



Universidad Internacional de La Rioja
Escuela Superior de Ingeniería y Tecnología

Grado en Ingeniería Informática

Desarrollo de un prototipo sobre una aplicación de sistemas adaptativos de aprendizaje.

Trabajo fin de estudio presentado por:	Luis Fernando Peón García
Línea de investigación:	Desarrollo de software
Directora:	Dra. Claudia Blanca González Calleros
Fecha:	19 de Julio de 2023
Repositorio del código fuente:	http://sourceforge.net/p/phrasemaster

Resumen

Este estudio ha desarrollado una estrategia para la traducción de frases del inglés al español dirigida a estudiantes de doce años en el ámbito educativo. El objetivo principal es crear un prototipo, diseñado con la herramienta “Visual Studio” de un sistema de aprendizaje adaptativo que mejore la habilidad de los estudiantes para traducir frases escritas en inglés. La estrategia se basa en proporcionar contenido con frases en inglés y sus traducciones al español para facilitar la comprensión y el aprendizaje del vocabulario. El proceso consiste en presentar la frase en inglés, permitir que el estudiante la traduzca al español y luego mostrar la traducción correcta. El estudiante puede seleccionar la dificultad de cada traducción, lo que afecta el intervalo de repetición de las frases siguiendo el principio de repetición espaciada.

El programa se basa en las respuestas del alumno para ajustar los intervalos de repetición, prolongándolos si la respuesta es correcta y acortándolos si es incorrecta. La aplicación utiliza un algoritmo de repetición espaciada para programar preguntas en momentos estratégicos y mejorar la retención del contenido.

Se ha utilizado el Lenguaje de Modelado Universal (UML) para el diseño y la documentación del sistema para elaborar diagramas que representen la estructura y las relaciones del sistema de manera clara y precisa.

Palabras clave:

Sistemas adaptativos de aprendizaje, repetición espaciada, sistemas de aprendizaje de idiomas, aprendizaje de traducción.

Abstract

This study has developed a strategy for translating sentences from English to Spanish targeted at twelve-year-old students in the educational field. The main objective is to create a prototype, designed using the "Visual Studio" tool, of an adaptive learning system that enhances students' ability to translate written English sentences. The strategy is based on providing content with English sentences and their Spanish translations to facilitate comprehension and vocabulary learning. The process involves presenting the English sentence, allowing the student to translate it into Spanish, and then showing the correct translation. The student can select the difficulty level for each translation, which affects the repetition interval of the sentences following the spaced repetition principle.

The program relies on the student's responses to adjust the repetition intervals, prolonging them if the response is correct and shortening them if it is incorrect. The application utilizes a spaced repetition algorithm to schedule questions at strategic moments and improve content retention.

The Universal Modeling Language (UML) has been used for system design and documentation to create diagrams that represent the structure and relationships of the system in a clear and precise manner.

Keywords:

Adaptive learning systems, spaced repetition, language learning systems, translation learning.

Índice de contenidos

Contenido

1.	introducción	8
1.1.	Motivación	8
1.2.	Planteamiento del trabajo	9
1.3.	Estructura del trabajo	10
2.	Contexto y Estado del Arte.....	12
2.1.	Contexto.....	12
2.2.	Estado del arte	16
2.2.1.	Definición y conceptualización	16
2.2.2.	Métodos de recopilación y análisis de datos	20
2.2.3.	Antecedentes y evolución	21
2.2.4.	Actualidad de los sistemas adaptativos de aprendizaje.....	23
2.2.5.	Los sistemas adaptativos en el aprendizaje de idiomas.....	24
3.	Objetivos y metodología de trabajo.....	25
3.1.	Objetivo general.....	25
3.2.	Objetivos específicos	25
3.3.	Metodología de trabajo	26
3.3.1.	Herramientas utilizadas.....	26
3.3.2.	Obtención de los datos de la base del conocimiento	28
3.3.3.	Procedimiento de evaluación de la aplicación.....	29
4.	Desarrollo	30
4.1.	Análisis	31

4.1.1.	Análisis del público objetivo	31
4.1.2.	Especificación de requisitos	32
4.1.3.	Descripción de las especificaciones desarrolladas en el prototipo.....	35
4.1.4.	Instrucciones de uso de la aplicación.....	36
4.2.	Diseño	40
4.2.1.	Descripción del sistema	40
4.2.2.	Diagrama de casos de uso	42
4.2.3.	Diagrama de actividades	44
4.3.	Desarrollo de la aplicación.....	45
4.3.1.	Diagrama de clases	46
4.3.2.	Diagrama de secuencias	50
4.4.	Implementación.....	53
4.4.1.	Descripción del fichero “Principal.cs”	53
4.4.2.	Descripción del fichero “Inicio.cs”	54
4.4.3.	Descripción del fichero “Resultado.cs”	54
4.5.	Evaluación	56
4.5.1.	Criterios de evaluación.....	56
4.5.2.	Herramientas para evaluación	56
5.	Conclusiones y trabajo futuro	59
5.1.	Conclusiones del trabajo.....	59
5.1.1.	Descripción de los logros en cada objetivo parcial	59
5.1.2.	Resultado de la evaluación de la herramienta.....	59
5.2.	Líneas de trabajo futuro	60
	Referencias bibliográficas.....	61
	Anexo A. Código fuente de “Principal.cs”	67

Anexo B. Código fuente de “Inicio.cs”	72
Anexo C. Código fuente de “Resultados.cs”	73
Anexo D. Ejemplo de cuestionario de opinión.	74

Índice de figuras

Figura 1. Puerta lógica AND con feedback	17
Figura 2. Máquina de Enseñanza.....	21
Figura 3. Violet UML Editor.....	27
Figura 4. Visual Studio	28
Figura 5. Adverso de una tarjeta Anki	34
Figura 6. Reverso de una tarjeta de Anki.....	35
Figura 7. Formulario de Inicio.....	37
Figura 8. Muestra frase para traducir.....	38
Figura 9. Frase traducida	38
Figura 10. Formulario de resultados.....	39
Figura 12. Descripción del modelo de los datos.....	40
Figura 13. El modelo instruccional	41
Figura 14. Esquema de la arquitectura.....	42
Figura 11. Diagrama de casos de uso	43
Figura 15. Diagrama de actividades.....	45
Figura 17. Diagrama de clases	46
Figura 16. Diagrama de secuencias	51
Figura 18. Muestra de resultados.....	57

1. introducción

En este trabajo se pretende mostrar las razones fundamentales que han motivado su elaboración, así como explicar su contenido y cómo está organizado. Se pretende brindar ayuda al lector para comprender mejor la temática y el alcance del trabajo, lo que le permitirá profundizar en los detalles.

1.1.Motivación

La aplicación aquí desarrollada está diseñada específicamente para alumnos de doce años, enfocada en el aprendizaje de idiomas. A continuación, se presentan las razones por las cuales es útil un sistema tutor de estas características para el tipo de usuario anteriormente mencionado:

El aprendizaje de cada niño es diferente; algunos pueden tener un dominio más rápido del idioma, mientras que otros pueden necesitar más tiempo y apoyo adicional; un sistema adaptativo puede ajustar el contenido y el nivel de dificultad para adaptarse a las necesidades individuales de cada estudiante, permitiendo un progreso óptimo sin dejar rezagados a los alumnos que necesita (Brusilovsky & Peylo, 2003).

Los sistemas de aprendizaje adaptativos pueden brindar una retroalimentación inmediata y específica después de cada actividad, lo que permite a los niños comprender sus fortalezas y áreas de mejora; esto les ayuda a corregir errores y aprender de ellos de manera más eficiente; también refuerza la confianza y la motivación de los estudiantes al proporcionarles información clara sobre su progreso(Brusilovsky & Peylo, 2003).

La aplicación que se ha llevado a cabo en el presente trabajo se fundamenta en la técnica de repetición espaciada, lo cual resulta beneficioso dado que mejora la retención a largo plazo al emplear intervalos óptimos de repaso. Este hecho evita la omisión de las palabras o conceptos aprendidos (Tabibian et al., 2019).

1.2.Planteamiento del trabajo

Se ha elaborado una aplicación didáctica especial diseñada para niños de doce años que ofrece un entorno de aprendizaje interactivo y entretenido, lo que fomenta el desarrollo de habilidades cognitivas.

Se aborda este problema debido a la importancia fundamental que tiene el vocabulario para dominar un idioma. La traducción de frases se presenta como una estrategia efectiva para aprender y practicar un vocabulario específico. Al mejorar la adquisición y retención del vocabulario, se busca mejorar la competencia lingüística de los estudiantes y facilitar su comunicación en el nuevo idioma; esto, a su vez, contribuirá a un aprendizaje más sólido y a una mayor confianza en el uso del idioma extranjero.

Más concretamente, con la propuesta de traducción de frases se desea potenciar las siguientes competencias lingüísticas (Reverte Oliver, 2023):

- ✓ Comprensión lectora. Al traducir frases, los estudiantes deben comprender el significado y la estructura de las oraciones en el idioma origen y encontrar la forma apropiada de expresarlas en el idioma destino. Esto aumenta su capacidad de comprensión lectora y su capacidad para interpretar textos en inglés.
- ✓ Vocabulario. La traducción de frases brinda a los estudiantes la oportunidad de ampliar su conocimiento y uso de vocabulario en ambos idiomas. A través de la traducción, los estudiantes pueden aprender nuevas palabras, expresiones idiomáticas y términos específicos.
- ✓ Gramática. Las traducciones de frases requieren ser cuidadosas con la estructura gramatical y las normas sintácticas de ambas lenguas: esto puede mejorar su comprensión y aplicación de la gramática, identificando y corrigiendo posibles errores gramaticales durante el proceso de traducción.

La adaptación de aprendizaje es una técnica que se ajusta tanto al nivel inicial como al progreso del estudiante, permitiéndole marcar su propio ritmo de aprendizaje. Esto ayuda a los profesores a ahorrar tiempo al crear materiales didácticos y no necesitan su presencia física durante el proceso de enseñanza. El sistema registra el progreso de los alumnos, lo que permite continuar desde cualquier punto y proporciona información útil en tiempo real para el docente (Pugliese, 2016).

Como los estudiantes están habituados a utilizar dispositivos móviles, estos sistemas se adaptan fácilmente a su forma de vida. La información que el estudiante recibe se puede actualizar en tiempo real, lo que garantiza la relevancia y actualización continua de los contenidos. No obstante, los sistemas en línea requieren una mayor atención por parte del alumno. Por otro lado, estos sistemas permiten llegar a estudiantes que están dispersos geográficamente y que pueden aprender de manera asíncrona, liberándolos de un horario rígido (Hong & Kinshuk, 2020).

Los sistemas de adaptación de aprendizaje son sumamente importantes para los estudiantes, ya que pueden personalizar el aprendizaje de forma individual, adaptándose a sus circunstancias, preferencias y necesidades. También son relevantes para los educadores, porque les proporciona información sobre el progreso de los estudiantes; con dicha información, pueden descubrir sus fortalezas y debilidades; con todo ello, pueden elaborar o adaptar los planes de estudios. Además, son útiles para el sistema educativo en general, ya que pueden ayudar a elaborar planes generales en cada nivel o ciclo de cada estudio en particular (Pugliese, 2016).

1.3. Estructura del trabajo

El presente documento se estructura en los siguientes capítulos:

1. Capítulo 1. Introducción

En este apartado se presenta una exposición de los motivos que han llevado a realizarlo, una descripción de su planteamiento y una explicación de su estructura.

2. Capítulo 2. Contexto y Estado del Arte

Este capítulo describe el entorno en el que se aplican los sistemas adaptativos y las herramientas de desarrollo.

3. Capítulo 3. Objetivos y metodología de trabajo

Se especifica la finalidad del trabajo desarrollado, así como el medio utilizado para conseguir dicho fin.

4. Capítulo 4. Desarrollo

Este capítulo es el más extenso de todos. En él se detallan los requisitos del sistema, tales como la gestión del perfil de usuario, el tratamiento de la información del conocimiento, la evaluación y el análisis de la progresión, así como la gestión de la retroalimentación de la información. Se han utilizado diversas herramientas para la representación de dichos requisitos, tales como: diagrama de casos de uso, de actividad, de secuencia, de clases y de estado. Para completar la descripción de los requisitos, en esta sección se ha presentado un prototipo. Este apartado se estructura siguiendo el estándar del modelo ADDIE (Análisis, diseño, desarrollo, implementación y evaluación).

5. Capítulo 5. Conclusiones y trabajo futuro

Finalmente, se detallan las conclusiones derivadas de la realización de todo el proceso, una síntesis de cómo ha sido todo el trabajo desarrollado, señalando los aspectos más importantes del mismo, haciendo propuestas para una posible ampliación y mejora.

2. Contexto y Estado del Arte

A continuación, se presentan los elementos que conforman el entorno de los sistemas de aprendizaje adaptativos y se define la situación actual de los mismos.

2.1.Contexto

En este contexto se ha elaborado una estrategia para la traducción de frases del inglés al español en el ámbito educativo de estudiantes de doce años. El enfoque se basa en proporcionar contenido que incluye frases en inglés junto con sus respectivas traducciones al español, lo cual facilita la comprensión de las palabras utilizadas en el texto. Este recurso resulta útil para que los alumnos aprendan y practiquen un vocabulario específico mientras trabajan en la traducción de frases.

El proceso de aprendizaje se realiza de la siguiente manera: se presenta una frase en inglés y se solicita al estudiante que la traduzca al español; después, se muestra la traducción correcta; al final, el estudiante selecciona la dificultad que considera haber tenido al realizar la traducción, utilizando su propio criterio. En función de la dificultad obtenida la frase se repetirá en el tiempo. También se proporcionan estadísticas sobre el rendimiento del alumno, como el porcentaje de respuestas correctas.

De esta manera se consiguen generar las contribuciones más significativas obtenidas en los sistemas adaptativos de aprendizaje (Chen et al., 2018):

- ✓ Personalización del aprendizaje. Los sistemas adaptativos posibilitan la adaptación del contenido, el ritmo y el enfoque de la enseñanza en función de las necesidades individuales de cada estudiante. Esto ayuda a que los alumnos aprendan su proceso de enseñanza-aprendizaje.
- ✓ Acceso a el aprendizaje flexible. Los sistemas de aprendizaje en línea brindan la posibilidad de acceder al contenido y llevar a cabo actividades en cualquier momento y desde cualquier lugar con acceso a internet. Esto ayuda a aprender más fácilmente y a aprender más, especialmente para aquellos que tienen problemas de tiempo o distancia.

Desarrollo de un prototipo sobre una aplicación de sistemas adaptativos de aprendizaje.

- ✓ Retroalimentación inmediata. Los sistemas adaptativos brindan información instantánea a los estudiantes acerca de su rendimiento y progreso. Esto les permite identificar de manera rápida sus fortalezas y debilidades, y enfocar sus esfuerzos en áreas específicas que requieren mejoras.
- ✓ Mayor compromiso y motivación: Las técnicas de adaptación pueden emplearse como la retroalimentación positiva para mantener el compromiso y la motivación de los estudiantes.

Es necesario considerar que la herramienta previamente concebida se enfoca en los estudiantes de doce años sin necesidades particulares. No obstante, en este contexto se han detectado algunas dificultades que suelen ser habituales en los sistemas de aprendizaje adaptativos (Chen et al., 2018).

- ✓ En el uso con estudiantes que tienen necesidades específicas, pueden surgir dificultades en la adaptación a ciertos estilos de aprendizaje.
- ✓ La aplicación y los sistemas de aprendizaje en general pueden dar retroalimentación individualizada, pero no se reciben directamente de la persona; los educadores pueden dar retroalimentación personalizada. Si no se interacciona directamente, los estudiantes pueden limitar su aprendizaje y limitar su aspecto humano y social.
- ✓ Si los estudiantes se acostumbran excesivamente a utilizar estas aplicaciones, pueden perder su habilidad para adquirir conocimientos a través de otras fuentes y métodos. Esto podría afectar su capacidad para enfrentar desafíos educativos sin la asistencia de la tecnología, lo que podría afectar su autonomía y habilidades para resolver problemas.

Para paliar las consecuencias negativas anteriormente mencionadas se pueden tomar las siguientes medidas (Chen et al., 2018):

- ✓ Adaptar la aplicación creando diferentes registros en las bases de datos para diferentes perfiles de alumnos.
- ✓ Enfatizar la importancia de la interacción humana. Los sistemas de adaptación de aprendizaje deben ser considerados como herramientas complementarias y no como reemplazos completos de la interacción humana.

- ✓ Promover el uso equilibrado de la tecnología. Los educadores deben fomentar la exploración de diversas fuentes de información, el intercambio de ideas en clase, la discusión en clase y la ejecución de actividades prácticas.

La plataforma educativa interactiva se ha concebido para los estudiantes de doce años con el propósito de que puedan adquirir conocimientos personalizados y eficaces.

Este sistema está diseñado teniendo en cuenta las necesidades y características específicas de dichos alumnos, quienes se encuentran en una etapa crucial del desarrollo cognitivo y emocional.

