

Vol. 20, Núm. 3, 2018

Reconocimiento afectivo y gamificación aplicados al aprendizaje de Lógica algorítmica y programación

Affective Recognition and Gamification Applied to Learning Algorithmic Logic and Programming

Ramón Zatarain Cabada (*) rzc777@hotmail.com

(*) Instituto Tecnológico de Culiacán
(Recibido: 11 de octubre de 2016; Aceptado para su publicación: 7 de febrero de 2017)

Cómo citar: Zatarain, R. (2018). Reconocimiento afectivo y gamificación aplicados al aprendizaje de Lógica algorítmica y programación. *Revista Electrónica de Investigación Educativa*, 20(3), 115-125. <https://doi.org/10.24320/redie.2018.20.3.1636>

Resumen

En este trabajo se presenta un ambiente de aprendizaje que usa técnicas de reconocimiento afectivo con gamificación, dirigido a la enseñanza de Lógica algorítmica y programación. Dicho ambiente fue evaluado y contrastado con estudiantes de Ingeniería. El método consistió en evaluar el aprendizaje de la Lógica algorítmica usando técnicas tradicionales contra aprendizaje con reconocimiento automático de emociones y manejo motivacional usando gamificación. Se realizaron pruebas y se aplicaron encuestas a 42 estudiantes, quienes fueron divididos en dos grupos y se utilizaron dos configuraciones distintas del sistema para evaluar las técnicas de gamificación implementadas. Los resultados demostraron que el aprendizaje del estudiante es estadísticamente mejor si se toma en cuenta el estado afectivo del estudiante y si éste es motivado por medio de la gamificación.

Palabras clave: Programa informático didáctico, aprendizaje asistido, gamificación, ambiente educacional.

Abstract

This paper presents a learning environment that uses affective recognition techniques and gamification to teach algorithmic logic and programming. This environment was evaluated and contrasted with engineering students. The method consisted in assessing student learning of algorithmic logic using traditional techniques compared to learning with automatic recognition of emotions and motivational management using gamification. Tests and surveys were conducted with 42 students, who were divided into two groups, and two different system configurations were used to assess the gamification techniques implemented. The results showed that learning is statistically better if students' affective state is taken into account, and if students are motivated through gamification.

Keywords: Educational software, computer assisted, gamification, educational environment.

I. Introducción

Aprender a programar sigue siendo una de las tareas más demandantes para la mayoría de los estudiantes. En la actualidad, muchas carreras de nivel bachillerato y licenciatura –e incluso aquellas ajenas a la ciencia de la computación– contienen en su plan de estudio materias de programación de computadoras, tratando así de contribuir al perfil del estudiante al proporcionarle competencias para seleccionar y desarrollar aplicaciones que solucionen los problemas de su entorno profesional.

La problemática actual reside en que los estudiantes de cursos de programación manifiestan desinterés por la materia, generando así problemas de aprendizaje y, por ende, elevadas tasas de reprobación. Para mejorar el método de enseñanza y aprendizaje de Lógica algorítmica y programación de computadoras se requiere implementar técnicas modernas de motivación en estudiantes acostumbrados a utilizar las nuevas tecnologías, entre ellas la de juegos por computadora. También se necesita conocer y entender cuál es el estado emocional del alumno, cuándo adquiere nuevos conocimientos en un área de alto nivel intelectual, cómo es el desarrollo de *software* por medio de un lenguaje de programación.

El estado afectivo o emocional de un estudiante influye en su desempeño durante el proceso de aprendizaje. Un estudiante fomenta su memorización y comprensión cuando se siente comprometido y motivado (Landowska, 2014). El aprendizaje afectivo es un término usado para describir los fenómenos de influencia de estados emocionales en la cognición humana y el aprendizaje.

Existe un principio básico que ha sido difundido ampliamente acerca del efecto de las emociones en el aprendizaje: “Los niveles de excitación altos son mejores que los bajos” (Landowska, 2014), por lo que es preferible que el estudiante se encuentre disgustado a que esté aburrido. Sin embargo, las emociones de alta y baja excitación no son suficientes para los procesos educativos; el aprendizaje se ve favorecido también por la concentración, el compromiso y el estado mental (Picard y Klein, 2002).

Por otra parte, la gamificación (también llamada ludificación) es un término que se utiliza para describir aquellas características de un sistema interactivo que tienen como objetivo motivar y comprometer a los usuarios finales a través del uso y la mecánica de estímulos comúnmente incluidos en los juegos (Deterding, Sicart, Nacke, O’Hara y Dixon, 2011). Algunos elementos de gamificación utilizados son: puntos, recompensas, progresión, insignias y tablas de posiciones.

Los objetivos principales de esta investigación fueron desarrollar un sitio web que funja como tutor para la enseñanza de Lógica algorítmica y programación, reconocer las emociones que los estudiantes están presentando en tiempo real, integrar técnicas de gamificación para motivar a los estudiantes y validar si el sistema propuesto contribuye a mejorar el aprendizaje en los estudiantes.

