



Universidad Autónoma del Estado de México

Centro Universitario UAEM Valle de Chalco

**METODOLOGÍA PARA LA IMPLEMENTACIÓN DE SOFTWARE EDUCATIVO BASADA
EN ELEMENTOS TECNOLÓGICOS Y EDUCATIVOS CENTRADOS EN EL PROCESO
ENSEÑANZA-APRENDIZAJE**

T E S I S

QUE PARA OBTENER EL GRADO DE

DOCTOR EN CIENCIAS DE LA COMPUTACIÓN

P R E S E N T A

MTRO. DANIEL BENITO MORAN

DIRECTORA:

DRA. ANABELEM SOBERANES MARTÍN

CO-DIRECTORA:

DRA. MAGALLY MARTÍNEZ REYES

TUTORA:

DRA. MARÍA DE LOURDES LÓPEZ GARCÍA

VALLE DE CHALCO SOLIDARIDAD, MÉXICO MARZO 2023



CUVCH

**METODOLOGÍA PARA LA IMPLEMENTACIÓN
DE SOFTWARE EDUCATIVO BASADA EN
ELEMENTOS TECNOLÓGICOS Y EDUCATIVOS
CENTRADOS EN EL PROCESO ENSEÑANZA-
APRENDIZAJE**

RESUMEN

En ocasiones, las metodologías convencionales para el desarrollo de software no satisfacen las necesidades educativas debido a que se enfocan en automatizar un proceso lineal, que contrasta con el proceso de enseñanza-aprendizaje, el cual es adaptativo al contexto. Asimismo, en la mayoría de los casos, teorías educativas y procesos ingenieriles no son conjuntados de manera óptima durante el desarrollo del software educativo, lo que a veces resulta en productos que no cumplen con los resultados esperados.

En este sentido, el desarrollo de software educativo requiere una planificación cuidadosa y detallada, que incluya la identificación de las necesidades de los actores de proceso enseñanza-aprendizaje, definición de los objetivos de aprendizaje, especificación de los requisitos del software y la selección de las herramientas o tecnologías adecuadas para su creación. Una vez que se ha planificado y diseñado el software educativo, se procede a su implementación y evaluación, que implica la programación, pruebas y revisión de la aplicación para asegurarse de que cumple con los requisitos y objetivos de aprendizaje establecidos.

De esta manera, la propuesta nombrada “*Metodología Conjunta para Aplicaciones Educativas*” o MCAE establece (a través de métodos, instrumentos y técnicas) los elementos necesarios para los equipos desarrolladores, con el fin de que se cumpla lo mencionado en el párrafo anterior.

El presente trabajo siguió una investigación aplicada abordando la fundamentación, problematización y conceptualización de una metodología para el desarrollo de software educativo considerando elementos utilizados en las áreas que tienen injerencia durante este proceso -educación y tecnología-.

MCAE está basada en el ciclo de vida de desarrollo de software y las etapas del proceso enseñanza-aprendizaje, en ella se proponen instrumentos para la planeación, control y ejecución del proyecto, los cuales funcionan como técnicas que combinan las necesidades a cubrir de cada campo de estudio para obtener elementos funcionales durante la elaboración de aplicaciones informáticas que sirvan de apoyo en el contexto educativo.

La propuesta contribuirá a mejorar las prácticas de los equipos de trabajo multidisciplinarios, así como fungir de guía. Por último, se espera que el software, desarrollado bajo esta propuesta, aumente la probabilidad de cumplir con los objetivos de aprendizaje para los que se construya.

Palabras clave: Ingeniería de sistemas, Metodología, Método de enseñanza, Software didáctico, Tecnología educativa.

ABSTRACT

It is occasionally difficult for conventional methodologies for software development to satisfy educational needs because they seek to automate a linear process, which contrasts with the teaching-learning process that is adaptive to the context. Likewise, aspects such as educational theories and engineering processes are not optimally combined during the development of educational software, which is why, sometimes, the products do not provide optimal results.

In this sense, the development of educational software requires careful and detailed planning, which includes the identification of the needs of the actors in the teaching-learning process, the definition of learning objectives, the specification of the software requirements and the selection of the tools or technologies suitable for its creation. Once the educational software has been planned and designed, it proceeds to its implementation and evaluation, which involves programming, testing, and reviewing the application to ensure that it meets the established requirements or learning objectives.

In this way, the proposal named “Joint Methodology for Educational Applications” or MCAE (Because of the Spanish acronym) establishes, (through methods, instruments, and techniques) the necessary elements for the development teams, in order to comply with what is mentioned in the previous paragraph.

The present work followed an applied investigation addressing the foundation, problematization and conceptualization of a methodology for the development of educational software considering elements used in the areas that have interference during this process -education and technology-.

MCAE is based on the software development life cycle and the stages of the teaching-learning process; it proposes instruments for the planning, control, and execution of the project, which serve as techniques that combine the needs to be covered in each field. Study to obtain functional elements during the development of computer applications that serve as support in the educational context.

The proposal will contribute to improve the practices of multidisciplinary work teams, as well as act as a guide. Finally, it is expected that the software, developed under this proposal, increases the probability of meeting the learning objectives for which it is built.

Keywords: Educational methods, Educational software, Educational technology, Methodology, Systems design.

ÍNDICE

1. INTRODUCCIÓN.....	17
1.1 Justificación.....	17
1.2 Planteamiento del problema.....	19
1.3 Hipótesis.....	21
1.4 Objetivos.....	21
General.....	21
Específicos.....	22
2. MARCO TEÓRICO.....	23
2.1. Fundamento educativo.....	24
Proceso Enseñanza-Aprendizaje.....	25
Teorías educativas.....	27
2.2. Tecnología Educativa.....	31
2.3. Fundamento tecnológico.....	34
Desarrollo de software.....	35
Software educativo.....	36
Metodologías para el desarrollo de software educativo.....	38
2.4. Análisis comparativo de metodologías de software educativo.....	45

3. METODOLOGÍA	54
4. PROPUESTA	57
4.1. Metodología Conjunta para Aplicaciones Educativas (MCAE).....	60
Fase 1: Análisis tecnopedagógico.....	63
Fase 2: Diseño tecnopedagógico	76
Fase 3: Implementación.....	79
Fase 4. Pruebas.....	80
Fase 5: Mantenimiento funcional y educativo.....	82
Fase transversal: JAD (Diseño Conjunto de Aplicaciones).....	84
5. RESULTADOS.....	89
5.1. Validación por juicio de expertos.....	89
5.2. Validación por equipos de trabajo.....	103
5.3. Uso de MCAE para el desarrollo de un REA.....	109
6. CONCLUSIONES.....	145
REFERENCIAS.....	151
ANEXOS.....	168

ÍNDICE DE TABLAS

Tabla 1. Análisis de elementos del proceso enseñanza-aprendizaje y desarrollo de software entre metodologías	49
Tabla 2. Análisis de los productos obtenidos de las metodologías investigadas..	52
Tabla 3. Equipo de trabajo para el desarrollo del proyecto.....	58
Tabla 4. Fiabilidad y validez del cuestionario aplicado a los especialistas.....	100
Tabla 5. Usabilidad percibida por parte de los equipos de desarrollo.....	105
Tabla 6. Procedimiento para la determinación del nivel de usabilidad para MCAE	107
Tabla 7. Respuestas obtenidas a partir de la aplicación del SUS	139
Tabla 8. Procedimiento SUS aplicado a las repuestas de profesores y estudiantes	142

ÍNDICE DE FIGURAS

Figura 1. Organigrama propuesto para el desarrollo del proyecto	59
Figura 2. Esquematización de la Metodología Conjunta para Aplicaciones Educativas	62
Figura 3. Cargas de trabajo para cada integrante en las diferentes etapas de MCAE	63
Figura 4. Conceptualización de la definición tecnopedagógica	64
Figura 5. Ejemplo de plantilla para la orden del día.....	87
Figura 6. Ejemplo de plantilla para las minutas de reunión.....	88
Figura 7. Porcentaje de participación durante el desarrollo de proyectos educativos en cada nivel académico	91
Figura 8. Experiencia en años de los especialistas entrevistados	91
Figura 9. Respuestas a la pregunta: MCAE es clara es para su aplicación durante el desarrollo de software educativo	93
Figura 10. Respuestas a la pregunta: MCAE cumple con los métodos suficientes para el desarrollo de software educativo.....	94
Figura 11. Respuestas a la pregunta: MCAE cumple con las técnicas suficientes para el desarrollo de software educativo	94
Figura 12. Respuestas a la pregunta: MCAE cumple con las estrategias suficientes para el desarrollo de software educativo	94

Figura 13. Respuestas a la pregunta: MCAE cumple con los instrumentos suficientes para el desarrollo de software educativo	95
Figura 14. Respuestas a la pregunta: MCAE contiene los métodos adecuados para el desarrollo de software educativo	95
Figura 15. Respuestas a la pregunta: Los métodos son claros para su aplicación durante el desarrollo de software educativo	96
Figura 16. Respuestas a la pregunta: Los métodos contienen los elementos suficientes para el desarrollo de software educativo.....	97
Figura 17. Respuestas a la pregunta: Las técnicas son claras para su aplicación durante el desarrollo de software educativo	97
Figura 18. Respuestas a la pregunta: Las técnicas contienen los elementos suficientes para el desarrollo de software educativo	97
Figura 19. Respuestas a la pregunta: Las estrategias son claras para su aplicación durante el desarrollo de software educativo	98
Figura 20. Respuestas a la pregunta: Las estrategias contienen los elementos suficientes para el desarrollo de software educativo	98
Figura 21. Respuestas a la pregunta: Los instrumentos son claros para su aplicación durante el desarrollo de software educativo	98
Figura 22. Respuestas a la pregunta: Los instrumentos contienen los elementos suficientes para el desarrollo de software educativo	99
Figura 23. Mockup de la distribución del REA	124
Figura 24. Mockup de la presentación del vocabulario	125

Figura 25. Mockup de la presentación del tema	125
Figura 26. Bosquejo de diseño para las acciones del usuario en el sistema	126
Figura 27. Bosquejo de diseño para el orden de las acciones del estudiante en el sistema	127
Figura 28. Mapa del sitio del REA	128
Figura 29. Diagrama de casos de uso del REA	128
Figura 30. Diagrama de secuencia para “realizar actividades”	128
Figura 31. Diagrama de secuencia para “estudiar tema”	129
Figura 32. Pantalla de principal para el REA	131
Figura 33. Pantalla presentación del tema	132
Figura 34. Pantalla presentación del tema	133
Figura 35. Pantalla de bienvenida	134
Figura 36. Pantalla de motivación para los estudiantes	134
Figura 37. Pantalla donde se reutilizan materiales externos	135
Figura 38. Pantalla donde se presenta el vocabulario	136
Figura 39. Pantallas para la producción propia del estudiante (lectura, escritura y oral)	137

1. INTRODUCCIÓN

Esta investigación da continuidad al proyecto titulado *Desarrollo de un sistema web desde el enfoque de tecnología educativa para difundir información sobre el reino fungi*, debido a que fueron identificadas problemáticas durante el desarrollo de software educativo, desde desconocimiento en los alcances de las áreas involucradas (educación y tecnología) hasta inconvenientes significativos de conjunción (Benito, 2019; Benito y Soberanes, 2021). En esta sección se desarrolla por qué y para qué de la investigación, la hipótesis que se busca validar y los objetivos a los que se pretende llegar.

1.1 Justificación

Algunos autores establecen que no necesariamente las metodologías tradicionales no funcionan para la creación de software educativo, pero sí plantean la necesidad de explorar nuevas metodologías y enfoques para el desarrollo de software educativo que sean más eficaces y adaptables a las necesidades de los usuarios y entorno educativo (Tovar *et al.* 2014; Plomp, 2007; García *et al.*, 2002). Debido a

que las metodologías para el desarrollo de software fueron concebidas para automatizar, en su mayoría, procesos lineales, lo cual no concuerda con la naturaleza de la enseñanza y aprendizaje, que indica que son procesos adaptativos; derivado de los problemas de conjunción entre las áreas de tecnología y educación (Area, 2009).

Dicho esto, la importancia de esta investigación radica en entender estas problemáticas que surgen desde la concepción de la Tecnología Educativa (TE) y la dificultad histórica que ha significado la unión de ambas áreas que la conforman, con el fin de proponer una metodología que desarrolle software que impacte de manera positiva al proceso enseñanza-aprendizaje.

En este sentido, para llegar a un software educativo funcional, el cual cumpla con el objetivo para el que fue creado, es indispensable incluir aspectos pedagógicos en las fases tempranas de la metodología (análisis y diseño), garantizando así que el usuario final pueda satisfacer sus necesidades educativas, por lo que las características del usuario deben ser consideradas en el proceso de desarrollo de software (Galvis, 1992).

A estos aspectos, desde el enfoque tecnológico, son denominados usabilidad de sistemas; por otro lado, desde el punto de vista educativo, se les conoce como aprendizaje centrado en el alumno. Ambos sitúan al alumno o usuario al centro del proceso educativo, considerando sus necesidades, características y problemáticas para ofrecerle una educación adaptada a él. Si las técnicas utilizadas en el diseño instruccional están diseñadas específicamente para el individuo, el

procesamiento de información mejora, y se obtiene un aprendizaje significativo (De Luque, 2006); de aquí que, al contar con un software que responde a las características particulares de los estudiantes, la realización de un diseño instruccional se facilita y optimiza para el profesor.

En función de calificaciones, varios estudios (Galindo, 2015; Pérez, 2014; Morales y Vera, 2007; entre otros) revelan que el índice de reprobación disminuye, de manera que, el aprendizaje obtenido por los estudiantes que son instruidos con apoyo de un software educativo aumenta, además del desarrollo de otras habilidades tales como: la colaboración durante el proceso educativo, incremento de la creatividad en los productos que se presentan, sensibilización y dominio de los medios informáticos, por mencionar algunas (Guerrero *et al.*, 2016). Del mismo modo, el uso de una terminología propia de la TE, establece un proceso de desarrollo tecnopedagógico homogéneo, puesto que no se presta a interpretaciones como sucede con los conceptos utilizados hasta ahora (Delavant y Fernández, 2008).

Por lo mencionado, toma importancia este proyecto de investigación para coordinar los fundamentos pedagógicos y desarrollo ingenieril durante la generación de software educativo.

1.2 Planteamiento del problema

Las aplicaciones informáticas son utilizadas como una forma de adquirir información de carácter educativo, científico, cultural, entre otros; por estas razones resultan ser de las principales fuentes de recursos informativos en la actualidad. De manera que, la mayoría de información que se maneja en ellas

recurrentemente no es filtrada, en función de calidad, o presentada de una manera idónea para convertirla en conocimiento.

En consecuencia, la información resulta ser compleja para la comprensión de los usuarios debido a que las aplicaciones no están pensadas en facilitar un proceso de entendimiento, y menos aún, de conversión a conocimiento. En el ámbito educativo se busca realizar esta transformación para lograr sus objetivos, lo cual ha sido un problema histórico durante la construcción de este tipo de aplicaciones.

En este contexto, a causa de una separación de las áreas que conforman a la TE, y como resultado, no son conjuntados de manera adecuada aspectos de ambas ciencias tales como las teorías educativas y procesos ingenieriles, debido a esto, la terminología empleada durante el desarrollo del software que apoya al proceso enseñanza-aprendizaje resulta ser ambigua por una deficiente homologación de los conceptos educativos y tecnológicos. Al existir esta separación, se construyen sistemas tecnológicos que cubren la necesidad de automatizar un proceso, en consecuencia, el sistema resulta poco efectivo; del mismo modo, al no vislumbrar el alcance y las limitaciones de cada área, el desarrollo de materiales didácticos digitales y aplicaciones educativas se ven restringidas.

En este sentido, se plantean las siguientes preguntas de investigación:

- ¿Cómo integrar elementos tecnológicos y educativos durante el desarrollo de una metodología para generar software educativo que permita una forma conjunta de trabajo entre tecnología y educación?
- ¿Qué elementos debe contener la metodología basada en el estudiante para generar software educativo que sirva como apoyo en el proceso enseñanza-aprendizaje?

1.3 Hipótesis

El enlace de elementos pedagógicos con tecnológicos en el diseño de una metodología de desarrollo de software educativo centrada en los actores del proceso enseñanza-aprendizaje, mediante una integración idónea de métodos, instrumentos o técnicas; planteará una forma conjunta de trabajo a las áreas de tecnología y educación, en consecuencia, el proceso enseñanza-aprendizaje apoyado por el uso de la tecnología será beneficiado.

1.4 Objetivos

Se presenta la meta a la cual se dirigió la investigación, así mismo, los alcances y delimitaciones que se contemplaron durante el desarrollo de esta.

General

Desarrollar una metodología fundamentada en estrategias pedagógicas y tecnológicas para la implementación de software educativo a fin de establecer una unificación en las áreas que conforman a la TE.

Específicos

- a) Investigar el estado del arte, de tal manera que permita determinar los elementos a considerarse para el desarrollo de la metodología.
- b) Identificar conceptos utilizados en investigaciones relacionadas con la TE y así esbozar una posible terminología de esta área.
- c) Analizar estrategias de tecnología y educación a integrar en la metodología, para la creación de software educativo que sirva como apoyo en el cumplimiento de los objetivos de aprendizaje.
- d) Determinar el contexto en que se aplicará la metodología, para observar cómo se realiza el proceso de desarrollo.
- e) Evaluar la eficacia de la metodología propuesta mediante el análisis del software educativo desarrollado con esta, para determinar el impacto que representa durante el proceso enseñanza-aprendizaje.

En los siguientes capítulos se expondrán los fundamentos teóricos del área tecnológica y pedagógica en los que se basó la investigación, identificando los componentes que cumplen con actividades clave durante el desarrollo de software educativo. Además, se describirá la metodología empleada para llevar a cabo este proyecto y, por último, se presentará la conceptualización y uso de la propuesta desarrollada.

2. MARCO TEÓRICO

Durante los últimos cuarenta años, el concepto de tecnología ha cambiado a causa del interés que se le dedica a cada componente que lo constituye; por ejemplo, en principio, el foco de atención eran los medios (información); posteriormente, fue la comunicación; con el paso de los años, se incorporó a la educación, lo cual (y dado por las posibilidades educativas), se facilitó por el avance de la tecnología tanto en la información como en la comunicación. En este sentido, debido a esta naturaleza cambiante, se ha buscado adaptar estas características y evolución en la educación de forma extensa e inclusiva, buscando apoyar y optimizar el proceso enseñanza-aprendizaje, y con ello, alcanzar de manera óptima los objetivos de aprendizaje (Mujica, 2020).

En esta sección se desarrolla el soporte teórico de la presente tesis, se inicia con principios educativos tales como corrientes psicológicas, teorías educativas, proceso enseñanza-aprendizaje, entre otros; se plantean sus definiciones,

características y principios fundamentales. Después, se establecen conceptos de Tecnología Educativa y, por último, en el sustento tecnológico se determinan términos y aspectos relevantes durante el desarrollo de software.

2.1. Fundamento educativo

La enseñanza y el aprendizaje se han estudiado como un proceso en conjunto, en el cual no se considera un concepto separado del otro; sin embargo, surge una división en el período moderno del pensamiento humano que se da en el sentido de las aportaciones de la psicología y la pedagogía. La psicología retoma de la filosofía el análisis de la construcción del pensamiento y aprendizaje a partir de algunas escuelas o corrientes filosóficas. A continuación, se mencionan las siguientes como fundamento:

- El racionalismo: el conocimiento se deriva de la razón, sin la participación de los sentidos (Schunck, 2012).
- El estructuralismo: la conciencia posee una estructura compleja, por lo cual es necesario estudiarla por separado (Schunck, 2012).
- El funcionalismo: expone que los procesos mentales y las conductas nos ayudan a adaptarnos por lo cual el conocimiento está ligado a la experiencia (Schunck, 2012).
- El empirismo: todos los conceptos surgen en la experiencia, son sobre o aplicables a cosas que pueden ser experimentadas, asimismo, todas las creencias o proposiciones racionalmente aceptables son justificables o cognoscibles sólo a través de la experiencia (Duignan *et al.*, 2020).

- El nihilismo (explícito y activo): propone la transformación de los principios y valores educativos tradicionales, a la construcción de nuevas concepciones e interpretaciones educativas que requiere el mundo moderno (Ruíz y Pachano, 2006).

Proceso Enseñanza-Aprendizaje

Esta separación continúa hasta la década de los 50, donde Highet remarca la importancia de vincular la enseñanza y aprendizaje en una única conceptualización (Schunck, 2012). En este sentido, de manera tradicional, el proceso educativo a modo de actividad conjunta del profesor y los estudiantes se realiza en dos procesos relacionados: enseñanza es la actividad docente y aprendizaje la estudiantil; sin embargo, adecuando un significado contemporáneo, ambas actividades son desempeñadas indistintamente por los actores mencionados. A este proceso se le llama enseñanza-aprendizaje, de acuerdo con Paladines (2016) es “el procedimiento mediante el cual se transmiten conocimientos especiales o generales sobre una materia, sus dimensiones en el fenómeno del rendimiento académico a partir de los factores que determinan su comportamiento” (pág. 136). También es concebido por Abreu *et al.* (2018) como:

...el espacio en el cual el principal protagonista es el alumno y el profesor cumple con una función de facilitador de los procesos de aprendizaje. Son los alumnos quienes construyen el conocimiento a partir de leer, de aportar sus experiencias y reflexionar sobre ellas, de intercambiar sus puntos de vista con sus compañeros y el profesor. En este espacio, se pretende que el alumno disfrute el aprendizaje y se comprometa con él de por vida (pág. 612).

A partir de las definiciones presentadas del proceso de enseñanza-aprendizaje (PE-A), es posible deducir su importancia en la educación, del mismo modo, dada su naturaleza compleja, una rama de la pedagogía dedicada al análisis del PE-A resulta necesaria, a esta área se le denomina *didáctica*; Contreras (1994) la define como la disciplina encargada de explicar el PE-A, con el fin de generar propuesta para su aplicación con fines educativos.

De acuerdo con Flores (1997), la didáctica en calidad de ciencia ha atravesado distintas etapas, hasta llegar a lo que hoy se conoce como didáctica crítica, Morán (2004) plantea el análisis de la práctica docente, los roles de los actores involucrados y el significado de lo que nace de esta interacción. Para realizar este análisis, es necesario conocer los elementos que integran al PE-A (Flores, 1997):

- Instrumentación didáctica: entendida como la planeación didáctica, en la cual se organizan diversos factores con el fin de favorecer el desarrollo de habilidades y actitudes por parte del alumno, en un periodo determinado.
- Conceptualización del aprendizaje: cabe señalar que es un proceso no lineal, en el cual el docente se desempeña a manera de mediador de contenidos y planea actividades.
- Teoría del aprendizaje: sustentada en la constructivista y de la información.
- Programa: es un eje imprescindible, pues en él se encuentran plasmados los aprendizajes mínimos, dosificados en un periodo de tiempo en el cual el estudiante debe consolidarlos.
- Objetivos: definen lo que se espera que el alumno logre.

- Contenidos: están condicionados por avances científicos, tecnológicos y sociales, se modifican acorde a lo que el alumno requiere apropiarse.
- Estrategias didácticas de enseñanza: diseñadas y utilizadas por el facilitador o “agente de enseñanza para promover y facilitar el aprendizaje significativo de los estudiantes” (Alonso-Tapia citado por Flores *et al.*, 2017, pág. 13).
- Estrategias didácticas de aprendizaje: “utilizadas por el estudiante para reconocer, aprender y aplicar la información y/o contenido” (Alonso-Tapia citado por Flores *et al.*, 2017, pág. 13).
- Evaluación: permite al docente y estudiante conocer y valorar el progreso y entendimiento alcanzado de los temas explorados.

Se hace evidente que el PE-A tiene como eje central la ejecución de un programa de contenidos dosificados, en la planeación, realizada por el docente, se plasma la organización secuencial de los temas, al igual que las actividades programadas, en función de lograr los objetivos de aprendizaje (Flores, 1997).

Teorías educativas

El PE-A integra teorías que buscan apoyar de manera óptima durante el aprendizaje y enseñanza, Gimeno (2008) define a esta última como la técnica que orienta el aprendizaje con el fin de consolidar objetivos y establece que la teoría de la enseñanza es la encargada de estructurar de modo sistemático al aprendizaje, evidenciando así, la relación entre estos dos conceptos.

En este sentido, la teoría de la enseñanza guía los elementos que intervienen en el aprendizaje, de forma que se requiere tomar en cuenta las diversas situaciones

en las que el proceso educativo se desarrolla (Gimeno, 2008); por lo tanto, la enseñanza debe contener una fundamentación teórica sólida basada en teorías educativas que posibiliten llevarla a la práctica, con el fin de considerar problemas y variables presentes durante la ejecución (Flores, 1997). Heredia y Sánchez (2013) las definen como “un conjunto de constructos ligados entre sí que observa describe y explica el proceso de aprendizaje de las personas y aquello con lo que se piensa que está relacionado dicho proceso” (pág. 13).

Por lo que es preciso señalar que Schunck (2012), Urías *et al.* (2015), Ríos (2001), entre otros, refieren las siguientes teorías como representativas en esta área de estudio:

- El conductismo se fundamenta en la observación de la conducta y su análisis. En esta corriente, el uso de las TIC se da a través de las máquinas de enseñanza, utilizadas por Skinner (1968), en lo que se denomina tecnificación de la enseñanza o instrucción programada.
- La teoría sociocultural de Lev Semiónovich Vygotski, y en especial, su concepto de zona de aprendizaje próximo el cual “expresa de forma concentrada una visión psicológica del hombre. Puede interpretarse como un sistema donde se identifican el sujeto que aprende, el sistema simbólico que es aprendido y el sujeto que enseña, como elementos en un espacio de relación” (Corral, 2001, pág. 72). En esta teoría, el uso de la tecnología se realiza mediante entornos de aprendizaje apoyados de ella, pueden ser consideradas herramientas psicológicas y no se limitan a ser instrumentos meramente técnicos, de modo que generalmente son

entendidas y usualmente utilizadas, lo que maximiza los beneficios del proceso de enseñanza aprendizaje durante su uso (Valbuena y Quintana, 2014).

- El cognitivismo tiene como idea principal que el aprendizaje ocurre cuando el alumno incorpora información nueva en una memoria a largo plazo, además, la participación del alumno es totalmente activa en el proceso de aprendizaje, por lo cual debe vincular información existente con nueva mediante esquemas mentales, para lograrlo emplea estrategias metacognitivas, las cuales incluyen ordenar y procesar nueva información, además de generar estrategias de aplicación para resolución de problemas. En cuanto a las TIC, el cognitivismo postula mayor efectividad cuando están presentes de forma sincrónica en la comunicación entre docente-alumno y también entre alumnos; asimismo, el uso principal de la tecnología radica en la elaboración de mapas conceptuales y mentales, pues en esta corriente, la adquisición de conocimientos es a través de representaciones y tareas basadas en la repetición (Valdez, 2012).
- El constructivismo plantea la necesidad de brindar al alumno medios requeridos (generación de andamiajes o también llamados puentes cognitivos) que le permitan estructurar procedimientos propios para la resolución de problemáticas, lo cual conlleva una renovación de ideas ante cada situación y le permite aprender continuamente, esto sustentado por los aportes de Jean William Fritz Piaget y Lev Semiónovich Vygotski (Ortiz, 2015). Respecto al uso de las TIC, el

teórico David Junasen, expone tres variaciones, la primera consiste en la alfabetización informática, en donde se aprende acerca de la computadora; la segunda implica el aprendizaje desde el equipo y se le conoce como instrucción programada; en la tercera, la máquina se convierte en un recurso educativo ligado al proceso de enseñanza-aprendizaje, en este caso el alumno adquiere conocimientos con la PC (Hipatia, 2014). Por último, todo esto busca estar basado en los cuatro pilares de la educación propuestos por Delors (1996), los cuales exponen una nueva forma de conceptualizar la educación en contra posición con la manera tradicional, que limita la educación a la adquisición de contenidos y supone adquirir habilidades que sean útiles para toda la vida de los estudiantes; los cuatro principios son:

- *Aprender a conocer.* Implica comprender el mundo que nos rodea, para así lograr desarrollarse a nivel profesional y comunicarse eficazmente con otros.
- *Aprender a aprender.* Se relaciona con mantener activa la mente ejercitando la memoria y el pensamiento, capacitando al estudiante para resolver un sin número de situaciones y permitiéndole desarrollar la capacidad de trabajar en conjunto con otros individuos.
- *Aprender a ser.* Se enfoca en el desarrollo personal del alumnado, construyendo capacidades como autonomía y responsabilidad, de este modo, permite una educación que aprecia las diferentes

capacidades del individuo como el razonamiento, habilidades físicas, sentido estético y artístico, elocuencia, entre otras.

- *Aprender a convivir*. Tiene que ver con el desarrollo de habilidades empáticas para vivir en sociedad, facilitando y abriendo las puertas a una comunicación efectiva, basada en valores como pluralismo, solidaridad y trabajo conjunto para construir la paz.

Así mismo, la tecnología aplicada en la educación no sólo se ha apoyado de las teorías de aprendizaje, sino de varios campos disciplinarios tales como el curricular, la comunicación y los sistemas, entre otros. A pesar de que estas teorías aportan elementos de significancia, es la integración de todos los ámbitos disciplinares lo que conlleva, en mayor frecuencia, al resultado esperado. Por último, las teorías buscan ser aplicadas en el contexto educativo, el cual, tiende a ser dinámico y adaptativo, por lo que se infiere una evolución de ellas a partir de los cambios del entorno (Mujica, 2020).

2.2. Tecnología Educativa

De forma paralela a la evolución de las teorías pedagógicas, el crecimiento tecnológico avanzaba paulatinamente, y con ello su uso en distintos ámbitos, llegando así al educativo; Mujica (2020) menciona que la tecnología se ha incorporado de manera significativa dentro de la educación, debido a las ventajas inmediatas que brinda, dadas por la familiaridad o cercanía que tienen los estudiantes con ella, y por ende, una mejor comprensión y atención a los contenidos de aprendizaje presentados por este medio. Con el paso del tiempo, se

acuñaría el concepto de Tecnología Educativa (TE), definido por Area (2009, pág. 18):

Una forma sistemática de diseñar, desarrollar y evaluar el proceso total de enseñanza-aprendizaje, en términos de objetivos específicos, basada en las investigaciones sobre el mecanismo del aprendizaje y la comunicación que, aplicando una coordinación de recursos humanos, metodológicos, instrumentales y ambientales, conduzca a una educación eficaz.

Del mismo modo, de acuerdo con Cabero (2007) es un concepto que integra distintas ciencias, tecnologías y técnicas; por ejemplo: física, ingeniería, pedagogía, psicología, teoría de sistemas y algunas otras. Además, acentúa que dados los cambios experimentados en su trayectoria y las diferentes áreas que la integran, podemos llamarla una disciplina viva; debido a esto, el concepto de TE posee diversos significados, pues puede ser descrito desde diversas perspectivas y lo califica de contradictorio significativo, por los debates generados alrededor de él. Cabero (2007) establece cinco etapas en la evolución de la TE:

1. Desarrollo primario y esbozos del área.
2. Uso de medios de comunicación masiva y audiovisuales con fines educativos.
3. Incorporación de la psicología conductista en el proceso de enseñanza-aprendizaje.
4. Integración del enfoque sistémico en la educación.

