

UNIVERSIDAD DEL VALLE DE GUATEMALA
Facultad de Ingeniería



**Aplicación de Estrategia de Gamificación en el Aprendizaje de
Programación Básica para Estudiantes de Primer Año de la
Universidad del Valle de Guatemala**

Trabajo de graduación presentado por Sebastián José Solorzano Pérez
para optar al grado académico de Licenciado en Ingeniería en Ciencias
de la Computación y Tecnologías de la Información

Guatemala,

2025

UNIVERSIDAD DEL VALLE DE GUATEMALA
Facultad de Ingeniería



**Aplicación de Estrategia de Gamificación en el Aprendizaje de
Programación Básica para Estudiantes de Primer Año de la
Universidad del Valle de Guatemala**

Trabajo de graduación presentado por Sebastián José Solorzano Pérez
para optar al grado académico de Licenciado en Ingeniería en Ciencias
de la Computación y Tecnologías de la Información

Guatemala,

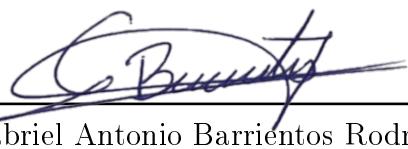
2025

Vo.Bo.:

(f) 
Ing. Angelica Gabriela Rocha Guevara

Tribunal Examinador:

(f) 
Ing. Angelica Gabriela Rocha Guevara

(f) 
PhD. Gabriel Antonio Barrientos Rodriguez

Fecha de aprobación: Guatemala, noviembre de 2025.

Lista de figuras	IX
Lista de cuadros	XI
Resumen	XIII
Abstract	XV
1. Introducción	1
2. Justificación	3
3. Objetivos	5
3.1. Objetivo general	5
3.2. Objetivos específicos	5
4. Marco teórico	7
4.1. Gamificación en la Educación	7
4.1.1. Elementos clave de la gamificación	7
4.1.2. Teorías del Aprendizaje Relacionadas	8
4.1.3. Beneficios educativos	8
4.1.4. Riesgos y limitaciones	8
4.1.5. Casos de uso en plataformas educativas	9
4.2. Enseñanza de Programación en Principiantes	9
4.2.1. Habilidades cognitivas necesarias	9
4.2.2. Dificultades comunes de estudiantes principiantes	9
4.2.3. Enfoques pedagógicos en cursos introductorios	10
4.2.4. Relevancia de Python como primer lenguaje	10
4.3. Evaluación del Impacto en el Aprendizaje y Motivación	10
4.3.1. Métodos para medir aprendizaje	10
4.3.2. Instrumentos para medir motivación	11
4.3.3. Estudios previos sobre impacto de gamificación en educación	11
4.4. Diseño de Plataformas Educativas Interactivas	11

4.4.1.	Elementos clave de una buena UX/UI en educación	12
4.4.2.	Arquitectura de sistemas educativos web	12
4.4.3.	Uso de analítica educativa para el seguimiento del progreso	13
4.5.	Tecnologías de Inteligencia Artificial para la Educación	14
4.5.1.	Definición general de IA y modelos aplicables	14
4.5.2.	Casos de uso en educación	14
4.5.3.	Ética y limitaciones	15
5. Metodología		17
5.1.	Tipo y diseño de la investigación	17
5.2.	Población y muestra	17
5.3.	Técnicas e Instrumentos de Recolección de Datos	18
5.3.1.	Registro de Progreso	18
5.3.2.	Cuestionario de Usabilidad y Motivación	18
5.3.3.	Registro de Retroalimentación Automática de IA	19
5.4.	Procedimiento	19
5.4.1.	Diseño y Desarrollo de la Plataforma	19
5.4.2.	Implementación y Prueba Piloto	23
5.4.3.	Recolección de Datos	24
5.4.4.	Análisis y Conclusiones	24
5.5.	Arquitectura de la Plataforma	24
5.5.1.	Explicación	25
5.6.	Diseño de Base de Datos	25
5.7.	Técnicas de análisis de datos	26
5.8.	Consideraciones éticas	27
6. Resultados		29
6.1.	Resultados de la plataforma	29
6.1.1.	Puntos obtenidos por usuario	29
6.1.2.	Nivel máximo alcanzado	30
6.1.3.	Insignias obtenidas	30
6.1.4.	Tasa de aciertos en ejercicios	31
6.2.	Resultados de la encuesta	32
6.2.1.	Uso de la plataforma	32
6.2.2.	Aprendizaje	33
6.2.3.	Motivación	34
6.2.4.	Modulo de Inteligencia Artificial	35
6.2.5.	Percepción general de la plataforma	36
6.3.	Resultados complementarios	37
7. Discusión		39
8. Conclusiones		45
9. Bibliografía		47

10. Anexos	51
10.1. Ejemplo de Lección en Formato Markdown	51
10.2. Insignias Disponibles	54
10.3. Modelo Completo de la Base de Datos	56
10.4. Resultados complementarios	57

Lista de figuras

1.	Modelo de diseño UX/UI: proceso paso a paso para plataformas educativas [24].	12
2.	Ejemplos de diferentes disposiciones de dashboards: abierto, estratificado, tabular y agrupado [28].	13
3.	Ejemplo de ejercicio de opción múltiple dentro de la plataforma.	20
4.	Ejemplo de ejercicio de completar código dentro de la plataforma.	21
5.	Ejemplo de ejercicio de código desde cero dentro de la plataforma.	21
6.	Ejemplo del diseño visual de una lección en la plataforma.	22
7.	Vista del perfil del usuario mostrando puntos acumulados e insignias obtenidas.	23
8.	Diagrama general de la arquitectura de la plataforma.	24
9.	Puntos obtenidos por usuario.	30
10.	Nivel máximo alcanzado por los estudiantes.	30
11.	Insignias obtenidas por los estudiantes.	31
12.	Tasa de aciertos en ejercicios.	31
13.	Resultados de usabilidad de la plataforma.	32
14.	Nube de palabras de respuestas sobre aspectos a mejorar en la usabilidad de la plataforma.	33
15.	Resultados relacionados con el aprendizaje.	33
16.	Nube de palabras de respuestas sobre los conceptos mejor aprendidos a través de la plataforma.	34
17.	Resultados relacionados con la motivación.	34
18.	Nube de palabras de respuestas sobre los elementos de gamificación más motivadores.	35
19.	Resultados relacionados con el módulo de inteligencia artificial.	35
20.	Nube de palabras de respuestas sobre aspectos a mejorar en la retroalimentación generada por la IA.	36
21.	Resultados relacionados con la percepción general de la plataforma.	36
22.	Nube de palabras de comentarios y sugerencias adicionales de los estudiantes.	36
23.	Modelo completo de la base de datos.	56

Lista de cuadros

1.	Modelo de base de datos	26
2.	Insignias por nivel completado	54
3.	Insignias por completar niveles sin errores	55
4.	Insignias por completar niveles con menos de 5 errores	55
5.	Insignias por progreso entre niveles	56
6.	Insignias iniciales del usuario	56
7.	Resultados generales obtenidos por los estudiantes en la plataforma	57
8.	Respuestas sobre la usabilidad de la plataforma (escala Likert)	58
9.	Principales respuestas sobre la usabilidad de la plataforma	58
10.	Respuestas sobre el aprendizaje obtenido mediante la plataforma (escala Likert)	59
11.	Principales conceptos aprendidos mediante la plataforma	59
12.	Respuestas sobre la motivación de los estudiantes durante el uso de la plataforma (escala Likert)	60
13.	Elementos de gamificación que resultaron más motivadores para los estudiantes	60
14.	Respuestas sobre la retroalimentación proporcionada por la IA (escala Likert)	61
15.	Principales comentarios sobre mejoras en la retroalimentación de la IA	61
16.	Respuestas sobre la percepción general de la plataforma (escala Likert)	62
17.	Comentarios adicionales y sugerencias sobre la plataforma	62

Resumen

El aprendizaje de programación básica representa un gran desafío para estudiantes que se enfrentan por primera vez a la lógica computacional, las estructuras de control y la resolución de problemas mediante código. La falta de conocimientos previos, las metodologías tradicionales de enseñanza y la idea de la programación como una materia difícil generan frustración y desmotivación, lo cual afecta el desempeño académico.

Ante esta problemática, este proyecto propone el desarrollo de una estrategia de gamificación aplicada a la enseñanza de programación básica en Python, orientada a estudiantes de primer año de la Universidad del Valle de Guatemala, sin limitarse únicamente a quienes cursan Ingeniería en Ciencias de la Computación y Tecnologías de la Información. La estrategia incluye el diseño de niveles progresivos, retos interactivos y un sistema de recompensas con el fin de reforzar conceptos clave y promover el aprendizaje.

Además, se integra un modelo de inteligencia artificial que analiza las respuestas ingresadas por los estudiantes, identifica errores y proporciona retroalimentación automatizada. Se espera que esta propuesta contribuya a mejorar la motivación y el compromiso en los cursos de programación.

Abstract

The basic learning of programming presents a significant challenge for students encountering computational logic, control structures, and problem-solving through code for the first time. The lack of prior knowledge, traditional teaching methodologies, and the perception of programming as a difficult subject often lead to frustration and demotivation, negatively impacting academic performance.

To address this issue, this project proposes the development of a gamification strategy applied to teaching basic programming in Python, aimed at first-year students at the University of Valle de Guatemala, extending beyond those enrolled in Computer Science and Information Technology programs. The strategy includes the design of progressive levels, interactive challenges, and a reward system to reinforce key concepts and foster learning.

Additionally, an artificial intelligence model is integrated to analyze students' responses, identify errors, and provide automated feedback. It is anticipated that this approach will enhance motivation and engagement in programming courses.

CAPÍTULO 1

Introducción

El aprendizaje de programación básica es uno de los retos más importantes para estudiantes universitarios que ingresan a carreras relacionadas con tecnología y computación. Este desafío no se limita a quienes cursan Ingeniería en Ciencias de la Computación y Tecnologías de la Información, sino también a estudiantes de otras carreras que deben aprender los fundamentos de programación como parte de su formación general. La lógica computacional, las estructuras de control y la resolución de problemas mediante código suelen percibirse como conceptos abstractos, difíciles de comprender sin conocimientos previos. Esta percepción genera frustración, ansiedad y desmotivación, impactando negativamente en el rendimiento académico y la permanencia estudiantil [1].

En este contexto, diversas investigaciones han identificado limitaciones en las metodologías tradicionales de enseñanza de programación, destacando la necesidad de emplear métodos de enseñanza más dinámicos y centrados en el estudiante [2]. Entre las estrategias innovadoras con mayor potencial se encuentra la gamificación, definida como el uso de elementos de juego para mejorar la motivación, el compromiso y el aprendizaje [3]. La gamificación permite transformar procesos educativos en experiencias interactivas y motivadoras, promoviendo el aprendizaje activo y la retención de conocimientos [4].

El presente proyecto propone el desarrollo de una estrategia de gamificación aplicada a la enseñanza de programación básica en Python, dirigida a estudiantes de primer año de la Universidad del Valle de Guatemala, independientemente de su carrera. Esta estrategia considera el diseño de niveles progresivos, retos interactivos, recompensas, insignias, y un sistema de retroalimentación inmediata que incentive la participación continua del estudiante. El objetivo es facilitar la comprensión de los conceptos de programación mediante una plataforma que combine el aprendizaje con la motivación generada por elementos de juego.

Además, se integra un modelo de inteligencia artificial (GPT-4o-mini), que permite analizar las respuestas ingresadas por los estudiantes, identificar errores comunes y generar retroalimentación automatizada. Esta funcionalidad responde a la necesidad de brindar apoyo personalizado, incluso en entornos con alta exigencia académica, aprovechando los avances en evaluación automática de código [5].

Con la implementación de esta plataforma gamificada, se busca aumentar la retención de conocimientos y reducir el desinterés en los cursos de programación básica. Este enfoque tiene el potencial de ser replicable y escalable, y convertirse en una herramienta clave para transformar la enseñanza de la programación en contextos educativos similares.

CAPÍTULO 2

Justificación

El aprendizaje de programación básica representa un desafío para los estudiantes principiantes, especialmente aquellos que no cuentan con conocimientos previos en lógica computacional. Las dificultades más comunes incluyen la falta de comprensión de los conceptos fundamentales, la frustración ante los errores y la carencia de metodologías didácticas efectivas que acompañen el proceso de aprendizaje. [1]

La enseñanza tradicional de programación suele centrarse en clases expositivas y ejercicios prácticos repetitivos, lo cual puede resultar poco motivador para los estudiantes y contribuir a la percepción de que la programación es una disciplina inaccesible [2]. Este contexto evidencia la necesidad de explorar nuevas estrategias que faciliten el aprendizaje y aumentar la motivación en los estudiantes.

La gamificación, definida como el uso de elementos de juego en contextos educativos, ha demostrado ser una herramienta eficaz para aumentar la motivación, el compromiso y el aprendizaje activo en diferentes disciplinas educativas [3]. Específicamente, su aplicación en el ámbito educativo permite diseñar experiencias más interactivas y atractivas para los estudiantes, impulsando el aprendizaje a través de retos, niveles de progresión y recompensas.

Estudios evidencian que los entornos gamificados pueden mejorar significativamente la motivación intrínseca del estudiante, aumentando su interés y perseverancia en materias percibidas como difíciles [4]. De esta forma, la gamificación no solo actúa como un incentivo externo, sino que también favorece el desarrollo de competencias clave como la resolución de problemas, el pensamiento crítico y la autonomía en el aprendizaje.

A pesar del potencial de la gamificación para transformar el proceso de aprendizaje, aún se observa una limitada implementación de estrategias gamificadas específicamente orientadas a la enseñanza de programación básica. Esta limitación representa una oportunidad para investigar enfoques innovadores que puedan complementar las metodologías tradicionales y responder a las necesidades de los estudiantes.

Además, también se reconoce la importancia de ofrecer retroalimentación personalizada

en el aprendizaje de la programación. La evaluación automatizada mediante inteligencia artificial se ha explorado como un mecanismo que permite escalar este tipo de apoyo, teniendo en consideración que cada estudiante aprende en diversos ritmos y estilos de aprendizaje sin comprometer la calidad del acompañamiento [5]. Esta línea de innovación educativa cobra especial relevancia en contextos con alta demanda y recursos limitados.

Dado el potencial comprobado tanto de la gamificación como de la inteligencia artificial en contextos educativos, resulta pertinente explorar su integración como una vía para responder a las limitaciones presentes en la enseñanza de programación básica. Profundizar en esta línea no solo permite atender necesidades actuales de los estudiantes, sino también ampliar el conocimiento sobre estrategias pedagógicas innovadoras que puedan ser aplicadas en diversos entornos de aprendizaje.

CAPÍTULO 3

Objetivos

3.1. Objetivo general

Aplicar una estrategia de gamificación como herramienta innovadora para la enseñanza de programación básica, facilitando el aprendizaje y evaluando su impacto en la motivación de los estudiantes.

3.2. Objetivos específicos

- Diseñar un modelo educativo basado en gamificación que emplee niveles, retos y recompensas para reforzar conceptos clave de programación en Python.
- Integrar un modelo de inteligencia artificial (GPT-4o-mini) que permita analizar las respuestas ingresadas por los estudiantes, identificar errores y generar retroalimentación automatizada.
- Evaluar el impacto de la estrategia en la motivación y aprendizaje de los estudiantes mediante encuestas y análisis de progreso dentro de la plataforma.

CAPÍTULO 4

Marco teórico

4.1. Gamificación en la Educación

La gamificación ha tomado fuerza como una estrategia educativa innovadora que busca aumentar la motivación, el compromiso y el aprendizaje con la implementación de elementos propios de los juegos en contextos de enseñanza. Esta técnica no requiere de la creación de videojuegos completos, sino la incorporación de componentes como puntos, niveles, insignias, retroalimentación inmediata y retos, que promueven la participación del estudiante en entornos de aprendizaje [3].

4.1.1. Elementos clave de la gamificación

Diversos estudios han identificado un conjunto de elementos en experiencias gamificadas [3]. Entre ellos se encuentran:

- Puntos, utilizados como sistema de recompensa para indicar progreso.
- Niveles, que estructuran el contenido en diferentes etapas y motivan la mejora continua.
- Insignias, que funcionan como reconocimiento de logros alcanzados.
- Retroalimentación inmediata, que permite al estudiante conocer sus errores y aciertos en tiempo real.
- Retos, que plantean objetivos concretos y alcanzables, manteniendo el interés.