1.1. Software de aprendizaje de Programación

En la actualidad existe una amplia gama de herramientas que apoyan a los estudiantes en tareas de aprendizaje de temas relacionados con las ciencias de la computación. Dentro de este campo del conocimiento uno de los más influyentes, y que está relacionado con el aprendizaje de la programación de computadoras, es lo que se conoce como *pensamiento computacional* (Loannidou, Bennett, Repenning, Koh y Basawapatna, 2011), que es en realidad una nueva forma abstracta de solucionar problemas por medio de un pensamiento algorítmico.

Dentro de esta nueva área de la computación existen herramientas que se emplean para desarrollar un pensamiento computacional y solucionar problemas por medio de programación de computadoras. Estas herramientas de aprendizaje pueden ser sistemas e-learning o CAI (*Computer Assisted Instruction*), Sistemas Tutores Inteligentes (STI) y los llamados Juegos serios (*serious games*). A continuación se describen los trabajos desarrollados en el campo de aprendizaje de programación de computadoras por medio de software.

El **e-learning** se define como educación transmitida por medio de un dispositivo digital (una computadora o dispositivo móvil) destinado a apoyar el aprendizaje (Clark y Mayer, 2016). Las herramientas *e-learning* de apoyo para el aprendizaje de la programación pretenden facilitar el aprendizaje haciendo uso de marcos de trabajo preestablecidos, que facilitan la implementación de los algoritmos y programas a los estudiantes. Un ejemplo de estos sistemas es *Greenfoot* (Utting, Cooper, Kölling, Maloney y Resnick,

2010), ambiente de desarrollo integrado educativo dirigido a la enseñanza y aprendizaje de programación. Otro ejemplo es *App Inventor* (Crawford y Dominguez, 2013), herramienta de programación visual basada en bloques y usada para diseñar y construir aplicaciones móviles totalmente funcionales para el sistema operativo Android.

Los **STI** son sistemas de instrucción basados en computadoras que cuentan con modelos de contenidos que especifican qué enseñar, así como las estrategias de enseñanza que describen cómo se debe enseñar (Sharma, Ghorpade, Sahni y Saluja, 2014). Algunos ejemplos de STI de programación son STI PHP (Weragama, 2013) y CSTutor (Buchanan y LaViola Jr, 2012). STI PHP se desarrolló para enseñar a los estudiantes a programar en el lenguaje de PHP. Por otra parte, CSTutor ayuda en el aprendizaje del lenguaje C#, incorporando un modelo de aprendizaje donde las actividades de enseñanza y aprendizaje se diseñan en un entorno de la vida real.

Los **Juegos serios** son aquellos que además de divertir enseñan, de esta forma atraen y seducen a los estudiantes más fácilmente que otras herramientas de enseñanza (Charsky, 2010). Hay muchos beneficios que se asocian con los juegos, incluyendo los aspectos de motivación inherentes a los mismos, el razonamiento lógico complejo, la interactividad y la experiencia en desarrollo que pueden fomentar el aprendizaje (Kazimoglu, Kiernan, Bacon y MacKinnon, 2012). A partir de esta idea surge el término “gamificación”.

Dos de los juegos serios de mayor interés para este trabajo son *Program your robot* (Kazimoglu et al., 2012) y *Cargo-Bot* (Tessler, Beth y Lin, 2013); el primero es un juego que permite a los estudiantes obtener habilidades de pensamiento computacional como son: construir algoritmos, depurar errores y ejecutar el algoritmo; y *Cargo-Bot*, por su parte, plantea una nueva metodología de enseñanza de recursión, donde los estudiantes pueden practicar resolviendo problemas cada vez más complejos.

Una de las principales desventajas de estas herramientas de software de aprendizaje para lenguajes de programación es que no contemplan el uso del estado afectivo o emocional de los estudiantes para la personalización de los contenidos. De acuerdo con Immordino-Yang y Damasio (2007), “las emociones actúan como un ‘timón’ en la transferencia de conocimientos y habilidades, y juegan un papel importante y necesario en la toma de decisiones y en la creatividad”. Por otra parte, las emociones positivas (por ejemplo: interesado, enganchado, motivado) influyen en el aprendizaje al afectar la atención de los estudiantes, la motivación, el uso de estrategias de aprendizaje y la autorregulación del aprendizaje (Pekrun, 2014). Además, las experiencias emocionales positivas juegan un rol primordial en el desempeño académico y producen un impacto considerable en el éxito de los estudiantes (Pekrun, Elliot y Maier, 2009).

EasyLogic es una herramienta que combina las técnicas previamente descritas (herramientas *e-learning* para enseñanza de programación, sistemas tutores inteligentes y gamificación) que permite a un estudiante diseñar e implementar algoritmos de una forma simple, pero poderosa. La principal contribución de esta herramienta es la integración de la detección de emociones y su respuesta en la aplicación de técnicas motivacionales dentro de un sistema que enseña a un estudiante el difícil proceso de construcción de algoritmos y programas.

EasyLogic se conforma de 3 secciones principales: 1) Aprende. Cuenta con diversos cursos que incluyen una serie de ejercicios o lecciones, para cada una de las distintas estructuras de control que existen; 2) Imagina y crea. Permite a los estudiantes diseñar sus propios algoritmos y posteriormente ejecutarlos, y 3) Codifica. Ayuda a que los estudiantes programen directamente en JavaScript en lugar de usar los bloques gráficos.

