

UNIVERSIDAD CATÓLICA SANTO TORIBIO DE MOGROVEJO
FACULTAD DE INGENIERÍA
ESCUELA DE INGENIERÍA DE SISTEMAS Y COMPUTACIÓN



Sistema informático para el proceso enseñanza – aprendizaje de gramática mediante un videojuego educativo personalizado para los estudiantes de segundo grado de secundaria de la IE Serafín Filomeno

**TESIS PARA OPTAR EL TÍTULO DE
INGENIERO DE SISTEMAS Y COMPUTACIÓN**

AUTOR

Jose Leonardo Rodas Vigo

ASESOR

Juan Antonio Torres Benavides

<https://orcid.org/0000-0002-0133-119X>

Chiclayo, 2025

**Sistema informático para el proceso enseñanza – aprendizaje de gramática
mediante un videojuego educativo personalizado para los estudiantes de
segundo grado de secundaria de la IE Serafín Filomeno**

PRESENTADA POR
Jose Leonardo Rodas Vigo

A la Facultad de Ingeniería de la
Universidad Católica Santo Toribio de Mogrovejo
Tesis para optar el título de

INGENIERO DE SISTEMAS Y COMPUTACIÓN

APROBADA POR

Marlon Eugenio Vilchez Rivas
PRESIDENTE

Roger Ernesto Alarcón García
SECRETARIO

Juan Antonio Torres Benavides
VOCAL

Dedicatoria

Dedicado a mis padres: Flor Vigo Luna y Rober Joselito Rodas Suarez; y hermanas: Lizardi Rodas Vigo y Noelia Rodas Vigo por darme la fuerza y apoyo necesario para seguir adelante cuando me sentía derrotado.

Agradecimientos

Agradecimientos a mis padres: Flor Vigo Luna y Rober Joselito Rodas Suarez; y hermanas: Lizardi Rodas Vigo y Noelia Rodas Vigo, y a Tsumugi Kotobuki por siempre estar presentes y creer en mí.

Articulo(Rodas Vigo).docx

INFORME DE ORIGINALIDAD

13%	11%	4%	5%
INDICE DE SIMILITUD	FUENTES DE INTERNET	PUBLICACIONES	TRABAJOS DEL ESTUDIANTE

FUENTES PRIMARIAS

1	hdl.handle.net Fuente de Internet	2%
2	tesis.usat.edu.pe Fuente de Internet	1%
3	Submitted to Universidad TecMilenio Trabajo del estudiante	1%
4	repositorio.unasam.edu.pe Fuente de Internet	<1%
5	trepo.tuni.fi Fuente de Internet	<1%
6	www.researchgate.net Fuente de Internet	<1%
7	repositorio.unicauca.edu.co:8080 Fuente de Internet	<1%
8	Submitted to BENEMERITA UNIVERSIDAD AUTONOMA DE PUEBLA BIBLIOTECA Trabajo del estudiante	<1%
9	revistas.usantotomas.edu.co Fuente de Internet	

Índice

Resumen	6
Abstract	7
Introducción	8
Revisión de literatura	12
Materiales y métodos	14
Resultados y discusión.....	16
Conclusiones.....	37
Resultado:.....	37
Recomendaciones.....	39
Referencias	41
Anexos.....	43

Resumen

El presente proyecto desarrolló un sistema informático en forma de videojuego educativo personalizado para mejorar el proceso de enseñanza-aprendizaje de gramática en estudiantes de segundo grado de secundaria de la Institución Educativa Serafín Filomeno. El videojuego, creado bajo la plataforma Unity, fue diseñado como un juego de rol (RPG) en 2D con estética pixelada, con el objetivo de captar la atención de los estudiantes y motivarlos a aprender temas de gramática, como la tildación y la ortografía, de manera interactiva. Se implementó inteligencia artificial (IA) para ajustar la dificultad de los retos gramaticales según el nivel de conocimiento de cada estudiante, proporcionando retroalimentación inmediata y adaptativa. Durante el proceso de desarrollo, se llevaron a cabo varias iteraciones del videojuego, incluyendo una fase beta con la participación de estudiantes y docentes. Los resultados obtenidos indicaron que los estudiantes mostraron un mayor interés y mejoraron su rendimiento en gramática, demostrando la efectividad del videojuego como herramienta pedagógica.

Palabras clave: Videojuego educativo, gramática, enseñanza personalizada, inteligencia artificial, retroalimentación.

Abstract

This project developed a computer system in the form of a personalized educational video game to improve the teaching-learning process of grammar in second grade high school students at the Serafin Filomeno Educational Institution. The video game, created under the Unity platform, was designed as a 2D role-playing game (RPG) with pixelated aesthetics, with the aim of capturing the attention of students and motivating them to learn grammar topics, such as accentuation and spelling, interactively. Artificial intelligence (AI) was implemented to adjust the difficulty of grammar challenges according to each student's level of knowledge, providing immediate and adaptive feedback. During the development process, several iterations of the video game were carried out, including a beta phase with the participation of students and teachers. The results obtained indicated that the students showed greater interest and improved their performance in grammar, demonstrating the effectiveness of the video game as a pedagogical tool.

Keywords: Educational video game, grammar, personalized teaching, artificial intelligence, feedback.

Introducción

La dificultad de los estudiantes para aprender gramática de manera efectiva se debía a que muchas veces se les presentaba de forma aburrida y monótona. Esto hacía que los estudiantes perdieran el interés y la motivación por aprender gramática, lo que se traducía en un bajo rendimiento académico; el aprendizaje de la gramática se había convertido en un reto dentro del sistema educativo. Según Andrés Alberto, la enseñanza tradicional de la gramática no lograba motivar a los estudiantes, resultando en un bajo nivel de comprensión y aplicación. Este fenómeno se evidenciaba en los resultados de las evaluaciones a nivel internacional y nacional, donde el dominio de la gramática era una de las áreas más deficientes [1],[2], [3].

A escala global, los resultados del programa para la evaluación internacional de estudiantes o informe PISA 2022 mostraron que el rendimiento promedio en lectura no variaba mucho en su media, mostrando que desde 2009 los únicos países latinoamericanos con crecimiento en lectura eran Chile y Paraguay [4]. Esto indicaba dificultades crecientes en comprensión lectora y habilidades gramaticales a nivel global. En Latinoamérica, un estudio realizado por UNESCO en 2017 encontró que el 51% de los estudiantes de 15 años tenían un nivel deficiente en comprensión de lectura [5]. Esto se debía en parte a deficiencias en el aprendizaje de reglas gramaticales. En Perú, la prueba PISA 2022 mostró que el país no había tenido mejoras en el área de lectura y escritura, mientras que países como Paraguay mostraban un crecimiento exponencial cada año [3]. Sólo el 13% de estudiantes peruanos alcanzó los niveles superiores de desempeño de escritura. Esto se relaciona con dificultades en el dominio de la gramática. A nivel local, en Moyobamba, Diaz Aguilar analizó los resultados de la evaluación censal de estudiantes llevada a cabo por el ministerio de educación en Perú en 2019, el estudio encontró que, en la provincia de Moyobamba, el 54% de estudiantes de educación secundaria se ubicó en el nivel de inicio en el área de comunicación (que incluye gramática y redacción)[6], esto se vincula a falta de conocimiento de normas gramaticales básicas. En resumen, los indicadores mostraron dificultades significativas y generalizadas en el aprendizaje de la gramática tanto globalmente como en Perú y Moyobamba.

A nivel de la institución educativa Serafín, La problemática evidente en las aulas es la falta de atención y motivación en los alumnos, ya sea directamente por la falta de métodos de enseñanza llamativos o por el hecho de que los métodos tradicionales no fueron creados

para dar una enseñanza personalizada. Según el maestro Joselito Rodas Suarez, al observar la dinámica en el aula, ha notado una problemática central: la dificultad para mantener la atención y motivación de los estudiantes durante las clases de gramática. Un problema que agrava de forma directa y notoria los calificativos de los estudiantes y genera la percepción compartida de que los métodos tradicionales de enseñanza no sólo son ineficaces, sino también deficientes en su impacto. La resistencia a los métodos tradicionales es evidente, ya que los educadores en muchos casos no logran encontrar nuevos métodos de enseñanza más didácticos o simplemente no tienen la capacidad de adaptación a estos nuevos métodos de enseñanza ocasionando que los alumnos sigan recibiendo educación en una forma que los perjudica y que ya no va acorde con la actualidad donde la tecnología rige muchos aspectos y lo tradicional ya no llama la atención de los estudiantes ya que estos muestran un mayor interés e inclinación hacia la tecnología, en contraste con la apatía hacia las metodologías tradicionales. Este cambio en las preferencias de los estudiantes presenta una oportunidad para replantear la forma en que se aborda la enseñanza de gramática, especialmente considerando la aceptación y el entusiasmo que genera la tecnología entre ellos.

Entre los factores asociados a estas dificultades de aprendizaje se encontró la falta de interés de los estudiantes, así como métodos de enseñanza memorísticos y descontextualizados[7], además la falta de herramientas educativas atractivas y efectivas para enseñar gramática, y la falta de motivación de los estudiantes para aprender gramática[8].