Estos componentes no solo mejoran la experiencia del usuario, sino que también promueven un aprendizaje más significativo al activar mecanismos de recompensa y esfuerzo progresivo [4].

4.1.2. Teorías del Aprendizaje Relacionadas

La teoría del flujo describe un estado mental en el que las personas se sienten involucradas en una actividad, experimentando enfoque y pérdida de la noción del tiempo. Este estado se logra cuando las habilidades están en equilibrio con el nivel de desafío de la tarea y hay metas claras [6]. Este concepto ha sido utilizado para justificar el uso de la gamificación en entornos educativos, donde elementos como niveles, recompensas y retroalimentación pueden facilitar dicho estado.

Por otro lado, la Teoría de la Autodeterminación sostiene que la motivación humana mejora cuando se satisfacen tres necesidades: autonomía, competencia y relación social [7]. Los entornos gamificados pueden apoyar estas necesidades al ofrecer al estudiante control sobre su progreso, retroalimentación constante sobre su rendimiento, y elementos como insignias.

4.1.3. Beneficios educativos

Los beneficios de la gamificación en entornos educativos han sido muy reconocidos [3]. La gamificación puede:

- Incrementar la motivación tanto intrínseca como extrínseca.
- Aumentar el compromiso del estudiante con las actividades de aprendizaje.
- Mejorar la retención de conocimientos a través de práctica continuas.
- Desarrollar habilidades como la resolución de problemas y la autonomía.

Además, se ha observado que los estudiantes participan más activamente en cursos gamificados, y muestran una actitud más positiva hacia materias consideradas como difíciles [4].

4.1.4. Riesgos y limitaciones

La gamificación, cuando no se implementa de forma adecuada, puede generar efectos negativos en el aprendizaje. En un estudio se analizó el uso inapropiado de elementos gamificados en una aplicación educativa, y encontraron que ciertos diseños pueden provocar que los usuarios:

- Se enfoquen en la obtención de recompensas o logros dentro del sistema, más que en el aprendizaje real.
- Prioricen el cumplimiento de tareas para mantener rachas o acumular puntos, en lugar de un compromiso con el contenido educativo.
- Pierdan interés o abandonen cuando las dinámicas no están alineadas con los objetivos de aprendizaje.

Estos resultados muestran la importancia de un diseño cuidadoso de los elementos de gamificación, de modo que apoyen el aprendizaje y eviten distracciones o motivaciones extrínsecas que puedan interferir con la experiencia educativa. [8]

4.1.5. Casos de uso en plataformas educativas

Duolingo es una de las plataformas de aprendizaje de idiomas más populares del mundo. Emplea elementos de gamificación como rachas, insignias, vidas, puntos de experiencia y niveles. Estudios sobre la efectividad de su diseño indican que estos elementos incrementan la motivación y la participación del usuario, especialmente en fases iniciales del aprendizaje [9].

Sin embargo, también se han identificado riesgos. En un estudio, se observó que algunos usuarios se enfocan en mantener sus rachas, en lugar de priorizar el aprendizaje, lo que puede llevar a una experiencia superficial [8].

CodeCombat es una plataforma gamificada que enseña programación mediante el uso de misiones en un entorno tipo videojuego. A través de desafíos progresivos, los estudiantes escriben código para avanzar en niveles. Investigaciones indican que el uso de enfoques personalizados y gamificados puede mejorar la motivación y el compromiso en el aprendizaje de programación [10].

4.2. Enseñanza de Programación en Principiantes

Enseñar programación a estudiantes sin experiencia previa representa un desafío tanto en entornos académicos como en plataformas tecnológicas. Aprender a programar implica dominar habilidades abstractas, enfrentarse a un lenguaje formal y desarrollar una mentalidad lógica enfocada a la resolución de problemas. Esta combinación de factores suele provocar frustración, desmotivación y abandono en los cursos iniciales [1], [11]. Además, el lenguaje de programación elegido, el entorno de desarrollo, la modalidad del curso y el apoyo influyen en la experiencia de aprendizaje del estudiante.

4.2.1. Habilidades cognitivas necesarias

El aprendizaje de programación básica requiere habilidades de pensamiento lógico, abstracción y resolución de problemas mediante algoritmos. Muchos estudiantes presentan conocimientos previos insuficientes en estas áreas, lo que dificulta su progreso [11]. La carga cognitiva puede ser alta si no se abordan habilidades organizativas, como ordenar los pensamientos, analizar problemas y encontrar soluciones prácticas antes de escribir código [12].

4.2.2. Dificultades comunes de estudiantes principiantes

Es común que los estudiantes principiantes enfrenten dificultades relacionadas con la sintaxis del lenguaje, semántica de los operadores y estructuras de control. Además, muchos

expresan frustración cuando no encuentran apoyo para resolver estos problemas, como se evidenció en los comentarios de los estudiantes de un curso en línea, quienes mencionaron la dificultad para obtener ayuda y la falta de instrucciones detalladas [11].

Además, los proyectos abiertos generan dudas al momento de buscar, decidir e integrar ejemplos de código. Los estudiantes enfrentan problemas al momento de seleccionar ejemplos y adaptarlos correctamente a su propio código. Estas dificultades son relevantes en la programación abierta, donde los estudiantes con frecuencia intentan implementar funcionalidades que aún no han aprendido, lo cual puede retrasar su progreso [13].

4.2.3. Enfoques pedagógicos en cursos introductorios

Un enfoque utilizado en la enseñanza de la programación es el uso de *worked examples*, que ha evidenciado ser útil para reducir la carga cognitiva del estudiante. Este enfoque permite que el estudiante observe una solución paso a paso, facilitando la comprensión de los conceptos antes de que deba resolver problemas por sí mismo. El uso de *worked examples* es útil en contextos como la programación, donde los estudiantes deben aprender no solo la sintaxis, sino también la resolución lógica de problemas [14].

4.2.4. Relevancia de Python como primer lenguaje

Python es considerado un lenguaje ideal para principiantes debido a su sintaxis y su estructura de programación simple. Comparado con otros lenguajes como Java, se ha demostrado que Python facilita el aprendizaje de programación. Un estudio reveló que los estudiantes que aprenden Python muestran una mejora en su rendimiento de aprendizaje, motivación y confianza en sus habilidades de programación. Estos resultados indican que Python es adecuado para enseñar a estudiantes sin experiencia previa, ofreciendo un aprendizaje más accesible y efectivo [15].

4.3. Evaluación del Impacto en el Aprendizaje y Motivación

La evaluación en el contexto educativo se entiende como un proceso estructurado que busca analizar y valorar el progreso de los estudiantes en base a los objetivos de aprendizaje definidos. Este enfoque permite no solo evaluar el desempeño, sino también obtener información clave para ajustar y mejorar la práctica educativa [16].

4.3.1. Métodos para medir aprendizaje

Las evaluaciones brindan retroalimentación durante el proceso de aprendizaje. Investigaciones muestran que la integración de herramientas tecnológicas, como quizzes, y el uso de estrategias de evaluación formativa, como mapas conceptuales, autoevaluaciones, preguntas orales y ensayos breves, dentro del entorno educativo mejora la retención y el desempeño

en comparación con métodos tradicionales. Además, permite a los docentes ofrecer retroalimentación significativa en cada tarea, fomentando el desarrollo del conocimiento cognitivo [17].

4.3.2. Instrumentos para medir motivación

En el estudio de la motivación académica, existen varios instrumentos que permiten evaluar este concepto de forma efectiva. Uno de los más reconocidos es la Escala de Motivación Académica (AMS, por sus siglas en inglés), ampliamente utilizada por su validez y trayectoria en contextos universitarios. Esta herramienta ayuda a comprender qué factores mantienen al estudiantado motivado [18].

Además, los cuestionarios tipo Likert son frecuentemente empleados para medir variables como la motivación, debido a su capacidad para recopilar grandes cantidades de datos de manera eficiente. Para su diseño adecuado, es fundamental comprender el concepto a evaluar, definir categorías de respuesta, establecer un modelo de medición y realizar pruebas piloto [19].

4.3.3. Estudios previos sobre impacto de gamificación en educación

Una revisión reciente concluye que la gamificación tiene efectos positivos pequeños a moderados sobre el aprendizaje y la motivación (entre 0.25 y 0.56), aunque advierte que el efecto motivacional puede disminuir con la repetición o falta de novedad [20].

Un estudio experimental sobre insignias digitales en entornos gamificados (TELEs, por sus siglas en español) mostró mejoras significativas en la motivación intrínseca de estudiantes, aunque el impacto en rendimiento académico fue más moderado [21].

4.4. Diseño de Plataformas Educativas Interactivas

Una plataforma educativa debe integrar principios pedagógicos, diseño UX/UI pensado para estudiantes, y una arquitectura técnica eficiente. Los objetivos incluyen facilitar el aprendizaje, personalizar experiencias y permitir el monitoreo del progreso estudiantil. Un diseño intuitivo y accesible es importante, ya que estudios recientes indican que un gran número de usuarios abandonan un sitio web debido a una mala experiencia de usuario, lo que indica la importancia de una UX bien diseñada [22].

Un ejemplo de esto es Duolingo, que demuestra cómo un diseño claro, con navegación simple, retroalimentación inmediata y recompensas visuales, favorece la permanencia de los estudiantes y la retención del aprendizaje [9]

4.4.1. Elementos clave de una buena UX/UI en educación

Una excelente experiencia de plataforma comienza con un buen diseño UX, que lleva a una interfaz de usuario accesible. Los diseños centrados en el aprendizaje favorecen la motivación y retención de los estudiantes. Un estudio muestra que las plataformas con UX intuitiva, navegación comprensible y elementos interactivos logran una mejora en la experiencia educativa [22].

Una tendencia en las plataformas educativas interactivas es la integración de dashboards de análisis de datos de aprendizaje, los cuales proporcionan a los estudiantes y profesores herramientas para monitorear el progreso y optimizar las experiencias educativas [23].

El modelo de diseño UX/UI presentado en un estudio ofrece un proceso paso a paso que conecta la teoría con la práctica, adaptándose a plataformas educativas, como se muestra en la Figura 1. Este modelo inicia con la planificación de la experiencia de usuario (UX), donde se definen las necesidades de los usuarios, seguida de la creación de personas y recorridos de usuario para comprender su interacción con la plataforma. Posteriormente, se desarrolla la fase de *wireframing*, en la cual se estructura la interfaz y el flujo de navegación, asegurando una experiencia clara y accesible [24].

Durante la fase de diseño de interacciones, se elaboran escenarios de uso y casos específicos que permiten optimizar la interacción entre los usuarios y la plataforma. Este diseño debe validarse a través de prototipos y pruebas de usuario, ajustando y mejorando continuamente la experiencia. Finalmente, se realiza la transición del prototipo a la interfaz final, asegurando que el diseño sea consistente y escalable a largo plazo, mediante un sistema de diseño que mantenga la coherencia visual y funcional en todas las etapas de desarrollo [24].

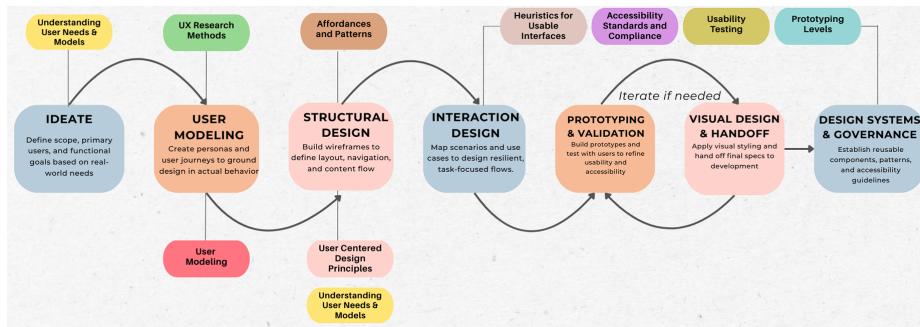


Figura 1: Modelo de diseño UX/UI: proceso paso a paso para plataformas educativas [24].

4.4.2. Arquitectura de sistemas educativos web

Las plataformas educativas suelen seguir una arquitectura cliente-servidor, donde el procesamiento se distribuye entre el navegador (cliente) y el servidor, mejorando eficiencia y escalabilidad. Las aplicaciones web utilizan el navegador como cliente, comunicándose con el servidor a través de protocolos específicos [25].

El *Frontend* es la parte visible para el usuario, construida con tecnologías como HTML, CSS y JavaScript. Su objetivo es proporcionar una experiencia de usuario accesible, permi-

tiendo la interacción fluida con la plataforma [26].

El *Backend*, por su parte, gestiona las bases de datos, verifica sesiones de usuario y procesa las solicitudes del *Frontend*. Esta división entre las dos capas permite mantener la seguridad, optimizar el control del sistema y facilitar la actualización de la plataforma sin afectar la experiencia del usuario [26].

Esta separación funcional entre el *Frontend* y el *Backend* asegura que cada capa se enfoque en tareas específicas, intercambiando información de manera eficiente para ofrecer una experiencia escalable [26].

4.4.3. Uso de analítica educativa para el seguimiento del progreso

La analítica educativa implica recolectar, procesar y visualizar datos para comprender y mejorar el aprendizaje. Los dashboards permiten a los estudiantes ver su desempeño y a los profesores monitorear el progreso de la clase. Su efectividad depende de un diseño accesible y de su capacidad para adaptarse a las necesidades de cada usuario [23].

Además, la personalización de la analítica es importante, ya que los estudiantes pueden comparar su rendimiento con el de la clase y también con su propio historial, lo que favorece una motivación y permite realizar ajustes en su aprendizaje según sus metas personales [27]. Asimismo, algunas plataformas incluyen funcionalidades como el *drill-down* que permite a los usuarios obtener una visión más detallada de sus resultados, facilitando la identificación de áreas específicas de mejora [28].

Entre los patrones de organización más utilizados se encuentran los diseños abiertos, que organizan los elementos de manera flexible; los estratificados, que ordenan la información en capas jerárquicas; los tabulares, que favorecen la precisión al presentar datos en formato de tabla; y los agrupados, que combinan números, iconos y gráficos en unidades semánticas integradas. Como se muestra en la Figura 2, cada uno de estos esquemas permite organizar grandes volúmenes de información y, en el contexto educativo, facilita que los estudiantes comprendan con mayor claridad su progreso y detecten áreas de mejora [28].



Figura 2: Ejemplos de diferentes disposiciones de dashboards: abierto, estratificado, tabular y agrupado [28].

4.5. Tecnologías de Inteligencia Artificial para la Educación

La inteligencia artificial (IA) se ha convertido en una herramienta clave en el área educativa gracias a su capacidad para personalizar el aprendizaje, ofrecer retroalimentación inmediata y automatizar tareas evaluativas. Estas tecnologías facilitan la adaptación al ritmo del estudiante y permiten escalar la enseñanza sin reducir la calidad pedagógica [29].

4.5.1. Definición general de IA y modelos aplicables

La IA se refiere a sistemas computacionales diseñados para imitar funciones cognitivas humanas como el aprendizaje, la resolución de problemas y la toma de decisiones. Estas tecnologías incluyen técnicas de aprendizaje automático (*machine learning*), redes neuronales y procesamiento del lenguaje natural (NLP, por sus siglas en inglés). La evolución de modelos más avanzados ha permitido que las máquinas desarrollen capacidades como el reconocimiento de patrones, el razonamiento estadístico y la generación de contenido lógico [30].

Los modelos de lenguaje a gran escala (LLMs, por sus siglas en inglés) representan un gran avance al estar entrenados con miles de millones de parámetros. Estos modelos se basan en la arquitectura Transformer, la cual emplea mecanismos de *self-attention* para procesar texto en paralelo y comprender relaciones contextuales entre palabras [31].

Los LLMs se entrena n con aprendizaje supervisado con grandes volúmenes de datos sin procesar, y aprenden a predecir la siguiente palabra en una secuencia. Al hacerlo millones de veces, modelan patrones lingüísticos, sintaxis y semántica, mejorando su capacidad para entender y generar texto lógico [30].

