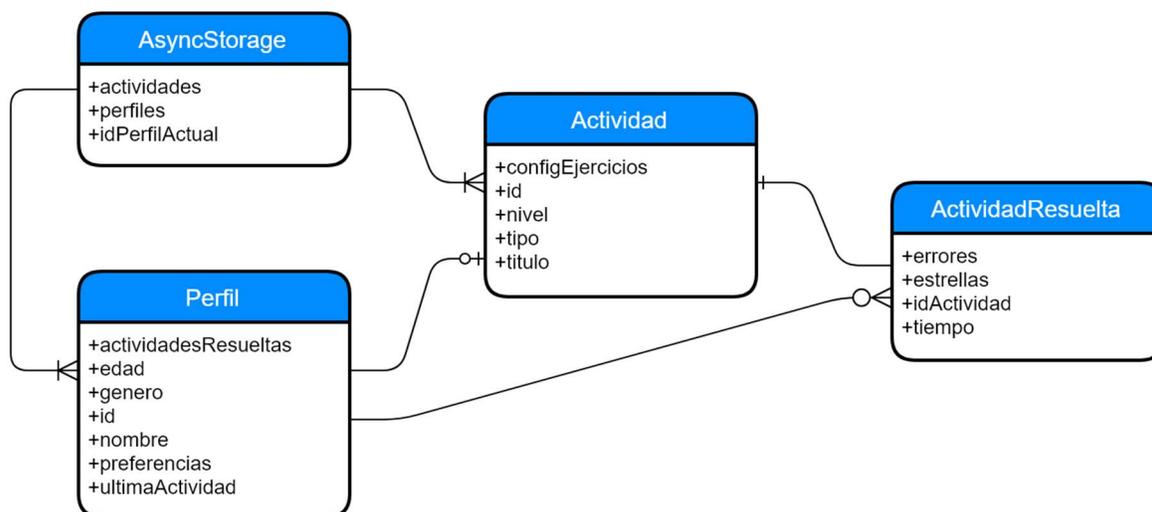


Figura 4-20. Diagrama del modelo de datos de implementación.

Tabla 4-8. Entidades de Datos.

Entidad	Descripción	Propiedades
AsyncStorage	Representa el acceso global a los datos procesados por AsyncStorage.	actividades: almacena entidades Actividad.
		perfiles: almacena entidades Perfil.
		idPerfilActual: identificador del perfil seleccionado.
Perfil	Representa la información de un usuario dentro de la aplicación.	actividadesResueltas: información de las actividades que ha resuelto el usuario.
		preferencias: configuración del usuario para su sesión en la aplicación.
		ultimaActividad: almacena la última actividad que inició.
Actividad	Representa la información de la actividad y contiene la plantilla.	configEjercicios: es la plantilla que se utilizará para generar los ejercicios de la actividad, difiere según el tipo.
ActividadResuelta	Contiene la información recopilada de una actividad que se ha resuelto.	errores: cantidad de errores que el usuario cometió en los ejercicios de la actividad.
		estrellas: cantidad de estrellas con la intención de representar el desempeño del usuario en la actividad.
		idActividad: identificador de la actividad que se resolvió.
		tiempo: duración en segundos de resolución de la actividad.

4.5.4. React navigation

React Navigation es una librería creada por la comunidad de React Native en su búsqueda por una solución escrita en JavaScript que permitiera una navegación sencilla y extensible (React Native Community, 2019). Hay varios tipos de navegadores disponibles de los cuales solo se listan los que se utilizaron en este software:

- **Switch Navigator (Navegador tipo interruptor):** permite una sola interfaz a la vez por lo que no se puede “regresar a la anterior”, comportamiento indicado para autenticación.
- **Bottom Tab Navigator (Navegador tipo pestañas):** muestra una barra de pestañas para intercambiar entre distintas interfaces.
- **Stack Navigator (Navegador tipo pila):** apila las interfaces y permite transición entre ellas. Se utilizó uno para el conjunto de interfaces de registro e intercambio de perfil y otro para las actividades.

4.6. Depuración y pruebas

En general la aplicación fue desarrollada y depurada sobre un dispositivo Moto G5 Plus y posteriormente se realizaron pequeños ajustes de la interfaz gráfica para adecuarse a las características de la tableta Samsung Galaxy S2, dispositivo que se utilizaría en los experimentos. Gracias al desarrollo multiplataforma y a la elección generacional equivalente del iPad Air 2 la aplicación funcionó adecuadamente.

La Figura 4-21 muestra la aplicación desplegada en el dispositivo Moto G5 Plus, la Figura 4-22 en una tableta Samsung Galaxy S2 y la Figura 4-23 en un iPad Air 2. Todas las figuras presentan la interfaz de la actividad básica de tipo secuencia.

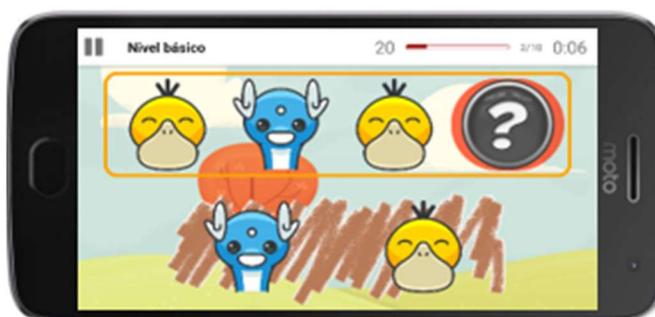


Figura 4-21. Patrony ejecutándose en un Moto G5 Plus.

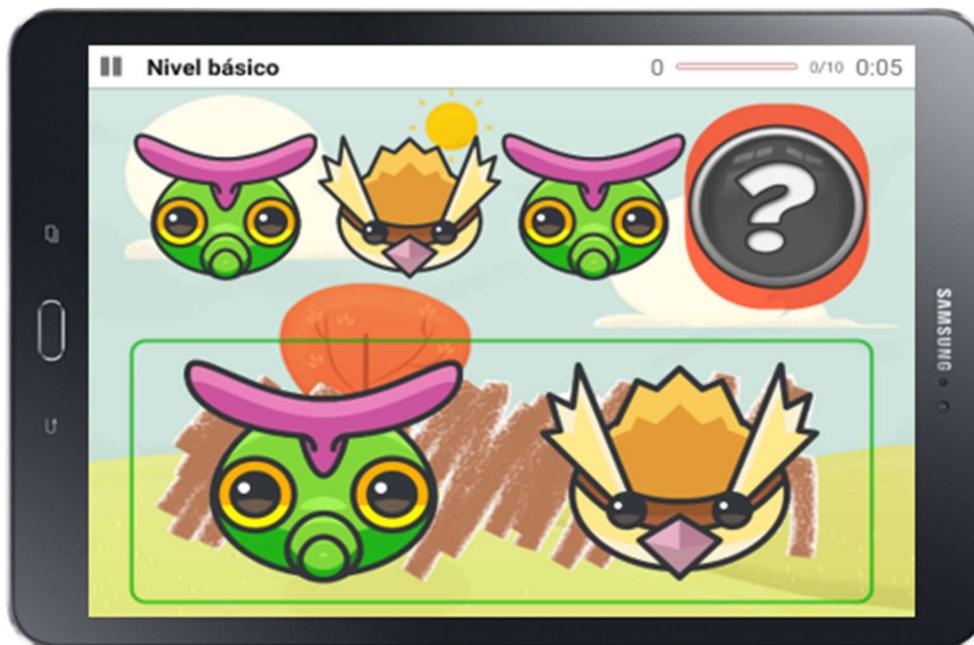


Figura 4-22. Patrony ejecutándose en una Galaxy Tab S2.