Las causas del problema identificado fueron la falta de innovación en los métodos de enseñanza de gramática, y la falta de aceptación de los videojuegos como herramienta educativa por parte de algunos docentes ya que muchos docentes consideran que los videojuegos son enemigos de la educación, lo que puede limitar su uso en el aula[8].

Las consecuencias examinadas incluyen la carencia de entusiasmo por parte de los estudiantes hacia el aprendizaje de la gramática, así como la ineficacia de los enfoques convencionales de enseñanza[9]. La falta de destreza en la lectoescritura en español entre alumnos universitarios ha llevado a la propuesta de reintegrar la enseñanza específica de la gramática en las aulas. Esto implica la necesidad de crear un enfoque pedagógico en

gramática que estimule la comprensión del lenguaje entre los estudiantes[10].

Aunque las soluciones previamente desarrolladas han demostrado su eficacia, aún no son consideradas como un modelo que la educación tradicional actual esté dispuesta a adoptar y aplicar[8]. Una solución propuesta consistió en la implementación de un videojuego educativo de escritorio enfocado en la narrativa de las culturas de la costa norte del Perú. Se demostró que esta medida tuvo impactos positivos en el progreso del aprendizaje de los estudiantes de primer grado de secundaria, específicamente en el curso de Personal Social de la institución educativa. El aporte de esta herramienta fue la contribución de manera beneficiosa al proceso educativo, proporcionando una experiencia de aprendizaje más inmersiva y atractiva relacionada con las culturas locales. Esta iniciativa tiene el propósito de mejorar significativamente el proceso de enseñanza y aprendizaje[11]. El desarrollo de un juego llamado "Geometry game" creado específicamente para apoyar en el área de matemáticas, buscando demostrar su eficiencia creando dos grupos de alumnos, donde se diseñó y se implementó la funcionalidad del juego en un grupo mientras que el otro aprendía con educación tradicional, se realizaron pruebas y se generaron reportes, demostrando que el primer grupo aprendió las bases matemáticas con más facilidad y rapidez[12].

Luego de ver la gran cantidad de información sobre los métodos de enseñanza desactualizados, el poco interés y poca motivación del estudiante al momento de aprender, se formuló la pregunta:

¿Cómo contribuir en el proceso de enseñanza-aprendizaje de los estudiantes Moyobambinos de segundo grado de nivel secundario de la Institución Educativa Serafín Filomeno en el tema de gramática?

A nivel social, la educación a través del juego es una tendencia que viene cobrando fuerza en los últimos años. Diversos estudios demuestran que el aprendizaje se facilita cuando se presenta de forma lúdica y entretenida[13]. Los videojuegos, en particular, permiten una alta inmersión del jugador en la experiencia educativa[14]. Sin embargo, son escasos los juegos destinados específicamente a la enseñanza de la gramática en español. Esta investigación desarrolló un videojuego de rol (RPG) que enseña gramática a través de una historia interactiva. El beneficiario directo fue la plantilla estudiantil de la institución

educativa Serafín Filomeno para reforzar sus conocimientos gramaticales de forma entretenida. Este proyecto contribuyó a la mejora del dominio de la gramática en la región, aspecto clave para la educación y el desarrollo profesional[9]. Desde el punto de vista tecnológico, sirvió para explorar nuevas formas de gamificación aplicadas a la enseñanza de la gramática.

Desde el punto de vista científico, esta investigación permitió validar la efectividad de los videojuegos como herramienta de enseñanza de la gramática, aportando datos concretos sobre su impacto en el aprendizaje, además, los algoritmos creados para la construcción del videojuego será un gran aporte para el conocimiento científico. Los resultados obtenidos fueron publicados en revistas educativas, enriqueciendo la literatura científica sobre gamificación aplicada a la educación.

En cuanto a la justificación económica, el videojuego educativo representa una solución económicamente accesible y sostenible. Al ser de uso gratuito y no requerir materiales impresos, reduce significativamente los gastos tanto para estudiantes como para instituciones educativas. Los alumnos no necesitan adquirir libros de gramática adicionales ni invertir en plataformas pagas para reforzar sus conocimientos. Asimismo, los docentes cuentan con una herramienta evaluativa y pedagógica sin coste alguno, lo que les permite optimizar su enseñanza sin depender de recursos físicos. Esta accesibilidad económica permite que el aprendizaje de la gramática esté al alcance de cualquier estudiante con una computadora básica, contribuyendo a una educación más equitativa y moderna.

Desde la perspectiva tecnológica, el proyecto empujó a maestros y alumnos a investigar y aplicar tecnologías de vanguardia en materia de videojuegos educativos, con el objetivo de generar contenidos adaptativos y análisis de métricas de aprendizaje. Dicho conocimiento podría replicarse en títulos educativos posteriores, impulsando la industria de serious games en español.

En conclusión, este videojuego representó una innovadora herramienta educativa con beneficios concretos para un amplio público hispanohablante. Los resultados de la investigación podrían extrapolarse para crear juegos que enseñen otros aspectos tanto del idioma como de otras áreas de la educación y contribuir así al fortalecimiento de la educación a nivel global.

Se logró desarrollar un videojuego educativo con un sistema personalizado, basado en inteligencia artificial, que utilice bloques de código con desafíos gramaticales y retroalimentación instantánea, con el propósito de mejorar el proceso de enseñanza-aprendizaje de la gramática en estudiantes, centrado en implementar un sistema de juego dinámico y variado para una experiencia personalizada para cada estudiante se desarrolló un sistema de bloques programados que contengan desafíos gramaticales específicos, la lógica para evaluar respuestas y retroalimentación instantánea luego de haber identificado la mejor técnica de IA que se adecue a las especificaciones del proyecto y así poder implementar una progresión gradual en la dificultad de los desafíos gramaticales, adaptándolos al nivel de conocimiento del estudiante.

Revisión de literatura

La utilización de videojuegos como instrumento educativo ha experimentado un crecimiento notable en los últimos tiempos. Diversas investigaciones han corroborado que los juegos tienen la capacidad de estimular a los estudiantes, contribuyendo así a mejorar su rendimiento académico en diversas disciplinas[15].

En el área de enseñanza de idiomas, los videojuegos también han mostrado resultados positivos. Un estudio de Gee analizó cómo los buenos videojuegos incorporan principios de aprendizaje efectivo que podrían aplicarse a la educación [16]. Entre estos principios se encuentran proporcionar retos asequibles, aprendizaje activo, escalado nivelado, práctica reiterada y retroalimentación inmediata.

Otros autores han explorado el uso de videojuegos para la enseñanza específica de gramática en inglés. Ranalli evaluó un juego para practicar el uso de artículos en inglés y encontró mejoras significativas en el desempeño de los estudiantes[17]. Sykes y Reinhardt desarrollaron un juego de rol para práctica de vocabulario y estructuras gramaticales, reportando alta motivación y compromiso de los alumnos[18].

En cuanto al español, Grimshaw y Cardoso analizaron el juego ¡Zombies! Aliens! Robots! Oh My!, diseñado para practicar vocabulario y gramática en español. Los autores

concluyeron que el juego promovía la práctica efectiva al proveer contextos significativos y requerir uso activo del lenguaje objetivo.

Estos estudios demostraron el potencial de los videojuegos para complementar la enseñanza de gramática de manera atractiva y efectiva. Queda espacio para más investigación sobre cómo diseñar juegos óptimos según los objetivos y nivel de los estudiantes.

Existen antecedentes más locales como el caso de El "Geometry Game", un juego educativo de varios jugadores involucra a un participante inmerso en diferentes áreas donde sabios presentan enigmas para que el jugador gane puntos y pueda mejorar su personaje. El jugador debe asumir un papel activo mediante la manipulación de su personaje, y, según la cantidad de desafíos resueltos, podrá avanzar al siguiente nivel utilizando un globo para viajar a otras áreas del juego.

El aprendizaje que usa de base a los juegos se enfoca en aplicar los principios del mismo para enseñar contenido lúdico educativo, también conocido como gamificación. Esta metodología combina el constructivismo, que implica aprender mediante la práctica, con la mecánica de juego para convertir material educativo en una experiencia lúdica. La investigación se centró en la gamificación de una asignatura de inglés para ingeniería a través de un videojuego serio llamado The Engineer. Este prototipo, destinado a ingenieros industriales, tiene como finalidad impartir instrucción de inglés técnico mediante 6 misiones que poseen objetivos específicos. El artículo expone los fundamentos de los videojuegos educativos y la gamificación en el ámbito del aprendizaje de lenguas extranjeras, proporcionando detalles acerca de la creación, características y contenidos de "The Engineer". En síntesis, la investigación presenta un videojuego educativo destinado a ingenieros con el propósito de enseñarles inglés técnico, desarrollado como parte de un proyecto respaldado por la Generalitat de Valencia (España).

Asimismo, existe una investigación que aborda la situación actual de la educación en Perú, resaltando la carencia de integración de las Tecnologías de la Información y Comunicación (TICs) en el entorno educativo. Sugiere que, para lograr un impacto más

significativo, es crucial evidenciar los beneficios de incorporar estas tecnologías en la formación de los estudiantes. Adicionalmente, sugiere la integración de videojuegos educativos como un recurso complementario al proceso de enseñanza, tomando como referencia la situación específica de la Institución Educativa Aplicación N°10836 en el distrito de José Leonardo Ortiz. Esta institución enfrenta desafíos relacionados con el desempeño académico de los estudiantes en el área de Personal Social.