Al igual que muchas herramientas de autor y de programación modernas, *EasyLogic* utiliza bloques gráficos. La interfaz del sistema se desarrolló utilizando la librería de Google Blockly (Fraser, 2013). La figura 1 muestra cómo se visualiza dicha interfaz, a la izquierda se muestra un listado de las clasificaciones de los bloques; cuando el usuario selecciona una clasificación se muestran todos los bloques disponibles para esa clasificación. El usuario podrá arrastrar y soltar los bloques, con la finalidad de crear sus propios algoritmos.

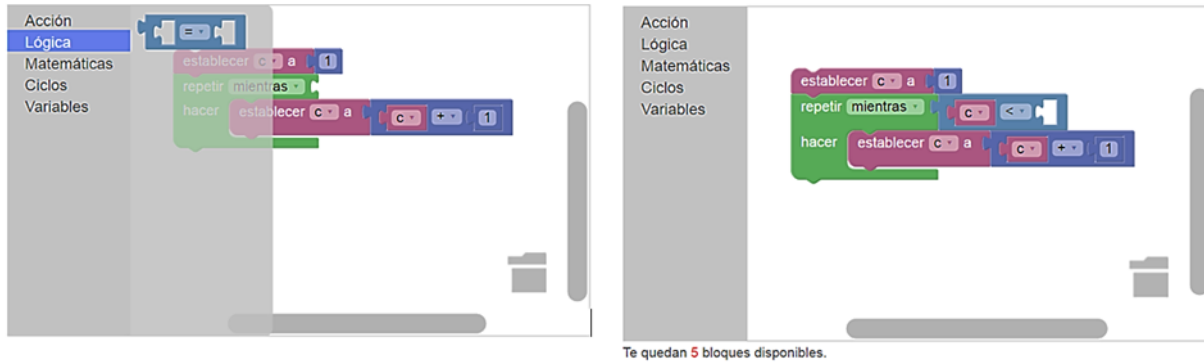


Figura 1. Interfaz gráfica para la edición de algoritmos

Una vez que el usuario procede a realizar un ejercicio, *EasyLogic* evalúa cada 30 segundos variables como: el tiempo invertido en el ejercicio, cantidad de ejecuciones y la emoción que más se ha detectado recientemente en el estudiante para determinar si éste necesita algún tipo de ayuda. Las ayudas consisten en ventanas emergentes que se muestran automáticamente cuando se detecta que el estudiante las necesita y se dividen en tres tipos: iniciales, informativas y motivacionales.

La figura 2 muestra la interfaz principal de *EasyLogic* para resolver un ejercicio. En la imagen se aprecian números que señalan diferentes componentes de la interfaz: 1) Barra de opciones que redireccionan hacia otras ventanas, 2) Nombre del curso que se está realizando, 3) Nivel del ejercicio actual, 4) Clasificación de los bloques gráficos, 5) Área para crear el algoritmo por medio de los bloques gráficos, 6) Juego que se animará una vez ejecutado el algoritmo, 7) Cantidad de bloques disponibles y 8) Botón para ejecutar el algoritmo y otro para ver el código asociado al algoritmo creado.



Figura 2. Interfaz para la solución de ejercicios

Para acceder a los cursos los usuarios deben iniciar sesión utilizando su correo electrónico y contraseña. Una vez seleccionado el curso el sistema carga automáticamente el siguiente ejercicio aún sin resolver por el estudiante. Algunos ejercicios están asociados a ayudas iniciales; una vez cargado el ejercicio se muestra la ayuda inicial en caso de requerirse. Para no aburrir al estudiante con ayudas que ya ha leído anteriormente, se guarda información de todas aquellas ayudas que se le han mostrado al estudiante y se ha confirmado que la información ha sido leída.

Una vez cargado el ejercicio el sistema pasa a un estado de espera por un tiempo específico, para que el estudiante trate de resolver el ejercicio. Si el ejercicio fue resuelto, se calculan los puntos obtenidos en base al tiempo invertido en el ejercicio y a la cantidad de ejecuciones requeridas, en caso de obtener un nuevo trofeo éste es mostrado al estudiante. Por el contrario, si el ejercicio no ha sido resuelto se

analizan los datos del usuario (tiempo transcurrido, estado emocional y cantidad de ejecuciones) para verificar si necesita algún tipo de ayuda extra (informativa o motivacional). De ser así se obtiene de una base de datos una ayuda para el ejercicio actual (ver figura 3).