Los alumnos superan las limitaciones de la lógica que tienen que ver con la sensación de que tienen experiencias sensoriales directas. Experimentan el desarrollo de la inteligencia formal, propia de los adultos, que se caracteriza por su flexibilidad y capacidad de abstracción. Esto les ayuda a razonar con propuestas abstractas, analizar hipótesis de forma sistemática y resolver problemas complejos. Durante este período de transición de las "actividades prácticas" a las "actividades formales", los alumnos tienen la oportunidad de participar en el estudio sistemático de los fundamentos de las materias. No obstante, el contenido y la lógica de las asignaturas requiere un pensamiento independiente, habilidades para juzgar, comparar, extraer conclusiones y generalizar. Durante este proceso de aprendizaje, se adquiere la capacidad de pensar abstracto, la capacidad de análisis y síntesis. Los estudiantes también experimentan una tendencia más amplia en la búsqueda de causas subyacentes de los sucesos y eventos, lo que demuestra su importancia fundamental para el progreso del proceso educativo. Asimismo, poseen habilidades críticas al cuestionar ideas y afirmaciones, incluso las que provienen de figuras de autoridad como profesores, y se aplican esa misma capacidad crítica a su propio pensamiento, mostrando un interés en dominar la forma en que piensan (Serrano & Gascó, 2009)

La duración de las operaciones formales es diferente al período anterior, ya que el pensamiento puede trascender la realidad y se enfoca en el manejo de proposiciones verbales en lugar de la realidad tangible. La comprensión de la relación entre el lenguaje y el pensamiento es un proceso mental que tiene una interacción significativa con los procesos mentales existentes. El pensamiento formal se estructura mediante el lenguaje, lo cual posibilita el reemplazo de objetos por palabras y la comprensión de la realidad. El lenguaje

desempeña un papel esencial como medio y clave para el desarrollo de la mente. La ayuda a los preadolescentes a desarrollar un lenguaje preciso y rico no solo les permite expresar sus ideas de manera correcta, sino que también les contribuye a estructurar su pensamiento (Serrano & Gascó, 2009).

Este estudio pretende presentar una aplicación interactiva que ofrece un contenido personalizado y adaptado a cada estudiante, teniendo en cuenta la identificación de las características mencionadas. Se emplea un algoritmo inteligente para ajustar la dificultad de las lecciones y brindar feedback inmediato, lo cual posibilita una comprensión personal y personalizada, además, muestra la evolución de los alumnos.

2.2. Estado del arte

En este apartado se pretende presentar una visión integral de los progresos más recientes y relevantes.

2.2.1. Definición y conceptualización

En esta sección se presentan las circunstancias y factores que rodean los sistemas de aprendizaje adaptativos.

2.2.1.1. Concepto de sistema adaptativo

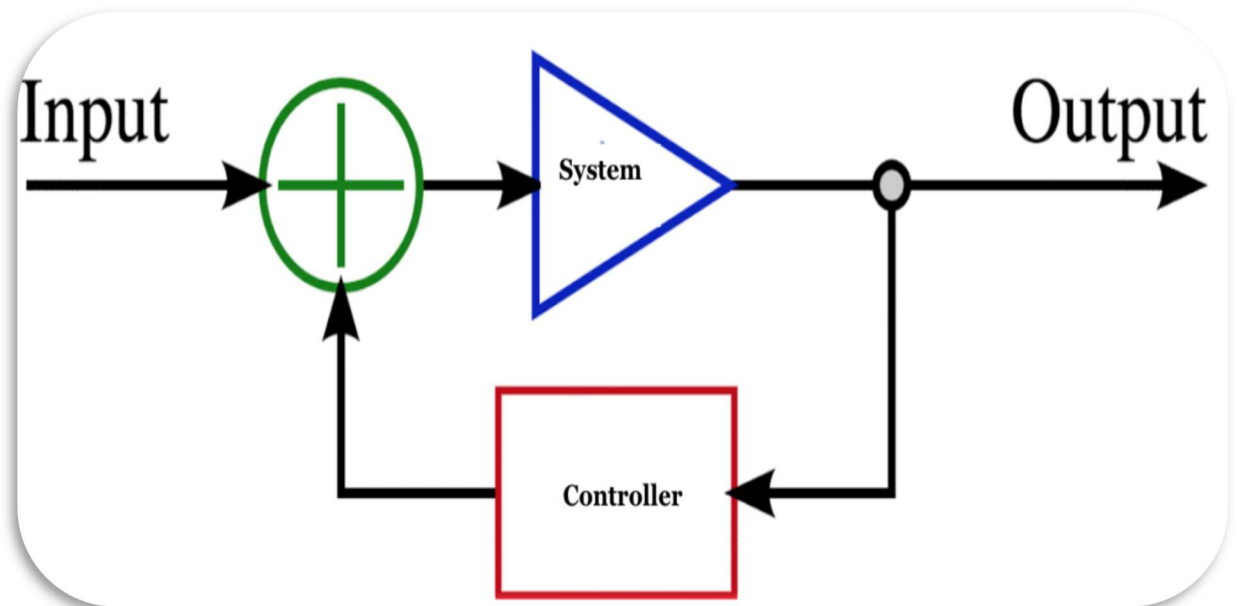
Se trata de una asociación de elementos organizados que poseen la capacidad de adaptarse a las circunstancias variables del entorno. El nuevo resultado tiene como objetivo optimizar la respuesta que el usuario recibe. Los datos de salida generados en un instante dado se utilizarán como datos de entrada en un momento posterior. Tienen aplicaciones en una gran variedad de áreas, como la industria, la medicina, la economía, etc. En contraste, un sistema lineal es aquel que no usa su salida como una nueva entrada de datos (Mareels & Polderman, 1996).

Las características de un sistema adaptativo son las siguientes (Nguyen et al., 2008):

1. Su arquitectura es más compleja que la de los sistemas lineales.
2. Se ajustan automáticamente a las circunstancias que se presentan.
3. A menudo requieren un entrenamiento, es decir, un período de adaptación y un número de nuevas entradas autogeneradas para optimizar su respuesta.

En la Figura 1 se presenta un ejemplo sencillo de sistema adaptativo, en el que se muestra una puerta lógica cuyo valor de entrada estará determinado por la salida del instante anterior.

Figura 1. Puerta lógica AND con feedback



Fuente: <https://www.gazeta.gt/la-cibernetica-y-el-hacer-retroalimentador/>, s. f.

2.2.1.2. Aprendizaje personalizado

Se trata de aquellos procedimientos que tienen como objetivo adaptar los contenidos en función de la evolución del aprendizaje o las características particulares de cada alumno. Debe tenerse en cuenta que cada estudiante parte de unos conocimientos previos diferentes y que cada uno tiene su propio ritmo de aprendizaje (Walkington & Bernacki, 2020).

Se diferencia del aprendizaje clásico en que este último no establece una distinción entre las necesidades, intereses y capacidades de los diferentes alumnos, por lo que los contenidos y las estrategias de enseñanzas serán los mismos para todos los alumnos. (Sadovaya et al., 2016).

Algunas características propias del aprendizaje personalizado son:

1. El aprendizaje debe comenzar con una evaluación inicial para conocer el nivel de conocimientos del alumno al principio de la enseñanza.
2. Deberá haber una actualización de los conocimientos que el estudiante va adquiriendo.

3. Se requiere una flexibilidad en la selección de contenidos y estrategias de enseñanzas para adaptarlas al alumno.

Es importante que el aprendizaje personalizado se complemente con estrategias de aprendizaje conductista y constructivista. El aprendizaje conductista es un aprendizaje que se produce mediante la asociación de ideas con estímulos; el aprendizaje se produce en respuesta a estímulos positivos. En el aprendizaje constructivista, la persona construye el aprendizaje a partir de un conocimiento anterior y asociándolo con una nueva experiencia (George E. Hein, 1991).

2.2.1.3. Sistemas de aprendizaje adaptativo

Se trata de sistemas que utilizan técnicas de aprendizaje automático para guiar al alumno en todo el proceso de aprendizaje, ajustando el material educativo al ritmo individual de cada alumno (Chen et al., 2018).

El objetivo es mejorar en velocidad, cantidad y calidad el proceso de aprendizaje.

Las características de todo sistema de aprendizaje adaptativos son (Terol, 2019):

1. Utilizan un sistema de retroalimentación, es decir, las respuestas dadas como salida en un momento servirán como entrada en un momento posterior.
2. El sistema debe almacenar los datos de respuesta del estudiante para que se produzca la retroalimentación.
3. Tanto el contenido como el entorno que se ofrece al estudiante deben estar personalizados.
4. El proceso de selección de contenidos y de tareas propuestas se realiza de forma automática, siguiendo unas premisas predefinidas.
5. Promueven el trabajo independiente y generan estímulos beneficiosos.
6. Proporcionan datos inmediatos acerca del nivel de acierto de una pregunta que el estudiante ha contestado.

2.2.1.4. Características de los sistemas adaptativos de aprendizaje

Se ha considerado que los sistemas adaptativos aportan las siguientes ventajas (Pugliese, 2016).

- ✓ Se ajusta tanto al nivel inicial del estudiante como a su progreso.
- ✓ Cada alumno marca su ritmo de aprendizaje, el sistema es una herramienta que le ayuda a guiarlo.
- ✓ Ahorra tiempo a los profesores en su labor de elaboración del material didáctico; además, no es necesaria la presencia física de los mismos para que se lleve a cabo el proceso de enseñanza.
- ✓ Los elementos de autocorrección automática mejoran la eficiencia del proceso. La respuesta inmediata del sistema, en lugar de una respuesta más tardía, hace que el alumno identifique su error de manera más certera.
- ✓ La evolución del proceso de aprendizaje queda registrada, lo que permite continuar el proceso en dicho punto. Además, la información almacenada puede ser usada por el docente como información útil; esta información puede ser obtenida en tiempo real.
- ✓ Se adaptan con facilidad a la nueva forma de vida de los alumnos. Estos suelen estar habituados a usar dispositivos móviles.
- ✓ La información que recibe el alumno se podrá actualizar en tiempo real.
- ✓ Los sistemas en línea exigen una mayor atención del alumno.
- ✓ Permiten llegar a un grupo de alumnado disperso geográficamente y de manera asíncrona, lo que hace que no estén sometidos a un horario estricto.

2.2.1.5. Arquitectura de un sistema adaptativo de aprendizaje

Según el autor que se consulte, hay una gran variedad de arquitecturas de sistemas adaptativos, sin embargo, existen algunos módulos básicos que forman el común denominador de todas ellas y que analizaremos a continuación.

El modelo de usuario o estudiante mantiene la información personal de cada estudiante, sus datos personales, conocimientos previos, resultados, itinerario de aprendizaje y objetivos (Fröschl et al., 2023).

La base de datos del conocimiento contiene los datos asociados con los contenidos o temas que el estudiante debe aprender; incluye todo tipo de contenido educativo: texto, imágenes, gráficos, animaciones, etc. (Mihalca et al., 2008).

El modelo instruccional establece la ruta que debe seguir el estudiante; selecciona los contenidos adecuados de la base de datos en función de los datos existentes en el modelo usuario, evalúa el nivel de los nuevos conocimientos adquiridos y, con esa información, se actualizan los datos del modelo usuario (Christoph, Fröschl et al., 2023).

El entorno de usuario es del medio de comunicación mediante el cual el estudiante interactúa con la aplicación. Es muy variable según el tipo de sistema de aprendizaje y los contenidos que se deseen enseñar. Es el responsable de presentar todo el material pedagógico, tales como texto y contenido multimedia; debe interactuar con el usuario, recibiendo las respuestas que este le ofrece.

2.2.2. Métodos de recopilación y análisis de datos

Se han desarrollado diversas técnicas y enfoques para obtener información en los sistemas adaptativos de aprendizaje, lo que permite la identificación y recopilación de datos en el sistema de investigación adaptativo. Algunas de las más habituales son la observación directa, que permite registrar el comportamiento del estudiante en tiempo real mientras interactúa con el sistema; los cuestionarios y encuestas, que permiten recopilar información sobre las percepciones y actitudes del estudiante hacia el sistema; los registros de actividades, que capturan el progreso y desempeño del estudiante a lo largo del tiempo; y el análisis de registros. Estas técnicas ayudan a entender mejor el proceso de aprendizaje y a hacer ajustes personalizados en la entrega de contenido y actividades (Weibelzahl, 2001).

La recopilación de interacciones en los sistemas adaptativos de aprendizaje permite obtener información sobre las respuestas del estudiante a las preguntas, el tiempo que dedica a diferentes actividades, las selecciones que realiza al navegar por el sistema y la retroalimentación proporcionada. Este material presenta una base significativa en la identificación de áreas de mejora y evaluación (Bull & Kay, 2016).

El registro del tiempo dedicado a actividades en los sistemas adaptativos de aprendizaje es fundamental para evaluar el nivel de compromiso y distribución del tiempo de los estudiantes en diferentes áreas de estudio. Proporciona una visión detallada de cómo los estudiantes utilizan su tiempo, lo cual es útil para evaluar su motivación; además, permite identificar intereses y dificultades específicas mediante el análisis del tiempo invertido en diferentes actividades; también proporciona información sobre la distribución del tiempo de estudio, lo cual es relevante en entornos en línea. Debido a los datos proporcionados, los instructores y diseñadores pueden ajustar la entrega de contenido para mejorar el rendimiento del estudiante (Bull & Kay, 2016).

El seguimiento del comportamiento de los estudiantes en los sistemas adaptativos de aprendizaje es necesario para recopilar datos sobre su interacción y su compromiso. Se registran aspectos como la navegación, el tiempo de acceso, los patrones de estudio y otras acciones llevadas a cabo en la plataforma; estos datos proporcionan una visión detallada del nivel de compromiso y motivación de los estudiantes, lo que permite evaluar su participación activa. Asimismo, se pueden identificar patrones y tendencias (Bull & Kay, 2016).

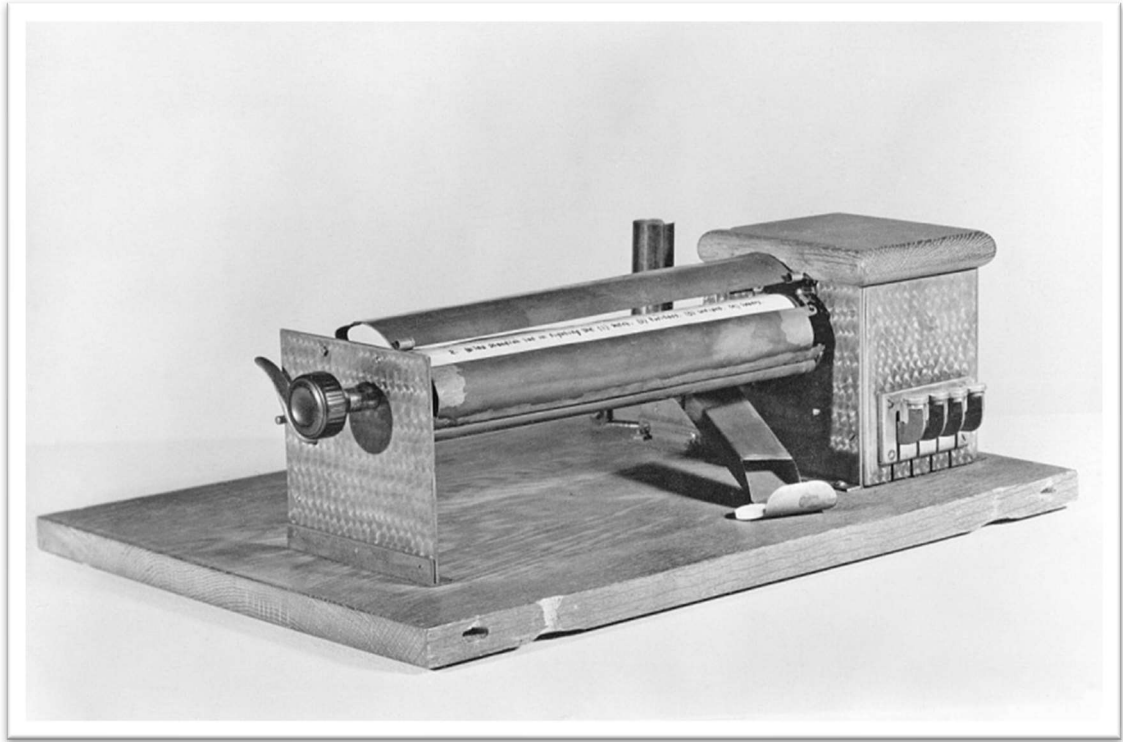
2.2.3. Antecedentes y evolución

A continuación, se presenta el desarrollo de los sistemas adaptativos de aprendizaje en orden cronológico.

- ✓ Durante las décadas de 1920 y 1930. Los primeros sistemas se elaboran en función del principio de estímulo y respuesta. Se fundamentan en algoritmos sencillos que refuerzan la respuesta correcta. Uno de los sistemas más famosos en la época fue la conocida “Máquina de Enseñanza” desarrollada por Sidney Pressey. Consistía en una máquina que leía las preguntas contenidas en tarjetas y el alumno debía responder pulsando la tecla que creía estar asociada con la respuesta correcta.

Ver la figura 2.

Figura 2. Máquina de Enseñanza



Fuente: (National Museum of American History, s. f.)

- ✓ Años cincuenta. El término de instrucción programada fue introducido en 1950 por B. F. Skinner. Se trata de dividir los contenidos en partes más pequeñas de forma que se pueda presentar al estudiante de manera secuencial y más sencilla. El concepto de refuerzo positivo se concreta en la "Teaching Machine", en este caso, además de preguntas en formato de texto, también se utiliza el audio. Otra innovación que se introdujo fue el denominado "dominio de aprendizaje", en el que un estudiante debía haber demostrado un conocimiento positivo de un tema determinado para pasar al siguiente (Casas, 2002).
- ✓ Década de 1970. Se introduce el concepto de retroalimentación, las máquinas ya almacenaban los resultados obtenidos anteriormente. Se trata de los primeros sistemas que presentan la arquitectura de los actuales, y están compuestos por los cuatro módulos siguientes: una base de conocimientos, un modelo de usuario, un módulo pedagógico o instruccional y una interfaz. El proyecto más relevante se originó a finales de la década y fue denominado ICAI o Intelligent Computer Assisted Instruction. (Dede, 1986).

- ✓ A partir de los años 80. Los sistemas expertos se fueron perfeccionando para brindar apoyo a la enseñanza. Con la llegada de internet, muchos sistemas permitían al estudiante realizar sus clases en línea, de manera que no se requería la presencia física en un aula. En los últimos años, los sistemas han utilizado el término Big Data para analizar los contenidos actualizados y obtener un mejor perfil del estudiante (Williamson, 2017).