Materiales y métodos

Para el proyecto, la investigación más acertada sería la aplicada ya que la investigación aplicada se refiere a la producción de trabajos auténticos con el fin de obtener nuevos saberes, aunque su enfoque principal se orienta hacia un propósito o meta práctica particular [25].

Para la investigación que se propone, que busca desarrollar un sistema informático (un videojuego RPG) con el propósito de mejorar el proceso de enseñanza-aprendizaje de la asignatura de Comunicación, se inclina hacia un enfoque más aplicado. Aunque el desarrollo de un videojuego educativo para mejorar la enseñanza de gramática implica la implementación de conceptos educativos y pedagógicos, se considera una investigación aplicada. La razón principal es que el enfoque del proyecto está orientado hacia la aplicación práctica del conocimiento teórico en un contexto específico, con el objetivo de resolver un problema educativo concreto[25].

Las características que indican que se trata de una investigación aplicada incluyen:

- **Finalidad Práctica:** El objetivo principal es perfeccionar el proceso de enseñanza y aprendizaje de la gramática a través de un videojuego, con aplicaciones directas en un entorno educativo.
- **Aplicación de Conocimientos:** Se emplea el conocimiento teórico sobre la gramática y los principios del diseño educativo para desarrollar un videojuego con propósitos educativos específicos.
- **Orientación a la Solución de Problemas Concretos:** La intención es mejorar la tasa de aprobación en gramática, proporcionar retroalimentación, y facilitar el proceso de enseñanza del docente.

Aunque hay un componente de investigación básica en la integración de la gramática en el diseño del videojuego, el enfoque general parece estar más orientado hacia la aplicación

directa de la teoría educativa para resolver un problema específico en el contexto de enseñanza y aprendizaje.

1. Métodos de investigación

TABLA I

Método	Sustento por el cual será empleado en la investigación
Deductivo	Para hacer un razonamiento lógico de la premisa general de la institución y llegar a conclusiones específicas.
Implementación	Servirá para aplicar y llevar a cabo una estrategia, plan y programa de manera efectiva.
Inductivo	Para inferir principios generales a partir de observaciones específicas y casos particulares.

TABLA II
TECNICAS DE INVESTIGACIÓN

Técnicas	Instrumento
Pruebas Piloto	Listas de verificación y observación para evaluar la jugabilidad
Opinión de especialistas	Registros verificables
Observación de participante	Un diario de campo que documente observaciones directas, interacciones clave, y reflexiones sobre el entorno educativo

2. Metodología de desarrollo

Como se mencionó en las bases teóricas, la metodología que se utilizará será la SUM, con los siguientes pasos:

Roles

Maneja 4 roles que serán distribuidos entre el desarrollador del sistema y los usuarios finales: equipo de desarrollo, productor interno, cliente y verificador beta. Donde equipo de desarrollo y productor interno sería llevado por mi persona, mientras que cliente y verificador beta serían los roles de los usuarios finales, en este caso los docentes y estudiantes.

Ciclo de Vida

El ciclo de vida de la metodología se desarrolla en 5 fases interactivas y secuenciales a excepción de la fase de gestión de riesgos que sirve para poder reaccionar a cualquier inconveniente por lo tanto esta fase se ejecuta durante todo el proyecto.

Concepto: En esta fase se define el videojuego, se aclara el estilo, el tipo, la jugabilidad, público al que va dirigido entre otros puntos. Para el caso se definió el juego como un RPG o juego de rol con una jugabilidad 2D vista desde arriba y con un público objetivo de estudiantes de secundaria a los que se les debe enseñar gramática.

Planificación: en esta fase se debe el definir el cronograma que seguirá el proyecto, con fechas y objetivos exactos, buscando también el formalizar las características funcionales y no funcionales que el proyecto requerirá.

Elaboración: ya con la fase de planificación bien desarrollada se puede empezar con el desarrollo del videojuego, buscando siempre cumplir los objetivos y fechas del cronograma y buscando una relación de apoyo con los usuarios finales para poder hacer todas más mejoras pertinentes en su momento.

Beta: Es la fase en la que se buscará más interacción con los usuarios, ya que estos serán encargados de probar el videojuego en un estado beta, esta fase se aplica para poder recibir retroalimentación directa de los usuarios y poder corregir errores o bugs que no se había hecho visibles hasta ese momento.

Cierre: Luego de la fase Beta y haber corregido todos los errores encontrados, se pasa a la fase final que es la de cierre o despliegue donde se dará el videojuego ya terminado para ser usado directamente por los estudiantes y docentes.

Resultados y discusión

I. RESULTADOS

El videojuego se incluyó en el género de RPG (Role playing game) referente a juego de rol donde el estudiante hará el rol de aventurero que visita diferentes lugares para resolver diversos retos referentes a gramática.

Para el ahorro y buen uso de recursos el juego fue creado en 2D con vista desde arriba,

bajo la estética de 8bits tanto para la sección de sonido como la sección visual, es un apartado artístico que usa el pixel art para mostrar todo lo que el juego quiere ofrecer, donde los colores son representados a través de píxeles y así se ahorra recursos al momento de ejecutar el juego, en este caso el juego se representará en un pixel art de 16x16 para mantener una estética armoniosa y agradable para el jugador

Estudiante

Desde el apartado del estudiante, se presenta una pantalla desde donde puede acceder y manejar los apartados de:

Jugar: en esta pantalla se muestran los temas de gramática que el juego ofrece, los cuales son tildación y ortografía. Luego de elegir alguna de las opciones, pasa a un mapa interactivo donde selecciona el subtema relacionado con su anterior elección. Al elegir alguna de las ciudades del mapa interactivo, procede a jugar dentro de ella, pudiendo moverse e interactuar con los NPC de la ciudad. Estos le dan diferentes retos gramaticales según el tema seleccionado y dependiendo de su conocimiento. El estudiante debe resolver y superar los retos para finalmente encontrarse con una "estatua sagrada" que le ofrece un último reto gramatical para finalizar la sesión.

Mi progreso: Este apartado muestra el desempeño del estudiante en los diferentes temas que trata el videojuego. El estudiante puede filtrar los datos por tema o subtema para tener más claridad al analizar los datos, los cuales se presentan en gráficos que puede interpretar e inferir.

Objetivos	Crear interfaces fáciles de entender e interactuar para una fácil navegación entre las pantallas del juego
	Crear minijuegos que suban su dificultad dependiendo del nivel de conocimientos y habilidad del jugador
	Proveer retroalimentación instantánea referente al fallo si el jugador falla en algún minijuego
Genero de Videojuego	RPG (Role-Playing Game) o Juego de rol en español
GAMEPLAY	
Acción	Reacción
Click izquierdo del ratón	Acción / interacción con los botones del videojuego
Presionar W	Mover jugador hacia arriba
Presionar A	Mover jugador a la derecha
Presionar S	Mover jugador hacia abajo
Presionar D	Mover jugador a la izquierda
Presionar R	Interactuar con NPC cercano al jugador

Presionar ESPACIO	Aumenta la velocidad del jugador
<u>Esc</u>	Mostrar opciones de navegación desde dentro de los mapas
Mover ratón	Navegar entre las opciones
CARACTERISTICAS	
Gráficos	Pixel Art <ul style="list-style-type: none"> • Pantallas en las que se interactúa • Minijuegos • Animaciones de jugador
Jugabilidad e interacción	Uso de teclado y ratón para interacción con el videojuego
Sonido	<ul style="list-style-type: none"> • Música ambiental de uso libre

	<ul style="list-style-type: none"> • Genero 8bits
Roles	
Existe un personaje principal al que se puede controlar a través del teclado y con el cual se puede mover entre mapas y en los mapas para interactuar con aldeanos y completar minijuegos	
Historia	
No existe	
Ambientación	
Diferentes ciudades con estilos medievales	
Público objetivo	
Estudiantes de tercer grado de educación secundaria de la I.E Serafín Filomeno	

1.2. PLANIFICACIÓN

En esta fase se presentó los requerimientos que funciones y no funcionales del videojuego la secuencia de acciones que se realizaran para la ejecución de juego, desde su creación hasta su despliegue y entrega final.

Alcance

Desarrollo del videojuego

Para el correcto desarrollo del videojuego se tuvo en cuentas los siguientes puntos:

Software de desarrollo

Para el proyecto se usó la plataforma de creación de videojuegos unity 2D por su vasta documentación, popularidad y estándares y su gran variedad de apoyos e facilidades al momento de desarrollar el videojuego.

Base de datos

Se usó los servicios de RDS (Relational database service) de AWS (Amazon web services) los cuales son un servicio proporcionado por Amazon que ofrece velocidad, seguridad y estabilidad al momento de usarlas, se usara la versión gratuita durante la fase beta para minimizar costos.

Contenido educativo

Temas

Los temas que abarcó el videojuego serán directamente relacionados a gramática, los temas y subtemas a tratar son:

Tildación:

- Tildación general
- Tildación especial

Ortografía de letras:

- Uso de B y V
- Uso de S, C y Z
- Uso de G y J

Metodología

La metodología de enseñanza se basó en la retroalimentación de conceptos ya aprendidos en clase ya que el videojuego no se encargará de educar, sino de poner a prueba los conocimientos adquiridos en clase, y de ser el caso, proporcionará retroalimentación.