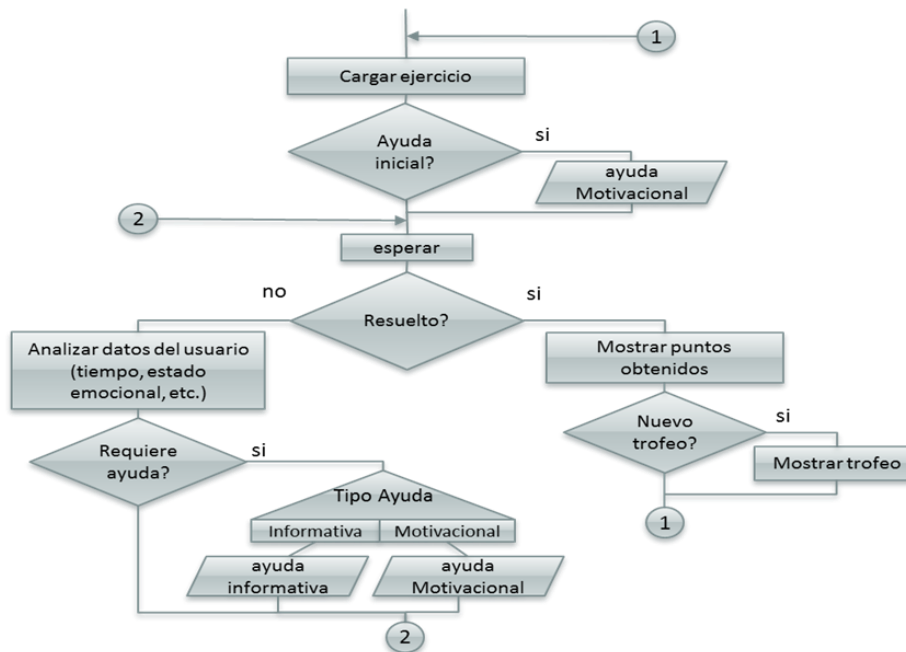


Figura 3. Diagrama del proceso de los cursos *EasyLogic*

Para el reconocimiento emocional se utilizó una Plataforma de Reconocimiento Multimodal de Emociones (PREMOC) (Zatarain, Barrón-Estrada y Muñoz, 2016), que consiste en un servicio web que puede utilizar imágenes faciales, audio, texto y opcionalmente señales electroencefalográficas (EEG), las cuales procesa para extraer sus características y alimentar una red neuronal que da como resultado una emoción asociada utilizando un integrador de lógica difusa. Cuando un usuario está realizando un ejercicio, *EasyLogic* envía una imagen facial del usuario a PREMOC cada 15 segundos y guarda la emoción que éste presentó, para ser procesada posteriormente y determinar si se deberá mostrar algún tipo de ayuda al usuario. *EasyLogic* sólo utiliza de PREMOC el reconocimiento de expresiones faciales.

Para la realización de los ejercicios, *EasyLogic* combina tres aspectos importantes: reconocimiento de afecto, uso de la herramienta y aplicación de técnicas de gamificación. La figura 4 muestra los resultados generados por un estudiante al momento de realizar el ejercicio 5 del curso “Estructura selectiva”. La parte Afectiva (A) evaluó el estado emocional del estudiante utilizando PREMOC. Se registró que el estudiante se encontraba aburrido, por lo que se procedió a mostrar una ayuda motivacional debido a que el usuario empleó mucho tiempo para resolver el ejercicio; a los 120 segundos se mostró una Ayuda Informativa. La Herramienta (B) mostró el juego a resolver y proporcionó los bloques adecuados para dar solución al problema. La parte de Gamificación (C), calculó un total de 73 puntos de un máximo de 100, debido a que el usuario tardó bastante tiempo (168 segundos) y además requirió de dos ejecuciones del algoritmo. Para este ejercicio no se obtuvo ningún trofeo nuevo, debido a que necesitaba 500 puntos para obtener el siguiente trofeo. Opcionalmente el estudiante tiene acceso a la tabla de posición que muestra el puntaje actual de todos los participantes.

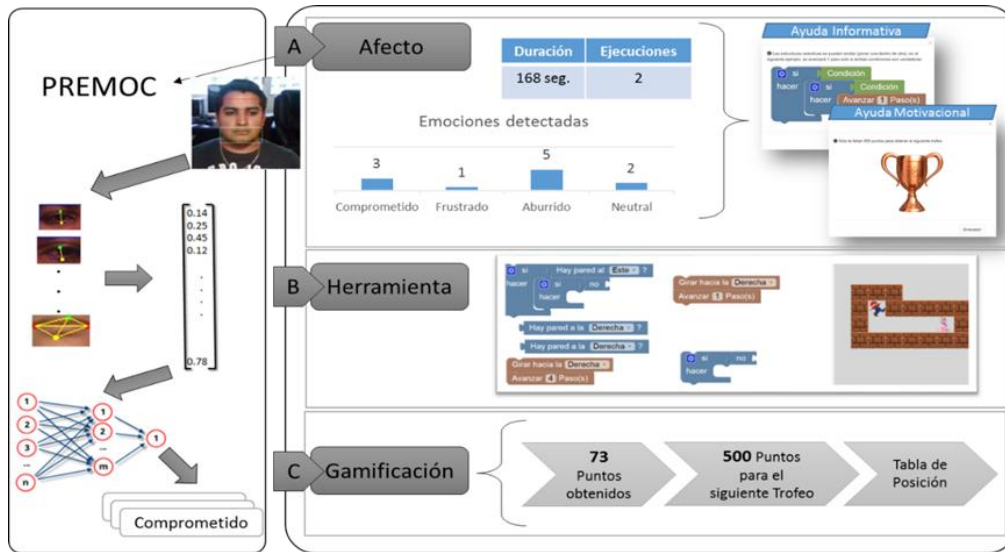


Figura 4. Uso de *EasyLogic* resolviendo el Ejercicio 5 del curso "Estructura Selectiva"

II. Método

Para evaluar el logro de los objetivos propuestos en esta investigación, el sistema fue aplicado con estudiantes del Instituto Tecnológico de Culiacán, Sinaloa (México). En la evaluación participaron estudiantes de la asignatura "Algoritmos y lenguajes de programación", quienes cursan la carrera de Ingeniería Industrial. Se seleccionaron 42 estudiantes y se formaron dos grupos de forma aleatoria (Grupo 1 y Grupo 2). El tipo de evaluación fue Pre-Test + Intervención + Post-Test.