5. Nuevas directrices debido a la introducción de la psicología cognitiva y constructivista, aunado a los nuevos planteamientos epistemológicos en lo educativo.

Asimismo, a partir del avance de la TE, se establece una teoría de aprendizaje que busca aprovechar los componentes tecnológicos de manera integral, la cual es nombrada conectivismo; esta integra principios propios de las teorías de caos, redes, complejidad y autoorganización, además, la adquisición de saberes es un proceso que sitúa al interior de ambientes no definidos y compuestos por elementos cambiantes (Aparici, 2013).

De esta manera, el aprendizaje, entendido como conocimiento aplicable, puede encontrarse externamente, en una organización o base de datos, y su principal objetivo es establecer conexiones entre conjuntos de información, mismas que permiten aprender. En este contexto, el uso de herramientas tecnológicas para apoyar el PE-A se incrementó, como resultado, el desarrollo de software especializado para apoyarlo fue tomando más importancia.

Sin embargo, durante la construcción de este tipo de material utilizando métodos propios de la TE, se encontraron una serie de problemas; uno de los principales fue la omisión o inadecuada integración de procedimientos o técnicas de enseñanza-aprendizaje, dado que, al ser estudiada desde dos campos separados, tecnología y educación, existen distintas posturas en torno a ella, por lo cual no ha sido posible su consagración como disciplina única (Area, 2009).

2.3. Fundamento tecnológico

Es común realizar actividades cotidianas con el apoyo de algún sistema informático, lo que ha beneficiado a diversas áreas como la industria, servicios públicos, finanzas, entretenimiento y, por supuesto, educación (Pineda, 2017); sin embargo, a lo largo de los años, han surgido problemáticas durante el desarrollo de programas digitales, tales como la poca capacidad de los desarrolladores para predecir tiempos, esfuerzo, costos, al igual que para satisfacer los requerimientos del usuario, entre otros; es principalmente por esto que nace la ingeniería de software en calidad de área de estudio (García-Peñalvo, 2018; Ruiz y Aguilera, 2007).

La Ingeniería de Software (IS) se define como la “disciplina que se encarga de la aplicación de procedimientos, métodos, técnicas, tecnologías y herramientas, provenientes de diversos estudios (computación, gestión de proyectos, ingeniería, el entorno y otras ciencias), de manera sistémica, en el proceso de desarrollo y mantenimiento de software” (Espinoza, 2013). En este sentido, la IS es fundamental por dos puntos (Pineda, 2017; Sommerville, 2011):

1. Las situaciones en las que la comunidad se apoya con software complejo y sofisticado se han hecho habituales, y por ello, resulta necesario producirlo de manera confiable, económica y rápida.
2. El uso de procesos, métodos, técnicas y metodologías propias de la IS resulta en productos con menores costos de tiempo, dinero y esfuerzo.

Desarrollo de software

Frecuentemente, el desarrollo de software o ciclo de vida del software se realiza de manera empírica, aunque es una forma cómoda de realizarlo, en la mayoría de los casos el producto es poco efectivo; dado que, si bien se llevan a cabo las etapas esenciales (concepción, desarrollo, prueba y explotación¹) de uno u otro modo, al no tener un orden establecido se divaga entre posibilidades y realidades de la problemática a resolver (Ruiz y Aguilera, 2007); en la elaboración de software educativo sucede cuando no se consideran los objetivos de aprendizaje y, por ende, el resultado no cubre las necesidades del PE-A.

De esta manera, a menudo los requerimientos de software se implementan sin un análisis y diseño previo, denotando una mala toma de decisiones. Una de las problemáticas más comunes, es la falta de vinculación entre los requisitos que se tienen que cubrir, lo cual conlleva a una deficiencia al enlistar y definir las prioridades desde los puntos de vista del cliente y el usuario final, para evitarlo, el desarrollador debe considerar cautelosamente las características y necesidades del cliente y del usuario, estimar y determinar un plazo de tiempo y costos. Otro inconveniente, surge en el momento en que las pruebas del sistema son realizadas, pues se encuentran defectos de diversa índole que obligan a retroceder para corregirlos, la mayoría de ellos provienen de errores de programación, requerimientos mal implementados o comprendidos por el desarrollador; a raíz de estas dificultades, el uso de metodologías es de suma importancia para evitar o minimizar los errores antes mencionados. En este contexto, la selección de una

¹ Instalación y aceptación del software.

metodología o modelo de Ingeniería de Software debe realizarse a conciencia, considerando su adaptabilidad al sistema a desarrollar (Ruiz y Aguilera, 2007).

Asimismo, se recomienda involucrar a los posibles usuarios durante el proceso de desarrollo de software con el fin de tener un producto final que atienda las expectativas y necesidades de los consumidores finales. Por lo que atendiendo la problemática que aborda esta investigación, para el desarrollo de software educativo es esencial tomar en cuenta las destrezas del alumnado, independientemente de la orientación didáctica, además de considerar el estilo y objetivos de aprendizaje, con la finalidad de diseñar una herramienta que les permita una mejor comprensión del tema (López y Rojano, 2007).

Software educativo

En el apartado anterior, se menciona por primera vez al software educativo que de acuerdo con Sánchez (2002), se entiende como cualquier programa informático que presente características en su estructura y funcionalidad que apoyen el acto educativo. En otras palabras, una definición más específica, es aquel material didáctico desarrollado en una computadora con el propósito de ser utilizado durante el proceso enseñanza aprendizaje (Brito *et al.*, 2009; Sánchez, 1993).

El software educativo puede ser empleado en diversas áreas de conocimiento, y de formas variadas, desde cuestionarios hasta simulaciones, brindando distintos ambientes de aprendizaje adaptables a las particularidades de los alumnos, permitiéndoles interactuar a niveles variables; sin embargo, todos poseen las siguientes características fundamentales (Candelario-Dorta, 2018):

1. Poseen un fin didáctico
2. Tienen como soporte una computadora
3. Permiten un intercambio de información entre el alumno y el contenido educativo; es decir son interactivos
4. Son adaptables a las características particulares de cada alumno
5. Pueden utilizarlo con conocimientos básicos, es decir, su uso es sencillo para los estudiantes

Elementos del software educativo

Un número relevante de software didáctico, al igual que la mayoría de los sistemas concebidos sin naturaleza educativa, están compuestos de tres módulos fundamentales que son marcadamente definidos (Brito *et al.*, 2009):

- El primero es un módulo encargado de administrar la comunicación con el cliente, entrada y salida de datos, interfaces, entre otros.
- El segundo incluye de manera organizada el contenido informático del software, es lo que se realiza en el diseño de datos.
- El tercero que administra las acciones de la computadora y las reacciones a las operaciones hechas por los usuarios, en otras palabras, el motor.

Tipos de software educativo

Es relevante señalar que existen diversas clasificaciones de software educativo, Sánchez (1993) menciona que la mayoría de ella coincide en cuatro tipos principales:

- De ejercitación, se caracteriza por presentar ejercicios a resolver por el alumno.
- Tutorial, presenta contenidos educativos y evalúa conocimientos adquiridos respecto a la información presentada.
- Simulación, el cual imita un fenómeno o el funcionamiento de un objeto mostrando al alumno un proceso de forma detallada.
- Juegos instruccionales, que combinan la estructura de una simulación con el desarrollo de un juego interactivo.

En este sentido, se comienzan a vislumbrar las diferencias particulares entre el software convencional y el educativo, así como, las deficiencias que presentan las metodologías de desarrollo de software tradicionales, por lo cual, se buscaron alternativas para solucionar los inconvenientes mencionados.

Metodologías para el desarrollo de software educativo

Utilizar métodos y metodologías propios para la construcción de sistemas informáticos, los dotan de una naturaleza ingenieril, puesto que fomentan que las tareas realizadas sean sistemáticas y repetibles con el objetivo de satisfacer los requerimientos del usuario y de esta manera lograr los objetivos planteados; así mismo, este uso provee una perspectiva y herramientas particulares en la afrontación, notación y procedimientos para resolver el problema (Pineda, 2017). Liviu (2014) define a una metodología para la elaboración de software como una serie de reglas y pautas utilizadas durante la investigación, planeación, diseño, desarrollo, pruebas, configuración y mantenimiento de un software. Esta también

incluye valores fundamentales útiles para el equipo desarrollador y herramientas usadas a lo largo del proceso mencionado.

En este sentido, la definición anterior muestra que todas las actividades están orientadas a satisfacer aspectos técnicos del resultado, lo cual, mayormente funciona cuando se automatiza un proceso donde las variables generalmente son estáticas; sin embargo, en el momento en el que los procesos tienden a ser cambiantes, se producen factores que dificultan cumplir con el objetivo, como sucede en la educación (Abud, 2009). Por esta razón, durante el transcurso de los años, se han desarrollado una serie de modelos o metodologías buscando cumplir lo mencionado y así lograr los objetivos de aprendizaje. A continuación, se presentan ocho de ellas:

- Ingeniería de software educativo planteada por Galvis (1992) es una metodología híbrida que tiene como base las metodologías de desarrollo de software mediante el paradigma orientado a objetos y la incorporación de estrategias educativas. El autor menciona que va dirigida a educadores e informáticos, haciendo una separación de las partes involucradas. Considera las fases:
 1. Análisis
 2. Diseño
 3. Desarrollo
 4. Pruebas piloto y
 5. Pruebas de campo (durante todo el desarrollo)

- Modelo de análisis, diseño instruccional, tecnológico y evaluación (ADITE) formulado por Polo (2003) es un modelo de diseño instruccional que pretende adaptar los canales tecnológicos de transmisión de la información durante el proceso enseñanza-aprendizaje. Este modelo surge desde la perspectiva constructivista y el proceso de desarrollo se realiza de forma iterativa, donde sus fases se superponen, la cuales son:
 1. Análisis
 2. Diseño instruccional
 3. Diseño tecnológico
 4. Evaluación
- Metodología Orientada a Objetos para Desarrollar Software Multimedia e Hipermedia (MOOMH) desarrollada por Benigni (2004) nace a partir de la combinación de varias metodologías orientadas a objetos y de desarrollo de software multimedia. Su objetivo es crear sistemas hipermedia y enfatizar en la relevancia que tienen las interfaces. Esta metodología está conformada por cuatro modelos:
 1. Modelo de requerimientos
 2. Modelo de análisis
 3. Modelo de diseño
 4. Modelo de implementación
- Metodología para el desarrollo de software multimedia educativo (MEDESME) se centra en describir los componentes y tareas necesarias para crear el ambiente de manera general, pedagógica y técnica o estética (García *et al.*, 2016). Los autores utilizan ocho fases, además, remarcan que

dentro de ellas se tienen que cumplir cada uno de los pasos que las componen. A continuación, se elabora una lista de las fases:

1. Concepto o pre-producción
 - a. Concepción del proyecto
 - b. Agentes implicados
 - c. Viabilidad
 - d. Justificación
2. Análisis
 - a. Objetivos y metas de la aplicación
 - b. Identificación de los elementos, procesos y actividades
 - c. Estrategias de aprendizaje
 - d. Tareas de aprendizaje
 - e. Perfil de usuario
 - f. Entorno de aprendizaje
 - g. Análisis del contenido
 - h. Requerimientos técnicos del entorno de aprendizaje
 - i. Requerimientos de usuario
 - j. Requerimientos funcionales
3. Diseño
 - a. Ficha general
 - b. Ficha técnica psicopedagógica
 - c. Ficha educativa-técnica
 - d. El esquema y mapa de navegación; storyboard; diagrama de análisis de tareas, de casos de uso y de entidad-relación

- e. Edición de plantillas
 - f. Edición de elementos de control
 4. Desarrollo
 - a. Selección de las herramientas de desarrollo
 - b. Manual técnico
 - c. Edición de los elementos multimedia
 5. Implementación
 6. Evaluación y validación del programa
 - a. Prueba piloto
 - b. Evaluación de contenido
 - c. Evaluación del entorno general, pedagógico y técnico/estético
 - d. Evaluación de conocimientos o aprendizajes esperados
 7. Producción
 8. Elaboración del material complementario
- Metodología de Thales creada por Madueño citado por Dueñas *et al.* (2017) es el resultado de la combinación de estrategias tecnológicas y educativas. Las seis fases se implementan de manera cíclica, con base en el nivel de complejidad que suponga la separación de actividades; además, se realizan revisiones continuas durante el desarrollo y producto en versiones no terminadas, en relación con los resultados que se consiguen. Las etapas son las siguientes:
 1. Planeación
 2. Diseño
 3. Producción

4. Prueba piloto
 5. Evaluación
 6. Mejoramiento
- Metodología Dinámica para el Desarrollo de Software Educativo, propuesta por Arias *et al.* (2015) la cual se concibe de entender a la computadora en calidad de instrumento dinámico. Así mismo, las posibilidades multimedia son conceptualizadas como ventajas y se toman en consideración durante la producción del diseño instruccional (el cual es base de los sistemas que apoyan al proceso educativo) a partir de la primera fase. Esta metodología tiene cuatro etapas:
 1. Diseño Educativo
 2. Producción
 3. Realización
 4. Implementación

Se debe resaltar que la evaluación es una fase transversal que se considera durante las anteriores. Por último, se trabaja como en una metodología ágil, no es requisito terminar una etapa para continuar con la siguiente, se pueden generar prototipos para realizar validaciones o correcciones a partir de los comentarios de estos.

- Desarrollo de Software Educativo: una propuesta metodológica, en la que Quintero *et al.* (2005) plantean que, durante la revisión de las metodologías, se vislumbra la separación de las áreas de estudio involucradas, resultando en que la elaboración de sistemas se convierta en trayectos extensos y difíciles, en consecuencia, genera elevados costos de producción. En este

sentido, los autores buscan establecer una sinergia entre las teorías educativas y las etapas del desarrollo de software, a través de una materialización gráfica, en la cual se reconocen todos los elementos del proyecto, con el fin de vislumbrar la individualidad de cada componente y su integración como conjunto. De esta manera, los desarrolladores tienen la oportunidad de operacionalizar determinada fase verificando la sucesión de modo integrado. En este marco, se establecen los siguientes puntos:

1. Estudio Preliminar
 2. Diseño Instruccional
 3. Desarrollo de guiones
 4. Fase de construcción
 5. Validación
- Modelo para el desarrollo de software educativo basado en competencias (MODESEC), creado por EDUPMEDIA (Caro y Toscano, 2009), se conceptualiza a partir de racionalizar la experiencia de profesores y jóvenes investigadores que han obtenido a lo largo de tres años durante la elaboración de software ajustado al entorno educativo y fomentando las competencias. Se plantean cinco etapas basadas en este sistema:
 1. Diseño educativo
 2. Diseño multimedial
 3. Diseño computacional
 4. Producción
 5. Aplicación

No obstante, a pesar de estos esfuerzos encaminados a obtener el objetivo de aprendizaje, los resultados no han sido los esperados, García *et al.* (2002) establecen que pese a “[...] la importancia cada vez mayor de los computadores y el *courseware* en la educación, siguen existiendo serios problemas en el desarrollo de software educativo” (pág. 1). Los autores lo atribuyen a un vacío teórico y metodológico del área, asimismo a la compleja incorporación de los aspectos didácticos a los tecnológicos y viceversa, resultando en errores de integración de estrategias, de conceptualización de los productos deseados, de comunicación entre los integrantes, por mencionar algunos. Del mismo modo, hoy en día se mantienen estas problemáticas, por lo que, esta afirmación continua vigente.

2.4. Análisis comparativo de metodologías de software educativo

A partir de la descripción realizada de las metodologías y métodos, se presenta un análisis de estas, las cuales fueron seleccionadas por ser recurrentes en artículos especializados del área (Quintero *et al.*, 2005; García, 2016; Caro *et al.*, 2009; Serna *et al.*, 2012; Tafur *et al.*, 2016; Dueñas *et al.*, 2017; Marcano y Benigni, 2014; entre otros) y representan una ventaja para fortalecer aspectos de la investigación.

En este sentido, se determinan dos criterios para el análisis mediante el método comparativo, el primero es el tecnológico, en el cual se considera al desarrollo de software o ciclo de vida de software, pero no se debe confundir con una metodología, el ciclo es algo que sólo sucede desde su concepción hasta su

obsolescencia (Whitten y Bentley, 2008) y que se recomienda contemplar en las metodologías. Después de una revisión bibliográfica (Sommerville, 2011; Kendall y Kendall, 2011; Braude y Bernstein, 2016; Whitten y Bentley, 2008; entre otros), se consideran los elementos más relevantes y se establecen cinco fases, las cuales se describen a partir de la propuesta de Kendall y Kendall (2011):

- **Análisis:** se indaga sobre las necesidades que el sistema debe cubrir durante el proceso a automatizar, se genera a partir de procedimientos, herramientas y técnicas específicas que sirven como apoyo a establecer estos requisitos. En esta fase, se determina a una persona que toma decisiones estructuradas y analiza su ejecución. Por último, se realiza un anteproyecto de lo que desarrollará, en el que se resumen lo que se descubrió acerca de los usuarios, la capacidad de uso y conveniencia de los sistemas existentes; además, añade una estimación del costo-beneficio de las posibilidades junto con recomendaciones personales.
- **Diseño:** se usa lo realizado en la etapa anterior para generar un diseño lógico del software. En esta fase se diseñan los métodos que apoyaran a los usuarios durante su interacción con el sistema buscando ser lo más exacta posible; así mismo, se elabora el diseño de la base de datos que guardará la información indispensable requerida. Finalmente, se esquematizan los controles y procesos que salvaguarden al sistema y a los datos contenidos, y con ello, crear paquetes de especificación del software para los programadores.
- **Desarrollo:** con base en el diseño hecho, los programadores buscan realizar el sistema con los requerimientos que inicialmente se solicitaron. En esta

etapa, los codificadores fungen como protagonistas, puesto que no sólo escriben un código, sino que la mayoría de las veces arreglan fallos sintácticos del software que se suscitaron durante las fases anteriores, además de apoyar indirectamente a la siguiente.

- Pruebas: previamente a la entrega, el sistema se debe poner a prueba, experimentando con su funcionamiento puesto que resulta mucho más barato identificar errores en este punto. La primera parte de esta fase se hace por lo programadores durante el desarrollo, la siguiente con especialistas, el equipo de trabajo y una muestra de los usuarios finales. Inicialmente, se realizan varias pruebas con datos y situaciones ficticias, posteriormente con información real con la que el sistema trabajará.
- Mantenimiento: al finalizar las fases anteriores, se establece una rutina para la conservación del sistema creado durante el tiempo que esté en funcionamiento. Aquí los programadores realizan labores rutinarias en el software, las cuales funcionan como actualizaciones que buscan optimizar el funcionamiento, corregir errores espontáneos, entre otras. El trabajo de la fase del análisis debe determinar un mantenimiento mínimo necesario del software.

El segundo, el educativo, se establece al PE-A de la misma forma que en el primer criterio, se determina a partir de una investigación teórica del concepto (Carballo, 1978; Medina y Mata, 2009; Yáñez, 2016; entre otros), de manera que se definen tres fases y son precisadas con base en lo mencionado por Carballo (1978):

- **Planeación:** consta en anticipar los diferentes elementos que se tomarán en cuenta durante el proceso educativo y elegir de manera anticipada los medios didácticos que se emplearán para alcanzar los objetivos de aprendizaje. No es un proceso estático, debido a su naturaleza activa se deben contemplar tres componentes elementales:
 - La construcción debe realizarse desde el punto de vista de a quiénes estará destinada la enseñanza, debido a que a lo largo del desarrollo se hacen evidentes las diferencias singulares de los estudiantes.
 - El proceso educativo no es algo que se produce de modo independiente, sino que está entrelazado con el entorno social y cultural del individuo.
 - La forma en la que aprenden las personas generalmente no es de interés, puesto que se da de manera inherente; sin embargo, en cuestiones educativas se tiene que considerar cómo se está haciendo, por lo cual se debe de apoyar de las teorías educativas existentes.
- **Ejecución:** el docente conduce las actividades y medios que son realizados y utilizados por los estudiantes.
- **Evaluación:** se genera de manera sistemática, constante e integral para conocer el nivel de consolidación de los objetivos de aprendizaje.

Cabe señalar, que en la literatura se pueden o no encontrar los nombres de las fases exactamente que se establecen en esta investigación, pero para fines prácticos y comenzar a homologar términos, se realiza una aproximación y determinan los mencionados, los cuales se muestran en la Tabla 1; en esta se representan mediante una **x** los elementos presentes en la propuesta de los autores, con un **–** los que no

son visibles y con un **no se especifica** si los puntos no son explícitos y pueden estar inferidos.

Tabla 1. Análisis de elementos del proceso enseñanza-aprendizaje y desarrollo de software entre metodologías

Metodología/ Modelo	Proceso para el desarrollo de software					Proceso enseñanza-aprendizaje		
	Análisis	Diseño	Implementación	Pruebas	Mantenimiento	Planeación	Ejecución	Evaluación
Ingeniería de Software Educativo (Galvis, 1992)	x	x	x	x	-	x	No se especifica	x
ADITE (Polo, 2003)	-	x	-	-	-	x	No se especifica	x
MOOMH/OOHDM (Benigni, 2004)	x	x	x	-	-	x	-	x
MEDESME (García <i>et al.</i> , 2016)	x	x	x	x	-	x	-	x
THALES (Madueño citado por Dueñas, 2017)	x	x	x	x	x	x	-	x
MODESEC (Caro y Toscano, 2009)	x	x	x	x	-	x	-	-
Metodología dinámica para el desarrollo de software educativo (Arias <i>et al.</i> , 2015)	x	No se especifica	x	-	-	x	No se especifica	x
Desarrollo de software educativo: una propuesta metodológica (Quintero <i>et al.</i> , 2005)	x	x	x	x	-	x	-	-

Fuente: Elaboración propia.

Por último, se presentan los entregables a manera de parámetros indicativos, se debe mencionar que se busca solucionar la problemática que constituye la homologación de términos mediante el establecimiento de estos durante las etapas

de la metodología propuesta. La determinación de estos productos se realizó con base en lo recurrente de la literatura:

- Especificación de requerimientos: es el proceso en el que se listan los requerimientos del usuario y software; de manera ideal, deben ser claros, no ambiguos, fáciles de entender, completos y consistentes. Se tienen que describir los requisitos funcionales (qué va a hacer el sistema) y no funcionales (cómo lo va a realizar) de modo que sean comprensibles para los usuarios finales, quienes no poseen un conocimiento técnico detallado (Sommerville, 2011).
- Modelado del sistema: se desarrollan modelos abstractos y cada uno representa una perspectiva del software. Por lo general, se asocia a algún tipo de notación gráfica desarrollada en cualquier estándar como lo es el Lenguaje de Modelado Unificado (Sommerville, 2011).
- Sistema o aplicación: es un conjunto de instrucciones que le dice a la computadora qué realizar. Comprende grupos completos de programas, procedimientos y rutinas asociadas con la operación del software (Britannica, 2020).
- Informes: son escritos sobre lo sucedido de una situación, proceso o prueba; en estos se describen cómo fueron indagados o comprobados, su significado, las conclusiones a las que se llegan al finalizar el proyecto y, en algunos casos, las recomendaciones que se quieran proponer (Sharma y Mohan, 2004).
- Actualizaciones y adaptabilidad: En primera instancia, las actualizaciones proporcionan modificaciones al software existente; de tal modo que se

cambian características y funciones de la aplicación, se reparan errores conocidos y se modifica la interfaz del usuario (Vanier y Rashidi, 2016). Por otro lado, la adaptabilidad es el grado en el que un sistema se ajusta a las alteraciones en su entorno; en principio, debe tolerar cambios en su ambiente sin intervención externa; sin embargo, en caso de que no, se apoyará de las actualizaciones (Subramanian y Chung, 2001).

- **Diseño instruccional:** consiste en la resolución de problemas instruccionales mediante un análisis sistemático de las condiciones de aprendizaje y un diseño que satisfaga a la experiencia de este. Por lo tanto, se refiere a la transformación sistemática de principios de enseñanza y aprendizaje en planes o especificaciones para actividades o material didáctico; esta planificación está basada en teorías de aprendizaje (Smith y Ragan, 1999; Molenda *et al.*, 2003).
- **Secuencia didáctica:** es un grupo de ejercicios didácticos definido en un orden específico que considera el progreso del estudiante. El proceso empieza con una producción inicial (inicio), es seguida por una serie de actividades (desarrollo) y termina con una producción final (cierre) (Soler *et al.*, 2013).
- **Instrumentos de Evaluación:** son recursos estructurados diseñados para fines específicos, plasman una serie de hechos y capacidades que se esperan haber logrado al finalizar una intervención. Así mismo, especifican los valores que serán utilizados en la evaluación, acorde con lo que se requiera ponderar, es posible utilizar diferentes tipos de medios (Dirección General de Desarrollo Curricular, 2012; Peersman, 2014).

Hay que enfatizar, que a partir de la Tabla 1 se visualizan fases que no necesariamente terminan en estos productos, en el sentido de que cada autor de la metodología o modelo respectivo no lo especifica de manera abierta o simplemente no lo toma en consideración, por lo cual se hizo un análisis de los productos de las metodologías que se muestran en la Tabla 2, el significado de la simbología, al igual que en la Tabla 1, es una **x** para los elementos presentes en la propuesta de los autores, con un **-** los que no son visibles y con un **no se especifica** si los puntos no son explícitos y pueden estar inferidos.

Tabla 2. Análisis de los productos obtenidos de las metodologías investigadas

Metodología/Modelo	Proceso para el desarrollo de software					Proceso enseñanza-aprendizaje		
	Especificación de requerimientos	Diseño	Sistema o aplicación	Informes	Actualizaciones y adaptabilidad	Diseño instruccional	Secuencia didáctica	Instrumentos de Evaluación
Ingeniería de Software Educativo (Galvis, 1992)	x	x	x	x	-	No se especifica	-	x
ADITE (Polo, 2003)	-	x	-	-	-	x	-	x
MOOMH/OOHDM (Benigni, 2004)	x	x	x	-	-	No se especifica	-	x
MEDESME (García <i>et al.</i> , 2016)	x	x	x	x	-	x	-	x
THALES (Madueño citado por Dueñas, 2017)	x	x	x	x	x	No se especifica	-	x
MODESEC (Caro y Toscano, 2009)	x	x	x	x	-	x	-	-
Metodología dinámica para el desarrollo de software educativo (Arias <i>et al.</i> , 2015)	x	No se especifica	x	-	-	x	-	x
Desarrollo de software educativo: una propuesta metodológica (Quintero <i>et al.</i> , 2005)	x	x	x	x	-	x	-	-

Fuente: Elaboración propia.

De esta manera, el análisis es presentado desde el punto de vista de un equipo de trabajo multidisciplinario, por lo que es altamente recomendable que el procedimiento sea explícito en cuanto a lo que se requiere, algunas de las veces esto puede ser inferido, pero al ser conceptos e integrantes con perspectivas diferentes, resulta insuficiente, puesto que se realizan interpretaciones dispares y, por ende, el grupo desarrollador entra en conflicto. También, se observan los alcances de las diversas metodologías, y con ello, las áreas de oportunidad en las que hará injerencia este proyecto.

Finalmente, los aspectos teóricos presentados en esta sección buscan establecer en el lector la terminología y contexto necesario para entender esta investigación; asimismo, se resalta la importancia y conexión entre las variables, en el siguiente capítulo se muestra el procedimiento y elementos que se necesitaron para llevar a cabo el desarrollo y propuesta de metodología de desarrollo de software educativo.

3. METODOLOGÍA

Esta tesis sigue una metodología basada en el tipo de investigación aplicada, Martínez (2019) establece que su alcance no se limita a documentar un hecho en términos de expansión del conocimiento, sino que busca incidir en la realidad, problematizando, analizando y generando una propuesta con el fin de modificar la situación presentada.

Primeramente, mediante el estudio del estado del arte, se establece la evolución de la tecnología y educación, así como el problema de conjunción de estas para establecer a la TE; además, se muestra la ambigüedad en la terminología utilizada hasta el momento. Después, la construcción de software educativo sin fundamentación pedagógica puesto que se ha identificado la dificultad de replicar o eliminar factores tanto funcionales como no funcionales en otros proyectos educativos.

En segunda estancia, la comparación y análisis de ocho metodologías existentes se realizó mediante el método comparativo definido “el procedimiento de comparación sistemática de objetos de estudio que, por lo general, es aplicado para llegar a generalizaciones empíricas y a la comprobación de hipótesis” (Nohlen, 2020, pág. 47); por lo que, mediante una investigación documental se identifican y establecen las fases que componen al proceso enseñanza-aprendizaje y ciclo de vida de desarrollo de software pedagógicos y tecnológicos, de igual forma, los productos o entregables de estas etapas, así como su importancia e impacto dentro su respectivo procedimiento. Posteriormente, se hizo una búsqueda de metodologías que impactan en un contexto socio-temporal similar al que permea a este trabajo, para después comparar las fases y productos entre estas.

Con base en lo anterior, se establecen los elementos pedagógicos y tecnológicos que se usaron para la propuesta tales como diseño instruccional, listado de requerimientos, planeación didáctica, UML (Lenguaje de Modelado Unificado), validación de los expertos de ambas áreas, entre otros fundamentados en teorías educativas o estándares del desarrollo de software; y con ello, la conceptualización de Metodología Conjunta para Aplicaciones Educativas (MCAE); la cual se compone de cinco fases establecidas a partir del ciclo de vida de desarrollo de software, se definen a partir de tres componentes: (1) descripción, que precisa la justificación, características y panorama general de la etapa; (2) procedimiento, donde se indica cómo funciona y cómo se implementa; (3) productos, los cuales son los resultados o entregables (algunos de ellos mencionados en el inicio del párrafo), buscando un control, seguimiento y evidencia de lo desarrollado. Estos aspectos buscan definir el contexto de manera

detallada, para un entendimiento óptimo del equipo de trabajo, y con esto, que no se infieran aspectos que puedan afectar al objetivo final del sistema educativo.

Como paso final, MCAE se somete a una validación conformada de tres puntos: (1) validación por juicio de expertos conformada por 15 especialistas en Tecnología Educativa y con experiencia en equipos de desarrollo de software educativo, sin importar su rol. Al finalizar la revisión, emitieron su opinión mediante un cuestionario validado con alfa de Cronbach; (2) validación por equipos de trabajo conformada por tres grupos, estos elaboraron software educativo mediante MCAE. Se proporcionó acompañamiento antes (explicando la metodología a grandes rasgos), durante (solventado dudas y realizando modificaciones en MCAE a partir de los comentarios) y después (recopilando sus impresiones a través de una herramienta que proporcione el nivel de usabilidad); (3) uso de MCAE para la creación de un REA, con esto se buscó mostrar un ejemplo de cómo se aplica la propuesta, asimismo, que fungiera como guía para los equipos que hagan uso de ella.