4.5.2. Casos de uso en educación

La inteligencia artificial se aplica en distintos contextos educativos para apoyar tanto la enseñanza como el aprendizaje. Entre sus usos más comunes se encuentran los siguientes:

- Evaluación automática: sistemas que califican tareas escritas o de programación comparando con respuestas esperadas. Una revisión sistemática indica que aproximadamente un 65 % de las investigaciones reportan mejoras en el rendimiento estudiantil al recibir retroalimentación automática. Además, la retroalimentación permite al docente personalizar el contenido de aprendizaje de acuerdo con las necesidades de los estudiantes. [32].
- Aprendizaje personalizado: plataformas que analizan el progreso y los estilos de aprendizaje de cada estudiante para adaptar dinámicamente tanto el ritmo como el tipo de contenido. Este enfoque no solo optimiza la experiencia de aprendizaje, sino que también potencia la capacidad de los estudiantes para aprender de manera independiente, según sus intereses y estilos de aprendizaje [33].

4.5.3. Ética y limitaciones

Aunque la IA ofrece beneficios, también presenta retos que deben enfrentarse de manera cuidadosa:

- Privacidad y transparencia: muchos sistemas requieren la recolección de datos sensibles, lo que hace necesario garantizar la protección de la identidad, el almacenamiento seguro y criterios de uso. Además, es importante contar con un consentimiento informado, una recolección de datos transparente, y ofrecer control sobre la información personal. También se debe evitar la vigilancia no intencionada, protegiendo así la privacidad [31].
- Sesgo algorítmico: si los modelos se entrena n con datos sesgados, pueden mantener desigualdades o estereotipos, afectando la equidad educativa. Los LLMs, al entrenarse con datos no filtrados, pueden mantener sesgos relacionados con género y raza. Para evitar esto, se han propuesto enfoques como el uso de estrategias de muestreo para equilibrar la distribución de los datos de entrenamiento [31].
- Dependencia excesiva: recurrir en exceso a la IA para resolver dudas o tomar decisiones puede dificultar el desarrollo del pensamiento crítico del estudiante. Este tipo de dependencia se produce cuando aceptan las recomendaciones generadas por la IA sin cuestionarlas. Además, la tendencia a priorizar soluciones rápidas y eficientes, en lugar de preferir enfoques más lentos pero prácticos, afecta las habilidades cognitivas [34].

CAPÍTULO 5

Metodología

5.1. Tipo y diseño de la investigación

La presente investigación fue de tipo aplicada, ya que tuvo como objetivo aplicar una estrategia de gamificación como herramienta innovadora para la enseñanza de programación básica en Python, orientada a estudiantes universitarios de primer año, con el propósito de facilitar el aprendizaje y evaluar su impacto en la motivación y el rendimiento académico de los estudiantes.

El enfoque utilizado fue cuantitativo-descriptivo, dado que se recopilaron y analizaron métricas de uso de la plataforma (niveles y ejercicios completados, insignias y habilidades obtenidas), acompañadas con una medición de usabilidad y motivación a través de encuestas.

El diseño fue no experimental y transeccional. Fue no experimental porque no se manipularon variables de manera controlada, sino que se observó el comportamiento de los estudiantes en el uso de la plataforma. Fue transeccional porque la recolección de datos se realizó en un único momento, al finalizar el periodo de prueba de una semana. Este diseño permitió obtener una evaluación inicial de la efectividad y usabilidad.

5.2. Población y muestra

La población objetivo estuvo conformada por estudiantes de primer año de la Universidad del Valle de Guatemala asignados en el curso de Algoritmos y Programación Básica, independientemente de la carrera.

La muestra fue no probabilística por conveniencia, ya que incluyó a los estudiantes de

las secciones en las que se incorporó el uso de la plataforma como parte de las actividades académicas del curso. De esta manera, la participación se dio dentro del desarrollo normal del curso, asegurando la inclusión de todos los estudiantes en el estudio. Participaron un total de 33 estudiantes, correspondientes a las secciones seleccionadas del curso.

Criterios de inclusión

- Estar asignado en el curso de Algoritmos y Programación Básica en el ciclo correspondiente.

Criterios de exclusión

- Ser externo a la universidad o no estar cursando la asignatura.

La selección de esta población garantizó que los contenidos de las lecciones estuvieran alineados con el programa oficial del curso, lo que permitió medir qué tan adecuado es el contenido para el aprendizaje y la aceptación del recurso.

5.3. Técnicas e Instrumentos de Recolección de Datos

Para obtener información completa sobre la experiencia de uso y el impacto de la estrategia de gamificación, se utilizaron tres instrumentos principales:

5.3.1. Registro de Progreso

Este instrumento permitió recopilar datos cuantitativos directamente desde la plataforma, con el fin de medir el desempeño real de los participantes durante su interacción con las lecciones y actividades.

- **Instrumento:** Sistema de base de datos en PostgreSQL.
- **Datos recolectados:** Ejercicios completados, número de intentos por ejercicio, insignias obtenidas, habilidades desbloqueadas y progreso global del usuario (incluyendo la categoría alcanzada y el nivel en el que se encontraba).
- **Ventaja:** Al basarse en datos registrados, se evitó depender de lo que los estudiantes recordaban sobre su propio progreso, lo que permitió obtener datos más precisos y objetivos de su avance real dentro de la plataforma.

5.3.2. Cuestionario de Usabilidad y Motivación

Con el propósito de complementar los datos objetivos obtenidos durante la experiencia de uso, se aplicó un cuestionario al finalizar la interacción con la plataforma. Este instrumento buscó recopilar percepciones de los estudiantes sobre la facilidad de uso, la claridad del contenido y el nivel de motivación generado.

- **Instrumento:** Cuestionario con escala tipo Likert aplicada al finalizar el uso de la plataforma.
- **Objetivo:** Evaluar la facilidad de uso, la claridad del contenido, el atractivo visual y el nivel de motivación generado por la experiencia gamificada.
- **Ventaja:** Combinó la evaluación de aspectos técnicos de usabilidad con la medición del impacto motivacional, lo que permitió comprender si la plataforma funcionó correctamente y si motivó la continuidad del aprendizaje.

5.3.3. Registro de Retroalimentación Automática de IA

Para analizar la función de la inteligencia artificial dentro de la plataforma, se monitoreó la interacción de los usuarios con el módulo de retroalimentación automática, registrando la frecuencia y el tipo de respuestas generadas por el sistema.

- **Instrumento:** Módulo integrado con GPT-4o-mini.
- **Datos recolectados:** Frecuencia y tipo de correcciones generadas, el nivel de detalle en las explicaciones y los errores detectados.
- **Ventaja:** Permitió analizar la utilidad de la IA como asistente educativo sin requerir una intervención manual constante.

5.4. Procedimiento

Para llevar a cabo el proyecto, se siguió un procedimiento dividido en distintas etapas que abarcaron desde el diseño y desarrollo de la plataforma hasta su implementación, recolección de datos y análisis de resultados.

5.4.1. Diseño y Desarrollo de la Plataforma

Definición de Contenidos

Los temas de las lecciones se establecieron siguiendo el programa oficial del curso de Algoritmos y Programación Básica, lo que garantizó que se abordaran los conceptos esenciales: variables, estructuras condicionales, bucles, funciones y manejo básico de datos.

Estructura de Niveles

La plataforma se organizó en 10 niveles principales, directamente alineados con el contenido oficial del curso, asegurando una progresión lógica del aprendizaje. Además, se añadieron 4 niveles complementarios creativos, enfocados en bibliotecas visuales y de desarrollo de juegos como Turtle y Pygame, con el propósito de impulsar la motivación y la exploración práctica de la programación en contextos atractivos.

Tipos de Ejercicios Implementados

Para promover una práctica activa, se diseñaron diferentes tipos de ejercicios que se ajustaran a distintos niveles de complejidad y estilos de aprendizaje. Estos ejercicios se integraron junto a cada lección para reforzar la comprensión teórica mediante la aplicación práctica.

- **Opción múltiple:** Permitieron medir de manera rápida y objetiva la comprensión conceptual. A través de varias alternativas de respuesta, se identificó si el estudiante reconocía definiciones, conceptos básicos y relaciones entre elementos del lenguaje de programación. En la Figura 3 se muestra un ejemplo del diseño visual de este tipo de ejercicio.

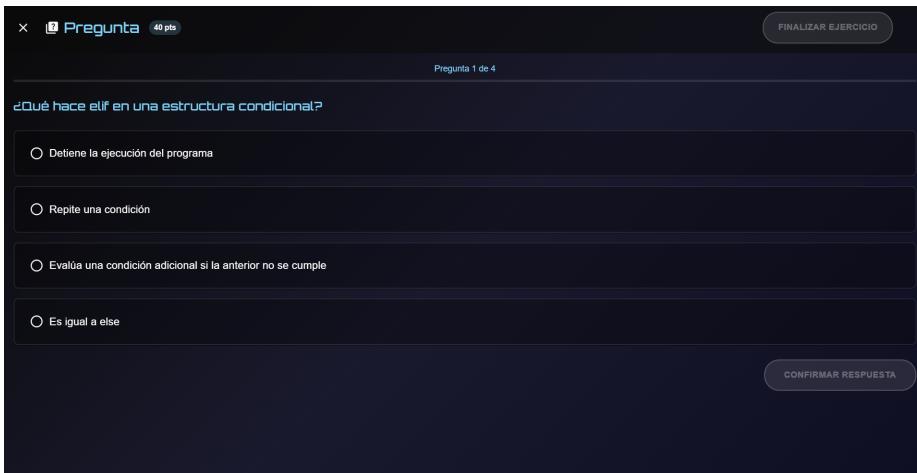


Figura 3: Ejemplo de ejercicio de opción múltiple dentro de la plataforma.

- **Completar código:** Este formato reforzó la sintaxis y la lógica de programación sin exigir la escritura completa de un programa. Los fragmentos de código con espacios en blanco o errores intencionales obligaron al estudiante a razonar paso a paso y prestar atención al detalle. En la Figura 4 se presenta un ejemplo del diseño utilizado para este tipo de ejercicio.

The screenshot shows a dark-themed interface for a coding exercise. At the top, it says "Ejercicio de Código 16 pts". On the right, there is a button labeled "TERMINAR EJERCICIO". In the center, a message reads "Completa el siguiente bloque para manejar todas las condiciones:". Below this is a code editor window titled "main.py" containing the following Python code:

```

main.py
1 nota = 85
2 if nota >= 90:
3     print("Excelente")
4 else:
5     print("Muy bien")
6 else:
7     print("Sigue intentando")

```

To the right of the code editor is a "Consola" window with the instruction "Presiona 'Comprobar' para ver el resultado de tu código...".

Figura 4: Ejemplo de ejercicio de completar código dentro de la plataforma.

- **Código desde cero:** Este tipo de ejercicio evaluó la capacidad de aplicar conocimientos en escenarios más abiertos. A partir de una instrucción inicial, los estudiantes diseñaron soluciones completas, integrando y organizando conceptos previamente aprendidos. La Figura 5 muestra un ejemplo del diseño visual utilizado para este tipo de ejercicio.

The screenshot shows a dark-themed interface for a coding exercise. At the top, it says "Ejercicio de Código 16 pts". On the right, there is a button labeled "TERMINAR EJERCICIO". In the center, a message reads "Escribe un programa que solicite una contraseña y verifique si es correcta (la contraseña correcta es 'secreto').". Below this is a code editor window titled "main.py" containing the following Python code:

```

main.py
1

```

To the right of the code editor is a "Consola" window with the instruction "Presiona 'Comprobar' para ver el resultado de tu código...".

Figura 5: Ejemplo de ejercicio de código desde cero dentro de la plataforma.

Material de Apoyo en las Lecciones

Cada lección incluyó explicaciones teóricas en texto, acompañadas de tablas para resumir diferencias entre conceptos y ejemplos de código que mostraron la aplicación práctica de los contenidos. Para asegurar que los estudiantes dedicaran tiempo real a la lectura, se incorporó un temporizador mínimo de 15 segundos en cada lección antes de poder avanzar. Además, se estableció como requisito completar el 100 % de las lecciones de un nivel para acceder a los ejercicios, promoviendo así una progresión estructurada y evitando que los usuarios avanzaran sin haber comprendido los contenidos.

En la Figura 6 se presenta una vista representativa del material de apoyo dentro de la plataforma, donde se integran el texto explicativo, los ejemplos de código y los elementos visuales que guían el aprendizaje del estudiante.

The screenshot shows a dark-themed lesson page titled "Lección". At the top right, there is a checkmark icon and the text "La lección Completada!". Below the title, there is a list of bullet points and code snippets:

- Los parámetros permiten que tus funciones sean más flexibles y puedas controlar su comportamiento desde afuera.

```
def sumar(a, b):
    resultado = a + b
    print("La suma de (a) y (b) es (resultado)")
sumar(5, 8) # Llamamos a la función con argumentos
```

- Aquí, a y b son parámetros. Cuando llamas a la función, les das un valor (eso se llama argumento).
- Puedes hacer funciones que usen parámetros para procesar información variable.

```
def repetir_texto(texto, veces):
    for i in range(veces):
        print(texto)
repetir_texto("Python es genial", 3)
```

- Esto te permite reutilizar una misma función pero con resultados diferentes. ¡Muy poderoso!

Extra: ¿Qué es return?

- Algunas funciones devuelven un resultado con la palabra return. Sirve cuando necesitas usar el valor después.

```
def sumar(a, b):
    return a + b
resultado = sumar(5, 3)
print("La suma es", resultado)
```

- Usar return es importante cuando la función debe entregar un valor para usarlo luego.

At the bottom right, there is a blue button labeled "SIGUIENTE LECCIÓN".

Figura 6: Ejemplo del diseño visual de una lección en la plataforma.

Adicionalmente, en el Anexo 10.1 se incluye un fragmento de ejemplo en formato *Markdown*, el cual fue utilizado como estructura base para almacenar y desplegar los contenidos teóricos dentro de la base de datos del sistema. Este formato permitió una integración flexible entre texto, código y elementos visuales en la interfaz.

Sistema de Puntos

Los estudiantes recibieron puntos de experiencia al completar los ejercicios de cada lección. La cantidad de puntos variaba según el nivel de dificultad del ejercicio: con los ejercicios de tipo básico los estudiantes ganaban 10 puntos, mientras que con los de mayor complejidad, como los de código desde cero, podían ganar hasta 15 puntos.

Este sistema buscó impulsar la práctica constante y la superación personal, ya que los puntos se acumulaban de manera visible en el perfil del usuario, lo que generaba una sensación de progreso y logro individual. En la Figura 7 se muestra la vista general del perfil, donde los estudiantes podían consultar su puntuación total y los logros obtenidos.

Sistema de Insignias

Junto con el sistema de puntos, se integraron distintos tipos de insignias como refuerzo motivacional, siguiendo principios de gamificación que demuestran que recompensas visuales incrementan el compromiso del estudiante. Estas insignias se otorgaban automáticamente al cumplir ciertos objetivos, como:

- Por completar niveles.

- Por cumplir categorías.
- Por finalizar ejercicios sin errores.

En la misma Figura 7 se visualiza la sección del perfil donde se presentan las insignias obtenidas por el usuario, cada una con su ícono distintivo. Además, en el Anexo 10.2 se incluye una tabla con la descripción completa de las insignias disponibles en la plataforma y su forma de obtención.

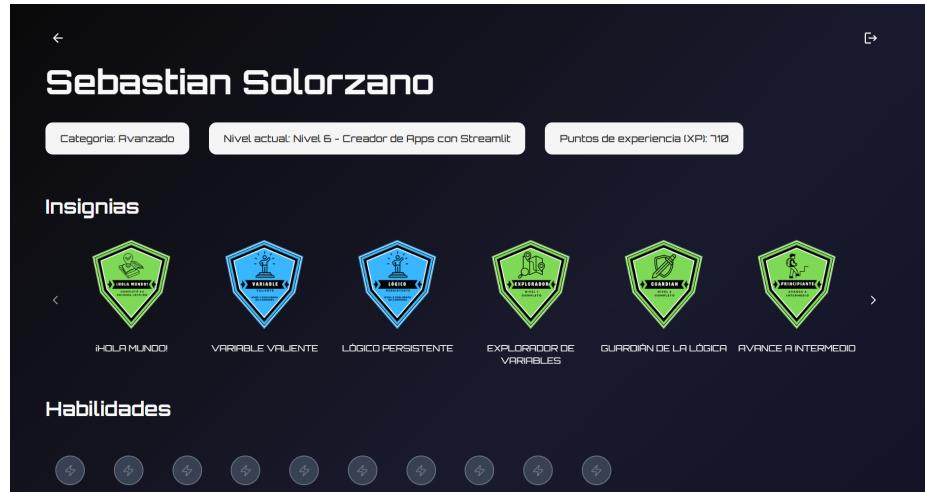


Figura 7: Vista del perfil del usuario mostrando puntos acumulados e insignias obtenidas.