Figura 4-23. *Patrony* ejecutándose en un iPad Air 2.

4.6.1. Pruebas

Para efectuar las pruebas de la aplicación, se solicitó a los estudiantes de posgrado que probaran la aplicación en el celular y que ofrecieran una retroalimentación. Los comentarios emitidos por las personas que probaron el sistema permitieron realizar algunos ajustes y revisar casos que no se habían contemplado en la implementación. Dichos casos se describen en la Tabla 4-9.

Tabla 4-9. Relación de comentarios y acciones tomadas.

Comentario	Acción realizada
Los componentes y el texto son muy pequeños.	Se agrandó el tamaño del texto y se ajustaron los componentes.
La pantalla de nuevo perfil es poco intuitiva y tediosa.	Se modificó el formulario para que fuera más gráfico, con imágenes y botones, se dejó solo el uso del teclado para el nombre.
Navegación confusa al cambiar o hacer un perfil nuevo.	Se reacomodaron los contenedores y navegadores para un flujo más adecuado.
No hay instrucciones de qué hacer en la actividad.	Se agregaron instrucciones antes de iniciar las actividades.

Los ejercicios son sencillos y cortos.	<ul style="list-style-type: none">• Se añadieron nuevas figuras para mostrar en las secuencias.• Se ajustaron la cantidad y posición de las incógnitas para aumentar la dificultad.• Se agregaron más actividades.
--	--

4.6.2. Dificultades

Durante el desarrollo de la aplicación se presentaron diversas dificultades que cambiaron la implementación esperada y alargaron el tiempo planeado.

- La interfaz gráfica se desarrolló sobre un smartphone Moto G5 Plus, al probarlo sobre el dispositivo que se iba a utilizar para los experimentos, una Tablet Samsung Galaxy S2, se tuvieron que realizar algunos pequeños ajustes.
- Implementación de arrastrar y soltar (*drag and drop*) complicada, se decidió dejar solo toques (*taps*).
- El estado de la aplicación se manejó inicialmente en componentes arbitrarios que volvieron complejo el funcionamiento. Para arreglar esto se refactorizó el código para que hubiese un flujo único de arriba hacia abajo del estado de los componentes.
- Se intentó implementar un tutorial paso a paso con la librería react-native-copilot sin éxito por incompatibilidad con los componentes construidos para la app.

Capítulo 5

5. Experimentos y resultados

El capítulo actual explica el proceso para la evaluación de la efectividad de la aplicación desarrollada mediante la herramienta pre-test/post-test.

5.1. Intervención con estudiantes

El estudio se llevó a cabo en dos grupos de quinto año de primaria: uno de 29 alumnos en una escuela pública vespertina y el otro de 14 alumnos en una escuela privada matutina, siendo un total de 43 alumnos participantes.

La Figura 5-1 presenta el procedimiento ejecutado en cada escuela para efectuar la intervención con el grupo de estudiantes.

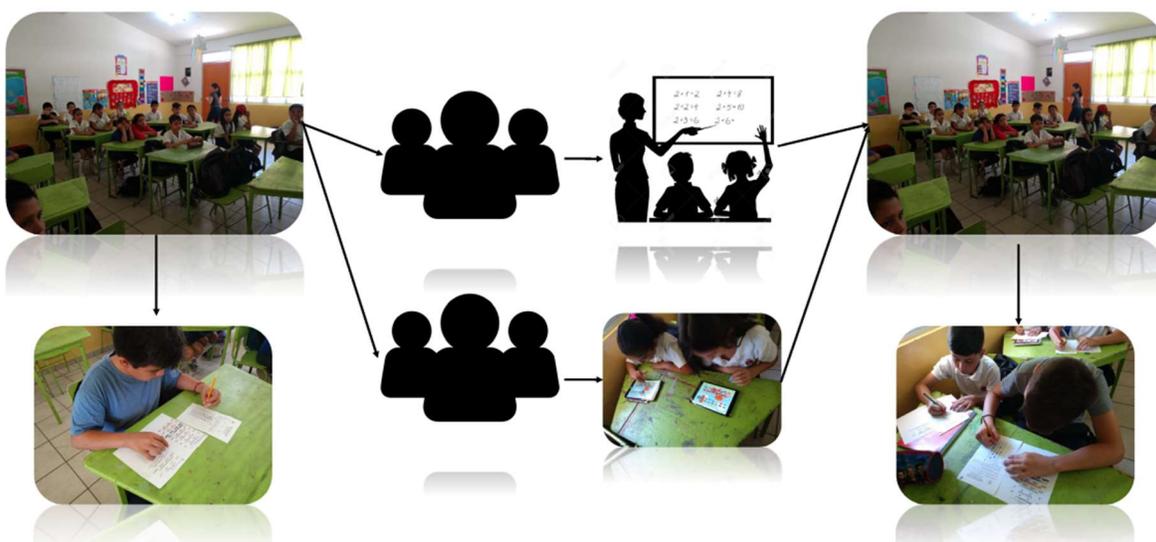


Figura 5-1. Procedimiento de evaluación pre-test/post-test

El procedimiento de cada escuela tuvo una duración aproximada de 90 minutos y fue el siguiente:

1. Organización con el docente y presentación con los alumnos.

2. Aplicación de examen pre-test.
3. División grupal.
4. 15 minutos de clase tradicional con docente y pizarra al grupo de control y 15 minutos de uso de aplicación por parte del grupo experimental.
5. Aplicación de examen post-test.
6. Análisis de resultados.

El examen pre-test se encuentra en el anexo llamado Examen Pre-Test. Este examen incluye secuencias numéricas que involucran operaciones aritméticas cuya implementación en la aplicación no fue concretada por lo que en el post-test, mostrado en el anexo llamado Examen Post-Test, ya no se presentan.

La división grupal se llevó a cabo mediante la asignación de los docentes a dos grupos que ellos consideraran balanceados. Para confirmar este balanceo se consideraron los resultados del pre-test, de manera que el promedio de ambos grupos fuera aproximadamente el mismo.