En los capítulos anteriores se mostraron los elementos que fundamentan la base teórica de esta investigación; en las siguientes secciones se plantean y justifican los elementos integrados del proyecto, de igual forma, se muestran y analizan los resultados.

4. PROPUESTA

En esta sección, se presenta la conceptualización de la metodología considerando la fundamentación teórica expuesta y las recomendaciones establecidas por expertos. Asimismo, la propuesta está orientada a un grupo de trabajo con roles y responsabilidades claramente definidas, en la Tabla 3 se define esto e información adicional con la que se cubrirán los puestos recomendados del equipo desarrollador.

Tabla 3. Equipo de trabajo para el desarrollo del proyecto

PUESTO	FORMACIÓN ACADÉMICA MÍNIMA	PERFIL	ROLES	RESPONSABILIDADES
Líder de proyecto	Licenciatura en administración o afines.	Experto en administración y gestión de proyectos	Dirección de proyectos	Dirección de proyectos y coordinación de estrategias para el cumplimiento del proyecto. Facilitador.
Experto educativo	Licenciatura en pedagogía o afines.	Especialista en el proceso enseñanza-aprendizaje	Analista educativo	Aplicación de entrevistas al cliente para la obtención y definición de todos los requerimientos educativos para su implementación en un software. Supervisión y validación de las decisiones de implementación buscando cumplir los requerimientos educativos mediante acuerdos con el experto tecnológico.
Experto disciplinar²	Licenciatura (recomendable).	Especialista en definición de la temática	Asesor disciplinar	Creación y definición de las temáticas de una asignatura específica. Apoyar al experto educativo.
Experto tecnológico	Ingeniería en computación o afines.	Especialista en el desarrollo de software	Analista tecnológico	Aplicación de entrevistas al cliente para la obtención y definición de todos los requerimientos tecnológicos para su implementación en un software. Supervisión y validación de las decisiones de implementación buscando cumplir los requerimientos tecnológicos mediante acuerdos con el experto educativo.
Diseñador	Ingeniería en computación o afines.	Especialista en conceptualización de sistemas.	Diseñador	Elaboración de mockups o bosquejos de interfaz y diagramas mediante el estándar UML.
Programador	Ingeniería en computación o afines.	Especialista en codificación de software	Implementador	Desarrollo de software mediante algún lenguaje de programación, determinado por el proyecto.
Auditor (tester)	Ingeniería en computación o afines.	Amplios conocimientos de instalaciones de media, alta tensión y subestaciones eléctricas.	Pruebas	Revisión de software para la detección de fallos, asimismo que cumpla con lo definido con los requerimientos mediante las técnicas de caja blanca y caja negra.

Fuente: Elaboración propia.

Sin embargo, si las condiciones no permiten contar con todos los integrantes del equipo de trabajo propuesto, el desarrollo se tendrá que realizar con al menos un especialista en educación y uno en tecnología, enfatizando que la interacción entre ambos debe ser estrecha durante todo el proceso de desarrollo del software, con el

² Elemento no activo del proceso de desarrollo, funge como apoyo.

fin de homologar conceptualizaciones de tecnología en educación y viceversa; no obstante, la jerarquía y puestos (Figura 1) deberán ser cubiertos y definidos en su totalidad desde el inicio del proyecto y de común acuerdo por el recurso humano disponible. Finalmente, por lo regular, el experto educativo será capaz establecer el contenido temático; sin embargo, cuando su perfil y sus aptitudes no se adecuen para definirlo, será necesario apoyarse de un experto disciplinar con el fin de que los contenidos temáticos, así como las actividades, materiales, etc., sean los adecuados para cumplir el objetivo de aprendizaje.



Figura 1. Organigrama propuesto para el desarrollo del proyecto. Fuente: Elaboración propia.

Después de haber definido los recursos humanos, es necesario establecer el flujo de trabajo durante el desarrollo de software educativo; por lo que en los siguientes apartados se desglosa de manera detallada la metodología, técnicas e instrumentos a utilizar en el proyecto.

4.1. Metodología Conjunta para Aplicaciones Educativas (MCAE)

La propuesta es una combinación del ciclo de vida de desarrollo de software, el Diseño Conjunto de Aplicaciones (JAD o *Joint Application Design*) y elementos pedagógicos como lo son: diseño instruccional, secuencia didáctica, teorías educativas, entre otros. De esta manera, se establecen cinco etapas y un eje que permeará a cada una de estas, las fases son:

1. Análisis tecnopedagógico: es la etapa fundamental en la que se cimenta toda la metodología, en esta se obtendrán, procesarán y plantearán las bases pedagógicas y tecnológicas del software a desarrollar, así como la planificación del proceso, que incluya las herramientas y recursos a utilizar.
2. Diseño tecnopedagógico: con base en lo obtenido en la fase anterior, se desarrollarán apoyos visuales que faciliten a los expertos de cada área comprender y proponer soluciones óptimas que se ajusten a las necesidades.
3. Implementación: se codifica la solución mediante herramientas de desarrollo de software.
4. Pruebas: se verifica el correcto funcionamiento del software o sistema, así como que se cumplan con las características solicitadas; lo anterior debe ser realizado en primera instancia por el programador, miembros del equipo y expertos; en segundo lugar, por los usuarios finales.
5. Mantenimiento funcional y educativo: se deben programar revisiones periódicas para establecer y corregir fallos, optimizar la aplicación después

de la entrega del producto; así como, adaptabilidad hacia las necesidades de los usuarios y escalabilidad, si es necesaria una actualización relevante.

Finalmente, se aplica el JAD durante todas las fases descritas, mediante reuniones entre las áreas pedagógica y la tecnológica, en las cuales se establecerán acuerdos de desarrollo buscando homologaciones para la implementación. Para las reuniones se deben cubrir tres aspectos: (1) preparación que dará como resultado el orden del día, (2) la sesión de entendimiento con mapas de conocimiento³ como producto y (3) una revisión de los puntos establecidos en minutas (Carmel *et al.*, 1994; Balda y Vicenzi, 1997).

Por otro lado, se hará uso de la ventaja que representa que un experto en educación sea parte del equipo de trabajo, puesto que la disponibilidad facilitará los acuerdos rápidos, los cuales son similares a los acuerdos de desarrollo, con la diferencia de que se realizarán al momento de la duda y se agregaran o se discutirán posteriormente en las sesiones formales. A su vez, las fases de MCAE tiene un orden secuencial, cada etapa no puede empezar sin haber culminado su antecesora; no obstante, es posible regresar si se llega a identificar alguna omisión o falla durante la realización del ciclo actual, de tal modo, que se genera un flujo cíclico entre fases contiguas; así mismo, durante la implementación del mantenimiento funcional y educativo, es necesario retornar a la primera etapa,

³ “... los mapas de conocimiento señalan el camino hacia las fuentes del conocimiento y la información y estructuran el panorama de conocimiento con la representación de elementos e interrelaciones de los dominios del conocimiento. Al igual que los mapas geográficos, los mapas de conocimiento no tratan de capturar todos los aspectos del conocimiento, pero si simplifican y se concentran en algunos aspectos de este.” (El Assafiri *et al.*, 2017)

esto con el fin de revisar y replantear los defectos de ejecución y optimización desde la concepción del sistema, con el objetivo de alcanzar un nivel superior en la calidad del producto, debido a que el grupo desarrollador se ve obligado a repetir el proceso de principio a fin. Lo anterior está representado en la Figura 2.

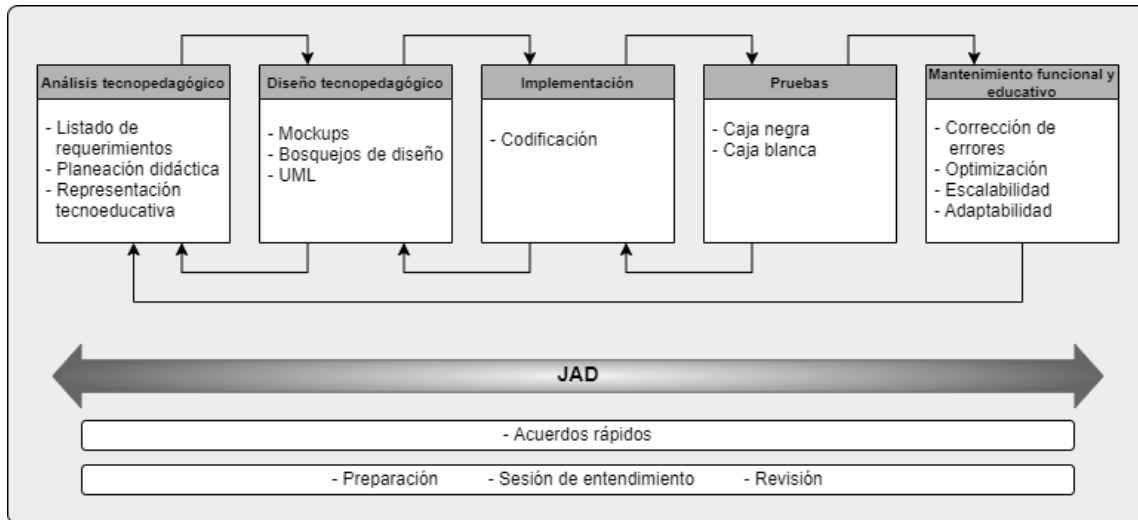


Figura 2. Esquematización de la Metodología Conjunta para Aplicaciones Educativas.
Fuente: Elaboración propia.

Por otro lado, las cargas de trabajo se definieron mediante los roles y responsabilidades, no obstante, resulta conveniente mostrar en qué parte de la metodología se recomiendan hacer uso de las capacidades de cada miembro del equipo de desarrollo; en la Figura 3 se representa lo descrito.

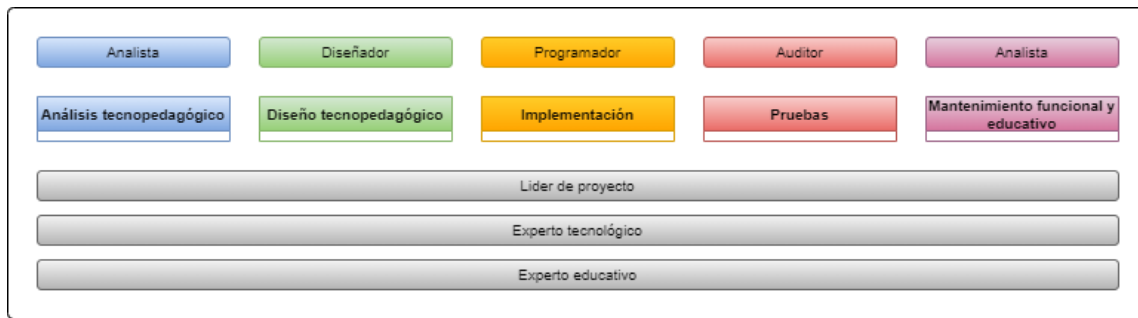


Figura 3. Cargas de trabajo para cada integrante en las diferentes etapas de MCAE. Fuente: Elaboración propia.

A continuación, se describe a detalle cada una de las fases de MCAE, asimismo, se establecerá el fundamento tecnológico y pedagógico utilizado para la concepción de cada elemento considerado.

Fase 1: Análisis tecnopedagógico

Descripción

El diseño instruccional y la ingeniería de software son evidentemente diferentes, no obstante, la literatura pedagógica está redescubriendo conceptos ya tratados en partes de la disciplina computacional, por ende, terminan relacionándose de algún modo (Adnan y Ritzhaupt, 2018).

En concreto, la creación de ambientes de enseñanza-aprendizaje tiene bastante en común con el ciclo de vida del desarrollo de software; por ejemplo, ADDIE, un modelo genérico de diseño instruccional, engloba cinco actividades principales: análisis, diseño, desarrollo, implementación y evaluación; mientras que el modelo de cascada, habitual en la industria de software, está compuesto por:

análisis de requerimientos, diseño, desarrollo, implementación, verificación y mantenimiento, sobresaliendo las concordancias entre ambos (Gustafson y Branch 2002; Willis, 2008; Adnan y Ritzhaupt, 2018).

En este sentido, es posible inferir varios puntos donde las homologaciones de conceptos son viables o útiles durante el desarrollo del software educativo. En esta investigación se plantea una estrecha similitud entre las etapas de análisis de la ingeniería de requerimientos y del diseño instruccional, dado que en ambas busca obtener los requisitos a tomar en cuenta en la solución a crear, así como establecer y entender el contexto donde será aplicada.

De este modo, se producen herramientas que desempeñan papeles fundamentales dentro de sus respectivos procesos: listado de requerimientos, para definir los aspectos tecnológicos, y planeación didáctica, con lo que se determinara el contexto educativo; sin embargo, ambos instrumentos definen particularidades que aún son complicadas de integrar durante la elaboración de software educativo, por lo que se propone usar estos elementos de manera en que se integren las áreas involucradas y genere una definición tecnopedagógica, ilustrado en la Figura 4.

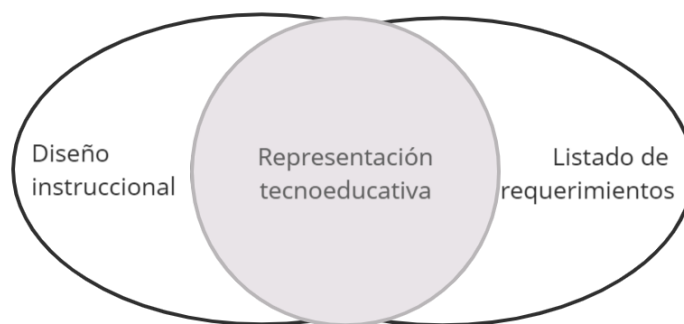


Figura 4. Conceptualización de la definición tecnopedagógica. Fuente: Elaboración propia.

Procedimiento

El PE-A es una actividad que, usualmente, resulta compleja (esto deducido de lo presentado en las secciones anteriores); no obstante, si se logran definir las variables que engloban al proceso de manera idónea, los objetivos son alcanzados con resultados óptimos. Por lo que la obtención de información para comprender el contexto educativo se torna crucial, puesto que el resultado es el fundamento para toda la planeación del cómo se ejecutará la enseñanza y el aprendizaje; en consecuencia, la formulación de un instrumento de recopilación debe ser minuciosa para obtener un escenario pedagógico claro y conciso que permita establecer las bases del desarrollo tecnológico o de la creación de recursos informáticos. Por consiguiente, se parte del supuesto que el desarrollo tecnológico pertenece al contexto educativo, de modo que en esta fase se ofrecen técnicas y herramientas para realizar lo mencionado.

En primera instancia, se establecen las entrevistas debido a que es una de las técnicas más completas para obtener información; en este sentido, se propone un guion que contiene preguntas que buscan fungir como punto de partida durante estas; sin embargo, la experiencia del entrevistador será un factor fundamental, puesto que establecerá la ruta a seguir, la complejidad y el vocabulario a usar con base en el entrevistado, la indagación y profundidad en los detalles, entre otros. Asimismo, la categorización no busca instaurar alguna separación o diferencia entre los componentes del sistema, su función es definir los objetivos o el rumbo de los datos que se recaban. En relación con esto, hay cuestionamientos que

obtienen la misma respuesta, lo cual no supone un problema, ya que operan en calidad de elementos de control para determinar la concordancia de lo obtenido.

En este contexto, con base en un análisis de los elementos pedagógicos que se deben considerar en el PE-A (UNESCO, 1998; Pérez, 2021; Carballo, 1978; Medina y Mata, 2009; Yáñez, 2016; entre otros), aunado a los parámetros tecnológicos del estándar internacional de Especificación de Requerimientos IEEE830 (Gómez, 2011; Pfleeger, 2002), se establecen los aspectos a abordar en la entrevista: (1) el modelo educativo, que determina las actitudes y aptitudes del estudiante que se busca formar; (2) la teoría pedagógica, que describe las funciones de los actores en el PE-A y cómo ejecutarlo; (3) las estrategias de enseñanza y aprendizaje, que son la vía de transmisión y adquisición del conocimiento; (4) el diseño instruccional, que involucra todos los procesos de planeación y ejecución del PE-A; (5) el desarrollo tecnológico, el cual marca el rumbo técnico para la construcción de los elementos informáticos, este último obtiene información que complementa a los anteriores y al mismo tiempo es enriquecido por los mismos. A continuación, se presenta el instrumento (se recomienda que se aplicado por los expertos de ambas áreas):

Guion para la obtención de requerimientos tecnopedagógicos	
1. Modelo educativo	
<i>Aspecto a obtener</i>	<i>Definición filosófica-pedagógica</i>
	¿Qué tipo de alumno se quiere formar?
2. Teoría pedagógica	
<i>Aspecto a obtener</i>	<i>Definición de los actores</i>
	¿Quién aprende?
	¿Quién enseña?
<i>Aspecto a obtener</i>	<i>Establecer el PE-A</i>
	¿Cómo se lleva a cabo el aprendizaje y enseñanza?
	¿Cuáles son las actividades de los actores durante el proceso?
3. Estrategias de enseñanza y aprendizaje	
	<i>Docente</i>
<i>Aspecto a obtener</i>	<i>Contenidos dirigidos al docente</i>
	¿Qué elementos necesita para enseñar?
<i>Aspecto a obtener</i>	<i>Ajuste de necesidades (comunicación, manejo de contenidos y evaluación)</i>
	¿Cómo ajustas las necesidades de comunicación, manejo de contenidos?
	¿Cómo es el proceso de evaluación?
<i>Aspecto a obtener</i>	<i>Nivel de dominio cognitivo-afectivo-volitivo</i>
	¿Cuáles son los objetivos de aprendizaje que se tratan de cubrir?
	¿Cuáles son las habilidades socioemocionales (habilidades blandas/actitudes) que se buscan desarrollar?
	<i>Discente</i>
<i>Aspecto a obtener</i>	<i>Economía cognitiva (saturación de contenidos)</i>
	¿Cuánto tiempo inviertes en realizar una actividad académica?
	¿Cómo planeas el desarrollo de tus actividades académicas?

<i>Aspecto a obtener</i>	<i>Motivación</i>
	¿Qué activa tu interés al aprender?
<i>Aspecto a obtener</i>	<i>Comunicación síncrona y asíncrona</i>
	¿Cuál es la modalidad educativa?
	¿Tienes comunicación en tiempo real docente-discente?
4. Diseño Instruccional	
<i>Aspecto a obtener</i>	<i>Contexto educativo</i>
	Define el contexto educativo donde se impartirá el tema/curso.
	¿En qué grado escolar se impartirá el tema/curso?
	¿Cuál es el tema/curso que se impartirá?
	¿En cuánto tiempo estará distribuido el tema/curso?
	¿Cuáles son los objetivos de aprendizaje?
	¿Cuál es la importancia de lograr los objetivos de aprendizaje?
	Define la metodología a utilizar durante la(s) sesión/sesiones (secuencia didáctica)
	Establecer actividades en el software.
	Establecer actividades con el software.
	Establecer actividades sin el software.
	¿Cuáles son los temas transversales del tema/curso?
	Define los recursos a utilizar durante el tema/curso
	Define los espacios a utilizar durante el tema/curso

	¿Cómo se llevará a cabo el proceso de evaluación?		
Establecer evaluación profesor-alumno	Definir los instrumentos de evaluación en el software		
	Definir los instrumentos de evaluación con el software		
	Definir los instrumentos de evaluación sin el software		
Establecer evaluación alumno-profesor			
Establecer autoevaluación			
5. Desarrollo tecnológico			
<i>Aspecto a obtener</i>	<i>Ambiente físico</i>		
	¿Cuál es la ubicación del equipo donde funcionará el sistema?		
	Indicar si el sistema se instalará y funcionará en una ubicación o en varias		
	Describe las condiciones ambientales que se deban considerar tal como temperatura, humedad, problemas eléctricos o magnetismo		
<i>Aspecto a obtener</i>	<i>Interfaz</i>		
	¿Cuál es el origen o de dónde provienen los datos?		
	¿Los datos alimentan a otros sistemas?		
	¿Cuál es el formato de los datos?		
<i>Aspecto a obtener</i>	<i>Usuarios y factores humanos</i>		
	¿Quién dará uso al sistema?		
	¿Serán varios tipos de usuarios? ¿Cuáles serán sus privilegios o restricciones durante el uso del sistema?		
	¿Cuál es el nivel de habilidad tecnológica de cada tipo de usuario?		
	¿Qué clase de capacitación requerirá cada tipo de usuario para el uso del sistema?		
	¿Qué tan fácil le será al usuario comprender y utilizar el sistema?		
	¿Qué tan difícil le resultará al usuario hacer uso indebido del sistema?		

<i>Aspecto a obtener</i>	<i>Funcionalidad</i>
	¿Qué hará el sistema?
	¿Cuándo lo hará?
	¿Existen varios modos de operación? (por ejemplo: modo normal, modo seguro, modo a prueba de errores; control parental; entre otros).
	¿Cómo y cuándo se realizará el mantenimiento o actualización del sistema?
	Describe las restricciones de velocidad de ejecución, tiempo de respuesta o rendimiento
<i>Aspecto a obtener</i>	<i>Documentación</i>
	¿Cuánta documentación se requiere?
	¿Debe estar en digital, en papel o en ambos?
	¿A qué tipo de personas está orientado la diferente información?
<i>Aspecto a obtener</i>	<i>Datos</i>
	¿Cuáles son los tipos de datos que se esperan en la entrada y salida?
	¿Con qué frecuencia serán recibidos o enviados?
	¿Cuántos datos fluyen a través del sistema?
	¿Cada cuánto tiempo debe respaldarse la información?
<i>Aspecto a obtener</i>	<i>Recursos</i>
	¿Qué recursos se requieren para construir, usar y mantener el sistema?
	¿Cuánto espacio de almacenamiento será ocupado por el sistema?
	Define las fechas del plan de trabajo para el desarrollo
	¿Cuál es el monto máximo de dinero a gastar en el desarrollo?

<i>Aspecto a obtener</i>	<i>Seguridad</i>
	¿Cómo será el acceso al sistema o a la información? (controlado, libre, etc.)
	¿Hay alguna manera en especial para aislar los datos de un usuario de los de otros? En caso de que la respuesta sea sí, descríbala.
	¿Hay alguna manera en especial para aislar los programas de usuario de los otros programas y del sistema operativo? En caso de que la respuesta sea sí, descríbala.
	¿Con qué frecuencia deben hacerse copias de respaldo?
	¿Dónde deben almacenarse las copias de respaldo?
	¿Qué precauciones se deben tomar contra el fuego, el daño provocado por agua o el robo?
<i>Aspecto a obtener</i>	<i>Aseguramiento de la calidad</i>
	Además de confiabilidad, disponibilidad, facilidad de mantenimiento, seguridad ¿qué otros atributos son requeridos para asegurar la calidad?
	¿Cómo deben mostrarse las características del sistema a personas ajenas al mismo?
	¿Cómo debe detectar y generar un reporte de defectos el sistema?
	¿Cuánto es el tiempo para la recuperación del sistema después de una falla?
	¿Cómo se deben corregir errores o mejorar el sistema durante el mantenimiento?
	¿Cuáles son las reglas de trabajo para la eficiencia del uso de recursos y tiempo de respuesta?
	¿Cómo debe moverse el sistema de una ubicación a otra o de un equipo de trabajo a otro?

Después de la aplicación de la entrevista, es preciso concentrar la información obtenida y definir lo que se requiere implementar, por lo que, en esta etapa, se solicitan tres productos, en los que se logren plasmar las bases tecnopedagógicas

que servirán como parámetros durante la construcción del software; en el siguiente punto se describen a detalle.

Productos

Para iniciar, se busca que el contexto educativo sea determinado, lo cual se logrará mediante las primeras cuatro categorías del guion para la obtención de requerimientos tecnopedagógicos y definido por el experto pedagógico. En este sentido, el primer producto es la planeación didáctica que instituye de manera general las variables a considerar y pueden afectar la concepción del software educativo. Estos elementos se establecieron con base en lo descrito por Comunidades Digitales para el Aprendizaje en Educación Superior (2015), Murcia (2020), Albarrán (2014), entre otros. A continuación, se presenta el instrumento:

Planeación didáctica					
Nombre del proyecto					
Institución		Grupo			
Profesor					
Número de unidad		Campo de formación		Eje/ contenido/ ámbito	
Periodo		Asignatura		Tema/ contenidos	
Número de Sesiones					
Justificación					
Presentación-contextualización					
Conceptualización pedagógica	1. Modelo educativo:				
	2. Teoría pedagógica:				
	3. Definición de roles:				
	4. Definición de espacios de aprendizaje				
Evidencia de aprendizaje (competencias)					

Nivel Cognitivo-Afectivo-Volutivo			
Objetivo General			
Objetivos Específicos			
Metodología			
Fase	Actividades	Recursos	Tiempo
<i>Apertura</i>			
<i>Desarrollo</i>			
<i>Cierre</i>			
Temas transversales			
Evaluación			
Criterios de evaluación		Instrumentos de evaluación	Tipo de evaluación

Del mismo modo, es necesario definir las características del software desde el punto de vista tecnológico, dado que la base pedagógica determinada anteriormente no

es suficiente para establecer qué es lo que se va a implementar, por lo cual, el segundo producto de la fase será un listado de requerimientos que contendrá los elementos mínimos a considerar durante el desarrollo, con este se instituirán los requisitos funcionales y no funcionales del sistema. A continuación, se presenta la herramienta formulada:

Listado de Requerimientos						
Identificador	Nombre	Descripción	Complejidad	Impacto	Clasificación	Comentarios

Hasta el momento, se vislumbra que la información concentrada y analizada es observada desde puntos de vista que no están relacionados; sin embargo, su similitud ya fue establecida al inicio de esta etapa. En tales circunstancias, se encuentra necesario un vínculo o puente de integración de ambas perspectivas; por lo que, como producto final, es propuesta una representación tecnopedagógica, la cual busca combinar lo definido en la planeación didáctica y en el listado de requerimientos, en este formato se enlazan requisitos del software con su base educativa, por ende, la aplicación del primero se realizará bajo las condiciones del segundo; asimismo, es fundamental ser lo más específico posible en su llenado,

puesto que su resultado tiene que estar libre de interpretaciones o ambigüedades de diseño e implementación. A continuación, se presenta el instrumento:

Representación tecnoeducativa	
Identificador	
Requerimiento	
Definición pedagógica	
Requerimiento tecnopedagógico	
Descripción detallada de la implementación	

Fase 2: Diseño tecnopedagógico

Descripción

En este punto, se deben establecer puentes de comunicación entre el área tecnológica y la educativa, mediante la aplicación de técnicas de diseño convencionalmente utilizadas durante el desarrollo de software. Asimismo, se

recomienda tomar en cuenta aspectos de usabilidad⁴, convenciones⁵, la determinación o consideración de los canales de aprendizaje de los usuarios finales, entre otros. Por último, se busca que la complejidad (en términos técnicos de ingeniería de software) durante la ejecución de la fase se dé gradualmente, empezando con técnicas entendidas por todo el equipo de trabajo y terminando con elementos meramente ingenieriles.

Procedimiento

En primera instancia, se elaborarán los *mockups* del software con base en lo obtenido en la etapa anterior; estos son definidos como una representación visual con un nivel medio a alto de fidelidad de la apariencia del resultado final y que muestra conceptos básicos de su funcionalidad (Cao *et al.*, 2020). En estos se debe reflejar las necesidades a cubrir y ser lo más específicos posible, puesto que la distribución de los elementos del sistema, la tipografía, la presentación de la información, los colores de la pantalla y demás componentes correspondientes al *front end* estarán fundamentados en aspectos pedagógicos, por lo que cambiar o sustituir lo determinado podría afectar el objetivo que se busca alcanzar.

Después, se crearán los bosquejos de diseño que son prácticamente borradores UML, en estos no se seguirá el estándar, pero sí se partirá de él; esto

⁴ “Medida en la cual un producto puede ser utilizado por determinados usuarios para conseguir objetivos específicos con efectividad, eficiencia y satisfacción, en un contexto de uso determinado” (Organización Internacional de Normalización (ISO) citada por Dimuro, 2014, pág. 6).

⁵ “Las convenciones son una serie de normas no escritas, adquiridas por el uso cotidiano, que todo el mundo conoce, entiende e interpreta prácticamente de la misma forma” (Dimuro, 2014, pág. 12).

con base en el supuesto que UML tiene elementos fáciles de comprender sin importar el campo de conocimientos de los integrantes del equipo. Es relevante establecer tres reglas: la primera, aplicar la simbología del diagrama lo más apegado al estándar, si no es viable, utilizar caracteres cuyo significado pueda ser intuitivamente entendido por el grupo de trabajo; la segunda, usar un lenguaje totalmente convencional evitando tecnicismos que dificulten su entendimiento, asimismo, las notas u oraciones de cada boceto serán tan específicas como sea posible; la tercera, en la elaboración deberán de estar involucrados ambos expertos de área. El resultado será el puente de comunicación entre el educador y el programador.

Por último, con base en los bosquejos de diseño se elaborarán los diagramas UML, estos tienen que seguir el estándar sin modificación alguna, puesto que deben ser entendidos por el programador y serán con los que se desarrollará la codificación.

Productos

Al terminar esta fase, los entregables serán los siguientes:

- *Mockups*
- Bosquejos de diseño
- Diagramas UML

Fase 3: Implementación

Descripción

Esta etapa se refiere a la traducción del diseño hacia un lenguaje de programación, tal como Java, Python, C#, entre otros. La implementación será una tarea meramente ingenieril, pero el área educativa fungirá en calidad de auditora dentro de sus posibilidades y capacidades.

Procedimiento

La implementación se realizará solamente cuando el diseño haya sido terminado, el programador decidirá la forma óptima en la que necesita desarrollar el sistema; sin embargo, cada decisión deberá ser discutida y validada con el experto en educación. Asimismo, la codificación debe cumplir con las siguientes características (Tsui *et al.*, 2014; Gómez *et al.*, 2019):

- Código legible: debe ser inteligible para otros programadores.
- Código trazable: debe existir correspondencia entre cada elemento del código y cada componente de diseño, asimismo a la inversa, de forma que sea posible trazar estos procedimientos.
- Código correcto: la codificación debe satisfacer a los requerimientos.
- Código completo: todos los requerimientos se cumplen.
- Código eficiente (performance): el sistema debe funcionar con rapidez, cumpliendo con las características mencionadas.

Por último, es altamente recomendable hacer comentarios en el código, al igual que documentar esta fase con lenguaje que no contenga tecnicismos, con el fin de que el equipo de trabajo pueda comprender y así poder realizar las observaciones pertinentes del proceso.

Productos

El entregable de esta etapa será el software con todas las características definidas en los requerimientos, por otro lado, si se opta por desarrollar el sistema por módulos, estos deberán ser unificados para cumplir con el producto de la fase. Asimismo, para llevar el control durante esta etapa, se propone la siguiente matriz que define código relevante del software:

Identificador	Código por realizar	Resultado esperado	Comentario
Evidencia del código (agregar descripción detallada (paso a paso) con imágenes).			

Fase 4. Pruebas

Descripción

En esta etapa, se busca establecer el correcto funcionamiento del sistema y que se haya cumplido con las especificaciones establecidas, y así, alcanzar los objetivos

de aprendizaje, de igual forma, tomar las impresiones de los usuarios finales para generar las modificaciones correspondientes si son necesarias.