2.2.4. Actualidad de los sistemas adaptativos de aprendizaje. Desafíos y oportunidades

Dado que Internet ofrece la posibilidad de estudiar a distancia, los sistemas no solo permiten la personalización de los contenidos, sino que también el hecho de que el alumno pueda seleccionar el momento y el lugar en los que desee hacer uso del sistema. Asimismo, posibilita el desarrollo de enseñanzas basadas en el aprendizaje colaborativo, lo que a su vez mejora la comunicación del profesor con los estudiantes y la de estos entre sí. Por otro lado, el uso de técnicas de análisis de datos identifica patrones de aprendizaje del alumno, lo que permite optimizar el modelo educativo del estudiante. (Sarıyalçınkaya et al., 2021)

Algunos de los sistemas más usados en la actualidad son los siguientes:

- ✓ ALEKS: Es un programa en línea de aprendizaje de varios niveles de matemáticas que guía al alumno a moviéndose a través de una ruta de aprendizaje personalizada (<https://www.aleks.com/>, s. f.).
- ✓ Duolingo: Es una aplicación en línea que permite el aprendizaje de idiomas a través de diversas plataformas (<https://es.duolingo.com/>, s. f.).
- ✓ Smart Sparrow: Diseñado para crear cursos personalizados e iterativos (<https://www.smartsparrow.com/>, s. f.).
- ✓ Khan Academy: Plataforma creada por una organización sin ánimo de lucro, para llevar la enseñanza a cualquier lugar (<https://es.khanacademy.org/>, s. f.).
- ✓ Anki: La base del conocimiento se estructura en tarjetas con preguntas y respuestas, que se seleccionan automáticamente siguiendo la lógica de la teoría de repetición espaciada (<https://apps.ankiweb.net/>, s. f.).

Las aplicaciones anteriores han realizado diversas aportaciones en el ámbito educativo. Estas características incluyen la personalización del aprendizaje mediante el uso de algoritmos adaptativos, la flexibilidad y accesibilidad a través de plataformas en línea, la retroalimentación inmediata para corregir errores y comprender conceptos, el aprendizaje interactivo mediante técnicas de gamificación, y la disponibilidad de recursos educativos gratuitos que permiten un acceso amplio y equitativo al conocimiento. Las contribuciones obtenidas han permitido a optimizar la experiencia de aprendizaje de los estudiantes, así como a fomentar su autonomía y motivación.

2.2.5. Los sistemas adaptativos en el aprendizaje de idiomas

Los sistemas de aprendizaje de idiomas abarcan desde herramientas de traducción asistida por ordenador y sistemas de traducción automática, hasta plataformas de aprendizaje de idiomas en línea y software especializado de ayuda a la traducción. Estos sistemas ofrecen una amplia gama de recursos interactivos, tales como ejercicios de gramática, juegos, vídeos y audios, con el fin de adquirir habilidades lingüísticas como la comprensión auditiva, la expresión oral, la lectura y escritura. Además, proporcionan retroalimentación inmediata para corregir errores y personalizar el contenido en función de las necesidades de cada estudiante (Chiang et al., 2012).

Existen diversas plataformas, tales como “Duolingo, Rosetta Stone, Babbel, Memrise y Lingvist”, las cuales ofrecen a los usuarios la oportunidad de realizar ejercicios interactivos de traducción de frases y oraciones, así como otros tipos de ejercicios para mejorar sus habilidades lingüísticas. Además, se adaptan al nivel y las necesidades individuales de cada estudiante, lo que les permite mejorar su competencia en inglés y expandir su vocabulario (Chiang et al., 2012).

3. Objetivos y metodología de trabajo

En esta sección se especifican las metas o resultados que se esperan alcanzar, el enfoque, la estrategia y procedimientos que se han utilizado.

3.1. Objetivo general

Elaborar un prototipo de un sistema de aprendizaje adaptativo, enfocado en estudiantes de doce años que cursan el idioma inglés, con el propósito de mejorar su habilidad de traducir frases escritas en inglés.

3.2. Objetivos específicos

El objetivo mencionado anteriormente puede dividirse en los siguientes objetivos parciales:

- ✓ Analizar las necesidades de aprendizaje, identificando los objetivos del curso seleccionado, recopilando información sobre el perfil de alumnado y su contexto.
- ✓ Elaborar la arquitectura e interfaz de usuario de la aplicación; debe incluir una serie de conceptos conceptuales para establecer una estructura, componentes y flujo de interacción.
- ✓ Planificar la estructura y los contenidos, definiendo los métodos de aprendizaje, los materiales instruccionales y las estrategias de evaluación.
- ✓ Evaluar la eficacia de la aplicación concebida; para ello se evalúa la utilidad de los resultados, se recopilan datos sobre el rendimiento y la calidad de la instrucción.

3.3. Metodología de trabajo

En este apartado se detallan los procedimientos, técnicas y herramientas que se han usado para el desarrollo del trabajo.

Identificación y determinación del problema que es objeto de estudio. Consiste en una investigación preliminar, consultado documentación existente tal como: artículos de investigación, libros en formato electrónicos, páginas webs, etc.

Definición del enfoque del trabajo, se identifican los aspectos específicos que van a desarrollar, descritos en la introducción.

Consulta en las fuentes académicas. Con el objetivo de contextualizar, conocer el estado actual del tema y la posibilidad de generar nuevas aportaciones, se ha realizado una revisión bibliográfica; esto requiere la selección de información considerada como útil para su posterior análisis, evaluando su calidad y su adecuación al asunto tratado.

Verificación y evaluación de las fuentes académicas, se ha considerado su adecuación para el presente trabajo y se ha verificado su autenticidad y considerado la adecuación. Para confirmar que las fuentes encontradas son válidas, es esencial evaluar, relevancia y confiabilidad de estas; para dicho propósito se han utilizado consultas en línea de bibliotecas, base de datos y artículos publicados en sitios de prestigio como "Scopus" o "Web of Science. Estas fuentes han sido citadas conniventemente siguiendo las normativas fijadas en el estándar APA 7.

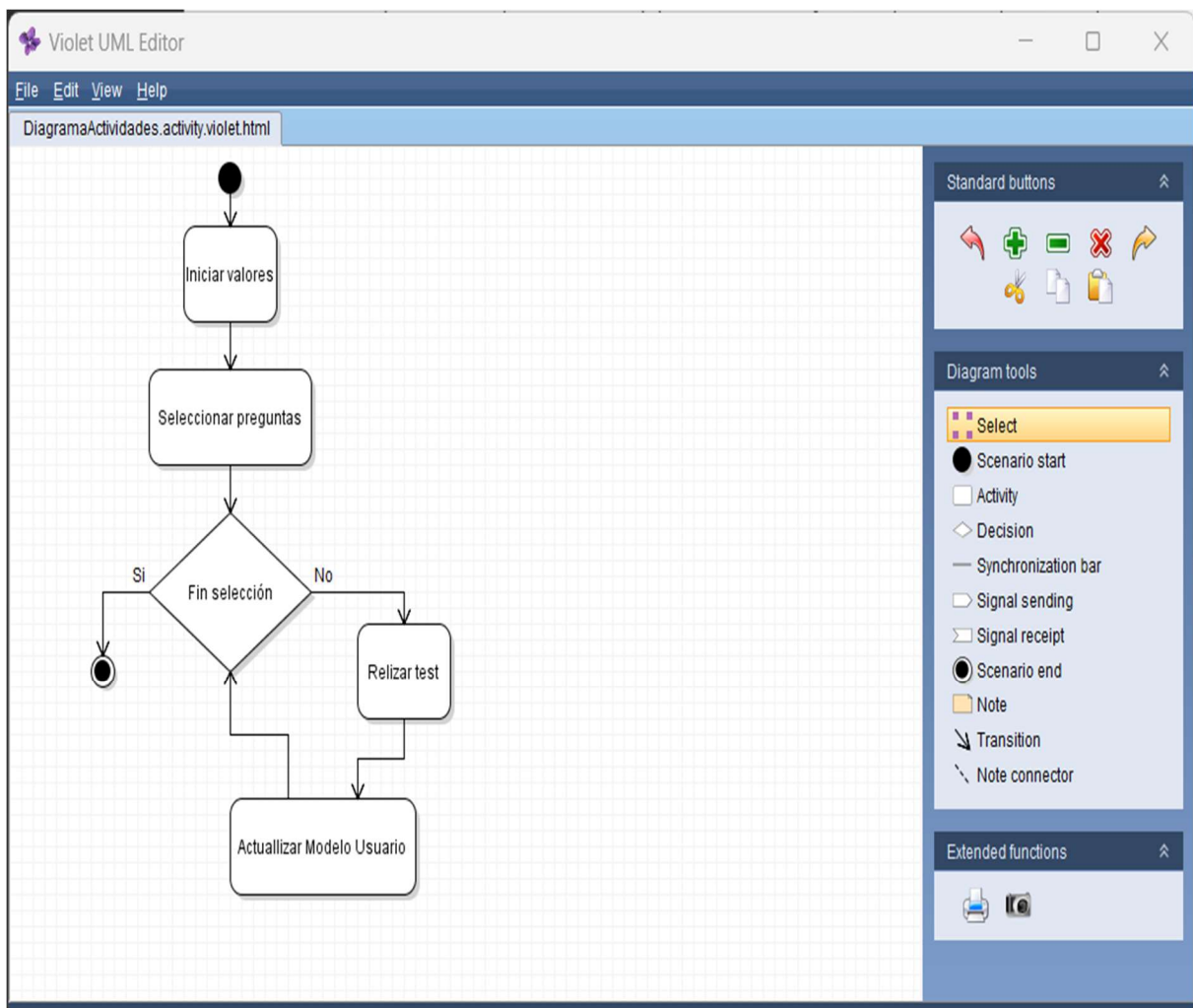
3.3.1. Herramientas utilizadas

Se ha utilizado el Lenguaje de Modelado Universal (UML) como una valiosa herramienta en el proceso de representación y documentación de los diversos elementos del sistema. El UML proporciona un conjunto de notaciones gráficas estandarizadas que permiten visualizar y comunicar de manera clara y precisa la estructura, comportamiento y relaciones de los componentes del sistema (Rodríguez & Perez, 2023).

Con el propósito de facilitar y respaldar esta tarea, se ha empleado la herramienta "Violet UML." Esta herramienta ha demostrado una gran eficacia en la creación de diagramas UML, ofreciendo una interfaz intuitiva y funcionalidades prácticas que facilitan el proceso de

modelado. Violet UML posibilita la creación de diagramas de casos de uso, diagramas de clases, diagramas de secuencia y otros tipos de diagramas UML de manera eficiente (*Violet UML Editor : easy to use, completely free, s. f.*). Ver figura 3.

Figura 3. Violet UML Editor



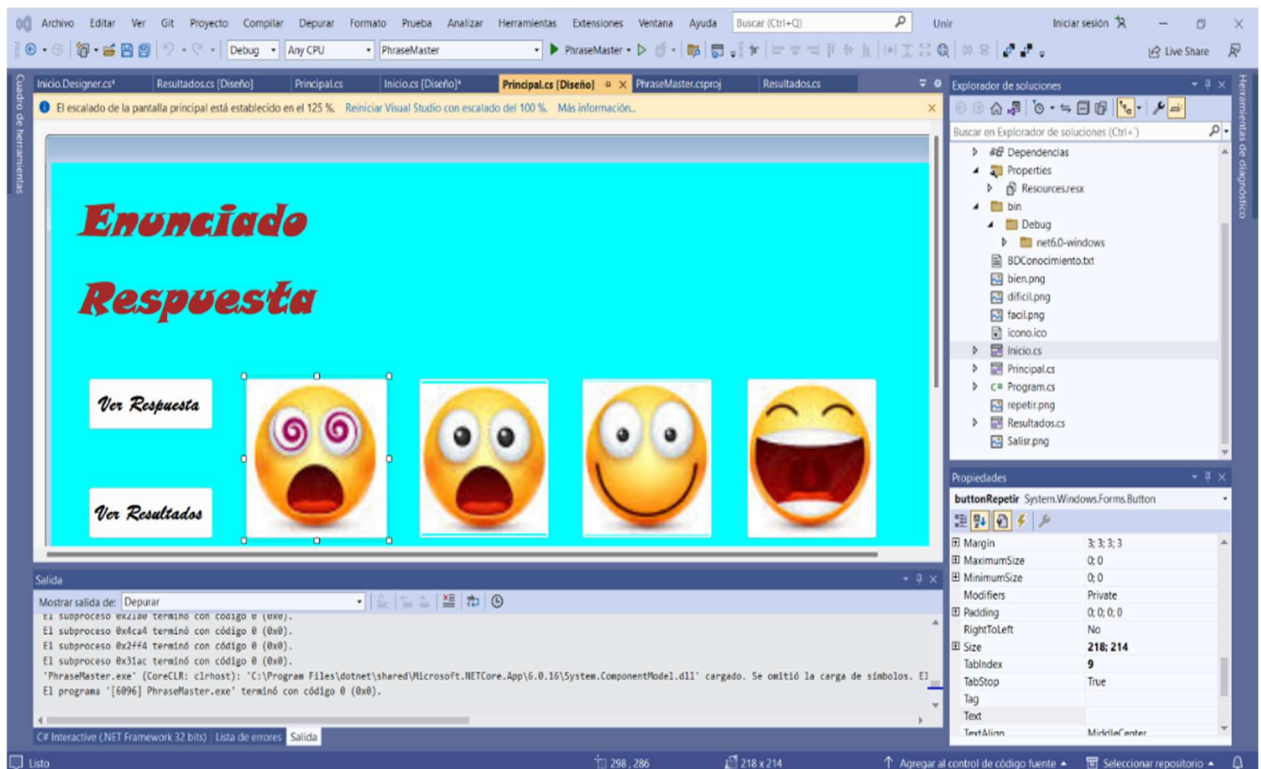
Fuente: Elaboración propia

Con el propósito de proporcionar una visión exhaustiva del sistema y posibilitar su evaluación, se ha elaborado un prototipo utilizando la herramienta de programación y diseño gráfico Visual Studio. El prototipo elaborado ofrece una valiosa oportunidad para llevar a cabo pruebas y recibir comentarios tempranos de los usuarios y las partes interesadas. Al

Desarrollo de un prototipo sobre una aplicación de sistemas adaptativos de aprendizaje.

proporcionar una perspectiva aproximada del producto final, se brinda a los usuarios una comprensión más precisa de cómo se verá y funcionará el sistema en su totalidad. Asimismo, ofrece la oportunidad de detectar posibles mejoras y ajustes antes de progresar en el desarrollo completo. Ver figura 4.

Figura 4. Visual Studio



Fuente: Elaboración propia.

Visual Studio, tiene muchas herramientas y capacidades de desarrollo, ha sido utilizado para diseñar y construir el prototipo de manera eficaz. La combinación de su entorno de desarrollo integrado (IDE) y su habilidad para programar diferentes lenguajes ha permitido crear una interfaz de usuario funcional y atractiva, así como implementar las funcionalidades importantes del sistema (*Visual Studio*, s. f.).

3.3.2. Obtención de los datos de la base del conocimiento

Dentro del contexto de la traducción de las frases que los estudiantes deben abordar, se han utilizado las siguientes fuentes con el fin de recopilar los datos requeridos:

Desarrollo de un prototipo sobre una aplicación de sistemas adaptativos de aprendizaje.

- ✓ "Education.com" es un recurso educativo en línea que ofrece una amplia gama de materiales para los estudiantes; en su sección de traducción, existen ejercicios adaptados para practicar la traducción del inglés al español con estudiantes de doce años (*Educational Games, Worksheets, and More for Kids*, s. f.).
- ✓ "ESL Tower" es un sitio web enfocado en la educación del inglés como segundo idioma; brinda una amplia variedad de recursos y actividades, tales como ejercicios de traducción para estudiantes de inglés en diversos niveles (*English Grammar, Vocabulary, Pronunciation Exercises for ESL Teachers and Students*, s. f.).

3.3.3. Procedimiento de evaluación de la aplicación

Para examinar el funcionamiento de la aplicación, se han considerado los siguientes aspectos:

- ✓ Se recopilan comentarios y opiniones de los estudiantes sobre la aplicación, se pregunta si encuentran útil la metodología de traducción de frases, si se sienten motivados y comprometidos durante el proceso de aprendizaje, y si consideran que la aplicación se ajusta a sus necesidades individuales.
- ✓ Se ha estudiado cómo los estudiantes han mejorado su habilidad para recordar las traducciones y si han mejorado su vocabulario y habilidad en inglés.
- ✓ A través del uso de estadísticas proporcionadas por la aplicación para evaluar el rendimiento de los estudiantes.
- ✓ Realización de un seguimiento personalizado de los estudiantes mientras utilizan la aplicación. Observación de su nivel de participación, interacción y compromiso durante las actividades de traducción.

Los resultados de dicha evaluación se detallan el apartado 4.5.

4. Desarrollo

Para el desarrollo de la aplicación se ha usado el modelo de diseño instruccional ampliamente utilizado en la educación, denominado ADDIE.

El modelo ADDIE consta de cinco etapas fundamentales: Análisis, Diseño, Desarrollo, Implementación y Evaluación. Se recopila información relacionada con las necesidades y recursos disponibles durante la etapa de análisis. En la etapa de diseño, se establecen los objetivos y se planifica el material del curso o programa. En el transcurso del proceso de desarrollo, se elabora el material de aprendizaje en concordancia con los objetivos establecidos. La implementación se refiere a la entrega eficaz del curso o programa a los estudiantes. Para finalizar, la evaluación se enfoca en examinar y examinar los resultados obtenidos con el propósito de identificar áreas de mejora y efectuar ajustes necesarios para optimizar el proceso de aprendizaje (Allen, 2006).

Este apartado se estructura siguiendo las fases anteriormente mencionadas.

4.1. Análisis

Implica la recopilación y estudio de información relevante para comprender las necesidades de aprendizaje y establecer las bases para el diseño del programa de instrucción.

La información recopilada en esta etapa sirve como base para el diseño del programa, definiendo los objetivos de aprendizaje y estableciendo las estrategias y enfoques adecuados para lograr esos objetivos (Robert, 2009).