En este diseño se mide el conocimiento de dos grupos; a uno (Grupo 1) se le proporcionó la herramienta *EasyLogic* para que aprendieran el material didáctico con ella, mientras el otro (Grupo 2) aprendió de la forma tradicional (en clase con un maestro). Para evaluar la herramienta *EasyLogic*, se efectuaron las actividades que se mencionan en la tabla I.

Tabla I. Actividades de Evaluación de *EasyLogic*

Etapa	Tarea	Tiempo
1	Explicación del motivo de la evaluación	10 minutos
2	Examen Pre-Test	40 minutos
3	Introducción al ambiente de <i>EasyLogic</i>	15 minutos
4	Utilización de <i>EasyLogic</i> por parte de los estudiantes	1 semana
5	Examen Post-Test	40 minutos

El Grupo 1 realizó las cinco etapas descritas en la tabla I y el Grupo 2 las actividades de las etapas 1, 2 y después tomó clases en su salón de forma tradicional. Los temas en ambos grupos fueron estructuras algorítmicas de control, con una duración de 5 horas de clase. Por último, al grupo 2 se le aplicó también el mismo examen (Post-Test) que al Grupo 1. Durante la evaluación con el Grupo 1 la herramienta *EasyLogic* monitoreaba constantemente el estado emocional de los estudiantes; se decidió utilizar la herramienta con dos configuraciones:

- 1) Con las características completas (detección emocional, aplicación de gamificación con trofeos y puntaje). 10 estudiantes trabajaron de esta manera.
- 2) Se desactivó la detección de afecto y la parte de gamificación, por lo que se omitieron los mensajes donde se mostraba la cantidad de puntos obtenidos y la alerta que mostraba los trofeos obtenidos (sólo se mostraron ayudas iniciales e informativas). 11 estudiantes utilizaron la herramienta con esta configuración.

Para esto, se establecieron tres hipótesis: la primera en relación con el desempeño académico (mayor aprendizaje) y la segunda y tercera en relación con la usabilidad y las ayudas de la herramienta *EasyLogic*.

H1. El aprendizaje del estudiante es estadísticamente mejor si se toma en cuenta el estado afectivo del estudiante y si éste es motivado por medio de técnicas de gamificación.

H2. La facilidad de uso de *EasyLogic* ayuda a mejorar la comprensión de la Lógica algorítmica.

H3. Las ayudas de *EasyLogic* ayudan a resolver más ejercicios.

III. Resultados

Al utilizar *EasyLogic* se guardó información concerniente a los ejercicios que los estudiantes estaban realizando, por ejemplo: el número total de ayudas mostradas a cada uno de los estudiantes, la cantidad de ejercicios que resolvieron, el tiempo promedio (segundos) que les tomó resolver cada ejercicio, y el tiempo máximo y mínimo que duraron resolviendo algún ejercicio durante el uso de la herramienta. La tabla II muestra los datos obtenidos durante el uso de la herramienta por parte de los estudiantes.

Tabla II. Datos obtenidos durante la evaluación

Alumno	Ayudas totales	Ejercicios Resueltos	Tiempo promedio (seg.)	Ejecuciones promedio	Tiempo máximo (seg.)	Tiempo mínimo (seg.)
1	22	16	90	2	270	16
2	15	13	87	2	124	31
3	34	14	286	9	632	40
4	15	11	91	1	149	26
5	20	16	104	2	413	16
6	29	16	124	3	306	40
7	13	11	70	1	137	20
8	18	14	91	3	225	24
9	15	11	165	4	323	60
10	19	15	77	2	241	18
11	23	20	256	6	1477	22
12	13	16	86	1	303	15
13	24	19	138	1	837	23
14	15	12	253	5	1847	35
15	15	13	134	3	883	21
16	17	12	148	2	476	34
17	24	13	99	1	203	42
18	18	20	98	1	547	13
19	29	19	294	4	1296	33
20	20	20	114	2	401	25
21	25	19	193	3	1442	19

Durante la evaluación de la herramienta *EasyLogic* se monitoreaba constantemente el estado emocional de los estudiantes. Los resultados en la tabla III muestran la suma total de las emociones detectadas por los 10 estudiantes que usaron la herramienta con la configuración 1.

Tabla III. Total de emociones detectadas durante la evaluación

Emoción	Total
Aburrido	80
Comprometido	164
Frustrado	79
Neutral	3204

Las evaluaciones fueron calificadas con un máximo de 100. Las evaluaciones Pre-Test y Post-Test del Grupo 1, que utilizó *EasyLogic*, resultó con un promedio de calificación de 82.19 y 83.57 en el examen post-test. Los resultados del Grupo 2, que aprendió de la forma tradicional, arrojaron un promedio de 84.28 y 82.95 en el examen post-test.