Procedimiento

Esta fase se divide en un par de momentos, en el primero se ejecutarán dos tipos de pruebas de escritorio⁶: (1) caja blanca, que comprueban que las unidades internas se implementen correctamente, en conjunto con estructuras y relaciones, con el fin de reducir errores internos; y (2) caja negra, que revisa las funciones externas, así como que el software contenga las especificaciones necesarias y cumpla con los requerimientos del usuario (Rodríguez, 2012). El auditor decidirá las condiciones en las que se realizaran estas técnicas, siempre y cuando se atienda con lo descrito.

En el segundo momento, se ejecutarán pruebas de campo donde los usuarios finales (o usuarios con características muy similares) utilizarán el software por un periodo limitado de tiempo, posteriormente se propone aplicar el SUS (*System Usability Scale* o Escala de Usabilidad de un Sistema), este es un instrumento con diez cuestionamientos que indagan en el usuario para establecer o valorar el grado de usabilidad de productos y servicios, de los cuales algunos son sistemas informáticos (Brooke, 1996); no obstante en este momento se puede hacer uso de cualquier otro instrumento que mida la usabilidad del sistema.

⁶ Pruebas *in situ* (ver Anexo I).

Productos

Para definir el cumplimiento de la etapa, se tomarán en cuenta las evidencias de las pruebas de escritorio y las de campo. Para las primeras que se componen por las de caja blanca y caja negra, se establece el siguiente formato de control:

Identificador	Prueba por realizar	Resultado esperado	Estatus	Comentario
Evidencia de la prueba (agregar descripción detallada (paso a paso) con imágenes).				

Por otro lado, la evidencia de las pruebas de campo se realizará mediante la aplicación del SUS (o algún otro instrumento que mida la usabilidad del sistema), recordado que esta escala cuenta con su propio instrumento de recopilación (ver Anexo II); asimismo, para considerar que un proyecto tiene un nivel óptimo, SUS deberá arrojar una calificación entre 80 hasta 100 (Brooke, 1996).

Fase 5: Mantenimiento funcional y educativo

Descripción

El desarrollo de software no se detiene cuando se entrega el sistema, sino que continua a través de su vida útil; en cuanto se despliega en el contexto educativo para el que fue creado (Sommerville, 2011), es inevitable seguir adaptándolo y escalándolo para que siga cumpliendo con los objetivos de aprendizaje.

Procedimiento

Esta etapa se lleva a cabo después de la entrega y despliegue del software educativo generado, su duración será el tiempo acordado entre el equipo de desarrollo y el encargado en solicitar el sistema, se recomienda que el periodo sea mínimo de un año. La fase no seguirá un orden establecido, sólo se tendrán que considerar los siguientes aspectos:

- El mantenimiento funcional indagará para encontrar y corregir los errores que muestre el software, buscando optimizar su funcionamiento. Se realizará en cuanto un usuario final reporte una falla durante el uso del sistema, no obstante, se deberán programar periódicamente con el fin de continuar con un desempeño correcto, estos periodos de tiempo serán producto de acuerdos establecidos entre el cliente y el equipo de trabajo. Asimismo, se podrán agregar o adaptar características de la definición tecnológica.
- El mantenimiento educativo se ejecutará al inicio de cada ciclo escolar, en este se buscará adaptar y escalar el sistema hacia cubrir las nuevas necesidades de aprendizaje con base en el contexto de enseñanza-aprendizaje que representan los actuales estudiantes que estudiarán el curso o tema.

Productos

La medición del cumplimiento de esta fase se dará mediante las evidencias de corrección de errores, optimización, adaptabilidad o escalabilidad. Dado que lo

anterior define una modificación en el software, para llevar un seguimiento adecuado, se establece el siguiente control de cambios:

Identificador	Cambio por realizar	Descripción del cambio	Tipo de cambio	Justificación	Comentarios

Fase transversal: JAD (Diseño Conjunto de Aplicaciones)

Descripción

Durante las cinco etapas es necesario obtener, controlar e implementar de manera clara y concisa los requerimientos tecnopedagógicos, debido que es común que existan deficiencias en la forma en que estarán representados en un sistema informático. Para evitar esto, se utilizará JAD (*Joint Application Design* o Diseño Conjunto de Aplicaciones) que es una de las metodologías líderes para el involucramiento y participación del usuario en el proceso de desarrollo del software (Carmel *et al.*, 1994).

JAD involucra al usuario en una serie de sesiones que, tradicionalmente, abordan de manera poco explícita el aspecto informático; sin embargo, dada la naturaleza considerablemente estructurada de las reuniones JAD, y el esencial rol del facilitador, tienen el potencial de convertirse en juntas altamente entendibles para el cliente (Carmel *et al.*, 1994). De igual forma, se implementa sobre ciclos de vida del desarrollo de software y metodologías de análisis estructurado;

conserva la organización de estas, mientras que expande la participación a los usuarios en las etapas especificación de requerimientos y diseño (Balda y Vicenzi, 1997).

Procedimiento

Se hará uso del JAD durante todo el proceso de desarrollo de software y es recomendable que sea aplicado constantemente. En primera instancia, la participación del usuario será entendida y parcialmente sustituida por el experto educativo, el cual fungirá como mediador y entendedor de las necesidades educativas que sean expresadas por parte del cliente, recordando que el experto tecnológico lo concebirá desde el apartado de su dominio. En este rol, el experto educativo estará en constante comunicación con el cliente, validando las homologaciones realizadas por el equipo de trabajo; asimismo, todo el conocimiento recabado del problema a resolver, lo apoyarán a proponer soluciones que satisfagan los requerimientos solicitados.

De igual forma, la comunicación entre los expertos debe ser lo más continua posible, buscando generar acuerdos que sean óptimos para el desarrollo tecnológico y educativo; un punto esencial será que ellos tienen que estar conformes con las resoluciones establecidas. Lo anterior, se realiza mediante juntas denominadas sesiones de entendimiento, las cuales tendrán los siguientes tres momentos (Balda y Vicenzi, 1997):

- Preparación: integrada por actividades que preparan la fase siguiente, participan el líder de sesión (gerente de proyecto) y los dos analistas (experto educativo y experto tecnológico).
- Sesión: reuniones donde los requerimientos y diseño del sistema son desarrollados en forma conjunta; el líder de sesión funge como facilitador de la dinámica de grupo y guía a los participantes en la elaboración de actividades, mientras tanto los analistas registran resultados.
- Revisión: los productos de las actividades previas producen las salidas formales del JAD; en conjunto, el líder de sesión y los analistas traducen las conclusiones de la reunión en un documento. Se realiza un prototipo con los resultados del diseño.

Además, como se mencionó al inicio del capítulo, se aprovecha la ventaja que representa un equipo multidisciplinario, debido a que el experto educativo y el tecnológico pueden realizar acuerdos rápidos (producto de la accesibilidad en la comunicación), los cuales son similares a los acuerdos de desarrollo con la diferencia de que no se establecen en las sesiones de entendimiento, sino que se toman en el momento que sea necesaria una modificación o surja algún inconveniente en la implementación que requiera una decisión al instante. De esta manera, se resuelven situaciones que no pueden esperar a las reuniones formales; no obstante, se tendrán que dialogar las decisiones tomadas durante estas y definir las ventajas de la determinación o si es necesario recular y replantear una nueva solución, no olvidando plasmar todo esto en un documento.

Productos

Los resultados esperados para cada momento son los siguientes: a) preparación: orden del día (Figura 5); b) sesión: mapas de conocimiento; c) revisión: minutas (Figura 6).

ORDEN DEL DÍA

Título de la reunión		Fecha
		Hora de inicio – Hora de finalización
Reunión organizada por	Nombre	
Asistentes:	Lista de asistentes	
Lea:	Lista de lectura	
Traiga:	Lista de suministros	
Hora de inicio – Hora de finalización	Actividad Elemento de línea 1 Moderador Elemento de línea 2 Moderador Elemento de línea 3 Moderador	Ubicación
Hora de inicio – Hora de finalización	Actividad Elemento de línea 1 Moderador Elemento de línea 2 Moderador Elemento de línea 3 Moderador	Ubicación
Hora de inicio – Hora de finalización	Actividad Elemento de línea 1 Moderador Elemento de línea 2 Moderador Elemento de línea 3 Moderador	Ubicación
Hora de inicio – Hora de finalización	Actividad Elemento de línea 1 Moderador Elemento de línea 2 Moderador Elemento de línea 3 Moderador	Ubicación
Instrucciones adicionales: Incluya instrucciones adicionales, comentarios o indicaciones en esta sección.		

Figura 5. Ejemplo de plantilla para la orden del día. Fuente: Plantillas de Microsoft Office (2023).

Minutas de la reunión

Llamada al orden
Tuvo lugar una reunión de [Nombre de la organización o del equipo] en [Ubicación] el [fecha]. Asistentes incluidos en [lista de nombres de los asistentes]. Miembros no asistentes incluidos en [lista de nombres].

Aprobación de las minutas
[Para reemplazar cualquier texto de marcador de posición (por ejemplo, este), selecciona una línea o un párrafo y comienza a escribir. Para obtener los mejores resultados, no incluyas espacios a la derecha ni a la izquierda de los caracteres de la selección.]

Informes
[¿Necesitas agregar títulos o más texto? ¡No hay problema! Para aplicar fácilmente cualquiera de los formatos de texto de este documento, ve a la galería Estilos en la ficha Inicio de la cinta de opciones.]

Asuntos pendientes
[Añade texto aquí].

Moción
[Añade texto aquí].

Nuevos asuntos
[Añade texto aquí].

Anuncios
[Añade texto aquí].

Levantamiento de la sesión
[Añade texto aquí].

Secretario Fecha de aprobación

Figura 6. Ejemplo de plantilla para las minutas de reunión. Fuente: Plantillas de Microsoft Office (2023).

En resumen, han sido planteadas las fases y elementos de la MCAE, por lo cual, en el siguiente capítulo se presentan las opiniones otorgadas por expertos de ambas áreas de estudio que revisaron la conceptualización de MCAE; así como, equipos de trabajo que implementaron software educativo mediante la propuesta; por último, la descripción de un Recurso Educativo Abierto creado con MCAE.

5. RESULTADOS

Se establecen dos procesos como criterios de admisión de la propuesta, el primero es una validación por juicio de expertos, en esta MCAE es sometida a revisión mediante profesionales especializados en Tecnología Educativa y la segunda es una validación por equipos de trabajo donde estos presentan sus impresiones; además se muestra el desarrollo de un Recurso Educativo Abierto mediante MCAE.

5.1. Validación por juicio de expertos

La propuesta se envió por correo electrónico a 50 expertos, donde 60% son especialistas mexicanos, 20% latinoamericanos (Honduras, Colombia, Argentina, Chile y Brasil) y 20% europeos (Grecia, España y Portugal). El procedimiento que se solicitó realizar fue el siguiente:

1. Revisar el documento adjunto. Este presentaba de manera detallada MCAE.

2. Contestar un cuestionario mediante Google Forms (ver Anexo III).
3. Describir impresiones y recomendaciones a integrar o descartar de la propuesta.

Con base en lo anterior, se obtuvo la colaboración de 15 especialistas (30%), los cuales 9 fueron de mujeres y 6 de hombres, este número de personas resulta óptimo para realizar un juicio por expertos. Cabero y Llorente (2013) discuten el número de participantes y, a partir de una revisión bibliográfica, mencionan un mínimo de 7 especialistas y que no sobrepasen los 50. En otra investigación, Cabero *et al.* (2021) instauran a 15 como número mínimo de colaboradores, todo en función de la complejidad y alcance de la investigación; haciendo énfasis en que la selección de los colaboradores se dio en función de su experiencia desarrollando software educativo sin importar el papel que desempeñaron o nivel educativo al que fue beneficiado. Dicho esto, se presentan las respuestas del cuestionario en Google Forms y los porcentajes mostrados en gráficas:

Se dividió en dos bloques, en el primero se busca precisar el perfil del especialista. En el cuestionamiento uno, se plantea *¿ha desarrollado software educativo?* A lo cual, el 100% contesta que sí, asimismo, en el segundo se solicita *¿en qué nivel educativo se desarrolló esta experiencia?* De la totalidad de los encuestados, el 20% comenta en preescolar, 33.3% en nivel medio básico, 20% en nivel medio superior, 66.7% en nivel superior y 13.3% en nivel posgrado (Figura 7). Cabe mencionar que esta opción era de elección múltiple, por lo que los especialistas pudieron seleccionar varias opciones.

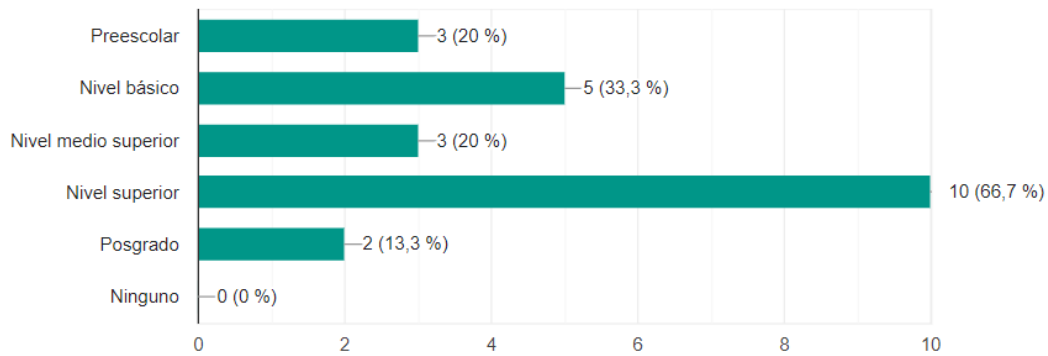


Figura 7. Porcentaje de participación durante el desarrollo de proyectos educativos en cada nivel académico. Fuente: Elaboración propia.

De igual forma, para la pregunta tres se establece los años de experiencia en Tecnología Educativa; 2 de los encuestados entre 1 y 2, 4 entre 3 y 5, 9 más de 5 (Figura 8).

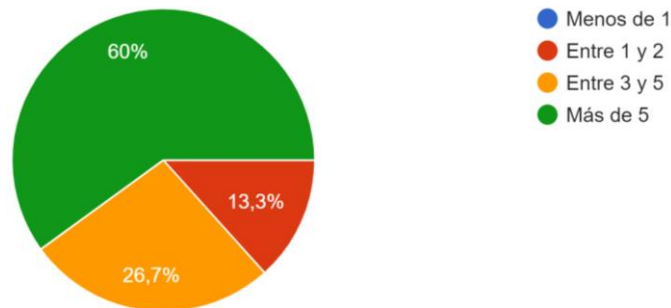


Figura 8. Experiencia en años de los especialistas entrevistados. Fuente: Elaboración propia.

Por último, en la siguiente lista, se muestran las respuestas otorgadas por los especialistas sobre los roles que han desempeñado durante el desarrollo de software educativo:

- Líder y desarrollador
- Analista

- Gerente de proyecto, experto tecnológico, diseñador, programador, auditor
- Analista, desarrollador
- Analista, diseñador, implementador
- Experto educativo, experto tecnológico
- Experto tecnológico, programador
- Experto tecnológico
- Desarrollador full stack
- Mayormente programador, aunque regularmente se requiere cubrir más de un rol a la vez, como diseñador y tester
- Líder de proyecto, diseñadora
- Compartir la experiencia como profesor
- Data analysis
- Analista y auditor

Por otro lado, la sección 2 está diseñada para obtener las impresiones de MCAE. El cuestionamiento *MCAE es clara es para su aplicación durante el desarrollo de software educativo*, 3 personas están muy de acuerdo, 9 están de acuerdo, mientras que 3 son neutrales ante esta afirmación (Figura 9).

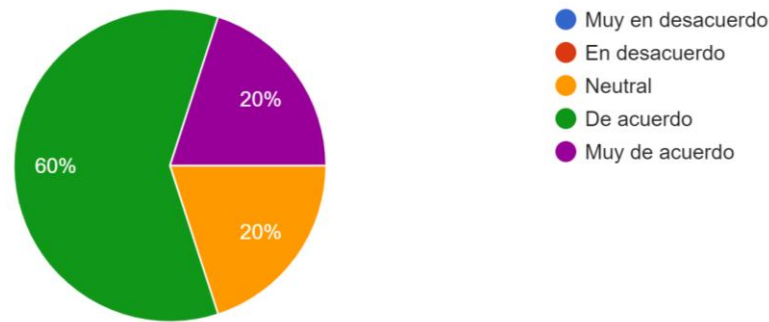


Figura 9. Respuestas a la pregunta: *MCAE es clara es para su aplicación durante el desarrollo de software educativo.* Fuente: *Elaboración propia.*

De la misma manera, las siguientes cuatro preguntas se busca establecer la suficiencia de lo que se propone: (1) *MCAE cumple con los métodos suficientes para el desarrollo de software educativo* donde 1 encuestado menciona estar muy de acuerdo, 13 de acuerdo y 1 neutral (Figura 10); (2) *MCAE cumple con las técnicas suficientes para el desarrollo de software educativo*, 2 muy de acuerdo, 9 de acuerdo, 3 neutral y 1 en desacuerdo (Figura 11); (3) *MCAE cumple con las estrategias suficientes para el desarrollo de software educativo*, 3 muy de acuerdo, 7 de acuerdo, 4 neutral y 1 en desacuerdo (Figura 12); (4) *MCAE cumple con los instrumentos suficientes para el desarrollo de software educativo*, 2 muy de acuerdo, 8 de acuerdo y 5 neutral (Figura 13).

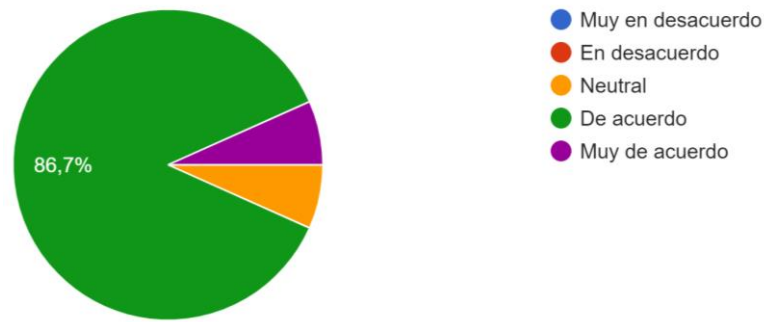


Figura 10. Respuestas a la pregunta: MCAE cumple con los métodos suficientes para el desarrollo de software educativo. Fuente: Elaboración propia.

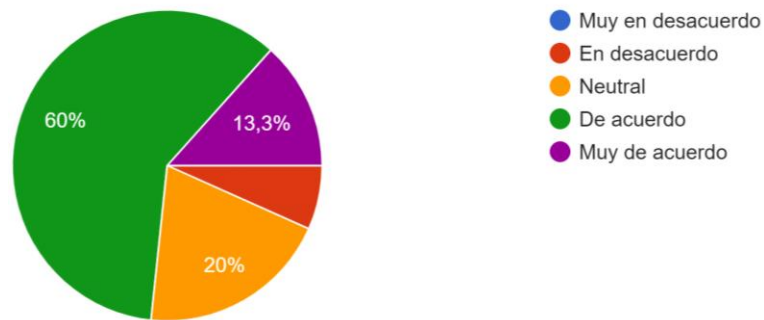


Figura 11. Respuestas a la pregunta: MCAE cumple con las técnicas suficientes para el desarrollo de software educativo. Fuente: Elaboración propia.

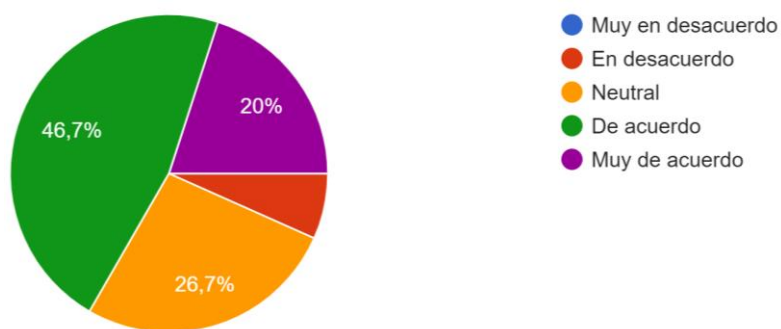


Figura 12. Respuestas a la pregunta: MCAE cumple con las estrategias suficientes para el desarrollo de software educativo. Fuente: Elaboración propia.

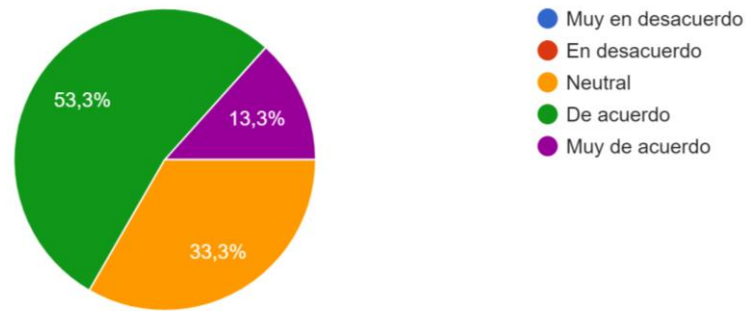


Figura 13. Respuestas a la pregunta: *MCAE cumple con los instrumentos suficientes para el desarrollo de software educativo*. Fuente: Elaboración propia.

Después, se busca determinar la idoneidad de los elementos integrados, *MCAE contiene los métodos adecuados para el desarrollo de software educativo*, 2 personas definen estar muy de acuerdo, 9 de acuerdo, 3 neutral y 1 en desacuerdo (Figura 14).

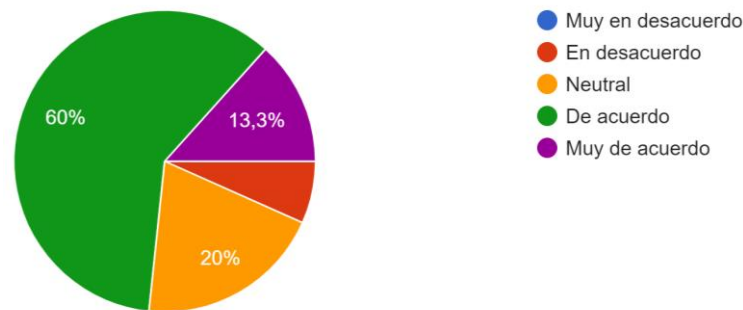


Figura 14. Respuestas a la pregunta: *MCAE contiene los métodos adecuados para el desarrollo de software educativo*. Fuente: Elaboración propia.

Posteriormente, las ocho próximas preguntas buscan determinar la claridad y suficiencia de los elementos propuestos para su uso durante el proceso de desarrollo. Las preguntas y sus respuestas son: (1) *Los métodos son claros para*

su aplicación durante el desarrollo de software educativo, 4 expertos establecen estar muy de acuerdo, 8 de acuerdo y 3 neutral (Figura 15); (2) *Los métodos contienen los elementos suficientes para el desarrollo de software educativo*, 1 muy de acuerdo, 9 de acuerdo, 4 neutral y 1 en desacuerdo (Figura 16); (3) *Las técnicas son claras para su aplicación durante el desarrollo de software educativo*, 2 muy de acuerdo, 7 de acuerdo y 6 neutral (Figura 17); (4) *Las técnicas contienen los elementos suficientes para el desarrollo de software educativo*, 1 muy de acuerdo, 8 de acuerdo, 5 neutral y 1 en desacuerdo (Figura 18); (5) *Las estrategias son claras para su aplicación durante el desarrollo de software educativo*, 3 muy de acuerdo, 8 de acuerdo, 3 neutral y 1 en desacuerdo (Figura 19); (6) *Las estrategias contienen los elementos suficientes para el desarrollo de software educativo*, 2 muy de acuerdo, 8 de acuerdo, 4 neutral y 1 en desacuerdo (Figura 20); (7) *Los instrumentos son claros para su aplicación durante el desarrollo de software educativo*, 4 muy de acuerdo, 6 de acuerdo, 4 neutral y 1 en desacuerdo (Figura 21); (8) *Los instrumentos contienen los elementos suficientes para el desarrollo de software educativo*, 2 muy de acuerdo, 9 de acuerdo, 2 neutral y 2 en desacuerdo (Figura 22).

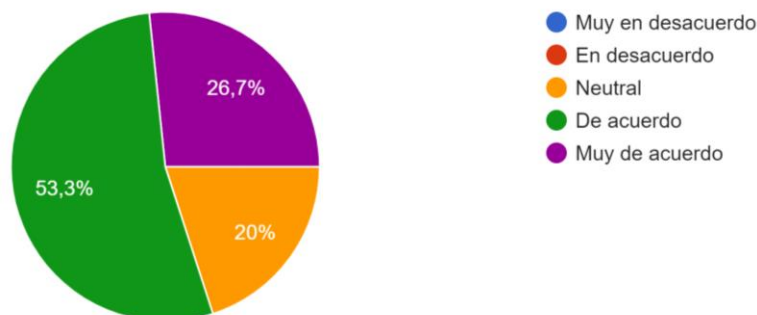


Figura 15. Respuestas a la pregunta: *Los métodos son claros para su aplicación durante el desarrollo de software educativo*. Fuente: Elaboración propia.

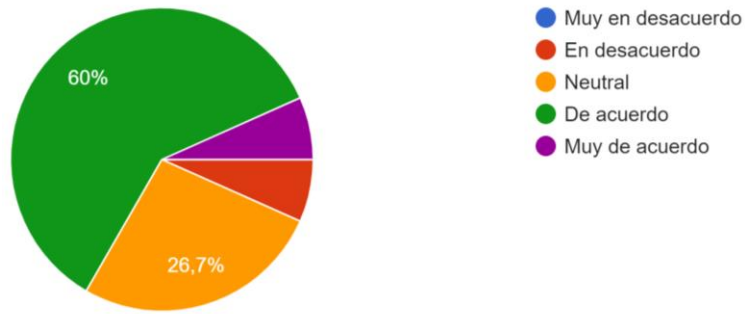


Figura 16. Respuestas a la pregunta: Los métodos contienen los elementos suficientes para el desarrollo de software educativo. Fuente: Elaboración propia.

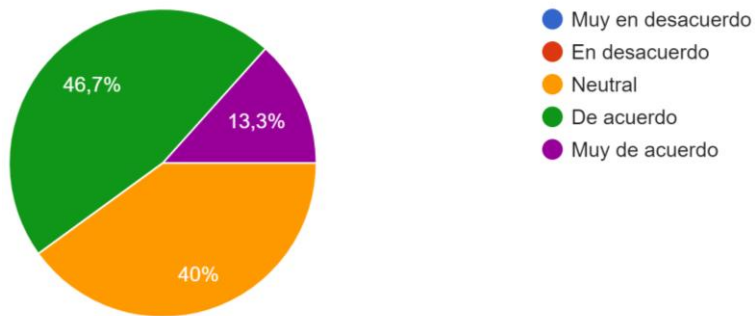


Figura 17. Respuestas a la pregunta: Las técnicas son claras para su aplicación durante el desarrollo de software educativo. Fuente: Elaboración propia.

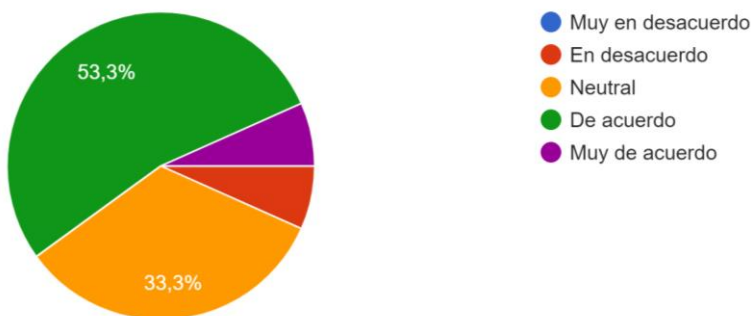


Figura 18. Respuestas a la pregunta: Las técnicas contienen los elementos suficientes para el desarrollo de software educativo. Fuente: Elaboración propia.

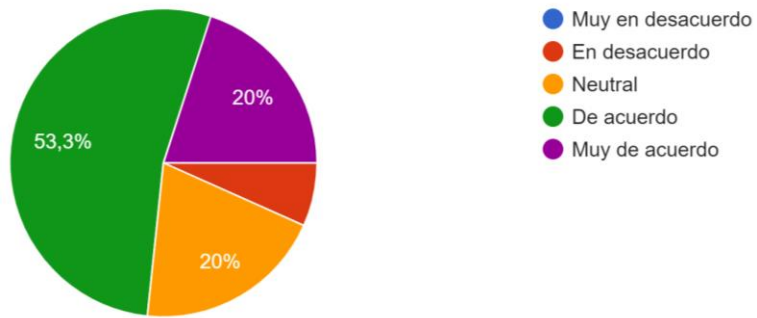


Figura 19. Respuestas a la pregunta: *Las estrategias son claras para su aplicación durante el desarrollo de software educativo.* Fuente: Elaboración propia.

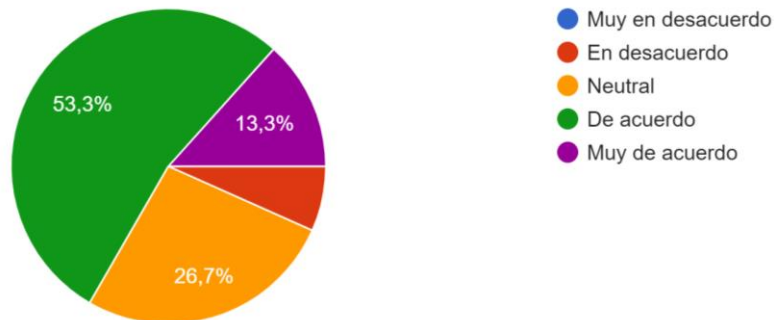


Figura 20. Respuestas a la pregunta: *Las estrategias contienen los elementos suficientes para el desarrollo de software educativo.* Fuente: Elaboración propia.

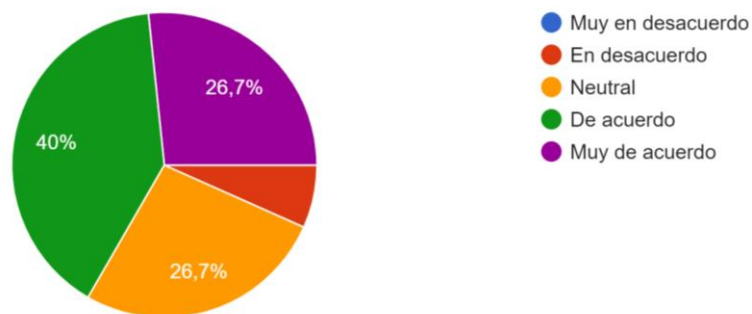


Figura 21. Respuestas a la pregunta: *Los instrumentos son claros para su aplicación durante el desarrollo de software educativo.* Fuente: Elaboración propia.