4.1.1. Análisis del público objetivo

La aplicación de traducción de frases está dirigida a estudiantes de doce años que están estudiando inglés como segundo idioma. El público objetivo es estudiante de esta edad que desean mejorar su capacidad de traducción y comprensión del idioma inglés.

La estrategia de aplicación se enfoca en brindar información relevante para su nivel de conocimiento y enfoque en el vocabulario específico: el uso de frases en inglés y sus correspondientes traducciones al español posibilita a los estudiantes comprender mejor las palabras utilizadas y mejorar su comprensión del texto.

La aplicación se ajusta a las necesidades individuales de los estudiantes al permitirles seleccionar la dificultad de las traducciones según su propio criterio. Les brinda la oportunidad de trabajar en el vocabulario y las frases que consideran más desafiantes, lo que fomenta un aprendizaje personalizado y autodirigido.

Durante el proceso de elaboración del ejercicio, se ha establecido la determinación del nivel de conocimiento previo de los estudiantes, teniendo como referencia el currículum oficial de la materia. Esto ha requerido la realización de una evaluación minuciosa de los contenidos y habilidades establecidos en el currículum para el nivel adecuado. Este análisis permite comprender el nivel de dominio que los estudiantes tienen en los diferentes aspectos del tema o área de estudio específico (Bautista, 2010).

Se revisan libros de texto, materiales didácticos o recursos en línea diseñados para alumnos de doce años. Los libros de texto son una fuente confiable y estructurada de contenido educativo, ya que suelen seguir los principios curriculares establecidos. Estos libros pueden proporcionar ejemplos de frases y ejercicios de traducción que se ajusten a los temas y vocabulario relevantes para los alumnos (Bautista, 2010).

4.1.2. Especificación de requisitos

Se enumeran las características que debe cumplir el software. Es necesario comprender los requerimientos del cliente y mejorar su comunicación con él; así como facilitar la definición de los objetivos y el aseguramiento de que los desarrolladores cumplan con dichos requisitos (Salamea Bravo et al., 2020).

La norma general es brindar una oportunidad de aprendizaje efectiva para el estudiante. Se deberán presentar las respuestas correctas al estudiante y un resumen de los resultados que va obteniendo. El análisis eficaz de los resultados permite al estudiante ser consciente de los conceptos que necesita mejorar, además facilita el aprendizaje autónomo y una motivación en el aprendizaje. Un resumen de los resultados del progreso del estudiante es útil para evaluar el progreso y ser conscientes de aquellas áreas que no se han superado o se han llevado a cabo con dificultad.

Se enumeran las funciones específicas que cualquier sistema debe proporcionar para satisfacer los requisitos del usuario. Se requiere que el sistema sea capaz de:

- ✓ Proporcionar el contenido pedagógico adecuado teniendo en cuenta la progresión del estudiante (Chen et al., 2018).
- ✓ Proporcionar un mecanismo de feedback para lograr la tarea mencionada anteriormente. Será la herramienta que permita ajustar el progreso. (Chen et al., 2018).
- ✓ Informar del resumen de los resultados. Se proporcionan resultados del proceso de aprendizaje; se muestra información sobre la eficacia del desarrollo del proceso, también se determinará en qué medida se han alcanzado los objetivos (Chen et al., 2018).

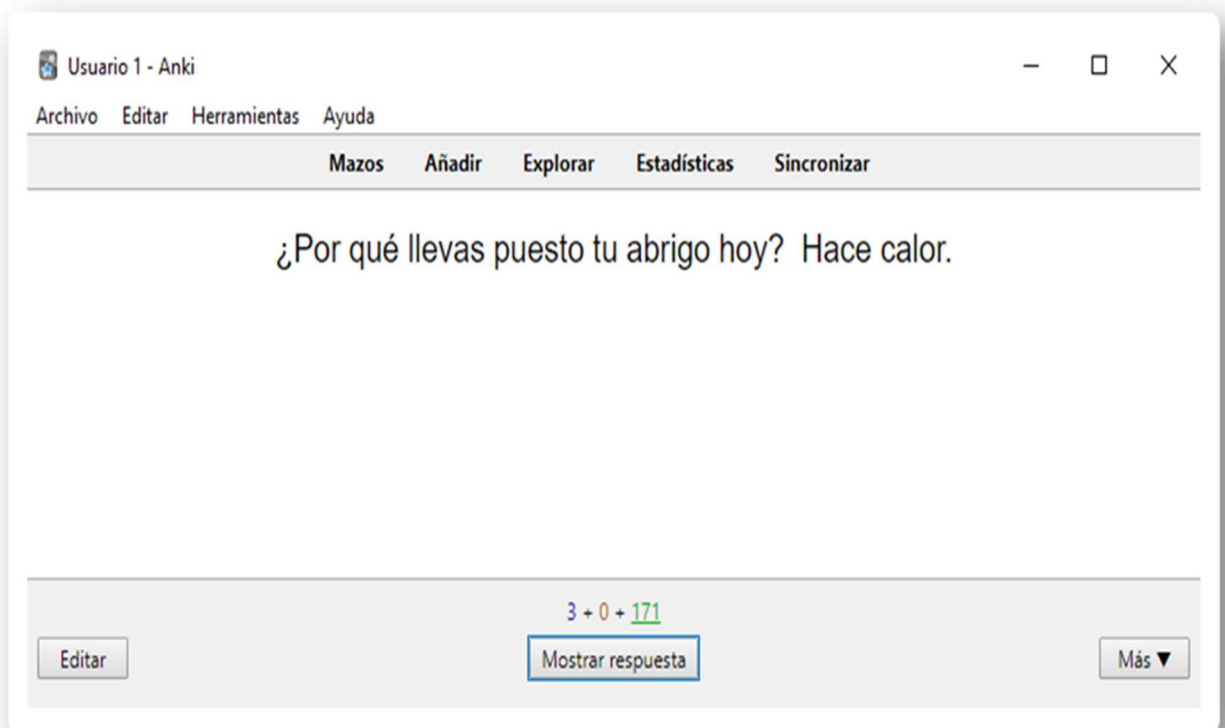
La aplicación realizada en presente trabajo se basa en la técnica de repetición espaciada (Spaced Repetition System, SRS); es una técnica de aprendizaje que utiliza intervalos de tiempo óptimos para repasar los contenidos con el objetivo de mejorar el recuerdo de los mismos en el tiempo. Se recuerda mejor la información cuando la revisamos en momentos específicos, distribuidos en el tiempo (Reddy et al., 2016).

La técnica de repetición espaciada se utiliza para mejorar el proceso de revisión, lo que tiene un impacto en los conceptos con mayor dificultad y priorizando aquellos que requieren una mayor consolidación en la memoria a largo plazo. Al evaluar los contenidos en determinados momentos, se mejora la retención, se optimiza el tiempo y el esfuerzo necesarios para el aprendizaje (Hernández, 2013).

Aplicaciones de aprendizaje de idiomas diversas, como Duolingo, Anki y Memrise, utilizan algoritmos de repetición espaciada para estudiar contenidos relacionados con los idiomas, como las reglas gramaticales, el vocabulario, la pronunciación, la comprensión auditiva, etcétera.

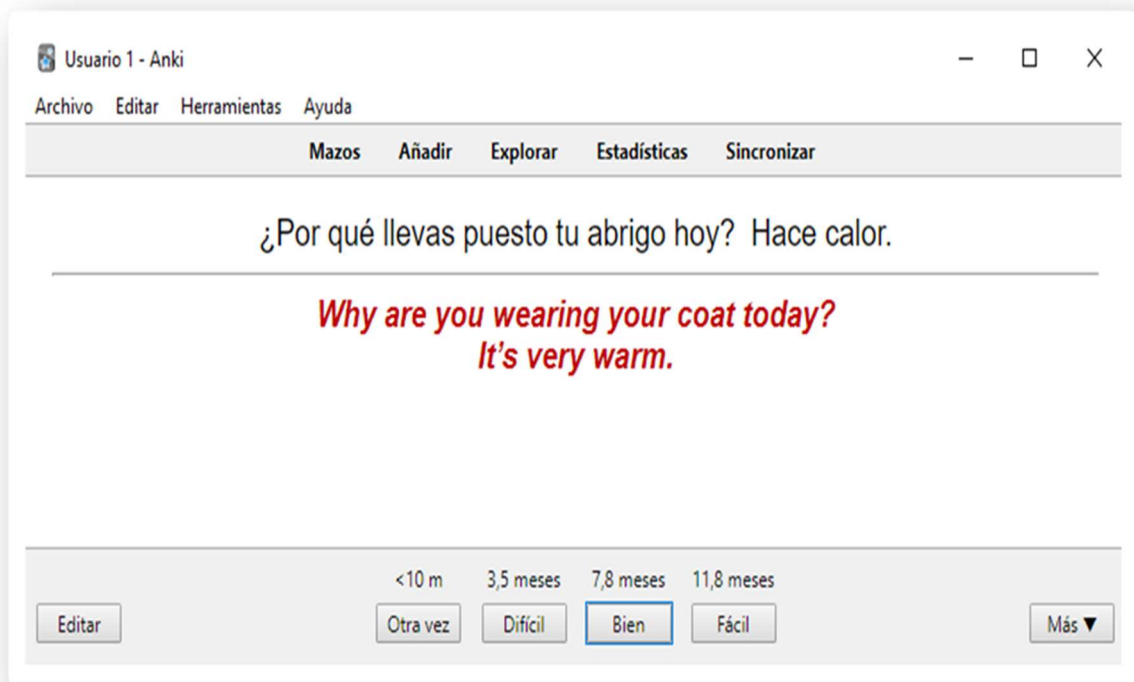
Los contenidos aparecen en la pantalla del usuario en forma de pregunta, ocultando la respuesta correcta. El usuario debe adivinarla. Véase la Figura 5. Posteriormente, se muestra la respuesta y una serie de opciones que evalúan el grado de dificultad que el usuario ha encontrado para su solución; debe seleccionar una opción y esta será almacenada y considerada para su evaluación y la posible repetición en el tiempo (*SuperMemo 2004*, s. f.), véase Figura 6.

Figura 5. Adverso de una tarjeta Anki



Fuente: Elaboración propia.

Figura 6. Reverso de una tarjeta de Anki



Fuente: Elaboración propia.

4.1.3. Descripción de las especificaciones desarrolladas en el prototipo.

Se ha planteado una metodología para la traducción de frases del inglés al español en el contexto de la enseñanza de estudiantes de doce años, que se lleva a cabo con estudiantes de dicha edad. Se ofrece un contenido con frases en inglés y sus correspondientes traducciones al español, lo que permite una mejor comprensión de las palabras utilizadas en la interpretación del texto. Este recurso es útil para que los alumnos aprendan y practiquen un vocabulario específico mientras trabajan en la traducción de frases. El proceso consiste en presentar la frase en inglés inicialmente para que el estudiante la traduzca al español, y posteriormente se muestra la traducción correcta. Finalmente, el estudiante selecciona la dificultad que considera haber tenido al realizar dicha traducción, basándose en su propio criterio.

Según el criterio de dificultad que el estudiante hubiera elegido, podrá tardar más o menos en repetir, siguiendo el criterio de repetición espaciada.

El programa se basa en las respuestas del alumno para ajustar los intervalos de repetición de las preguntas. Si el estudiante responde correctamente, el intervalo de repetición se prolonga; si el estudiante responde incorrectamente, el intervalo se acorta.

La aplicación presenta las cuestiones en momentos específicos, utilizando el algoritmo de repetición espaciada con el fin de incrementar la retención. Las preguntas se presentan de nuevo justo antes de que el alumno esté a punto de olvidar la información, lo que ayuda a reforzar el recuerdo.

También se realiza un seguimiento del progreso del alumno, registrando las respuestas correctas e incorrectas, y ajustando los intervalos de repetición en consecuencia. Además, se muestran estadísticas sobre el rendimiento del alumno, como el porcentaje de respuestas correctas.

4.1.4. Instrucciones de uso de la aplicación.

A continuación, se describen los pasos para utilizar la aplicación:

- ✓ Ventana de inicio: Al abrir la aplicación, se mostrará al alumno una ventana de inicio en forma de formulario, donde encontrará diversas opciones para iniciar el cuestionario.
- ✓ Selección de usuario: En la ventana de inicio, se presenta una lista que permite al alumno seleccionar su nombre de usuario. Debe elegir su nombre de la lista para comenzar con el cuestionario. Véase Figura 7.

Figura 7. Formulario de Inicio



Fuente: elaboración propia.

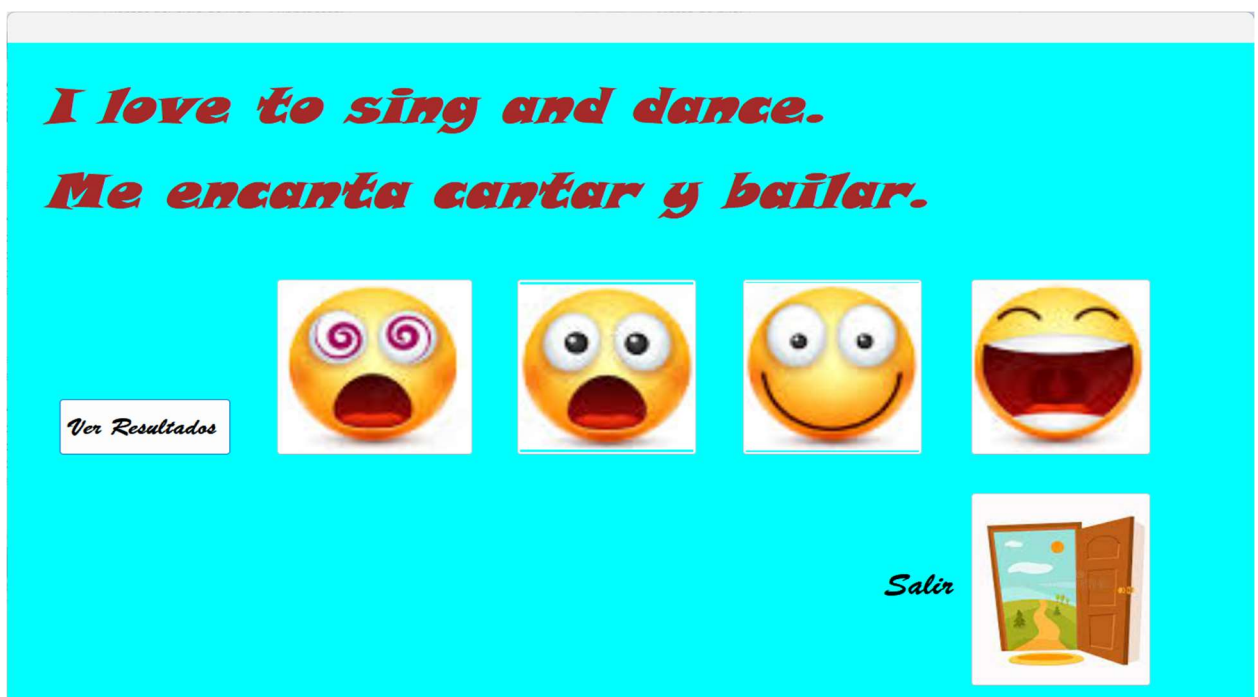
- ✓ Acceso al cuestionario: Una vez seleccionado el alumno, podrá acceder al formulario de realización del cuestionario. Dicho formulario contendrá las preguntas correspondientes al día.
- ✓ Preguntas diarias: El sistema seleccionará automáticamente las preguntas que correspondan al día actual. Estas preguntas se mostrarán de manera secuencial en el formulario de realización.
- ✓ Realización de preguntas: El alumno deberá leer atentamente cada pregunta y proporcionar su respuesta. A continuación, se podrá visualizar la respuesta correcta para verificar la respuesta proporcionada.

Figura 8. Muestra frase para traducir



Fuente: elaboración propia

Figura 9. Frase traducida



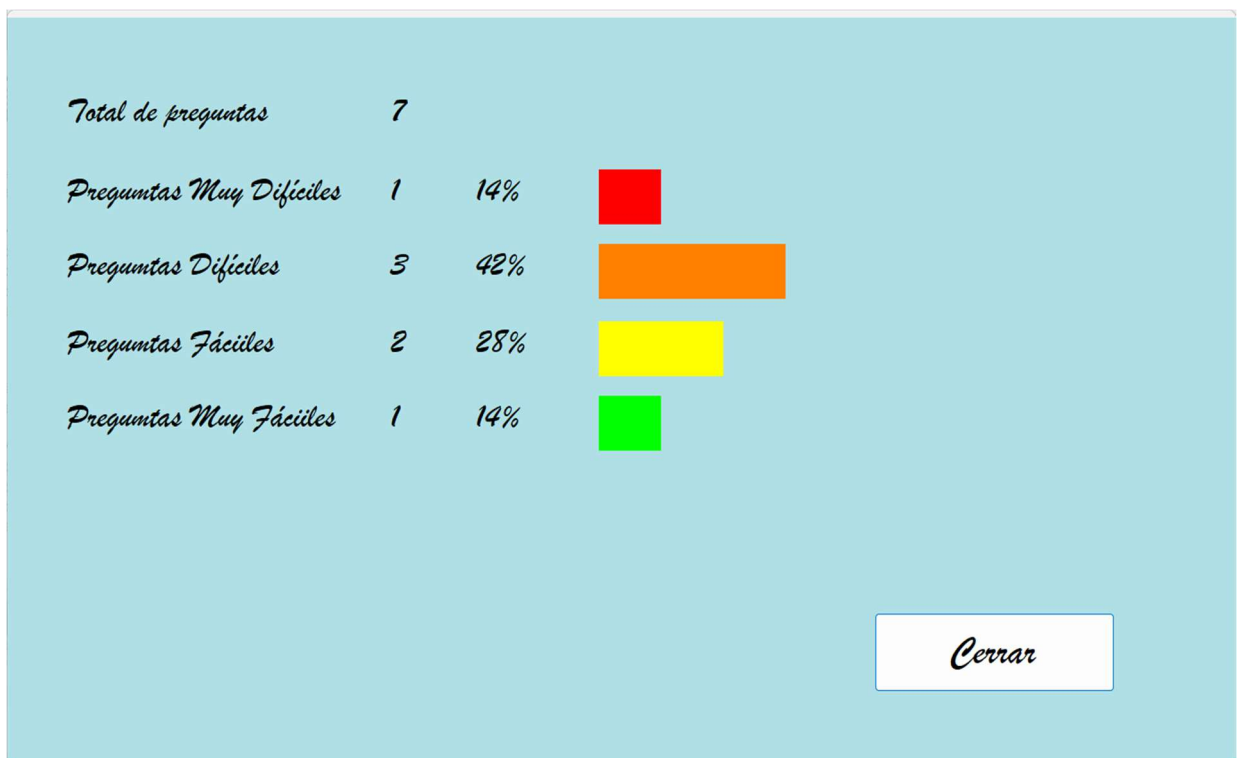
Fuente: elaboración propia

Desarrollo de un prototipo sobre una aplicación de sistemas adaptativos de aprendizaje.