En el caso de los datos de la escuela pública, de los 29 alumnos que presentaron el pre-test, 4 de ellos no presentaron el post-test, por lo que fueron descartados. De la misma manera, otros 5 alumnos presentaron datos que provocaban ruido en la muestra (*outlayers*), resultando en una muestra final neta de 20 alumnos en esta escuela.

5.2. Análisis de resultados

Se realizaron análisis de los resultados del examen pre-test para confirmar la división de los grupos de control y experimental, después del post-test se observó la diferencia de promedio en los exámenes entre los grupos para notar las diferencias, finalmente se calculó la ganancia relativa para cada estudiante.

5.2.1. Resultados de pre-test

En la Figura 5-2 se aprecian las calificaciones de los 34 alumnos en el examen pre-test aplicado a ambas escuelas, con un promedio de 5.26.

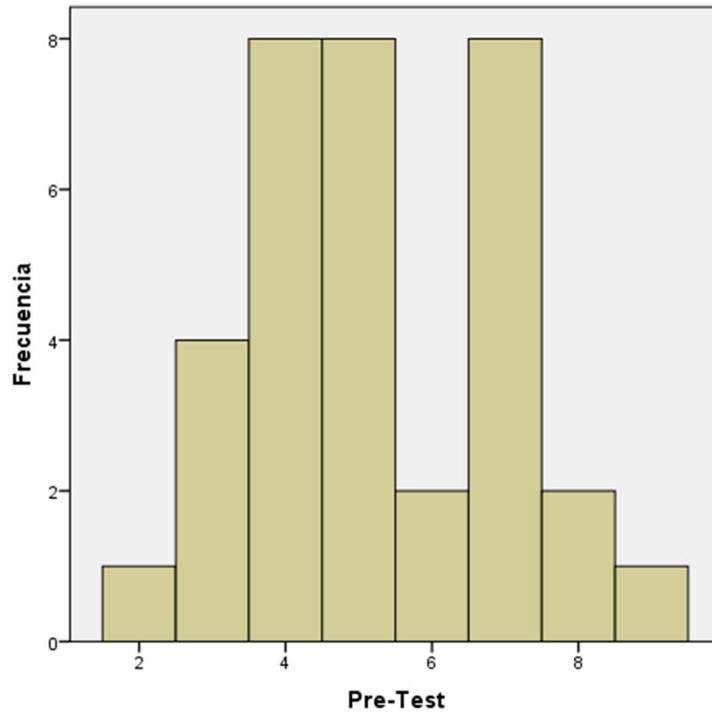


Figura 5-2. Histograma de calificaciones del examen pre-test.

Los estudiantes de la escuela pública tuvieron un promedio de 4.7, siendo solo 6 de los 20 (30%) alumnos quienes aprobaron. Por su parte, la escuela privada tuvo 7 de los 14 (50%) con un promedio grupal de 6.07. La Figura 5-3 muestra una gráfica de aprobación del examen pre-test señalando las escuelas.

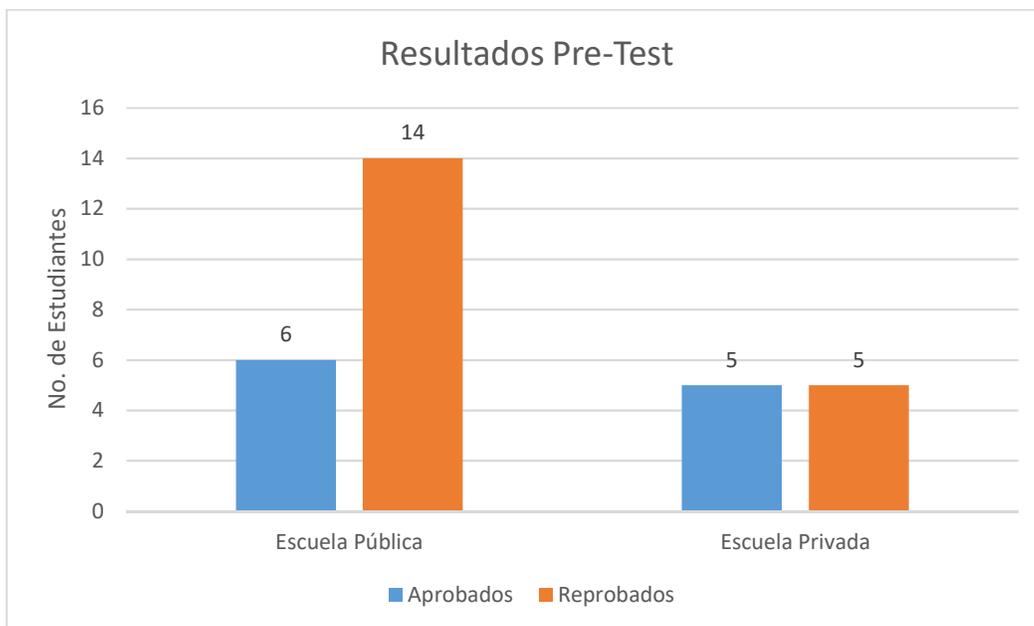


Figura 5-3. Estudiantes aprobados y reprobados del examen pre-test.

5.2.2. Resultados de post-test

Después de que los estudiantes del grupo experimental interactuaran con la aplicación y que el docente impartiera la clase tradicional al grupo de control se sometieron al examen post-test. Las calificaciones del examen post-test se encuentran en la Figura 5-4, siendo el promedio de 5.94.

Los estudiantes de la escuela pública tuvieron un promedio de 5.36 y la escuela privada obtuvo un 7.29. En ambas escuelas se observó que aumentó la cantidad de estudiantes aprobados, ilustrado en la Figura 5-5.