Evaluación con Inteligencia Artificial

La retroalimentación de los ejercicios de código se realizó mediante IA GPT-4o-mini, lo que permitió ofrecer respuestas inmediatas y personalizadas a cada estudiante. Esto habría sido inviable de manera manual en grupos numerosos, y garantizó un acompañamiento más cercano durante el aprendizaje.

Asignación de Habilidades

Las habilidades se asignaron al terminar una lección completa, en lugar de hacerlo después de cada ejercicio. De esta manera, se aseguró que el estudiante comprendiera todos los contenidos de la lección en conjunto antes de recibir el reconocimiento, promoviendo un aprendizaje más sólido.

5.4.2. Implementación y Prueba Piloto

Una vez desarrollada la plataforma, se realizó una prueba piloto con estudiantes del curso de Algoritmos y Programación Básica. El objetivo fue evaluar su funcionamiento, usabilidad y capacidad para mantener la motivación durante el aprendizaje.

- Se presentó la plataforma a los estudiantes en una sesión introductoria.
- Se asignó un conjunto de lecciones y ejercicios a completar en un periodo de una semana.
- Se registraron automáticamente todas las interacciones en la base de datos.

5.4.3. Recolección de Datos

Posteriormente, se recopilaron los datos obtenidos tanto de la plataforma como de los instrumentos aplicados, con el fin de realizar un análisis que combinara información cuantitativa y cualitativa.

- Se extrajeron métricas de uso desde la base de datos.
- Se aplicó un cuestionario con escala tipo Likert.

5.4.4. Análisis y Conclusiones

Finalmente, se procedió a analizar los datos recolectados y a extraer conclusiones sobre la efectividad de la estrategia de gamificación implementada.

- Se procesaron las métricas y cuestionarios.
- Se interpretaron los resultados y se elaboró el informe final.

5.5. Arquitectura de la Plataforma

La Figura 8 muestra la estructura general de la plataforma desarrollada, la cual integra distintos componentes que trabajan de forma conjunta para ofrecer una experiencia educativa gamificada. A continuación, se describen los principales elementos que conforman su arquitectura.

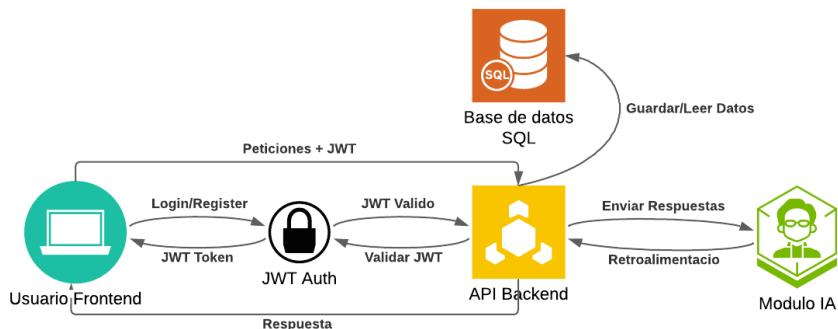


Figura 8: Diagrama general de la arquitectura de la plataforma.

5.5.1. Explicación

La arquitectura de la plataforma estuvo compuesta por los siguientes componentes principales:

- **Frontend (React):** Gestionó la interfaz de usuario y la interacción visual. Se encargó de mostrar las lecciones, ejercicios y resultados de manera interactiva, garantizando una experiencia fluida y atractiva para el usuario.
- **Backend (FastAPI):** Procesó la lógica de negocio y gestionó la comunicación entre el *frontend* y la base de datos. Además, implementó autenticación y autorización mediante JWT (JSON Web Tokens) para proteger las rutas del *frontend* y los endpoints del API, asegurando que solo los usuarios autenticados tuvieran acceso a los recursos protegidos.
- **Base de Datos (PostgreSQL):** Almacenó la información de los usuarios, su progreso en los ejercicios, los resultados obtenidos y las insignias ganadas. La base de datos estuvo estructurada de manera eficiente para garantizar consultas rápidas y un acceso seguro.
- **IA GPT-4o-mini:** Se encargó de generar retroalimentación automática para los ejercicios de código. Al recibir el código del estudiante, la IA analizó y proporcionó correcciones personalizadas en tiempo real, lo que permitió a los estudiantes aprender de sus errores y mejorar en el proceso.

5.6. Diseño de Base de Datos

La estructura de las tablas en la base de datos permitió almacenar y gestionar los datos esenciales relacionados con los estudiantes, las lecciones, el progreso y los logros obtenidos. En el Cuadro 1 se presenta un resumen del modelo de base de datos implementado, junto con la descripción y justificación de cada una de sus principales entidades.

Tabla	Descripción	Justificación
usuarios	Datos de estudiantes	Permitió gestionar a los usuarios y realizar seguimiento del progreso.
niveles	Niveles de dificultad	Organizó el contenido en etapas progresivas.
lecciones	Contenidos de cada lección	Facilitó la secuencia de contenidos y el progreso.
preguntas	Ejercicios asociados a cada lección	Permitió evaluar conocimientos adquiridos.
progreso	Seguimiento del avance del estudiante	Permitió llevar un registro de los contenidos completados por cada usuario.
intentos	Intentos y puntajes obtenidos de preguntas	Facilitó el seguimiento del desempeño en cada ejercicio y permitió generar retroalimentación.
insignias	Logros desbloqueables	Incrementó la motivación mediante recompensas simbólicas.
habilidades	Habilidades ganadas	Documentó el progreso en competencias específicas.

Cuadro 1: Modelo de base de datos

El modelo completo de la base de datos se presenta en el Anexo 10.3.

5.7. Técnicas de análisis de datos

Para interpretar los resultados obtenidos a partir de los distintos instrumentos de recolección, se emplearon técnicas de análisis tanto cuantitativas como cualitativas. Estas permitieron comprender el desempeño de los estudiantes dentro de la plataforma y evaluar su percepción sobre la experiencia de aprendizaje gamificada.

- **Datos cuantitativos:** Se realizó un análisis descriptivo con Python (`pandas`) y Excel para calcular promedios, porcentajes y distribuciones:
 - Promedio de lecciones completadas.
 - Porcentaje de ejercicios completados y correctos.
 - Distribución de insignias obtenidas.
 - Calificación promedio de usabilidad.
- **Datos cualitativos:** Se aplicó una categorización de comentarios en las siguientes categorías:
 - Facilidad de uso.
 - Claridad.
 - Motivación.
 - Efectividad de retroalimentación.

5.8. Consideraciones éticas

Los estudiantes participaron en la investigación como parte de una actividad evaluada dentro del curso de Algoritmos y Programación Básica. La calificación de la actividad correspondió únicamente al uso y cumplimiento de los ejercicios asignados, sin que los resultados individuales influyeran en su nota final.

Los datos recolectados fueron almacenados de manera anónima y utilizados exclusivamente con fines académicos. Únicamente se registró el correo de los participantes para acceder a la plataforma, sin vincularlo con los resultados de desempeño ni compartirlo con terceros.

Se garantizó la confidencialidad, el respeto y la protección de la información de los participantes en todo momento, asegurando que los datos obtenidos no se utilizaran fuera del ámbito académico ni para fines distintos a la investigación.

CAPÍTULO 6

Resultados

6.1. Resultados de la plataforma

El análisis de los registros de la plataforma permitió identificar la participación, el nivel de compromiso y el desempeño general de los 33 estudiantes. Los datos fueron recopilados a través del registro de actividades de la plataforma, lo que permitió evaluar de manera objetiva la interacción de los usuarios con los distintos niveles y ejercicios.

6.1.1. Puntos obtenidos por usuario

En la Figura 9 se presentan los puntos acumulados por cada estudiante dentro de la plataforma. El promedio de puntaje obtenido fue de 378 puntos, con un máximo de 1060 puntos y un mínimo de 0. La mayoría de los estudiantes se ubicaron en un rango intermedio de rendimiento.

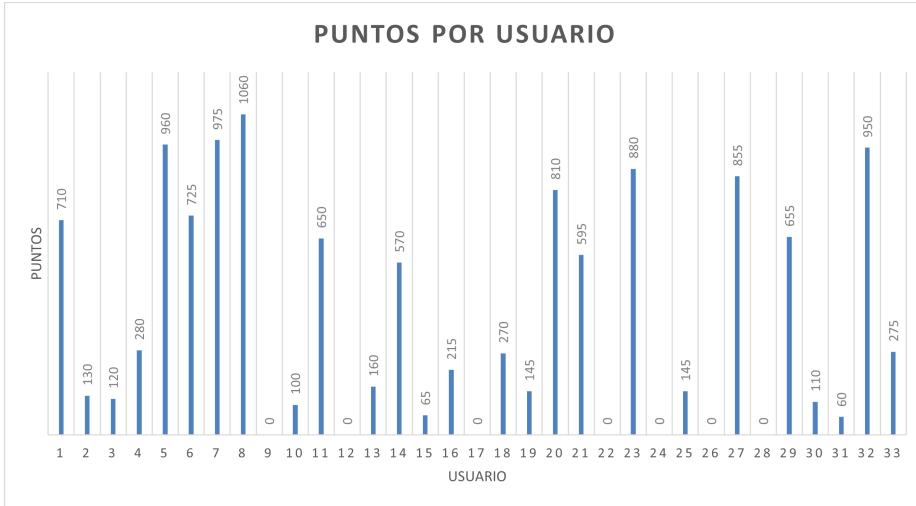


Figura 9: Puntos obtenidos por usuario.

6.1.2. Nivel máximo alcanzado

La Figura 10 muestra la distribución del nivel más alto alcanzado por los estudiantes dentro de la plataforma. De los 33 participantes, 22 lograron avanzar hasta el nivel 5, mientras que 11 permanecieron en el nivel 1 sin progresar.

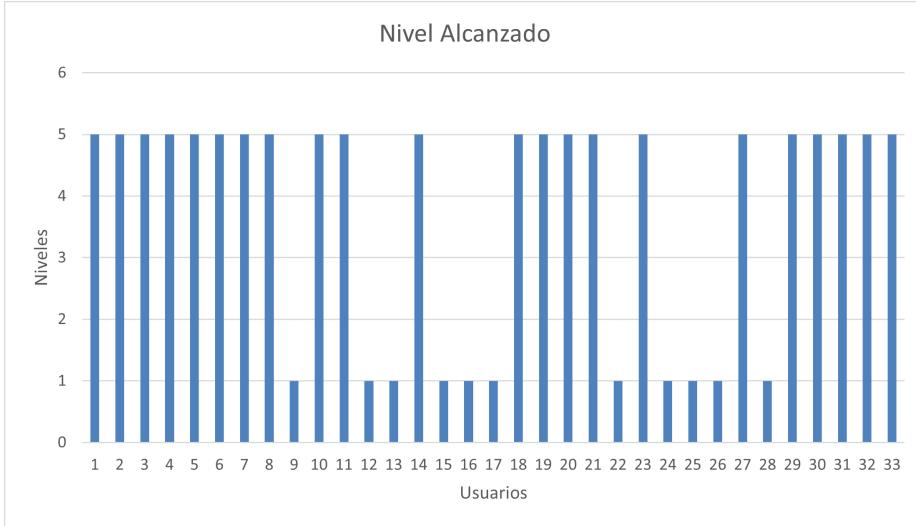


Figura 10: Nivel máximo alcanzado por los estudiantes.

6.1.3. Insignias obtenidas

En la Figura 11 se presentan las insignias obtenidas por los estudiantes como reflejo de los logros alcanzados. En promedio, los estudiantes obtuvieron 6 insignias, con un máximo

de 14 y un mínimo de 2. Cabe señalar que las 2 insignias mínimas corresponden a las básicas iniciales, otorgadas al comenzar la plataforma.

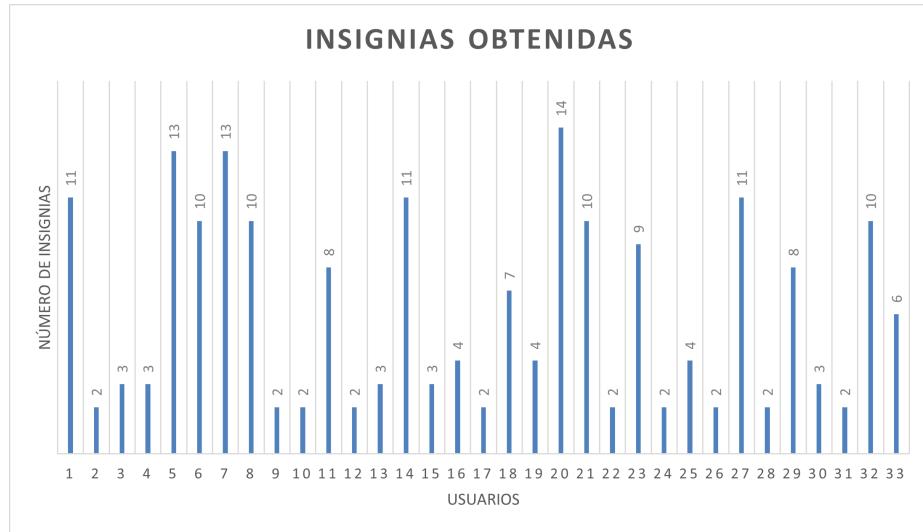


Figura 11: Insignias obtenidas por los estudiantes.

6.1.4. Tasa de aciertos en ejercicios

La Figura 12 muestra el porcentaje de respuestas correctas en los ejercicios de la plataforma. El promedio de aciertos fue de 64 %, con un máximo de 89 % y un mínimo de 32 %. Estos resultados muestran un rango amplio de rendimiento: algunos estudiantes alcanzaron un nivel alto de precisión en sus respuestas, mientras que otros tuvieron mayores dificultades en la resolución de los ejercicios.

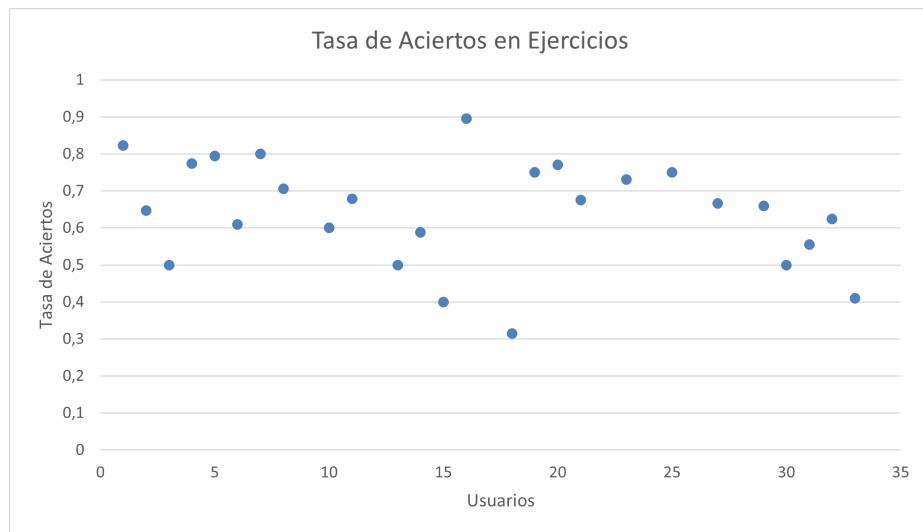


Figura 12: Tasa de aciertos en ejercicios.

6.2. Resultados de la encuesta

La encuesta permitió evaluar la percepción de los estudiantes en relación con distintos aspectos de la plataforma. Para facilitar la interpretación, los resultados se agruparon por categorías.

6.2.1. Uso de la plataforma

En la Figura 13 se muestran los resultados de cuatro aspectos clave relacionados con la usabilidad: facilidad de uso, navegación, diseño visual y tiempo de espera en las lecciones.

- Facilidad de uso: el 88 % de los estudiantes (22 en total) estuvo de acuerdo o muy de acuerdo en que la plataforma es fácil de usar.
- Navegación: el 95 % (24 estudiantes) indicó que la navegación entre niveles y lecciones es clara e intuitiva, con solo una opinión en desacuerdo.
- Diseño visual: aunque el 76 % valoró positivamente el diseño visual (19 estudiantes), se observaron opiniones más diversas, con un 16 % en neutralidad (4) y un 8 % en desacuerdo (2).
- Tiempo de espera: el 80 % (20 estudiantes) consideró adecuado el contador de lectura, mientras que solo un 8 % lo evaluó negativamente.