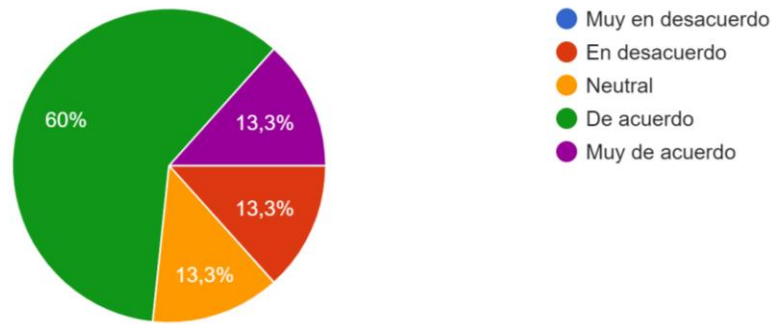


Figura 22. Respuestas a la pregunta: Los instrumentos contienen los elementos suficientes para el desarrollo de software educativo. Fuente: Elaboración propia.

Con base en las respuestas obtenidas, se aplicó el Alfa de Cronbach al instrumento, con el objetivo de medir su fiabilidad y validez, tanto de contenido como constructo, la fórmula a aplicar es $\alpha = \frac{k}{k-1} \left[1 - \frac{\sum Vi}{Vt} \right]$ donde k es el número de preguntas, Vi es la varianza de cada pregunta y Vt es la varianza de la sumatoria; esta representación se presenta en la Tabla 4 (Hernández-Sampieri *et al.*, 2014). Por último, el valor de cada respuesta se definió de la siguiente forma:

- Muy en desacuerdo = 1
- En desacuerdo = 2
- Neutral = 3
- De acuerdo = 4
- Muy de acuerdo = 5

Tabla 4. *Fiabilidad y validez del cuestionario aplicado a los especialistas.*

Especialista	Pregunta														Σ	α
	1	2	3	4	5	6	7	8	9	10	11	12	13	14		
I	3	3	3	3	3	3	3	3	3	3	3	3	3	3	42	0.953
II	4	4	2	3	3	4	4	3	3	3	4	3	4	2	46	
III	4	4	4	5	5	5	5	5	4	4	5	5	5	5	65	
IV	4	4	3	3	4	3	4	3	3	3	3	3	4	4	48	
V	5	4	4	4	4	5	4	4	4	4	4	4	4	4	58	
VI	3	4	4	4	4	4	4	4	3	4	4	4	4	4	54	
VII	4	4	4	4	4	3	4	4	4	4	4	4	4	4	55	
VIII	3	4	4	2	3	2	3	2	3	2	2	2	2	2	36	
IX	5	4	5	5	4	4	5	4	4	4	5	5	5	5	64	
X	4	4	4	4	3	4	4	4	3	3	4	4	3	4	52	
XI	4	4	4	3	3	4	4	4	4	4	3	3	3	3	50	
XII	4	4	3	4	4	4	5	4	5	5	4	4	5	4	59	
XIII	4	4	4	4	4	4	3	4	4	4	4	4	4	4	55	
XIV	5	5	5	5	5	4	5	4	5	4	5	4	5	4	65	
XV	4	4	4	4	4	4	4	3	4	3	4	4	3	4	53	
Varianza	0.428	0.143	0.6	0.743	0.457	0.6	0.495	0.524	0.495	0.543	0.695	0.638	0.838	0.781	69.267	

Fuente: Elaboración propia.

George y Mallery (2003, p. 231) mencionan la siguiente escala para categorizar un instrumento: >0.9 es excelente, >0.8 es bueno, >0.7 es aceptable, >0.6 es cuestionable, >0.5 es pobre, <0.5 es inaceptable. Por lo que, con base en el 0.953 obtenido, se puede establecer que es un instrumento con excelente confiabilidad de contenido y constructo.

Por último, se buscó definir las impresiones y comentarios, así como las recomendaciones hacia la propuesta. De manera general, lo mencionado por los especialistas es lo siguiente:

1. Asignar poco tiempo a las pruebas piloto y de aceptación es una de las razones de sobrepasar el costo y tiempo de entrega de un producto.
2. A través de un conjunto de herramientas, técnicas, estrategias y métodos didácticos, se busca afianzar un contenido, motivar y darle sentido al conocimiento, así como evaluar, diagnosticar y analizar las capacidades y dificultades de los alumnos.
3. Las responsabilidades del auditor (tester) se limitan a las técnicas (de caja blanca y negra) a una clasificación muy superficial, quizá convendría no especificar para no ser limitativos.
4. Respecto al flujo de trabajo se menciona que no se puede empezar una fase si no se ha terminado otra, lo que representa un desarrollo poco ágil, sugiero que al menos se puedan hacer en paralelo la programación y las pruebas para no esperar hasta terminar de programar para hacer las revisiones.
5. Considero que es necesaria una metodología para el desarrollo de software educativo y la iniciativa de MCAE me parece buen inicio.

6. Esta metodología es similar a la de cascada de desarrollo de software, actualmente las metodologías de vanguardia son las ágiles, en donde van analizando, diseñando y codificando; sin embargo, tal vez la clave es el tiempo de desarrollo y claro, eso depende de lo robusto del sistema. En conclusión, es una buena propuesta de metodología.
7. La metodología es interesante, aunque las fases son muy similares a las metodologías existente el enfoque que tiene hacia la parte educativa puede ser de ayuda principalmente para los desarrolladores que no están familiarizados con el ámbito educativo.
8. Considero interesante la propuesta de la metodología MCAE; sin embargo, sería relevante mencionar que están tomando de las otras metodologías o métodos para poder llevar a cabo la nueva propuesta, ya que pareciera que tomaron todo de la metodología JAD.
9. Esta metodología es una propuesta excelente consiguiendo un enfoque adaptativo, ya que parece homologar metodologías existentes con un objetivo específico, el propósito es mantener un trabajo muy fluido lleno de comunicación.
10. Hay que considerar que cada grupo es diferente y cuenta con experiencias y habilidades diferentes a los anteriores, es difícil la tarea de homogenizar un software educativo que funcione para la mayoría de los estudiantes y cuando quizá se piensa que se la logrado el objetivo, puede encontrarse ya en el final de su vida útil. Frecuentemente aparecen nuevas tendencias tecnológicas que resultan más atractivas para los estudiantes y como

profesores o desarrolladores de software educativo debemos adoptar y adaptar las nuevas tendencias tecnológicas en educación.

11. Interesante que cada fase señale la descripción, procedimiento y productos.
12. La propuesta es bastante interesante, detalla bastante bien la forma y flujo de trabajo buscando que los inconvenientes sean los menos posibles.
13. Siempre tomar en cuenta lo que ya existe para desarrollar algo mejor.
14. El formato de control de prueba, entiendo que es un formato para casos de prueba, de ser así creo que puede ser complementado con datos generales como fecha o quién lo realiza.

A partir de lo listado, se pueden establecer modificaciones de impacto en MCAE para seguir mejorando la experiencia de uso. Por otro lado, a continuación, se presentan las opiniones e impresiones obtenidas de los grupos desarrolladores, así como sus recomendaciones.

5.2. Validación por equipos de trabajo

Se dio a conocer la propuesta a tres equipos que desarrollarían software educativo, esto como parte de la legitimización de MCAE. Este punto se generó en dos momentos, el primero antes de empezar a trabajar donde se expuso en qué consistían las fases para obtener opiniones de mejoras y poder ser implementadas; algunas de las sugerencias fueron elaborar un glosario para términos desconocidos o poco comunes, agregar instrucciones o ejemplos de llenado en los instrumentos propuestos, presentar un sistema hecho bajo MCAE, entre otras.

El segundo momento se llevó a cabo después del proceso de desarrollo con el objetivo de medir el grado de usabilidad de la propuesta, para esto se utilizó SUS (Brooke, 1996), las respuestas a las afirmaciones del instrumento son: uno (totalmente en desacuerdo), dos (en desacuerdo), tres (ni de acuerdo ni en desacuerdo), cuatro (de acuerdo) y cinco (totalmente de acuerdo). Este instrumento se aplicó en conjunto de algunas preguntas abiertas buscando obtener las impresiones cualitativas generadas por la aplicación de MCAE en la elaboración de sus sistemas (ver Anexo IV). Las respuestas emitidas por los tres equipos de desarrolladores se presentan a continuación:

- Creo que me gustaría utilizar esta metodología con frecuencia: 100% establece estar de acuerdo.
- Encontré la metodología innecesariamente compleja: 33.3% menciona estar totalmente en desacuerdo mientras 66.7% ni de acuerdo ni en desacuerdo.
- Pensé que la metodología era fácil de usar: 66.7% define estar de acuerdo y 33.3% ni de acuerdo ni en desacuerdo.
- Creo que necesitaría el apoyo de un especialista para poder utilizar esta metodología: 66.7% confirma estar en desacuerdo mientras 33.3% ni de acuerdo ni en desacuerdo.
- Encontré que los diversos elementos de esta metodología estaban bien integrados: 33.3% comenta estar de acuerdo, 33.3% de acuerdo y 33.3% ni en acuerdo ni en desacuerdo.
- Pensé que había demasiadas inconsistencias en esta metodología: 33.3% explica estar totalmente en desacuerdo, 33.3% en desacuerdo mientras 33.3% ni de acuerdo ni en desacuerdo.

- Me imagino que la mayoría de la gente aprendería a utilizar esta metodología muy rápidamente: 33.3% exhibe estar ni de acuerdo ni en desacuerdo, 33.3% de acuerdo y 33.3% totalmente de acuerdo.
- Encontré la metodología muy complicada de aplicar: 33.3% expone estar Totalmente en desacuerdo, 33.3% en desacuerdo mientras 33.3% ni de acuerdo ni en desacuerdo.
- Me sentí muy seguro usando la metodología: 100% declara estar de acuerdo
- Necesitaba aprender muchas cosas antes de empezar a usar esta metodología: 66.7% dice estar totalmente en desacuerdo y 33.3% en desacuerdo.

La representación de lo anterior se muestra en la Tabla 5, recordando que: 1 es totalmente en desacuerdo, 2 en desacuerdo, 3 ni de acuerdo ni en desacuerdo, 4 de acuerdo y 5 totalmente de acuerdo.

Tabla 5. Usabilidad percibida por parte de los equipos de desarrollo.

Ítem	Equipo I	Equipo II	Equipo III
I	5	4	4
II	2	3	1
III	4	4	5
IV	1	3	2
V	4	4	5
VI	1	3	1
VII	4	4	5
VIII	1	3	1
IX	5	4	4
X	1	2	1

Fuente: Elaboración propia.

Posteriormente, se determina la usabilidad de los sistemas en una graduación entre el 0 y 100, a mayor puntuación, mayor es su usabilidad. El procedimiento para realizar es el siguiente (Brooke, 1996; Benito, 2019):

- Para cada ítem impar se resta uno a la cantidad determinada por encuestado.
- Para cada ítem par, a la cantidad de cinco se le resta el valor asignado por los usuarios.
- Se realiza la sumatoria de lo anterior para cada usuario.
- Al resultado de esta sumaria se realiza una multiplicación por 2.5 de cada entrevistado.
- Por último, se realiza una sumatoria de todos los resultados obtenidos y se divide entre el número de equipos desarrolladores.

La fórmula que busca representar lo anterior es $\phi = \frac{\Sigma[(\Sigma(x-1) + \Sigma(5-y))*2.5]}{n}$

y se interpreta de la siguiente manera:

- x = valores de las preguntas impares
- y = valores de los ítems pares
- n = número de usuarios
- ϕ = nivel de usabilidad

La aplicación del análisis determinado para obtener el valor de SUS se vislumbra en la Tabla 6.

Tabla 6. Procedimiento para la determinación del nivel de usabilidad para MCAE.

Ítem	Equipo I	Equipo II	Equipo III	$\Sigma[(\Sigma(x-1) + \Sigma(5-y)) * 2.5]$	ϕ
I (x - 1)	4	3	3	247.5	82.5
II (5 - y)	3	2	4		
III (x - 1)	3	3	4		
IV (5 - y)	4	2	3		
V (x - 1)	3	3	4		
VI (5 - y)	4	2	4		
VII (x - 1)	3	3	4		
VIII (5 - y)	4	2	4		
IX (x - 1)	4	3	3		
X (5 - y)	4	3	4		
Σ	36	26	37		
$\Sigma * 2.5$	90	65	92.5		

Fuente: Elaboración propia.

A partir de los resultados obtenidos mediante SUS y estableciendo la escala definida en las características por la herramienta, determinando que el mínimo requerido es 68 para que sea aceptable de lo evaluado y 100 siendo la excelencia, se determina que con 82.5, de manera general, MCAE se define correctamente a las necesidades de los equipos de trabajo, aunque se deben hacer modificaciones a partir de los comentarios proporcionados por los ítems abiertos del instrumento.

A continuación, se presentan las respuestas:

- En esta fase sugerimos ser un poco más específicos en cuanto a las preguntas para la obtención de requerimientos, debido a que en algunas el experto no sabía a qué se refería; sin embargo, este apartado permitió conocer mejor las necesidades que se tenían para poder construir el proyecto.

- En la parte tecnológica me resultó difícil responder las preguntas, por lo que considero necesario trabajar con el experto en tecnología para optimizar esta fase.
- Consideramos que esta fase se encuentra bien estructurada, porque contiene los elementos necesarios para poder realizar el diseño previo tanto físico como su funcionalidad a través de los diagramas UML.
- Conjunta claramente ambas disciplinas, mostrando coherencia y consistencia.
- Esta fase la consideramos muy necesaria, una de las ventajas significativas es que debido a la estructura de la matriz que debe de ser llenada, se puede visualizar mejor la codificación que se desea realizar especificando el resultado al que se desea llegar.
- Tal vez haría falta un manual o instructivo con los términos utilizados
- Es fácil de utilizar y permite al docente el manejo de contenidos de forma dinámica.
- Este apartado nos permitió realizar las pruebas acordes a los usuarios establecidos, para verificar el funcionamiento correcto del proyecto, y consideramos que la matriz de pruebas está bien estructurada.
- Quizá agregar un ciclo de retroalimentación.
- El manejo es fácil, los alumnos responden positivamente y les brinda autonomía después de esta fase.
- Esta fase de igual forma se encuentra bien especificado en cuanto a los elementos que se deben de considerar para poder realizar la parte del mantenimiento al termino y entrega del proyecto.

- Sería interesante observar lo que sucede al utilizar la metodología ya en el aula y con los alumnos/profesores.
- Es una metodología muy completa para la realización de aplicaciones educativas; sin embargo, puede llegar a ser tedioso estar realizando demasiados cuadros en algunas partes de las fases, la ventaja que le vemos a esta metodología es que nos permite visualizar de forma precisa el resultado final de lo que se desea desarrollar en cuanto a aplicaciones educativas.
- Agregar ejemplos o casos de estudio completos como guía para futuras implementaciones.
- Me parece una metodología completa, fácil de utilizar y con muchas ventajas respecto al desarrollo de competencias en los estudiantes. Une de forma clara lo tecnológico con lo pedagógico, recalcando la importancia de un trabajo conjunto.

5.3. Uso de MCAE para el desarrollo de un REA

Se generó un Recurso Educativo Abierto (REA) para el apoyo a la enseñanza del inglés en niños de una primaria privada del Estado de México mediante MCAE. Este recurso fue publicado en el número 26, volumen 14 de la revista *Diálogos sobre educación* de la Universidad de Guadalajara. Por lo que gran parte de ese trabajo está definido en esta sección, debido a que se busca dar un ejemplo detallado para la implementación de MCAE en el desarrollo de un software educativo, con el fin de servir como punto de referencia para los equipos que

busquen utilizar MCAE es sus proyectos. Este sentido, Benito y Soberanes (2022) definen lo siguiente:

Durante la etapa del análisis tecnopedagógico, se realizaron entrevistas mediante el “*guion para la obtención de requerimientos tecnopedagógicos*”; esto con la finalidad de determinar las características del REA, así como, las necesidades que se buscaban cubrir con el mismo.

Cabe mencionar, las preguntas fueron aplicadas por el líder de proyecto con ayuda de los expertos en tecnología y educación; los entrevistados fueron los actores del PE-A, en este caso, docentes que impartían el tema en los grados en los que se iba a emplear el recurso. A continuación, se presentan el resultado de lo obtenido, este fue llenado por los tres entrevistadores:

Guion para la obtención de requerimientos tecnopedagógicos	
1. Modelo educativo	
<i>Aspecto a obtener</i>	<i>Definición filosófica-pedagógica</i>
	¿Qué tipo de alumno se quiere formar?
	<i>Un alumno dotado de habilidades para gestionar su aprendizaje guiado por un docente</i>
2. Teoría pedagógica	
<i>Aspecto a obtener</i>	<i>Definición de los actores</i>
	¿Quién aprende?
	<i>El alumno</i>
	¿Quién enseña?
	<i>El docente, apoyado por herramientas tecnológicas</i>
<i>Aspecto a obtener</i>	<i>Establecer el PE-A</i>
	¿Cómo se lleva a cabo el aprendizaje y enseñanza?
	<i>Se utiliza direct method en el PE-A, en primer lugar, se requiere una actividad de arranque, como un juego o una acción para activar la disposición a aprender, luego se presenta un vocabulario a utilizar en la sesión, posteriormente la gramática a emplear y la práctica con distintos tipos de ejercicios que aumentan el grado de dificultad, finalmente se realiza una evaluación.</i>
	¿Cuáles son las actividades de los actores durante el proceso?
	<i>El docente actúa como mediador entre el software y el estudiante, el alumno realiza actividades interactuando con la plataforma.</i>
3. Estrategias de enseñanza y aprendizaje	
	<i>Docente</i>
<i>Aspecto a obtener</i>	<i>Contenidos dirigidos al docente</i>
	¿Qué elementos necesita para enseñar?
	<i>Material didáctico, marcadores, plataforma</i>
<i>Aspecto a obtener</i>	<i>Ajuste de necesidades (comunicación, manejo de contenidos y evaluación)</i>
	¿Cómo ajustas las necesidades de comunicación, manejo de contenidos?
	¿Cómo es el proceso de evaluación?
	<i>La evaluación es continua, se evalúa cada ejercicio y al final se obtiene un producto audiovisual para evaluar las habilidades productivas.</i>
<i>Aspecto a obtener</i>	<i>Nivel de dominio cognitivo-afectivo-volitivo</i>
	¿Cuáles son los objetivos de aprendizaje que se tratan de cubrir?
	<i>El alumno es capaz de describir su rutina diaria utilizando presente simple en primera y segunda persona así como vocabulario de acciones.</i>
	¿Cuáles son las habilidades socioemocionales (habilidades blandas/actitudes) que se buscan desarrollar?
	<i>El alumno desarrolla autonomía en su aprendizaje, además de habilidades sociales para compartir sus actividades diarias.</i>

	Discente
Aspecto a obtener	<i>Economía cognitiva (saturación de contenidos)</i>
	¿Cuánto tiempo inviertes en realizar una actividad académica? <i>En promedio cada actividad realizada en clase debe tener una duración de 10 a 15 minutos con el fin de diversificar técnicas y mantener la concentración y entusiasmo por parte de los alumnos.</i>
	¿Cómo planeas el desarrollo de tus actividades académicas? <i>Deben ser estructuradas de lo simple a lo complejo, sin saturar de información y utilizando distintas técnicas para diversificar las actividades</i>
Aspecto a obtener	<i>Motivación</i>
	¿Qué activa tu interés al aprender? <i>Material que pueda ser manipulado</i>
Aspecto a obtener	<i>Comunicación síncrona y asíncrona</i>
	¿Cuál es la modalidad educativa? <i>Presencial</i>
	¿Tienes comunicación en tiempo real docente-discente? <i>Sí</i>
4. Diseño Instruccional	
Aspecto a obtener	<i>Contexto educativo</i>
	Define el contexto educativo donde se impartirá el tema/curso. <i>Un grupo de una escuela primaria particular bilingüe ubicada en el Estado de México, el grupo consta de 20 alumnos con acceso a dispositivos electrónicos conectados a internet en todos los casos; presentan distintos estilos de aprendizaje, sienten el visual el predominante.</i>
	¿En qué grado escolar se impartirá el tema/curso? <i>2do grado</i>
	¿Cuál es el tema/curso que se impartirá? <i>Presente simple en inglés con rutinas diarias en primera y tercera persona</i>
	¿En cuánto tiempo estará distribuido el tema/curso? <i>Una sesión de 2 horas</i>
	¿Cuáles son los objetivos de aprendizaje? <i>El alumno produce oraciones simples en presente describiendo rutinas diarias</i>
	¿Cuál es la importancia de lograr los objetivos de aprendizaje? <i>Los estudiantes desarrollarán habilidades en el manejo del idioma de forma cercana al describir lo que ellos hacen diariamente, conectando su realidad con el tema de la clase.</i>
	Define la metodología a utilizar durante la(s) sesión/sesiones (secuencia didáctica)
Establecer actividades en el software.	<i>Ice breaker Actividades de gramática Evaluación de gramática</i>
Establecer actividades con el software.	<i>Presentación de vocabulario Presentación de gramática Evaluación (Escritura y envío de video)</i>

	Establecer actividades sin el software.	<i>Actividades de interacción entre compañeros de clase. Actividades en libro de texto Actividad de cierre</i>	
	¿Cuáles son los temas transversales del tema/curso?		
	<i>Pronombres personales y verbos</i>		
	Define los recursos a utilizar durante el tema/curso		
	<i>Software, dispositivos con acceso a internet, marcadores, flashcards</i>		
	Define los espacios a utilizar durante el tema/curso		
	<i>Aula de computo</i>		
	¿Cómo se llevará a cabo el proceso de evaluación?		
	Establecer evaluación profesor-alumno	Definir los instrumentos de evaluación en el software	<i>Ejercicios de gramática y escritura de control y libres.</i>
		Definir los instrumentos de evaluación con el software	<i>Video</i>
		Definir los instrumentos de evaluación sin el software	<i>Actividades de interacción entre pares y actividades en el libro.</i>
	Establecer evaluación alumno-profesor	<i>No aplica</i>	
	Establecer autoevaluación	<i>No aplica</i>	
5. Desarrollo tecnológico			
<i>Aspecto a obtener</i>	<i>Ambiente físico</i>		
	¿Cuál es la ubicación del equipo donde funcionará el sistema?		
	<i>En un servidor web</i>		
	Indicar si el sistema se instalará y funcionará en una ubicación o en varias		
	<i>No se instalará, se visualizará en equipos móviles y de escritorio mediante un navegador web</i>		
	Describe las condiciones ambientales que se deban considerar tal como temperatura, humedad, problemas eléctricos o magnetismo		
	<i>No aplica</i>		
<i>Aspecto a obtener</i>	<i>Interfaz</i>		
	¿Cuál es el origen o de dónde provienen los datos?		
	<i>Los Datos provienen de los alumnos</i>		
	¿Los datos alimentan a otros sistemas?		
	<i>No por ahora, pero se desea integrar sistemas más complejos para la enseñanza de inglés como segunda lengua.</i>		
	¿Cuál es el formato de los datos?		
	<i>Mayormente texto, con algunos apartados numéricos</i>		
<i>Aspecto a obtener</i>	<i>Usuarios y factores humanos</i>		
	¿Quién dará uso al sistema?		
	<i>Docentes y alumnos</i>		
	¿Serán varios tipos de usuarios? ¿Cuáles serán sus privilegios o restricciones durante el uso del sistema?		
	<i>Usuario único</i>		

	¿Cuál es el nivel de habilidad tecnológica de cada tipo de usuario? <i>Básico</i>
	¿Qué clase de capacitación requerirá cada tipo de usuario para el uso del sistema? <i>Mediante el manual de usuario</i>
	¿Qué tan fácil le será al usuario comprender y utilizar el sistema? <i>Lo más sencillo posible</i>
	¿Qué tan difícil le resultará al usuario hacer uso indebido del sistema? <i>El usuario tendrá actividades definidas, por lo que está opción no aplica.</i>
<i>Aspecto a obtener</i>	<i>Funcionalidad</i>
	¿Qué hará el sistema? Auxiliará el proceso de enseñanza aprendizaje mediante diversas actividades que evalúan el desempeño del estudiante en el manejo del tema.
	¿Cuándo lo hará? <i>Cuando se imparta el tema “Presente simple afirmativo con primera y segunda persona” en un aula de clases para niños.</i>
	¿Existen varios modos de operación? (por ejemplo: modo normal, modo seguro, modo a prueba de errores; control parental; entre otros). <i>No aplica</i>
	¿Cómo y cuándo se realizará el mantenimiento o actualización del sistema? <i>Después de la impartición del tema, se recopilarán observaciones y se harán las modificaciones pertinentes.</i>
	Describe las restricciones de velocidad de ejecución, tiempo de respuesta o rendimiento <i>Adaptar las características recientes de los nuevos dispositivos móviles y a equipos de escritorio de mediano o bajo rendimiento</i>
<i>Aspecto a obtener</i>	<i>Documentación</i>
	¿Cuánta documentación se requiere? <i>Manual técnico y de usuario</i>
	¿Debe estar en digital, en papel o en ambos? <i>Ambos</i>
	¿A qué tipo de personas está orientado la diferente información? <i>A profesores y alumnos</i>
<i>Aspecto a obtener</i>	<i>Datos</i>
	¿Cuáles son los tipos de datos que se esperan en la entrada y salida? <i>No aplica</i>
	¿Con qué frecuencia serán recibidos o enviados? <i>No aplica</i>
	¿Cuántos datos fluyen a través del sistema? <i>No aplica</i>
	¿Cada cuánto tiempo debe respaldarse la información? <i>No aplica</i>

<i>Aspecto a obtener</i>	<i>Recursos</i>
	¿Qué recursos se requieren para construir, usar y mantener el sistema?
	<i>A decisión del equipo de trabajo</i>
	¿Cuánto espacio de almacenamiento será ocupado por el sistema?
	<i>A decisión del equipo de trabajo considerando que sea lo óptimo posible.</i>
	Define las fechas del plan de trabajo para el desarrollo
	<i>Entrega diciembre del 2021, uso primer semestre del 2022</i>
	¿Cuál es el monto máximo de dinero a gastar en el desarrollo?
	<i>\$ 10 000</i>
<i>Aspecto a obtener</i>	<i>Seguridad</i>
	¿Cómo será el acceso al sistema o a la información? (controlado, libre, etc.)
	<i>Libre</i>
	¿Hay alguna manera en especial para aislar los datos de un usuario de los de otros? En caso de que la respuesta sea sí, descríbala.
	<i>No</i>
	¿Hay alguna manera en especial para aislar los programas de usuario de los otros programas y del sistema operativo? En caso de que la respuesta sea sí, descríbala.
	<i>No</i>
	¿Con qué frecuencia deben hacerse copias de respaldo?
	<i>No aplica</i>
	¿Dónde deben almacenarse las copias de respaldo?
	<i>No aplica</i>
	¿Qué precauciones se deben tomar contra el fuego, el daño provocado por agua o el robo?
	<i>No aplica</i>
<i>Aspecto a obtener</i>	<i>Aseguramiento de la calidad</i>
	Además de confiabilidad, disponibilidad, facilidad de mantenimiento, seguridad ¿qué otros atributos son requeridos para asegurar la calidad?
	<i>A consideración del equipo de trabajo</i>
	¿Cómo deben mostrarse las características del sistema a personas ajenas al mismo?
	<i>No aplica, cualquier persona puede ingresar al recurso</i>
	¿Cómo debe detectar y generar un reporte de defectos el sistema?
	<i>No aplica</i>
	¿Cuánto es el tiempo para la recuperación del sistema después de una falla?
	<i>No mayor a 72 horas naturales</i>
	¿Cómo se deben corregir errores o mejorar el sistema durante el mantenimiento?
	<i>Mediante una actualización del equipo de desarrollo.</i>
	¿Cuáles son las reglas de trabajo para la eficiencia del uso de recursos y tiempo de respuesta?
	<i>No aplica</i>
	¿Cómo debe moverse el sistema de una ubicación a otra o de un equipo de trabajo a otro?
	<i>Debe ser sencillo el mover el sistema de un servidor a otro</i>

Después de aproximadamente nueve entrevistas, el experto en educación genera un análisis a partir de lo definido con el guion para la obtención de requerimientos tecnopedagógicos; el resultado es plasmado en la planeación didáctica mostrada a continuación:

Planeación didáctica					
Nombre del proyecto	<i>My daily routine</i>				
Institución	<i>Anónima</i>	Grupo	<i>2do</i>		
Profesor	<i>Anónimo</i>				
Número de unidad	<i>3</i>	Campo de formación	<i>Comunicación</i>	Eje/ contenido/ ámbito	
Periodo	<i>2021-2022</i>	Asignatura	<i>Inglés</i>	Tema/ contenidos	<i>Presente simple afirmativo en primera y segunda persona</i>
Número de Sesiones	<i>Una sesión de 2 horas</i>				
Justificación	<i>El presente simple en inglés es el punto de partida para estructurar oraciones simples, en este tema es posible conjuntar vocabulario visto con anterioridad y es el primer contacto del alumno con los componentes de la oración</i>				
Presentación-contextualización	<i>Un grupo de 20 alumnos en una escuela primaria privada bilingüe en el Estado de México, la escuela cuenta con un salón de cómputo y de proyección de material audiovisual con acceso a internet y un equipo disponible para cada alumno. Los estilos de aprendizaje en el grupo son variados, siendo el visual-auditivo el más recurrente.</i>				
Conceptualización pedagógica	5. Modelo educativo: <i>Nueva Escuela Mexicana</i>				
	6. Teoría pedagógica: <i>Constructivismo</i> <i>Teoría del aprendizaje significativo</i>				
	7. Definición de roles: <i>El docente es mediador de contenidos</i> <i>El alumno mantiene un papel activo</i>				

	8. Definición de espacios de aprendizaje <i>Salón de clases</i> <i>Aula de computo</i>		
Evidencia de aprendizaje (competencias)	<ul style="list-style-type: none"> - Productos: <i>Cuadernos</i> <i>Video</i> <i>Actividades de plataforma</i> 		
Nivel Cognitivo-Afectivo-Volutivo			
Objetivo General			
<i>Describe su rutina diaria utilizando vocabulario específico.</i>			
Objetivos Específicos			
<ul style="list-style-type: none"> - <i>Reconoce pronombres personales</i> - <i>Estructura oraciones simples</i> - <i>Identifica el orden de las actividades diarias</i> - <i>Produce textos orales y escritos</i> 			
Metodología			
Fase	Actividades	Recursos	Tiempo
<i>Apertura</i>	<i>Ice breaker</i>	<i>Plataforma</i>	<i>5 min</i>
<i>Desarrollo</i>	<ol style="list-style-type: none"> 1. <i>Presentación del vocabulario (35 min)</i> <ul style="list-style-type: none"> - <i>Identifica acciones que realizamos todos los días</i> - <i>Practica pronunciación con ayuda de imágenes</i> - <i>Clasifica las actividades según el momento del día en que las realizas: Mañana, tarde o noche</i> 2. <i>Grammar (70 min)</i> <ul style="list-style-type: none"> - <i>Observa las flashcards para recordar Personal pronouns</i> - <i>Utiliza las tarjetas para mostrar cómo se estructura una oración</i> - <i>Manipula la plataforma para formar tus propias oraciones</i> - <i>Comparte con tus compañeros algunas oraciones que formaste</i> - <i>Resuelve las actividades en el software</i> - <i>Comparte con un compañero las actividades que realizas todos los días</i> - <i>Resuelve las actividades en tu libro</i> 		<i>105 min</i>
<i>Cierre</i>	<ul style="list-style-type: none"> - <i>Actúa una actividad de tu rutina diaria para que tus compañeros adivinen, produce oraciones simples.</i> <p><u>El video y la evaluación de plataforma son tarea.</u></p>		<i>10 min</i>
Temas transversales	<ul style="list-style-type: none"> - <i>Pronombres personales</i> - <i>Acciones</i> 		

Evaluación		
Criterios de evaluación	Instrumentos de evaluación	Tipo de evaluación
<i>Video en plataforma.</i> - <i>Habilidades sociales (Presentación y cierre)</i> - <i>Pronunciación</i> - <i>Estructura de oraciones</i>	<i>Rubrica</i>	<i>Cualitativa</i>
<i>Ejercicio en Plataforma</i> - <i>Gramática correcta</i> - <i>Ortografía</i>	<i>Plataforma</i>	<i>Cuantitativa</i>
<i>Actividades en clase</i> - <i>Estructura gramatical</i> - <i>Ortografía</i> - <i>Habilidades sociales</i>	<i>Registro descriptivo</i>	<i>Formativa</i>

Simultáneamente, el experto tecnológico generó el listado de requerimientos donde es establecido el análisis de las necesidades tecnológicas que se implementaron durante el desarrollo.