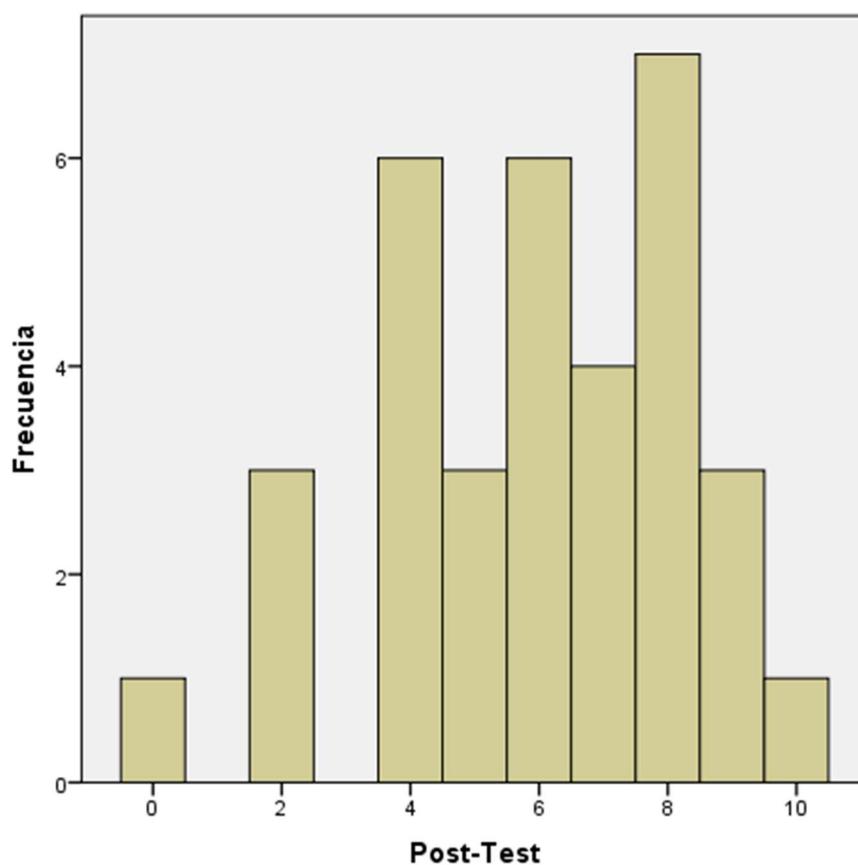


Figura 5-4. Histograma de calificaciones del examen post-test.

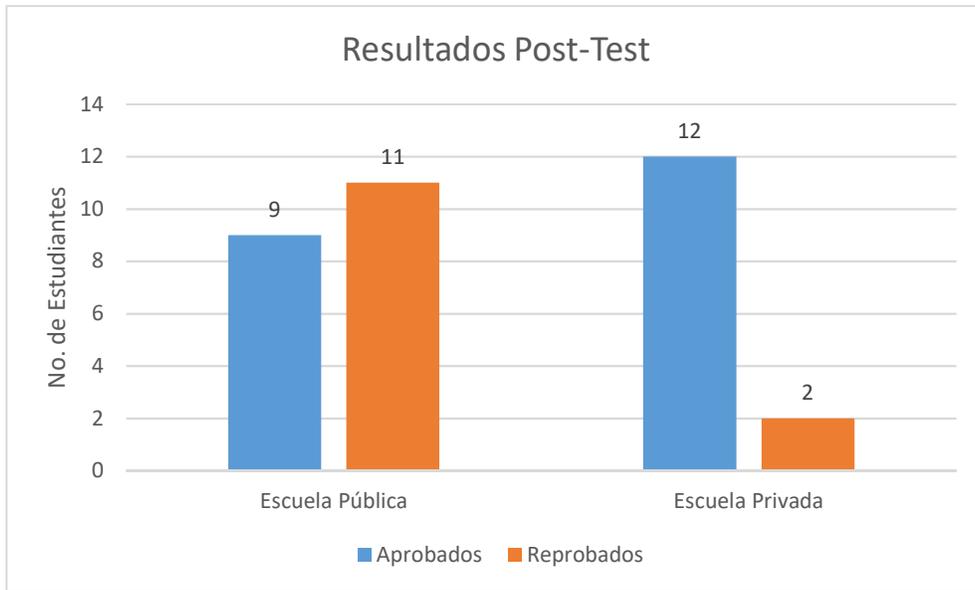


Figura 5-5. Estudiantes aprobados y reprobados del examen post-test.

5.2.3. Resultados comparativos pre-test y post-test

Al tener los resultados de pre-test y post-test definidos fueron graficados para hacer una comparativa buscando el reflejo de que aquellos que utilizaron la aplicación tuvieran un mejor desempeño que los que no.

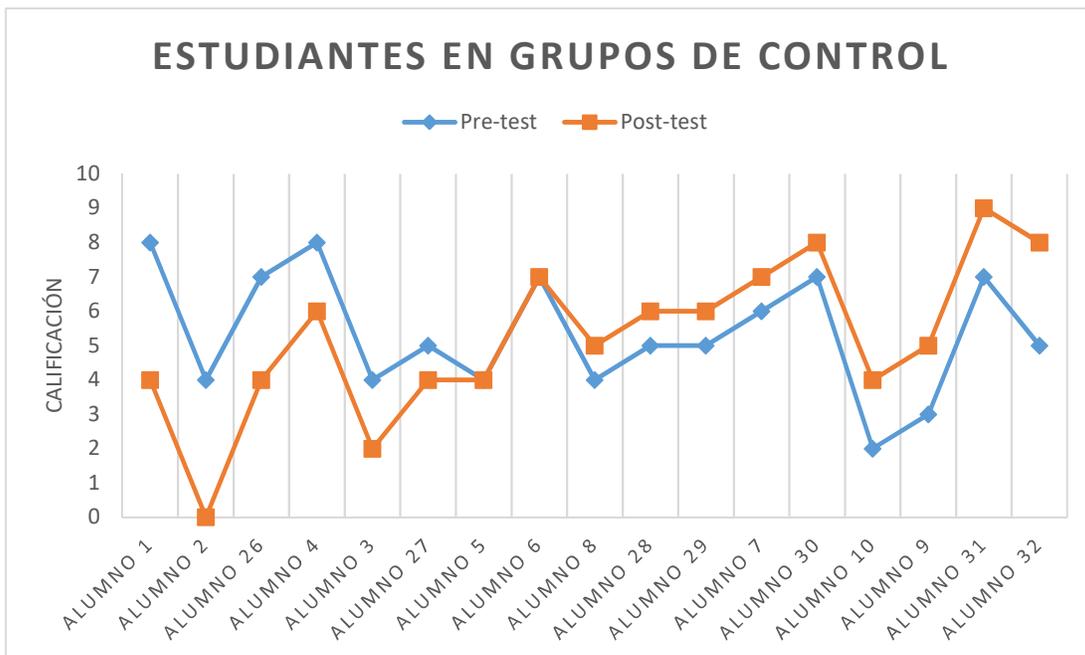


Figura 5-6. Relación de calificaciones pre-test y post-test en grupos de control.

Afortunadamente sí es apreciable el impacto positivo de quienes utilizaron *Patrony* al examinar la Figura 5-6 respecto a la Figura 5-7. Los estudiantes más sobresalientes del grupo de control tuvieron una diferencia de dos puntos más, en cambio, los estudiantes del grupo experimental mejoraron de tres a cinco puntos más.