En las respuestas abiertas (ver Figura 14), varios estudiantes indicaron que la plataforma es clara, fácil de usar y les permitió identificar aspectos a mejorar en sus propios códigos. También valoraron su utilidad para el aprendizaje y la organización del contenido. Por otro lado, algunas sugerencias incluyeron mejorar la retroalimentación en los ejercicios y evitar que se avance con respuestas incorrectas.

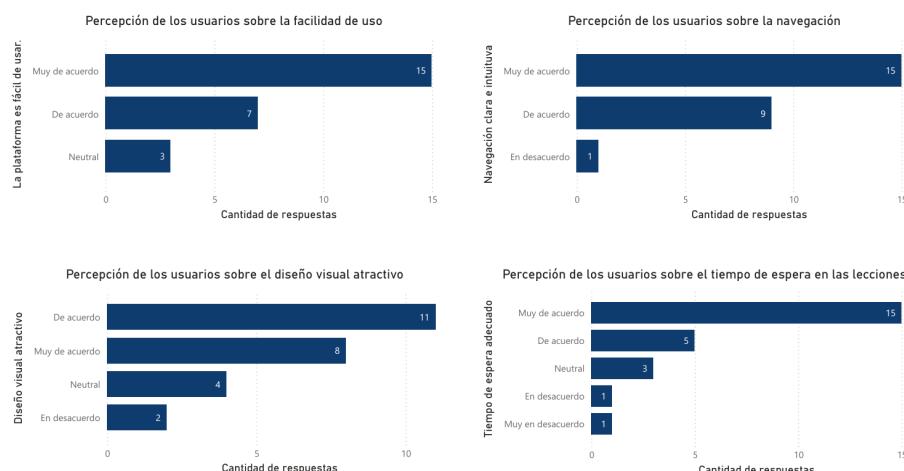


Figura 13: Resultados de usabilidad de la plataforma.



Figura 14: Nube de palabras de respuestas sobre aspectos a mejorar en la usabilidad de la plataforma.

6.2.2. Aprendizaje

En cuanto al aprendizaje, la Figura 15 muestra los resultados de tres aspectos: comprensión de conceptos básicos, utilidad de los ejercicios para reforzar lo aprendido, y desarrollo de habilidades.

- Comprensión de conceptos básicos: el 92 % de los usuarios (23 estudiantes) consideró que la plataforma les ayudó a comprender mejor los fundamentos de Python.
- Utilidad de los ejercicios: el 84 % (21 estudiantes) coincidió en que los ejercicios reforzaron lo aprendido.
- Desarrollo de habilidades: el 76 % (19 estudiantes) consideró haber adquirido nuevas habilidades, aunque un 24 % (6) se mostró neutral.

En las respuestas abiertas (ver Figura 16), los estudiantes mencionaron haber aprendido o reforzado temas clave como funciones, ciclos *while* y *for*, uso de variables, tipos de datos (enteros, decimales, cadenas), y estructuras como condicionales y bucles anidados. Varios destacaron que la plataforma les ayudó a aclarar dudas previas y comprender mejor los conceptos básicos de programación.

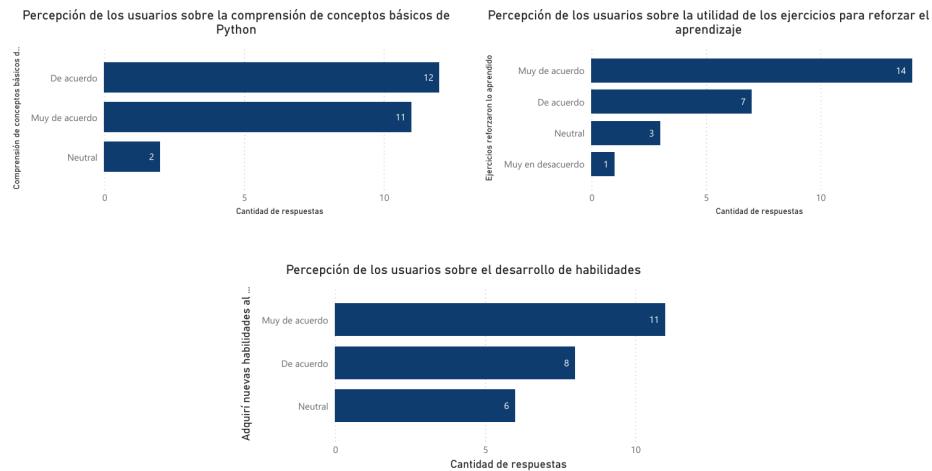


Figura 15: Resultados relacionados con el aprendizaje.



Figura 16: Nube de palabras de respuestas sobre los conceptos mejor aprendidos a través de la plataforma.

6.2.3. Motivación

En la Figura 17 se muestran los resultados de tres aspectos relacionados con la motivación generada por elementos de gamificación: puntos e insignias, niveles como retos alcanzables y comparación del propio progreso.

- Puntos e insignias: el 68 % (17 estudiantes) los consideró motivadores, mientras que una minoría expresó neutralidad o desacuerdo.
- Niveles como reto alcanzable: fue el aspecto mejor valorado, con un 80 % (20 estudiantes) que lo destacó como factor clave de motivación.
- Progreso personal: el 80 % (20 estudiantes) señaló que comparar su propio avance resultó motivador.

En las respuestas abiertas (ver Figura 18), los estudiantes mencionaron como elementos más motivadores los retos, los puntos y las insignias. También destacaron la presentación de niveles como un recorrido y la visualización del progreso como factores clave para mantener el interés durante el aprendizaje.

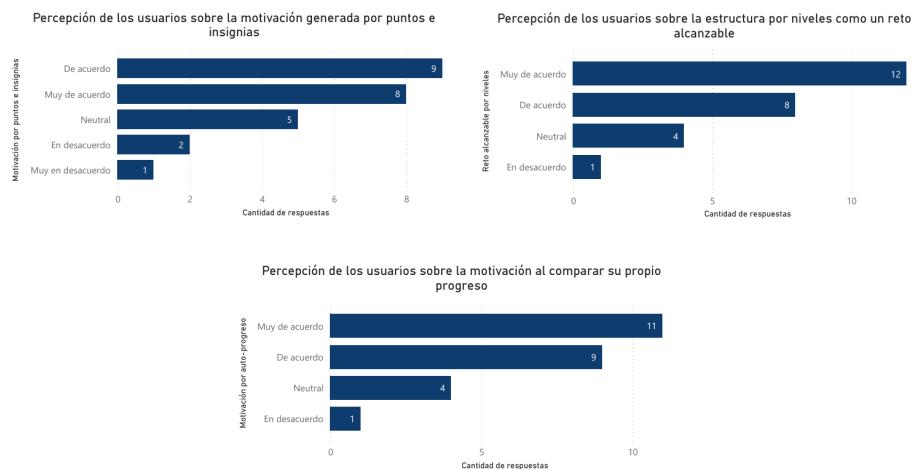


Figura 17: Resultados relacionados con la motivación.



Figura 18: Nube de palabras de respuestas sobre los elementos de gamificación más motivadores.

6.2.4. Modulo de Inteligencia Artificial

En relación con la retroalimentación automática, la Figura 19 muestra los resultados de tres aspectos: claridad de los mensajes, utilidad para corregir errores y preferencia por retroalimentación inmediata frente a la de un profesor.

- Claridad de los mensajes: el 88 % (22 estudiantes) coincidió en que fueron claros y comprensibles.
- Utilidad para corregir errores: el 84 % (21 estudiantes) indicó que la IA les ayudó a identificar y corregir errores.
- Preferencia por retroalimentación inmediata: el 84 % (21 estudiantes) señaló que prefieren la retroalimentación inmediata de la IA en lugar de esperar la corrección de un profesor.

En las respuestas abiertas (ver Figura 20), varios usuarios expresaron que no cambiarían nada, ya que consideraron útil y clara la retroalimentación. Sin embargo, algunos sugirieron mejorar la explicación de los errores, incluir ejemplos más específicos, hacer los mensajes más concisos y adaptarlos mejor al nivel del usuario.

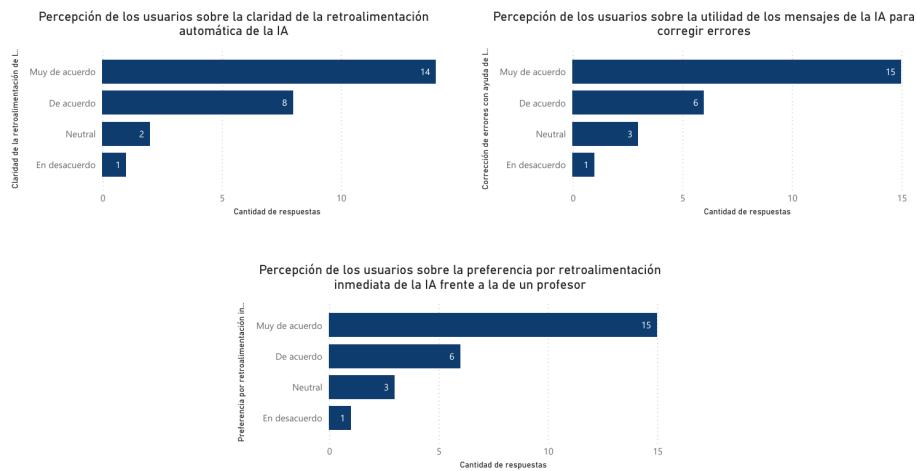


Figura 19: Resultados relacionados con el módulo de inteligencia artificial.



Figura 20: Nube de palabras de respuestas sobre aspectos a mejorar en la retroalimentación generada por la IA.

6.2.5. Percepción general de la plataforma

En la Figura 21 se presentan los resultados sobre el cumplimiento de expectativas de aprendizaje y la disposición de los usuarios a recomendar la plataforma a nuevos estudiantes. Ambos aspectos fueron altamente valorados.

- Cumplimiento de expectativas: el 88 % (22 estudiantes) indicó que la plataforma cumplió con sus expectativas de aprendizaje.
- Recomendación a nuevos estudiantes: el 92 % (23 estudiantes) recomendaría la plataforma a quienes inician en programación.

En las respuestas abiertas (ver Figura 22), muchos usuarios expresaron su satisfacción con la experiencia. Se destacaron frases como "excelente aplicación", "vale la pena", "muy completa", y "me pareció muy entretenido aprender de esta forma". Algunos afirmaron que la plataforma les habría sido útil incluso para exámenes y que sería ideal utilizarla antes de iniciar contenidos teóricos. También mencionaron que les ayudó a reforzar conocimientos previos o aclarar dudas existentes, y varios indicaron que les gustaría seguir usándola en el futuro para reforzar sus aprendizajes. También se sugirieron mejoras como agregar más ejercicios, incluir sonidos o animaciones, permitir compartir logros y mejorar aspectos visuales y explicativos.

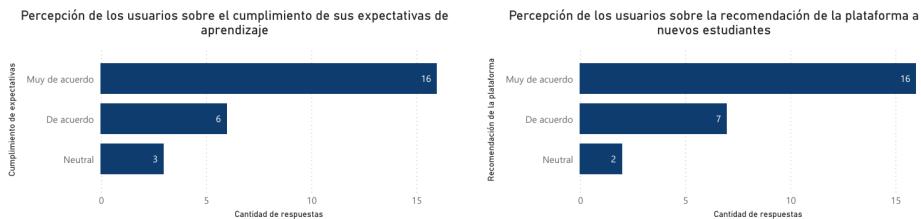


Figura 21: Resultados relacionados con la percepción general de la plataforma.



Figura 22: Nube de palabras de comentarios y sugerencias adicionales de los estudiantes.

6.3. Resultados complementarios

Para una descripción más detallada de los resultados, se incluyen en el Anexo 10.4 las tablas completas con los valores obtenidos en cada categoría (puntos por usuario, niveles alcanzados, insignias, tasas de aciertos y resultados de la encuesta).

CAPÍTULO 7

Discusión

El análisis de los puntos acumulados por cada estudiante mostró una gran variabilidad, con un promedio de 378 puntos, un máximo de 1060 puntos y un mínimo de 0 puntos. Esta variabilidad representa distintos niveles de participación y compromiso entre los estudiantes. Mientras algunos demostraron un alto compromiso al acumular más de mil puntos, otros no realizaron actividades, evidenciado en los puntajes mínimos. Esta diferencia puede interpretarse como un indicador de motivación extrínseca, donde los puntos funcionan como un sistema de recompensa que refuerza el esfuerzo dentro del proceso de aprendizaje (Ver Figura 9).

Asimismo, el análisis de las insignias obtenidas por los estudiantes reflejó los distintos logros alcanzados dentro de la plataforma. En promedio, los estudiantes obtuvieron 6 insignias, con un máximo de 14 y un mínimo de 2, correspondientes a las insignias básicas iniciales otorgadas al comenzar la plataforma. Esto evidencia que algunos estudiantes solo completaron las actividades iniciales, mientras que quienes avanzaron más acumularon un mayor número de insignias, lo que refleja distintos niveles de progresión y persistencia en el aprendizaje (Ver Figura 11).

Tanto los puntos como las insignias representan elementos clave de la gamificación, al permitir que el estudiante visualice su progreso y experimente una sensación de logro [3], [4]. Estos sistemas de recompensa ayudan a mantener la atención y el compromiso al activar mecanismos de logro y satisfacción que impulsan la continuidad en la práctica [3]. Sin embargo, su efectividad depende de un diseño equilibrado, en el que los incentivos externos se integren con estrategias que favorezcan la motivación intrínseca.

Los resultados de la encuesta sobre motivación refuerzan estos hallazgos. El 68 % de los estudiantes consideró motivadores los puntos e insignias, mientras que el 80 % destacó los niveles como un reto alcanzable y el mismo porcentaje señaló que comparar su propio progreso resultó motivador. En las respuestas abiertas, los estudiantes mencionaron los retos,

los puntos y las insignias como los principales factores que mantuvieron su interés durante el proceso de aprendizaje. En general la percepción sobre la motivación fue positiva, aunque algunos estudiantes expresaron neutralidad o desacuerdo, lo que indica que la efectividad de estos elementos puede variar según las preferencias y el nivel de compromiso (Ver Figura 17).

Estos resultados coinciden con los principios de la teoría de la autodeterminación, donde la motivación intrínseca se potencia cuando el usuario percibe competencia [11], [12]. Los sistemas de puntos e insignias aumentan el sentido de competencia al mostrar evidencia del avance, mientras que los niveles y retos brindan una estructura progresiva que favorece la percepción de superación. Además, la posibilidad de visualizar el propio progreso promueve el control sobre su aprendizaje, al permitir que el estudiante regule su ritmo de aprendizaje y establezca metas personales.

Desde la perspectiva de la analítica educativa, la distribución de puntos e insignias permite identificar el progreso y niveles de participación dentro de la plataforma [23], [27]. El análisis de estos datos facilita comprender cómo los estudiantes interactúan con los contenidos, ofreciendo información para ajustar y detectar áreas que requieren refuerzo. De esta forma, los sistemas de recompensas no solo cumplen una función motivacional, sino que también como instrumentos de monitoreo del aprendizaje [9].

El análisis del progreso de los estudiantes muestra la distribución del nivel más alto alcanzado por los estudiantes dentro de la plataforma. De los 33 participantes, 22 lograron avanzar hasta el nivel 5, mientras que 11 permanecieron en el nivel 1 sin progresar. Esto indica que una parte considerable de los estudiantes logró avanzar hasta los niveles intermedios, aunque existió un grupo que no superó los niveles iniciales. Esto refleja la diferencia en el ritmo de aprendizaje y dominio de los conceptos, común en la enseñanza de programación básica (Ver Figura 10).

Los niveles de la plataforma representan una estructura de aprendizaje progresiva, diseñada para guiar al estudiante mediante un proceso de andamiaje pedagógico. Esto permite que cada estudiante avance progresivamente, alineándose con el principio de la Zona de Desarrollo Próximo propuesto por Vygotsky, donde el progreso ocurre cuando el estudiante recibe apoyo adecuado [35]. La estructura por niveles también se relaciona con la teoría del flujo, al equilibrar el nivel de desafío con las habilidades del estudiante, manteniendo así la concentración y la sensación de dominio durante la experiencia de aprendizaje [6].