Con base en los resultados anteriores se observa que en ambos grupos el promedio de los exámenes es similar. Si comparamos los resultados del Post-Test de ambos grupos de prueba, podemos ver una diferencia de 0.62%, aunado a que el Grupo 2 obtuvo un promedio mayor en el examen Pre-Test que el Grupo 1, lo que significa que el Grupo 2 tiene una tendencia negativa en relación al promedio grupal, en comparación con el Grupo 1.

Comparando los resultados obtenidos por la herramienta, como se muestra en la figura 5, se observaron mejores resultados con los estudiantes que utilizaron la Configuración 1, en relación con el tiempo que les tomó resolver los ejercicios.

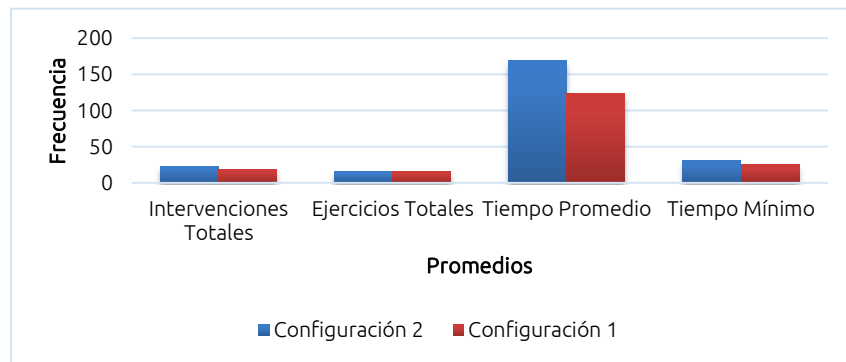


Figura 5. Comparación del Grupo 1 con base en las configuraciones

Sin embargo, no hubo una diferencia en el promedio del total de ejercicios resueltos por los estudiantes. Con base en estos resultados, el uso de gamificación que menciona la hipótesis no fue aquí probado; sin embargo, la figura 6 muestra una comparación de los resultados de las evaluaciones del Grupo 1, donde se puede observar una tendencia positiva en el incremento del aprendizaje en los estudiantes que utilizaron la configuración 1. En cambio, con la configuración 2 se muestra una tendencia negativa, pero que no es muy significativa. Este resultado puede ser considerado como la prueba de que el uso de gamificación en los estudiantes ayuda a mejorar el aprendizaje en los estudiantes.

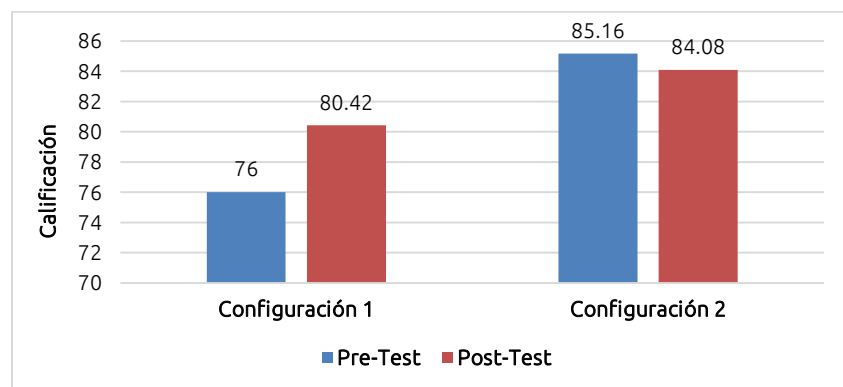


Figura 6. Comparación del Grupo 1 con base en las Evaluaciones

Sin embargo, los resultados de 4 diferentes pruebas estadísticas (ver tabla IV) no indican una diferencia significativa entre los resultados obtenidos en los exámenes Pre-Test y Post-Test debido a que el valor de la significancia obtenida en dichas pruebas fue mayor a 0.05. Por lo tanto, la hipótesis 1 (H1) que se relaciona con el desempeño académico del estudiante es rechazada.

Tabla IV. Resultados obtenidos en pruebas estadísticas

Prueba	Objetivo	Significancia
Kolmogorov-Smirnov	Evaluar normalidad entre resultados de las pruebas	0.149
Shapiro-Wilk	Evaluar normalidad entre resultados de las pruebas	0.262
Wilcoxon	Evaluar la media de las diferencias de las pruebas	0.856
Friedman	Análisis de varianza	0.782

Por otra parte, el hecho de que la herramienta realice una intervención dependiendo del estado emocional de un estudiante, en este caso a través de una ayuda visual, puede ser determinante para una mejor comprensión de los diversos temas que se tienen que abordar a lo largo de un curso escolar.