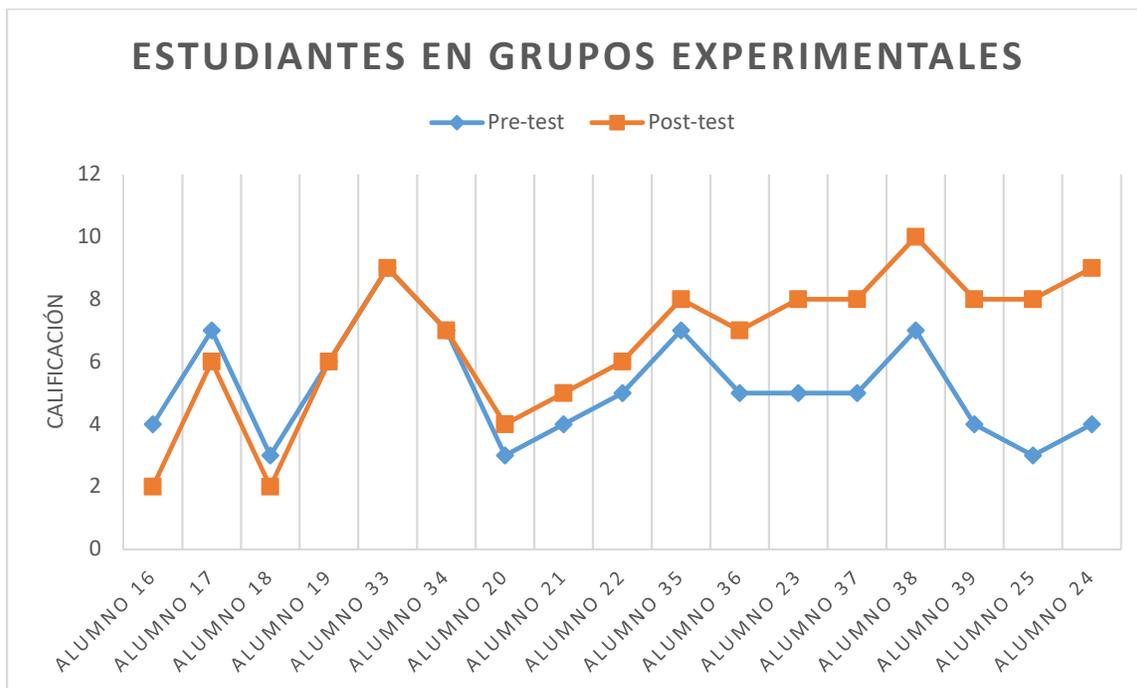


Figura 5-7. Relación de calificaciones pre-test y post-test en grupos experimentales.

Las gráficas están ordenadas de menor a mayor ganancia relativa por estudiante, explicada a continuación.

5.2.4. Ganancia relativa

Con los números de pre-test y post-test obtenidos recibimos una sugerencia por parte de la Dra. Julieta Noguez durante mi estancia de investigación en Ciudad de México para calcular las ganancias relativas de los estudiantes mediante la fórmula presentada en la Ecuación 1, semejante a (Hake, 1998).

$$g_i = \frac{Post_i - Pre_i}{100 - Pre_i}$$

Ecuación 1. Ganancia relativa del estudiante.

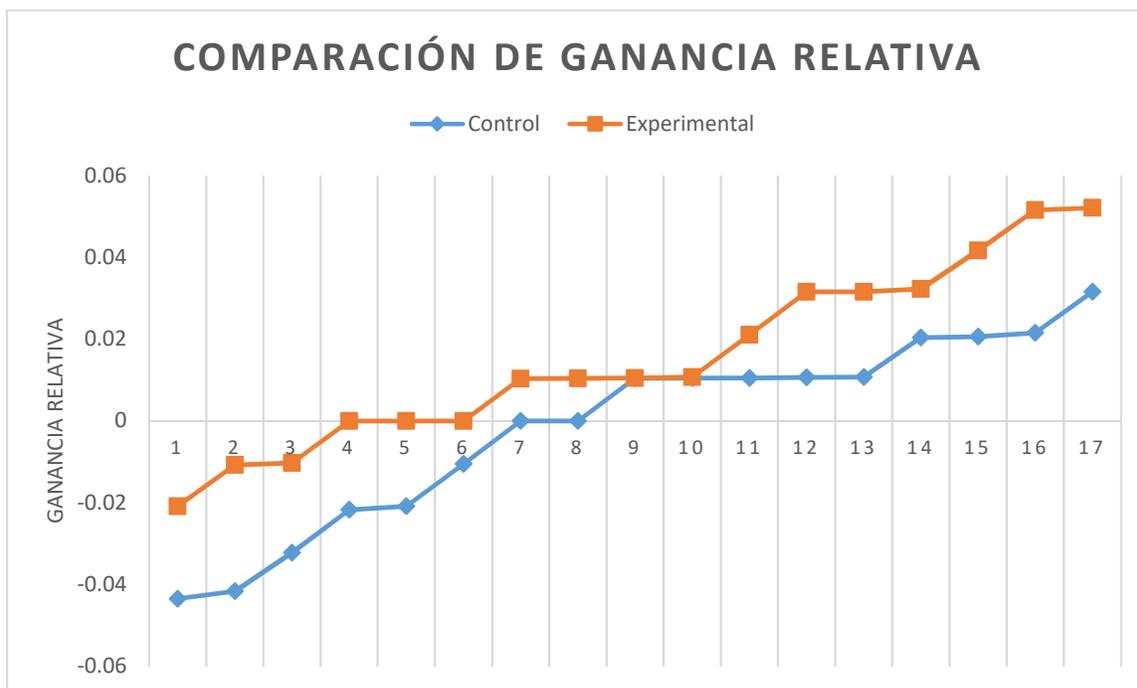


Figura 5-8. Comparación de ganancia relativa de los estudiantes.

Al calcular las ganancias relativas y graficarlas, como se muestra en la Figura 5-8, se puede visualizar una constante mayor ganancia en los estudiantes del grupo experimental. Esto significa que los alumnos que utilizaron *Patrony* tienen un mayor grado de habilidad al momento de reconocer patrones.

5.2.5. Datos recopilados por *Patrony*

La aplicación almacena para cada usuario la puntuación, errores y tiempo de las actividades que completa. A continuación, se presentan las gráficas (Figura 5-10, Figura 5-11, Figura 5-11) con los promedios de las variables cognitivas mencionadas de los 17 alumnos correspondientes a los grupos experimentales considerados. La puntuación máxima fue determinada por nosotros al resolver las actividades sin ningún error. Los errores esperados fueron establecidos de acuerdo nuestro punto de vista y corroborada con el docente. Asimismo, el tiempo máximo se asignó bajo nuestro mejor esfuerzo.