El análisis del dominio del aprendizaje presenta el porcentaje de respuestas correctas en los ejercicios de la plataforma. El promedio general fue de 64 %, con un máximo de 89 % y un mínimo de 32 %. Esto evidencia diferentes niveles de comprensión y estrategias cognitivas entre los estudiantes (Ver Figura 12). Los porcentajes altos indican un dominio de los conceptos, mientras que los más bajos reflejan dificultades típicas del aprendizaje inicial de programación, como errores lógicos, comprensión de bucles o uso de condicionales [11].

Estas diferencias en la tasa de aciertos pueden explicarse por las habilidades cognitivas necesarias para la programación, como el pensamiento lógico y la resolución de problemas. Muchos estudiantes principiantes presentan limitaciones en estas áreas, lo que incrementa la carga cognitiva al momento de analizar y estructurar soluciones [12]. Este tipo de dificultades afecta en el avance dentro de la plataforma, especialmente en aquellos que no lograron

superar los primeros niveles.

Los niveles y los ejercicios de la plataforma también actúan como una evaluación formativa, ofreciendo retroalimentación inmediata que permite identificar avances y áreas de mejora. Este tipo de evaluación continua favorece el aprendizaje y promueve una comprensión más completa de los conceptos, a diferencia de los métodos tradicionales de evaluación [17].

En el análisis del aprendizaje percibido se presentan los resultados relacionados con la percepción de aprendizaje de los estudiantes en tres aspectos: comprensión de conceptos básicos, utilidad de los ejercicios y desarrollo de habilidades. El 92 % de los participantes (23 estudiantes) indicó haber comprendido mejor los fundamentos de Python gracias a la plataforma, el 84 % (21 estudiantes) consideró que los ejercicios reforzaron lo aprendido, y el 76 % (19 estudiantes) indicó haber adquirido nuevas habilidades, aunque un 24 % (6 estudiantes) se mostró neutral (Ver Figura 15).

En las respuestas abiertas, los estudiantes indicaron que lograron aprender o reforzar temas como el uso de variables, ciclos, condiciones, funciones y estructuras anidadas. Estos resultados reflejan que la plataforma tuvo un impacto positivo en el aprendizaje cognitivo y práctico, ya que los estudiantes no solo comprendieron los conceptos teóricos, sino que también lograron aplicarlos en ejercicios (Ver Figura 16).

El diseño de la plataforma favoreció un aprendizaje basado en la práctica, apoyado en principios pedagógicos. Este enfoque se alinea con la metodología de los *worked examples*, en la cual los estudiantes observan soluciones paso a paso antes de pasar a la resolución de problemas. Este tipo de actividades reduce la carga cognitiva inicial y facilita la comprensión de estructuras lógicas del lenguaje de programación [14].

Además, la integración de ejercicios prácticos y retroalimentación inmediata funcionó como una estrategia de evaluación formativa, que permitió a los estudiantes identificar errores, reforzar conocimientos y avanzar de forma progresiva. Varias investigaciones indican que este tipo de evaluación mejora la retención y el desempeño, al impulsar el desarrollo de habilidades cognitivas mediante la práctica constante [17].

Cuando se comparan estos resultados con las métricas, como la tasa promedio de aciertos del 64 %, se observa una relación entre la percepción de aprendizaje y el rendimiento real. Esto indica que los estudiantes no solo se sintieron más capaces, sino que alcanzaron un dominio de los contenidos. La percepción de aprendizaje adquirida por la interacción con la plataforma refleja el éxito del enfoque pedagógico utilizado.

Los resultados obtenidos con relación a la experiencia de usuario y la valoración general de la plataforma reflejan una alta satisfacción por parte de los estudiantes, tanto en aspectos de diseño y funcionalidad como en la percepción global del recurso educativo.

En el análisis de la experiencia de usuario se presentan los resultados correspondientes a los cuatro indicadores de usabilidad: facilidad de uso, navegación, diseño visual y tiempos de espera. En términos generales, la percepción fue muy positiva. El 88 % de los estudiantes (22 en total) consideró que la plataforma es fácil de usar, mientras que el 95 % (24 estudiantes) indicó que la navegación entre niveles y lecciones es clara. Respecto al diseño visual, el 76 % (19 estudiantes) lo evaluó favorablemente, aunque se observaron algunas opiniones neutrales

(16 %) y en desacuerdo (8 %). En cuanto a los tiempos de espera, el 80 % (20 estudiantes) los consideró adecuados (Ver Figura 13).

En las respuestas abiertas, los estudiantes destacaron la claridad de la interfaz, la facilidad para usar la plataforma y los ejercicios para reforzar sus conocimientos. También valoraron positivamente la organización del contenido y el diseño visual, aunque algunos sugirieron mejoras relacionadas con la retroalimentación automática y el control de avance en los ejercicios. Estos hallazgos evidencian una experiencia de usuario fluida y un diseño funcional que favorece la concentración y reduce la carga cognitiva (Ver Figura 14).

Varios estudios indican que una experiencia de usuario adecuada en entornos educativos digitales depende de una arquitectura de interacción clara, accesible y centrada en las necesidades del estudiante. Un diseño UX efectivo contribuye a la motivación y retención del aprendizaje, ya que permite que los estudiantes se enfoquen en el contenido y no en las dificultades de navegación o comprensión visual [22].

En el análisis de la percepción general se presentan los resultados de la valoración de la plataforma, centrados en dos aspectos principales: cumplimiento de expectativas y disposición a recomendarla. En el primer caso, el 88 % de los estudiantes (22 en total) indicó que la plataforma cumplió con sus expectativas de aprendizaje. Por otro lado, el 92 % (23 estudiantes) afirmó que recomendaría la herramienta a nuevos usuarios interesados en aprender programación (Ver Figura 21).

Las respuestas abiertas complementan estos datos con comentarios positivos que reflejan satisfacción general. Varios afirmaron que la plataforma les habría sido útil antes de recibir contenidos teóricos. También se mencionaron sugerencias para agregar más ejercicios, incluir sonidos o animaciones, y mejorar aspectos visuales o explicativos, lo que muestra una apropiación activa del recurso y un interés a su mejora continua (Ver Figura 22).

Estos resultados evidencian que la combinación entre un diseño centrado en la experiencia del usuario y elementos de gamificación tiene un impacto positivo en la percepción general y en la aceptación del entorno educativo. Investigaciones previas respaldan que la gamificación puede incrementar la motivación intrínseca, aumentar la participación y mejorar la retención del aprendizaje [3], [4].

Los resultados relacionados con la retroalimentación automática generada por inteligencia artificial muestran una percepción positiva por parte de los estudiantes, tanto en términos de claridad como de utilidad para el aprendizaje. En el análisis se presentan los resultados de los tres aspectos evaluados: claridad de los mensajes, utilidad para corregir errores y preferencia por retroalimentación inmediata frente a la de un profesor (Ver Figura 19).

El 88 % de los estudiantes (22 en total) indicó que los mensajes generados por la IA fueron claros y comprensibles. Asimismo, el 84 % (21 estudiantes) señaló que la retroalimentación les ayudó a identificar y corregir errores en su código, y el mismo porcentaje indicó preferencia por recibir retroalimentación inmediata en lugar de esperar correcciones manuales. Estos datos reflejan que la retroalimentación automática fue percibida como un recurso positivo para el aprendizaje y la mejora continua durante el proceso de programación.

En las respuestas abiertas, la mayoría de los estudiantes expresó satisfacción con la retroalimentación generada por la IA, destacando su utilidad y claridad. Algunos comentarios,

sugirieron posibles mejoras, como ofrecer explicaciones más detalladas de los errores, incluir ejemplos adicionales o adaptar el nivel de detalle de las respuestas según el nivel del usuario. Estas respuestas evidencian que, aunque el sistema cumple adecuadamente su función principal, existen oportunidades para mejorar la personalización y precisión de los mensajes (Ver Figura 20).

La aplicación de tecnologías de inteligencia artificial en entornos educativos permite ofrecer retroalimentación inmediata, personalizada y continua, aspectos fundamentales para el aprendizaje. La retroalimentación inmediata promueve que el estudiante reflexione sobre sus errores de forma oportuna, favoreciendo la comprensión conceptual y el ajuste de estrategias cognitivas [29].

De acuerdo con la investigaciones reciente, las herramientas impulsadas por IA permiten adaptar el aprendizaje a los ritmos y estilos de cada estudiante, fortaleciendo la motivación al ofrecer experiencias más personalizadas [33].

Además, el uso de retroalimentación automatizada puede integrarse dentro de estrategias de evaluación formativa, donde la interacción entre la respuesta del estudiante y la retroalimentación del sistema promueve la reflexión, el ajuste del conocimiento y el desarrollo del pensamiento crítico [17]. En este contexto, la retroalimentación generada por IA no solo cumple una función correctiva, sino también orientadora, al guiar al estudiante hacia la comprensión de sus errores.

CAPÍTULO 8

Conclusiones

La aplicación de una estrategia de gamificación para la enseñanza de programación básica permitió mejorar la motivación, la participación y el aprendizaje de los estudiantes. Los resultados obtenidos evidencian que los elementos, como los niveles, la retroalimentación inmediata y las recompensas, contribuyen a mantener el interés durante el proceso de aprendizaje y facilitan la comprensión de los conceptos fundamentales. De acuerdo con los resultados, el 80 % de los estudiantes destacó los niveles como un factor de motivación, mientras que el 68 % consideró que los puntos e insignias influyeron en su interés por aprender.

El modelo educativo diseñado demostró que la gamificación puede convertir el aprendizaje de la programación en un proceso más atractivo. Los niveles progresivos, los ejercicios y las recompensas impulsaron la práctica de forma continua y mejoraron la retención del conocimiento. Este modelo reforzó la comprensión de los conceptos claves de Python, lo cual se reflejó en que el 92 % de los estudiantes consideró haber comprendido mejor los fundamentos del lenguaje y el 84 % indicó que los ejercicios le ayudaron a reforzar lo aprendido.

La incorporación del modelo de inteligencia artificial (GPT-4o-mini) permitió ofrecer retroalimentación automatizada, identificando errores en las respuestas de los estudiantes y generando retroalimentación inmediata. Este elemento demostró ser apoyo esencial dentro del proceso de aprendizaje, ya que ayudó a corregir errores y fortaleció la comprensión de cada tema. El 84 % indicó que la retroalimentación les permitió corregir errores. Sin embargo, algunos estudiantes señalaron que algunas respuestas podían mejorarse, lo que representa una oportunidad de mejora para futuras versiones.

El análisis de las encuestas y del progreso dentro de la plataforma evidenció un impacto positivo de la gamificación en la motivación y el aprendizaje. La mayoría de los estudiantes indicaron sentirse motivados con la metodología. Asimismo, el 88 % señaló que la plataforma cumplió con sus expectativas de aprendizaje y el 92 % indicó que la recomendaría a otros estudiantes, lo que confirma la aceptación del modelo.

CAPÍTULO 9

Bibliografía

- [1] J. I. Fuentes-Rosado y M. Moo-Medina, “Dificultades de aprender a programar,” *Revista Digital Educación en Ingeniería*, vol. 12, págs. 76-82, 24 2017.
- [2] E. A. Jones, C. A. Jimenez, P. I. Ormeño y N. A. Poblete, “Metodologías activas para la enseñanza de programación a estudiantes de ingeniería civil informática,” *Formación Universitaria*, vol. 15, págs. 53-60, 3 2022.
- [3] D. Dicheva, C. Dichev, G. Agre y G. Angelova, “Gamification in education: A systematic mapping study,” *Journal of Educational Technology & Society*, vol. 18, págs. 75-68, 3 2015.
- [4] P. Buckley y E. Doyle, “Gamification and student motivation,” *Interactive Learning Environments*, vol. 24, págs. 1162-1175, 6 2016.
- [5] M. Messer, N. C. Brown, M. Kölling y M. Shi, “Automated grading and feedback tools for programming education: A systematic review,” *ACM Transactions on Computing Education*, vol. 24, págs. 1-43, 1 2024.
- [6] M. Csikszentmihalyi, *Flow: The Psychology of Optimal Experience*. New York: HarperPerennial, 1990.
- [7] R. M. Ryan y E. L. Deci, “Self-determination theory and the facilitation of intrinsic motivation, social development, and well-being,” *American Psychologist*, vol. 55, págs. 68-78, 1 2000.
- [8] R. H. Mogavi, B. Guo, Y. Zhang, E.-U. Haq, P. Hui y X. Ma, “When Gamification Spoils Your Learning: A Qualitative Case Study of Gamification Misuse in a Language-Learning App,” *arXiv preprint*, 2022.
- [9] K. K. James y R. E. Mayer, “Learning a second language by playing a game,” *Applied Cognitive Psychology*, vol. 33, págs. 669-674, 4 2019.
- [10] K. Ishaq y A. Alvi, “Personalization, Cognition, and Gamification-Based Programming Language Learning: A State-of-the-Art Systematic Literature Review,” *arXiv preprint*, 2023.

- [11] S. Singh, "Identifying Learning Challenges Faced by Novice/Beginner Computer Programming Students: An Action Research Approach," en *Proceedings of the QuA-SoQ/SEED Workshop @ APSEC*, dic. de 2022, págs. 63-71.
- [12] O. Kurniawan, C. Jégourel, N. T. S. Lee, M. D. Mari y C. M. Poskitt, "Steps before syntax: Helping novice programmers solve problems using the PCDIT framework," *arXiv preprint*, 2021.
- [13] W. Wang et al., "Novices' learning barriers when using code examples in open-ended programming," en *Proceedings of the 26th ACM Conference on Innovation and Technology in Computer Science Education V*. 1, 2021, págs. 394-400.
- [14] J. Sweller, "The worked example effect and human cognition," *Learning and Instruction*, vol. 16, págs. 165-169, 2 2006.
- [15] H. C. Ling, K. L. Hsiao y W. C. Hsu, "Can students' computer programming learning motivation and effectiveness be enhanced by learning Python language? A multi-group analysis," *Frontiers in Psychology*, vol. 11, 2020.
- [16] J. M. F. Torrico y C. L. F. Zubieta, "La evaluación en el proceso de aprendizaje," *Perspectivas*, págs. 15-30, 19 2007.
- [17] T. Hagos y D. Andargie, "Effects of Technology-Integrated Formative Assessment on Students' Retention of Conceptual and Procedural Knowledge in Chemical Equilibrium Concepts," *Science Education International*, vol. 33, págs. 383-391, 4 oct. de 2022.
- [18] L. Sánchez-Bolívar y A. Martínez-Martínez, "Factores Relacionados con la Motivación del Alumnado Universitario e Instrumentos para su Evaluación: Una Revisión Sistemática," *Revista Electrónica Educare*, vol. 26, págs. 484-505, 2 jul. de 2022.
- [19] T. Nemoto y D. Beglar, "Likert-Scale Questionnaires," en *JALT 2013 Conference Proceedings*, nov. de 2014, págs. 1-6.
- [20] M. Khoshnoodifar, A. Ashouri y M. Taheri, "Effectiveness of Gamification in Enhancing Learning and Attitudes: A Study of Statistics Education for Health School Students," *Journal of Advances in Medical Education & Professionalism*, vol. 11, págs. 230-238, 4 oct. de 2023.
- [21] J. Luo, "Validating the Impact of Gamified Technology-Enhanced Learning Environments on Motivation and Academic Performance: Enhancing TELEs with Digital Badges," *Frontiers in Education*, vol. 9, nov. de 2024.
- [22] T. K. Miya e I. Govender, "UX/UI design of online learning platforms and their impact on learning: A review," *International Journal of Research in Business and Social Science*, vol. 11, págs. 316-327, 10 2022.
- [23] K. Williamson y R. Kizilcec, "A Review of Learning Analytics Dashboard Research in Higher Education: Implications for Justice, Equity, Diversity, and Inclusion," en *Proceedings of the 12th International Learning Analytics and Knowledge Conference (LAK22)*, mar. de 2022, págs. 260-270.
- [24] J. Natarajan, "From Wireframes to Interfaces: Developing a Practical Framework for UI/UX Design," *From Wireframes to Interfaces: Developing a Practical Framework for UI/UX Design*, 2025.
- [25] L. Shklar y R. Rosen, *Web Application Architecture*. John Wiley & Sons, 2009, vol. 36.