Listado de Requerimientos					
Identificador	Nombre	Descripción	Complejidad	Impacto	Comentarios
001NF	Hosting	El sistema está alojado en un servicio de hosting.	Baja	Bajo	
0020F	Tema	El sistema será usado para la enseñanza del tema “ <i>Simple Present</i> ”.	Media	Alto	El contenido será definido por agentes externos. Es necesario realizar la representación tecnoeducativa.
0030F	Ejercicios	El sistema presentará ejercicios y actividades relacionadas con el tema.	Media	Alto	El contenido será definido por agentes externos. Es necesario realizar la representación tecnoeducativa.
004NF	Usuarios	El sistema será usado por alumnos de los primeros grados de primaria.	Media	Medio	Sin embargo, podrá ser utilizado por cualquier persona interesada por aprender el tema.
005NF	Compatibilidad	El sistema podrá ser visualizado desde cualquier dispositivo, SO o navegador.	Baja	Alto	
0060F	Eficiencia	El sistema estará desarrollado para que cualquier funcionalidad y operación sea menor a 5 segundos.	Baja	Bajo	
007NF	Re-Despliegue	El sistema reanudará operaciones, no más de 72 horas naturales después de cualquier fallo o mantenimiento,	Alta	Bajo	
008NF	Adaptabilidad	El sistema buscará adaptarse a las nuevas necesidades cada que sea necesario.	Alta	Alto	Por lo regular sucederá a cada inicio de ciclo académico.
009NF	Escalabilidad	El sistema contendrá las características necesarias para el aprendizaje del tema; sin embargo, se podrán agregar nuevas si se identifican que son necesarias para el aprendizaje de este o temas relacionados.	Alta	Medio	
010NF	Estructura	El sistema tendrá una estructura definida a partir de la metodología de enseñanza (<i>Direct Method</i>).	Baja	Alto	
011NF	Distribución	El sistema tendrá una distribución de información basada en métricas de usabilidad.	Baja	Alta	Se establecerá mediante la representación tecnoeducativa.

Por último, mediante un par de sesiones de entendimiento, se establecieron homologaciones con fundamento pedagógico para la implementación de elementos del software, estos fungirán como recursos de apoyo de la enseñanza del tema de aprendizaje. Los resultados fueron los siguientes:

Representación tecnoeducativa	
Identificador	001RTE
Requerimiento	El sistema tendrá una distribución de información basada en métricas de usabilidad.
Definición pedagógica	<p>Modelo educativo: Nueva Escuela Mexicana</p> <p>Teoría pedagógica: Constructivismo Teoría del aprendizaje significativo</p>
Requerimiento tecnopedagógico	El sistema tendrá una distribución de información basada en métricas de usabilidad mediante la Teoría del aprendizaje significativo.
Descripción detallada de la implementación	<p>Desarrollar lo siguiente:</p> <ul style="list-style-type: none"> - Separar la pantalla en cuatro apartados: <ul style="list-style-type: none"> o Header, parte superior o Menu, parte central izquierda o Body, parte central derecha (esta área sea la más grande). o Footer, parte inferior.

Representación tecnoeducativa	
Identificador	002RTE
Requerimiento	El sistema será usado para la enseñanza del tema “ <i>Simple Present</i> ”.
Definición pedagógica	<p>Modelo educativo: Nueva Escuela Mexicana</p> <p>Teoría pedagógica: Constructivismo Teoría del aprendizaje significativo</p>
Requerimiento tecnopedagógico	El sistema presentará el vocabulario a utilizar durante la lección “ <i>Simple Present</i> ” de la materia de inglés mediante el modelo de la “Nueva Escuela Mexicana” estableciendo al “Constructivismo” como filosofía del proceso enseñanza-aprendizaje.
Descripción detallada de la implementación	<p>Desarrollar lo siguiente:</p> <ul style="list-style-type: none"> - Separar la sección en dos partes. - En la parte inferior establecer un carrusel con miniaturas de imágenes. - Al hacer clic en cualquier imagen de carrusel, en la parte superior mostrar la imagen en mayor tamaño. - Por último, al lado de la imagen grande, mostrar en un texto alusivo a la imagen.

Representación tecnoeducativa	
Identificador	003RTE
Requerimiento	El sistema será usado para la enseñanza del tema “ <i>Simple Present</i> ”.
Definición pedagógica	<p>Modelo educativo: Nueva Escuela Mexicana</p> <p>Teoría pedagógica: Constructivismo Teoría del aprendizaje significativo</p>
Requerimiento tecnopedagógico	El sistema mostrará el tema “ <i>Simple Present</i> ” de la materia de inglés mediante el modelo de la “Nueva Escuela Mexicana” estableciendo al “Constructivismo” como filosofía del proceso enseñanza-aprendizaje.
Descripción detallada de la implementación	<p>Desarrollar lo siguiente:</p> <ul style="list-style-type: none"> - Tabla con tres columnas y tres filas. - En la primera fila se estipularán los elementos para la formación de las oraciones con el siguiente orden: <i>Subject</i> (1), + (2), <i>Verb</i> (3). - En la segunda fila se insertarán dos listas, la primera (1) contendrá los pronombres personales, la segunda (3) contendrá los verbos más comunes utilizados en un vocabulario cotidiano. - En la tercera fila aparecerán imágenes alusivas al pronombre (1) y verbo (3) seleccionado. - Por último, fuera de la tabla en la parte inferior, se formará la oración creada en ambas listas.

Al terminar este punto, se puede hacer uso de los *mockups* para representar de manera gráfica lo que se busca implementar; sin embargo, este equipo de trabajo lo definió hasta la siguiente fase, lo cual es meramente por forma de desarrollo.

De manera general, la fase se resume en:

Una plataforma enfocada a alumnos de primaria baja cuyo uso apoye la enseñanza de la materia de inglés, específicamente de tiempo verbal presente simple en primera y segunda persona del singular y plural; dicho recurso ayudará desarrollar autonomía en la construcción del conocimiento por parte de los estudiantes, desarrollando así distintas habilidades. El proceso de enseñanza-aprendizaje da inicio con un “*warmer*” que les permite adentrarse a la mecánica de la sesión mediante el juego o la música; posteriormente, tiene lugar el contenido temático, dividido en tres momentos: inicio, dónde se introduce el vocabulario a utilizar; desarrollo, donde se presenta el contenido gramatical o tema a estudiar; cierre, en el cual se consolida y evalúa el aprendizaje a través de la producción de textos orales o escritos y la evaluación continua así como la formal.

Los actores en dicho proceso poseen roles definidos, el profesorado actúa como mediador de contenidos y conocimientos aplicando las técnicas y estrategias de aprendizaje necesaria para ayudar a los alumnos a construir conocimientos; los estudiantes adquieren destrezas y aptitudes

que no sólo les beneficiaran en el ámbito educativo, sino como seres sociales, todo esto orientado por el docente; los alumnos además realizan actividades de consolidación y fortalecimiento del conocimiento, las cuales pueden incluir producción de oraciones simples enfocadas al vocabulario y gramática vistos, complementar fragmentos, para luego producir por ellos mismos de forma oral o escrita, siendo esto último, fundamental en su proceso de aprendizaje, pues el nuevo idioma les está brindando las herramientas para hablar de ellos y sus contextos, apoyando sus procesos sociales. Es por lo anterior que las actividades propuestas se buscan adaptar a sus necesidades y a los objetivos de aprendizaje que se enlistan a continuación (Benito y Soberanes, 2022):

- Expresar actividades diarias utilizando presente simple en primera persona y vocabulario de rutinas.
- Que el alumno sea capaz de comunicarse y recibir información de forma oral y escrita.
- Conocer el vocabulario sobre rutinas.
- Describir rutinas matutinas, vespertinas y nocturnas.
- Utilizar el simple presente para hablar sobre rutinas.
- Construir oraciones sencillas y formular respuestas ante preguntas.
- Interaccionar de forma significativa, con cierta autonomía y eficacia para expresar información en inglés.
- Estructurar textos breves en inglés que contengan elementos básicos de cohesión.

- Hacer uso de las TIC con el fin de mostrar las habilidades adquiridas en el tema.
- Trabajar de forma colaborativa.

En la etapa del diseño tecnopedagógico, se utilizaron técnicas variadas para conceptualizar el software; la primera fue el uso de *mockups* (Figuras 25 - 27) que sirven para obtener una visualización anticipada de lo que se busca construir y recibir retroalimentación de los usuarios (Cao *et al.*, s/f); esto fue realizado por el líder de proyecto y los expertos con apoyo de un diseñador experimentado.

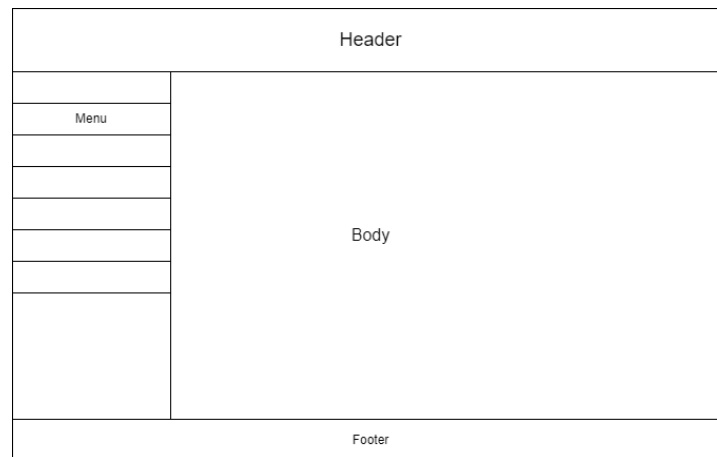


Figura 23. Mockup de la distribución del REA. Fuente: Elaboración propia.

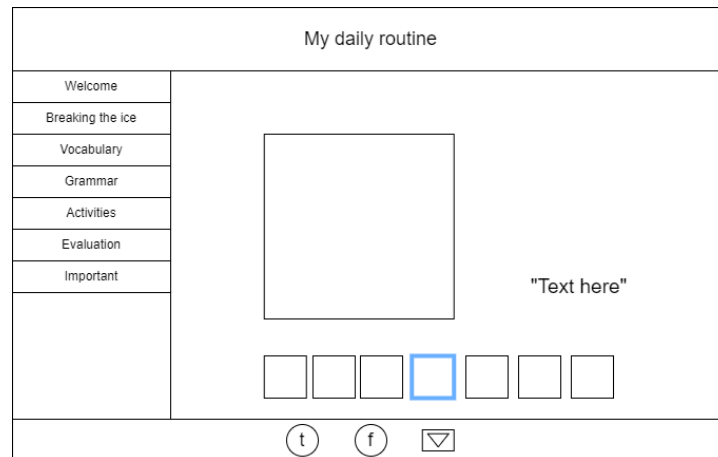


Figura 24. Mockup de la presentación del vocabulario. Fuente: Elaboración propia.

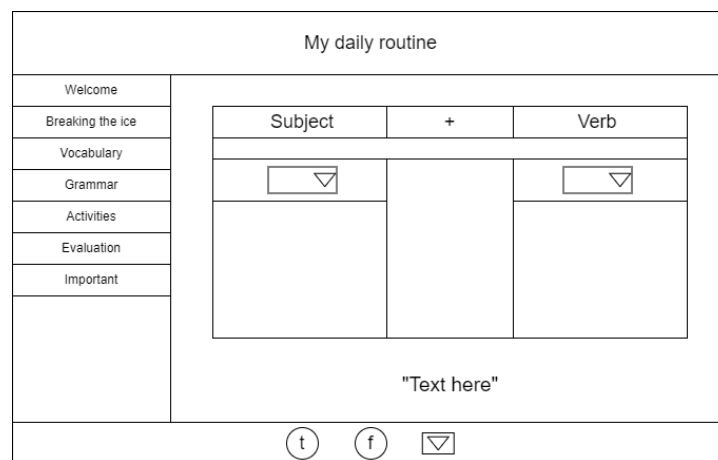


Figura 25. Mockup de la presentación del tema. Fuente: Elaboración propia.

Después, mediante una sesión de entendimiento, el experto tecnológico y el diseñador explicaron de manera general la esencia de algunos diagramas UML, al terminar, el experto educativo plasmó lo que pensó que realizaba el estudiante dentro del sistema (Figura 28) y cómo lo hacía (Figura 29); esto, en supervisión de los otros integrantes del equipo.



Figura 26. Bosquejo de diseño para las acciones del usuario en el sistema. Fuente: Elaboración propia.

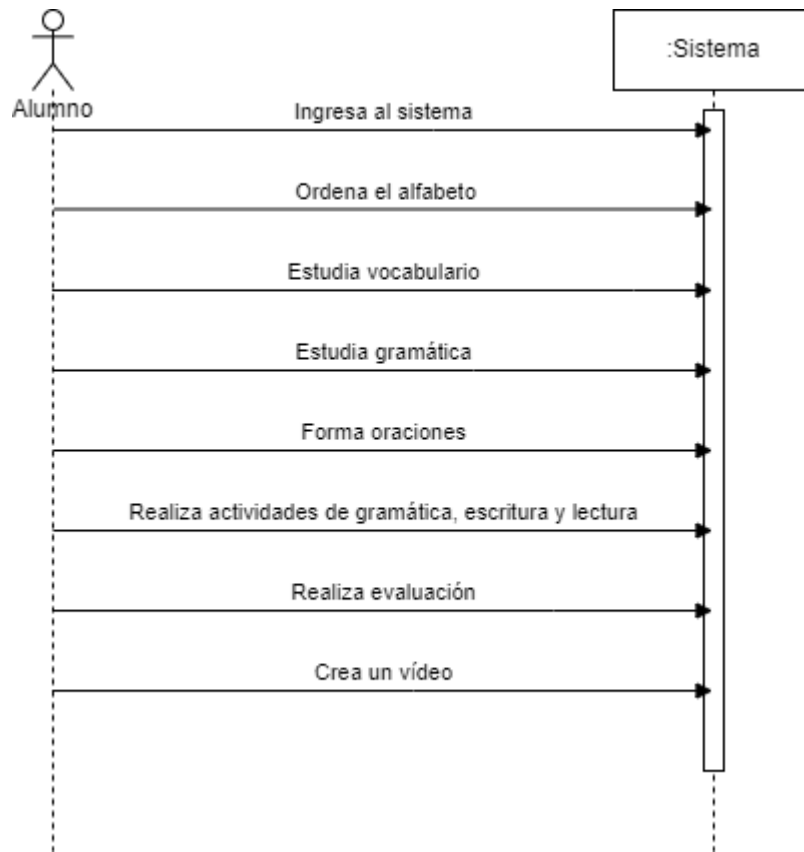


Figura 27. Bosquejo de diseño para el orden de las acciones del estudiante en el sistema.
Fuente: Elaboración propia.

En este punto, el experto educativo entendía cómo se estaba llevando el desarrollo, por lo que durante las sesiones de entendimiento no había problema alguno al momento de la toma de decisiones. Es por lo que, a partir de los bosquejos de diseño, se estableció un diseño formal con estándares; en primera instancia, se definió el mapa de sitio (Figura 30); posteriormente, el diagrama de casos de uso (Figura 31); por último, los diagramas de secuencia (Figuras 32 y 33).

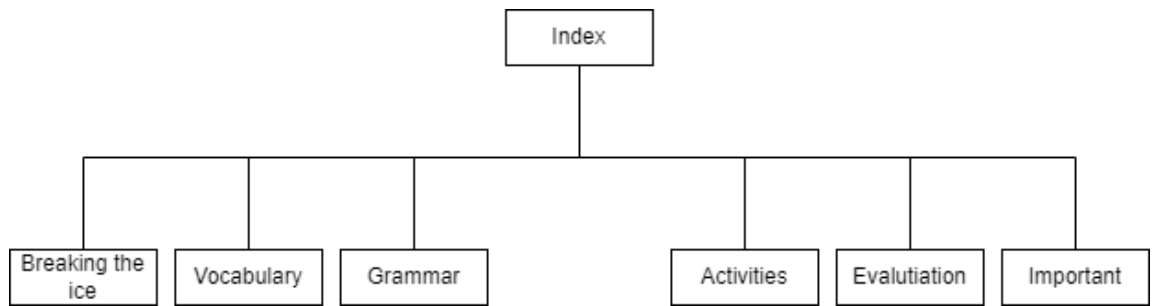


Figura 28. Mapa del sitio del REA. Fuente: Elaboración propia.

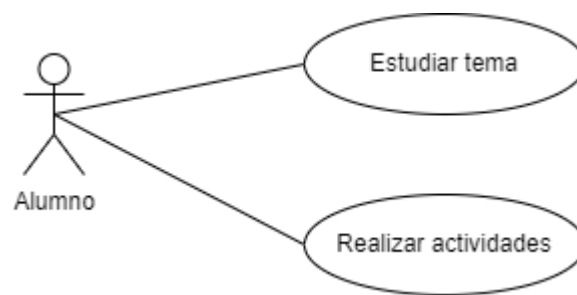


Figura 29. Diagrama de casos de uso del REA. Fuente: Elaboración propia.

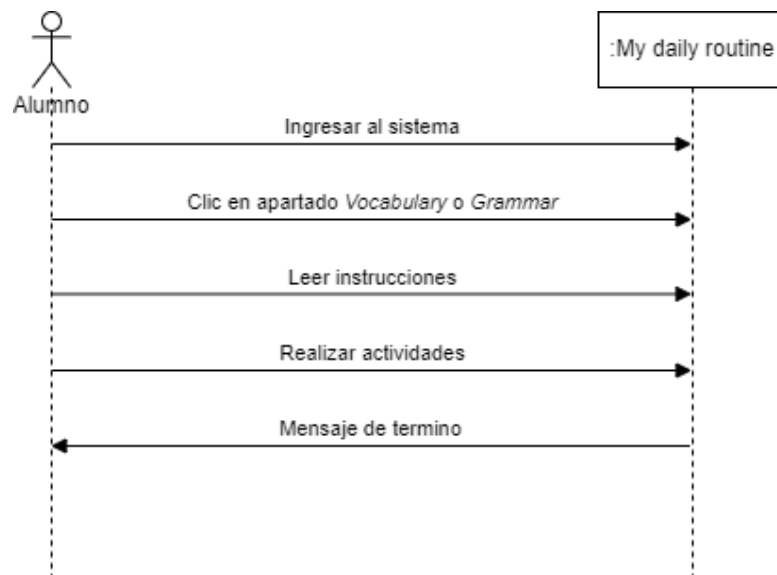


Figura 30. Diagrama de secuencia para “realizar actividades”. Fuente: Elaboración propia.

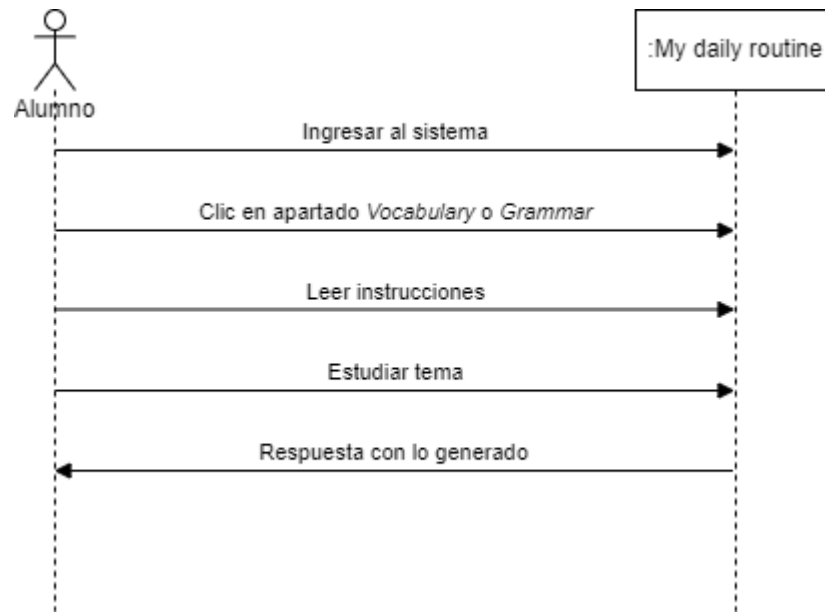


Figura 31. Diagrama de secuencia para “estudiar tema”. Fuente: Elaboración propia.

La etapa de implementación fue llevada a cabo por el experto tecnológico que fungió como programador. En esta fase se validaba cada módulo o código desarrollado (Figura 16), el experto educativo no dejaba de retroalimentarse de la codificación del sistema, si bien no entendía los estándares, ambos expertos tenían diálogos para entender y llevar a cabo de manera óptima la homologación de elementos del software sin dejar de lado la justificación pedagógica.

Identificador	Código por realizar	Resultado esperado	Comentario
001IM	Carrusel vocabulario	En un apartado del sistema se podrá visualizar una lista de imágenes que contendrán palabras establecidas por el experto educativo. Se espera dinamismo durante la navegación del carrusel.	Poner especial atención en los colores e imágenes seleccionadas.
<pre> <li class="carousel_slide"> <figure> <div> </div> <figcaption style="font-size: 2.5em"> Wake Up </figcaption> </figure> <li class="carousel_slide"> <figure> <div> </div> <figcaption style="font-size: 2.5em"> Get Dressed </figcaption> </figure> <li class="carousel_slide"> <figure> <div> </div> <figcaption style="font-size: 2.5em"> Have Breakfast </figcaption> </figure> </pre>			

Para entender de manera particular la toma de decisiones y el fundamento pedagógico, se explicará el porqué de estas. En primera instancia, se establecieron los colores a utilizar en el REA con base en lo descrito por Ortiz (2014); la autora subraya que en el ambiente educativo los colores son facilitadores, dado que asisten en captar la atención de los estudiantes, reforzando el aprendizaje y estimulando emociones al mismo tiempo en su proceso de aprendizaje. De acuerdo con Zubillaga *et al.* (2016) la psicología del color resulta fundamental en el desarrollo de los individuos, dado que

“en el inconsciente de como capta la sensación de los espacios educativos, el cual tiene influencia en el estado de ánimo o actitud de estos en su desarrollo e interacción”. Dicho esto, se seleccionaron tres colores principales en el diseño, como se aprecia en la Figura 34, explicados por Heller (2004):

- a) Naranja. Tiene un efecto en el humor, y en el contexto educativo apoya en la asimilación de contenidos.
- b) Verde. Promueve la concentración, brindando armonía, en el ambiente escolar se recomienda para resaltar ideas dentro de contenidos extensos.
- c) Gris. Al ser un tono neutro, proporciona equilibrio

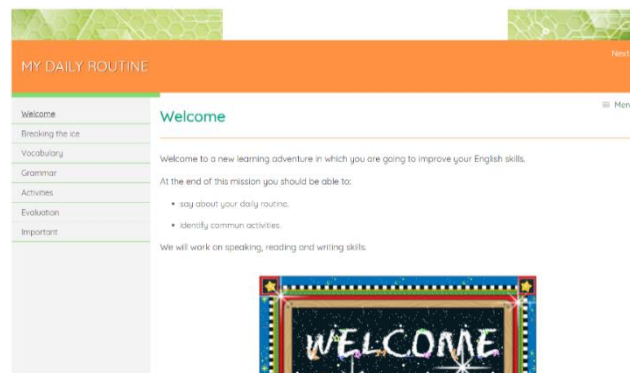


Figura 32. Pantalla de principal para el REA. Fuente: Benito y Soberanes (2022).

El REA se diseña a la luz de del constructivismo, alimentado por otras teorías pedagógicas; la estructura se basa en el modelo de adquisición de conceptos de Bruner, mismo que puntualiza que la secuencia didáctica se

debe pasar en exploración, tomando en cuenta el contenido temático los materiales necesarios, considerando los intereses de los estudiantes y vislumbrado diversificación para la consolidación del aprendizaje, guiando a los alumnos a través de distintas actividades con grados variables de dificultad (Santiváñez, 2004).

Tünnermann (2011) sostiene que la educación se debe inclinar hacia la promoción de un estudiantado que interactúe de manera creativa con la información, siendo así capaz de generar su propio conocimiento. En este caso, si bien la gramática se presenta de manera tradicional, los elementos interactivos brindan un cambio de escenario de aprendizaje, el usuario puede elegir entre distintos sujetos y verbos, apoyado por imágenes que facilitan la comprensión de los significados. Una vez que ha seleccionado los componentes principales de la oración, esta aparece escrita en pantalla (Figura 35).

Subject	+	Verb
Choose a person [Dropdown]		Choose a action [Dropdown]
	+	

Figura 33. Pantalla presentación del tema. Fuente: Benito y Soberanes (2022).

Además, la presentación del contenido está basada en el constructivismo, según está corriente, “la influencia educativa debe entenderse en términos de ayuda encaminada a mejorar los procesos vinculados a la actividad constructiva del alumno y tiene por finalidad generar la necesaria aproximación entre los significados que construye el alumno y los significados que representan los contenidos curriculares” (Serrano y Pons, 2011), lo anterior brinda al alumno la oportunidad de construir la gramática desde su individualidad e interactuando con el sistema, el cual le muestra diversas posibilidades de construcción de oraciones (Figura 36).

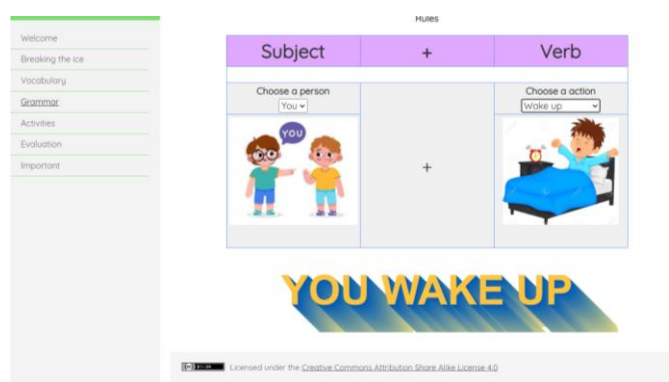


Figura 34. Pantalla presentación del tema. Fuente: Benito y Soberanes (2022).

Asimismo, el elemento de motivación es imprescindible en el proceso de enseñanza-aprendizaje, desde el conductismo se busca producir una respuesta positiva por parte de los estudiantes, por lo cual se diseñó una página de bienvenida (Figura 37), misma en la que se puede ver el tema a aprender, el listado de actividades a realizar y su objetivo.

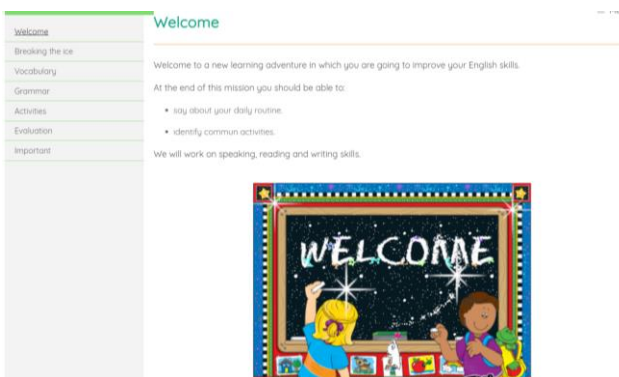


Figura 35. Pantalla de bienvenida. Fuente: Benito y Soberanes (2022).

Continuando con el enfoque, se le da un estímulo positivo final al estudiante, mostrándole un mensaje al terminar la unidad, con el cual se busca que se sienta satisfecho por haber terminado todas las actividades de aprendizaje (Figura 38). La edad resulta de relevancia, pues en este caso los niños se encuentran bajo la influencia del paradigma estímulo-respuesta.

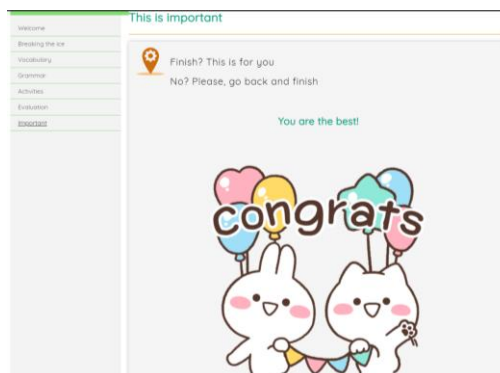


Figura 36. Pantalla de motivación para los estudiantes. Fuente: Benito y Soberanes (2022).

De igual forma, este recurso se sustenta en el conectivismo, puesto que su base es tecnológica y necesita de internet para su acceso; se sostiene que éste modifica “la naturaleza esencial del conocimiento. El conductor es más importante (sic) que el contenido que conduce” (Siemens citado por Bates, 2015, párr. 3); aunado al uso de recursos provenientes de la red (Figura 39).



Figura 37. Pantalla donde se reutilizan materiales externos. Fuente: Benito y Soberanes (2022).

Por último, la metodología de enseñanza establecida es *Direct Method* (Método directo), en la cual la exposición del alumno a la lengua es total; éste tiene como base el modelo de conocimientos previos de Ausubel, expuesto por Santivañez (2004) en tres fases: la primera, en la cual el docente presenta el tema con el uso de materiales didácticos y explica a los estudiantes una actividad inicial, el REA se aprecia con la presentación del vocabulario donde el discente elige una imagen proveniente de un grupo para ser mostrada con su forma escrita (Figura 40); en la segunda fase, el alumnado realiza una serie de actividades de aprendizaje guiadas por el docente, la correcta secuenciación en esta fase es esencial en el logro de un

aprendizaje significativo, lo cual se muestra en la presentación del contenido gramatical (Figura 36); por último, en la fase tres, el estudiante realiza una transferencia de aprendizaje a un producto original, en el que consolida el manejo de contenidos, en el REA se presenta en una lectura, la producción de un texto y la realización de un vídeo. (Figura 41).