Al finalizar los ejercicios se encuestó a los sujetos de prueba del Grupo 1, se utilizó la escala tipo Likert para limitar las respuestas disponibles. La figura 7 muestra los resultados de dos de las preguntas más significativas de la encuesta (acerca de la usabilidad y las ayudas de la herramienta):

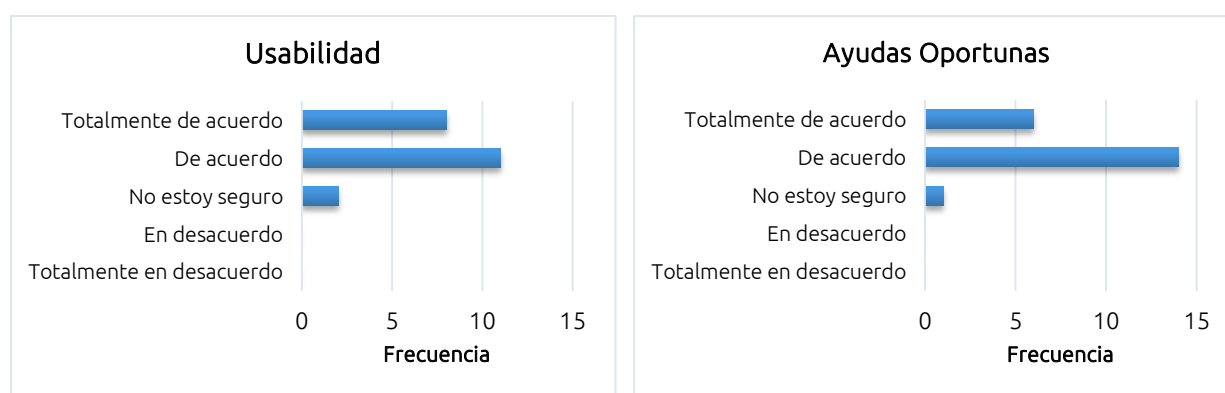


Figura 7. Resultados de 2 preguntas de las encuestas

La mayoría de los sujetos de prueba concordaron en que las ayudas mostradas por el sistema han sido oportunas y contribuyen de manera adecuada a la realización de los ejercicios. Otro dato interesante es que los usuarios piensan que el sistema es muy amigable y fácil de usar, incluso sin conocimientos previos de la herramienta. Esto prueba las hipótesis 2 y 3 (H2 y H3) con relación a que la usabilidad y las ayudas del sistema inciden en el aprendizaje de la Lógica algorítmica y aumentan el rendimiento de los estudiantes al resolver ejercicios.

IV. Conclusiones

En este trabajo se presenta *EasyLogic*, un ambiente afectivo para el aprendizaje de Lógica algorítmica y programación. La herramienta auxilia a los estudiantes en el proceso de aprendizaje utilizando técnicas de gamificación –además de monitorear el estado emocional del estudiante, con el fin de intervenir en caso de que el estudiante lo precise.

El desarrollo de este tipo de proyectos es muy complejo, por lo cual se optó por recurrir a software ya desarrollado y probado, como es el caso de *Blockly* (Fraser, 2013) para la interfaz gráfica de los algoritmos y *PREMOC* (Zatarain et al., 2016), para los algoritmos de inteligencia artificial de reconocimiento emocional.

Las pruebas realizadas, han demostrado que dos de las tres hipótesis establecidas en relación con la mejor comprensión de la Lógica algorítmica (H2) y el incremento en la solución de ejercicios (H3) fueron probadas. Esto nos lleva a concluir que es importante la detección del estado emocional del estudiante, sobre todo en casos en los que se encuentre en un estado emocional negativo (por ejemplo, aburrido), y confirma los resultados de investigaciones que establecen que el estado afectivo de un estudiante influye en su desempeño durante el proceso de aprendizaje, y mejora notablemente cuando se siente comprometido y motivado. La H1 fue rechazada debido a que se realizaron diversas pruebas estadísticas

no paramétricas para comparar los resultados obtenidos del Pre-Test y el Post-Test de ambos grupos de prueba, pero ninguna de ellas fue significativa. Este resultado se debe a la similitud de los promedios grupales de ambas pruebas y a la distinta complejidad manejada en los exámenes (en el Post-Test se incluyeron ejercicios con mayor complejidad).

Por otro lado, comprobamos que la gamificación puede convertirse en un método importante para motivar al estudiante y despertar su creatividad en la solución de problemas complejos. Esta conclusión está apoyada por estudios como los de Kazimoglu et al. (2012), quienes señalan el efecto positivo de los juegos en el fomento del aprendizaje.

Las pruebas realizadas han demostrado que la detección del estado emocional del estudiante puede influir favorablemente al ser utilizado para realizar una intervención en caso de que el estudiante lo necesite, ya sea reorientando la atención (cuando el estudiante esté aburrido) o mostrando ayudas informativas (cuando está frustrado. Sin embargo, los resultados obtenidos muestran una pequeña diferencia en la prolongación del uso de la herramienta implementando las técnicas de gamificación; se encontró una relación positiva con el tiempo que les tomó a los estudiantes resolver cada uno de los ejercicios, pero creemos que se deben realizar más pruebas (con un mayor número de estudiantes), que arrojen resultados más concluyentes.

Existen varias limitaciones en el estudio que necesitan ser abordadas en trabajos futuros. Una es que el tamaño de la muestra fue pequeño, dando a los resultados estadísticos menos fiabilidad; otra es que los estudiantes fueron seleccionados de una sola universidad (Instituto Tecnológico de Culiacán) y sólo de nivel licenciatura, por lo que los resultados no pueden generalizarse.