- [26] A. F. Mestres, “Introducción al frontend y backend,” *Revista Openaccess*, 2018.
- [27] A. F. Wise y J. Vytasek, “Learning Analytics Implementation Design,” en *Handbook of Learning Analytics*. The Society for Learning Analytics Research, 2017, vol. 1, págs. 151-160.
- [28] B. Bach et al., “Dashboard Design Patterns,” *IEEE Transactions on Visualization and Computer Graphics*, vol. 29, págs. 342-352, 1 ene. de 2023.
- [29] L. Yan et al., “Practical and ethical challenges of large language models in education: A systematic scoping review,” *British Journal of Educational Technology*, vol. 55, págs. 90-112, 1 2024.
- [30] S. Minaee et al., “Large language models: A survey,” *arXiv preprint*, 2024.
- [31] A. Vaswani et al., “Attention is all you need,” en *Advances in Neural Information Processing Systems*, 2017.
- [32] A. P. Cavalcanti et al., “Automatic feedback in online learning environments: A systematic literature review,” *Computers and Education: Artificial Intelligence*, vol. 2, 2021.
- [33] S. Wu et al., “A comprehensive exploration of personalized learning in smart education: From student modeling to personalized recommendations,” *arXiv preprint*, 2024.
- [34] C. Zhai, S. Wibowo y L. D. Li, “The effects of over-reliance on AI dialogue systems on students’ cognitive abilities: A systematic review,” *Smart Learning Environments*, vol. 11, 1 2024.
- [35] A. Álvarez y P. del Río, “Educación y desarrollo: la teoría de Vigotsky y la zona de desarrollo próximo,” en *Desarrollo psicológico y educación*, C. Coll, J. Palacios y A. Marchesi, eds. Madrid, España: Alianza Editorial, 1990, vol. 2, págs. 93-120.

CAPÍTULO 10

Anexos

10.1. Ejemplo de Lección en Formato Markdown

A continuación, se presenta un ejemplo de cómo se estructuró una lección en formato *Markdown* para su almacenamiento y despliegue en la plataforma.

```
# Nivel 5 - Arquitecto de Funciones

## Lección 1: ¿Qué es una función?
Una función en Python siempre comienza con la palabra clave def, seguida del nombre que le das y unos paréntesis ().

```python
def saludar():
 print("¡Hola, programador!")

saludar() # Llamamos a la función para que se ejecute
```

Una función es una porción de código que tiene un nombre y puedes usarla muchas veces sin repetir el mismo código. Si no la llamas, no se ejecuta.

## Lección 2: Mi primera función

```python
def sumar_dos_numeros():
 resultado = 5 + 3
 print("La suma es:", resultado)

sumar_dos_numeros()
```
```

Las funciones ayudan a organizar mejor tu código y a evitar repetir instrucciones.

Cada vez que llamas `sumar_dos_numeros()`, se ejecuta el bloque de código que definiste dentro.

Lección 3: Funciones con parámetros

Los parámetros permiten que tus funciones sean más flexibles y puedas controlar su comportamiento desde afuera.

```
```python
def sumar(a, b):
 resultado = a + b
 print(f"La suma de {a} y {b} es {resultado}")

sumar(5, 8) # Llamamos a la función con argumentos
```

```

Aquí, `a` y `b` son parámetros. Cuando llamas a la función, les das un valor (eso se llama argumento).

Puedes hacer funciones que usen parámetros para procesar información variable.

```
```python
def repetir_texto(texto, veces):
 for _ in range(veces):
 print(texto)

repetir_texto("Python es genial", 3)
```

```

Esto te permite reutilizar una misma función pero con resultados diferentes. ¡Muy poderoso!

Extra: ¿Qué es `return`?

Algunas funciones devuelven un resultado con la palabra `return`. Sirve cuando necesitas usar el valor después.

```
```python
def sumar(a, b):
 return a + b

resultado = sumar(5, 3)
print("La suma es:", resultado)
```

```

Usar `return` es importante cuando la función debe entregar un valor para usarlo luego.

Lección 4: Modularidad en acción

Puedes combinar funciones más pequeñas para crear cosas más complejas. Eso se llama modularidad.

```
```python
def obtener_doble(numero):
 return numero * 2

def imprimir_doble(numero):
 doble = obtener_doble(numero)
 print(f"El doble de {numero} es {doble}")

imprimir_doble(10)
```

```

Cada función hace una sola cosa. Al juntarlas, podemos construir procesos más completos.

Lección 5: Mini proyecto guiado - Calculadora simple

Copia y pega este código en tu editor y ejecútalo para ver cómo funciona la calculadora.

```
```python
def sumar(a, b):
 return a + b

def restar(a, b):
 return a - b

def calculadora():
 x = 10
 y = 5
 print("Suma:", sumar(x, y))
 print("Resta:", restar(x, y))

calculadora()
```

```

Usamos funciones con parámetros y funciones que devuelven valores (return).

Aquí aplicamos:

- Funciones reutilizables
- Parámetros de entrada
- Retorno de valores
- Organización del código

Bonus Lección: Buenas prácticas con funciones

Reglas y consejos al usar funciones:

- Usa nombres descriptivos (por ejemplo, sumar, no f1)
- No te olvides de llamar a la función o no pasará nada

- Usa `return` si necesitas que la función devuelva un valor
- Divide problemas grandes en funciones pequeñas (modularidad)

10.2. Insignias Disponibles

A continuación, se presentan los diferentes tipos de insignias disponibles en la plataforma, organizadas por categoría según los criterios de logro y desempeño de los estudiantes. Cada cuadro muestra el nombre de la insignia y el requisito para obtenerla.

El Cuadro 2 presenta las insignias otorgadas al completar cada nivel de la plataforma, las cuales reflejan el progreso general del estudiante dentro de la plataforma gamificada.

| Nombre de la insignia | Descripción del logro |
|--------------------------------|---|
| Explorador de variables | Completó el Nivel 1 (Explorador de Variables) |
| Guardián de la lógica | Completó el Nivel 2 (Guardián de la Lógica) |
| Maestro de decisiones | Completó el Nivel 3 (Maestro de Decisiones) |
| Domador de bucles | Completó el Nivel 4 (Domador de Bucles) |
| Forjador de funciones | Completó el Nivel 5 (Forjador de Funciones) |
| Creador de apps con Streamlit | Completó el Nivel 6 (Creador de Apps con Streamlit) |
| Mago de módulos | Completó el Nivel 7 (Mago de Módulos) |
| Guardián de listas y cadenas | Completó el Nivel 8 (Guardián de Listas y Cadenas) |
| Maestro de diccionarios | Completó el Nivel 9 (Maestro de Diccionarios) |
| Invocador de errores | Completó el Nivel 10 (Invocador de Errores) |
| Aprendiz de Turtle | Completó el Nivel 11 (Aprendiz del Dibujo con Turtle) |
| Arquitecto visual con Turtle | Completó el Nivel 12 (Arquitecto Visual con Turtle) |
| Iniciador de juegos con Pygame | Completó el Nivel 13 (Iniciador de Juegos con Pygame) |
| Maestro de juegos con Pygame | Completó el Nivel 14 (Maestro de Juegos con Pygame) |

Cuadro 2: Insignias por nivel completado

En el Cuadro 3 se muestran las insignias obtenidas por completar los ejercicios de un nivel sin tener errores, representando un desempeño perfecto en las evaluaciones.

| Nombre de la insignia | Descripción del logro |
|--------------------------------|----------------------------------|
| Precisión variable | Completó el Nivel 1 sin errores |
| Lógica sin falla | Completó el Nivel 2 sin errores |
| Decisión perfecta | Completó el Nivel 3 sin errores |
| Bucle impecable | Completó el Nivel 4 sin errores |
| Forjador de funciones perfecto | Completó el Nivel 5 sin errores |
| Creador de apps perfecto | Completó el Nivel 6 sin errores |
| Mago de módulos | Completó el Nivel 7 sin errores |
| Lógica de listas perfecta | Completó el Nivel 8 sin errores |
| Diccionario perfecto | Completó el Nivel 9 sin errores |
| Sin errores ante errores | Completó el Nivel 10 sin errores |
| Dibujo Turtle perfecto | Completó el Nivel 11 sin errores |
| Arquitecto visual perfecto | Completó el Nivel 12 sin errores |
| Iniciador de juegos | Completó el Nivel 13 sin errores |
| Juego perfecto | Completó el Nivel 14 sin errores |

Cuadro 3: Insignias por completar niveles sin errores

El Cuadro 4 incluye las insignias obtenidas por completar los ejercicios con una cantidad mínima de errores, reconociendo un desempeño destacado pero no perfecto.

| Nombre de la insignia | Descripción del logro |
|----------------------------------|--|
| Variable valiente | Terminó el Nivel 1 con menos de 5 errores |
| Lógico persistente | Terminó el Nivel 2 con menos de 5 errores |
| Decisiones constantes | Terminó el Nivel 3 con menos de 5 errores |
| Bucle resistente | Terminó el Nivel 4 con menos de 5 errores |
| Forjador de funciones en proceso | Terminó el Nivel 5 con menos de 5 errores |
| Creador de apps en marcha | Terminó el Nivel 6 con menos de 5 errores |
| Módulos en marcha | Terminó el Nivel 7 con menos de 5 errores |
| Aprendiz de listas | Terminó el Nivel 8 con menos de 5 errores |
| Diccionario en construcción | Terminó el Nivel 9 con menos de 5 errores |
| Aprendiz de excepciones | Terminó el Nivel 10 con menos de 5 errores |
| Dibujo preciso | Terminó el Nivel 11 con menos de 5 errores |
| Arquitecto preciso | Terminó el Nivel 12 con menos de 5 errores |
| Iniciador notable | Terminó el Nivel 13 con menos de 5 errores |
| Juego notable | Terminó el Nivel 14 con menos de 5 errores |

Cuadro 4: Insignias por completar niveles con menos de 5 errores

El Cuadro 5 muestra las insignias relacionadas con el progreso entre categorías, diseñadas para motivar al usuario a continuar su avance dentro de la plataforma.

| Nombre de la insignia | Descripción del logro |
|-----------------------|------------------------------------|
| Avance a intermedio | Completó los niveles Principiantes |
| Avance a avanzado | Completó los niveles Intermedios |
| Avance a experto | Completó los niveles Avanzados |
| Experto | Completó los niveles Expertos |

Cuadro 5: Insignias por progreso entre niveles

Finalmente, el Cuadro 6 presenta las insignias iniciales otorgadas al usuario al comenzar su experiencia dentro de la plataforma.

| Nombre de la insignia | Descripción del logro |
|---------------------------|--|
| Primer paso | Se registró en la plataforma |
| ¡Hola mundo! | Completó su primera lección |
| Primer ejercicio resuelto | Completó su primer ejercicio correctamente |

Cuadro 6: Insignias iniciales del usuario

10.3. Modelo Completo de la Base de Datos

A continuación en la Figura 23 se presenta el modelo completo de la base de datos utilizada en la plataforma, mostrando las tablas, sus atributos y las relaciones entre ellas.

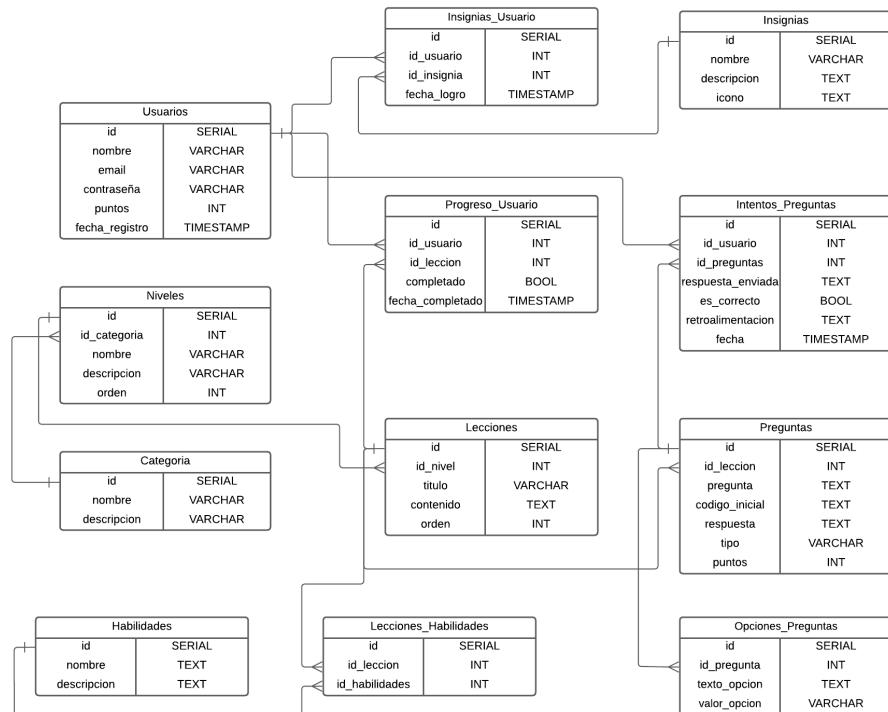


Figura 23: Modelo completo de la base de datos.

10.4. Resultados complementarios

A continuación se presentan los cuadros con los resultados completos obtenidos por los estudiantes durante el uso de la plataforma y en la encuesta. Estos datos complementan el análisis mostrado en el capítulo de resultados.

El Cuadro 7 muestra los valores recopilados de la base de datos del sistema, incluyendo el puntaje total acumulado, el nivel alcanzado, el número de insignias obtenidas y la tasa de aciertos registrada por cada participante. Estos resultados reflejan el nivel de progreso y desempeño alcanzado dentro de la plataforma.

| Usuario | Puntos | Nivel alcanzado | Insignias | Tasa de aciertos |
|----------------|---------------|------------------------|------------------|-------------------------|
| 1 | 710 | 5 | 11 | 0.8228 |
| 2 | 130 | 5 | 2 | 0.6471 |
| 3 | 120 | 5 | 3 | 0.5000 |
| 4 | 280 | 5 | 3 | 0.7742 |
| 5 | 960 | 5 | 13 | 0.7938 |
| 6 | 725 | 5 | 10 | 0.6100 |
| 7 | 975 | 5 | 13 | 0.8000 |
| 8 | 1060 | 5 | 10 | 0.7063 |
| 9 | 0 | 1 | 2 | - |
| 10 | 100 | 5 | 2 | 0.6000 |
| 11 | 650 | 5 | 8 | 0.6790 |
| 12 | 0 | 1 | 2 | - |
| 13 | 160 | 1 | 3 | 0.5000 |
| 14 | 570 | 5 | 11 | 0.5882 |
| 15 | 65 | 1 | 3 | 0.4000 |
| 16 | 215 | 1 | 4 | 0.8947 |
| 17 | 0 | 1 | 2 | - |
| 18 | 270 | 5 | 7 | 0.3151 |
| 19 | 145 | 5 | 4 | 0.7500 |
| 20 | 810 | 5 | 14 | 0.7701 |
| 21 | 595 | 5 | 10 | 0.6750 |
| 22 | 0 | 1 | 2 | - |
| 23 | 880 | 5 | 9 | 0.7312 |
| 24 | 0 | 1 | 2 | - |
| 25 | 145 | 1 | 4 | 0.7500 |
| 26 | 0 | 1 | 2 | - |
| 27 | 855 | 5 | 11 | 0.6667 |
| 28 | 0 | 1 | 2 | - |
| 29 | 655 | 5 | 8 | 0.6588 |
| 30 | 110 | 5 | 3 | 0.5000 |
| 31 | 60 | 5 | 2 | 0.5556 |
| 32 | 950 | 5 | 10 | 0.6240 |
| 33 | 275 | 5 | 6 | 0.4098 |

Cuadro 7: Resultados generales obtenidos por los estudiantes en la plataforma

El Cuadro 8 muestra las respuestas de los estudiantes en la encuesta de usabilidad, relacionados con la facilidad de uso, navegación, diseño visual y tiempo de lectura. En el Cuadro 9 se presentan las sugerencias de mejora expresadas por los estudiantes en la pregunta abierta.