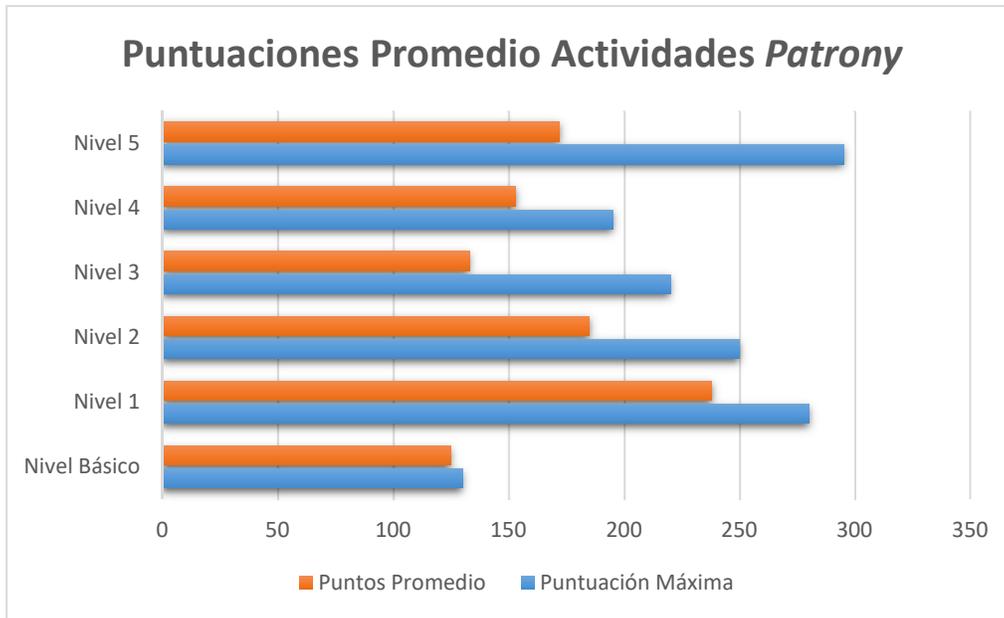


Figura 5-9. Puntuaciones promedio por actividad de *Patrony*.

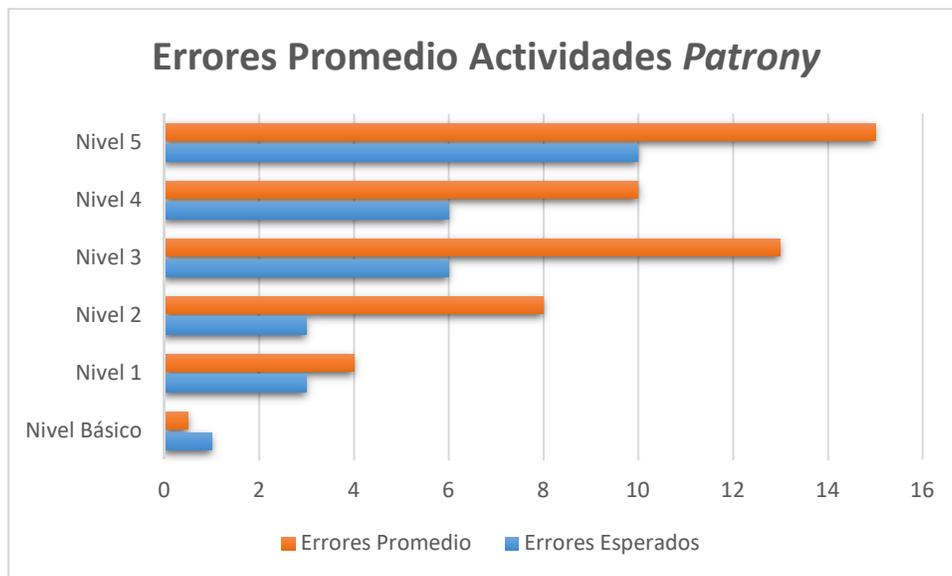


Figura 5-10. Errores promedio por actividad de *Patrony*.

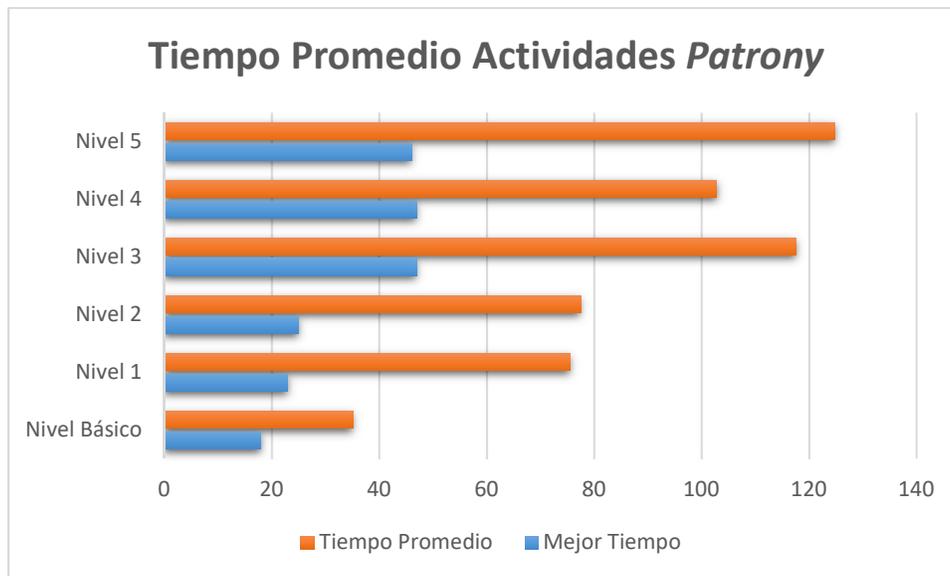


Figura 5-11. Tiempo promedio por actividad en *Patrony*.

Los datos aquí presentados pueden ser útiles para mejorar aspectos de la aplicación como la dificultad de las actividades, un mecanismo anti ráfagas de respuestas aleatorias y más estímulos durante las actividades, que derivan de la conducta presentada por los estudiantes durante los experimentos.

5.3. Conclusiones

De acuerdo con los números, se aprecia una mejora sustancial en los exámenes post-test de los grupos experimentales contra los grupos de control, sin embargo, algunos alumnos presentaron calificaciones que apuntan directamente a aspectos como falta de seriedad, atención, motivación, desinterés, dificultad para detectar patrones, estado de ánimo etc. De la misma manera, estos aspectos pueden ser los causantes de que algunos alumnos tuvieran un desempeño más pobre en el examen post-test que en el pre-test.

Por otra parte, se observó que en la escuela privada los estudiantes se familiarizaron con la aplicación y la finalizaron rápidamente, utilizando lo que restaban de sus 15 minutos de uso para mejorar sus puntuaciones, contrario a la escuela pública donde se empleó más tiempo en la comprensión de la interacción con la aplicación y solo unos pocos llegaron a intentar mejorar sus récords.

Otra manera de analizar los datos fue a través del cálculo de la ganancia relativa, la cual mostró que los estudiantes del grupo experimental siempre presentar un mayor índice.