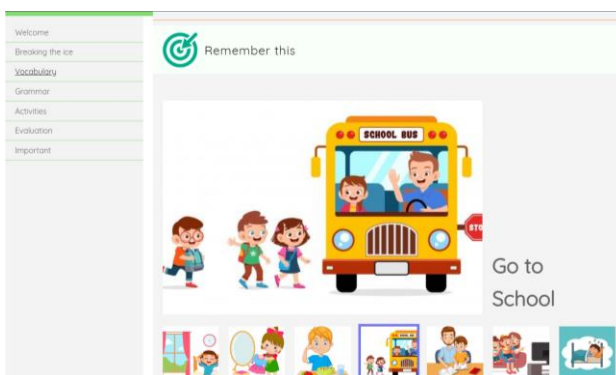


Figura 38. Pantalla donde se presenta el vocabulario. Fuente: Benito y Soberanes (2022).

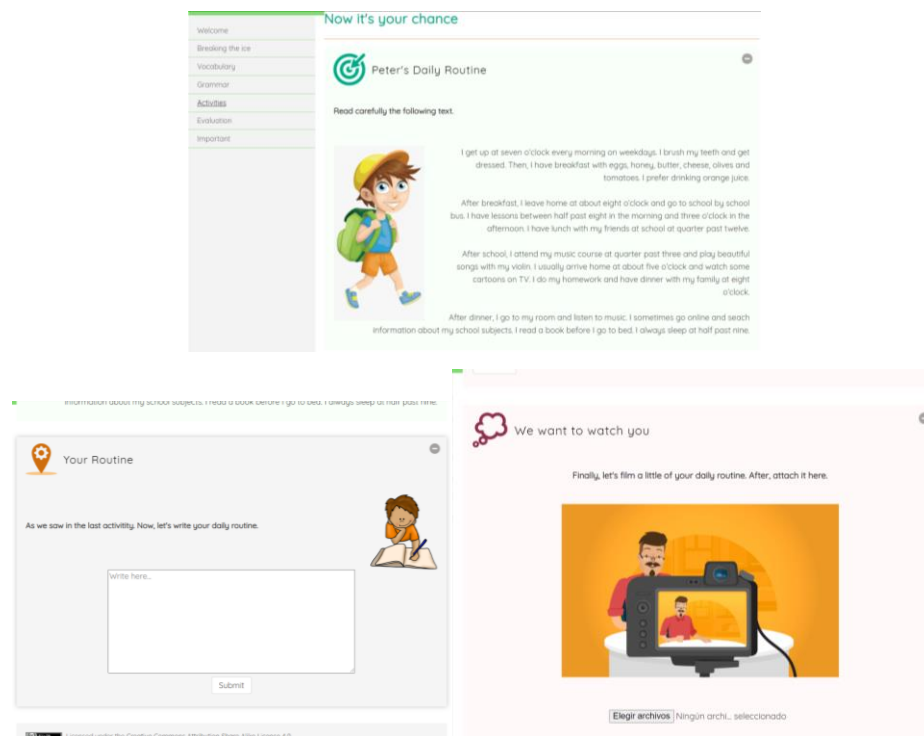
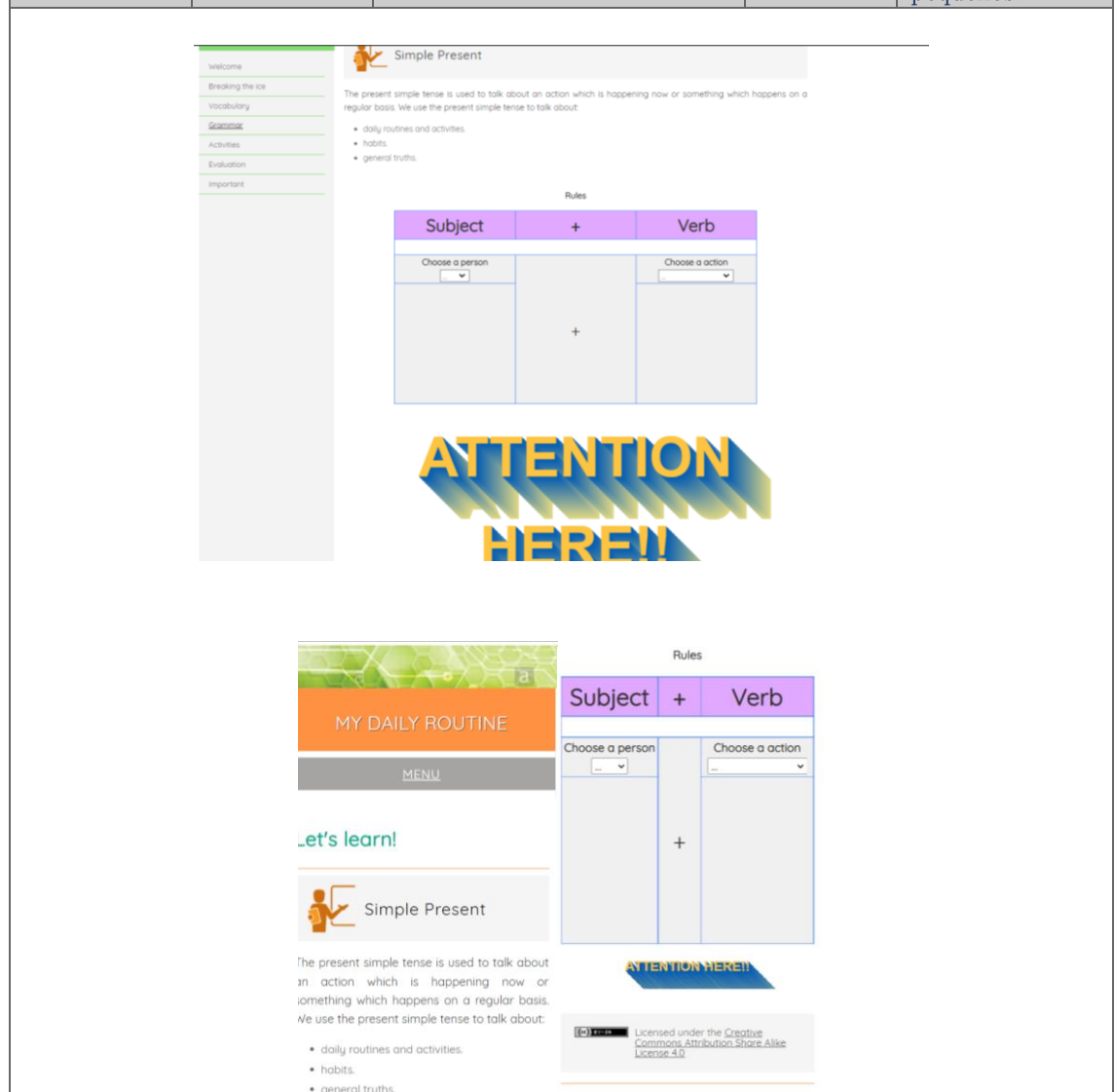


Figura 39. Pantallas para la producción propia del estudiante (lectura, escritura y oral).
Fuente: Benito y Soberanes (2022).

Posteriormente, en la etapa de pruebas, se comprueba el funcionamiento operativo del REA, con el objetivo de identificar posibles errores en programación; asimismo, determinar en qué forma lo conseguido en las fases previas incidió en el proceso enseñanza-aprendizaje.

Identificador	Prueba por realizar	Resultado esperado	Estatus	Comentario
001TS	Responsivo	El sistema se adapta correctamente a cada dispositivo electrónico.	Terminada	La resolución es clara tanto en dispositivos grandes como pequeños.



Se empleó el sistema propuesto en las sesiones asignadas para el aprendizaje del presente simple con un grupo de 15 estudiantes de segundo grado en

una escuela primaria privada en el Estado de México. Luego, se aplicaron dos instrumentos de recolección de información sobre la percepción de uso tanto de estudiantes como de docentes.

El primer instrumento empleado fue el SUS, esta herramienta se aplicó a 20 personas divididas en 15 estudiantes y cinco instructores, los resultados obtenidos se observan en la Tabla 7.

Tabla 7. Respuestas obtenidas a partir de la aplicación del SUS.

		Ítem									
		I	II	III	IV	V	VI	VII	VIII	IX	X
Estudiante	1	5	1	5	3	4	1	5	1	5	1
	2	3	3	3	2	5	1	3	3	3	1
	3	5	1	5	1	5	1	5	1	5	3
	4	4	1	5	1	4	2	4	1	4	2
	5	3	3	3	2	5	1	3	1	3	1
	6	5	1	5	2	5	1	5	2	5	2
	7	5	1	5	1	5	1	5	1	5	1
	8	5	1	3	2	4	1	4	2	5	1
	9	4	1	5	1	4	1	5	1	5	1
	10	5	1	5	1	5	1	5	1	5	1
	11	4	1	5	1	5	2	4	1	5	2
	12	5	1	5	2	4	1	5	1	5	1
	13	4	1	5	1	4	1	4	1	4	1
	14	5	1	5	2	5	1	5	1	5	2
	15	5	1	5	1	5	1	5	1	5	1
Profesor	1	5	1	5	1	4	1	5	1	4	1
	2	5	2	5	1	5	1	5	1	5	1
	3	5	1	5	2	5	1	5	1	5	1
	4	4	1	5	1	4	1	5	1	5	1
	5	5	1	5	1	5	1	5	1	4	1

Fuente: Benito y Soberanes (2022).

En este sentido, se establecen las respuestas a cada cuestionamiento mencionadas por los estudiantes:

1. *Creo que me gustaría utilizar este sistema con frecuencia:* 13.3% muestra estar neutral ante esta afirmación, 26.7% de acuerdo y 60% totalmente de acuerdo.
2. *Encontré el sistema innecesariamente complejo:* 86.7% totalmente en desacuerdo y 13.3% neutral.
3. *Pensé que el sistema era fácil de usar:* 20% está neutral mientras que 80% totalmente de acuerdo.
4. *Creo que necesitaría el apoyo de un técnico para poder utilizar este sistema:* 53.3% totalmente en desacuerdo, 40% en desacuerdo mientras 6.7% neutral.
5. *Encontré que las diversas funciones de este sistema estaban bien integradas:* 40% de acuerdo y 60% totalmente de acuerdo.
6. *Pensé que había demasiada inconsistencia en este sistema:* 86.7% totalmente en desacuerdo, por su parte 13.3% en desacuerdo.
7. *Me imagino que la mayoría de la gente aprendería a utilizar este sistema muy rápidamente:* 13.3% neutral, 26.7% de acuerdo, 60% totalmente de acuerdo.
8. *Encontré el sistema muy complicado de usar:* 80% totalmente en desacuerdo, 13.3% en desacuerdo y 6.7% neutral.
9. *Me sentí muy seguro usando el sistema:* 13.3% neutral, 13.3% de acuerdo, mientras que el 73.3% se mostró estar totalmente de acuerdo.

10. *Necesitaba aprender muchas cosas antes de empezar con este sistema:* 66.7% totalmente en desacuerdo, 26.7 en desacuerdo y 6.7% neutral.

Por otro lado, se muestra lo expresado por los profesores:

1. 20% de acuerdo y 80% totalmente de acuerdo.
2. 80% totalmente en desacuerdo y 20% en desacuerdo.
3. 100% totalmente de acuerdo.
4. 80% totalmente de acuerdo mientras que 20% en desacuerdo.
5. 40% de acuerdo y 60% totalmente de acuerdo.
6. 100% totalmente en desacuerdo.
7. 100% totalmente de acuerdo.
8. 100% totalmente en desacuerdo.
9. 40% de acuerdo, mientras que 60% se estableció estar totalmente de acuerdo.
10. 100% totalmente en desacuerdo.

Posteriormente, se aplica el procedimiento para determinar el nivel de usabilidad mediante el SUS, el cual ya fue descrito en la validación por equipos de trabajo, esto se presenta en la Tabla 8.

Tabla 8. Procedimiento SUS aplicado a las repuestas de profesores y estudiantes.

		$\Sigma(x - 1) + \Sigma(5 - y)$	$(\Sigma(x - 1) + \Sigma(5 - y)) * 2.5$	ϕ
	1	37	92.5	
	2	27	67.5	
	3	38	95	
	4	34	85	
	5	29	72.5	
	6	37	92.5	
	7	40	100	
Estudiante	8	34	85	90.3
	9	38	95	
	10	40	100	
	11	36	90	
	12	38	95	
	13	36	90	
	14	38	95	
	15	40	100	
	1	38	95	
	2	39	97.5	
Profesor	3	39	97.5	96.5
	4	38	95	
	5	39	97.5	

Fuente: Benito y Soberanes (2022).

Con base en lo obtenido, se encuentra una diferencia de 6.2 en la percepción del nivel de usabilidad entre alumnos y profesores mediante SUS; este instrumento plantea que valores a partir de 68 reflejan que es un sistema que está por arriba del promedio, siendo 100 el valor máximo (Brooke, 1996), por lo cual con la ϕ obtenida se determina que el REA tiene un valor idóneo para su uso.

Posteriormente, en la siguiente etapa se realizó una entrevista tanto a docentes como a estudiantes, esta busca recolectar rasgos cualitativos

percibidos por los usuarios durante el uso sistema. A partir de sus repuestas, se determinó lo siguiente:

- La mayoría de los alumnos puede usar la plataforma sin apoyo extra, aunque hay algunos que lo requieren.
- La presentación del vocabulario demanda de una actividad de consolidación antes de su empleo dentro de la gramática; los estudiantes suponían trabajar con el mismo antes de iniciar con el contenido gramatical, refiriendo un salto considerable en su uso en conjunto con la gramática.
- La presentación de las estructuras gramaticales resulto satisfactoria para los niños, dada su naturaleza sumamente visual e interactiva, permitiéndoles construir oraciones por sí mismos.
- Para la actividad de la lectura, los discentes perciben un vacío, pues esperan trabajar con su contenido antes de realizar productos propios.
- La navegación en la plataforma les pareció sencilla, permitiéndoles ir de una sección a otra.
- Hay discordancia entre el vocabulario trabajado al inicio y el empleado en las actividades
- Las instrucciones fueron sencillas de seguir, y dado el apoyo visual, fueron cuasi intuitivas, lo cual permitió el desarrollo de las actividades sin dificultad alguna.
- La evaluación cumple con su objetivo, brindando al docente evaluar diversos aspectos, así como habilidades comunicativas.

- La naturaleza interactiva de las actividades capta la atención de los alumnos, aunque se sugiere disminuir el número de que sean escritas.
- Luego de utilizar la plataforma, el docente considera conveniente la adición de una sección auditiva, ya sea cuando formen una oración correctamente o al concluir una sección, como parte de la motivación.
- De manera general, la plataforma posibilita al docente enseñar el tema y consolidar su aprendizaje; a los estudiantes les parece atractivo el diseño y el poder manipular contenidos les motiva.

Por último, el docente al frente del grupo realizó una evaluación cuantitativa del aprendizaje, derivada de la cantidad de aciertos obtenidos por los discentes en cada actividad y su participación en clase. Las calificaciones se asentaron en una escala del 0 al 10, donde el 20% obtuvo 10, 40% entre 9.0 y 9.9, y 40% entre 8.0 y 8.9. Esto contrastaba con las calificaciones obtenidas en otros grupos y generaciones, debido a que se presentan más recurrentemente calificaciones reprobatorias o por debajo de 8.0 y, a su vez, calificaciones de excelencia académica casi en nulas ocasiones son asentadas.

Con base en lo mencionado, las validaciones presentadas son una parte fundamental que tuvieron como objetivo evaluar la eficacia y eficiencia de los métodos o procesos propuestos para alcanzar los objetivos de aprendizaje. Esto permitió: comprobar la eficacia de la metodología, evaluar la eficiencia de la metodología, identificar limitaciones o áreas de mejora, así como, asegurar la replicabilidad y reproducibilidad.

6. CONCLUSIONES

La tecnología educativa puede proporcionar oportunidades únicas para mejorar la enseñanza y el aprendizaje, puede ayudar a satisfacer las necesidades de una población estudiantil cada vez más diversa y conectada; sin embargo, su éxito depende de la planificación cuidadosa y la implementación adecuada, así como de la capacitación de los educadores para utilizar eficazmente la tecnología en el aula. Además, es relevante tener en cuenta que la tecnología no es una solución genérica para todos los problemas educativos, y que debe complementar, en lugar de reemplazar, la educación tradicional.

Por lo que, la integración de elementos pedagógicos con elementos tecnológicos en el desarrollo de software puede ser benéfica. Al utilizar técnicas pedagógicas en el diseño de software, se puede mejorar la experiencia del usuario y la facilidad de uso de este, lo que puede aumentar la eficiencia y la productividad de los usuarios. Asimismo, la tecnología puede mejorar la eficacia de la enseñanza

y el aprendizaje al proporcionar herramientas interactivas y de colaboración para el desarrollo de habilidades y conocimientos. Naturalmente, la integración de elementos pedagógicos con tecnológicos puede ser una forma efectiva de mejorar tanto los sistemas informáticos como la educación.

En este sentido, MCAE busca garantizar software educativo de calidad que cumpla con los objetivos de aprendizaje estableciendo un proceso de desarrollo que aborde las perspectivas tanto educativas como tecnológicas, con ayuda de elementos de estas dos áreas: diseño instruccional, planeación didáctica, listado de requerimientos, UML, entre otros; con lo que se establece un equilibrio pedagógico-tecnológico que define una importancia igualitaria de ambos enfoques.

Durante el análisis comparativo, se establecieron elementos, productos y fases que se consideran relevantes para la construcción del software, los cuales no fueron encontrados de manera evidente y en su totalidad en las metodologías estudiadas; por ejemplo, ingeniería de software educativo (Galvis, 1992) ha sido frecuentemente citada debido al nivel de detalle en cada elemento, asimismo por ser pionera en su objetivo; sin embargo se omite algunas características del proceso enseñanza-aprendizaje como la ejecución, del mismo modo, se vislumbra un lenguaje técnico de ambas áreas, dificultando su entendimiento e implementación con equipos multidisciplinarios; adicionalmente, al igual que MEDESME (García et al., 2016), Metodología dinámica para el desarrollo de software educativo (Arias et al., 2015) y las otras metodologías analizadas (Tabla 1), tienen un enfoque tecnológico, caen en la problemática recurrente del desarrollo de software educativo.

La propuesta, *Metodología Conjunta para Aplicaciones Educativas*, busca resaltar la relevancia de cada uno de estos componentes al integrarlos de manera explícita y visible para el equipo de trabajo durante el proceso de desarrollo, tal como fue la integración de la planeación y secuencia didáctica como uno de los productos del desarrollo de software, entre algunas otras situaciones. Esto se vislumbró durante su aplicación en el desarrollo de un recurso educativo abierto para la enseñanza del presente simple en inglés (Benito y Soberanes, 2022); sin embargo, se considera que, para obtener una metodología que permita el desarrollo de software educativo que cumpla con los objetivos de aprendizaje, no es suficiente con una sola implementación para ser confiable, por lo cual, se presentan los siguientes procesos de validación:

- Por juicio de expertos: se envió la propuesta a 15 profesionales que tengan experiencia en tecnología educativa y que han desarrollado software educativo. Se buscó que establecieran sus impresiones con el fin de integrar componentes que consideren faltantes y eliminar los innecesarios. En este criterio de ratificación, se obtuvieron respuestas alentadoras debido a que los especialistas plantean la necesidad de una metodología funcional de desarrollo de software educativo; el cuestionario aplicado mostraba una excelente validez y constructo con base en el alfa de Cronbach. Las respuestas obtenidas estaban entre el rango neutral hacia estar de acuerdo con las afirmaciones, con esto se vislumbra que la metodología contiene e integra elementos que facilitan la construcción de software con fines educativos; sin embargo, la mayoría mencionan modificaciones o recomendaciones en pro de hacer que el proceso sea más sencillo y

entendible. Algunas de estas se aplicaron en la versión que se presenta en esta tesis, tales como la creación de un glosario, la ejemplificación de la implementación de MCAE en la elaboración de un REA, definir los roles y perfiles del equipo de trabajo de manera puntual, considerar otro tipo de instrumentos, ser los más explícito posible en las instrucciones de implementación, entre algunos otras.

- Por equipos de trabajo: se entregó MCAE a equipos que desarrollaron software educativo. De esta forma, se obtuvo los puntos de vista, tales como facilidad de aplicación, de seguimiento, de organización; así como pertinencia de los elementos propuestos e integrados para su uso durante el periodo del proyecto, entre otras cosas. Los comentarios eran muy similares a los de los expertos, solicitaron instrucciones muy específicas y ejemplos de llenado de las herramientas propuestas, así como la guía mediante un ejemplo realizado con MCAE. Por otro lado, se aplicó SUS para determinar el nivel de usabilidad de la propuesta en los equipos, el resultado fue de 82.5, mostrando una usabilidad buena, aunque es evidente que la propuesta es perfectible; se propone recopilar más opiniones de equipos y supervisar las repercusiones de estas durante el desarrollo del proyecto, debido que, al ser sólo tres validaciones, el resultado se ve limitado a pocas perspectivas, en el entendido que, a más valoraciones, mayores probabilidades de cumplir los objetivos de aprendizaje debido a la consideración de las mayores problemáticas posibles en la elaboración del sistema, con el fin de disminuir errores de implementación o de entendimiento.

De esta manera, el objetivo de esta investigación es presentar la propuesta de una metodología de desarrollo de software educativo, que sirva de guía para equipos de desarrollo que requieran la integración del contexto educativo con elementos tecnológicos y viceversa, además que identifiquen el impacto que tendrá un software elaborado bajo los parámetros de MCAE. Por lo que algunas recomendaciones que se derivan son:

- Comprender las necesidades y expectativas de los usuarios del software educativo, especialmente de los estudiantes y docentes.
- Adaptar MCAE para adecuarla a las características del proyecto y del equipo de desarrollo.
- Definir claramente los objetivos de aprendizaje y el alcance del proyecto (como en cualquier desarrollo), para evitar o disminuir el impacto negativo en el proceso y por ende en el proceso de enseñanza-aprendizaje del estudiante.
- Fomentar la colaboración y comunicación constante entre los miembros del equipo de desarrollo y usuarios del software educativo, sin menospreciar los puntos de vista, definiendo que esto puede tener un impacto positivo durante el proyecto.
- Realizar pruebas de usabilidad y de funcionalidad en todas las etapas del proceso de desarrollo, para asegurar la calidad del software educativo.
- Proponer mejoras MCAE según los resultados obtenidos y la retroalimentación de los usuarios, con el fin optimizar el proceso de desarrollo, el resultado final y lograr los objetivos de aprendizaje.

- Capacitar al equipo de desarrollo en MCAE, así como a los usuarios en el uso del software educativo.

Al seguir estas recomendaciones, se puede lograr un proceso de desarrollo de software educativo más efectivo y con resultados satisfactorios para los usuarios; así como abrir más líneas de investigación que impacten en la literatura de la Tecnología Educativa.

Por último, como forma de difundir lo generado, se hizo la publicación de dos artículos en revistas indexadas en el padrón de CONACYT y uno está en proceso de evaluación:

- "Proposal of an educational technology approach for a web system construction. Application case: teaching about the Fungi kingdom", publicado en *ie Revista de Investigación Educativa de la REDIECH*, volumen 12 del año 2021.
- "Recurso Educativo Abierto para la enseñanza del presente simple en inglés: propuesta que integra componentes pedagógicos y tecnológicos en su desarrollo", publicado en la revista *Diálogos sobre educación*, número 26, volumen 14 del año 2022.
- "Proposal for developing educational software: Joint Methodology for Educational Applications (JMEA)", el cual no ha sido publicado y continua en proceso de revisión.

REFERENCIAS

- Abreu, Y., Barrera, A., Brejjo, T. y Bonilla, I. (2018). El proceso de enseñanza-aprendizaje de los Estudios Lingüísticos: su impacto en la motivación hacia el estudio de la lengua. *Mendive*, 16(4), 610-623. <http://mendive.upr.edu.cu/index.php/MendiveUPR/article/view/1462>
- Abud, M. (2009). MeISE: Metodología de Ingeniería de Software Educativo. *Revista Internacional de Educación en Ingeniería*, 2(1), 1-9. <https://cutt.ly/5j8NOoM> (No disponible).
- Albarrán, M. (2014). *Diseño instruccional de objetos de aprendizaje. Una propuesta con base en la metodología del CATED-UNAM sobre: "importancia de las categorías y las variables en la investigación social"*. (Licenciatura) Universidad Nacional Autónoma de México.
- Adnan, N. y Ritzhaupt, A. (2018) Software Engineering Design Principles Applied to Instructional Design: What can we Learn from our Sister Discipline? *TechTrends*, 62, 77–94. <https://doi.org/10.1007/s11528-017-0238-5>
- Aparici, R. (2013). *Conectados en el ciberespacio* (1ra ed.). UNED.

- Area, M. (2009). *Introducción a la Tecnología Educativa*. Universidad de La Laguna. <https://campusvirtual.ull.es/ocw/file.php/4/ebookte.pdf>
- Arias, M., López, A. y Honmy, R. (2015). Metodología Dinámica para el Desarrollo de Software Educativo. *Virtual Educa*. 1-8. <http://espacio.uned.es/fez/view/bibliuned:1296>
- Balda, M. y Vicenzi, A. (1997). *Administrador de Flujo de Tareas y Documentos para la Especificación de Requerimientos*. Universidad Nacional de la Plata. http://sedici.unlp.edu.ar/bitstream/handle/10915/2158/Documento_completo.pdf?sequence=1&isAllowed=y
- Bates, A. (2019). *Teaching in a Digital Age* (2da ed.). Tony Bates Associates Ltd. <https://pressbooks.bccampus.ca/teachinginadigitalagev2/>
- Benigni, G. (2004). Una metodología orientada a objetos para la producción de software multimedia. *Saber*, 16(1), 26-32. <http://201.249.180.234/handle/123456789/3824>
- Benito, D. (2019). *Desarrollo de un sistema web desde el enfoque de Tecnología Educativa para difundir información sobre el reino fungi* (Maestría). Universidad Autónoma del Estado de México.
- Benito, D. y Soberanes, A. (2021). Proposal of an educational technology approach for a web system construction. Application case: teaching about the Fungi kingdom. *ie Revista de Investigación Educativa de la REDIECH*, 12(2021). https://doi.org/10.33010/ie_rie_rediech.v12i0.1126
- Benito, D. y Soberanes, A. (2022). Recurso Educativo Abierto para la enseñanza del presente simple en inglés: propuesta que integra componentes

- pedagógicos y tecnológicos en su desarrollo. *Diálogos sobre educación*, 26(14). DOI: <https://doi.org/10.32870/dse.v0i26.1130>.
- Biggs, J. y Tang, C. (2011). *Teaching for quality learning at university: What the student does* (4ta ed.). Open University Press.
- Braude, J. y Bernstein, M. (2016). *Software engineering: Modern approaches* (2da ed.). Waveland Press.
- Britannica, T. Editors of Encyclopaedia (2020). Software. *Encyclopedia Britannica*. <https://www.britannica.com/technology/software>
- Brito, M., Jara, C. y Larenas C. (2009). *El Software Educativo en el Aula* (Licenciatura). Universidad Academia de Humanismo Cristiano.
- Brooke, J. (1996). SUS: a "quick and dirty" usability scale en *Usability Evaluation in Industry*, (1ra ed.). Taylor and Francis. <https://www.taylorfrancis.com/chapters/edit/10.1201/9781498710411-35/sus-quick-dirty-usability-scale-john-brooke>
- Cabero, J. (2007). Tecnología educativa: Su evolución histórica y su conceptualización en *Tecnología educativa*. McGraw-Hill. <https://dialnet.unirioja.es/servlet/articulo?codigo=2275382>
- Cabero, J., Barroso, J., Palacios, A. y Llorente C. (2021) Evaluación de t-MOOC universitario sobre competencias digitales docentes mediante juicio de expertos según el Marco DigCompEdu. *Revista de educación a distancia*, 66(21). <https://doi.org/10.6018/red.476891>

- Cabero, J. y Llorente, M. (2013). La aplicación del juicio de experto como técnica de evaluación de las tecnologías de la información y comunicación (TIC). *Revista de Tecnología de Información y Comunicación en Educación*, 7(2). https://www.researchgate.net/publication/260750592_La_aplicacion_del_juicio_de_experto_como_tecnica_de_evaluacion_de_las_tecnologias_de_la_informacion_y_comunicacion_TIC
- Candelario-Dorta, O. (2018). El software en el proceso de enseñanza aprendizaje de la Física. *Edusol*, 18(63), 123-136. <https://dialnet.unirioja.es/servlet/articulo?codigo=6572874>
- Cao, J., Ellis, M. y Khachatryan, N. (2020). *The guide to mockups. Mockups, methods, and best practices.* UXPin. <https://www.pdf-archive.com/2016/10/14/the-guide-to-mockups/the-guide-to-mockups.pdf>
- Carballo, S. (1978). Fases del proceso Enseñanza-Aprendizaje. *Revista educación*, 2(2), 47-57. <https://revistas.ucr.ac.cr/index.php/educacion/article/view/18680>
- Carmel, E., George, J. y Nunamaker, J. (1995). Examining the process of electronic-JAD. *Journal of End User Computing*, 7(1), 13-22. <http://doi.org/10.4018/joec.1995010102>
- Caro, M. y Toscano, R. (2009). MODESEC: Modelo para el desarrollo de software educativo basado en competencias. *Nuevas Ideas En Informática Educativa*, 5, 188-200. <http://www.tise.cl/volumen5/TISE2009/Documento23.pdf>
- Comunidades Digitales para el Aprendizaje en Educación Superior. (2015). *Modelo de diseño instruccional.* Universidad de Colima.

<https://www.codaes.mx/content/repositoriocdg/000090/Modelo-DI-CODAES.pdf>

Contreras, J. (1994). *Enseñanza, curriculum y profesorado* (2da ed.). Ediciones Akal.

Corral, R. (2001). El concepto de zona de desarrollo próximo: una interpretación. *Revista Cubana de Psicología*, 18(1), 72-76.
<http://pepsic.bvsalud.org/pdf/rcp/v18n1/09.pdf>

Dede, C. (1996). The evolution of distance education: Emerging technologies and distributed learning. *American Journal of Distance Education*, 10(2), 4-36.
<https://doi.org/10.1080/08923649609526919>

Delavant, M. y Fernández, R. (2008). *Educación y tecnología: un binomio excepcional* (1ra ed.). Grupo editor K.

Delors, J. (1996). Los cuatro pilares de la educación en *La educación encierra un tesoro. Informe a la UNESCO de la Comisión internacional sobre la educación para el siglo XXI*. Santillana/UNESCO.

De Luque, A. (2006). *Aprendizaje centrado en el alumno: metodología para una escuela abierta* (1ra ed.). Narcea.