Personalización

La IA que se incluyó en el videojuego se encargará de proporcionar retos gramaticales a los estudiantes según su nivel de conocimientos sobre el tema, ya que escogerá de un banco de retos los mejores dependiendo de la dificultad que el estudiante pueda afrontar.

Implementación

Para la implementación se necesitó una computadora de sobre mesa o una laptop que tenga características iguales o superiores a estas:

- Procesador Intel Core i5-8250U de 1.60Hz.
- Disco duro con espacio libre de 5Gb.
- Memoria RAM de 4gb.
- Sistema operativo Windows 10.

Plataforma	
Sistema operativo	Windows 10
CPU	Intel Core i5-8250U de 1.60Hz.
Memoria RAM	4 gb
almacenamiento	1 gb
Herramientas y tecnologías	
Motor de videojuego	Unity 2D versión 2022.3.18f1

Lenguaje de programación	C#
Método de entrada - inputs	Teclado y ratón

Requerimientos funcionales

Estudiante

- El videojuego presenta al jugador una serie de retos gramaticales, como completar frases, corregir errores gramaticales, identificar partes del discurso o directamente cuestionarios.
- El sistema de IA analiza el desempeño del jugador en los retos y ajustar dinámicamente la dificultad de los siguientes retos en función del nivel de conocimiento de gramática del jugador.
- El jugador empieza en un nivel intermedio de dificultad y este se adaptará a su nivel de conocimientos después de cada sesión gracias a la IA.
- El juego proporciona retroalimentación inmediata sobre la precisión de las respuestas del jugador y ofrecer explicaciones para mejorar en caso de errores, esta

retroalimentación

sucede luego de que el estudiante falle algún reto gramatical e irá directamente dirigido hacia su error.

- Debe haber un sistema de revisión de progresión que permita al jugador revisar su propio desempeño dentro del juego y como este ha ido evolucionando.

Docente

- El videojuego debe proporcionarle información concreta sobre el desempeño de cada uno de sus estudiantes a través de gráficos y calificativos.

- El videojuego debe proporcionar informes detallados con las calificaciones de todos los estudiantes asignados a un mismo docente.

Requerimientos no funcionales

- **Rendimiento:** El juego es capaz de ajustar la dificultad de los retos en tiempo real sin provocar retrasos o interrupciones en la experiencia del jugador.

- **Usabilidad:** La interfaz de usuario es intuitiva y fácil de entender, permitiendo al jugador centrarse en los retos gramaticales sin distraerse con controles complicados.

- **Adaptabilidad:** La IA es capaz de adaptarse a diferentes estilos de aprendizaje y velocidades de progreso de los jugadores, ofreciendo una experiencia personalizada para cada uno.

- **Seguridad:** Los datos del jugador, como su progreso y desempeño en los retos, son almacenados de manera segura y protegidos contra accesos no autorizados.

- **Escalabilidad:** El juego es capaz de manejar un número variable de jugadores y adaptarse a cambios en la carga de usuarios sin comprometer su rendimiento o funcionalidad.

1.3. ELABORACIÓN

1.3.1. Iteración 1: Preparación del proyecto

Etapa 1: Descripción

Se creó un proyecto con las características necesarias para la creación de un videojuego 2D RPG y dentro del mismo crear un ambiente ordenado para poder tener un mejor manejo de toda la información del mismo, y no generar desorden con la creación de nuevos archivos dentro del mismo.

Etapa 2: Desarrollo

Al momento de la creación del proyecto se seleccionó la opción de 2D que unity mismo proporciona para que durante la elaboración se pueda tener las herramientas justas para el

desarrollo

Dentro del proyecto se creó carpetas para almacenar la información que se llegue utilizar, estas carpetas fueron creadas dentro de la carpeta de Assets ya que es la carpeta principal que unity usa para almacenar todos los archivos referentes al desarrollo. Se creó carpetas como:

- **Animaciones:** Para almacenar las animaciones que se creen, ya sea del jugador, del entorno o de las pantallas.
- **AssetsFree:** Para almacenar los archivos descargados de internet y de uso libre
- **Fuentes:** Para guardar las fuentes que se utilizaron en la realización del proyecto
- **MUSICA:** Para guardar todas las pistas de audio que se usaron.
- **Scenes:** Para almacenar las escenas que se crearon durante el proyecto.
- **Scripts:** Para almacenar los scripts que se crearon.

1.3.2. Iteración 2: Creación de Base de Datos

Etapa 1: Descripción

Se creo un base de datos para el almacenamiento de los datos del videojuego

Etapa 2: Desarrollo

Para este proyecto se creo la base de datos en Postgresql y se subirá a una base de datos en la nube, en este caso RDS de AWS.

```

1
2 -- Tabla "maestro"
3 CREATE TABLE maestro (
4   id SERIAL PRIMARY KEY,
5   usuario VARCHAR(20) NOT NULL,
6   contrasena VARCHAR(8) NOT NULL
7 );
8
9 -- Tabla "tildacion" con puntuación generada y columnas renombradas
10 CREATE TABLE tildacion (
11   id SERIAL PRIMARY KEY,
12   tGeneral INTEGER,
13   tEspecial INTEGER,
14   puntuacion NUMERIC(5, 2) GENERATED ALWAYS AS ((tGeneral + tEspecial) / 2.0) STORED
15 );
16
17 -- Tabla "ortografia"
18 CREATE TABLE ortografia (
19   id SERIAL PRIMARY KEY,
20   bv INTEGER,
21   scz INTEGER,
22   gj INTEGER,
23   puntuacion NUMERIC(5, 2) GENERATED ALWAYS AS ((bv + scz + gj) / 3.0) STORED
24 );
25
26 -- Tabla "gramatica"
27 CREATE TABLE gramatica (
28   id SERIAL PRIMARY KEY,
29   id_tildacion INTEGER,
30   id_ortografia INTEGER,
31   puntuacion NUMERIC(5, 2),
32   FOREIGN KEY (id_tildacion) REFERENCES tildacion(id),
33   FOREIGN KEY (id_ortografia) REFERENCES ortografia(id)
34 );
35
36 -- Trigger para actualizar la puntuacion en "gramatica"
37 CREATE OR REPLACE FUNCTION actualizar_puntuacion_gramatica()
38 RETURNS TRIGGER AS $$
39 BEGIN
40   NEW.puntuacion = (
41     (SELECT puntuacion FROM tildacion WHERE id = NEW.id_tildacion) +
42     (SELECT puntuacion FROM ortografia WHERE id = NEW.id_ortografia)
43   ) / 2.0;
44   RETURN NEW;
45 END;
46 $$ LANGUAGE plpgsql;
47
48 CREATE TRIGGER trigger_actualizar_gramatica
49 BEFORE INSERT OR UPDATE ON gramatica
50 FOR EACH ROW
51 EXECUTE FUNCTION actualizar_puntuacion_gramatica();
52
53 -- Tabla "estudiante"
54 CREATE TABLE estudiante (
55   id SERIAL PRIMARY KEY,
56   usuario VARCHAR(20) NOT NULL,
57   contrasena VARCHAR(8) NOT NULL,
58   id_maestro INTEGER,
59   id_gramatica INTEGER,
60   FOREIGN KEY (id_maestro) REFERENCES maestro(id),
61   FOREIGN KEY (id_gramatica) REFERENCES gramatica(id)
62 );
63
64

```

1.3.3. Iteración 3: Importación de Assets

Etapa 1: Descripción

Se necesitaron de diferentes assets para poder realizar el proyecto, entre ellos los assets comprados desde la tienda oficial de unity, los cuales fueron importados al proyecto

Etapa 2: Desarrollo

Dentro del proyecto se da la facilidad de ingresar con cuenta propia de unity y ya que la compra hecha en la tienda oficial esta relaciona a mi cuenta solo tengo que llamar los assets que deseo importar

1.3.4. Iteración 4: Pantallas de usuario Alumno

Etapa 1: Descripción

Se implementó las pantallas de usuario del alumno con su respectivo apartado gráfico,

su estética y la distribución de las opciones que la pantalla.

Etapa 2: Desarrollo

Para la creación de pantallas o interfaces unity 2D ofrece muchas formas de crearlas, pero para la optimización de este proyecto se usó la opción de scene ya que estos crean canvas en las que se puede usar imágenes, texto, botones y demás opciones interactivables a través del uso del mouse para crear interfaces y la opción de isometric rule Tile para la creación de mapas explorables ya que estos crean un Grid que funciona como un lienzo con capas en el que se puede trabajar con assets para la creación de mapas en los que el jugador pueda moverse.

1.3.5. Iteración 5: Navegación entre pantallas

Etapa 1: Descripción

Se implementó la navegación entre pantallas para que el usuario pueda moverse entre ellas con las opciones que ellas mismas ofrecen.

Etapa 2: Desarrollo

En caso de que la pantalla haya sido creada a través de un canvas solo es necesario la creación de un botón al que se le debe cambiar el aspecto gráfico y el texto para dar información exacta de su uso.