Algunas actividades para futuros estudios son realizar evaluaciones en otras escuelas y en otros niveles como preparatoria. También se requiere agregar cursos que aborden temas específicos, como condiciones múltiples, funciones y recursividad; completar más ejercicios para cada uno de los cursos, incrementando gradualmente la dificultad, para que el estudiante extienda su aprendizaje; y permitir una modalidad colaborativa, donde más de un estudiante colabore en la construcción del algoritmo/código, en tiempo real.

Referencias

Buchanan, S. y LaViola Jr., J. (2012). *CSTutor: a pen-based tutor for data structure visualization*. Actas del 43rd ACM Technical Symposium on Computer Science Education (pp. 565-570), Raleigh, NC. doi:10.1145/2157136.2157297

Charsky, D. (2010). From edutainment to serious games: a change in the use of game characteristics. *Games and Culture*, 5(2), 177-198. doi:10.1177/1555412009354727

Clark, R. C. y Mayer, R. E. (2016). *E-learning and the science of instruction: proven guidelines for consumers and designers of multimedia learning*. San Francisco, CA: John Wiley & Sons.

Crawford, S. y Domínguez, J. J. (October, 2013). *MIT app inventor: enabling personal mobile computing*. Conferencia presentada en el Programming for Mobile and Touch PRoMoTo 2013, Indianapolis, IN. Recuperado de [arXiv:1310.2830v2](https://arxiv.org/abs/1310.2830v2)

Deterding, S., Sicart, M., Nacke, L., O'Hara, K. y Dixon, D. (2011). *Gamification: using game design elements in non-game contexts*. Actas 2011 Annual Conference Extended Abstracts on Human Factors in Computing Systems, 2425-2428. Vancouver, Canadá. doi:10.1145/1979742.1979575

Fraser, N. (2013). Google blockly [a visual programming editor]. Recuperado de <https://neil.fraser.name/news/2013/12/31/>

Immordino-Yang, M. H. y Damasio, A. (2007). We feel, therefore we learn: the relevance of affective and social neuroscience to education. *Mind, Brain and Education*, 1(1), 3-10.

Kazimoglu, C., Kiernan, M., Bacon, L. y MacKinnon, L. (2012). A serious game for developing computational thinking and learning introductory computer programming. *Procedia-Social and Behavioral Sciences*, 47,

1991-1999. Recuperado de doi:10.1016/j.sbspro.2012.06.938

Landowska, A. (October, 2014). Affective learning manifesto –10 Years Later. In R. Ørngreen y K. T. Levinsen (Eds.), *Proceedings of the 13th European Conference on e-Learning ECEL-2014* (pp. 281-288).

Loannidou, A., Bennett, V., Repenning, A., Koh, K. H. y Basawapatna, A. (Abril, 2011). *Computational thinking patterns*. Documento presentado en 2011 Annual Meeting of the American Educational Research Association (AERA), Nueva Orleáns, LA. Recuperado de <https://files.eric.ed.gov/fulltext/ED520742.pdf>

Pekrun, R. (2014). *Emotions and learning*. International Academy of Education's Educational Practices Series. Belley, Francia: Gonnet Imprimeur. Recuperado de <http://unesdoc.unesco.org/images/0022/002276/227679e.pdf>

Pekrun, R., Elliot, A. J. y Maier, M. A. (2009). Achievement goals and achievement emotions: testing a model of their joint relations with academic performance. *Journal of Educational Psychology*, 101(1), 115-135. Recuperado de doi:10.1037/a0013383

Picard, R. W. y Klein, J. (2002). Computers that recognise and respond to user emotion: theoretical and practical implications. *Interacting with Computers*, 14(2), 141-169. doi:10.1016/S0953-5438(01)00055-8

Sharma, S., Ghorpade, S., Sahni, A. y Saluja, N. (2014). Survey of Intelligent Tutoring Systems : a review on the development of expert/intelligent tutoring systems, various teaching strategies and expert tutoring system design suggestions. *International Journal of Engineering Research & Technology*, 3(11), 37-42. Recuperado de <https://www.ijert.org/browse/volume-3-2014/november-2014-edition>

Tessler, J., Beth, B. y Lin, C. (2013). Using cargo-bot to provide contextualized learning of recursion. En actas Ninth Annual International ACM Conference on International Computing Education Research, ICER '13 (pp. 161-168), NuevaYork. doi:10.1145/2493394.2493411

Utting, I., Cooper, S., Kölling, M., Maloney, J. y Resnick, M. (2010). Alice, greenfoot, and scratch-a discussion. *Journal ACM Transactions on Computing Education*, 10(4), 1-11. doi:10.1145/1868358.1868364

Weragama, D. S. (2013). *Intelligent tutoring system for learning PHP*, (Tesis doctoral), Queensland University of Technology. Recuperada de <https://eprints.qut.edu.au/63202/>

Zatarain, R., Barrón-Estrada, M. L. y Muñoz, G. (2016). PREMOC: Plataforma de Reconocimiento Multimodal de Emociones. Trabajo presentado en el 8o. Congreso Mexicano de Inteligencia Artificial (COMIA), Tonanzintla, Puebla, México.