Dimuro, J. (2014). *Guía de usabilidad web* [Ebook]. <https://docplayer.es/6384823-Guia-de-usabilidad-web-juan-j-dimuro-2014-correccion-carolina-condado.html>

- Dirección General de Desarrollo Curricular (DGDC). (2012). *Las estrategias y los instrumentos de evaluación desde el enfoque formativo* (1ra ed.). Secretaría de Educación Pública.
- Dueñas, D., Toscano, R., Gómez, A. y Caro, M. (2017). Sinopsis de metodologías y modelos de software educativo. *Acta Scientiæ Informaticæ*, 1(1). 70-74.
<https://revistas.unicordoba.edu.co/index.php/asinf/article/view/1164>
- Duignan, B., Fumerton, R., Quinton, A. y Quinton, B. (2020). *Empiricism*. *Encyclopedia Britannica*. <https://www.britannica.com/topic/empiricism>
- El Assafiri, Y., Nogueira, D., Ramírez, K. y Díaz, A. (abril, 2017). *Mapas de conocimiento. Breve estudio del estado del arte*. Ponencia presentada en VIII Convención Científica Internacional “Universidad Integrada e Innovadora”. Varadero, Cuba.
https://www.researchgate.net/publication/321496601_Mapas_de_conocimiento_Breve_estudio_del_estado_del_arte
- Espinoza, A. (2013). *Manual para elegir una metodología de desarrollo de software dentro de un proyecto informático* (Licenciatura). Universidad de Piura.
- Flores, I. (1997). *Una aplicación de fundamentos teóricos-conceptuales de los procesos de enseñanza-aprendizaje al diseño de un diplomado en computación para adultos* (Maestría). Universidad Autónoma de Nuevo León.

- Flores, J., Ávila, J., Rojas, C., Sáez, F., Acosta, R., y Díaz, C. (2017). *Estrategias didácticas para el aprendizaje significativo en contextos universitarios*. Concepción, Universidad de Concepción.
- Gagne, R. y Briggs, L. (1976). *La planificación de la enseñanza. Sus principios* (1ra ed.). Trillas.
- Galindo, M. (2015). *Efectos del software educativo en el desarrollo de la capacidad de resolución de problemas matemáticos en estudiantes de 5 años IEI. n° 507 Canta* (Maestría). Universidad Peruana Cayetano Heredia.
- Galvis, A. (1992). *Ingeniería de software educativo* (1ra ed.). Ediciones Uniandes.
- García, E., González, J., Pérez, M., García, J. y Valdés, V. (2002). *¿Existe una situación de crisis del software educativo?* Memoria del VI Congreso Iberoamericano de Informática Educativa. Universidade de Vigo. https://www.researchgate.net/publication/242579286_Existencia_de_crisis_del_software_educativo
- García, E., Vite, O., Navarrate, M., García, M. y Torres, V. (2016). Metodología para el desarrollo de software multimedia educativo MEDESME. *CPU-e. Revista de Investigación Educativa*, (23), 216-226. <https://cpue.uv.mx/index.php/cpue/article/view/2169>
- García-Peñalvo, F. (2018). *Ingeniería del Software I - curso 2017-2018*. Grupo GRIAL.

- George, D., y Mallery, P. (2003). *SPSS for Windows step by step: A simple guide and reference. 11.0 update* (4ta ed.). Allyn & Bacon.
- Gimeno, J. (2008). *Comprender y transformar la enseñanza* (12ma ed.). Ediciones Morata, S.L.
- Gómez, M. (2011). *Notas del curso: Análisis de requerimientos*. Universidad Autónoma Metropolitana.
<http://ilitia.cua.uam.mx:8080/jspui/bitstream/123456789/168/1/619%20-%20G%c3%b3mez%20Fuentes%20Mar%c3%ada%20del%20Carmen.pdf>
- Gómez, M., Cervantes, J. y González, P. (2019). *Fundamentos de ingeniería de software*. Universidad Autónoma Metropolitana.
http://www.cua.uam.mx/pdfs/conoce/libroselec/Fundamentos_Ing_SW-VF.pdf
- Guerrero, I., Arévalo, D., González, E., Ramírez, Y. y Benítez, Y. (2016). Efectividad del software educativo sobre los defectos radiográficos en la asignatura de Imagenología Estomatológica. *Correo Científico Médico*, 20(2), 237-249. <https://www.medigraphic.com/cgi-bin/new/resumen.cgi?IDARTICULO=66576>
- Gustafson, K. y Branch, R. (2002). *Survey of instructional development models* (4ta ed.). ERIC Clearinghouse on Information & Technology.
- Heller, E. (2004). *Psicología del color* (1ra ed.). Editorial Gustavo Gili.
- Heredia, Y. y Sánchez, A. (2013). *Teorías del aprendizaje en el contexto educativo* [Ebook] (1ra ed.). Editorial Digital, Tecnológico de Monterrey.
<http://prod77ms.itesm.mx/podcast/EDTM/P231.pdf>

- Hernández-Sampieri, R., Fernández, C. y Baptista, M. (2014). *Metodología de la investigación* (6ta ed.). McGraw-Hill Interamericana.
- Hipatia, B. (2014). *Diseño y elaboración de un blog educativo como recurso complementario para el aprendizaje de estudios sociales, de acuerdo a la actualización y fortalecimiento curricular, en el octavo año de educación general básica del centro educativo "Teresa Samaniego" de la ciudad de Gualaquiza* (Licenciatura). Universidad Politécnica Salesiana.
- Januszewski, A. y Molenda, M. (2013). *Educational technology: A definition with commentary* (2da ed.). Routledge.
- Kendall, K. y Kendall, J. (2011). *Análisis y diseño de sistemas*. (8va ed.). Prentice Hall, Pearson.
- Lima, B. (1977). *Contribución a la Metodología de Trabajo Social* (2da ed.). Universidad Central de Venezuela.
- Liviu, M. (2014). Comparative study on software development methodologies. *Database Systems Journal*, 5(3), 37-55.
https://www.dbjournal.ro/archive/17/17_4.pdf
- López, M. y Rojano, J. (2007). *Avances en Tecnologías de la Información CNCIIC 2007* [Ebook] (1ra ed.). Universidad Autónoma Metropolitana, Asociación Nacional de Instituciones de Educación Informática.
https://www.researchgate.net/profile/Luis_Sanchez_Medel/publication/305617599_Sistema_por_vision_maquina_para_el_control_de_una_silla_de_ruedas_operada_por_reconocimiento_de_imagenes_de_instrucciones_gestuales/links/5795accd08aec89db7b6a131/Sistema-por-vision-maquina-

[para-el-control-de-una-silla-de-ruedas-operada-por-reconocimiento-de-
imagenes-de-instrucciones-gestuales.pdf](#)

Marcano, I. y Benigni, G. (2014). Análisis de alternativas metodológicas para el desarrollo de software educativo. *SABER, Revista Multidisciplinaria del Consejo de Investigación de la Universidad de Oriente*, 26(3), 297-304.

<https://www.redalyc.org/articulo.oa?id=427739473009>

Martínez, F. (2019). *El nuevo oficio del investigador educativo. Una introducción metodológica*. Universidad Autónoma de Aguascalientes.

Medina, A. y Mata, F. (2009). *Didáctica general* (2da ed.). Prentice Hall, Pearson.

<https://ceum-morelos.edu.mx/libros/didacticageneral.pdf>

Molenda, M., Reigeluth, C. y Nelson, L. (2003). Instructional Design en *Encyclopedia of Cognitive Science, Vol. 2*. Nature Publishing Group.

<https://doi.org/10.1002/0470018860.s00683>

Morales, F. y Vera, M. (2007). Eficiencia de un software educativo para dinamizar la enseñanza del cálculo integral. *Acción Pedagógica*, 16(1), 204-211.

<https://dialnet.unirioja.es/servlet/articulo?codigo=2968785>

Morán, P. (2004). La docencia como recreación y construcción del conocimiento Sentido pedagógico de la investigación en el aula. *Perfiles educativos*, 26(105-106), 41-72.

[http://www.scielo.org.mx/scielo.php?script=sci_arttext&pid=S0185-
26982004000100003&lng=es&tlng=es](http://www.scielo.org.mx/scielo.php?script=sci_arttext&pid=S0185-26982004000100003&lng=es&tlng=es)

- Morrison, G., Ross, S., Kalman, H. y Kemp, J. (2013). *Designing effective instruction* (7ma ed.). John Wiley & Sons.
- Mujica, R. (2020). Fundamentos de la Tecnología Educativa. *Revista Tecnológica-Educativa Docentes* 2.0, 8(1),15-20.
<https://dialnet.unirioja.es/servlet/articulo?codigo=8641877>
- Murcia, M. (2020). *Diseño instruccional para profes: Guía para la innovación educativa con TIC* (1.a ed.). Ediciones USTA.
<https://repository.usta.edu.co/bitstream/handle/11634/22541/Obracompleta.Coleccionmodular.2020Murciamiguel.pdf?sequence=1>
- Nohlen, D. (2020). Capítulo tercero: El método comparativo en *Antologías para el estudio y la enseñanza de la ciencia política. Volumen III: La metodología de la ciencia política*. Biblioteca Jurídica Virtual.
- Ortiz, D. (2015). El constructivismo como teoría y método de enseñanza. *Sophia, Colección de Filosofía de la Educación*, (19), 93-110.
<https://doi.org/10.17163/soph.n19.2015.04>
- Ortiz, G. (2014). El color, un facilitador didáctico. *Procesos psicológicos y sociales*, 10(1), 1-24. <https://www.uv.mx/psicologia/files/2014/09/El-color-un-facilitador-didactico.pdf>
- Ormrod, J. (2016). *Essentials of educational psychology: Big ideas to guide effective teaching* (4ta ed.). Pearson.
- Paladines, R. (2016). Incidencia de la autonomía educativa y el aprendizaje de los estudiantes del colegio Juan Montalvo del cantón Manta. *Dominio De Las Ciencias*, 2(4), 137-152. <https://doi.org/10.23857/dc.v2i4.221>

- Peersman, G. (2014). *Evaluative Criteria, Methodological Briefs: Impact Evaluation 3*. UNICEF Office of Research.
- Pérez, H. (2021). Discurso general del encuadre pedagógico / Entrevistada por Daniel Benito Moran. Canal de Microsoft Teams.
- Pérez, J. (2014). Empleo del software educativo y su eficiencia en el rendimiento académico del cálculo integral en la Universidad Peruana Unión, filial Tarapoto. *Apuntes Universitarios*. 1(1), 43-56.
https://revistas.upeu.edu.pe/index.php/ra_universitarios/article/view/25
- Pineda, L. (Coord.). (2017). *La Computación en México por especialidades académicas* (1ra ed.). Academia Mexicana de Computación, A. C.
- Pfleeger, S. (2002). *Ingeniería de software. Teoría y práctica* (1ra ed.). Pearson Education.
- Plomp, T. (2007). Educational Design Research: An Introduction en *An introduction to educational design research*, 9-35. SLO • Netherlands institute for curriculum development.
- Polo, M. (2003). Aproximación a un Modelo de Diseño: ADITE. *Docencia Universitaria*, 1(4), 67-83.
http://www.ucv.ve/fileadmin/user_upload/sadpro/Documentos/docencia_vol4_n1_2003/7_art.4Marina_Polo.pdf
- Pressman, R. (2014). *Ingeniería de software: Un enfoque práctico* (7^a ed.). Editorial McGraw Hill.

- Quintero, H., Portillo, L., Luque, R. y González, M. (2005). Desarrollo de software educativo: una propuesta metodológica. *TeloS*, 7(3), 383-396.
<https://www.redalyc.org/articulo.oa?id=99318837004>.
- Ríos, P. (2001). Concepción del Software Educativo desde la perspectiva pedagógica. *Quaderns Digitals*, (24).
https://www.researchgate.net/publication/299592595_CONCEPCION_D_EL_SOFTWARE_EDUCATIVO_DESDE_LA_PERSPECTIVA_PEDAGOGICA
- Rodríguez, E. (septiembre, 2012). Conceptos básicos de ingeniería de software. Ponencia presentada en CINVESTAV-Tamaulipas.
<https://www.tamps.cinvestav.mx/~ertello/swe/sesion01.pdf>
- Ruiz, D. y Pachano, L. (2006). El Nihilismo en la escuela contemporánea. *Educere*, 10(32), 83-90.
http://ve.scielo.org/scielo.php?script=sci_arttext&pid=S1316-49102006000100012&lng=es&tlng=es
- Ruiz, J. y Aguilera, O. (2007). Importancia de la Ingeniería de Software en la producción de software. *Ciencias Holguín*. 8(2), 1-8.
<https://www.redalyc.org/pdf/1815/181517923005.pdf>
- Sánchez, J. (1993). *Informática Educativa* (1ra ed.). Editorial Universitaria.
- Sánchez, L. M. (2002). *Software educativo: teoría y práctica*. McGraw-Hill Interamericana de España, S.A.U.

- Santiviáñez, V. (2004). La didáctica, el constructivismo y su aplicación en el aula. *Cultura*, 18(18), 137-148.
https://www.revistacultura.com.pe/revistas/RCU_18_1_la-didactica-el-constructivismo-y-su-aplicacion-en-el-aula.pdf
- Serna, E., Castro, C. y Botero T. (2012). SEDLO: software engineering for developing learning objects. *EATIS '12: Proceedings of the 6th Euro American Conference on Telematics and Information Systems*. 2012, 347-353. <https://dl.acm.org/doi/10.1145/2261605.2261658>
- Serrano, J. y Pons, R. (2011). El Constructivismo hoy: enfoques constructivistas en educación. *REDIE. Revista Electrónica de Investigación Educativa*, 13(1), 1-27. <https://www.redalyc.org/articulo.oa?id=15519374001>
- Schunck, D. (2012). *Teorías del aprendizaje* (6ta ed.). Pearson educación.
- Sharma, R. y Mohan, K. (2004). *Business Correspondence and Report Writing: a Practical Approach to Business & Technical Communication* (3ra ed.). McGraw-Hill.
- Skinner, B. (1968). *The technology of teaching*. Appleton-Century-Crofts.
- Slavin, R. E. (2018). *Educational psychology: Theory and practice* (12va ed.). Pearson.
- Smith, P. y Ragan, T. (1999). *Instructional Design* (3ra ed.). John Wiley & Sons Inc.
- Soler, B., Villacañas, L. y Pich, E. (2013). Creating and Implementing a Didactic Sequence as an Educational Strategy for Foreign Language Teaching. *Íkala*,

revista de lenguaje y cultura, 18(3), 31-43.
<https://doi.org/10.17533/udea.ikala.14099>

Sommerville, I. (2011). *Software Engineering* (9na ed.). Pearson.

Subramanian, N. y Chung, L. (2001). *Metrics for Software Adaptability*.
Department of Computer Science University of Texas.
<https://personal.utdallas.edu/~chung/ftp/sqm.pdf>

Tafur, C., Coba, J., Martínez, S., Rómulo, L. y Cruz, S. (2016). Software interactivo para el aprendizaje de las tablas de multiplicar de los números enteros positivos. *Encuentro Distrital de Educación Matemática*. 3, 344-350.
<http://funes.uniandes.edu.co/10187/1/Tafur2016Software.pdf>

Tovar, L., Bohórquez, J., y Puello, P. (2014). Propuesta metodológica para la construcción de objetos virtuales de aprendizaje basados en realidad aumentada. *Formación universitaria*, 7(2), 11-20.
<https://dx.doi.org/10.4067/S0718-50062014000200003>

Tsui, F., Karam, O. y Bernal, B. (2014). *Essentials of software Engineering* (3ra ed.). Jonas & Bartlett Learning books.

Tünnermann, C. (2011) El constructivismo y el aprendizaje de los estudiantes. *Universidades*, (48), 21-32.
<https://www.redalyc.org/articulo.oa?id=37319199005>

UNESCO. (1998) *Declaración Mundial sobre la Educación Superior en el siglo XXI: Visión y Acción*. Conferencia Mundial sobre la Educación Superior en

el siglo XXI. París, Francia.

https://unesdoc.unesco.org/ark:/48223/pf0000113878_spa

Urías, M., Torres, G., Valdés, A. y Antelo, M. (2015). Teorías que sustentan la Tecnología Educativa en *Aportes y reflexiones sobre la educación mediada por tecnologías*, 38-50. TABOOK.

https://www.researchgate.net/publication/290447950_Teorias_que_sustentan_la_Tecnologia_Educativa

Valbuena, W. y Quintana, M. (2014). Diseñando diseñadores. Vygostky, la teoría de la elaboración y las TIC en la didáctica del diseño. *Horizontes Pedagógicos*, 15(1).

<https://horizontespedagogicos.iberro.edu.co/article/view/414>

Valdez, F. (2012). *Teorías educativas y su relación con las tecnologías de la información y de la comunicación (TIC)*. (Ponencia inédita, XVII Congreso Internacional del Contaduría, Administración e Informática). Universidad Nacional Autónoma de México.

<https://clea.edu.mx/biblioteca/files/original/88d9d6779a5aab4815e05f82a90a4c7d.pdf>

Vania, K. y Rashidi, Y. (2016). Tales of Software Updates: The process of updating software. *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems*, 3215–3226. Association for Computing Machinery. <https://doi.org/10.1145/2858036.2858303>

Whitten, J. y Bentley, L. (2008). *Análisis de sistemas: diseño y métodos* (7ma ed.). McGraw-Hill.

Willis, J. (2008). *Qualitative research methods in education and educational technology*. Information Age Publishing Inc.

Yáñez, P. (2016). El proceso de aprendizaje: fases y elementos fundamentales. *Revista San Gregorio*, (11), 70-81.
<https://www.researchgate.net/publication/313843119> El proceso de aprendizaje fases y elementos fundamentales

Zubillaga, R., Muñoz, M. y López, F. (2016). *Influencia de la psicología del color en la educación superior*. Presentación, León, Guanajuato.

ANEXOS

Anexo I: Glosario

Término	Definición
Actores del proceso enseñanza-aprendizaje	<p>Personas o entidades que tienen un papel activo en la transmisión del conocimiento y adquisición de habilidades o competencias. Los actores del proceso de enseñanza-aprendizaje pueden dividirse en tres categorías (Slavin, 2018):</p> <ul style="list-style-type: none"> • Personas directamente involucradas: incluyen el profesor, estudiantes y padres. • Personas indirectamente involucradas: incluyen los administradores escolares, diseñadores de instruccionales y miembros de la comunidad. • Personas externas al proceso: incluyen investigadores o políticos que influyen en las políticas educativas y la financiación.
Bosquejos de diseño	Técnica apoyada de UML, que fungirá como borradores no formales de este, entendibles para el equipo de trabajo.
Diseño instruccional	Proceso de planificar, crear, implementar y evaluar entornos de aprendizaje efectivos y eficientes; incluye el diseño de materiales didácticos, selección de medios o tecnologías, y organización del ambiente de aprendizaje (Morrison <i>et al.</i> , 2013). Esta disciplina se enfoca en cómo se puede diseñar el proceso de enseñanza-aprendizaje para optimizar la adquisición de conocimientos y habilidades por parte de los estudiantes.
Experto educativo	Miembro del equipo de trabajo especialista en cuestiones educativas, particularmente en diseño instruccional.
Experto disciplinar	Miembro del equipo de trabajo que auxilia al experto educativo a desarrollar los temas y contenidos que se necesitan para lograr los objetivos de aprendizaje mediante el sistema elaborado.
Experto tecnológico	Miembro del equipo de trabajo especialista en cuestiones tecnológicas, particularmente en el desarrollo de software.
Listado de requerimientos	Lista detallada y precisa de lo que se espera que haga el software y de cómo debe hacerlo. Se utiliza para identificar y describir las funciones o características que debe tener el software para satisfacer el objetivo de aprendizaje (Pressman, 2014).
Metodología	Se refiere al estudio de los métodos como objeto de conocimiento. Es la teoría de los métodos que ordena las operaciones cognoscitivas y prácticas, en la acción racional profesional. Define cómo establecer los métodos, instrumentos y características adecuadas para la realización del software educativo (Lima, 1977).
Objetivo de aprendizaje	Descripción clara y específica de lo que se espera que el estudiante aprenda o sea capaz de hacer como resultado del proceso educativo. Este objetivo debe ser medible y evaluable (Biggs y Tang, 2011).

Pedagogía	Estudio de la educación y la enseñanza, que incluye la teoría, práctica e investigación sobre el proceso de enseñanza y aprendizaje. Se enfoca en cómo se puede diseñar y desarrollar entornos de aprendizaje efectivos, además de cómo se pueden mejorar los resultados educativos de los estudiantes (Ormrod, 2016).
Planeación didáctica	Proceso mediante el cual los docentes planifican y diseñan sus cursos o lecciones, estableciendo objetivos de aprendizaje, actividades de enseñanza-aprendizaje, evaluación del aprendizaje y recursos necesarios para lograr los objetivos de aprendizaje. Se basa en la selección y organización de contenidos curriculares, elección de metodologías de enseñanza, selección de recursos y materiales didácticos (Gagne y Briggs, 1976).
Proceso enseñanza-aprendizaje	Conjunto de acciones o estrategias que se utilizan para transmitir conocimientos y habilidades de un individuo (el profesor o facilitador) a otro (el estudiante o aprendiz). Este proceso implica una serie de pasos, como la planificación de la enseñanza, la presentación de los materiales de aprendizaje, la interacción entre el profesor y los estudiantes, la retroalimentación y la evaluación del aprendizaje (Ormrod, 2016).
Pruebas de campo	Pruebas del funcionamiento del software en despliegue, los usuarios finales o similares experimentan con los resultados de la implementación para encontrar posibles errores de funcionamiento o codificación.
Pruebas de escritorio	Pruebas del funcionamiento del software in situ, el equipo de trabajo experimenta con los resultados de la implementación para encontrar posibles errores de funcionamiento o codificación.
Pruebas in situ	Pruebas que se realiza en el lugar donde se desarrolla el software, es decir, en el mismo contexto o entorno en el que se está creando el producto.
Representación tecnoeducativa	Instrumento que sirve para transformar los requerimientos tecnológicos y educativos, en descripciones específicas y detalladas de lo que se implementará en el sistema a construir.
Software educativo	Conjunto de programas, recursos y herramientas digitales que se diseñan para apoyar el proceso de enseñanza y aprendizaje (Dede, 1996).
Tecnología educativa	Uso de la tecnología para mejorar la enseñanza y el aprendizaje, incluyendo el diseño, desarrollo y evaluación de entornos o recursos de aprendizaje. Esta disciplina se enfoca en cómo se pueden utilizar las tecnologías para crear soluciones de enseñanza o aprendizaje efectivas y relevantes (Januszewski y Molenda, 2013).
Tecnopedagógico	Palabra utilizada para establecer la compaginación del área tecnológica con la educativa mediante el uso de técnicas de estas creando y permeando productos con los aspectos más relevantes de las mencionadas.

Anexo II: SUS (Escala de Usabilidad de Sistemas)

System Usability Scale (SUS)

	Strongly Disagree	Strongly Agree
I think that I would like to use this product frequently.		
I found the product unnecessarily complex.		
I thought this product was easy to use.		
I think that I would need the support of a technical person to be able to use this product.		
I found the various functions in this product were well integrated.		
I thought there was too much inconsistency in this product		
I would imagine that most people would learn to use this product very quickly.		
I found this product very awkward to use.		
I felt very confident using this product.		
I needed to learn a lot of things before I could get going with this product.		

Anexo III: Cuestionario aplicado a los especialistas en tecnología educativa

Metodología Conjunta para Aplicaciones Educativas (MCAE)

Con la finalidad de conocer las opiniones de expertos en Tecnología Educativa sobre la propuesta de desarrollo de software educativo, se generó el siguiente cuestionario. Las respuestas proporcionadas se analizarán con fines académicos.

***Obligatorio**

1. ¿Ha desarrollado software educativo? *

Marca solo un óvalo.

- Sí
 No

2. ¿En qué nivel educativo ha desarrollado software educativo? *

Selecciona todos los que correspondan.

- Preescolar
 Nivel básico
 Nivel medio superior
 Nivel superior
 Posgrado
 Ninguno

3. Años de experiencia en Tecnología Educativa *

Marca solo un óvalo.

- Menos de 1
 Entre 1 y 2
 Entre 3 y 5
 Más de 5

4. ¿Qué rol ha desempeñado en el desarrollo de software educativo? *

Preguntas sobre la metodología

A continuación, se presentan cuestionamientos con el fin evaluar la metodología propuesta.

5. MCAE es clara para su aplicación durante el desarrollo de software educativo. *

Marca solo un óvalo.

- Muy en desacuerdo
 En desacuerdo
 Neutral
 De acuerdo
 Muy de acuerdo

6. MCAE cumple con los métodos suficientes para el desarrollo de software educativo. *

Marca solo un óvalo.

- Muy en desacuerdo
 En desacuerdo
 Neutral
 De acuerdo
 Muy de acuerdo

7. MCAE cumple con las técnicas suficientes para el desarrollo de software educativo. *

Marca solo un óvalo.

- Muy en desacuerdo
 En desacuerdo
 Neutral
 De acuerdo
 Muy de acuerdo

8. MCAE cumple con las estrategias suficientes para el desarrollo de software educativo. *

Marca solo un óvalo.

- Muy en desacuerdo
 En desacuerdo
 Neutral
 De acuerdo
 Muy de acuerdo

9. MCAE cumple con los instrumentos suficientes para el desarrollo de software educativo. *

Marca solo un óvalo.

- Muy en desacuerdo
 En desacuerdo
 Neutral
 De acuerdo
 Muy de acuerdo

10. MCAE contiene los métodos adecuados para el desarrollo de software educativo. *

Marca solo un óvalo.

- Muy en desacuerdo
 En desacuerdo
 Neutral
 De acuerdo
 Muy de acuerdo

11. Los métodos son claros para su aplicación durante el desarrollo de software educativo. *

Marca solo un óvalo.

- Muy en desacuerdo
 En desacuerdo
 Neutral
 De acuerdo
 Muy de acuerdo

12. Los métodos contienen los elementos suficientes para el desarrollo de software educativo. *

Marca solo un óvalo.

- Muy en desacuerdo
 En desacuerdo
 Neutral
 De acuerdo
 Muy de acuerdo

13. Las técnicas son claras para su aplicación durante el desarrollo de software educativo. *

Marca solo un óvalo.

- Muy en desacuerdo
- En desacuerdo
- Neutral
- De acuerdo
- Muy de acuerdo

14. Las técnicas contienen los elementos suficientes para el desarrollo de software educativo. *

Marca solo un óvalo.

- Muy en desacuerdo
- En desacuerdo
- Neutral
- De acuerdo
- Muy de acuerdo

15. Las estrategias son claras para su aplicación durante el desarrollo de software educativo. *

Marca solo un óvalo.

- Muy en desacuerdo
- En desacuerdo
- Neutral
- De acuerdo
- Muy de acuerdo

16. Las estrategias contienen los elementos suficientes para el desarrollo de software educativo. *

Marca solo un óvalo.

- Muy en desacuerdo
 En desacuerdo
 Neutral
 De acuerdo
 Muy de acuerdo

17. Los instrumentos son claros para su aplicación durante el desarrollo de software educativo. *

Marca solo un óvalo.

- Muy en desacuerdo
 En desacuerdo
 Neutral
 De acuerdo
 Muy de acuerdo

18. Los instrumentos contienen los elementos suficientes para el desarrollo de software educativo. *

Marca solo un óvalo.

- Muy en desacuerdo
 En desacuerdo
 Neutral
 De acuerdo
 Muy de acuerdo

19. Observaciones generales de la fase análisis tecnopedagógico *

20. Observaciones generales de la fase diseño tecnopedagógico *

21. Observaciones generales de la fase de implementación *

22. Observaciones generales de la fase de pruebas *

23. Observaciones generales de la fase de mantenimiento funcional y educativo *

24. Observaciones generales de MCAE *

Este contenido no ha sido creado ni aprobado por Google.

Google Formularios

Anexo IV: Cuestionario para obtener la usabilidad percibida de MCAE por parte de los equipos de desarrolladores.

Metodología Conjunta para Aplicaciones Educativas (MCAE) ED

Con la finalidad de conocer las opiniones de equipos de desarrollo de software educativo sobre la propuesta, se generó el siguiente cuestionario. Responde las preguntas en conjunto del grupo de desarrollo. Las respuestas proporcionadas se analizarán con fines académicos.

***Obligatorio**

1. *Creo que me gustaría utilizar esta metodología con frecuencia **

Marca solo un óvalo.

- Totalmente en desacuerdo
 En desacuerdo
 Ni de acuerdo ni en desacuerdo
 De acuerdo
 Totalmente de acuerdo

2. *Encontré la metodología innecesariamente compleja **

Marca solo un óvalo.

- Totalmente en desacuerdo
 En desacuerdo
 Ni de acuerdo ni en desacuerdo
 De acuerdo
 Totalmente de acuerdo

3. *Pensé que la metodología era fácil de usar* *

Marca solo un óvalo.

- Totalmente en desacuerdo
 En desacuerdo
 Ni de acuerdo ni en desacuerdo
 De acuerdo
 Totalmente de acuerdo

4. *Creo que necesitaría el apoyo de un especialista para poder utilizar esta metodología* *

Marca solo un óvalo.

- Totalmente en desacuerdo
 En desacuerdo
 Ni de acuerdo ni en desacuerdo
 De acuerdo
 Totalmente de acuerdo

5. *Encontré que los diversos elementos de esta metodología estaban bien integrados* *

Marca solo un óvalo.

- Totalmente en desacuerdo
 En desacuerdo
 Ni de acuerdo ni en desacuerdo
 De acuerdo
 Totalmente de acuerdo

6. *Pensé que habían demasiadas inconsistencias en esta metodología **

Marca solo un óvalo.

- Totalmente en desacuerdo
- En desacuerdo
- Ni de acuerdo ni en desacuerdo
- De acuerdo
- Totalmente de acuerdo

7. *Me imagino que la mayoría de la gente aprendería a utilizar esta metodología muy **
rápidamente

Marca solo un óvalo.

- Totalmente en desacuerdo
- En desacuerdo
- Ni de acuerdo ni en desacuerdo
- De acuerdo
- Totalmente de acuerdo

8. *Encontré la metodología muy complicada de aplicar **

Marca solo un óvalo.

- Totalmente en desacuerdo
- En desacuerdo
- Ni de acuerdo ni en desacuerdo
- De acuerdo
- Totalmente de acuerdo

9. *Me sentí muy seguro usando la metodología **

Marca solo un óvalo.

- Totalmente en desacuerdo
- En desacuerdo
- Ni de acuerdo ni en desacuerdo
- De acuerdo
- Totalmente de acuerdo

10. *Necesitaba aprender muchas cosas antes de empezar a usar esta metodología **

Marca solo un óvalo.

- Totalmente en desacuerdo
- En desacuerdo
- Ni de acuerdo ni en desacuerdo
- De acuerdo
- Totalmente de acuerdo

11. *Recomendaciones, ventajas, desventajas u observaciones generales de la fase de análisis tecnopedagógico **

12. Recomendaciones, ventajas, desventajas u observaciones generales de la fase *
diseño tecnopedagógico

13. Recomendaciones, ventajas, desventajas u observaciones generales de la fase *
implementación

14. Recomendaciones, ventajas, desventajas u observaciones generales de la fase *
pruebas

15. Recomendaciones, ventajas, desventajas u observaciones generales de la fase mantenimiento funcional y educativo *

16. Recomendaciones, ventajas, desventajas u observaciones generales de MCAE *

Este contenido no ha sido creado ni aprobado por Google.

Google Formularios