Para poder darle la navegabilidad requerida a cada botón se creó un script dentro del proyecto con extensión .cs el cual se trabajará con el lenguaje C# en el que se programará la navegabilidad entre las pantallas.

Este código fue asignado al canvas en el que se quería utilizar para que los botones pertenecientes a ese canvas puedan llamar la función creada y así al dar click en el botón este te lleve a la pantalla deseada.

Para poder tener el índice de cada mapa estos fueron agregados a la construcción de escenas. Esto le da un índice a cada pantalla para poder ser llamada.

En caso de ser Grid, la interacción tiene pasos extras ya que se tiene que crear un canvas dentro del Grid para poder almacenar el botón con el que se quiere interactuar.


1.3.6. Iteración 6: Animaciones de pantalla

Etapa 1: Descripción

El videojuego necesitó animaciones en diferentes secciones para no verse estático y poco llamativo para los estudiantes, para solucionar esto se utilizará una librería de uso libre que otorga animaciones en canvas

Etapa 2: Desarrollo

Se exportó la librería LeanTween, es una librería que se utiliza para dar animaciones de diferentes tipos a algunos objetos ya sean pertenecientes a un canvas o a un Grid.



```

1 private void Start()
2 {
3     // Animaciones de inicio
4     LeanTween.moveY(Titulo.GetComponent<RectTransform>(), -140, 1.5f).setEase(LeanTweenType.easeOutBounce);
5     LeanTween.moveX(Jugador.GetComponent<RectTransform>(), -40, 1.5f).setDelay(1.5f).setEase(LeanTweenType.easeOutBounce);
6     LeanTween.moveX(Jugadora.GetComponent<RectTransform>(), 1155, 1.5f).setDelay(1.5f).setEase(LeanTweenType.easeOutBounce);
7
8     // Asegúrate de que el Canvas está asignado a la cámara principal
9     if (menuCanvas.renderMode == RenderMode.ScreenSpaceCamera)
10    {
11        menuCanvas.worldCamera = Camera.main;
12    }
13
14    // Inicialmente, el menú está inactivo
15    MenuExtra.SetActive(false);
16    FondoNegro.SetActive(false);
17 }

```

Para su Uso se crea un script donde se llama a la librería y luego se crea instancias de animación para cada objeto que se quiera animar, el script creado es consumido por el canvas que contiene los objetos que se quiere animar para poder llamar la instancia o función exacta para cada objeto.

1.3.7. Iteración 7: Movimiento del Jugador

Etapa 1: Descripción

El personaje jugable tiene movilidad y animaciones para esa movilidad y así verse más atractivo para los jugadores

Etapa 2: Desarrollo

Para esto se escogió un asset de personaje que tenga los sprites de animación de

movimientos, estos sprites se ordenan a través de la interfaz de unity para poder elegir la velocidad de animación y la secuencia de sprites.

Con las animaciones listas se creó un script para lograr que el personaje se mueva al presionar WASD o las flechas en el teclado, se puede asignar ambas ya que unity al ser un motor de creación de videojuegos tiene comandos prefabricados para el movimiento del jugador.

```

1  public class MovimientoJugador : MonoBehaviour
2  {
3      [SerializeField] private float velocidad = 3f;
4      private Rigidbody2D jugadorRB;
5      private Vector2 movimientoInput;
6      private Animator animacionesJugador;
7      void Start()
8      {
9          jugadorRB = GetComponent<Rigidbody2D>();
10         animacionesJugador = GetComponent<Animator>();
11     }
12
13     void Update()
14     {
15         float movimientoX = Input.GetAxisRaw("Horizontal");
16         float movimientoY = Input.GetAxisRaw("Vertical");
17         movimientoInput = new Vector2(movimientoX, movimientoY).normalized;
18
19         animacionesJugador.SetFloat("Horizontal", movimientoX);
20         animacionesJugador.SetFloat("Vertical", movimientoY);
21         animacionesJugador.SetFloat("Velocidad", movimientoInput.sqrMagnitude);
22     }
23
24     private void FixedUpdate()
25     {
26         jugadorRB.MovePosition(jugadorRB.position + movimientoInput * velocidad * Time.fixedDeltaTime);
27     }
28 }
29 }
30

```

1.3.8. Iteración 8: Movimiento de la cámara

Fase 1: Descripción

La cámara debe seguir al personaje mientras este se mueve por el mapa creado y no debe perderlo de vista en ningún momento

Fase 2: Desarrollo

Para lograr esto se creó un script donde se especifica a la cámara que no debes perder de vista al objeto jugador, y debe seguirlo mientras se mueve, esto se logró actualizando la posición de la cámara cada segundo, así la cámara se moverá si siente que el personaje dejó de estar en el centro de la pantalla, al igual que dejara de moverse si está muy cerca de algún límite de la pantalla, así evitando que se vean zonas que no deberían verse y

priorizando los límites del mapa sobre el mantener al personaje en el centro.

1.3.9. Iteración 9: Objetivo específico 1 - Implementar un sistema de juego dinámico y variado para una experiencia personalizada para cada estudiante.

Fase 1: Descripción

Cada mapa dentro del videojuego tiene diversos minijuegos que se muestran de forma aleatoria y variada adaptando su dificultad al nivel de conocimientos del jugador

Fase 2: Desarrollo

De cara al desarrollo se han creado 3 minijuegos cumpliendo los requerimientos del objetivo específico, cada minijuego tiene su propia forma individual de resolverse, pero siguen una misma lógica en su armado. La lógica es la siguiente:

Importaciones: se hacen la importación de las librerías a utilizar:

Variables de instancia: Estas variables son componentes y datos necesarios para el funcionamiento del minijuego. SerializeField permite que estas variables sean visibles y configurables en el editor de Unity.

```

1  [SerializeField] private GameObject minijuegoPanel;
2  [SerializeField] private GameObject dialogoPanel;
3  [SerializeField] private TextMeshProUGUI textDialogo;
4  [SerializeField] private TextMeshProUGUI textExplicacion;
5  [SerializeField] private TextMeshProUGUI textCorrecto;
6  [SerializeField] private Button[] letterButtons;
7  [SerializeField] private Color correctColor;
8  [SerializeField] private Color incorrectColor;
9
10 [SerializeField] private MovimientoJugador movimientoJugador;
11 [SerializeField] private BaseDeDatosPalabrasExplicacion baseDeDatosPalabrasExplicacion;
12
13 private List<string> palabras;
14 private List<string> palabrasCorrectas;
15 private List<string> explicaciones;
16 private int palabraActual = 0;
17

```

Método start: Este método se llama al iniciar el juego. Inicializa el minijuego y desactiva el movimiento del jugador.

```

1 void Start()
2 {
3     InicializarMinijuego();
4     movimientoJugador.enabled = false;
5 }

```

Método InicializarMinijuego: Inicializa las variables y configura el texto del diálogo inicial. Obtiene las palabras, sus versiones correctas y las explicaciones de la base de datos.

```

1 public void InicializarMinijuego()
2 {
3     textDialogo.text = "Haz clic en la letra que debe llevar la tilde.";
4     textExplicacion.text = "";
5     textCorrecto.text = ""; // Limpiar "Correcto" al inicio
6     palabraActual = 0;
7
8     (palabras, palabrasCorrectas, explicaciones) = baseDeDatosPalabrasExplicacion.ObtenerPalabrasPorNivel(PuntuacionJugador.instancia.PuntosJugador);
9
10    SeleccionarPalabrasAleatorias();
11    CargarPalabra(palabraActual);
12 }
13
14

```

Luego, selecciona palabras aleatorias y carga la primera palabra.

Método SeleccionarPalabrasAleatorias: Selecciona aleatoriamente tres palabras y sus correspondientes versiones correctas y explicaciones.

```

1 private void SeleccionarPalabrasAleatorias()
2 {
3     List<string> seleccionadas = new List<string>();
4     List<string> seleccionadasCorrectas = new List<string>();
5     List<string> seleccionadasExplicaciones = new List<string>();
6     System.Random random = new System.Random();
7
8     while (seleccionadas.Count < 3 && palabras.Count > 0)
9     {
10        int index = random.Next(palabras.Count);
11        seleccionadas.Add(palabras[index]);
12        seleccionadasCorrectas.Add(palabrasCorrectas[index]);
13        seleccionadasExplicaciones.Add(explicaciones[index]);
14        palabras.RemoveAt(index);
15        palabrasCorrectas.RemoveAt(index);
16        explicaciones.RemoveAt(index);
17    }
18
19    palabras = seleccionadas;
20    palabrasCorrectas = seleccionadasCorrectas;
21    explicaciones = seleccionadasExplicaciones;
22 }
23