No se realizó un análisis a los datos recopilados, sin embargo, es probable que sirvan de ayuda para aspectos como: detección de actividades muy complicadas o sencillas, mejoramiento al sistema de puntuación, inclusión de nuevas dinámicas para reducir o limitar las respuestas aleatorias, etc.

Capítulo 6

6. Conclusiones y trabajo futuro

Este capítulo expone las conclusiones de esta tesis y el trabajo futuro propuesto.

6.1. Conclusiones

Parece que el recorrido por la implementación del Pensamiento Computacional en el plan educativo mexicano aún es largo. Sin embargo, las habilidades ahí están, inmiscuidas en muchos temas. *Patrony* añade una alternativa enfocada al desarrollo del reconocimiento de patrones en niños de primaria que no está influenciada por la impartición directa de un docente que pudiese empeorar o mejorar la asimilación del conocimiento por los alumnos, siendo así un recurso neutral para ellos.

La consideración de la aplicación por parte de los estudiantes como un juego y no como material educativo tiene un papel importante. Esto motivaba a apegarse a la superación de niveles que propiciaban una mejoría en su habilidad de reconocer patrones.

Es notorio que el empleo de tecnología en las aulas tiende a ser interesante para los alumnos; los hace salir de lo rutinario de la educación tradicional y les da la oportunidad de hacer uso de sus habilidades interactuando con los dispositivos electrónicos.

Se realizó la evaluación de la aplicación con 34 alumnos de primaria, mediante el instrumento pre-test/post-test. El grupo experimental mostró un desempeño superior que el grupo de control.

Los resultados de este trabajo se ven positivos y a la vez representan un área de oportunidad potencial. Existen muchos aspectos que pueden enriquecer el objetivo de la herramienta e incluso ampliarlo, tanto cualitativa como cuantitativamente.

Finalmente, los resultados presentados en la sección de Experimentos y resultados (Capítulo 5) permiten concluir que la Hipótesis planteada para este trabajo es aceptada.

6.2. Trabajo futuro

Enseguida se listan los aspectos a considerar para la continuación de este proyecto:

- Incluir actividades con operaciones aritméticas sobre números enteros y/o fraccionarios.
- Ampliar la variedad de actividades utilizando otras asignaturas.
- Aplicar lógica difusa para determinar el desempeño del usuario y adaptar las actividades a su capacidad.
- Aplicar pruebas de motivación y usabilidad al grupo experimental.

Bibliografía

- BBC. (2019). Introduction to Computational Thinking. Recuperado el 27 de mayo de 2019, de Bitesize website: <https://www.bbc.com/bitesize/guides/zp92mp3/revision/1>
- Bryan, A., & Volchenkova, K. N. (2016). BLENDED LEARNING: DEFINITION, MODELS, IMPLICATIONS FOR HIGHER EDUCATION. *Bulletin of the South Ural State University series "Education. Education Sciences"*, 8(2), 24–30.
<https://doi.org/10.14529/ped160204>
- Calderon, A. C., Crick, T., & Tryfona, C. (2015). Developing computational thinking through pattern recognition in early years education. *Proceedings of the 2015 British HCI Conference on - British HCI '15*, 259–260.
<https://doi.org/10.1145/2783446.2783600>
- Chou, Y. (2014). *Actionable Gamification: Beyond Points, Badges, and Leaderboards*.
- Clark, R. C., & Mayer, R. E. (2016). *E-learning and the science of instruction: Proven guidelines for consumers and designers of multimedia learning*. John Wiley & Sons.
- Code.org. (2019). About Us. Recuperado el 24 de mayo de 2019, de <https://code.org/international/about>
- Crompton, H. (2013). A Historical Overview of M-Learning. En *Handbook of Mobile Learning*. <https://doi.org/10.4324/9780203118764.ch1>
- Cuantrix. (2019). Cuantrix. Recuperado el 9 de mayo de 2019, de <https://cuantrix.mx/>
- Disney. (2019). Hour of Code. Recuperado el 15 de mayo de 2019, de <https://partners.disney.com/hour-of-code>
- Duolingo. (2019). Duolingo for Schools. Recuperado el 12 de septiembre de 2018, de <https://schools.duolingo.com/>
- Encyclopaedia Britannica. (2018). Pattern recognition. En *Encyclopaedia Britannica*. Recuperado de <https://www.britannica.com/technology/pattern-recognition-computer-science>

- Facebook Inc. (2019a). React Docs. Recuperado el 28 de mayo de 2019, de <https://es.reactjs.org/docs/thinking-in-react.html>
- Facebook Inc. (2019b). React Native. Recuperado el 20 de mayo de 2019, de <https://facebook.github.io/react-native/>
- Facebook Inc. (2019c). React Native Docs. Recuperado el 20 de mayo de 2019, de <https://facebook.github.io/react-native/docs/javascript-environment>
- Gerónimo-Castillo, G., & Rocha-Trejo, E. H. (2014). Edumóvil: incorporando la tecnología móvil en la educación primaria. *RIED. Revista Iberoamericana de Educación a Distancia*, 10(1), 63–71. <https://doi.org/10.5944/ried.1.10.1014>
- GmbH, L. (2019). Our Apps. Recuperado el 12 de julio de 2019, de <https://lotum.com/en/our-apps/>
- Google LLC. (2019). Computational Thinking for Educators. Recuperado el 27 de mayo de 2019, de <https://computationalthinkingcourse.withgoogle.com/unit>
- Grover, S., & Pea, R. (2013). Computational Thinking in K-12: A Review of the State of the Field. *Educational Researcher*, 42(1), 38–43. <https://doi.org/10.3102/0013189X12463051>
- Hake, R. R. (1998). Interactive-engagement versus traditional methods: A six-thousand-student survey of mechanics test data for introductory physics courses. *American Journal of Physics*, 66(1), 64–74. <https://doi.org/10.1119/1.18809>
- Hsu, T.-C., Chang, S.-C., & Hung, Y.-T. (2018). How to learn and how to teach computational thinking: Suggestions based on a review of the literature. *Computers & Education*, 126, 296–310. <https://doi.org/10.1016/J.COMPEDU.2018.07.004>
- Hunicke, R., Leblanc, M., & Zubek, R. (2004). MDA: A Formal Approach to Game Design and Game Research. Recuperado de https://www.researchgate.net/publication/228884866_MDA_A_Formal_Approach_to_Game_Design_and_Game_Research
- INEGI. (2018). *Encuesta Nacional sobre la Disponibilidad y uso de Tecnologías de la*