Finalización del cuestionario: El alumno deberá continuar respondiendo las preguntas hasta completar todas las preguntas seleccionadas para el día actual. Si en algún momento se desea salir del cuestionario, se podrá optar por salir y las respuestas serán guardadas.

Visualización de resultados: En cualquier momento, el alumno podrá visualizar los resultados obtenidos. Se podrá acceder a los resultados para verificar el progreso y obtener información sobre el desempeño. Véase Figura 10.

Figura 10. Formulario de resultados



Fuente: elaboración propia

4.2. Diseño

En la fase de diseño se definen la arquitectura de la aplicación. En cuanto a la arquitectura, se establece cómo se estructurarán y organizarán los componentes sistema, así como las tecnologías y lenguajes de programación que se utilizarán. En este caso, se ha decidido crear la aplicación en lenguaje C# utilizando el IDE "Visual Studio".

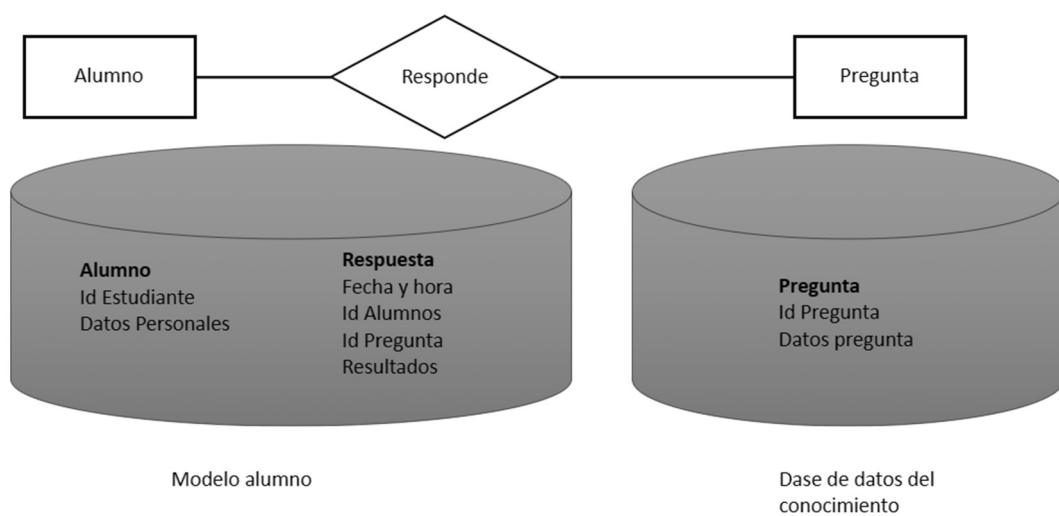
4.2.1. Descripción del sistema

El sistema adaptativo que se desarrolla en esta sección consta de cuatro componentes:

- 1.- El modelo estudiante: mantienen sus datos personales y de los resultados que van adquiriendo.
- 2.- Base del conocimiento: contiene los datos relacionados con los contenidos temáticos que van a ser tratados.

El modelo estudiante y la base del conocimiento constituyen los datos del programa, véase el esquema en la Figura 12.

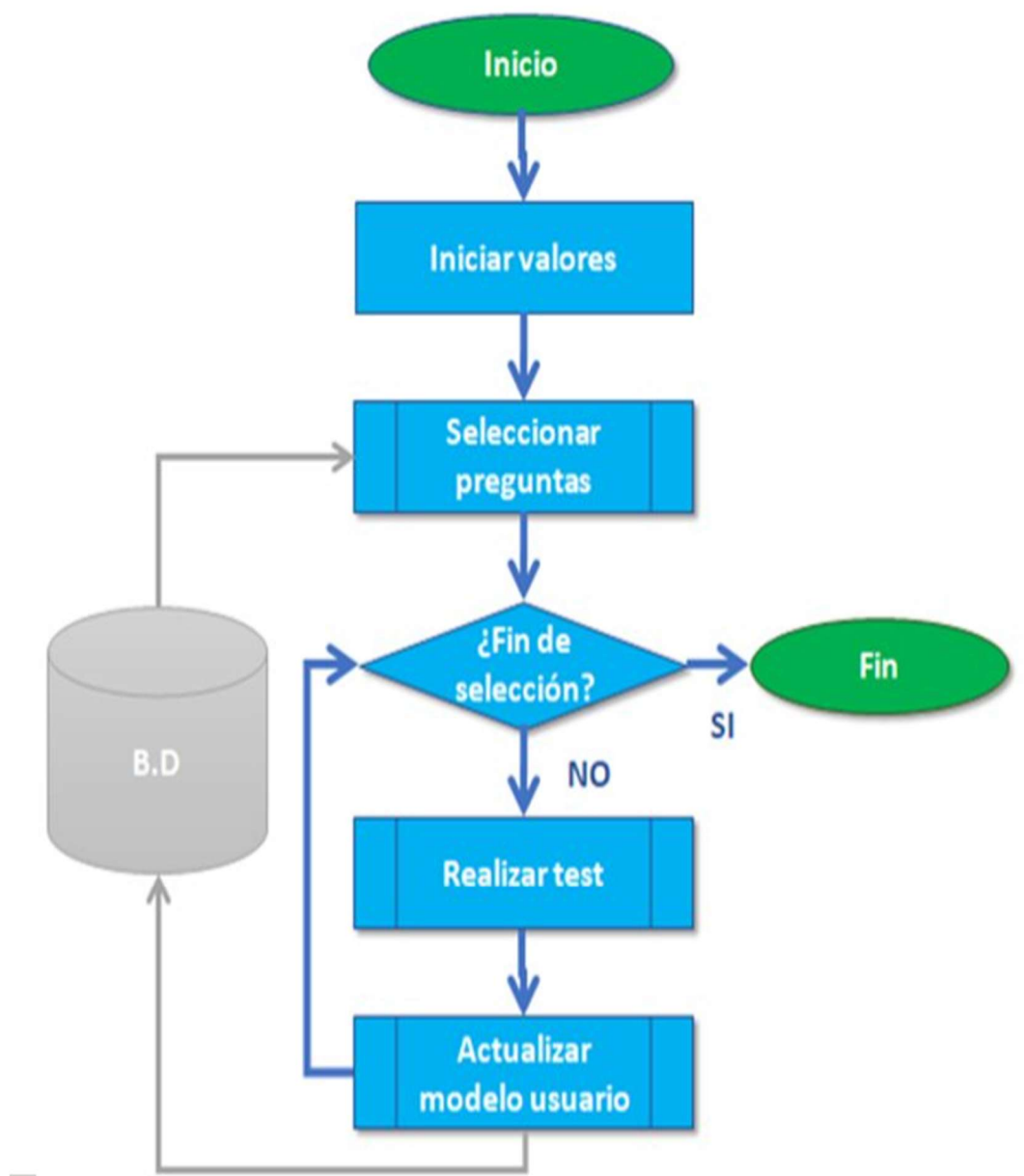
Figura 11. Descripción del modelo de los datos



Fuente: elaboración propia

3.- El modelo instruccional: es la parte del sistema que se encarga de gestionar los procesos. Se toman los datos del conocimiento en función de la información existente en el modelo usuario. Se analizan los conocimientos adquiridos y, con esa información, se actualizan los datos del modelo usuario, véase Figura 13.

Figura 12. El modelo instruccional

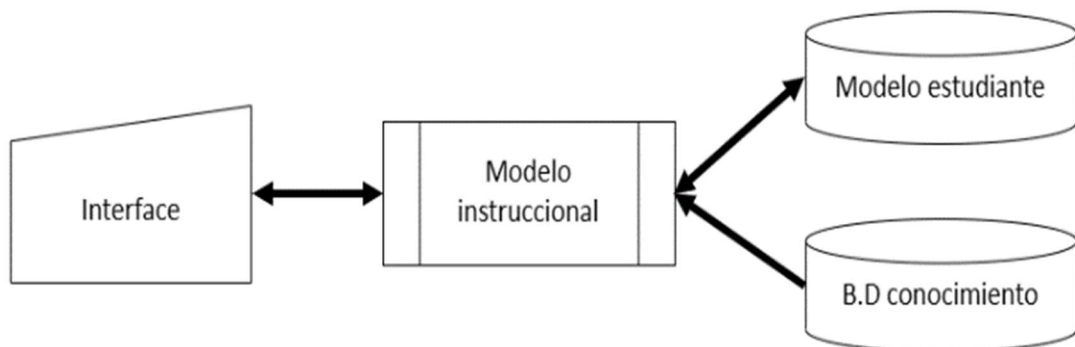


Fuente: elaboración propia

4.- Interfaz: Módulo que se encarga de interactuar con el estudiante.

Un esquema resumen de la arquitectura del sistema se detalla en la Figura 14.

Figura 13. Esquema de la arquitectura.



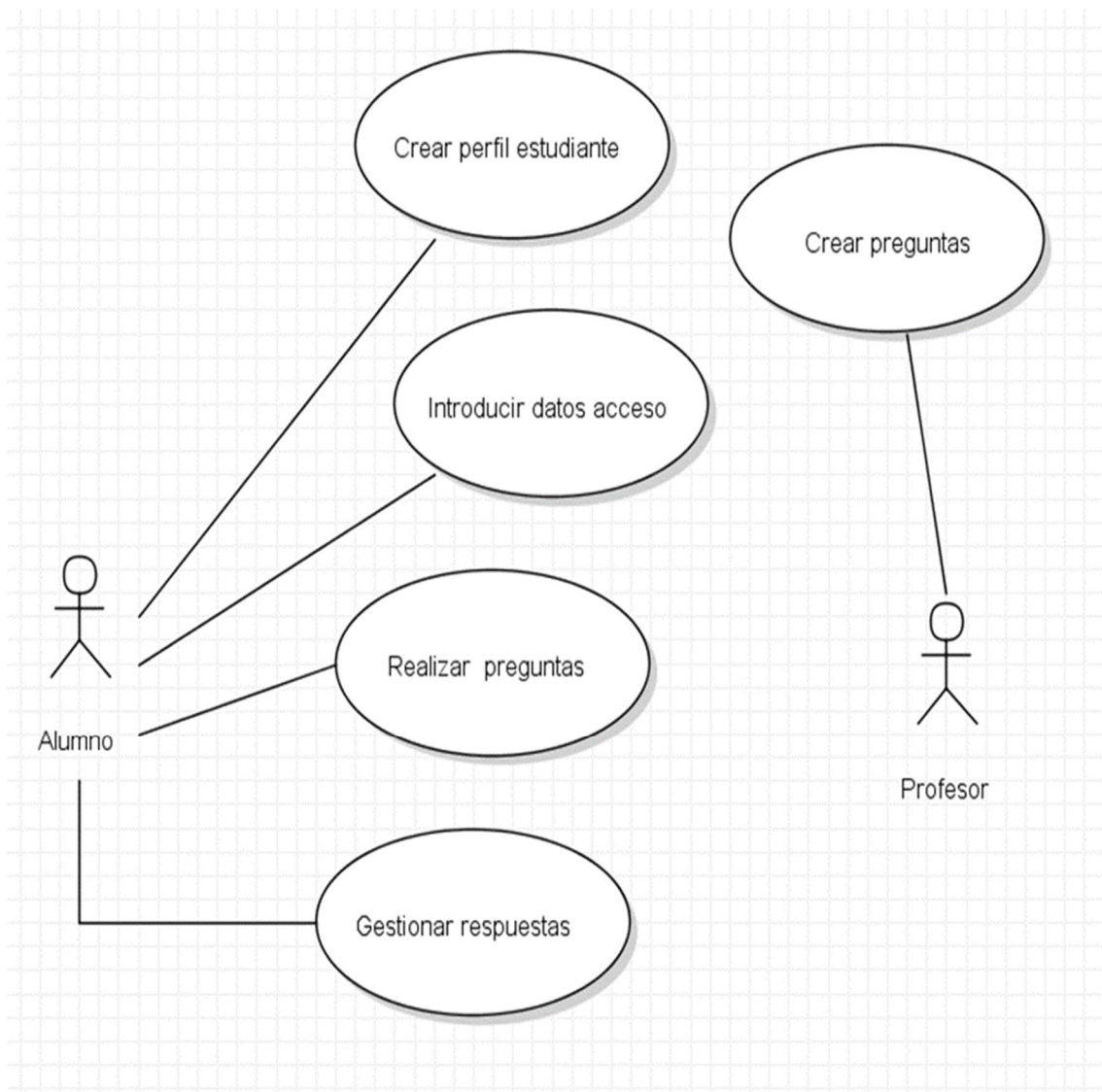
Fuente: elaboración propia

Este trabajo se centra en la creación de procesos propios del modelo instruccional. Los datos del modelo estudiante y la base de datos del conocimiento se administrarán mediante ficheros de tipo texto, dejando la implementación de las bases de datos para estudios posteriores.

4.2.2. Diagrama de casos de uso

Se presentan los diversos escenarios de interacción que existen entre los actores y el sistema, así como las relaciones entre ellos; se brinda una representación gráfica de cómo los usuarios interactúan con el sistema y qué características ofrece el sistema para satisfacer sus demandas (Fowler & Scott, 1999). El diagrama de casos de uso de la aplicación se puede observar en la Figura 11.

Figura 14. Diagrama de casos de uso



Fuente: elaboración propia

Actores:

- ✓ Alumno. Destinatario de la aplicación, realizará los ejercicios propuestos.
- ✓ Profesor: es el encargado de proporcionar la base del conocimiento.

Casos de uso:

- ✓ Crear preguntas: Graba las preguntas diseñadas por el profesor en la base de datos del conocimiento.

Desarrollo de un prototipo sobre una aplicación de sistemas adaptativos de aprendizaje.

- ✓ Crear perfil de estudiante: Gestiona los datos personales del alumno.
- ✓ Introducir datos de acceso: Da paso al inicio de cada sesión, para lo cual el estudiante debe introducir sus datos identificativos.
- ✓ Realizar preguntas: En estudiante interactúa con el sistema respondiendo a las preguntas propuestas.
- ✓ Gestionar respuesta: El sistema actualizará los datos del estudiante en función de los resultados obtenidos.

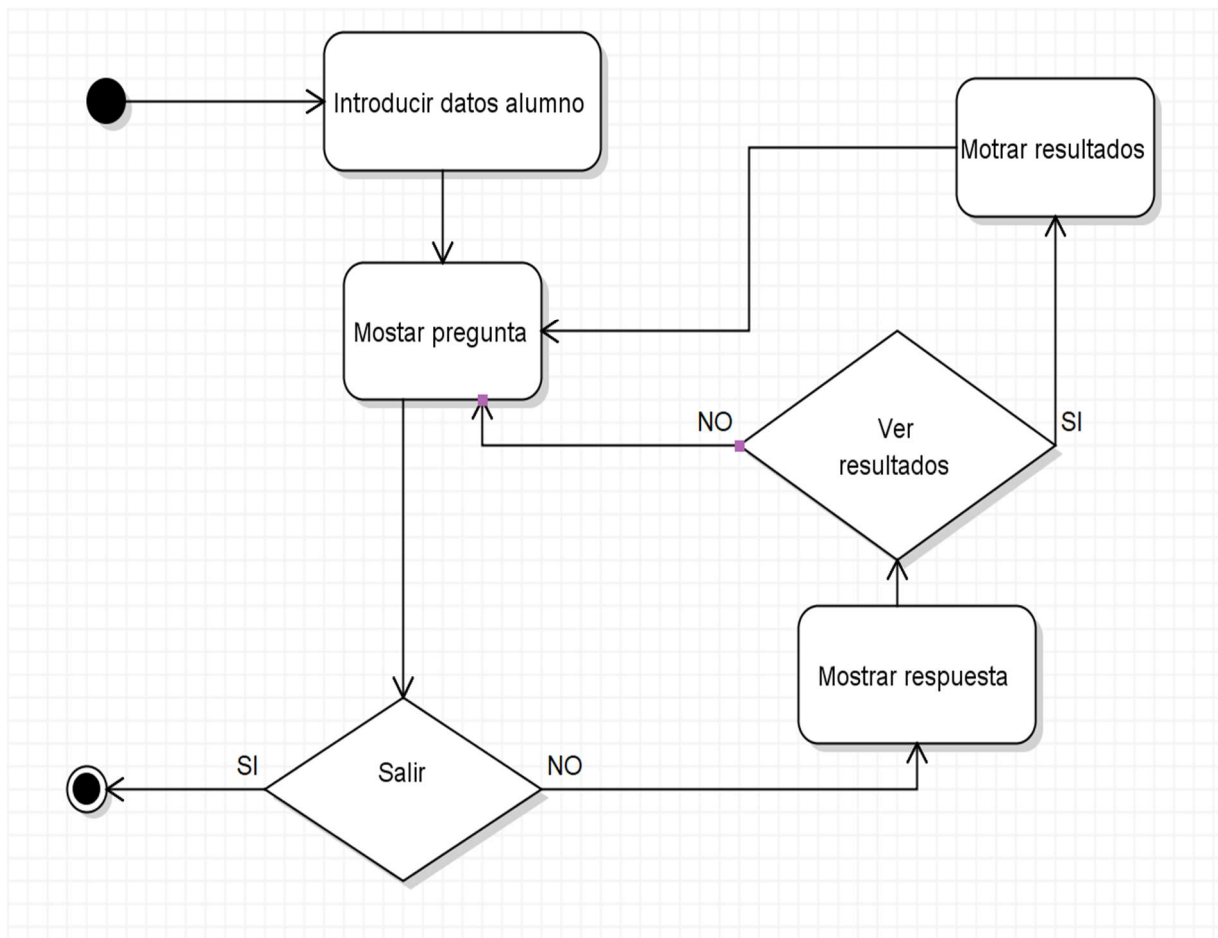
El prototipo desarrollado aquí se centra en los casos de estudios de "Realizar preguntas" y "Gestionar respuesta", dejando los casos de "Crear preguntas", "Crear perfil de estudiante" e "Introducir datos de acceso" para trabajos futuros.