```

Lógica individual de cada minijuego: Luego de preparar e iniciar el juego se programa su lógica de ejecución única e individual.

```

1  private void CargarPalabra(int index)
2  {
3      string palabra = palabras[index];
4      for (int i = 0; i < letterButtons.Length; i++)
5      {
6          if (i < palabra.Length)
7          {
8              letterButtons[i].GetComponentInChildren<TextMeshProUGUI>().text = palabra[i].ToString();
9              letterButtons[i].gameObject.SetActive(true);
10             letterButtons[i].GetComponent<Image>().color = Color.white;
11             letterButtons[i].onClick.RemoveAllListeners();
12
13             int buttonIndex = i;
14             letterButtons[i].onClick.AddListener(() =>
15
16 private int GetCorrectIndex(string palabra, string palabraCorrecta)
17 {
18     for (int i = 0; i < palabra.Length; i++)
19     {
20         if (palabra[i] != palabraCorrecta[i])
21         {
22             return i;
23         }
24     }
25     return -1;
26 }
27
28 private void AjustarPuntos(bool acierto)
29 {
30     int puntosPorPalabra = 0;
31
32     if (PuntuacionJugador.instancia.PuntosJugador < 25)
33     {
34         puntosPorPalabra = acierto ? 3 : -5;
35     }
36     else if (PuntuacionJugador.instancia.PuntosJugador < 75)
37     {
38         puntosPorPalabra = acierto ? 5 : -5;
39     }
40     else
41     {
42         puntosPorPalabra = acierto ? 10 : -7;
43     }
44
45     PuntuacionJugador.instancia.PuntosJugador += puntosPorPalabra;
46 }

```

1.3.10. Iteración 10: Objetivo específico 2 - Desarrollar bloques de código que contengan desafíos gramaticales específicos, la lógica para evaluar respuestas y retroalimentación instantánea

Fase 1: Descripción

Los minijuegos deben ser creado con una lógica para poder mostrar un desafío gramatical sobre un tema específico, la lógica para evaluar las respuestas y la retroalimentación instantánea.

Fase 2: Desarrollo

Como se explicó en la iteración anterior todos los minijuegos tienen un armado similar, solo varían cosas referentes a la lógica individual de cada uno, pero a grandes rangos cumplen los requerimientos del objetivo específico:

Lógica para evaluar respuestas: en este bloque se toma la puntuación del jugador que

```

1 private void SeleccionarPalabrasAleatorias()
2 {
3     List<string> seleccionadas = new List<string>();
4     List<string> seleccionadasCorrectas = new List<string>();
5     List<string> seleccionadasExplicaciones = new List<string>();
6     System.Random random = new System.Random();
7
8     while (seleccionadas.Count < 5 && palabras.Count > 0)
9     {
10        int index = random.Next(palabras.Count);
11        seleccionadas.Add(palabras[index]);
12        seleccionadasCorrectas.Add(palabrasCorrectas[index]);
13        seleccionadasExplicaciones.Add(explicaciones[index]);
14        palabras.RemoveAt(index);
15        palabrasCorrectas.RemoveAt(index);
16        explicaciones.RemoveAt(index);
17    }
18
19    palabras = seleccionadas;
20    palabrasCorrectas = seleccionadasCorrectas;
21    explicaciones = seleccionadasExplicaciones;
22 }
23

```

esta almacenada en otro script y se modifica según el jugador haya errado o acertado el minijuego

```

1 private void AjustarPuntos(bool acierto)
2 {
3     int puntosPorPalabra = 0;
4
5     if (PuntuacionJugador.instancia.PuntosJugador < 25)
6     {
7         puntosPorPalabra = acierto ? 3 : -5;
8     }
9     else if (PuntuacionJugador.instancia.PuntosJugador < 75)
10    {
11        puntosPorPalabra = acierto ? 5 : -5;
12    }
13    else
14    {
15        puntosPorPalabra = acierto ? 10 : -7;
16    }
17
18    PuntuacionJugador.instancia.PuntosJugador += puntosPorPalabra;
19 }
20

```

Retroalimentación: Al momento de armar la base de datos de donde se sacarán las palabras que mostrara el minijuego también se considera agregar la explicación de cada una de las palabras por si el jugador falla en esa palabra especifica al momento de desarrollar el videojuego. Esta explicación es llamada junto con la palabra seleccionada al momento de seleccionar palabras aleatorias ara el minijuego

La base de datos tiene este armado:

```

1 public List<string> palabrasDificiles = new List<string> { "Electron", "Cardiaco", "Telefonico", "Quimico", "Fisico" };
2 public List<string> palabrasDificilesCorrectas = new List<string> { "Electrón", "Cardiaco", "Telefónico", "Químico", "Físico" };
3 public List<string> explicacionesDificiles = new List<string>
4 {
5     "Electrón lleva tilde porque es una palabra aguda terminada en consonante 'n'.",
6     "Cardiaco lleva tilde porque es una palabra esdrújula.",
7     "Telefónico lleva tilde porque es una palabra esdrújula.",
8     "Químico lleva tilde porque es una palabra esdrújula.",
9     "Físico lleva tilde porque es una palabra esdrújula."
10 };
11

```

1.3.11. Iteración 11: Objetivo específico 3 - Implementar una progresión gradual en la dificultad de los desafíos gramaticales, adaptándolos al nivel de conocimiento del estudiante

Fase 1: Descripción

Los minijuegos deben regular su dificultad según el nivel de conocimientos del jugador, el estado al momento de calificar a un estudiante maneja tres niveles: en inicio, en proceso y destacado, Revisar anexo 3. Estos niveles son los niveles que el videojuego toma en cuenta y separa la complejidad de las palabras en 3 secciones, cada nivel tiene su propio conjunto de palabras y de calificación.

Fase 2: Desarrollo

Dentro de cada minijuego se consideró un bloque de código que siempre estará pendiente de la puntuación de estudiante para que se pueda actualizar la dificultad de los mismos en tiempo real según el desempeño del jugador.

```

1 private void AjustarPuntos(bool acertado)
2 {
3     int puntosPorPalabra = 0;
4
5     if (PuntuacionJugador.instancia.PuntosJugador < 25)
6     {
7         puntosPorPalabra = acertado ? 3 : -5;
8     }
9     else if (PuntuacionJugador.instancia.PuntosJugador < 75)
10    {
11        puntosPorPalabra = acertado ? 5 : -5;
12    }
13    else
14    {
15        puntosPorPalabra = acertado ? 10 : -7;
16    }
17
18    PuntuacionJugador.instancia.PuntosJugador += puntosPorPalabra;
19 }
20

```

Este método mantiene al tanto al minijuego sobre la puntuación del jugador y varía los puntos que gana o pierde según su puntuación general, además actualiza los puntos generales del jugador y estos son tomados por el método de selección aleatoria de palabras para que se seleccionen palabras según la puntuación del estudiante y así modificar su dificultad.

```

1 public class IaTemario : MonoBehaviour
2 {
3     public string APIKey;
4     [TextArea(3, 20)]
5     public string prompt;
6
7     private readonly string chatGPTurlAPI = "https://api.openai.com/v1/chat/completions";
8     private readonly string Temario = "Assets/Scripts";
9     private readonly string directory = "Temarios";
10
11     private RequestBodyChatGPT requestBodyChatGPT;
12     private ResponseBodyChatGPT responseBodyChatGPT;
13
14     public void SendRequest()
15     {
16         requestBodyChatGPT = new RequestBodyChatGPT
17         {
18             model = "gpt-3.5-turbo",
19             messages = new List<Message>
20             {
21                 new Message { role = "system", content = "You are a helpful assistant." },
22                 new Message { role = "user", content = prompt }
23             },
24             max_tokens = 2048,
25             temperature = 0
26         };
27
28         StartCoroutine(SendRequestAPI());
29     }
30

```

1.3.12. Iteración 12: Objetivo específico 4 - Identificar la mejor técnica de IA que se adecue a las especificaciones del proyecto.

Fase 1: Descripción

Buscar la Mejor manera de aplicar una IA de forma sistemática y ordenada para cumplir con el objetivo central y principal del proyecto, que en este caso es la creación de temarios según el desenvolvimiento del jugador en un tema específico

Fase 2: Desarrollo

Sé considero que la mejor técnica que se podría usar para conseguir temarios completos y congruentes sería a través de una API de una IA ya entrenada y de amplio conocimiento para que así los temarios entregados fuesen más exactos, para el presente proyecto se consideró que la mejor opción sería la API de CHATGPT por su gran potencial, conocimiento y reconocimiento en el medio.

La forma de absorber y usar la API es a través de una petición con la APIKEY que OPEN IA ofrece. Se aplico de la siguiente manera:

A través de ese código se consume a la API de CHATGPT gracias a la APIKEY que se registra desde el inspector, luego se hace peticiones a través de la API según la información que se recolecta una vez terminadas las interacciones con mis minijuegos y se consigue un temario



```

1  public class Diosa : MonoBehaviour
2  {
3      void Start() { }
4
5      void Update()
6      {
7          if (PlayerEnRango && Input.GetKeyDown(KeyCode.R) && !InteraccionCompleta)
8          {
9              if (!DialogoComenzo)
10             {
11                 IniciarDialogo(); // Inicia el diálogo si aún no ha comenzado
12             }
13             else
14             {
15                 TerminarDialogo(); // Termina el diálogo si ya ha comenzado
16             }
17         }
18     }
19
20
21     private void OnTriggerExit2D(Collider2D collision)
22     {
23         if (collision.gameObject.CompareTag("Player"))
24         {
25             PlayerEnRango = false;
26             Aviso.SetActive(false); // Oculta el aviso
27         }
28     }

```

1.4. BETA

La fase BETA fue realizada con el apoyo de los usuarios finales que serán los estudiantes de la institución Serafín Filomeno y el docente encargado del curso, A estos se les hizo llegar una versión beta del videojuego y un documento para que así pudiesen registrar sus sensaciones con respecto al uso del videojuego, de esta fase se pudo recopilar mucha información que fue de suma utilidad para mejorar y corregir el videojuego en varios sentidos, los reportes hechos por alumnos y docente se puede encontrar en el ANEXO 1.