| Usuario | Facilidad de uso | Navegación clara | Diseño visual | Tiempo de lectura |
|---------|------------------|------------------|----------------|-------------------|
| 1 | Muy de acuerdo | De acuerdo | De acuerdo | Muy de acuerdo |
| 2 | Muy de acuerdo | Muy de acuerdo | Muy de acuerdo | Neutral |
| 3 | De acuerdo | Muy de acuerdo | Muy de acuerdo | De acuerdo |
| 4 | Muy de acuerdo | Muy de acuerdo | De acuerdo | Muy de acuerdo |
| 5 | De acuerdo | Muy de acuerdo | Neutral | De acuerdo |
| 6 | Muy de acuerdo | Muy de acuerdo | De acuerdo | Muy de acuerdo |
| 7 | Muy de acuerdo | Muy de acuerdo | En desacuerdo | Muy de acuerdo |
| 8 | Muy de acuerdo | De acuerdo | Neutral | Muy de acuerdo |
| 9 | Muy de acuerdo | De acuerdo | De acuerdo | Muy de acuerdo |
| 10 | De acuerdo | De acuerdo | De acuerdo | Muy de acuerdo |
| 11 | Muy de acuerdo | Muy de acuerdo | Muy de acuerdo | Muy de acuerdo |
| 12 | Muy de acuerdo | Muy de acuerdo | De acuerdo | Muy de acuerdo |
| 13 | Muy de acuerdo | Muy de acuerdo | De acuerdo | De acuerdo |
| 14 | Muy de acuerdo | Muy de acuerdo | De acuerdo | Muy de acuerdo |
| 15 | Muy de acuerdo | De acuerdo | De acuerdo | Muy de acuerdo |
| 16 | Muy de acuerdo | Muy de acuerdo | Muy de acuerdo | Neutral |
| 17 | De acuerdo | Muy de acuerdo | Muy de acuerdo | Muy de acuerdo |
| 18 | Neutral | En desacuerdo | De acuerdo | Muy de acuerdo |
| 19 | Neutral | De acuerdo | Neutral | En desacuerdo |
| 20 | Muy de acuerdo | Muy de acuerdo | Muy de acuerdo | De acuerdo |
| 21 | De acuerdo | De acuerdo | En desacuerdo | Muy en desacuerdo |
| 22 | Neutral | De acuerdo | Muy de acuerdo | Neutral |
| 23 | De acuerdo | De acuerdo | Neutral | De acuerdo |
| 24 | Muy de acuerdo | Muy de acuerdo | De acuerdo | Muy de acuerdo |
| 25 | De acuerdo | Muy de acuerdo | Muy de acuerdo | Muy de acuerdo |

Cuadro 8: Respuestas sobre la usabilidad de la plataforma (escala Likert)

| Categoría | Comentarios |
|--------------------|--|
| Interfaz visual | Se sugirió mejorar la paleta de colores y modernizar la interfaz (“los colores son apagados”, “una interfaz más atractiva”). |
| Navegación y flujo | Se recomendó incluir un botón de “siguiente ejercicio” y facilitar el retorno entre niveles. |
| Retroalimentación | Los estudiantes propusieron mostrar el error específico en el código y evitar avanzar con respuestas incorrectas. |
| Funcionalidad | Algunos reportaron problemas técnicos como bloqueos o pérdida de sesión al intentar culminar una lección. |
| Tiempo de lectura | Algunos participantes consideraron que el tiempo de lectura era demasiado largo en relación con el contenido. |

Cuadro 9: Principales respuestas sobre la usabilidad de la plataforma

El Cuadro 10 presenta los resultados de la encuesta sobre el aprendizaje obtenido con la plataforma. Esta sección busca evaluar la percepción de los estudiantes respecto a la comprensión de conceptos de programación, la efectividad de los ejercicios y la adquisición de nuevas habilidades a través de los niveles. En el Cuadro 11 se muestran los temas que los estudiantes identificaron como mejor comprendidos gracias a su uso.

| Usuario | Comprensión de conceptos | Ejercicios refuerzan lo aprendido | Adquisición de nuevas habilidades |
|----------------|---------------------------------|--|--|
| 1 | De acuerdo | De acuerdo | Neutral |
| 2 | Muy de acuerdo | Muy de acuerdo | Muy de acuerdo |
| 3 | Muy de acuerdo | Muy de acuerdo | De acuerdo |
| 4 | Muy de acuerdo | Muy de acuerdo | Muy de acuerdo |
| 5 | De acuerdo | De acuerdo | Muy de acuerdo |
| 6 | De acuerdo | Muy de acuerdo | De acuerdo |
| 7 | Muy de acuerdo | De acuerdo | De acuerdo |
| 8 | De acuerdo | Neutral | Neutral |
| 9 | Muy de acuerdo | Muy de acuerdo | De acuerdo |
| 10 | De acuerdo | De acuerdo | De acuerdo |
| 11 | Muy de acuerdo | Muy de acuerdo | Muy de acuerdo |
| 12 | Muy de acuerdo | Muy de acuerdo | Muy de acuerdo |
| 13 | Muy de acuerdo | Muy de acuerdo | Neutral |
| 14 | De acuerdo | Muy de acuerdo | Muy de acuerdo |
| 15 | De acuerdo | De acuerdo | De acuerdo |
| 16 | Muy de acuerdo | Muy de acuerdo | Muy de acuerdo |
| 17 | De acuerdo | Muy de acuerdo | Muy de acuerdo |
| 18 | Neutral | Muy en desacuerdo | Neutral |
| 19 | De acuerdo | Neutral | De acuerdo |
| 20 | Muy de acuerdo | Muy de acuerdo | Muy de acuerdo |
| 21 | De acuerdo | De acuerdo | Neutral |
| 22 | De acuerdo | Neutral | Neutral |
| 23 | Neutral | De acuerdo | De acuerdo |
| 24 | Muy de acuerdo | Muy de acuerdo | Muy de acuerdo |
| 25 | De acuerdo | Muy de acuerdo | Muy de acuerdo |

Cuadro 10: Respuestas sobre el aprendizaje obtenido mediante la plataforma (escala Likert)

| Categoría | Comentarios |
|-------------------------------------|---|
| Funciones y definición de variables | Varios estudiantes mencionaron haber comprendido mejor el uso de funciones, la declaración de variables y su aplicación práctica en programas simples (“las funciones”, “cómo declarar una variable llamada nombre y asignarle mi nombre”). |
| Ciclos y estructuras de control | Se destacó la mejora en la comprensión de los ciclos <i>while</i> , <i>for</i> y el uso de <i>range</i> , además de las condiciones <i>if</i> y <i>not</i> . |
| Tipos de datos y operadores | Los estudiantes mencionaron una mejor comprensión sobre los tipos de datos, el uso de números enteros y decimales, y el operador módulo. |
| Aprendizaje general | Otros estudiantes señalaron un refuerzo general en todos los temas vistos o en conceptos que no dominaban previamente. |

Cuadro 11: Principales conceptos aprendidos mediante la plataforma

El Cuadro 12 muestra las respuestas de los estudiantes relacionadas con la motivación durante el uso de la plataforma, considerando elementos como los puntos, insignias y niveles. En el Cuadro 13 se presentan los elementos de la gamificación que los estudiantes señalaron como más motivadores.

| Usuario | Puntos e insignias motivaron a avanzar | Estructura por niveles como reto alcanzable | Comparar progreso motivó a mejorar |
|---------|--|---|------------------------------------|
| 1 | Neutral | Neutral | En desacuerdo |
| 2 | Neutral | Muy de acuerdo | Neutral |
| 3 | Muy de acuerdo | De acuerdo | Muy de acuerdo |
| 4 | Muy de acuerdo | Muy de acuerdo | Muy de acuerdo |
| 5 | De acuerdo | De acuerdo | De acuerdo |
| 6 | Neutral | De acuerdo | De acuerdo |
| 7 | De acuerdo | Neutral | Muy de acuerdo |
| 8 | De acuerdo | Muy de acuerdo | De acuerdo |
| 9 | De acuerdo | Muy de acuerdo | De acuerdo |
| 10 | De acuerdo | De acuerdo | De acuerdo |
| 11 | Muy de acuerdo | Muy de acuerdo | Muy de acuerdo |
| 12 | Muy de acuerdo | Muy de acuerdo | Muy de acuerdo |
| 13 | En desacuerdo | Muy de acuerdo | De acuerdo |
| 14 | Muy de acuerdo | Muy de acuerdo | Muy de acuerdo |
| 15 | En desacuerdo | De acuerdo | De acuerdo |
| 16 | De acuerdo | Muy de acuerdo | Muy de acuerdo |
| 17 | Neutral | De acuerdo | De acuerdo |
| 18 | Muy en desacuerdo | Neutral | Neutral |
| 19 | Neutral | En desacuerdo | Neutral |
| 20 | Muy de acuerdo | Muy de acuerdo | Muy de acuerdo |
| 21 | De acuerdo | Neutral | Muy de acuerdo |
| 22 | De acuerdo | De acuerdo | Neutral |
| 23 | De acuerdo | De acuerdo | De acuerdo |
| 24 | Muy de acuerdo | Muy de acuerdo | Muy de acuerdo |
| 25 | Muy de acuerdo | Muy de acuerdo | Muy de acuerdo |

Cuadro 12: Respuestas sobre la motivación de los estudiantes durante el uso de la plataforma (escala Likert)

| Categoría | Comentarios |
|---------------------------------|---|
| Puntos e insignias | Varios estudiantes mencionaron que los puntos e insignias fueron elementos visuales que generaron satisfacción y logro (“las insignias”, “los puntos hacen que quiera superarme”). |
| Retos y niveles | Los estudiantes destacaron la estructura por niveles y los retos como un factor de motivación al permitir avanzar progresivamente y percibir el progreso (“los retos”, “la división de temas por niveles hace menos intimidante la tarea”). |
| Habilidades y progreso personal | Algunos indicaron que las habilidades obtenidas y la comparación con su propio progreso aumentaron la mejora continua (“habilidades”, “el porcentaje de avance completado”). |
| Elementos visuales y diseño | Se valoró el diseño visual de las insignias y títulos de los niveles (“me gustaron las insignias, son visualmente atractivas”). |

Cuadro 13: Elementos de gamificación que resultaron más motivadores para los estudiantes

El Cuadro 14 presenta las respuestas de los estudiantes respecto a la retroalimentación proporcionada por la IA durante el uso de la plataforma. Se evaluó la percepción de utilidad para corregir errores, la claridad de las indicaciones y la confianza en el aprendizaje. En el Cuadro 15 se muestran los comentarios de los estudiantes sobre qué mejoras podrían implementarse en la retroalimentación de la IA.

| Usuario | Claridad de la retroalimentación | Me ayudó a corregir errores | Preferencia de retroalimentación con IA |
|---------|----------------------------------|-----------------------------|---|
| 1 | Muy de acuerdo | De acuerdo | Neutral |
| 2 | De acuerdo | Muy de acuerdo | De acuerdo |
| 3 | Muy de acuerdo | De acuerdo | Muy de acuerdo |
| 4 | Muy de acuerdo | Muy de acuerdo | Muy de acuerdo |
| 5 | Muy de acuerdo | Muy de acuerdo | Muy de acuerdo |
| 6 | De acuerdo | Muy de acuerdo | Muy de acuerdo |
| 7 | Muy de acuerdo | Muy de acuerdo | Muy de acuerdo |
| 8 | De acuerdo | De acuerdo | De acuerdo |
| 9 | De acuerdo | Neutral | Muy de acuerdo |
| 10 | De acuerdo | De acuerdo | De acuerdo |
| 11 | Muy de acuerdo | Muy de acuerdo | Muy de acuerdo |
| 12 | Muy de acuerdo | Muy de acuerdo | Muy de acuerdo |
| 13 | Muy de acuerdo | Muy de acuerdo | Muy de acuerdo |
| 14 | Muy de acuerdo | Muy de acuerdo | Muy de acuerdo |
| 15 | Muy de acuerdo | Muy de acuerdo | Muy de acuerdo |
| 16 | Muy de acuerdo | Muy de acuerdo | De acuerdo |
| 17 | De acuerdo | Muy de acuerdo | Neutral |
| 18 | Neutral | Neutral | De acuerdo |
| 19 | Neutral | Neutral | De acuerdo |
| 20 | De acuerdo | Muy de acuerdo | Muy de acuerdo |
| 21 | Muy de acuerdo | De acuerdo | Muy de acuerdo |
| 22 | En desacuerdo | En desacuerdo | En desacuerdo |
| 23 | De acuerdo | De acuerdo | Neutral |
| 24 | Muy de acuerdo | Muy de acuerdo | Muy de acuerdo |
| 25 | Muy de acuerdo | Muy de acuerdo | Muy de acuerdo |

Cuadro 14: Respuestas sobre la retroalimentación proporcionada por la IA (escala Likert)

| Categoría | Comentarios |
|-----------------------------|--|
| Corrección de errores | Varios estudiantes indicaron que la retroalimentación de la IA les ayudó a identificar y corregir errores, y que no realizarían modificaciones (“Nada, me ayudó a corregir mis errores”, “Considero que está super bien, fue muy útil que me ayudara a corregir mis errores”). |
| Claridad y comprensión | Algunos comentarios sugieren mejoras en la claridad y detalles de las indicaciones de la IA (“Tal vez que resalte dónde está el error”, “Que explique como quedaría corregido con un ejemplo”). |
| Confianza en el aprendizaje | La retroalimentación inmediata generó confianza en el aprendizaje, aunque algunos estudiantes señalaron que no siempre les dio seguridad completa (“Es buena y todo, solo no me da completa confianza”). |
| Sugerencias de mejora | Los estudiantes propusieron incluir retroalimentación más visual, ejemplos adicionales, o explicar el motivo de la corrección para reforzar el aprendizaje (“Me gustaría que explicara un poquito acerca del por qué de esas correcciones”). |

Cuadro 15: Principales comentarios sobre mejoras en la retroalimentación de la IA

El Cuadro 16 presenta las respuestas de los estudiantes respecto a su percepción general de la plataforma, evaluando si cumplió sus expectativas de aprendizaje y si la recomendarían a otros estudiantes. En el Cuadro 17 se muestran los comentarios adicionales y sugerencias que los participantes compartieron sobre el uso de la plataforma.

| Usuario | Plataforma cumplió expectativas | Recomendaría a otros estudiantes |
|----------------|--|---|
| 1 | De acuerdo | Muy de acuerdo |
| 2 | Muy de acuerdo | Muy de acuerdo |
| 3 | Muy de acuerdo | Muy de acuerdo |
| 4 | Muy de acuerdo | Muy de acuerdo |
| 5 | Muy de acuerdo | Muy de acuerdo |
| 6 | Muy de acuerdo | Muy de acuerdo |
| 7 | Muy de acuerdo | Muy de acuerdo |
| 8 | De acuerdo | De acuerdo |
| 9 | Muy de acuerdo | Muy de acuerdo |
| 10 | De acuerdo | De acuerdo |
| 11 | Muy de acuerdo | Muy de acuerdo |
| 12 | Muy de acuerdo | Muy de acuerdo |
| 13 | Muy de acuerdo | Muy de acuerdo |
| 14 | Muy de acuerdo | Muy de acuerdo |
| 15 | De acuerdo | De acuerdo |
| 16 | Muy de acuerdo | Muy de acuerdo |
| 17 | Muy de acuerdo | De acuerdo |
| 18 | Neutral | De acuerdo |
| 19 | Neutral | Neutral |
| 20 | Muy de acuerdo | Muy de acuerdo |
| 21 | Neutral | Neutral |
| 22 | De acuerdo | De acuerdo |
| 23 | De acuerdo | De acuerdo |
| 24 | Muy de acuerdo | Muy de acuerdo |
| 25 | Muy de acuerdo | Muy de acuerdo |

Cuadro 16: Respuestas sobre la percepción general de la plataforma (escala Likert)

| Categoría | Comentarios |
|---------------------------------|--|
| Ejercicios | Algunos estudiantes pidieron más ejercicios o mejoras en la interacción con la IA para resaltar errores (“Más ejercicios”, “que la IA resalte los errores cuando estemos haciendo el código”). |
| Interfaz y diseño | Se sugirieron cambios en los colores y diseño visual para hacer la plataforma más atractiva (“solo cambiaría los colores de la página para que se vea más llamativa y no tan oscura”). |
| Valor y utilidad | Los participantes resaltaron que la plataforma es útil para estudiantes principiantes y que refuerza el aprendizaje de manera entretenida (“En general fue bastante entretenido y lo seguiría jugando para aprender”). |
| Mejoras y sugerencias generales | Otros indicaron mejoras menores como compartir logros, agregar gráficos o animaciones, o ajustar la dinámica para mayor claridad (“Que se pueda compartir los logros o descargarlos”, “agregar algunos gráficos o animaciones para que simule más el ser un juego”). |

Cuadro 17: Comentarios adicionales y sugerencias sobre la plataforma