4.2.3. Diagrama de actividades

El diagrama de actividades en UML (Unified Modeling Language) es una herramienta gráfica utilizada para representar el flujo de trabajo o el comportamiento procedencia de un sistema o proceso, utilizada para modelar las actividades y acciones que ocurren en un sistema, dando lugar a la presencia de pasos y decisiones que se toman durante la ejecución de una actividad (Fowler & Scott, 1999).

El modelo instruccional es el encargado de realizar los procesos. A continuación, se describe gráficamente usando el diagrama de actividades. Véase Figura 15.

Figura 15. Diagrama de actividades



Fuente: elaboración propia

4.3. Desarrollo de la aplicación

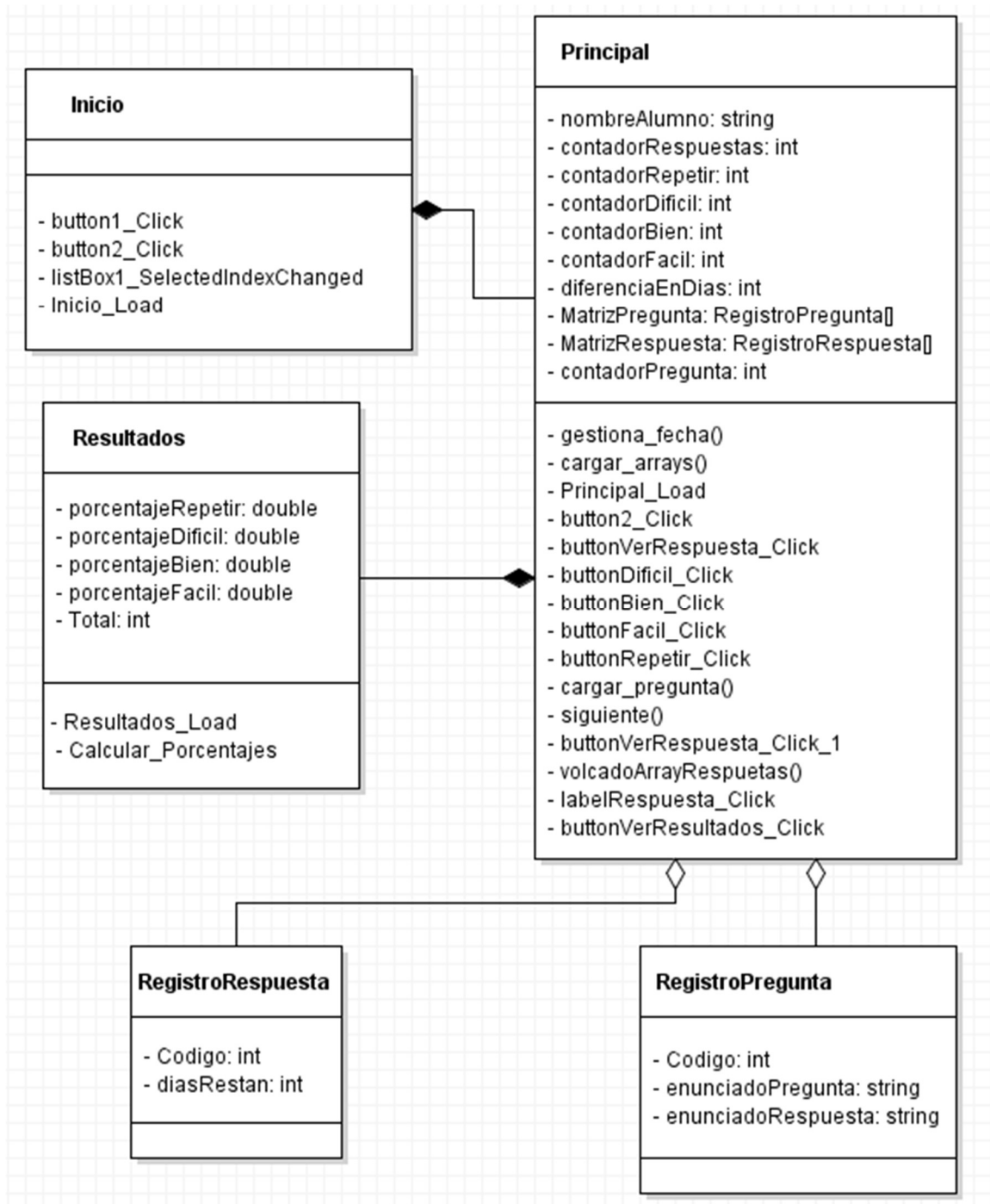
Se ha desarrollado una aplicación didáctica en C# utilizando el entorno de desarrollo Visual Studio. La aplicación se llama "PhraseMaster" y está diseñada para ayudar a los usuarios a aprender y practicar frases en diferentes idiomas.

La aplicación cuenta con una interfaz gráfica de usuario que permite a los usuarios interactuar de manera intuitiva. Utiliza componentes de Windows Forms para crear botones, etiquetas y otros elementos visuales.

4.3.1. Diagrama de clases

Se pueden identificar las siguientes relaciones entre las clases en la Figura 17.

Figura 16. Diagrama de clases



Fuente: elaboración propia

La clase "Principal" tiene una relación de agregación con la clase "RegistroPregunta" y la clase "RegistroRespuesta". Estas dos clases están declaradas dentro de la clase "Principal" como estructuras anidadas. La clase "Principal" tiene matrices de tipo "RegistroPregunta" y "RegistroRespuesta" que almacenan instancias de esas estructuras.

La clase "Inicio" tiene una relación de composición con la clase "Principal". En el método "button1_Click", se crea una instancia de la clase "Principal" y se muestra en pantalla. La clase "Inicio" es responsable de crear y mostrar instancias de la clase "Principal".

La clase "Principal" tiene una relación de composición con la clase "Resultados". En el método "buttonVerResultados_Click", se crea una instancia de la clase "Resultados" y se muestra en pantalla. La clase "Principal" es responsable de crear y mostrar instancias de la clase "Resultados".

4.3.1.1. Descripción de la clase "principal"

Hereda las propiedades y métodos de la clase Form.

Atributos de la clase:

nombreAlumno: variable estática de tipo string que almacena el nombre del alumno.

contadorRespuestas: variable estática de tipo int que lleva la cuenta de las respuestas totales.

contadorRepetir: variable estática de tipo int que lleva la cuenta de las respuestas marcadas como "repetir".

contadorDifícil: variable estática de tipo int que lleva la cuenta de las respuestas marcadas como "difícil".

contadorBien: variable estática de tipo int que lleva la cuenta de las respuestas marcadas como "bien".

contadorFacil: variable estática de tipo int que lleva la cuenta de las respuestas marcadas como "fácil".

diferenciaEnDias: variable de tipo int que almacena la diferencia en días entre la fecha actual y una fecha almacenada en un archivo.

MatrizPregunta: matriz de tipo `RegistroPregunta` que almacena las preguntas y respuestas relacionadas.

MatrizRespuesta: matriz de tipo `RegistroRespuesta` que almacena los códigos y días restantes de respuesta.

Métodos de la clase:

`Principal_Load()`: método que se ejecuta al cargar el formulario y realiza diversas operaciones como la gestión de fechas y la carga de las preguntas y respuestas.

`gestiona_fecha()`: método que gestiona la fecha actual y la diferencia en días entre la fecha actual y una fecha almacenada en un archivo.

`cargar_arrays()`: método que carga las preguntas y respuestas desde archivos y actualiza los días restantes con la diferencia en días.

`button2_Click()`: evento del botón "Salir" que guarda las respuestas en un archivo y cierra la aplicación.

`buttonVerRespuesta_Click()`: evento del botón "Ver Respuesta" que muestra la respuesta a la pregunta y habilita los botones para calificar la respuesta.

`buttonDificil_Click()`: evento del botón "Dificil" que incrementa los días restantes para la pregunta actual y actualiza los contadores correspondientes.

`buttonBien_Click()`: evento del botón "Bien" que incrementa los días restantes para la pregunta actual y actualiza los contadores correspondientes.

`buttonFacil_Click()`: evento del botón "Fácil" que incrementa los días restantes para la pregunta actual y actualiza los contadores correspondientes.

`buttonRepetir_Click()`: evento del botón "Repetir" que incrementa los días restantes para la pregunta actual y actualiza los contadores correspondientes.

`cargar_pregunta()`: método que carga la pregunta y respuesta actual en los controles correspondientes.

`siguiente()`: método que avanza a la siguiente pregunta, carga la nueva pregunta y oculta los controles relacionados a la calificación.

`buttonVerRespuesta_Click_1()`: evento del botón "Ver Respuesta" después de calificar la respuesta, muestra la respuesta y habilita los botones para la siguiente pregunta.

`volcadoArrayRespuestas()`: método que guarda las respuestas en un archivo.

`Resultados_Load()`: método que se ejecuta al cargar el formulario de resultados y realiza diversas operaciones como calcular los porcentajes de las respuestas calificadas.

`Calcular_Porcentajes()`: método que calcula los porcentajes de las respuestas calificadas.

`button1_Click()`: evento del botón "Cerrar" del formulario de resultados, cierra el formulario y regresa al formulario principal.

4.3.1.2. Descripción de la clase "Inicio"

Hereda las propiedades y métodos de la clase Form.

Métodos:

`private void button1_Click(object sender, EventArgs e)`: Este método se ejecuta cuando se hace clic en el botón "button1". Crea una instancia del formulario "Principal" llamado "formularioDestino" y lo muestra en la pantalla mediante el método "Show()".

`private void button2_Click(object sender, EventArgs e)`: Este método se ejecuta cuando se hace clic en el botón "button2". Llama al método "Exit()" de la clase "Application" para cerrar la aplicación.

`private void listBox1_SelectedIndexChanged(object sender, EventArgs e)`: Este método se ejecuta cuando se produce un cambio en la selección del ListBox "listBox1". Verifica si hay un elemento seleccionado en el ListBox y, si es así, asigna el valor de ese elemento a la propiedad estática "nombreAlumno" de la clase "Principal". Además, hace visible el botón "buttonAcceder".

4.3.1.3. Descripción de la clase "Resultados".

Hereda propiedades y métodos de la clase Form.

Atributos:

`private static double porcentajeRepetir`: Es un atributo estático de tipo `double` que almacena el porcentaje de respuestas clasificadas como "Repetir".

`private static double porcentajeDifícil`: Es un atributo estático de tipo `double` que almacena el porcentaje de respuestas clasificadas como "Difícil".

`private static double porcentajeBien`: Es un atributo estático de tipo `double` que almacena el porcentaje de respuestas clasificadas como "Bien".

`private static double porcentajeFacil`: Es un atributo estático de tipo `double` que almacena el porcentaje de respuestas clasificadas como "Fácil".

`private static int Total`: Es un atributo estático de tipo `int` que almacena el total de respuestas.

Métodos:

`private void Resultados_Load(object sender, EventArgs e)`: Este método se ejecuta cuando se carga el formulario "Resultados". Asigna los valores correspondientes a los contadores de respuestas (`contadorRespuestas`, `contadorRepetir`, `contadorDifícil`, `contadorBien`, `contadorFacil`) a las etiquetas correspondientes en el formulario. Luego, calcula los porcentajes para cada categoría de respuestas utilizando el método "`Calcular_Porcentajes()`" y muestra los porcentajes y barras de progreso correspondientes en el formulario.

`private void Calcular_Porcentajes()`: Este método calcula los porcentajes para cada categoría de respuestas en función de los valores de los contadores de respuestas. Utiliza el valor del `contadorRespuestas` para calcular los porcentajes de repetir, difícil, bien y fácil y los almacena en las variables `porcentajeRepetir`, `porcentajeDifícil`, `porcentajeBien` y `porcentajeFacil`, respectivamente.

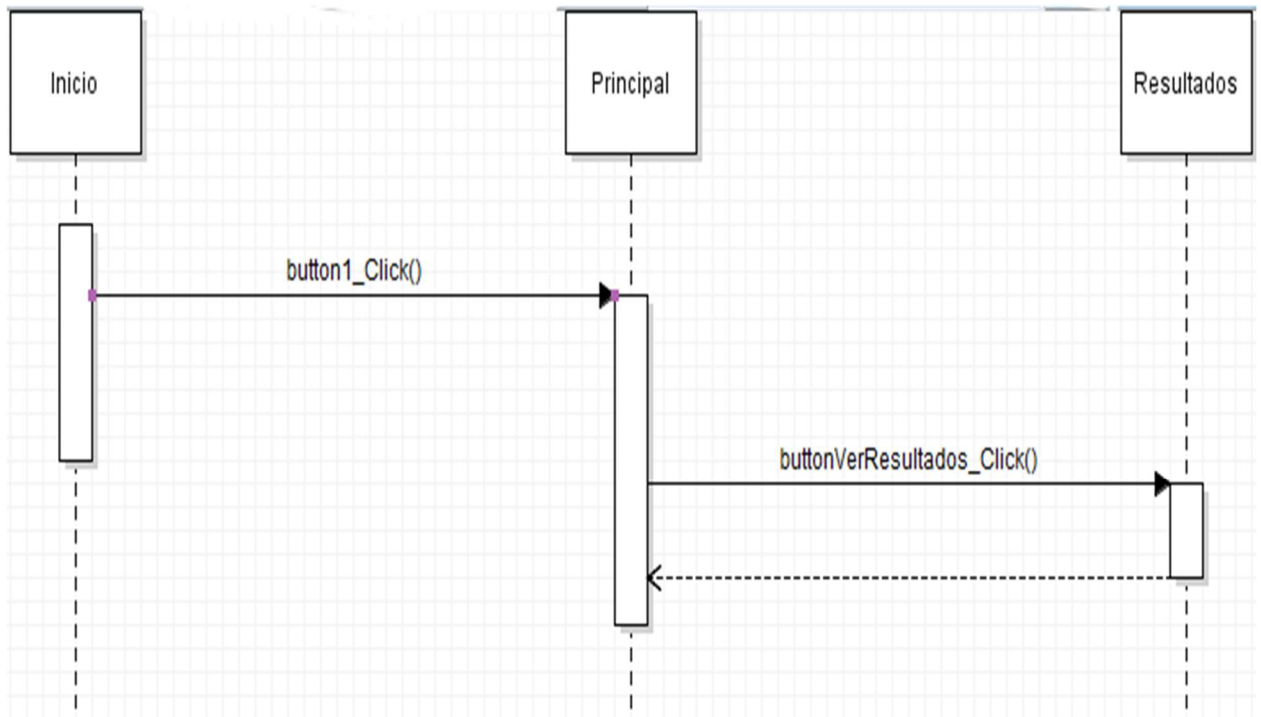
`private void button1_Click(object sender, EventArgs e)`: Este método se ejecuta cuando se hace clic en el botón "button1". Cierra el formulario actual mediante el método "`Close()`".

4.3.2. Diagrama de secuencias

Se describe el flujo de interacción entre los objetos y para identificar el orden de los eventos y las pantallas (Fowler & Scott, 1999).

A continuación, en la Figura 16, se muestra el diagrama de secuencias tal y como lo vería el usuario:

Figura 17. Diagrama de secuencias



Fuente: elaboración propia

La descripción de las secuencias se detalla en las siguientes fases

1. Inicio:
 - a. El usuario inicia la aplicación haciendo clic en un botón en la ventana de inicio.
 - b. Se crea una instancia del formulario Principal y se muestra al usuario.
2. El formulario Principal se carga y realiza varias acciones:
 - a. Llama al método gestiona_fecha para obtener la fecha actual y gestionar un archivo de texto.
 - b. Llama al método cargar_arrays para cargar datos desde archivos de texto en matrices.
 - c. Llama al método cargar_pregunta para mostrar la primera pregunta al usuario.

3. El usuario interactúa con el formulario Principal:
 - a. El usuario hace clic en el botón "Ver Respuesta" para revelar la respuesta a la pregunta actual.
 - b. El usuario selecciona una de las opciones de respuesta (Bien, Difícil, Fácil, Repetir) y hace clic en el botón correspondiente.
 - c. Dependiendo de la opción seleccionada, se actualiza la matriz de respuestas (MatrizRespuesta) con nuevos valores.
 - d. Se incrementa el contador de respuestas y se actualizan los contadores específicos de cada opción de respuesta (contadorRepetir, contadorDifícil, contadorBien, contadorFácil).
4. Verificación de preguntas disponibles:
 - a. Después de cada respuesta, se verifica si se ha alcanzado el límite máximo de preguntas o si aún hay preguntas disponibles.
 - b. Si se ha alcanzado el límite máximo de preguntas, se muestra un mensaje informativo al usuario.
 - c. Si aún hay preguntas disponibles, se muestra la siguiente pregunta al usuario llamando al método siguiente.
5. Repetición de pasos:
 - a. El usuario puede repetir los pasos 3 y 4 hasta que se alcance el límite máximo de preguntas.
6. Salida de la aplicación:
 - a. Cuando el usuario decide salir de la aplicación, hace clic en un botón "Salir" en el formulario Principal.
 - b. Antes de salir, se llama al método volcadoArrayRespuestas para guardar los datos actualizados de la matriz de respuestas en un archivo de texto.
 - c. La aplicación se cierra.