Información recogida de la BETA	Correcciones realizadas videojuego
Existían secciones específicas en algunos mapas donde el jugador podía caminar sobre secciones que deberían estar bloqueadas, ya sea árboles, casas o agua	Se recalibraron los Coliders2D para parchar el error y así solo permitir al jugador caminar sobre áreas específicas
El jugador pedía poder recorrer más rápido el mapa para así poder llegar más rápido a los minijuegos	Se implementó la función de aumentar la velocidad a la que el personaje se mueve cuando se presiona la tecla espacio y las teclas de movimiento
El jugador pedía una forma gráfica de ver su progreso en el mapa así como sus puntos	Se agregó una sección que lleva el conteo de las interacciones en el mapa y la posibilidad de ver el puntaje en el mapa específico
Mayor frecuencia de minijuegos	Se aumentó la frecuencia para sentir más dinamismo en los minijuegos
El movimiento usando las teclas A,W,S,D es confuso	Para los usuarios menos experimentados se agregó también la opción de poder mover al personaje con las flechas del teclado
Algunos minijuegos no funcionan como deberían	Se tomó en consideración cada bug reportado y se corrigió satisfactoriamente
Falta variedad de minijuegos	Se agregaron más minijuegos y variantes de algunos para no sentir monótona la jugabilidad
Que haya ambientación	Se agregó sonido ambiental personalizado a cada mapa para sentir una mejor acogida y menos monotonía mientras se desarrollan los minijuegos

1.5. CIERRE

Esta fase se llevó a cabo desplegando el juego de forma gratuita a todos los estudiantes de segundo grado de educación secundaria de la institución educativa Serafín Filomeno, estos probaron el juego y mostraron satisfacción y mejora en sus conocimientos de gramática, estos valores se pueden ver reflejados en una encuesta libre que se dio a los estudiantes para que contestaran y dieran experiencias, ver anexo 2.

Resultados y discusión

Esta fase se llevó a cabo desplegando el juego de forma gratuita a todos los estudiantes de segundo grado de educación secundaria de la institución educativa Serafín Filomeno. Un total de 20 estudiantes participaron en la fase de prueba y completaron la encuesta.

- 95% de los estudiantes (19 de 20) indicaron que les gustó cómo se ve el juego, lo que refleja una alta aceptación visual y estética del producto.

En cuanto al impacto en el aprendizaje:

- En el primer intento, 65% de los estudiantes (13 de 20) obtuvieron una puntuación de entre 0 y 25 puntos, y solo el 35% restante (7 de 20) logró entre 26 y 75 puntos.
- Para el tercer intento, el panorama cambió significativamente: 60% (12 de 20) lograron puntajes entre 76 y 100, mientras que el 40% restante (8 de 20) se mantuvo entre 26 y 75.

Esto representa una mejora clara en el rendimiento, con una progresión de los puntajes bajos a los altos entre el primer y tercer intento.

Estos datos evidencian que el videojuego no solo fue bien recibido, sino que además permitió a los estudiantes mejorar su desempeño gramatical en poco tiempo, confirmando así su efectividad como herramienta educativa accesible y de bajo costo, ver anexo 2.

Estos hallazgos coinciden con los estudios de Gee (2003) y Sykes y Reinhardt (2008), quienes sostienen que los videojuegos educativos incrementan el compromiso de los estudiantes al ofrecer un entorno lúdico y significativo para el aprendizaje. Avalo plenamente sus conclusiones, ya que he observado cómo el uso de elementos de gamificación y retroalimentación instantánea contribuyó al aprendizaje activo y al desarrollo de habilidades lingüísticas.

Los retos gramaticales adaptativos que implementé en el videojuego, basados en inteligencia artificial (IA), permitieron personalizar la experiencia de aprendizaje de cada

estudiante. Al igual que lo documentado por Ranalli (2008), quien destaca la eficacia de los videojuegos en la enseñanza de estructuras lingüísticas mediante tareas adaptativas, el sistema ajustó la dificultad de los retos en función del desempeño del estudiante, logrando mantener un equilibrio entre desafío y motivación. Respaldé estas afirmaciones con mis propios resultados, que mostraron una mejora gradual en el rendimiento de los estudiantes a medida que interactuaban con desafíos ajustados a su nivel de conocimiento.

Además, Grimshaw y Cardoso (2010) señalan que los videojuegos pueden proporcionar contextos de práctica auténticos y significativos para el aprendizaje del lenguaje. En mi caso, el uso de escenarios interactivos y desafíos gramaticales contextualizados brindó a los estudiantes una oportunidad de aplicar sus conocimientos de manera dinámica. La retroalimentación inmediata ofrecida en cada reto contribuyó a reforzar el aprendizaje y a corregir errores en tiempo real, un aspecto identificado como clave para el aprendizaje efectivo por Sykes y Reinhardt (2008). Apoyo firmemente esta perspectiva, ya que la evaluación que realicé mostró que los estudiantes adquirieron una mejor comprensión de las reglas gramaticales cuando recibieron correcciones y explicaciones al instante.

Los resultados de la fase beta del proyecto demostraron que la incorporación de un videojuego educativo basado en IA no solo incrementó la motivación de los estudiantes, sino que también generó mejoras sustanciales en su rendimiento académico en gramática. Estos hallazgos confirman que los videojuegos educativos, diseñados bajo un enfoque adaptativo y personalizado, son una herramienta eficaz para transformar el proceso de enseñanza- aprendizaje. A partir de mi experiencia, respaldo plenamente las investigaciones previas que destacan el valor de la gamificación y la tecnología en la educación.

En resumen, la falta de motivación en el aprendizaje de la gramática es un problema común que afecta tanto a nivel local como global, como lo evidencian las pruebas PISA y otros estudios de rendimiento educativo. Sin embargo, la solución implementada en este proyecto —el desarrollo de un videojuego educativo— demostró ser eficaz al mejorar tanto la motivación como el rendimiento de los estudiantes, tal como han señalado investigaciones anteriores en torno al uso de la gamificación en la educación. Los resultados obtenidos validan que el uso de videojuegos puede ser una estrategia efectiva para abordar los problemas de aprendizaje en áreas tradicionalmente difíciles de enseñar, como la gramática.

Conclusiones

Los listados de verificación aplicados durante la prueba piloto confirman que cada estudiante interactuó con un mínimo de tres minijuegos distintos en una sola sesión, lo que garantizó variedad y evitó la fatiga. La observación participante muestra que el 100 % de los alumnos exploró más de un mapa temático (tildación y ortografía), cumpliéndose la meta de ofrecer rutas de aprendizaje diferenciadas.

Durante el desarrollo y prueba del videojuego, se comprobó que los bloques de código diseñados lograron cumplir con su propósito pedagógico. Cada desafío gramatical fue programado para evaluar respuestas de forma automática y brindar **retroalimentación inmediata** basada en la causa del error cometido. La observación directa y los reportes de uso indicaron que todos los estudiantes recibieron una explicación contextualizada tras cada respuesta incorrecta, lo que reforzó el aprendizaje en tiempo real. Además, el 95 % de los estudiantes encuestados indicó que comprendió con facilidad los mensajes de retroalimentación. Esto evidencia que los bloques de evaluación no solo funcionaron técnicamente, sino que también contribuyeron significativamente al proceso de corrección y afianzamiento de conocimientos gramaticales.

Los registros de la base de datos y la encuesta evidencian un salto sustancial en el rendimiento:

Categoría de puntaje	Primer intento	Tercer intento
76 – 100 puntos	0 %	60 %
26 – 75 puntos	35 %	40 %
0 – 25 puntos	65 %	0 %

Esta transición muestra que **seis de cada diez estudiantes avanzaron al nivel más alto** tras solo tres rondas, lo que valida la eficacia de la progresión adaptativa.

Resultado:

El tamaño de la muestra necesario es de aproximadamente **108** individuos de una población de 150 para obtener resultados con un 95% de nivel de confianza y un margen de error del 5%.

El desarrollo del videojuego educativo personalizado logró cumplir con los objetivos propuestos, al mejorar significativamente el proceso de enseñanza-aprendizaje de gramática en estudiantes de segundo grado de secundaria. El videojuego, diseñado en un formato de rol (RPG) y con la incorporación de inteligencia artificial, permitió que los estudiantes recibieran desafíos adaptados a su nivel de conocimiento, lo cual resultó en una experiencia educativa más eficaz y atractiva. Esto se tradujo en una mejora considerable en el rendimiento de los estudiantes en temas de tildación y ortografía.

El videojuego educativo desarrollado cumplió el objetivo de crear interfaces intuitivas y fáciles de navegar, lo cual facilitó la interacción de los estudiantes con las diversas secciones del juego. Este diseño amigable permitió que los estudiantes se concentraran plenamente en los retos gramaticales, haciendo que la experiencia de aprendizaje fuera más accesible y efectiva. La navegación entre las pantallas del videojuego fue fluida, lo que contribuyó a una mayor satisfacción de los usuarios.