- Información en los Hogares 2017*. Recuperado de https://www.inegi.org.mx/contenidos/saladeprensa/boletines/2018/otrtemecon/endutih2018_02.pdf
- Instituto Tecnológico de Monterrey. (2016). Mati-Tec. Recuperado el 26 de julio de 2019, de <http://www.cem.itesm.mx/cms/ecsh/index.php/humanidadesdigitales/mati-tec.html>
- ISTE, & CSTA. (2011). *Operational Definition of Computational Thinking for K-12 Education*. Recuperado de <https://id.iste.org/docs/ct-documents/computational-thinking-operational-definition-flyer.pdf>
- Jalote, P. (2008). *A Concise Introduction to Software Engineering*. <https://doi.org/10.1007/978-1-84800-302-6>
- King.com LTD. (2019). Candy Crush. Recuperado el 8 de julio de 2019, de <https://king.com/es/game/candycrush>
- Krasnova, T. (2015). A Paradigm Shift: Blended Learning Integration in Russian Higher Education. *Procedia - Social and Behavioral Sciences*, 166, 399–403. <https://doi.org/10.1016/j.sbspro.2014.12.543>
- MIT Media Lab. (2019). Scratch. Recuperado el 20 de mayo de 2019, de <https://scratch.mit.edu/>
- React Native Community. (2019). React Navigation Docs. Recuperado el 28 de mayo de 2019, de <https://reactnavigation.org/docs/en/getting-started.html>
- Roscoe, J. F., Fearn, S., & Posey, E. (2014). Teaching Computational Thinking by Playing Games and Building Robots. *2014 International Conference on Interactive Technologies and Games*, 9–12. <https://doi.org/10.1109/iTAG.2014.15>
- Ruiz, J. (2013). Acerca del Surgimiento del Reconocimiento de Patrones en Cuba. *Revista Cubana de Ciencias Informáticas*, 169–192. Recuperado de http://scielo.sld.cu/scielo.php?script=sci_arttext&pid=S2227-18992013000200007
- Seaborn, K., & Fels, D. I. (2015). Gamification in theory and action: A survey. *International Journal of Human Computer Studies*, 74, 14–31.

<https://doi.org/10.1016/j.ijhcs.2014.09.006>

- Suárez-Suárez, N. E., & Custodio-Najar, J. (2014). Evolución de las tecnologías de información y comunicación en el proceso de enseñanza-aprendizaje. *Revista Vínculos*, 11(1), 209–220. <https://doi.org/10.14483/2322939X.8028>
- Wang, R. (2011). Demystifying Enterprise Gamification for Business. Recuperado de <https://www.constellationr.com/research/demystifying-enterprise-gamification-business>
- Wing, J. M. (2006). Computational thinking. *Communications of the ACM*, 49(3), 33. <https://doi.org/10.1145/1118178.1118215>
- Wing, J. M. (2008). Computational thinking and thinking about computing. *IPDPS Miami 2008 - Proceedings of the 22nd IEEE International Parallel and Distributed Processing Symposium, Program and CD-ROM*, 366, 3717–3725. <https://doi.org/10.1109/IPDPS.2008.4536091>
- Wing, J. M. (2010). Computational Thinking: What and Why? *The Link - The Magazine of the Carnegie Mellon University School of Computer Science*, (March 2006), 1–6. Recuperado de <https://www.cs.cmu.edu/~CompThink/resources/TheLinkWing.pdf>
- Wing, J. M. (2014). Computational thinking benefits society. *Journal of Computing Sciences in Colleges*. <https://doi.org/10.1145/1227504.1227378>
- Yadav, A., Hong, H., & Stephenson, C. (2016). Computational Thinking for All: Pedagogical Approaches to Embedding 21st Century Problem Solving in K-12 Classrooms. *TechTrends*, 60(6). <https://doi.org/10.1007/s11528-016-0087-7>
- Yadav, A., Mayfield, C., Zhou, N., Hambruch, S., & Korb, J. T. (2014). Computational Thinking in Elementary and Secondary Teacher Education. *ACM Transactions on Computing Education*, 14(1), 1–16. <https://doi.org/10.1145/2576872>
- Zapotecatl, J. L. (2018). *Introducción al pensamiento computacional: conceptos básicos para todos* (1a ed.). Recuperado de <http://www.pensamientocomputacional.org/Files/LibroPC.pdf>

Zatarain-Cabada, R., Barrón-Estrada, M. L., Ibañez-Espiga, M. B., & Uriarte-Portillo, A. (2018). Cuerpos y Planos Geométricos usando Realidad Aumentada y Computación Afectiva. *Research in Computing Science*, 147(8), 203–213.

Zatarain-Cabada, R., Barrón-Estrada, M. L., & Ríos-Félix, J. M. (2017). *Affective Learning System for Algorithmic Logic Applying Gamification*. https://doi.org/10.1007/978-3-319-62428-0_44

Anexos

A. Examen Pre-Test



Examen Pretest Reconocimiento de Patrones
 Instrucciones: En la hoja de respuestas rellene de manera completa el círculo que corresponda a la opción que completa la sucesión de figuras.

1. ?
 A) B) C) D)

2. ?
 A) B) C) D)

3. ?
 A) B) C) D)

4. ?
 A) B) C) D)

5. ?
 A) B) C) D)

6. ?
 A) B) C) D)

7. ?
 A) B) C) D)



Examen Pretest Reconocimiento de Patrones

8. ?
 A) B) C) D)

9. ?
 A) B) C) D)

10. ?
 A) B) C) D)

¿Cuáles son los términos faltantes de las siguientes sucesiones?

- | | | |
|------------------------------------|-----------------|----------------|
| 11. __, 3, 6, __, 15, 21, __ | A) 1, 9, 27 | B) 1, 10, 28 |
| | C) 0, 10, 30 | D) 0, 9, 27 |
| 12. 3, 6, __, 24, 48, __, __, 384 | A) 12, 86, 192 | B) 12, 69, 129 |
| | C) 12, 96, 182 | D) 12, 96, 192 |
| 13. __, 86, 73, __, 47, 34, 21, __ | A) 99, 60, 8 | B) 99, 59, 10 |
| | C) 99, 50, 8 | D) 98, 60, 9 |
| 14. 256, __, __, 32, __, 8, 4 | A) 138, 74, 16 | B) 128, 64, 18 |
| | C) 128, 64, 16 | D) 123, 64, 18 |
| 15. 2916, 972, __, __, 36, 12, __ | A) 75, 60, 50 | B) 70, 65, 65 |
| | C) 324, 108, 12 | D) 75, 75, 60 |

B. Examen Post-Test



Examen PostTest Reconocimiento de Patrones

Instrucciones: En la hoja de respuestas rellene de manera completa el círculo que corresponda a la opción que completa la sucesión de figuras.

1.

A) B) C) D)
2.

A) B) C) D)
3.

A) B) C) D)
4.

A) B) C) D)
5.

A) B) C) D)
6.

A) B) C) D)
7.

A) B) C) D)



Examen PostTest Reconocimiento de Patrones

8.

A) B) C) D)
9.

A) B) C) D)
10.

A) B) C) D)