4.4. Implementación

En este apartado se describe el código fuente del programa ya finalizado.

4.4.1. Descripción del fichero "Principal.cs"

Este fichero corresponde a un formulario que permite a los alumnos practicar la traducción de frases del inglés al español. Utiliza archivos de texto para cargar las preguntas y respuestas, registra las respuestas y proporciona una interfaz para ver los resultados.

El código fuente se encuentra en el Anexo A.

Está contenido dentro del espacio de nombres "PhraseMaster" y se define una clase llamada "Principal" que hereda de la clase "Form".

Hay varias variables y estructuras definidas al comienzo del código, incluyendo variables estáticas como "nombreAlumno" y contadores para el seguimiento de respuestas, repeticiones y dificultades.

Se definen dos estructuras llamadas "RegistroPregunta" y "RegistroRespuesta" para almacenar información sobre las preguntas y respuestas.

Hay dos matrices definidas: "MatrizPregunta" y "MatrizRespuesta", que se utilizan para almacenar las preguntas y respuestas cargadas desde archivos de texto.

El método "Principal_Load" se ejecuta cuando se carga el formulario principal y se encarga de gestionar la fecha, cargar las matrices y cargar la primera pregunta.

El método "gestiona_fecha" gestiona la fecha actual y la guarda en un archivo de texto para su posterior uso.

El método "cargar_arrays" carga los datos de las preguntas y respuestas desde archivos de texto en las matrices correspondientes.

Los métodos "buttonDifícil_Click", "buttonBien_Click", "buttonFacil_Click" y "buttonRepetir_Click" se ejecutan cuando se hacen clic en los botones de dificultad y actualizan los días restantes para la próxima pregunta según la dificultad seleccionada.

El método "cargar_pregunta" muestra la pregunta actual en la interfaz de usuario.

El método "siguiente" se llama después de cada respuesta y carga la siguiente pregunta.

El método "buttonVerRespuesta_Click_1" muestra la respuesta de la pregunta actual en la interfaz de usuario.

El método "volcadoArrayRespuestas" guarda los datos actualizados de las respuestas en un archivo de texto.

El método "buttonVerResultados_Click" abre un formulario llamado "Resultados" que muestra los resultados del cuestionario.

4.4.2. Descripción del fichero "Inicio.cs"

La clase "Inicio" se encarga de manejar la pantalla de inicio de la aplicación, permitiendo al usuario seleccionar un elemento de una lista y luego acceder a la pantalla principal de la aplicación con los datos correspondientes al elemento seleccionado.

El código fuente se encuentra en el Anexo B.

El método button1_Click() se ejecuta cuando se hace clic en el botón button1. Dentro de este método, se crea una instancia del formulario Principal y se muestra al usuario mediante el método Show().

El método button2_Click() se ejecuta cuando se hace clic en el botón button2. Su función es cerrar la aplicación mediante el método Exit().

El método listBox1_SelectedIndexChanged() se ejecuta cuando se selecciona un elemento en el listBox1, que es un control de lista. Dentro de este método, se obtiene el elemento seleccionado y se asigna a la propiedad estática nombreAlumno de la clase Principal. Además, se hace visible el botón buttonAcceder.

4.4.3. Descripción del fichero "Resultado.cs"

La clase Resultados es responsable de mostrar los resultados obtenidos por el usuario en la realización del cuestionario. Calcula los porcentajes correspondientes a cada tipo de resultado y los muestra en etiquetas y barras de progreso en el formulario. Además, proporciona un botón para cerrar el formulario y volver a la pantalla principal.

El código fuente se encuentra en el Anexo C.

El método `Resultados_Load()` se ejecuta cuando se carga el formulario de resultados. Dentro de este método, se actualizan las etiquetas (label) del formulario con los resultados obtenidos. Estos resultados son obtenidos de las propiedades estáticas `contadorRespuestas`, `contadorRepetir`, `contadorDifícil`, `contadorBien` y `contadorFacil` de la clase `Principal`. Luego, se llama al método `Calcular_Porcentajes()` para calcular los porcentajes correspondientes a cada tipo de resultado (repetir, difícil, bien y fácil). Finalmente, se actualizan las etiquetas y barras de progreso con los valores calculados.

El método `Calcular_Porcentajes()` se encarga de calcular los porcentajes de cada tipo de resultado en función del total de respuestas. Los resultados se almacenan en las variables estáticas `porcentajeRepetir`, `porcentajeDifícil`, `porcentajeBien` y `porcentajeFacil`.

El método `button1_Click()` se ejecuta cuando se hace clic en el botón `button1`. Su función es cerrar el formulario de resultados llamando al método `Close()`.

4.5. Evaluación

En esta sección se aborda la evaluación de los resultados de la aplicación implementada.

4.5.1. Criterios de evaluación.

El rendimiento de una aplicación didáctica se puede evaluar mediante diversos criterios fundamentales, como la adaptación al propósito educativo, la usabilidad, la interactividad, la personalización y adaptabilidad (Weibelzahl, 2001).

Se considera la exactitud de las traducciones realizadas, verificando si son consistentes y correctas.

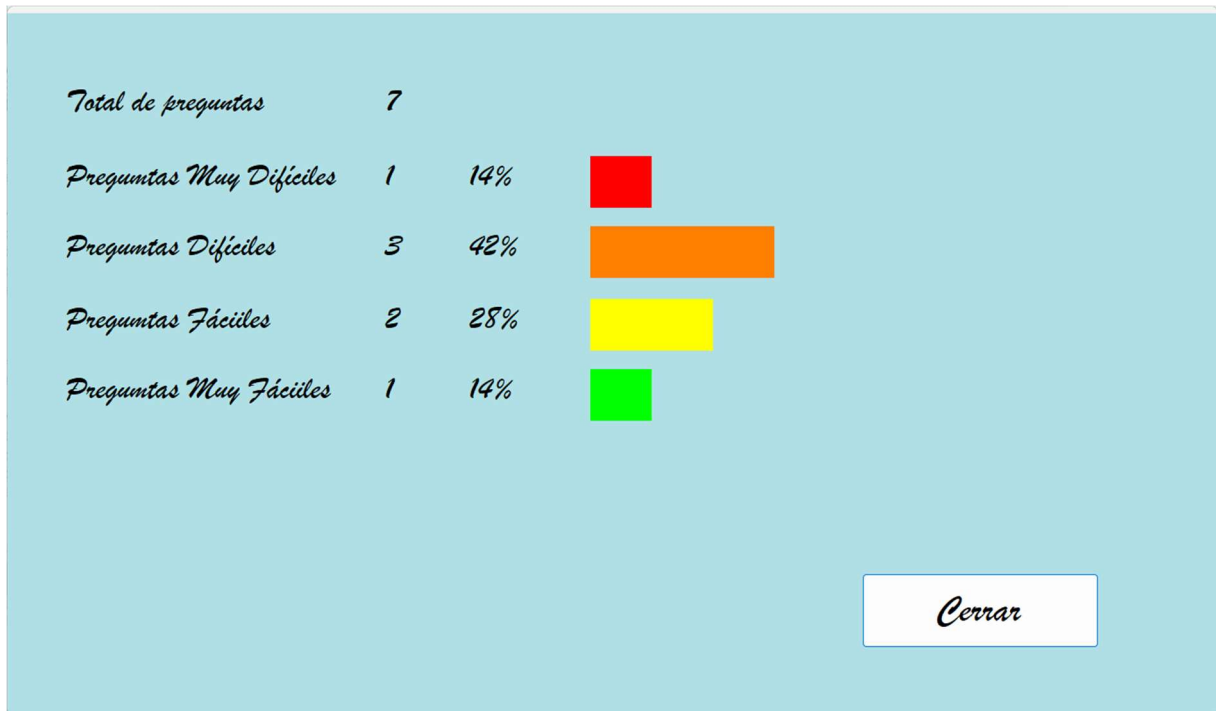
El profesor puede examinar el rendimiento del estudiante durante el uso de la aplicación y llevar a cabo un seguimiento de su progreso a lo largo del tiempo, lo que incluye la revisión de los resultados y la retroalimentación proporcionada.

4.5.2. Herramientas para evaluación

Se recopilaron estadísticas sobre el rendimiento de diez alumnos durante el período de uso de la herramienta. Estas estadísticas incluyeron el número de traducciones realizadas, el porcentaje de respuestas correctas, y el progreso general en el tiempo. Con base en estas estadísticas, se pudo evaluar el rendimiento individual de cada alumno y obtener una visión general del progreso del grupo.

La herramienta proporciona el formulario “Resultados” con un resumen de la evolución de cada sesión, véase figura 18.

Figura 18. Muestra de resultados



Fuente: elaboración propia

Tras la utilización de la aplicación, se procederá a la administración de un cuestionario con el fin de evaluar su experiencia y comprensión. El propósito del cuestionario consiste en adquirir información adicional acerca del aprendizaje y la eficacia de la herramienta de traducción de frases del idioma inglés al español.

Durante las sesiones de aplicación, se llevó a cabo un seguimiento personalizado de cada estudiante con el fin de evaluar su nivel de participación, interacción y compromiso. Se observaron cómo trataban las dificultades de la traducción, su nivel de precisión y la rapidez con la que aprendían y retenían nuevas palabras y frases. Este análisis permitió detectar patrones de aprendizaje individuales y determinar cómo la herramienta afectaba el avance de cada estudiante de forma única.

La revisión y evaluación de los cuestionarios de opinión son una herramienta útil para recopilar el nivel de satisfacción y comentarios del estudiante sobre la aplicación. Permiten analizar aspectos como la motivación, la satisfacción y la percepción del aprendizaje logrado,

Desarrollo de un prototipo sobre una aplicación de sistemas adaptativos de aprendizaje.

lo cual proporciona información relevante para mejorar y adaptar la aplicación de conformidad con las expectativas del estudiante(Jiménez, 2021).

El contenido del cuestionario está detallado en el Anexo D.

5. Conclusiones y trabajo futuro

El estudio se centró en herramientas de traducción, se utilizó repetición espaciada y vocabulario específico para mejorar las habilidades de traducción.

5.1. Conclusiones del trabajo

Se proporciona una descripción detallada de los logros alcanzados en cada objetivo establecido, así como el resultado obtenido tras la evaluación de la herramienta empleada.

5.1.1. Descripción de los logros en cada objetivo parcial

Analizar las necesidades de aprendizaje: Con el fin de examinar las exigencias de aprendizaje, se ha recopilado información acerca del curso seleccionado, el perfil de los estudiantes y su contexto.

Elaborar la arquitectura e interfaz de usuario de la aplicación: Con el fin de diseñar la arquitectura y la interfaz de usuario, se ha llevado a cabo el prototipo de la aplicación enfocada en el perfil del alumno, tal como se ha establecido en el párrafo anterior.

Planificar la estructura y los contenidos: Se han establecido los métodos de aprendizaje más adecuados para el contexto del curso, como la traducción de frases, y se han seleccionado los materiales instruccionales necesarios para el curso.

Evaluar la eficacia de la aplicación concebida: Se ha recopilado datos sobre el rendimiento y la calidad de la instrucción; se han utilizado encuestas para recopilar la retroalimentación de los alumnos.

5.1.2. Resultado de la evaluación de la herramienta

Mejora en el número de traducciones realizadas. Se observó un aumento en el número promedio de traducciones realizadas por cada estudiante a medida que avanzaban en las lecciones. Al comienzo, los estudiantes tienden a traducir un número limitado de frases, pero a medida que se familiarizaban con la aplicación y ganaban confianza, su participación

umentaba. Esto demuestra un mayor compromiso por parte de los estudiantes y su disposición a practicar y aplicar sus habilidades de traducción.

Uno de los factores más significativos en el progreso de los estudiantes fue el incremento en el porcentaje de respuestas correctas en las traducciones. Los estudiantes experimentaron una mejora significativa en su habilidad para comprender y traducir correctamente las frases del inglés al español a medida que avanzaban en las lecciones. Este hecho demuestra una adquisición efectiva de vocabulario y habilidades de traducción.

A medida que los estudiantes continuaron utilizando la aplicación y trabajando en las lecciones, se observó un progreso general en su capacidad para traducir frases más complejas. Al principio, los estudiantes se enfrentaban a desafíos con frases más difíciles, pero con el paso del tiempo lograron una mayor fluidez y precisión en la traducción.

5.2. Líneas de trabajo futuro

Se requiere llevar a cabo pruebas adicionales con un grupo de estudiantes más amplio para confirmar y consolidar los resultados obtenidos hasta el momento en el estudio de la traducción de frases del inglés al español; ampliar la muestra posibilitará detectar patrones y tendencias más precisas, evaluar la efectividad en diversos contextos educativos y recopilar comentarios para optimizar la experiencia de aprendizaje, incrementando de esta forma los resultados de los alumnos.

Sería positivo tener en cuenta la introducción de nuevas funciones en la aplicación, como la ampliación del banco de frases, la introducción de ejercicios interactivos, la presentación de comentarios contextualizada, la integración de recursos de audio, la incorporación de una función de búsqueda y la implementación de un modo de práctica libre. Estas mejoras en la funcionalidad enriquecerían la experiencia de aprendizaje de los estudiantes, fomentando una mayor comprensión y dominio tanto del inglés como del español.

La aplicación actualmente no tiene en cuenta los diferentes perfiles de usuario. Sería recomendable ampliar la información relacionada con los perfiles de usuario y considerar la inclusión de múltiples perfiles para lograr una mejor adaptabilidad a las necesidades individuales de los usuarios.

Referencias bibliográficas

- Allen, W. C. (2006). Overview and Evolution of the ADDIE Training System. *Advances in Developing Human Resources*, 8(4), 430-441.
<https://doi.org/10.1177/1523422306292942>
- Bautista, C. (2010, abril 14). Características del adolescente de educación secundaria y su proceso de crecimiento. *Monografias.com*.
<https://www.monografias.com/trabajos81/caracteristicas-adolescente-proceso-crecimiento/caracteristicas-adolescente-proceso-crecimiento>
- Brusilovsky, P., & Peylo, C. (2003). *Adaptive and Intelligent Web-based Educational Systems*.
- Bull, S., & Kay, J. (2016). Student Models That Invite the Learner In: The SMILI™ Open Learner Modelling Framework Technical Report 580. *I. J. Artificial Intelligence in Education*, 17.
- Casas, M. (2002). *The Use of Skinnerian Teaching Machines and Programmed Instruction in the United States 1960-1970*.
- Chen, Y., Li, X., Liu, J., & Ying, Z. (2018). Recommendation System for Adaptive Learning. *Applied Psychological Measurement*, 42(1), 24-41.
<https://doi.org/10.1177/0146621617697959>
- Chiang, T. H. C., Tsai, T. T. C., Li, I. Y. S., Hsiao, I. Y. T., & Yang, S. J. H. (2012). Technology Enhanced Language Learning in Virtual Worlds. En *Intelligent and Adaptive Learning Systems: Technology Enhanced Support for Learners and Teachers* (pp. 293-309). IGI Global. <https://doi.org/10.4018/978-1-60960-842-2.ch019>
- Christoph, Fröschl, Nguyen, L., & Phung Do. (2023). *Adaptive Learning: Definition, History & Methodology* [Adaptive Learning: Definition, History & Methodology]. study.com.

<https://study.com/academy/lesson/adaptive-learning-definition-history-methodology.html>

Dede, C. (1986). A review and synthesis of recent research in intelligent computer-assisted instruction. *International Journal of Man-Machine Studies*, 24(4), 329-353.
[https://doi.org/10.1016/S0020-7373\(86\)80050-5](https://doi.org/10.1016/S0020-7373(86)80050-5)

Educational Games, Worksheets, and More for Kids. (s. f.). Recuperado 3 de julio de 2023, de <https://www.education.com/>

English Grammar, Vocabulary, Pronunciation Exercises for ESL Teachers and Students. (s. f.). Recuperado 3 de julio de 2023, de <https://www.esltower.com/>

Fowler, M., & Scott, K. (1999). *UML gota a gota: Actualizado para cubrir la version 1*. Pearson Educación.

George E. Hein. (1991, octubre 15). *Constructivist Learning Theory*. Exploratorium.
<https://www.exploratorium.edu/education/ifi/constructivist-learning>

Hernández, A. M. (2013). *Aplicación para memorizar usando repetición espaciada*.
<https://upcommons.upc.edu/handle/2099.1/16934>

Hong, H. & Kinshuk. (2020). Mobile agents in adaptive learning systems. *IEEE International Workshop on Wireless and Mobile Technologies in Education Proceedings*, 165-166.
<https://doi.org/10.1109/WMTE.2002.1039243>

<https://apps.ankiweb.net/>. (s. f.). *Anki: Potentes e inteligentes tarjetas didácticas*. Recuperado 8 de abril de 2023, de <https://apps.ankiweb.net/>

<https://es.duolingo.com/>. (s. f.). *Aprende un idioma completamente gratis*. Duolingo. Recuperado 8 de abril de 2023, de <https://es.duolingo.com/>

<https://es.khanacademy.org/>. (s. f.). *Khan Academy | Práctica, lecciones y cursos en línea gratuitos*. Recuperado 8 de abril de 2023, de https://es.khanacademy.org/_render

<https://www.aleks.com/>. (s. f.). *ALEKS – Adaptive Learning & Assessment for Math, Chemistry, Statistics & More*. McGraw Hill ALEKS. Recuperado 8 de abril de 2023, de <https://www.aleks.com>

<https://www.gazeta.gt/la-cibernetica-y-el-hacer-retroalimentador/>. (s. f.). *Ideal_feedback_model.png (750x309)*. Recuperado 8 de abril de 2023, de https://www.gazeta.gt/wp-content/uploads/2022/06/Ideal_feedback_model.png

<https://www.smartsparrow.com/>. (s. f.). *Smart Sparrow*. Smart Sparrow. Recuperado 8 de abril de 2023, de <https://www.smartsparrow.com/>

Jiménez, M. P. (2021, abril 16). *Importancia de las encuestas en el aula*. https://colaboraeducacion30.juntadeandalucia.es/educacion/colabora/web/21401gt031/inicio/-/blogs/importancia-de-las-encuestas-en-el-aula?_33_redirect=https%3A%2F%2Fcolaboraeducacion30.juntadeandalucia.es%2Feducacion%2Fcolabora%2Fweb%2F21401gt031%2Finicio%3Fp_p_id%3D33%26p_p_lifecycle%3D0%26p_p_state%3Dnormal%26p_p_mode%3Dview%26p_p_col_id%3Dcolumn-2%26p_p_col_count%3D1

Mareels, I., & Polderman, J. W. (1996). Adaptive Systems. En I. Mareels & J. W. Polderman (Eds.), *Adaptive Systems: An Introduction* (pp. 1-26). Birkhäuser. https://doi.org/10.1007/978-0-8176-8142-5_1

Maribe Branch, Robert. (2009). *Instructional Design: The ADDIE Approach*. Springer.