En cuanto a la dificultad progresiva de los minijuegos, se logró implementar un sistema que adaptaba los desafíos a las habilidades y conocimientos del estudiante. Esta adaptación permitió a los jugadores enfrentarse a retos cada vez más complejos, favoreciendo su aprendizaje de manera gradual y personalizada. La metodología aplicada garantizó que los estudiantes no se sintieran abrumados ni desmotivados, manteniendo su interés en el proceso educativo.

La retroalimentación instantánea proporcionada durante los minijuegos fue otro elemento clave para el éxito del videojuego. Al ofrecer correcciones inmediatas a los errores de los estudiantes, se mejoró significativamente su comprensión de las reglas gramaticales, en especial en áreas como la tildación y la ortografía. Este enfoque permitió a los estudiantes aprender de manera activa y corregir sus fallos en tiempo real, reforzando el aprendizaje.

Asimismo, el sistema de progresión gradual en la dificultad de los desafíos gramaticales permitió que el videojuego mantuviera a los estudiantes desafiados, sin llegar a frustrarlos, adaptando el contenido de acuerdo a su avance. Esta progresión inteligente fomentó una mejora continua en el rendimiento de los estudiantes, logrando que los contenidos fueran accesibles, pero siempre retadores.

Finalmente, la incorporación de inteligencia artificial fue fundamental para personalizar

la experiencia de aprendizaje. Al identificar la mejor técnica de IA, se logró adaptar eficazmente los desafíos gramaticales al nivel de cada estudiante, lo que permitió una enseñanza más precisa y adaptada a las necesidades individuales. Esta tecnología facilitó una experiencia educativa más dinámica y eficiente, contribuyendo a una mejora significativa en los resultados de los estudiantes.

Los resultados de la fase beta evidenciaron que el videojuego no solo incrementó el interés y la motivación de los estudiantes hacia el aprendizaje de la gramática, sino que también facilitó una comprensión más profunda de los contenidos. La capacidad del videojuego para proporcionar retroalimentación instantánea y personalizada permitió a los estudiantes corregir errores en tiempo real, lo que contribuyó a una mayor asimilación de los conocimientos. En conjunto, estos hallazgos confirman que la gamificación es una herramienta valiosa en el contexto educativo, especialmente para temas que tradicionalmente son percibidos como difíciles o aburridos.

Recomendaciones

Es recomendable expandir el videojuego para abarcar otros temas de gramática y áreas de aprendizaje, lo que permitiría aplicar esta herramienta en distintos niveles educativos y materias. Además, se sugiere realizar pruebas con una mayor cantidad de estudiantes de diversas instituciones para obtener datos más representativos y evaluar el impacto del videojuego en diferentes contextos educativos.

Asimismo, se recomienda incorporar tecnologías emergentes, como la realidad aumentada o virtual, para aumentar la inmersión y el nivel de interacción de los estudiantes, lo que podría generar un impacto aún mayor en su motivación y rendimiento. También sería beneficioso integrar un sistema de seguimiento más detallado que permita a los docentes monitorizar el progreso individual de los estudiantes con mayor precisión, identificando sus fortalezas y áreas de mejora.

Adicionalmente, se sugiere realizar estudios comparativos entre diferentes enfoques de gamificación y enseñanza tradicional para identificar qué elementos del videojuego contribuyen de manera más significativa al éxito del aprendizaje. Finalmente, sería útil desarrollar una versión del videojuego que permita su uso en dispositivos móviles y otras plataformas, con el fin de ampliar su accesibilidad y su impacto potencial en una mayor cantidad de estudiantes.

Otra recomendación es la inclusión de un sistema más detallado de seguimiento del progreso de los estudiantes, que permita a los docentes obtener información más precisa sobre las áreas en las que los estudiantes necesitan mayor refuerzo. Finalmente, se sugiere realizar estudios comparativos entre diferentes enfoques de gamificación para determinar cuáles son los elementos más efectivos en la mejora del aprendizaje y motivación de los estudiantes.

Referencias

- [1] A. Alberto, A. Alberto Arroyo, y M. Agustina Carranza, “Propuestas metodológicas para la enseñanza de la gramática: Un enfoque por tareas PROPUESTAS METODOLÓGICAS PARA LA ENSEÑANZA DE LA GRAMÁTICA: UN ENFOQUE POR TAREAS”.
- [2] UMC, “Resultados de Evaluaciones de logros de resultados 2019”. Consultado: el 29 de octubre de 2023. [En línea]. Disponible en: <https://www.calameo.com/read/006286625b1d7f0cd7597?view=slide&page=1>
- [3] S. Markus, “Results of PISA Assessment”, 2019.
- [4] PISA, “Publications - PISA”. Consultado: el 29 de octubre de 2023. [En línea]. Disponible en: <https://www.oecd.org/pisa/publications/pisa-2018-results.htm>
- [5] UNESCO, “LOS FUTUROS QUE CONSTRUIAMOS”, 2023. [En línea]. Disponible en: <https://www.unesco.org/open-access/terms-use-ccbysa-sp>
- [6] D. Aguilar, E. Celia, D. Contreras Julián, y R. Mabel, “FACULTAD DE EDUCACIÓN E IDIOMAS PROGRAMA DE COMPLEMENTACIÓN PEDAGÓGICA”.
- [7] A. Cueva Lobelle, L. Ochoa Sierra, A. Cueva Lobelle, y L. Ochoa Sierra, “Estado de la cuestión sobre la enseñanza de la gramática”, *Zona Próxima*, núm. 37, pp. 147–169, jul. 2022, doi: 10.14482/ZP.37.374.852.
- [8] M. B. Marcé, “Universidad Internacional de La Rioja Facultad de Educación Trabajo fin de máster”.
- [9] V. C., R. A., y M. R, “Estrategias Pedagógicas para la Motivación Lectora a Través de un Software de Lectura Interactiva”, *Revista Interamericana de Investigación, Educación y Pedagogía*, vol. 2, núm. 2, 2009.
- [10] J. LÓPEZ MaLINA, “Procedimientos y estrategias en la enseñanza-aprendizaje de la gramática”, 2000.
- [11] A. Juan, C. Chapoñan, S. Asesor, M. Juan, y A. T. Benavides, “INGENIERO DE SISTEMAS Y COMPUTACIÓN UNIVERSIDAD CATÓLICA SANTO TORIBIO DE MOGROVEJO FACULTAD DE INGENIERÍA ESCUELA DE INGENIERÍA DE SISTEMAS Y COMPUTACIÓN”, 2019.
- [12] C. Carrasco, A. Por, M. Ricardo David Imán Espinoza PRESIDENTE, y I. Hector Miguel Zelada Valdiviezo Mgtr Huilder Juanito Mera Montenegro, “INGENIERO DE SISTEMAS Y COMPUTACIÓN”.
- [13] T. Anastasiadis, G. Lampropoulos, K. Siakas, T. Staff, y M. Student, “International Journal of Advances in Scientific Research and Engineering (ijasre) Digital Game-based Learning and Serious Games in Education”, 2018, doi: 10.31695/IJASRE.2018.33016.
- [14] C. Girard, J. Ecalte, y A. Magnan, “Serious games as new educational tools: How effective

are they? A meta-analysis of recent studies”, *J Comput Assist Learn*, vol. 29, núm. 3, pp. 207–219, jun. 2013, doi: 10.1111/j.1365-2729.2012.00489.x.

- [15] C. C. Liu, Y. B. Cheng, y C. W. Huang, “The effect of simulation games on the learning of computational problem solving”, *Comput Educ*, vol. 57, núm. 3, pp. 1907–1918, nov. 2011, doi: 10.1016/J.COMPEDU.2011.04.002.
- [16] J. P. Gee, “Good Video Games and Good Learning”. [En línea]. Disponible en: www.matthewhenryhall.com
- [17] Taylor & Francis, “Learning English with The Sims: Exploiting Authentic Computer Simulation Games for L2 Learning”, doi: 10.1080/09588220802447859.
- [18] S. L. Thorne y E. Watters, “Language at Play: Digital Games in Second and Foreign Language Teaching and Learning. Language Learning & Technology”, *Language Learning & Technology*, vol. 17, núm. 3, pp. 47–51, 2013, Consultado: el 29 de octubre de 2023. [En línea]. Disponible en: <http://llt.msu.edu/issues/october2013/review2.pdf>

Anexos

ANEXO 1: Link a los documentos de texto de la fase de testeo del proyecto:

<https://drive.google.com/drive/folders/1XTtyTf-FwkJ5ODoppMS9SCtdzf4t9f8?usp=sharing>

ANEXO 2: Link de encuesta libre:

https://docs.google.com/spreadsheets/d/1gew_fQ3PrIrO1ehXlj7qRAJ4RXN398ru54UF4MclRB8/edit?usp=sharing

ANEXO 3: Link a documentos de retroalimentación que usa el estado dependiendo del nivel de los estudiantes:

https://drive.google.com/drive/folders/11RYRRUnb_EznR35hJLD_IDG-eWZ_x1XL?usp=sharing