Mihalca, R., Andreescu, A., & Întorsureanu, I. (2008). *Knowledge Management in E-Learning Systems*.

National Museum of American History. (s. f.). *Pressey Teaching Machine*. Smithsonian Institution. Recuperado 6 de abril de 2023, de https://www.si.edu/object/pressey-teaching-machine%3Anmah_1367149

Nguyen, L., Do, P., & Fröschl, C. (2008). *Learner Model in Adaptive Learning*.

Pugliese, L. (2016, octubre 17). *Adaptive Learning Systems: Surviving the Storm*. Sistemas de aprendizaje adaptativo: sobrevivir a la tormenta. <https://er.educause.edu/articles/2016/10/adaptive-learning-systems-surviving-the-storm>

Reddy, S., Labutov, I., & Banerjee, S. (2016). A Queueing Network Model for Spaced Repetition. *Proceedings of the Third (2016) ACM Conference on Learning @ Scale*, 289-292. <https://doi.org/10.1145/2876034.2893436>

Reverte Oliver, B. (2023). La traducción como herramienta didáctica y de accesibilidad en las aulas de inglés de las escuelas oficiales de idiomas. *Didáctica (lengua y literatura)*. <https://doi.org/10.5209/dill.79629>

Rodríguez, S., & Perez, M. (2023). *Aplicación de Ingeniería Inversa a Documentos de Diseño de software: Adaptación al Compilador DocC para Generación de Diagramas de Clase*. <https://doi.org/10.13140/RG.2.2.36786.38087>

Sadovaya, V. V., Korshunova, O. V., & Nauruzbay, Z. Z. (2016). Personalized Education Strategies. *International Electronic Journal of Mathematics Education*, 11(1), 199-209. <https://doi.org/10.29333/iejme/324>

- Salamea Bravo, M. J., González Palacio, L., Oriol Hilari, M., & Farré Tost, C. (2020). *Estimación y priorización de requisitos no-funcionales para desarrollo de software: Estado del arte*. 150-157. <https://upcommons.upc.edu/handle/2117/336760>
- Sarıyalçınkaya, A., Karal, H., Altınay, F., & Altınay, Z. (2021). *Reflections on Adaptive Learning Analytics: Adaptive Learning Analytics* (pp. 61-84). <https://doi.org/10.4018/978-1-7998-7103-3.ch003>
- Serrano, J. E. A., & Gascó, E. R. (2009). *Tema 1. LA TRANSICIÓN ADOLESCENTE Y LA EDUCACIÓN*.
- SuperMemo 2004*. (s. f.). Recuperado 4 de abril de 2021, de <http://www.cc.kyoto-su.ac.jp/information/tesl-ej/ej40/m1.html>
- Tabibian, B., Upadhyay, U., De, A., Zarezade, A., Schölkopf, B., & Gomez-Rodriguez, M. (2019). Enhancing human learning via spaced repetition optimization. *Proceedings of the National Academy of Sciences*, 116(10), 3988-3993. <https://doi.org/10.1073/pnas.1815156116>
- Terol, M. (2019, agosto 27). *Aprendizaje adaptativo: Qué es, cuáles son sus características y cómo aplicarlo*. Aprendizaje adaptativo: qué es, cuáles son sus características y cómo aplicarlo. <https://www.virtualpro.co/noticias/aprendizaje-adaptativo--que-es--cuales-son-sus-caracteristicas-y-como-aplicarlo>
- Violet UML Editor: Easy to use, completely free*. (s. f.). Recuperado 3 de julio de 2023, de <http://alexdp.free.fr/violetumleditor/page.php>
- Visual Studio: IDE y Editor de código para desarrolladores de software y Teams*. (s. f.). Visual Studio. Recuperado 4 de julio de 2023, de <https://visualstudio.microsoft.com/es/>

- Walkington, C., & Bernacki, M. L. (2020). Appraising research on personalized learning: Definitions, theoretical alignment, advancements, and future directions. *Journal of Research on Technology in Education*, 52(3), 235-252.
<https://doi.org/10.1080/15391523.2020.1747757>
- Weibelzahl, S. (2001). *Evaluation of Adaptive Systems* (p. 294). https://doi.org/10.1007/3-540-44566-8_49
- Williamson, B. (2017). *Big Data in Education: The digital future of learning, policy and practice*.
<https://doi.org/10.4135/9781529714920>

Anexo A. Código fuente de “Principal.cs”.

Descrito en el punto 4.4.1

```

using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using System.Windows.Forms;
using System.Xml.Linq;

namespace PhraseMaster
{
    public partial class Principal : Form
    {
        public static string nombreAlumno;
        public static int contadorRespuestas = 0;
        public static int contadorRepetir = 0;
        public static int contadorDificil = 0;
        public static int contadorBien = 0;
        public static int contadorFacil = 0;
        private int diferenciaEnDias;

        public struct RegistroPregunta
        {
            public int Codigo;
            public string enunciadoPregunta;
            public string enunciadoRespuesta;
        }

        public struct RegistroRespuesta
        {
            public int Codigo;
            public int diasRestan;
        }

        public static RegistroPregunta[] MatrizPregunta = new
RegistroPregunta[100];
        public static RegistroRespuesta[] MatrizRespuesta = new
RegistroRespuesta[100];

        private int contadorPregunta = 0;

        public Principal()
        {
            InitializeComponent();
        }

        private void Principal_Load(object sender, EventArgs e)
        {
            gestiona_fecha();
        }
    }
}

```

Desarrollo de un prototipo sobre una aplicación de sistemas adaptativos de aprendizaje.

```

cargar_arrays();
cargar_pregunta();

}
private void gestiona_fecha()
{
    DateTime fechaActual = DateTime.Now;
    // Ruta del archivo de texto
    string rutaArchivo =
Path.Combine(AppDomain.CurrentDomain.BaseDirectory, "Datos" + nombreAlumno +
".txt");

    // Verificar si el archivo existe
    if (File.Exists(rutaArchivo))
    {
        // Leer la fecha desde el archivo
        string fechaTexto = File.ReadAllText(rutaArchivo);

        // Convertir la fecha del texto a DateTime
        if (DateTime.TryParse(fechaTexto, out DateTime fecha))
        {
            // Mostrar la fecha en la etiqueta (label)
            TimeSpan diferencia = fechaActual - fecha;
            diferenciaEnDias = diferencia.Days;
        }
    }
    else
    {
        // El archivo no existe
        MessageBox.Show("El archivo no existe ", "Error",
MessageBoxButtons.OK, MessageBoxIcon.Error);
    }

    File.WriteAllText(rutaArchivo, fechaActual.ToString());

}

private void cargar_arrays()
{
    // Ruta del archivo de texto
    string rutaArchivoBD =
Path.Combine(AppDomain.CurrentDomain.BaseDirectory, "BDConocimiento.txt");
    //label1.Text = "Respuestas" + nombreAlumno + ".txt";
    string rutaArchivoRespuesta =
Path.Combine(AppDomain.CurrentDomain.BaseDirectory, "Respuestas" + nombreAlumno +
".txt");

    // Verificar si el archivo existe
    if (File.Exists(rutaArchivoBD) && File.Exists(rutaArchivoRespuesta))
    {
        // Leer el archivo línea por línea
        using (StreamReader sr1 = new StreamReader(rutaArchivoBD))
        using (StreamReader sr2 = new StreamReader(rutaArchivoRespuesta))
        {
            string lineaPregunta;
            String lineaRespuesta;
            int i = 0;
            while ((lineaPregunta = sr1.ReadLine()) != null &&
(lineaRespuesta = sr2.ReadLine()) != null)

```

Desarrollo de un prototipo sobre una aplicación de sistemas adaptativos de aprendizaje.

```

    {
        string[] datosPregunta = lineaPregunta.Split('\t');
        MatrizPregunta[i].Codigo = int.Parse(datosPregunta[0]);
        MatrizPregunta[i].enunciadoPregunta = datosPregunta[1];
        MatrizPregunta[i].enunciadoRespuesta = datosPregunta[2];
        string[] datosRespuesta = lineaRespuesta.Split('\t');
        MatrizRespuesta[i].Codigo = int.Parse(datosRespuesta[0]);
        MatrizRespuesta[i].diasRestan =
int.Parse(datosRespuesta[1]);
        // Actualizacion de los dias que restan con la diferencia
de dias
        if (MatrizRespuesta[i].diasRestan > 0)
        {
            MatrizRespuesta[i].diasRestan =
MatrizRespuesta[i].diasRestan - diferenciaEnDias;
            if (MatrizRespuesta[i].diasRestan < 0)
            {
                MatrizRespuesta[i].diasRestan = 0;
            }
        }

        i++;
    }
}
else
{
    MessageBox.Show("El archivo no existe.--", "Error",
MessageBoxButtons.OK, MessageBoxIcon.Error);
}
}

private void button2_Click(object sender, EventArgs e)
{
    volcadoArrayRespuetas();
    Application.Exit();
}

private void buttonVerRespuesta_Click(object sender, EventArgs e)
{
    buttonBien.Visible = true;
    buttonDificil.Visible = true;
    buttonFacil.Visible = true;
    buttonRepetir.Visible = true;
    if (contadorPregunta == 100)
    {
        MessageBox.Show("No hay más preguntas por hoy", "Aviso",
MessageBoxButtons.OK, MessageBoxIcon.Information);
    }
}

private void buttonDificil_Click(object sender, EventArgs e)
{
    MatrizRespuesta[contadorPregunta].diasRestan =
MatrizRespuesta[contadorPregunta].diasRestan + 5;
    contadorRespuestas++;
    contadorDificil++;
}

```

```

        buttonVerResultados.Visible = true;
        siguiente();
    }

    private void buttonBien_Click(object sender, EventArgs e)
    {
        MatrizRespuesta[contadorPregunta].diasRestan =
MatrizRespuesta[contadorPregunta].diasRestan + 10;
        contadorRespuestas++;
        contadorBien++;
        buttonVerResultados.Visible = true;
        siguiente();
    }

    private void buttonFacil_Click(object sender, EventArgs e)
    {
        MatrizRespuesta[contadorPregunta].diasRestan =
MatrizRespuesta[contadorPregunta].diasRestan + 20;
        contadorRespuestas++;
        contadorFacil++;
        buttonVerResultados.Visible = true;
        siguiente();
    }

    private void buttonRepetir_Click(object sender, EventArgs e)
    {
        contadorRespuestas++;
        contadorRepetir++;
        buttonVerResultados.Visible = true;
        siguiente();
    }

    private void cargar_pregunta()
    {
        labelEnunciado.Text =
MatrizPregunta[contadorPregunta].enunciadoPregunta;
        labelRespuesta.Text =
MatrizPregunta[contadorPregunta].enunciadoRespuesta;
    }

    private void siguiente()
    {
        contadorPregunta++;
        cargar_pregunta();
        buttonVerRespuesta.Visible = true;
        labelRespuesta.Visible = false;
        buttonBien.Visible = false;
        buttonDificil.Visible = false;
        buttonFacil.Visible = false;
        buttonRepetir.Visible = false;
    }

    private void buttonVerRespuesta_Click_1(object sender, EventArgs e)
    {
        labelRespuesta.Visible = true;
        buttonVerRespuesta.Visible = false;
        buttonBien.Visible = true;
    }

```

Desarrollo de un prototipo sobre una aplicación de sistemas adaptativos de aprendizaje.

```
        buttonDificil.Visible = true;
        buttonFacil.Visible = true;
        buttonRepetir.Visible = true;
    }

    private void volcadoArrayRespuestas()
    {
        string rutaArchivoRespuesta =
Path.Combine(AppDomain.CurrentDomain.BaseDirectory, "Respuestas" + nombreAlumno +
".txt");

        using (StreamWriter escritor = new
StreamWriter(rutaArchivoRespuesta))
        {

            foreach (RegistroRespuesta datos in MatrizRespuesta)
            {
                escritor.WriteLine($"{datos.Codigo}\t{datos.diasRestan}");
            }

        }

    }

    private void buttonVerResultados_Click(object sender, EventArgs e)
    {
        Resultados formularioDestino = new Resultados();

        formularioDestino.Show();
    }
}
}
```

Anexo B. Código fuente de “Inicio.cs”.

Descrito en el punto 4.4.2

```
using System.Windows.Forms;

namespace PhraseMaster
{
    public partial class Inicio : Form
    {
        public Inicio()
        {
            InitializeComponent();
        }

        private void button1_Click(object sender, EventArgs e)
        {
            Principal formularioDestino = new Principal();

            formularioDestino.Show();
        }

        private void button2_Click(object sender, EventArgs e)
        {
            Application.Exit();
        }

        private void listBox1_SelectedIndexChanged(object sender, EventArgs e)
        {
            if (listBox1.SelectedItem != null)
            {
                Principal.nombreAlumno = listBox1.SelectedItem.ToString();
                buttonAcceder.Visible = true;
            }
        }
    }
}
```

Anexo C. Código fuente de “Resultados.cs”.

Descrito en el punto 4.4.3

```

using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using System.Windows.Forms;

namespace PhraseMaster
{
    public partial class Resultados : Form
    {
        private static double porcentajeRepetir;
        private static double porcentajeDificil;
        private static double porcentajeBien;
        private static double porcentajeFacil;
        private static int Total;

        public Resultados()
        {
            InitializeComponent();
        }

        private void Resultados_Load(object sender, EventArgs e)
        {
            labelTotal.Text = Principal.contadorRespuestas.ToString();
            labelRepetir.Text = Principal.contadorRepetir.ToString();
            labelDificil.Text = Principal.contadorDificil.ToString();
            labelBien.Text = Principal.contadorBien.ToString();
            labelFacil.Text = Principal.contadorFacil.ToString();
            Calcular_Porcentajes();
            labelPorcRepetir.Text = porcentajeRepetir.ToString() + "%";
            labelPorcDificil.Text = porcentajeDificil.ToString() + "%";
            labelPorcBien.Text = porcentajeBien.ToString() + "%";
            labelPorcFacil.Text = porcentajeFacil.ToString() + "%";
            pictureRojo.Width = (int)porcentajeRepetir * 5;
            pictureNaranja.Width = (int)porcentajeDificil * 5;
            pictureAmarillo.Width = (int)porcentajeBien * 5;
            pictureVerde.Width = (int)porcentajeFacil * 5;
        }

        private void Calcular_Porcentajes()
        {
            Total = Principal.contadorRespuestas;
            porcentajeRepetir = Principal.contadorRepetir * 100 / Total;
            porcentajeDificil = Principal.contadorDificil * 100 / Total;
            porcentajeBien = Principal.contadorBien * 100 / Total;
            porcentajeFacil = Principal.contadorFacil * 100 / Total;
        }

        private void button1_Click(object sender, EventArgs e)
        {
            this.Close();
        }
    }
}

```

Anexo D. Ejemplo de cuestionario de opinión.

¿Cuál es tu opinión general sobre la aplicación de aprendizaje de inglés adaptativo?

- a) Muy satisfactoria
- b) Satisfactoria
- c) Neutral
- d) Insatisfactoria
- e) Muy insatisfactoria

¿Cómo describirías la eficacia de la aplicación para ayudarte a aprender inglés?

- a) Muy eficaz
- b) Eficaz
- c) Moderadamente eficaz
- d) Poco eficaz
- e) Nada eficaz

¿En qué medida la aplicación se adapta a tus necesidades y nivel de aprendizaje?

- a) Se adapta perfectamente
- b) Se adapta bien
- c) Se adapta adecuadamente
- d) No se adapta lo suficiente
- e) No se adapta en absoluto

¿Qué opinas sobre la traducción de inglés a español proporcionada por la aplicación?

Desarrollo de un prototipo sobre una aplicación de sistemas adaptativos de aprendizaje.

- a) Muy precisa y confiable
- b) Precisa y confiable
- c) Aceptable
- d) Inexacta en ocasiones
- e) Muy inexacta e inconsistente

¿Consideras que la aplicación te ha ayudado a mejorar tu comprensión y uso del vocabulario en inglés?

- a) Mucho
- b) Bastante
- c) Moderadamente
- d) Poco
- e) Nada

¿Cómo calificarías la interfaz y la usabilidad de la aplicación?

- a) Muy intuitiva y fácil de usar
- b) Intuitiva y fácil de usar
- c) Razonablemente intuitiva y fácil de usar
- d) Poco intuitiva y difícil de usar
- e) Muy poco intuitiva y difícil de usar

¿Qué tan motivado te sientes al utilizar la aplicación para aprender inglés?

- a) Muy motivado
- b) Motivado
- c) Moderadamente motivado

d) Poco motivado

e) Nada motivado

¿Recomendarías esta aplicación a otros estudiantes de tu edad que deseen aprender inglés?

a) Definitivamente sí

b) Probablemente sí

c) No estoy seguro/a

d) Probablemente no

e) Definitivamente no

¿Tienes algún comentario adicional o sugerencia para mejorar la aplicación de aprendizaje de inglés adaptativo?